

Math Notes

Linear Algebra and Optimisation

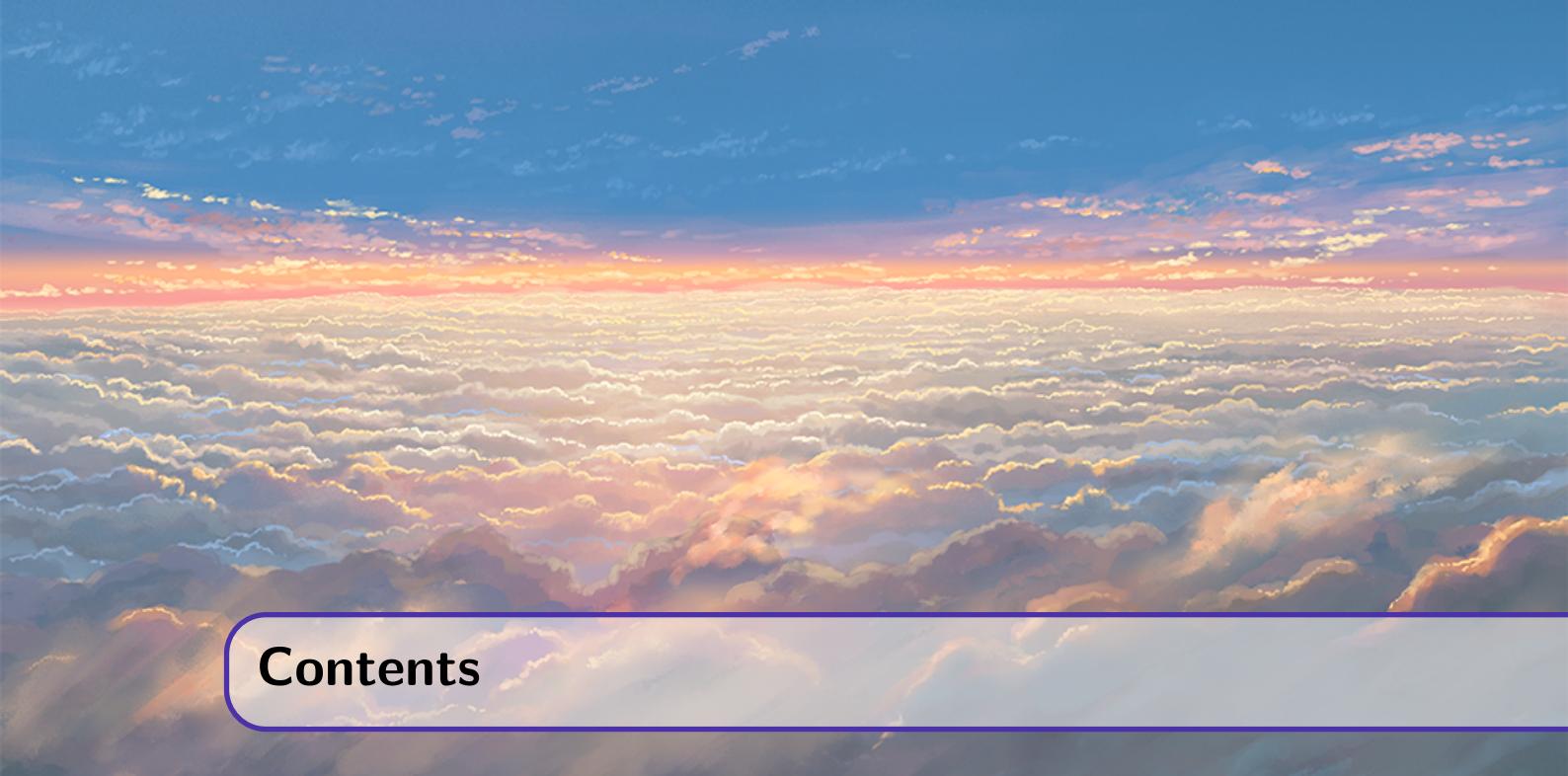
Xia Wenxuan

Written by Xia Wenxuan, 2021

<https://github.com/gegeji>

All the material in this document is taken from my textbook or video material, and some material uses optical character recognition (OCR) to aid input. It may contain typographical or content inaccuracies. This document is for my personal study only. I am not responsible for the accuracy of the content of the text.

Edited and Revised on December 16, 2021



Contents

I

Vectors

1	对向量的介绍	6
1.1	Vector	6
1.2	Vector Space	7
1.3	向量运算	7
1.4	内积	8
1.4.1	常用的内积等式	9
1.5	Cauchy-Schwartz Inequality	10
1.6	浮点运算	11
2	Linear Function	12
2.1	Linear Function	12
2.2	泰勒展开	13
2.3	高阶泰勒公式	15
2.4	Regression Model	16
3	Norm and Distance	17
3.1	Vector Norm	17
3.2	Root Mean Square Value (RMS)	19
3.3	Chebyshev's Inequality	19
3.4	Distance	19
3.4.1	Feature Distance and Nearest Neighbor	20

3.5	Standard Derivation	20
3.6	Angle	20
3.6.1	相关系数	21
4	优化问题初步	22
4.1	优化问题引入	22
4.2	Convex Set	23
4.2.1	Examples for Convex Functions	23
4.2.2	凸函数的充要条件	24
4.3	向量偏导	24
4.4	标量优化问题：投影问题	26
4.5	Clustering	26
5	Linear Independence	29
5.1	线性相关、线性无关	29
5.2	Basis	30
5.3	标准正交向量	30
5.4	Gram-Schmidt Algorithm	32
5.4.1	The Analysis of Gram-Schmidt Algorithm	33

II

Matrices

6	Matrices	36
6.1	Matrices	36
6.2	矩阵运算	37
6.2.1	Matrix Power	40
6.2.2	矩阵乘法的算法复杂度	40
6.2.3	矩阵向量乘积复杂度	41
6.3	Special Matrices and Matrices in Different Applications	42
6.3.1	$f(x) = Ax$ 中的 A	42
6.3.2	Selectors	43
6.3.3	图论：节点弧关联矩阵	44
6.3.4	Convolution	44
6.3.5	多项式	45
6.3.6	Fourier Transform	45
6.3.7	Semi-Definite Matrices	46
6.4	Gram 矩阵	47
7	Matrices Norms	48
7.1	矩阵范数	48
8	适定问题	52
8.1	The Definition of Well-posed Problem	52
8.2	绝对误差的界限	53

8.3 相对误差的界限	53
9 Inverse of Matrices	54
9.1 Left Inverse, Right Inverse, Inverse	54
9.2 Linear Equation Systems	55
9.2.1 线性方程组求解	55
9.3 Fundamental Theorem of Linear Algebra	56
9.4 Invertible Matrices	56
9.5 转置和共轭转置的逆	59
9.6 Gram Matrix 非奇异的性质	59
9.7 伪逆	59
10 Orthogonal Matrices	63
10.1 预备知识	63
10.1.1 标准正交向量	63
10.1.2 Gram 矩阵与标准正交的关系	63
10.1.3 矩阵-向量乘积与标准正交的关系	63
10.1.4 左可逆性与正交的关系	64
10.2 正交矩阵	65
10.3 Permutation Matrices	65
10.4 平面旋转	66
10.5 Householder Matrix	66
10.5.1 The Geometry of Householder Transformation	67
10.6 正交矩阵乘积	68
10.7 具有正交矩阵的线性方程	68
10.7.1 The Complexity of the Multiplication Ax of Orthogonal Matrix A	68
10.8 列标准正交的高矩阵	68
10.9 值域范围、列空间	69
10.9.1 投影到列标准正交的矩阵 A 的列空间	69
11 QR 分解与 Householder 变换	71
11.1 Triangular Matrices	71
11.1.1 高斯消元法	71
11.1.2 The Inverses of Triangular Matrices	73
11.2 QR Factorization	73
11.2.1 QR 分解的存在唯一性	74
11.2.2 复矩阵的 QR 分解	76
11.3 QR 分解的应用	76
11.3.1 QR 分解和求解线性方程组 $Ax = b$	77
11.3.2 QR 分解和求解伪逆 A^\dagger 、逆 A^{-1}	77
11.3.3 A 的列空间和 Q 的列空间相同	78
11.3.4 往 A 列空间上的投影也是往 Q 列空间上的投影	78

11.4 QR Algorithm Using Gram-Schmidt Algorithm	80
11.4.1 Gram-Schmidt Algorithm	81
11.4.2 基于 Gram-Schmidt 方法进行 QR 分解的时间复杂度	82
11.5 The Numerical Instability of QR Decomposition based on Gram-Schmidt Algorithm	83
11.6 QR Decomposition Using Householder Transformation	84
11.6.1 Householder Matrix	85
11.6.2 构造反射算子	86
11.6.3 Householder 三角化	88
11.6.4 Household-QR Algorithm	89
11.6.5 An Example for Householder Algorithm	91
11.6.6 Complexity of Householder Algorithm	92
11.7 Householder 变换进行 QR 分解的 Q 因子	92
11.7.1 Multiplication with Q factor	93
11.7.2 矩阵-向量积 $H_k x$ 算法复杂度	93
11.8 Fast Orthogonalization (Givens and Householder)	93
11.9 Recap: QR Decomposition	95
11.9.1 分治策略	95
11.9.2 非奇异矩阵的 QR 分解	95
11.9.3 使用 QR 分解求 A^{-1} 可以转换成 $R^{-1}Q^T$	95
11.9.4 QR 分解求解 A^{-1}	96
11.9.5 QR 分解求解线性方程组	96
12 LU 分解	97
12.1 Solving Linear Equation Systems	97
12.1.1 Linear Equation Systems	97
12.1.2 Elimination	97
12.2 LU 分解	99
12.2.1 $A = LDU$	100
12.2.2 L 、 U 矩阵的性质	100
12.2.3 Complexity of LU Decomposition	103
12.2.4 Example of LU Decomposition	103
12.3 Problem of LU Decomposition	104
12.4 $PA = LU$	104
12.5 舍入误差的影响	105
12.6 稀疏线性方程组	106

III

Least Squares

13 Least Squares	108
13.1 An Example: Measurement Problem	108
13.2 求解最小二乘法	110
13.3 The Geometry of Least Squares: 投影与 A 列空间的关系	112
13.4 正规方程	113

13.5 QR 分解求解最小二乘法	113
13.5.1 The Complexity of Solving Least Square Problem via QR Decomposition	114
13.6 求解正规方程可能带来的严重误差	114
13.7 梯度下降法	115
13.8 估计学习率（步长）α	116
14 Multi-objective Least Squares	119
14.1 Definition of Multi-objective Least Squares	119
14.2 求解多目标最小二乘问题	120
14.3 正则化数据拟合	120
14.4 图像逆问题	121
14.4.1 差分矩阵平滑最小二乘解	121
14.5 信号去噪	123
15 Constrained Least Squares	124
15.1 Karush—Kuhn—Tucker Conditions	124
15.1.1 An Example for Karush-Kuhn-Tucker Conditions	126
15.2 优化问题：最小范数优化问题	127
15.2.1 最小范数优化问题的例子	127
15.3 优化问题：点到线的最短距离	128
15.4 最小范数优化问题推导	128
15.5 QR 分解求解最小范数优化问题	129
15.5.1 QR 分解求解最小范数优化问题的复杂度	130
15.5.2 QR 分解求解最小范数优化问题的例子	130
15.6 最小二乘法约束问题	130
15.7 优化问题：分段多项式拟合问题	131
15.8 先验假设	132
15.9 优化问题：最小二乘法的带约束 KKT 条件	132
15.10 KKT 最优条件	133
15.11 LU 分解求解 KKT 最优条件	135
15.11.1 LU 分解求解 KKT 最优条件的时间复杂度	135
15.12 QR 分解求解 KKT 最优条件	136
15.13 KKT 最优条件求解复杂度：QR vs LU	137
15.14 Supplement Material: Karush-Kuhn-Tucker (KKT) 条件	138
15.14.1 等式约束优化问题	138
15.14.2 不等式约束优化问题	138
15.14.3 拉格朗日乘子法的例子	140
15.15 Supplement Material: 浅谈最优化问题的 KKT 条件	140
15.15.1 等式约束优化问题	141
15.15.2 不等式约束优化问题	141
15.15.3 KKT 乘子必须大于等于 0	143
15.15.4 总结：同时包含等式和不等式约束的一般优化问题	145

15.16 Supplement Material: 凸优化、拉格朗日乘子法和 KKT 条件	146
15.16.1 凸集的概念	146
15.16.2 凸优化	147
15.16.3 拉格朗日对偶性	147
15.16.4 总结	150
15.17 Supplementary Material: KKT Conditions, Linear Programming and Nonlinear Programming	150
15.17.1 Karush-Kuhn-Tucker Theorem(s)	150
15.17.2 Linear Programming and KKT Conditions - An Example	153
15.17.3 The Dual KKT Conditions	155
15.17.4 An Example from Nonlinear Programming	158
15.18 Supplementary Material: KKT Conditions	160
15.18.1 Dual Problem	160
15.19 Karush-Kuhn-Tucker (KKT) Conditions	160
16 非线性最小二乘法	162
16.1 非线性最小二乘法的定义	162
16.1.1 例子：距离测量定位	162
16.1.2 例子：多个相机视图定位	163
16.1.3 例子：模型拟合	163
16.1.4 例子：正交距离回归	164
16.1.5 非线性最小二乘法	164
16.1.6 二分类	164
16.2 梯度	164
16.2.1 Gradient and Directional Derivative	164
16.2.2 Jacobian Matrices	165
16.2.3 Hessian	165
16.2.4 Optimality conditions for twice differentiable g	167
16.3 求解非线性最小二乘法：目标梯度	168
16.4 Gauss-Newton Algorithm	169
16.5 Levenberg—Marquardt Algorithm	171
16.5.1 正则化参数	172
16.6 Newton Method	172
16.6.1 Interpretation of Newton Method	173
16.6.2 One Variable	173
16.6.3 Relation to Gauss—Newton method	173
16.6.4 Examples	174
16.6.5 Convergence of Newton's method	174
16.6.6 Newton's method for minimizing a convex function	174
16.6.7 Damped Newton method	175
16.7 Newton method for nonlinear least squares	176
16.7.1 Hessian of nonlinear least squares cost	176
16.7.2 Comparison	177
16.8 Unconstrained minimization problem	177
16.9 Local and global optimum	177

17	Fourier Series, Fourier Transform	179
17.1	基本概念	179
17.2	Fourier Series	181
17.3	Fourier Transform	184
17.4	Discrete Fourier Transform	185
18	Factorization of Matrices	187
18.1	主要的矩阵分解	187

19	Random Variables	189
19.1	随机变量的引入	189
19.2	随机变量的特点	189
19.3	优势	189
19.4	常见随机变量	189
19.4.1	离散型随机变量	189
19.4.2	两点分布	190
19.4.3	(n 重) 伯努利试验, 二项分布	190
19.4.4	泊松分布	190
19.4.5	连续型随机变量	191
19.4.6	均匀分布	191
19.4.7	指数分布	191
19.4.8	正态分布	191
19.4.9	标准正态分布	191
19.5	随机变量的分布函数	191
19.6	随机变量的分布函数性质	191
19.7	连续型随机变量及其概率密度	191
19.8	随机变量的数字特征	192
19.8.1	方差	192
19.8.2	协方差	192
19.8.3	相关系数	193
19.8.4	X, Y 不相关时候的性质	193
20	大数定律和中心极限定理	194
20.1	辛钦大数定律	194
20.1.1	辛钦大数定律条件	194
20.2	伯努利大数定理	194
20.3	中心极限定理	195

21	Sampling Distribution	196
21.1	统计量	196
21.2	常见统计量	196
21.3	卡方分布	197
21.3.1	卡方分布性质	197
21.4	t 分布	198
21.4.1	t 分布性质	198
21.5	F 分布	199
21.5.1	F 分布性质	199
21.6	正态总体的样本均值和样本方差的分布	200
22	Hypothesis Testing	203
22.1	正态总体均值方差的检验法	203
22.2	经验分布函数	203
22.3	Q-Q 图 (Quantile-quantile Plot)	203
22.4	卡方拟合优度检验	205
22.5	柯尔莫哥洛夫 (Kolmogorov-Smirnov) 检验	205
22.6	秩和检验	207
22.7	方差分析 (Analysis of Variance, ANOVA)	207
22.7.1	单因素方差分析	207
22.7.2	双因素方差分析方法	209
22.7.3	无交互影响的双因素方差分析	210
22.7.4	关于交互效应的双因素方差分析	211
22.8	多元线性回归	212
22.8.1	回归模型的假设检验	213
22.8.2	回归系数的假设检验和区间估计	213
22.8.3	利用回归模型进行预测	214
22.9	逐步回归	214
22.9.1	前进法	215
22.9.2	后退法	215
23	Bootstrap	216
23.1	估计量的标准误差的 Bootstrap 估计	216
23.2	估计量的均方误差的 Bootstrap 估计	217
23.3	Bootstrap 置信区间	217
23.4	参数 Bootstrap 方法	218
24	List Of Definitions	219

Vectors

1	对向量的介绍	6
1.1	Vector		
1.2	Vector Space		
1.3	向量运算		
1.4	内积		
1.5	Cauchy-Schwartz Inequality		
1.6	浮点运算		
2	Linear Function	12
2.1	Linear Function		
2.2	泰勒展开		
2.3	高阶泰勒公式		
2.4	Regression Model		
3	Norm and Distance	17
3.1	Vector Norm		
3.2	Root Mean Square Value (RMS)		
3.3	Chebyshev's Inequality		
3.4	Distance		
3.5	Standard Derivation		
3.6	Angle		
4	优化问题初步	22
4.1	优化问题引入		
4.2	Convex Set		
4.3	向量偏导		
4.4	标量优化问题：投影问题		
4.5	Clustering		
5	Linear Independence	29
5.1	线性相关、线性无关		
5.2	Basis		
5.3	标准正交向量		
5.4	Gram-Schmidt Algorithm		

1. 对向量的介绍

1.1 Vector

Definition 1.1.1 — Vector. 一个有序的数字列表.

$$\begin{bmatrix} -1.1 \\ 0.0 \\ 3.6 \\ -7.2 \end{bmatrix} \text{ 或者 } \begin{pmatrix} -1.1 \\ 0.0 \\ 3.6 \\ -7.2 \end{pmatrix} \text{ 或者 } (-1.1, 0, 3.6, -7.2)$$

表中的数字是元素(项、系数、分量). 元素的数量是向量的大小(维数, 长度). 大小为 n 的向量称为 n 维向量. 向量中的数字通常被称作标量.

用符号来表示向量, 比如 a, b , 一般小写字母表示. 其它表示形式 \mathbf{g}, \vec{a}

Definition 1.1.2 — n 维向量 a 的第 i 元素. n 维向量 a 的第 i 元素表示为 a_i .
有时 i 指的是向量列表中的第 i 个向量.

Definition 1.1.3 — $a = b$. 对于所有 i , 如果有 $a_i = b_i$, 则称两个相同大小的向量 a 和 b 是相等的, 可写成 $a = b$

Definition 1.1.4 — stacked vector. 假设 b, c, d 是大小为 m, n, p 的向量

$$a = \begin{bmatrix} b \\ c \\ d \end{bmatrix}$$

$$a = (b_1, b_2, \dots, b_m, c_1, c_2, \dots, c_n, d_1, d_2, \dots, d_p)$$

Definition 1.1.5 — 零向量. 所有项为 0 的 n 维向量表示为 0_n 或者 0.

Definition 1.1.6 — 全一向量. 所有项为 1 的 n 维向量表示为 $\mathbf{1}_n$ 或者 1.

Definition 1.1.7 — 单位向量. 当第 i 项为 1, 其余项为 0 时表示为 e_i

$$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Definition 1.1.8 — 稀疏向量. 如果一个向量的许多项都是 0, 该向量为稀疏 (Sparse) 的. 稀疏向量能在计算机上高效地存储和操作.

$\text{nnz}(x)$ 是指向量 x 中非零的项数 (number of non-zeros), 有时用 ℓ_0 表示.

向量 $x = (x_1, x_2)$ 可以在二维中表示一个位置或一个位移、图像、单词统计、颜色等.

1.2 Vector Space

Definition 1.2.1 — 向量空间 V . 设 V 是非空子集, P 是一数域, 向量空间 V 满足:

1. 向量加法: $V + V \rightarrow V$, 记作 $\forall x, y \in V$, 则 $x + y \in V$ (加法封闭)
 2. 标量乘法: $P \times V \rightarrow V$, 记作 $\forall x \in V, \lambda \in P$, 则 $\lambda x \in V$ (乘法封闭)
- 上述两个运算满足下列八条规则 ($\forall x, y, z \in V, \lambda, \mu \in P$)
1. $x + y = y + x$ (交换律)
 2. $x + (y + z) = (x + y) + z$ (结合律)
 3. V 存在一个零元素, 记作 0, $x + 0 = x$
 4. 存在 x 的负元素, 记作 $-x$, 满足 $x + (-x) = 0$
 5. $\forall x \in V$, 都有 $1x = x, 1 \in P$
 6. $\lambda(\mu x) = (\lambda\mu)x$
 7. $(\lambda + \mu)x = \lambda x + \mu x$
 8. $\lambda(x + y) = \lambda x + \lambda y$

Corollary 1.2.1 向量空间也称为线性空间.

Corollary 1.2.2 如果 $x, y \in \mathbb{R}^2$, 则 $x + y \in \mathbb{R}^2, \lambda x \in \mathbb{R}^2 (\lambda \in \mathbb{R})$.

Definition 1.2.2 — 数域. 数的非空集合 P , 且其中任意两个数的和、差、积、商 (除数不为零) 仍属于该集合, 则称数集 P 为一个数域.

■ **Example 1.1** 有理数 \mathbb{Q}

■ **Example 1.2** 实数 \mathbb{R}

$x, y \in \mathbb{R}, x = 1, y = 2 x + y \in \mathbb{R}, x \times y \in \mathbb{R}$

■ **Example 1.3** 复数 \mathbb{C}

1.3 向量运算

Definition 1.3.1 — 向量加法. n 维向量 a 和 b 可以相加, 求和形式表示为 $a + b$.

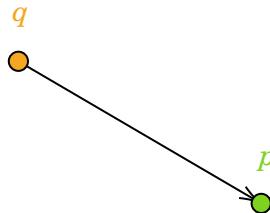
设向量 a, b, C 是向量空间 V 的元素, 即 $a, b, c \in V$.

1. 交换律: $a + b = b + a$
2. 结合律: $(a + b) + c = a + (b + c)$ (因此可写成 $a + b + c$)
3. $a + 0 = 0 + a = a$
4. $a - a = 0$

Corollary 1.3.1 — 向量位移相加. 如果二维向量 a 和 b 都表示位移, 则它们的位移之和为 $a + b$

■ **Example 1.4** 点 q 到点 p 的位移是 $p - q$.

Figure 1.1: The translation from q to p



Definition 1.3.2 — 标量与向量的乘法.

$$\beta a = \begin{bmatrix} \beta a_1 \\ \vdots \\ \beta a_n \end{bmatrix}$$

标量 β, γ 与向量 a, b 进行乘法, 有如下性质:

1. 结合律: $(\beta\gamma)a = \beta(\gamma a)$
2. 左分配律: $(\beta + \gamma)a = \beta a + \gamma a$
3. 右分配律: $\beta(a + b) = \beta a + \beta b$

Definition 1.3.3 — 线性组合. 对于向量 a_1, \dots, a_m 和标量 β_1, \dots, β_m ,

$$\beta_1 a_1 + \dots + \beta_m a_m$$

是向量的线性组合. β_1, \dots, β_m 是该向量的系数.

■ **Example 1.5** 对于任何向量 $b \in \mathbb{R}^n$, 有如下等式

$$b = b_1 e_1 + \dots + b_n e_n, b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

1.4 内积

Definition 1.4.1 — 内积. 在数域 \mathbb{R} 上的向量空间 V , 定义函数 $\langle \cdot, \cdot \rangle : V \times V \rightarrow \mathbb{R}$, 满足:

1. $\langle a, a \rangle \geq 0, \forall a \in V$, 当且仅当 $a = 0$ 时 $\langle a, a \rangle = 0$
2. $\langle \alpha a + \beta b, c \rangle = \alpha \langle a, c \rangle + \beta \langle b, c \rangle, \forall \alpha, \beta \in \mathbb{R}$, 且 $a, b, c \in V$
3. $\langle a, b \rangle = \langle b, a \rangle, \forall a, b \in V$

函数 $\langle \cdot, \cdot \rangle : V \times V \rightarrow \mathbb{R}$ 成为内积.

■ **Example 1.6** 在向量空间 \mathbb{R}^n 上, 计算两个向量对应项相乘之后求和函数

$$\langle a, b \rangle = a_1 b_1 + a_2 b_2 + \dots + a_n b_n = a^T b$$

where $a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}$, $b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \in \mathbb{R}^n$.

Proof. $\langle a, a \rangle = a_1 a_1 + a_2 a_2 + \cdots + a_n a_n = \sum_{i=1}^n a_i^2 \geq 0$, $\langle a, a \rangle = 0$, 则 $a = 0$

$$\begin{aligned} \langle \alpha a + \beta b, c \rangle &= (\alpha a_1 + \beta b_1) c_1 + (\alpha a_2 + \beta b_2) c_2 + \cdots + (\alpha a_n + \beta b_n) c_n \\ &= \alpha \sum_{i=1}^n a_i c_i + \beta \sum_{i=1}^n b_i c_i \\ &= \alpha \langle a, c \rangle + \beta \langle b, c \rangle \end{aligned}$$

$$\langle a, b \rangle = a^T b = b^T a = \langle b, a \rangle$$

■

内积的性质：交换律、结合律、分配律。

交换律： $a^T b = b^T a$

结合律： $(\gamma a)^T b = \gamma (a^T b)$

分配律： $(a + b)^T c = a^T c + b^T c$

1.4.1 常用的内积等式

Corollary 1.4.1 — 选出第 i 项.

$$e_i^T a = a_i$$

Corollary 1.4.2 — 向量每一项之和.

$$\mathbf{1}^T a = a_1 + \cdots + a_n$$

Corollary 1.4.3 — 向量每一项的平方和.

$$a^T a = a_1^2 + \cdots + a_n^2$$

Corollary 1.4.4 — 向量元素的平均值.

$$(\mathbf{1}/n)^T a = (a_1 + \cdots + a_n) / n$$

Corollary 1.4.5 — Selective sum. Let b be a vector all of whose entries are either 0 or 1. Then

$$b^T a$$

is the sum of the elements in a for which $b_i = 1$.

Definition 1.4.2 — The sum of block vectors. If the vectors a and b are block vectors, and the corresponding blocks have the same sizes (in which case we say they *conform*), then

$$a^T b = \begin{bmatrix} a_1 \\ \vdots \\ a_k \end{bmatrix}^T \begin{bmatrix} b_1 \\ \vdots \\ b_k \end{bmatrix} = a_1^T b_1 + \cdots + a_k^T b_k$$

内积用途很广.

■ **Example 1.7 — 计算同时出现的项目数.**

$$a = (0, 1, 1, 1, 1, 1, 1), \quad b = (1, 0, 1, 0, 1, 0, 0)$$

Here we have $a^T b = 2$, which is the number of objects in both A and B (i.e., objects 3 and 5). ■

■ **Example 1.8 — Weights, features, and score.** When the vector f represents a set of *features* of an object, and w is a vector of the same size (often called a *weight vector*), the inner product $w^T f$ is the sum of the feature values, scaled (or weighted) by the weights, and is sometimes called a *score*. ■

■ **Example 1.9**

$$p(x) = c_1 + c_2 x + \cdots + c_{n-1} x^{n-2} + c_n x^{n-1}$$

Let t be a number, $z = (1, t, t^2, \dots, t^{n-1})$ be the n -vector of powers of t . Then

$$c^T z = p(t)$$

1.5 Cauchy-Schwartz Inequality

Theorem 1.5.1 — Cauchy-Schwartz Inequality. 设 $\langle \cdot, \cdot \rangle$ 是向量空间 V 上的内积, $\forall x, y \in V$, 则有

$$|\langle x, y \rangle|^2 \leq \langle x, x \rangle \langle y, y \rangle$$

当 $x = -\lambda y$ 时, 有 $|\langle x, y \rangle|^2 = \langle x, x \rangle \langle y, y \rangle$.

Proof. 令 $\lambda \in \mathbb{R}$, 则有

$$0 \leq \langle x + \lambda y, x + \lambda y \rangle = \langle x, x \rangle + \lambda \langle y, x \rangle + \lambda \langle x, y \rangle + \lambda^2 \langle y, y \rangle = \langle x, x \rangle + 2\lambda \langle y, x \rangle + \lambda^2 \langle y, y \rangle$$

则

$$\lambda^2 \langle y, y \rangle + 2\lambda \langle y, x \rangle + \langle x, x \rangle \geq 0, \forall \lambda \in \mathbb{R}$$

所以

$$\nabla = (2\langle y, x \rangle)^2 - 4\langle y, y \rangle \langle x, x \rangle \leq 0$$

$$|\langle x, y \rangle|^2 \leq \langle x, x \rangle \langle y, y \rangle$$

当 $|\langle x, y \rangle|^2 = \langle x, x \rangle \langle y, y \rangle$ 时, 有

$$\langle x, x \rangle^2 + 2\lambda \langle y, x \rangle + \lambda^2 \langle y, y \rangle = 0$$

也即

$$\langle x + \lambda y, x + \lambda y \rangle = 0$$

因此 $x + \lambda y = 0$, 即 $x = -\lambda y$. ■

Theorem 1.5.2 — Cauchy-Schwarz 不等式的矩阵元素形式.

$$\left(\sum_{i=1}^n u_i v_i \right)^2 \leq \left(\sum_{i=1}^n u_i^2 \right) \left(\sum_{i=1}^n v_i^2 \right)$$

Proof. The Cauchy-Schwarz inequality can be proved using only ideas from elementary algebra in this case. Consider the following quadratic polynomial in x

$$0 \leq (u_1 x + v_1)^2 + \cdots + (u_n x + v_n)^2 = \left(\sum_i u_i^2 \right) x^2 + 2 \left(\sum_i u_i v_i \right) x + \sum_i v_i^2$$

Since it is nonnegative, it has at most one real root for x , hence its discriminant is less than or equal to zero. That is,

$$\left(\sum_i u_i v_i \right)^2 - \left(\sum_i u_i^2 \right) \left(\sum_i v_i^2 \right) \leq 0$$

which yields the Cauchy-Schwarz inequality. ■

Corollary 1.5.3 — Cauchy-Schwarz 不等式变体. 由 Cauchy-Schwarz 不等式

$$|\langle x, y \rangle|^2 \leq \langle x, x \rangle \langle y, y \rangle$$

可以推得

$$\begin{aligned} |\langle a, b \rangle| &\leq \|a\|_2 \|b\|_2 \\ \langle a, b \rangle &\geq -\|a\|_2 \|b\|_2 \end{aligned}$$

1.6 浮点运算

计算机以浮点格式存储(实)数值.

基本的算术运算(加法, 乘法等)被称为浮点运算(flop).

算法或操作的时间复杂度: 作为输入维数的函数所需要的浮点运算总数.

算法复杂度通常以非常粗略地近似估算.

(程序)执行时间的粗略估计: 计算机速度/flops, 目前的计算机大约是 1Gflops/秒 (10^9 flops/秒).

Corollary 1.6.1 假设有 n 维向量 x 和 y :

- $x + y$ 需要 n 次加法, 所以时间复杂度为 (n) flops.
- $x^T y$ 需要 n 次乘法和 $n - 1$ 次加法, 所以时间复杂度为 $(2n - 1)$ flops.
- 对于 $x^T y$, 通常将其时间复杂度简化为 $2n$, 甚至为 n .
- 当 x 或 y 是稀疏的时候, 算法的实际运算时间会比理论时间更少.

2. Linear Function

2.1 Linear Function

Definition 2.1.1 — Linear Function. f 是一个将 n 维向量映射成数的函数.

$$f : \mathbb{R}^n \rightarrow \mathbb{R}$$

线性函数 f 满足以下两个性质 ($k \in \mathbb{R}, x, y \in \mathbb{R}^n$) :

- 齐次性 (homogeneity): $f(kx) = kf(x)$
- 叠加性 (Additivity): $f(x + y) = f(x) + f(y)$

■ **Example 2.1** 求平均值: $f(x) = \frac{1}{n} \sum_{i=1}^n x_i$ 为线性函数.

■ **Example 2.2** 求最大值: $f(x) = \max \{x_1, x_2, \dots, x_n\}$ 并不是线性函数.

Proof. 令 $x = (1, -1), y = (-1, 1), \alpha = 0.5, \beta = 0.5$, 有

$$f(\alpha x + \beta y) = 0 \neq \alpha f(x) + \beta f(y) = 1$$

$$\begin{aligned} f(x + y) &= \max \{x_1 + y_1, x_2 + y_2, \dots, x_n + y_n\} \\ &\leq \max \{x_1, x_2, \dots, x_n\} + \max \{y_1, y_2, \dots, y_n\} \\ &\leq f(x) + f(y) \end{aligned}$$

Theorem 2.1.1 设 $\alpha_1, \dots, \alpha_m \in \mathbb{R}, u_1, \dots, u_m \in \mathbb{R}^n$, 则线性函数 f 满足

$$\begin{aligned}
 f(\alpha_1 u_1 + \alpha_2 u_2 + \dots + \alpha_m u_m) &= f(\alpha_1 u_1) + f(\alpha_2 u_2 + \dots + \alpha_m u_m) \\
 &= \alpha_1 f(u_1) + f(\alpha_2 u_2 + \dots + \alpha_m u_m) \\
 &= \alpha_1 f(u_1) + \alpha_2 f(u_2) + \dots + \alpha_m f(u_m)
 \end{aligned}$$

Definition 2.1.2 — 内积函数 (inner product function). 对于 n 维向量 a , 满足以下形式的函数被称为内积函数

$$f(x) = a^T x = a_1 x_1 + a_2 x_2 + \dots + a_n x_n$$

上述 $f(x)$ 可以看作是每项 x_i 的加权之和.

Corollary 2.1.2 内积函数都是线性的.

Proof.

$$\begin{aligned}
 f(\alpha x + \beta y) &= a^T (\alpha x + \beta y) \\
 &= a^T (\alpha x) + a^T (\beta y) \\
 &= \alpha (a^T x) + \beta (a^T y) \\
 &= \alpha f(x) + \beta f(y)
 \end{aligned}$$

■

Corollary 2.1.3 所有线性函数都是内积.

$$\begin{aligned}
 f(x) &= f(x_1 e_1 + x_2 e_2 + \dots + x_n e_n) \\
 &= x_1 f(e_1) + x_2 f(e_2) + \dots + x_n f(e_n)
 \end{aligned}$$

Proof. 假设 $f : \mathbb{R}^n \rightarrow \mathbb{R}$ 是线性函数, 那么可用 $f(x) = a^T x$ 来表示, a 为常量.

$$\begin{aligned}
 f(x) &= f(x_1 e_1 + x_2 e_2 + \dots + x_n e_n) \\
 &= x_1 f(e_1) + x_2 f(e_2) + \dots + x_n f(e_n)
 \end{aligned}$$

■

Definition 2.1.3 — 仿射函数 (affine function). 其一般形式为 $f(x) = a^T x + b$, 其中 $a \in \mathbb{R}^n$, $b \in \mathbb{R}$ 为标量.

Theorem 2.1.4 函数 $f : \mathbb{R}^n \rightarrow \mathbb{R}$ 为仿射函数需要满足

$$f(\alpha x + \beta y) = \alpha f(x) + \beta f(y), \alpha + \beta = 1, \alpha, \beta \in \mathbb{R}, x, y \in \mathbb{R}^n$$

2.2 泰勒展开

Definition 2.2.1 — 函数 f 第 i 个分量的一阶偏导数. 假设 $f : \mathbb{R}^n \rightarrow \mathbb{R}$, 函数 f 在 z 点可微

$$\begin{aligned}\frac{\partial f}{\partial z_i}(z) &= \lim_{t \rightarrow 0} \frac{f(z_1, \dots, z_{i-1}, z_i + t, z_{i+1}, \dots, z_n) - f(z)}{t} \\ &= \lim_{t \rightarrow 0} \frac{f(z + te_i) - f(z)}{t}\end{aligned}$$

Definition 2.2.2 — f 在点 z 的梯度.

$$\nabla f(z) = \begin{bmatrix} \frac{\partial f}{\partial z_1}(z) \\ \vdots \\ \frac{\partial f}{\partial z_n}(z) \end{bmatrix}$$

Definition 2.2.3 — Taylor's Approximation. 假设 $f : \mathbb{R}^n \rightarrow \mathbb{R}$, 函数 f 在 z 点充分光滑, 即处处可导.

$$\begin{aligned}f(x) &= f(z) + \frac{\partial f}{\partial x_1}(z)(x_1 - z_1) + \frac{\partial f}{\partial x_2}(z)(x_2 - z_2) + \dots + \frac{\partial f}{\partial x_n}(z)(x_n - z_n) \\ &\quad + \frac{1}{2!} \sum_{j=1}^n \sum_{i=1}^n \frac{\partial f^2}{\partial x_i \partial x_j}(z)(x_i - z_i)(x_j - z_j) + \dots\end{aligned}$$

Definition 2.2.4 — 一阶泰勒公式. 假设 $f : \mathbb{R}^n \rightarrow \mathbb{R}$, 函数 f 在 z 点可导

$$\hat{f}(x) = f(z) + \frac{\partial f}{\partial x_1}(z)(x_1 - z_1) + \dots + \frac{\partial f}{\partial x_n}(z)(x_n - z_n)$$

当 x 非常接近 z 时, $\hat{f}(x)$ 也非常接近 $f(z)$. $\hat{f}(x)$ 是关于 x 的一个仿射函数.

Corollary 2.2.1 — 一阶泰勒公式的内积形式.

$$\hat{f}(x) = f(z) + \nabla f(z)^T(x - z) \quad (\nabla f(z) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(z) \\ \vdots \\ \frac{\partial f}{\partial x_n}(z) \end{bmatrix})$$

一维时, $\hat{f}(x) = f(z) + f'(z)(x - z)$.

■ Example 2.3

$$f(x) = x_1 - 3x_2 + e^{2x_1+x_2-1}$$

$$\nabla f(x) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(x) \\ \frac{\partial f}{\partial x_2}(x) \end{bmatrix} = \begin{bmatrix} 1 + 2e^{2x_1+x_2-1} \\ -3 + e^{2x_1+x_2-1} \end{bmatrix}$$

函数 f 在 o 点的一阶泰勒公式为:

$$\hat{f}(x) = f(0) + \nabla f(0)^T(x - 0) = e^{-1} + (1 + 2e^{-1})x_1 + (-3 + e^{-1})x_2$$

2.3 高阶泰勒公式

泰勒公式利用多项式在一点附近逼近函数.

■ Example 2.4

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \cdots + (-1)^{k-1} \frac{x^{2k-1}}{(2k-1)!} + \frac{\sin [\xi + (2k+1)\frac{\pi}{2}]}{(2k+1)!} x^{2k+1}$$

一次逼近: $\sin x \approx x$

三次逼近: $\sin x \approx x - \frac{x^3}{3!}$

■

Proof.

$$f(x) = P_n(x) + R_n(x)$$

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)^2 + \cdots + a_n(x - x_0)^n$$

$$R_n(x) = o(x - x_0)^n$$

$$f(x) \approx P_n(x)$$

$$\therefore P_n(x_0) = f(x_0), P'_n(x_0) = f'(x_0), P''_n(x_0) = f''(x_0), \dots, P_n^{(n)}(x_0) = f^{(n)}(x_0)$$

$$\text{要求 } P_n(x_0) = f(x_0) \Rightarrow a_0 = f(x_0)$$

$$P'_n(x) = a_1 + 2a_2(x - x_0) + \cdots + na_n(x - x_0)^{n-1} \Rightarrow a_1 = f'(x_0)$$

$$\text{依此类推. } a_n = \frac{f^{(n)}(x_0)}{n!}$$

■

Corollary 2.3.1 — n 阶泰勒多项式.

$$\begin{aligned} P_n(x) &= f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 \\ &\quad + \cdots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n \end{aligned}$$

$$\text{where } a_n = \frac{f^{(n)}(x_0)}{n!}$$

Corollary 2.3.2 — 对于高阶余项的公式. 带拉格朗日余项的泰勒公式

$$\begin{aligned} f(x) &= f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 \\ &\quad + \cdots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n + \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)^{n+1} \end{aligned}$$

$$R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)^{n+1} (\xi \text{ 在 } x_0 \text{ 与 } x \text{ 之间})$$

Corollary 2.3.3 — 麦克劳林 (Maclaurin) 公式. 在零点展开麦克劳林 (Maclaurin) 公式

$$\begin{aligned} f(x) &= f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \cdots + \frac{f^{(n)}(0)}{n!}x^n \\ &\quad + \frac{f^{(n+1)}(\theta x)}{(n+1)!}x^{n+1} \quad (0 < \theta < 1) \end{aligned}$$

2.4 Regression Model

Definition 2.4.1 — Regression Model. 回归模型 (regression model) 为关于 x 的仿射函数

$$\hat{y} = x^T \beta + v$$

x 是特征向量 (*feature vector*), 它的元素 x_i 称为回归元 (*regressors*). n 维向量 β 是权重向量 (*weight vector*). 标量 v 是偏移量 (*offset*). 标量 \hat{y} 是预测值 (*prediction*). 表示某个实际结果或因变量, 用 y 表示.

3. Norm and Distance

3.1 Vector Norm

Definition 3.1.1 — Vector Norm. 在向量空间中存在一个函数 $\|\cdot\| : \mathbb{R}^n \rightarrow \mathbb{R}$, 且满足以下条件

- 齐次性: $\|\alpha x\| = |\alpha| \|x\|$, $\alpha \in \mathbb{R}$ 且 $x \in \mathbb{R}^n$;
- 三角不等式: $\|x + y\| \leq \|x\| + \|y\|$, $x, y \in \mathbb{R}^n$;
- 非负性: $\|x\| \geq 0$, $x \in \mathbb{R}^n$ 且 $\|x\| = 0 \Leftrightarrow x = 0$;

则称 $\|\cdot\|$ 为向量范数.

■ **Example 3.1 — ℓ_1 -范数 (曼哈顿范数, Manhattan norm) .**

$$\|x\|_1 = |x_1| + |x_2| + \dots + |x_n| \quad x, y \in \mathbb{R}^n, \alpha \in \mathbb{R}$$

Proof.

$$\|\alpha x\|_1 = |\alpha x_1| + |\alpha x_2| + \dots + |\alpha x_n| = |\alpha| \|x\|_1 \geq 0$$

$$\|x + y\|_1 = |x_1 + y_1| + \dots + |x_n + y_n| \leq |x_1| + |y_1| + \dots + |x_n| + |y_n| = \|x\|_1 + \|y\|_1$$

■ **Example 3.2 — ℓ_2 -范数 (欧几里得范数, Euclidean norm) .**

$$\|x\|_2 = \sqrt{(x_1^2 + x_2^2 + \dots + x_n^2)} = \sqrt{x^T x} = (\langle x, x \rangle)^{\frac{1}{2}}$$

Proof.

$$\|\alpha x\|_2 = (\langle \alpha x, \alpha x \rangle)^{\frac{1}{2}} = |\alpha| (\langle x, x \rangle)^{\frac{1}{2}} = |\alpha| \|x\|_2$$

$$\begin{aligned}
\|x+y\|_2^2 &= \langle x+y, x+y \rangle = \langle x, x \rangle + \langle x, y \rangle + \langle y, x \rangle + \langle y, y \rangle \\
&= \|x\|_2^2 + 2\langle x, y \rangle + \|y\|_2^2 \leq \|x\|_2^2 + 2\|x\|_2\|y\|_2 + \|y\|_2^2 \\
&= (\|x\|_2 + \|y\|_2)^2
\end{aligned}$$

$$\|x+y\|_2 \leq \|x\|_2 + \|y\|_2$$

■

Corollary 3.1.1 — 柯西—施瓦茨不等式.

$$|\langle x, y \rangle|^2 \leq \langle x, x \rangle \langle y, y \rangle = \|x\|_2^2 \|y\|_2^2$$

Definition 3.1.2 — ℓ_∞ -范数.

$$\|x\|_\infty = \max_{1 \leq i \leq n} |x_i|, x \in \mathbb{R}^n$$

Proof.

$$\begin{aligned}
\max_{1 \leq i \leq n} |x_i| &\leq (|x_1|^p + \cdots + |x_i|^p + \cdots + |x_n|^p)^{1/p} \\
&\leq \left(n \max_{1 \leq i \leq n} |x_i|^p \right)^{1/p} \\
&= n^{1/p} \max_{1 \leq i \leq n} |x_i| \\
&\rightarrow \max_{1 \leq i \leq n} |x_i| \quad (p \rightarrow \infty)
\end{aligned}$$

■

Definition 3.1.3 — ℓ_p -范数.

$$\|x\|_p = (x_1^p + x_2^p + \cdots + x_n^p)^{\frac{1}{p}}, \quad x \in \mathbb{R}^n, p \geq 1$$

ℓ_1 范数 $\|x\|_1$, ℓ_2 -范数 $\|x\|_2$, ℓ_∞ -范数是 ℓ_p -范数的特例.

证明可以使用以下两条不等式

Theorem 3.1.2 — Minkowski Inequality.

$$\left(\sum_{i=1}^n |x_i + y_i|^p \right)^{\frac{1}{p}} \leq \left(\sum_{i=1}^n |x_i|^p \right)^{\frac{1}{p}} + \left(\sum_{i=1}^n |y_i|^p \right)^{\frac{1}{p}}, \quad p \geq 1, x, y \in \mathbb{R}^n$$

Theorem 3.1.3 — Hölder Inequality.

$$\sum_{i=1}^n |x_i y_i| \leq \left(\sum_{i=1}^n |x_i|^p \right)^{1/p} \left(\sum_{i=1}^n |y_i|^q \right)^{1/q}, \quad \frac{1}{p} + \frac{1}{q} = 1, 1 < p, q < \infty$$

3.2 Root Mean Square Value (RMS)

Definition 3.2.1 — 向量 x 的均方值 (mean-square value). 向量 $x \in \mathbb{R}^n$ 的均方值 (mean-square value)

$$\frac{x_1^2 + x_2^2 + \cdots + x_n^2}{n} = \frac{\|x\|_2^2}{n}$$

Definition 3.2.2 — n 维向量 x 的均方根 (root-mean-square value, RMS).

$$\text{rms}(x) = \sqrt{\frac{x_1^2 + x_2^2 + \cdots + x_n^2}{n}} = \frac{\|x\|_2}{\sqrt{n}}$$

$\text{rms}(x)$ 给出了 $|x_i|$ 的“典型”(typical) 值. 例如, $\text{rms}(\mathbf{1}) = 1$ (与 n 无关). 均方根 (RMS) 值对于比较不同长度的向量大小是比较有用的.

3.3 Chebyshev's Inequality

Theorem 3.3.1 — Chebyshev's Inequality.

$$P(|X - \mu| \geq \varepsilon) \leq \frac{\sigma^2}{\varepsilon^2}$$

$$P(|X - \mu| < \varepsilon) \geq 1 - \frac{\sigma^2}{\varepsilon^2}$$

Theorem 3.3.2 — Chebyshev's Inequality. 假设 k 为向量 x 分量满足条件 $|x_i| \geq a$ 的个数, 即 $x_i^2 \geq a^2$ 的个数.

则满足 $|x_i| \geq a$ 的 x_i 数量不会超过 $\frac{\|x\|_2^2}{a^2}$.

Proof. 假设 k 为向量 x 分量满足条件 $|x_i| \geq a$ 的个数, 即 $x_i^2 \geq a^2$ 的个数。

将 a^2 移项, 可得到 $k \leq \frac{\|x\|_2^2}{a^2}$

因此

$$\|x\|_2^2 = x_1^2 + x_2^2 + \cdots + x_n^2 \geq ka^2$$

■

Corollary 3.3.3 — Chebyshev's Inequality Using RMS.

$$\text{rms}(x) = \sqrt{\frac{x_1^2 + x_2^2 + \cdots + x_n^2}{n}} = \frac{\|x\|_2}{\sqrt{n}}$$

$|x_i| \geq a$ 的项数占整体的比例不会超过 $\left(\frac{\text{rms}(x)}{a}\right)^2$, 即 $\frac{k}{n} \leq \left(\frac{\text{rms}(x)}{a}\right)^2$

3.4 Distance

Definition 3.4.1 — Euclidean distance. n 维向量 a 和 b 之间的欧氏距离

$$\text{dist}(a, b) = \|a - b\|_2$$

Definition 3.4.2 — RMS deviation. $\text{rms}(a - b)$ 是 a 和 b 之间的均方根偏差.

Theorem 3.4.1 — Triangular Inequality.

$$\|a - c\|_2 = \|(a - b) + (b - c)\|_2 \leq \|a - b\|_2 + \|b - c\|_2$$

3.4.1 Feature Distance and Nearest Neighbor

Definition 3.4.3 — Feature Distance. 如果 x 和 y 分别为两个实体的特征向量, 那么它们的特征距离 (feature distance) 为 $\|x - y\|_2$

Definition 3.4.4 — Nearest Neighbor. 给定向量 x , 一个组向量 Z_1, \dots, Z_m , 当 \hat{q}_j 满足:

$$\|x - z_j\|_2 \leq \|x - z_i\|_2, \quad i = 1, \dots, m$$

则称 z_j 是 x 的最近邻 (nearest neighbor)

3.5 Standard Derivation

Definition 3.5.1 — 算术平均值. 对于 n 维向量 x

$$\text{avg}(x) = \frac{\mathbf{1}^T x}{n}$$

Definition 3.5.2 — De-meaned Vector.

$$\tilde{x} = x - \text{avg}(x)\mathbf{1}$$

因此 $\text{avg } g(\tilde{x}) = 0$

Definition 3.5.3 — x 的标准差.

$$\text{std}(x) = \text{rms}(\tilde{x}) = \frac{\|x - (\mathbf{1}^T x/n) \mathbf{1}\|_2}{\sqrt{n}}$$

$\text{std}(x)$ 表示数据元素的变化程度. 对于常数 α , 当且仅当 $x = \alpha \mathbf{1}$ 时, $\text{std}(x) = 0$.

Theorem 3.5.1

$$\text{rms}(x)^2 = \text{avg}(x)^2 + \text{std}(x)^2$$

3.6 Angle

Definition 3.6.1 — 两个非零向量 a 和 b 之间的角 (angle).

$$\angle(a, b) = \arccos \left(\frac{a^T b}{\|a\|_2 \|b\|_2} \right)$$

$\angle(a, b)$ 的取值范围为 $[0, \pi]$, 且满足

$$a^T b = \|a\|_2 \|b\|_2 \cos(\angle(a, b))$$

在二维和三维向量之中, 这里的角与普通角度 (ordinary angle) 是一致的.

- $\theta = \frac{\pi}{2} = 90^\circ$: a 和 b 为正交, 写作 $a \perp b$ ($a^T b = 0$).
- $\theta = 0^\circ$: a 和 b 为同向的 ($a^T b = \|a\| \|b\|$).
- $\theta = 180^\circ$: a 和 b 为反向的 ($a^T b = -\|a\| \|b\|$).
- $\theta < \frac{\pi}{2} = 90^\circ$: a 和 b 成锐角 ($a^T b > 0$).
- $\theta > \frac{\pi}{2} = 90^\circ$: a 和 b 成钝角 ($a^T b < 0$).

Definition 3.6.2 — 球面的距离.

$$R\angle(a, b)$$

3.6.1 相关系数

给定向量 a 和 b , 其去均值向量为:

$$\tilde{a} = a - \text{avg}(a)\mathbf{1}, \tilde{b} = b - \text{avg}(b)\mathbf{1}$$

Definition 3.6.3 — a 和 b 的相关系数.

$$\rho = \frac{\tilde{a}^T \tilde{b}}{\|\tilde{a}\|_2 \|\tilde{b}\|_2} = \cos \angle(\tilde{a}, \tilde{b})$$

where $\tilde{a} \neq 0, \tilde{b} \neq 0$.

■ **Example 3.3** 高度相关的向量:

- 邻近地区的降雨时间序列.
- 类型密切相关文档的单词计数向量.
- 同行业中类似公司的日收益.

比较不相关的向量:

- 无关的向量.
- 音频信号 (比如, 在多轨录音中的不同轨).

负相关的向量:

- 深圳与墨尔本的每天气温变化

4. 优化问题初步

4.1 优化问题引入

Problem 4.1 假设 N 个样本向量 $x_1, \dots, x_N \in \mathbb{R}^n$, 需要找到中心向量 z 满足

$$\min_{z \in \mathbb{R}^n} \sum_{i=1}^N \|x_i - z\|_2^2$$

Definition 4.1.1 — 高阶无穷小记号 o . 设 x, y 是同一变化过程中的无穷小, 即 $x \rightarrow 0, y \rightarrow 0$, 如果它们极限

$$\lim_{x \rightarrow 0, y \rightarrow 0} \frac{y}{x} = 0$$

则称 y 是 x 的高阶无穷小, 记作 $y = o(x)$.

Corollary 4.1.1

$$\lim_{x \rightarrow 0, y \rightarrow 0} \frac{y}{Cx} = \frac{1}{C} \lim_{x \rightarrow 0, y \rightarrow 0} \frac{y}{x} = 0$$

也即则称 y 是 Cx 的高阶无穷小, 记作 $y = o(Cx)$.

Theorem 4.1.2 — 优化求解的必要条件. 假设函数 f 在 \hat{x} 可微, 则有

$$\hat{x} = \arg \min_{x \in \mathbb{R}^n} f(x) \Rightarrow \nabla f(\hat{x}) = 0$$

Proof. 假设函数 f 在 \hat{x} 一阶泰勒展开, 有

$$f(x) = f(\hat{x}) + \langle \nabla f(\hat{x}), x - \hat{x} \rangle + o(\|x - \hat{x}\|_2)$$

假设 $\delta f(\hat{x}) \neq 0$, 则令 $\tilde{x} = \hat{x} - t \nabla f(\hat{x})$, $t > 0$, 可得

$$f(\tilde{x}) = f(\hat{x}) - t\|\nabla f(\hat{x})\|_2^2 + o(t\|\nabla f(\hat{x})\|_2)$$

当 $t \rightarrow 0$ 则 $t\|\nabla f(\hat{x})\|_2 \rightarrow 0$, 高阶无穷小 $o(t\|\nabla f(\hat{x})\|_2) \rightarrow 0$

当 t 足够小时, 存在 $t\|\nabla f(\hat{x})\|_2 \geq o(t\|\nabla f(\hat{x})\|_2)$, 即

$$-t\|\nabla f(\hat{x})\|_2^2 + o(t\|\nabla f(\hat{x})\|_2) \leq 0$$

$$f(\tilde{x}) = f(\hat{x}) - t\|\nabla f(\hat{x})\|_2^2 + o(t\|\nabla f(\hat{x})\|_2) \leq f(\hat{x})$$

与 $\hat{x} = \arg \min_{\mathbf{R}^n} f(x)$ 矛盾.

$\nabla f(\hat{x}) = 0$, 是最优问题解的必要条件. 通常 $\nabla f(\hat{x}) = 0 \Leftrightarrow \hat{x} = \arg \min_{\mathbf{R}^n} f(x)$. ■

■ Example 4.1

$$f(x) = -x^2, \quad x \in \mathbb{R}, \hat{x} = \operatorname{argmin}_{\mathbb{R}} f(x)$$

$\nabla f(\hat{x}) = 0$, 则有 $-2\hat{x} = 0$, 即 $\hat{x} = 0$

$$f(\hat{x}) = 0 \geq f(x), \quad x \in \mathbb{R}$$

■

4.2 Convex Set

Definition 4.2.1 — 凸集. $\forall x, y \in \Omega, \alpha \in \mathbb{R}, 0 \leq \alpha \leq 1$ 有

$$\alpha x + (1 - \alpha)y \in \Omega$$

则定义域 $\Omega \in \mathbb{R}^n$ 称为凸的 (Convex) 集合
(域内两点连线之间都属于这个域)

Definition 4.2.2 — 凸函数. 设函数 $f(x)$ 定义于称为凸的定义域 $\Omega \in \mathbb{R}^n$ 满足

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y), \forall x, y \in \Omega, \alpha \in \mathbb{R}, 0 \leq \alpha \leq 1$$

称其为凸函数.

4.2.1 Examples for Convex Functions

■ Example 4.2

$$f(x) = x^2, x \in \mathbb{R}$$

$$\begin{aligned} f(\alpha x + (1 - \alpha)y) &= (\alpha x + (1 - \alpha)y)^2 \\ &= \alpha^2 x^2 + 2\alpha(1 - \alpha)xy + (1 - \alpha)^2 y^2 \\ &= \alpha x^2 + (1 - \alpha)y^2 + (\alpha^2 - \alpha)x^2 + (\alpha^2 - \alpha)y^2 + 2\alpha(1 - \alpha)xy \\ &= \alpha x^2 + (1 - \alpha)y^2 - \alpha(1 - \alpha)(x - y)^2 \\ &\leq \alpha x^2 + (1 - \alpha)y^2 = \alpha f(x) + (1 - \alpha)f(y) \end{aligned}$$

■

■ Example 4.3 $f(x) = \|x\|$, 其中 $\|\cdot\|$ 表示 \mathbb{R}^n 上的向量范数, $x \in \mathbb{R}^n$.

■

Proof.

$$\|ax + (1 - \alpha)y\| \leq \|ax\| + \|(1 - \alpha)y\| = |\alpha|\|x\| + |1 - \alpha|\|y\|$$

■

■ Example 4.4

$$f(x) = \|x\|_2^2, x \in \mathbb{R}^n$$

■

4.2.2 凸函数的充要条件

Theorem 4.2.1 — 可微函数 f 是凸函数的充要条件.

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle, \quad \forall x, y$$

Proof. 首先, 证明一维情况 $f : \mathbb{R} \rightarrow \mathbb{R}, \alpha \in [0, 1]$.

\Rightarrow 充分条件: $f(ax + (1 - \alpha)y) = f(x + (1 - \alpha)(y - x)) \leq \alpha f(x) + (1 - \alpha)f(y)$, 有

$$f(y) \geq f(x) + \frac{f(x + (1 - \alpha)(y - x)) - f(x)}{(1 - \alpha)(y - x)}(y - x)$$

令 $\alpha \rightarrow 1^-$, 则有 $f(y) \geq f(x) + f'(x)(y - x)$.

\Leftarrow 必要条件: 令 $y \neq x, z = ax + (1 - \alpha)y$ 则有

$$f(x) \geq f(z) + f'(z)(x - z), f(y) \geq f(z) + f'(z)(y - z)$$

可得

$$\begin{aligned} \alpha f(x) + (1 - \alpha)f(y) &\geq f(z) + \alpha f'(z)(x - z) + (1 - \alpha)f'(z)(y - z) \\ &= f(z) + f'(z)(\alpha x + (1 - \alpha)y - z) \\ &= f(z) \end{aligned}$$

证明 n 维情况 $f : \mathbb{R}^n \rightarrow \mathbb{R}$.

\Rightarrow 充分条件: 令 $g(t) = f(tx + (1 - t)y), t \in \mathbb{R}$, 则 $g'(t) = \langle \nabla f(tx + (1 - t)y), x - y \rangle$
由于 f 是凸函数, 证明 $g(t)$ 也是凸函数; 并可得 $g(0) \geq g(1) + g'(1)(-1)$, 得证.

\Leftarrow 必要条件: 与一维类似. (将 f' 改为 $\nabla f(z)^T$) ■

Theorem 4.2.2 如果可微函数 f 是凸函数, 则有

$$\hat{x} = \arg \min_{x \in \mathbb{R}^n} f(x) \Leftrightarrow \nabla f(\hat{x}) = 0$$

Proof. 已证 $\hat{x} = \arg \min_{x \in \mathbb{R}^n} f(x) \Rightarrow$ 可得 $\nabla f(\hat{x}) = 0$

只需证 $\nabla f(\hat{x}) = 0 \Rightarrow \hat{x} = \arg \min_{x \in \mathbb{R}^n} f(x)$.

由于函数 f 是可微凸的, 则有 $\forall x \in \mathbb{R}^n$,

$$\begin{aligned} f(x) &\geq f(\hat{x}) + \langle \nabla f(\hat{x}), x - \hat{x} \rangle \\ &= f(\hat{x}) + \langle 0, x - \hat{x} \rangle \geq f(\hat{x}) \end{aligned}$$

可得 $f(x) \geq f(\hat{x}), \hat{x} = \arg \min_{x \in \mathbb{R}^n} f(x)$. ■

4.3 向量偏导

Definition 4.3.1 — 向量对向量的导数.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, \mathbf{z} = \begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix}$$

$$\nabla f(\mathbf{z}) = \begin{bmatrix} \frac{\partial f(\mathbf{z})}{\partial z_1} \\ \vdots \\ \frac{\partial f(\mathbf{z})}{\partial z_n} \end{bmatrix}$$

■ **Example 4.5**

$$f(\mathbf{z}) = \mathbf{x}^T \mathbf{z} + \mathbf{z}^T \mathbf{z} = \sum_{i=1}^n \{x_i z_i + z_i^2\}$$

它的导数是

$$\nabla f(\mathbf{z}) = \begin{bmatrix} \frac{\partial f(\mathbf{z})}{\partial z_1} \\ \vdots \\ \frac{\partial f(\mathbf{z})}{\partial z_n} \end{bmatrix} = \begin{bmatrix} x_1 + 2z_1 \\ \vdots \\ x_n + 2z_n \end{bmatrix} = \mathbf{x} + 2\mathbf{z}$$

问题4.1中已知目标函数是凸函数. (见4.2, 4.3, 4.2.1)
则可以求解

Proof.

$$f(\mathbf{z}) = \sum_{i=1}^N \|x_i - \mathbf{z}\|_2^2 = \sum_{i=1}^N \langle x_i - \mathbf{z}, x_i - \mathbf{z} \rangle = \sum_{i=1}^N \{x_i^T x_i - 2x_i^T \mathbf{z} + \mathbf{z}^T \mathbf{z}\}$$

利用等价条件4.2.1

$$\nabla f(\mathbf{z}) = \sum_{i=1}^N \{-2x_i + 2\mathbf{z}\} = 0$$

(求导 4.3.1)

$$\mathbf{z} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i$$

■

另解:

Proof. $J(\mathbf{x}_0) = \sum_{i=1}^n \|x_0 - x_i\|^2$, 其中 $m = \frac{1}{n} \sum_{i=1}^n x_i$

$$\begin{aligned} J(\mathbf{x}_0) &= \sum_{i=1}^n \|(x_0 - m) - (x_i - m)\|^2 \\ &= \sum_{i=1}^n \|x_0 - m\|^2 - 2 \sum_{i=1}^n (x_0 - m)^T (x_i - m) + \sum_{i=1}^n \|x_i - m\|^2 \\ &= \sum_{i=1}^n \|x_0 - m\|^2 - 2(x_0 - m)^T \sum_{i=1}^n (x_i - m) + \sum_{i=1}^n \|x_i - m\|^2 \end{aligned}$$

因为

$$\sum_{i=1}^n (x_i - m) = \sum_{i=1}^n x_i - nm = \sum_{i=1}^n x_i - n \cdot \frac{1}{n} \sum_{i=1}^n x_i = 0$$

所以有

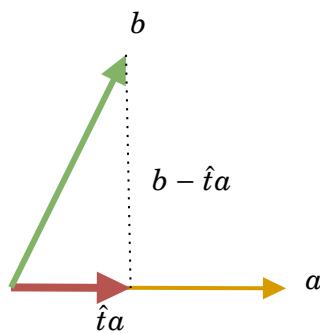
$$J(x_0) = \sum_{i=1}^n \|x_0 - m\|^2 + \sum_{i=1}^n \|x_i - m\|^2$$

即当 x_0 等于均值时，有最小均方.

■

4.4 标量优化问题：投影问题

Figure 4.1: Projection onto a line



Problem 4.2 假设 $a, b \in \mathbb{R}^n, a \neq 0, t \in \mathbb{R}$, 当 t 多大时, ta 到 b 之间的距离最小

$$\hat{t} = \min_t \|ta - b\|_2^2$$



t 是标量, ta 是向量.

定义

$$f(t) = \|ta - b\|_2^2 = \langle ta - b, ta - b \rangle = t^2 a^T a - 2ta^T b + b^T b$$

可以验证 $f(t)$ 满足凸函数的定义。

$$\nabla f(t) = 2ta^T a - 2a^T b = 0$$

$$\hat{t} = \frac{a^T b}{a^T a} = \frac{a^T b}{\|a\|_2^2}$$

4.5 Clustering

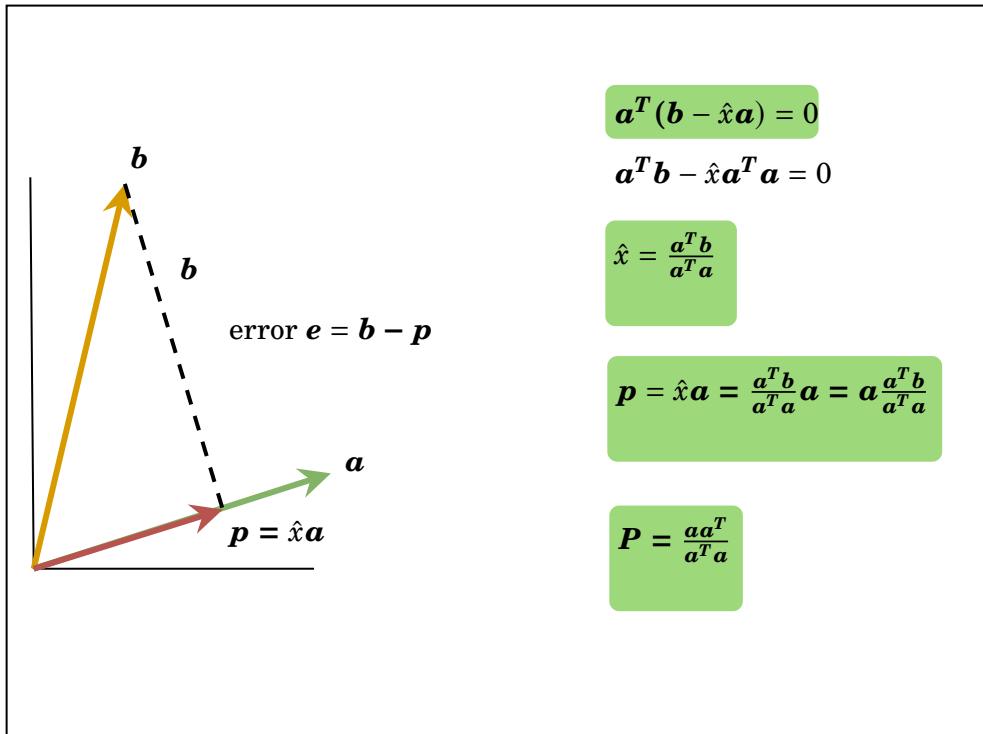
将物理或抽象对象的集合分成由类似特征组成的多个类的过程称为聚类 (*clustering*).

目标: 分成 k 个集合, 尽量使得同一个集合中的向量彼此接近.

Notation 4.1. 给定 N 个 n 维向量 $x_1, \dots, x_N \in \mathbb{R}^n$

- 标签 $c_i \in \{1, 2, \dots, k\}$ 表示向量 x_i 所属类别, 例如 $c_i = 2$ 表示 x_i 属于第 2 类.
- 对于 $j = 1, \dots, k$, $G_j = \{i : c_i = j\}$ 表示属于第 j 类的向量 x_i 的下标集合.

Figure 4.2: Projection onto a line from the prospective of linear algebra



- 向量 $z_j, j = 1, \dots, k$, 表示同属于 j 类的向量 $x_i, i \in G_j$ 的聚类中心.

聚类目标是找到向量 x_i 的“标签 c_i ”和“聚类中心 z_j ”

Problem 4.3

$$\min_{z_j} \sum_{i \in G_j} \|x_i - z_j\|_2^2, j = 1, \dots, k$$

$$c_i = \operatorname{argmin}_{j=\{1, \dots, k\}} \|x_i - z_j\|_2^2, i = 1, 2, \dots, N$$

k -means 算法是将 N 向量 $x_i \in \mathbb{R}^n$ 划分成 k 类的迭代聚类算法.

Algorithm 1: k -means Algorithm

- 在 N 个点中随机选取 k 个点, 分别作为聚类中心 z_j
- 更新聚类标签 c_i : 计算每个点 x_i 到 k 个聚类中心 z_j 的距离, 并将其分配到最近的聚类中心 z_j 所在的聚类中 $c_i = j$
- 更新聚类中心 z_j : 重新计算每个聚类现在的质心, 并以其作为新的聚类中心, 根据更新标签 c_i , 更新属于第 j 类下标集合 $G_j = \{i : c_i = j\}$, 重新计算 c_i 类的聚类中心 z_j
- 重复步骤 2、3, 直到所有聚类中心不再变化

Proof. 更新聚类标签 c_i :

$$\|x_i - z_j\|_2^2 = \operatorname{argmin} \{\|x_i - z_1\|_2^2, \|x_i - z_2\|_2^2, \dots, \|x_i - z_k\|_2^2\}$$

更新聚类中心 z_j :

$$\nabla f_j(z_j) = \sum_{i \in G_j} 2(x_i - z_j) = 0$$

$$z_j = \frac{1}{|G_j|} \sum_{i \in G_j} x_i$$

$|G_j|$ 表示集合 G_j 中元素的数目.

在每一次迭代中目标函数 J 都会下降, 直到聚类中心 z_1, \dots, z_k 和划分聚类标签集合 G_1, \dots, G_k 不再变化.

但是 k-means 算法依赖于初始随机生成的聚类中心, 只可得到目标函数 J 的局部最优.

解决方案: 使用不同的(随机的)初始聚类中心运行 k-means 算法若干次, 取目标函数 J 值最小的一次作为最终的聚类结果.

5. Linear Independence

5.1 线性相关、线性无关

Definition 5.1.1 — 线性相关 (linearly dependent). 定义：对于向量 $a_1, \dots, a_m \in \mathbb{R}^n$, 如果存在不全为零的数 $\beta_1, \dots, \beta_m \in \mathbb{R}$, 使得

$$\beta_1 a_1 + \dots + \beta_m a_m = 0$$

则称向量 a_1, \dots, a_m 是线性相关 (linearly dependent).

线性相关等价于至少有一个向量 a_i 是其它向量的线性组合.

Corollary 5.1.1 向量集 $\{a_1\}$ 是线性相关的, 当且仅当 $a_1 = 0$.

Corollary 5.1.2 向量集 $\{a_1, a_2\}$ 是线性相关的, 当且仅当其中一个 $a_1 = \beta a_2, \beta \neq 0$.

Definition 5.1.2 — 线性独立 (linearly independent). 如果 n 维向量集 $\{a_1, \dots, a_m\}$ 不是线性相关的, 即线性独立 (linearly independent), 也称线性无关, 即:

$$\beta_1 a_1 + \dots + \beta_m a_m = 0$$

当且仅当 $\beta_1 = \dots = \beta_m = 0$, 上述等式成立.

线性无关等价于不存在一个向量 a_i 是其它向量的线性组合.

Corollary 5.1.3 注: 一个 n 维向量集最多有 n 个线性无关的向量, 也就是说如果 n 维向量集有 $n+1$ 个向量, 那它们必线性相关

■ **Example 5.1** n 维单位向量 e_1, \dots, e_n 是线性独立的. ■

■ Example 5.2

$$a_1 = \begin{bmatrix} 1 \\ -2 \\ 0 \end{bmatrix}, \quad a_2 = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}, \quad a_3 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

$$\beta_1 a_1 + \beta_2 a_2 + \beta_3 a_3 = \begin{bmatrix} \beta_1 - \beta_2 \\ -2\beta_1 + \beta_3 \\ \beta_2 + \beta_3 \end{bmatrix} = 0$$

$$\beta_1 = \beta_2 = \beta_3 = 0$$

Theorem 5.1.4 假设 x 是线性无关向量 a_1, \dots, a_k 的线性组合:

$$x = \beta_1 a_1 + \dots + \beta_k a_k$$

则其系数 β_1, \dots, β_k 是唯一的, 即如果有:

$$x = \gamma_1 a_1 + \dots + \gamma_k a_k$$

则对于 $i = 1, \dots, k$, 有 $\beta_i = \gamma_i$.

Proof. 系数是唯一的原因:

$$(\beta_1 - \gamma_1) a_1 + \dots + (\beta_k - \gamma_k) a_k = x - x = 0$$

由于向量 a_1, \dots, a_k 线性无关, 有 $\beta_1 - \gamma_1 = \beta_k - \gamma_k = 0$.

5.2 Basis

■ **Definition 5.2.1 — 基 (Basis).** n 个线性独立的 n 维向量 a_1, \dots, a_n 的集合

■ **Definition 5.2.2 — 向量 b 在基底 a_1, \dots, a_n 下的分解.** 任何一个 n 维向量 b 都可以用它们的线性组合来表示

$$b = \beta_1 a_1 + \dots + \beta_n a_n$$

Proof. 同一向量的系数是唯一的.

■ **Example 5.3** e_1, \dots, e_n 是一组基, 那么 b 在此基底下的分解为

$$b = b_1 e_1 + \dots + b_n e_n, b = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \in \mathbb{R}^n$$

5.3 标准正交向量

■ **Definition 5.3.1 — Orthogonal Vectors.** 在 n 维向量集 a_1, \dots, a_k 中, 如果对于 $i \neq j$, 都有 $a_i \perp a_j$, 则称它们相互正交 (orthogonal).

Definition 5.3.2 — Orthonormal Vectors. 如果 n 维向量集 a_1, \dots, a_k 相互正交, 且每个向量的模长都为单位长度 1, 即对于 $i = 1, \dots, k$, 有 $\|a_i\|_2^2 = 1$, 则称它们是标准正交(orthonormal)的.

$$a_i^T a_j = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

Corollary 5.3.1 标准正交的向量集是线性无关的.

Corollary 5.3.2 根据线性无关的性质, 必有向量集向量个数 $k \leq n$.

Proof. The proof is by induction on the dimension n .

First consider a linearly independent collection a_1, \dots, a_k of 1-vectors. We must have $a_1 \neq 0$. This means that every element a_i of the collection can be expressed as a multiple $a_i = (a_i/a_1) a_1$ of the first element a_1 . This contradicts linear independence unless $k = 1$.

Next suppose $n \geq 2$ and the independence-dimension inequality holds for dimension $n - 1$.

Let a_1, \dots, a_k be a linearly independent list of n -vectors. We need to show that $k \leq n$. We partition the vectors as

$$a_i = \begin{bmatrix} b_i \\ \alpha_i \end{bmatrix}, \quad i = 1, \dots, k$$

where b_i is an $(n - 1)$ -vector and α_i is a scalar.

First suppose that $\alpha_1 = \dots = \alpha_k = 0$. Then the vectors b_1, \dots, b_k are linearly independent: $\sum_{i=1}^k \beta_i b_i = 0$ holds if and only if $\sum_{i=1}^k \beta_i a_i = 0$, which is only possible for $\beta_1 = \dots = \beta_k = 0$ because the vectors a_i are linearly independent. The vectors b_1, \dots, b_k therefore form a linearly independent collection of $(n - 1)$ -vectors. By the induction hypothesis we have $k \leq n - 1$, so certainly $k \leq n$.

Next suppose that the scalars α_i are not all zero. Assume $\alpha_j \neq 0$. We define a collection of $k - 1$ vectors c_i of length $n - 1$ as follows:

$$c_i = \begin{cases} b_i - \frac{\alpha_i}{\alpha_j} b_j, & i = 1, \dots, j - 1 \\ b_{i+1} - \frac{\alpha_{i+1}}{\alpha_j} b_j, & i = j, \dots, k - 1 \end{cases}$$

These $k - 1$ vectors are linearly independent: If $\sum_{i=1}^{k-1} \beta_i c_i = 0$ then

$$\sum_{i=1}^{j-1} \beta_i \begin{bmatrix} b_i \\ \alpha_i \end{bmatrix} + \gamma \begin{bmatrix} b_j \\ \alpha_j \end{bmatrix} + \sum_{i=j+1}^k \beta_{i-1} \begin{bmatrix} b_i \\ \alpha_i \end{bmatrix} = 0 \tag{5.1}$$

with

$$\gamma = -\frac{1}{\alpha_j} \left(\sum_{i=1}^{j-1} \beta_i \alpha_i + \sum_{i=j+1}^k \beta_{i-1} \alpha_i \right)$$

Since the vectors $a_i = (b_i, \alpha_i)$ are linearly independent, the eq. (5.1) only holds when all the coefficients β_i and γ are all zero. This in turns implies that the vectors c_1, \dots, c_{k-1} are linearly independent. By the induction hypothesis $k - 1 \leq n - 1$ so we have established that $k \leq n$ ■

Definition 5.3.3 — n 维向量的一个标准正交基. 当 $k = n$ 时, a_1, \dots, a_n 是 n 维向量的一个标准正交基.

Definition 5.3.4 — x 在标准正交基下的标准正交分解. 如果 a_1, \dots, a_n 是一个标准正交基, 对于任意维向量 x ;

$$x = (a_1^T x) a_1 + \dots + (a_n^T x) a_n$$

则称其为 x 在标准正交基下的标准正交分解.

这个分解可以用于计算不同标准正交基下的系数.

Proof. 由于正交向量的性质

$$a_i^T a_j = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

所以

$$a_i^T x = (a_1^T x) a_1^T a_1 + \dots + (a_i^T x) a_i^T a_i + \dots + (a_n^T x) a_n^T a_n = a_i^T x$$

■

5.4 Gram-Schmidt Algorithm

Algorithm 2: Gram-Schmidt Algorithm

Input: n 维向量 a_1, \dots, a_k

Output: 若这些向量线性无关, 返回标准正交基 q_1, \dots, q_k ; 若线性相关时判断 a_j 是 a_1, \dots, a_{j-1} 的线性组合

- 1 $q_1 = a_1 / \|a_1\|_2$
- 2 **while** $i = 2, \dots, k$ **do**
- 3 正交化: $\tilde{q}_i = a_i - (q_1^T a_i) q_1 - \dots - (q_{i-1}^T a_i) q_{i-1}$
- 4 检验线性相关: 如果 $\tilde{q}_i = 0$, 提前退出迭代
- 5 单位化: $q_i = \tilde{q}_i / \|\tilde{q}_i\|_2$
- 6 **end**

如果步骤 2 中未提前结束迭代, 那么 a_1, \dots, a_k 是线性独立的, 而且 q_1, \dots, q_k 是标准正交基.

如果在第 j 次迭代中提前结束, 说明 a_j 是 a_1, \dots, a_{j-1} 的线性组合, 因此 a_1, \dots, a_k 是线性相关的.

Theorem 5.4.1 q_1, \dots, q_{i-1}, q_i 是标准正交的.

Proof. 假设第 $i-1$ 次迭代成立, 即: $q_r \perp q_s, \forall r, s < i$.

正交化步骤保证有以下关系成立

$$\tilde{q}_i = a_i - (q_1^T a_i) q_1 - \dots - (q_{i-1}^T a_i) q_{i-1}$$

等式两边同时乘以 $q_j^T, j = 1, \dots, i-1$

$$\begin{aligned} q_j^T \tilde{q}_i &= q_j^T a_i - (q_1^T a_i) (q_j^T q_1) - \dots - (q_{i-1}^T a_i) (q_j^T q_{i-1}) \\ &= q_j^T a_i - q_j^T a_i \\ &= 0 \end{aligned}$$

$$\because q_j^T q_r = 0, j \neq r, q_j^T q_j = 1$$

$$\therefore \tilde{q}_i \perp q_1, \dots, \tilde{q}_i \perp q_{i-1}.$$

单位化步骤保证了 $q_i = \tilde{q}_i / \|\tilde{q}_i\|_2$, 即 q_1, \dots, q_i 是标准正交. ■

Algorithm 3: Gram-Schmidt Algorithm for Three Vectors

Input: Three independent vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$

Output: Three orthonormal vectors $\mathbf{q}_1 = \mathbf{A}/\|\mathbf{A}\|, \mathbf{q}_2 = \mathbf{B}/\|\mathbf{B}\|, \mathbf{q}_3 = \mathbf{C}/\|\mathbf{C}\|$.

1 Choose $\mathbf{A} = \mathbf{a}$

2

$$\mathbf{B} = \mathbf{b} - \frac{\mathbf{A}^T \mathbf{b}}{\mathbf{A}^T \mathbf{A}} \mathbf{A}$$

3

$$\mathbf{C} = \mathbf{c} - \frac{\mathbf{A}^T \mathbf{c}}{\mathbf{A}^T \mathbf{A}} \mathbf{A} - \frac{\mathbf{B}^T \mathbf{c}}{\mathbf{B}^T \mathbf{B}} \mathbf{B}$$

4 单位化

5.4.1 The Analysis of Gram-Schmidt Algorithm

假设 Gram-Schmidt 正交法未在第 i 次迭代提前终止:

Corollary 5.4.2 a_i 是 q_1, \dots, q_i 的一个线性组合.

$$a_i = \|\tilde{q}_i\|_2 q_i + (q_1^T a_i) q_1 + \dots + (q_{i-1}^T a_i) q_{i-1}$$

Proof.

$$\tilde{q}_i = a_i - (q_1^T a_i) q_1 - \dots - (q_{i-1}^T a_i) q_{i-1}$$

$$a_i = \tilde{q}_i + (q_1^T a_i) q_1 + \dots + (q_{i-1}^T a_i) q_{i-1}$$

注意有性质: $q_i = \tilde{q}_i / \|\tilde{q}_i\|_2$.

$$a_i = \|\tilde{q}_i\|_2 q_i + (q_1^T a_i) q_1 + \dots + (q_{i-1}^T a_i) q_{i-1}$$

■

则有

Corollary 5.4.3

$$q_i = \frac{a_i - (q_1^T a_i) q_1 - \dots - (q_{i-1}^T a_i) q_{i-1}}{\|\tilde{q}_i\|_2}$$

Corollary 5.4.4 q_i 是 a_1, \dots, a_i 的一个线性组合.

Proof. 归纳假设, 每个 q_{i-1} 都是 a_1, \dots, a_{i-1} 的线性组合:

$$\begin{aligned} q_2 &= \frac{a_2 - (q_1^T a_2) q_1}{\|q_2\|_2} \\ &= \frac{a_2 - (q_1^T a_2) \frac{a_1}{\|a_1\|_2}}{\|q_2\|_2} \\ q_3 &= \frac{a_3 - (q_1^T a_3) q_1 - (q_2^T a_3) q_2}{\|q_3\|_2} \end{aligned}$$

通过对 i 的归纳证明, 可得 q_i 是 a_1, \dots, a_i 的线性组合. ■

假设 Schmidt 正交法在第 j 次迭代提前终止:

Corollary 5.4.5 a_j 是 q_1, \dots, q_{j-1} 的一个线性组合.

$$a_j = (q_1^T a_j) q_1 + \dots + (q_{j-1}^T a_j) q_{j-1}$$

Proof.

$$\begin{aligned} \tilde{q}_i &= a_i - (q_1^T a_i) q_1 - \dots - (q_{i-1}^T a_i) q_{i-1} \\ 0 &= a_i - (q_1^T a_i) q_1 - \dots - (q_{i-1}^T a_i) q_{i-1} \\ a_i &= \|\tilde{q}_i\|_2 q_i + (q_1^T a_i) q_1 + \dots + (q_{i-1}^T a_i) q_{i-1} \end{aligned}$$

Corollary 5.4.6 a_j 是 a_1, \dots, a_{j-1} 的线性组合.

Proof. 每一个 q_1, \dots, q_{j-1} 都是 a_1, \dots, a_{j-1} 的线性组合.

因此 a_j 是 a_1, \dots, a_{j-1} 的线性组合. ■

Matrices

6	Matrices	36
6.1	Matrices	
6.2	矩阵运算	
6.3	Special Matrices and Matrices in Different Applications	
6.4	Gram 矩阵	
7	Matrices Norms	48
7.1	矩阵范数	
8	适定问题	52
8.1	The Definition of Well-posed Problem	
8.2	绝对误差的界限	
8.3	相对误差的界限	
9	Inverse of Matrices	54
9.1	Left Inverse, Right Inverse, Inverse	
9.2	Linear Equation Systems	
9.3	Fundamental Theorem of Linear Algebra	
9.4	Invertible Matrices	
9.5	转置和共轭转置的逆	
9.6	Gram Matrix 非奇异的性质	
9.7	伪逆	
10	Orthogonal Matrices	63
10.1	预备知识	
10.2	正交矩阵	
10.3	Permutation Matrices	
10.4	平面旋转	
10.5	Householder Matrix	
10.6	正交矩阵乘积	
10.7	具有正交矩阵的线性方程	
10.8	列标准正交的高矩阵	
10.9	值域范围、列空间	
11	QR 分解与 Householder 变换	71
11.1	Triangular Matrices	
11.2	QR Factorization	
11.3	QR 分解的应用	
11.4	QR Algorithm Using Gram-Schmidt Algorithm	
11.5	The Numerical Instability of QR Decomposition based on Gram-Schmidt Algorithm	
11.6	QR Decomposition Using Householder Transformation	
11.7	Householder 变换进行 QR 分解的 Q 因子	
11.8	Fast Orthogonalization (Givens and Householder)	
11.9	Recap: QR Decomposition	
12	LU 分解	97
12.1	Solving Linear Equation Systems	
12.2	LU 分解	
12.3	Problem of LU Decomposition	
12.4	$PA = LU$	
12.5	舍入误差的影响	
12.6	稀疏线性方程组	

6. Matrices

6.1 Matrices

Definition 6.1.1 — 矩阵. 矩阵是一个由数字构成的矩阵数组.

$$\begin{bmatrix} 0 & 1 & -2.3 & 0.1 \\ 1.3 & 4 & -0.1 & 0 \\ 4.1 & -1 & 0 & 1.7 \end{bmatrix} \quad \begin{pmatrix} 0 & 1 & -2.3 & 0.1 \\ 1.3 & 4 & -0.1 & 0 \\ 4.1 & -1 & 0 & 1.7 \end{pmatrix}$$

上述矩阵大小 (size) 为 3×4 , 矩阵的每一个元素 (element) 又称为系数 (coefficient);

Notation 6.1. 设 B_{ij} 表示矩阵 B 中第 i 行第 j 的元素

实数域中大小为 $m \times n$ 的矩阵集合写为 $\mathbb{R}^{m \times n}$

复数域中大小为 $m \times n$ 的矩阵集合写为 $\mathbb{C}^{m \times n}$

Definition 6.1.2 — 标量. 不区分一个 1×1 矩阵和一个标量.

Definition 6.1.3 — 向量. 不区分一个 $n \times 1$ 矩阵和一个向量.

Definition 6.1.4 — 行向量, 列向量. 一个 $1 \times n$ 矩阵被称为一个行向量.
一个 $n \times 1$ 矩阵被称为一个列向量.

Definition 6.1.5 — 高形, 宽形和方形矩阵. 一个大小为 $m \times n$ 的矩阵为:

- 高的, 如果 $m > n$
- 宽的, 如果 $m < n$
- 方的, 如果 $m = n$

Definition 6.1.6 — 分块矩阵. 分块矩阵的每一个元都是一个矩阵.

$$A = \begin{bmatrix} B & C \\ D & E \end{bmatrix}$$

其中 B, C, D, E 都是矩阵 (被称为矩阵 A 的子矩阵).

分块矩阵位于同一行的子矩阵行维度必须相等, 位于同一列的子矩阵列维度必须相等.

Definition 6.1.7 — 矩阵的列向量表示. 矩阵 $A \in \mathbb{R}^{m \times n}$, 可通过其列向量 (m -vector) 进行表示, 假设其列向量为 $a_1, \dots, a_n \in \mathbb{R}^m$, 则有

$$A = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}$$

Definition 6.1.8 — 矩阵的行向量表示. 矩阵 $A \in \mathbb{R}^{m \times n}$ 通过其行向量 b_1, \dots, b_m 进行表示

$$A = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}, b_i^T \in \mathbb{R}^n, i = 1, \dots, m$$

6.2 矩阵运算

Definition 6.2.1 — 矩阵数乘. 设矩阵 $A \in \mathbb{R}^{m \times n}$

$$\beta A = \begin{bmatrix} \beta A_{11} & \beta A_{12} & \cdots & \beta A_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ \beta A_{m1} & \beta A_{m2} & \cdots & \beta A_{mn} \end{bmatrix}, \beta \in \mathbb{R}$$

Definition 6.2.2 — 矩阵加法. 矩阵 $A, B \in \mathbb{R}^{m \times n}$ 的和为

$$A + B = \begin{bmatrix} A_{11} + B_{11} & A_{12} + B_{12} & \cdots & A_{1n} + B_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} + B_{m1} & A_{m2} + B_{m2} & \cdots & A_{mn} + B_{mn} \end{bmatrix}$$

Definition 6.2.3 — Transpose. 矩阵 A 的转置表示为 A^T

若 $A \in \mathbb{R}^{m \times n}$, 则 $A^T \in \mathbb{R}^{n \times m}$, 其被定义为: $(A^T)_{ij} = A_{ji}, i = 1, \dots, n; j = 1, \dots, m$

转置将原矩阵的行向量转化为列向量.

Corollary 6.2.1 — 转置的性质. 有如下性质:

- $(A^T)^T = A$
- 对称矩阵满足 $A^T = A$
- $(\beta A)^T = \beta A^T, (A + B)^T = A^T + B^T$

Definition 6.2.4 — 共轭转置. 矩阵 A 的共轭转置表示为 A^H

若 $A \in \mathbb{C}^{m \times n}$, 则 $A^H \in \mathbb{C}^{n \times m}$, 其被定义为: $(A^H)_{ij} = \bar{A}_{ji}, i = 1, \dots, n; j = 1, \dots, m$;

设矩阵 $A \in \mathbb{C}^{m \times n}$, 则其共轭转置为一个 $n \times m$ 矩阵

$$A^H = \begin{bmatrix} \bar{A}_{11} & \bar{A}_{21} & \cdots & \bar{A}_{m1} \\ \bar{A}_{21} & \bar{A}_{22} & \cdots & \bar{A}_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{A}_{1n} & \bar{A}_{2n} & \cdots & \bar{A}_{mn} \end{bmatrix}$$

Corollary 6.2.2 — 共轭转置的性质. 有如下性质:

- $(A^H)^H = A$
- Hermitian 矩阵满足 $A = A^H$
- $(\beta A)^H = \beta A^H$, $(A + B)^H = A^H + B^H$

Definition 6.2.5 — 矩阵乘法. 设矩阵 $A \in \mathbb{R}^{m \times p}$, $B \in \mathbb{R}^{p \times n}$, 那么矩阵 A 与 B 的乘积, 记作 $C = AB$, 矩阵 $C \in \mathbb{R}^{m \times n}$ 的第 i 行第 j 列元素 C_{ij}

$$C_{ij} = \sum_{k=1}^p A_{ik} B_{kj}$$

注意: 矩阵 A 的列大小必须等于 B 的行大小.

Corollary 6.2.3 — 矩阵乘法性质. 有如下性质:

- 结合律: $(AB)C = A(BC)$
- 分配律: $A(B + C) = AB + AC$
-

$$(AB)^T = B^T A^T, \quad (AB)^H = B^H A^H$$

- 一般情况下 $AB \neq BA$
- 对于方阵 A 有, $IA = AI = A$

Definition 6.2.6 — 分块矩阵乘法.

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} W & Y \\ X & Z \end{bmatrix} = \begin{bmatrix} AW + BX & AY + BZ \\ CW + DX & CY + DZ \end{bmatrix}$$

Definition 6.2.7 — 矩阵-向量乘积 Ax . 矩阵 $A \in \mathbb{R}^{m \times n}$ 和一个向量 $x \in \mathbb{R}^n$ 的积为

$$Ax = \begin{bmatrix} A_{11}x_1 + A_{12}x_2 + \cdots + A_{1n}x_n \\ A_{21}x_1 + A_{22}x_2 + \cdots + A_{2n}x_n \\ \vdots \\ A_{m1}x_1 + A_{m2}x_2 + \cdots + A_{mn}x_n \end{bmatrix}$$

Corollary 6.2.4 Ax 是矩阵 A 列向量的线性组合.

$$Ax = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = x_1 a_1 + \cdots + x_n a_n$$

可以引出 A 的 Row Picture 和 Column Picture 的概念。

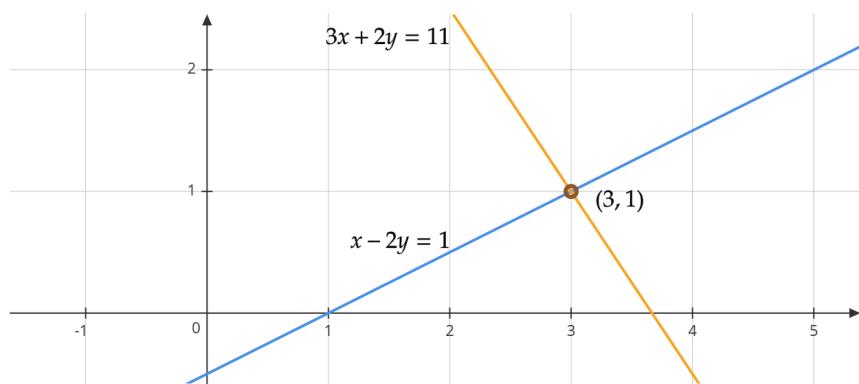
■ **Example 6.1**

$$\begin{cases} x - 2y = 1 \\ 3x + 2y = 11 \end{cases}$$

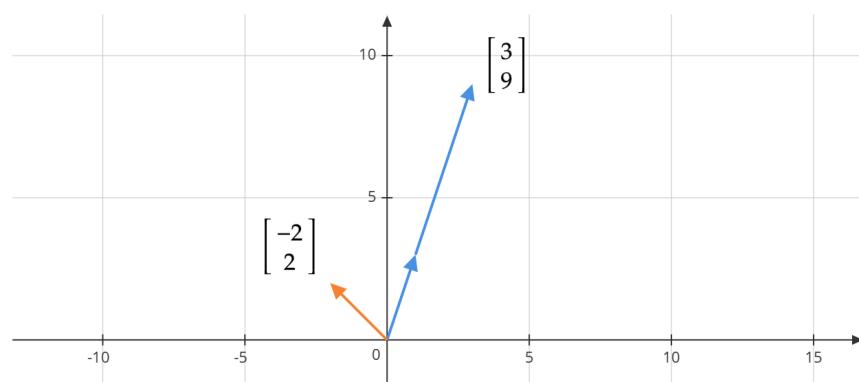
■

Figure 6.1: Row Picture and Column Picture for 6.1

(a) Row Picture



(b) Column Picture



Definition 6.2.8 — 矩阵-向量乘积函数 $f(x) = Ax$. 给定矩阵 $A \in \mathbb{R}^{m \times n}$, 定义函数 $f : R^n \rightarrow R^m, f(x) = Ax$, 其中 $A = [f(e_1) \dots f(e_n)]$.

Proof. 该函数为一个线性函数:

$$A(\alpha x + \beta y) = \alpha(Ax) + \beta(Ay)$$

任意一个线性函数都可以写成矩阵-向量乘积函数的形式

$$\begin{aligned} f(x) &= f(x_1 e_1 + x_2 e_2 + \dots + x_n e_n) \\ &= x_1 f(e_1) + x_2 f(e_2) + \dots + x_n f(e_n) \\ &= [f(e_1) \ \dots \ f(e_n)] \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \end{aligned}$$

因此 $f(x) = Ax$, 其中 $A = [f(e_1) \dots f(e_n)]$

■

6.2.1 Matrix Power

Definition 6.2.9 — Matrix Power. It makes sense to multiply a square matrix A by itself to form AA . We refer to this matrix as A^2 . Similarly, if k is a positive integer, then k copies of A multiplied together is denoted A^k . If k and l are positive integers, and A is square, then $A^k A^l = A^{k+l}$ and $(A^k)^l = A^{kl}$.

$$(A^{\ell+1})_{ij} = \sum_{k=1}^n A_{ik} (A^\ell)_{kj}$$

By convention we take $A^0 = I$, which makes the formulas above hold for all nonnegative integer values of k and l .

■ **Example 6.2 — Paths in a directed graph.**

$$A_{ij} = \begin{cases} 1 & \text{there is a edge from vertex } j \text{ to vertex } i \\ 0 & \text{otherwise} \end{cases}$$

■ **Example 6.3 — Linear dynamical system.**

$$x_{t+\ell} = A^\ell x_t$$

■

6.2.2 矩阵乘法的算法复杂度

一般矩阵的乘法

$C = AB$ with A of size $m \times p$ and B of size $p \times n$

The product matrix C has size $m \times n$, so there are mn elements to compute.

The i, j element of C is the inner product of row i of A with column j of B . This is an inner product of vectors of length p and requires $2p - 1$ flops. Therefore the total is $mn(2p - 1)$ flops, which we approximate as $2mnp$ flops. $O(mnp)$

稀疏矩阵的乘法

Suppose that A is $m \times p$ and sparse, and B is $p \times n$, but not necessarily sparse.

The inner product of the i th row a_i^T of A with the j th column of B requires no more than $2 \text{nnz}(a_i^T)$ flops.

Summing over $i = 1, \dots, m$ and $j = 1, \dots, n$ we get $2\text{nnz}(A)n$ flops.

If B is sparse, the total number of flops is no more than $2\text{nnz}(B)m$ flops.



Note that these formulas agree with the one given above, $2mnp$, when the sparse matrices have all entries nonzero.

三重矩阵相乘

$$D = ABC$$

with A of size $m \times n$, B of size $n \times p$, and C of size $p \times q$.

The matrix D can be computed in two ways, as $(AB)C$ and as $A(BC)$.

In the first method we start with AB ($2mnp$ flops) and then form $D = (AB)C$ ($2mpq$ flops), for a total of $2mp(n + q)$ flops.

In the second method we compute the product BC ($2npq$ flops) and then form $D = A(BC)$ ($2mnq$ flops), for a total of $2nq(m + p)$ flops.



You might guess that the total number of flops required is the same with the two methods, but it turns out it is not. The first method is less expensive when $2mp(n + q) < 2nq(m + p)$, i.e., when

$$\frac{1}{n} + \frac{1}{q} < \frac{1}{m} + \frac{1}{p}$$

■ **Example 6.4** As a more specific example, consider the product

$$ab^T c$$

where a, b, c are n vectors.

If we first evaluate the outer product ab^T , the cost is n^2 flops, and we need to store n^2 values. We then multiply the vector c by this $n \times n$ matrix, which costs $2n^2$ flops. The total cost is $3n^2$ flops.

If we first evaluate the inner product $b^T c$, the cost is $2n$ flops, and we only need to store one number (the result). Multiplying the vector a by this number costs n flops, so the total cost is $3n$ flops. For n large, there is a dramatic difference between $3n$ and $3n^2$ flops.

(The storage requirements are also dramatically different for the two methods of evaluating $ab^T c$: 1 number versus n^2 numbers.) ■

6.2.3 矩阵向量乘积复杂度

矩阵 $A \in \mathbb{R}^{m \times n}$ 和向量 $x \in \mathbb{R}^n$ 的乘积 $y = Ax$, 需要 $(2n - 1)m$ flops;

乘积 $y \in \mathbb{R}^m$, 每个元素需要做向量内积, 需要 $2n - 1$ flops;

当 n 足够大时, 复杂度近似于 $2mn$;

特殊情况:

- A 为对角矩阵: n flops
- A 为下三角矩阵: n^2 flops
- A 为稀疏矩阵时: flops $\ll 2mn$

6.3 Special Matrices and Matrices in Different Applications

Definition 6.3.1 — Zero Matrix. 所有元素都为 0 的矩阵.

记作

$$0, 0_{m \times n}$$

Definition 6.3.2 — 单位矩阵. 为方形矩阵, 其中对角线元素为 1, 其它元素为 0.

记作 I 或者 I_n .

Corollary 6.3.1 I_n 的每一列是一个单位向量, 例如

$$I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = [e_1 \ e_2 \ e_3]$$

Definition 6.3.3 — Symmetric Matrices.

$$A_{ij} = A_{ji}, A^T = A$$

Definition 6.3.4 — Hermitian Matrices. $A_{ij} = \bar{A}_{ji}, A^H = A$ (共轭复数).

Definition 6.3.5 — Diagonal Matrices. 对角线上元素不全为 0, 其余元素全为 0.

对角矩阵用于膨胀 (dilation).

Definition 6.3.6 — 下三角矩阵. 方形矩阵且当 $i < j$ 时 $A_{ij} = 0$.

Definition 6.3.7 — 上三角矩阵. 方形矩阵且当 $i > j$ 时 $A_{ij} = 0$

通过 Gram-Schmidt 正交化算法可以化为上三角或者下三角矩阵。

6.3.1 $f(x) = Ax$ 中的 A

引入上节 $f(x) = Ax$ (矩阵-向量乘积函数) 的概念,

■ **Example 6.5 — Permutation Matrices.** f 颠倒向量 x 中的元素的顺序, 一个线性函数 $f(x) = Ax$

$$A = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

是一个置换矩阵.

Proof.

$$Ax = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} x_3 \\ x_2 \\ x_1 \end{bmatrix}$$

■ **Example 6.6** f 对向量 x 中的元素进行升序排序, 非线性;

■ **Example 6.7** f 将向量 x 中的元素替换成相应的绝对值, 非线性;

■ **Example 6.8 — 反转矩阵.**

$$A = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 1 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 \end{bmatrix}$$

■

Proof.

$$Ax = \begin{bmatrix} x_n \\ x_{n-1} \\ \vdots \\ x_2 \\ x_1 \end{bmatrix}$$

■

■ **Example 6.9 — 循环移位矩阵.**

$$A = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}$$

■

Proof.

$$Ax = \begin{bmatrix} x_n \\ x_1 \\ x_2 \\ \vdots \\ x_{n-1} \end{bmatrix}$$

■

■ **Example 6.10 — 旋转矩阵.**

$$A = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

Ax 将向量 x 进行旋转, 角度为 θ .

■

■ **Example 6.11 — Reflection Matrices.** Suppose that y is the vector obtained by reflecting x through the line that passes through the origin, inclined θ radians with respect to horizontal. Then

$$y = \begin{bmatrix} \cos(2\theta) & \sin(2\theta) \\ \sin(2\theta) & -\cos(2\theta) \end{bmatrix} x$$

■

6.3.2 Selectors

- **Definition 6.3.8 — Selector matrices.**
- **Definition 6.3.9 — Downsampling.**

6.3.3 图论：节点弧关联矩阵

Definition 6.3.10 — 关联矩阵. 假设有向图 G 有 m 个顶点, n 条弧, 则关联矩阵 A 大小为 $m \times n$, 其中

$$A_{ij} = \begin{cases} 1 & \text{如果点 } i \text{ 是弧 } j \text{ 的终点} \\ -1 & \text{如果点 } i \text{ 是弧 } j \text{ 的起点} \\ 0 & \text{其它} \end{cases}$$

6.3.4 Convolution

Definition 6.3.11 — 一维卷积. 向量 $a \in \mathbb{R}^n$ 和向量 $b \in \mathbb{R}^m$ 的卷积是一个 $(n + m - 1)$ 维向量 $c \in \mathbb{R}^{m+n-1}$

$$c_k = \sum_{i+j=k+1} a_i b_j, \quad k = 1, \dots, n+m-1$$

记为 $c = a * b$

- **Example 6.12** 设 $n = 4, m = 3$

Figure 6.2: An example of convolution

a ₁	a ₂	a ₃	a ₄
----------------	----------------	----------------	----------------

b ₃	b ₂	b ₁
----------------	----------------	----------------

a ₁	a ₂	a ₃	a ₄
----------------	----------------	----------------	----------------

b ₃	b ₂	b ₁
----------------	----------------	----------------

$c_1 = a_1 b_1$

 $c_2 = a_1 b_2 + a_2 b_1$

 $c_3 = a_1 b_3 + a_2 b_2 + a_3 b_1$

 $c_4 = a_2 b_3 + a_3 b_2 + a_4 b_1$

 $c_5 = a_3 b_3 + a_4 b_2$

 $c_6 = a_4 b_3$

$$\begin{aligned} c_1 &= a_1 b_1 \\ c_2 &= a_1 b_2 + a_2 b_1 \\ c_3 &= a_1 b_3 + a_2 b_2 + a_3 b_1 \\ c_4 &= a_2 b_3 + a_3 b_2 + a_4 b_1 \\ c_5 &= a_3 b_3 + a_4 b_2 \\ c_6 &= a_4 b_3 \end{aligned}$$

Corollary 6.3.2 假设向量 a 和 b 分别是以下多项式的系数

$$p(x) = a_1 + a_2 x + \cdots + a_n x^{n-1}, q(x) = b_1 + b_2 x + \cdots + b_m x^{m-1}$$

则 $c = a^* b$ 是多项式 $p(x)q(x)$ 的系数.

$$p(x)q(x) = c_1 + c_2 x + \cdots + c_{m+n-1} x^{m+n-2}$$

Corollary 6.3.3 — 卷积性质. 有如下性质:

- 对称性: $a * b = b * a$
- 结合律: $(a * b) * c = a * (b * c)$
- 如果 $a * b = 0$, 则 $a = 0$, 或者 $b = 0$

Corollary 6.3.4 如果固定 a 或 b , 则 $c = a * b$ 是一个线性函数

■ **Example 6.13 — Toeplitz Matrix.** 4 维向量 a 和 3 维向量 b , 则 $c = a * b$

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{bmatrix} = \begin{bmatrix} a_1 & 0 & 0 \\ a_2 & a_1 & 0 \\ a_3 & a_2 & a_1 \\ a_4 & a_3 & a_2 \\ 0 & a_4 & a_3 \\ 0 & 0 & a_4 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} b_1 & 0 & 0 & 0 \\ b_2 & b_1 & 0 & 0 \\ b_3 & b_2 & b_1 & 0 \\ 0 & b_3 & b_2 & b_1 \\ 0 & 0 & b_3 & b_2 \\ 0 & 0 & 0 & b_3 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix}$$

■

6.3.5 多项式

Definition 6.3.12 — 多项式. 多项式 $p(t)$, 度为 $n - 1$, 系数为 x_1, x_2, \dots, x_n

$$p(t) = x_1 + x_2 t + x_3 t^2 + \cdots + x_n t^{n-1}$$

Definition 6.3.13 — Vandermonde Matrices. $p(t)$ 在 m 个点中 t_1, t_2, \dots, t_m 的值为

$$\begin{bmatrix} p(t_1) \\ p(t_2) \\ \vdots \\ p(t_m) \end{bmatrix} = \begin{bmatrix} 1 & t_1 & \cdots & t_1^{n-1} \\ 1 & t_2 & \cdots & t_2^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & t_m & \cdots & t_m^{n-1} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = Ax$$

矩阵 A 被称为 *Vandermonde* 矩阵.

6.3.6 Fourier Transform

Definition 6.3.14 — Discrete Fourier Transform (DFT). DFT 将 n 维复向量 x 映射为 n 维复向量 y ($\mathbb{C}^n \rightarrow \mathbb{C}^n$)

$$y_k = \sum_{\ell=1}^n x_\ell e^{-i \frac{2\pi}{n} (k-1)(\ell-1)}, k = 1, \dots, n$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega^{-1} & \omega^{-2} & \cdots & \omega^{-(n-1)} \\ 1 & \omega^{-2} & \omega^{-4} & \cdots & \omega^{-2(n-1)} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & \omega^{-(n-1)} & \omega^{-2(n-1)} & \cdots & \omega^{-(n-1)(n-1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix}$$

其中 $\omega = e^{2\pi i/n}$.

DFT 矩阵 W 的第 k 行第 l 列的元素为 $W_{kl} = \omega^{-(k-1)(l-1)}$.

Definition 6.3.15 — Discrete Inverse Fourier Transform.

$$x_\ell = \frac{1}{n} \sum_{k=1}^n y_k e^{i \frac{2\pi}{n} (k-1)(\ell-1)}, \ell = 1, \dots, n$$

6.3.7 Semi-Definite Matrices

Definition 6.3.16 — 半正定矩阵. 对称矩阵 $A \in \mathbb{R}^{n \times n}$ 称为半正定矩阵, 满足以下条件

$$x^T A x \geq 0 \quad \forall x \in \mathbb{R}^n$$

Definition 6.3.17 — 正定矩阵. 对称矩阵 $A \in \mathbb{R}^{n \times n}$ 称为正定矩阵, 满足以下条件

$$x^T A x > 0 \quad \forall x \neq 0$$

Definition 6.3.18 — 二次型. 如果对称矩阵 $A \in \mathbb{R}^{n \times n}$, 则 $x^T A x$ 是二次型

Proof.

$$x^T A x = \sum_{i=1}^n \sum_{j=1}^n x_i A_{ij} x_j = \sum_{i=1}^n A_{ii} x_i^2 + 2 \sum_{i>j} A_{ij} x_i x_j$$

■

■ **Example 6.14**

$$A = \begin{bmatrix} 9 & 6 \\ 6 & a \end{bmatrix}$$

$$x^T A x = 9x_1^2 + 12x_1 x_2 + ax_2^2 = (3x_1 + 2x_2)^2 + (a - 4)x_2^2$$

如果 $a > 4$, 矩阵 A 为正定矩阵:

$$x^T A x > 0 \quad \forall x \neq 0$$

如果 $a = 4$, 矩阵 A 为半正定矩阵, 但不是正定矩阵:

$$x^T A x \geq 0 \quad \forall x, \quad x^T A x = 0 \quad \exists x = \begin{bmatrix} 2 \\ -3 \end{bmatrix}$$

如果 $a < 4$, 矩阵 A 不是半正定矩阵:

$$x^T A x < 0 \quad \exists x = \begin{bmatrix} 2 \\ -3 \end{bmatrix}$$

■

Corollary 6.3.5 正定矩阵 A 都是非奇异的.

Proof.

$$Ax = 0 \quad \Rightarrow \quad x^T A x = 0 \quad \Rightarrow \quad x = 0$$

最后一步由正定性得到的. ($x^T A x > 0 \quad \forall x \neq 0$)

■

Theorem 6.3.6 — 正定矩阵对角元素性质. 正定矩阵 A 有正的对角元素.

$$A_{ii} = e_i^T A e_i > 0$$

Theorem 6.3.7 — 半正定矩阵对角元素性质. 每个半正定矩阵 A 都有非负的对角元素.

$$A_{ii} = e_i^T A e_i \geq 0$$

6.4 Gram 矩阵

Definition 6.4.1 — 实矩阵 A 的 Gram 矩阵.

$$G = A^T A = \begin{bmatrix} a_1^T \\ a_2^T \\ \vdots \\ a_n^T \end{bmatrix} [a_1, a_2, \dots, a_n] = \begin{bmatrix} a_1^T a_1 & a_1^T a_2 & \cdots & a_1^T a_n \\ a_2^T a_1 & a_2^T a_2 & \cdots & a_2^T a_n \\ \vdots & \vdots & \ddots & \vdots \\ a_n^T a_1 & a_n^T a_2 & \cdots & a_n^T a_n \end{bmatrix}$$

Definition 6.4.2 — 复矩阵的 A 的 Gram 矩阵.

$$G = A^H A = \begin{bmatrix} a_1^H a_1 & a_1^H a_2 & \cdots & a_1^H a_n \\ a_2^H a_1 & a_2^H a_2 & \cdots & a_2^H a_n \\ \vdots & \vdots & \ddots & \vdots \\ a_n^H a_1 & a_n^H a_2 & \cdots & a_n^H a_n \end{bmatrix}$$

Theorem 6.4.1 每个 Gram 矩阵都是半正定的.

Proof.

$$x^T A x = x^T B^T B x = \|Bx\|_2^2 \geq 0, \forall x$$

■

Theorem 6.4.2 如果 Gram 矩阵是正定的, 则要满足

$$x^T A x = x^T B^T B x = \|Bx\|_2^2 > 0 (\forall x \neq 0)$$

Corollary 6.4.3 如果 Gram 矩阵是正定的, 则 B 的列向量是线性无关的.

Proof.

$$\|Bx\|_2^2 > 0 (\forall x \neq 0)$$

所以 $\forall x \neq 0, Bx \neq 0$.

注意和线性无关 5.1.2 的定义进行参照.

■

7. Matrices Norms

7.1 矩阵范数

Definition 7.1.1 — Matrix Norm. 向量空间中存在一个函数 $\|\cdot\| : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$ 且满足以下条件:

- 齐次性: $\|\alpha A\| = |\alpha| \|A\|, \alpha \in \mathbb{R}$ 且 $A \in \mathbb{R}^{m \times n}$;
 - 三角不等式: $\|A + B\| \leq \|A\| + \|B\|, A, B \in \mathbb{R}^{m \times n}$;
 - 非负性: $\|A\| \geq 0, A \in \mathbb{R}^{m \times n}$ 且 $\|A\| = 0 \Leftrightarrow A = 0$;
- 则称 $\|\cdot\|$ 为矩阵范数.

向量空间 $\mathbb{R}^{m \times n}$ 矩阵范数:

■ **Example 7.1 — F-范数 (Frobenius norm).**

$$\|A\|_F = \left(\sum_{i=1}^n \sum_{j=1}^n a_{ij}^2 \right)^{\frac{1}{2}}$$

Proof.

$$\|A\|_F \geq 0$$

$$\|\alpha A\|_F = |\alpha| \|A\|_F, \alpha \in \mathbb{R}$$

$$\begin{aligned} \|A + B\|_F &= \left(\sum_{i=1}^n \sum_{j=1}^n (a_{ij} + b_{ij})^2 \right)^{\frac{1}{2}} \leq \left(\sum_{i=1}^n \sum_{j=1}^n (a_{ij})^2 \right)^{\frac{1}{2}} + \left(\sum_{i=1}^n \sum_{j=1}^n (b_{ij})^2 \right)^{\frac{1}{2}} \\ &= \|A\|_F + \|B\|_F \end{aligned}$$

Definition 7.1.2 — 从属于给定向量范数 $\|x\|_v$ 的矩阵范数. 设 $x \in \mathbb{R}^n, A \in \mathbb{R}^{m \times n}, \|\cdot\|_v$ 为一种向量范数. 则 $\frac{\|Ax\|_v}{\|x\|_v}$ 对所有 $x \neq 0$ 有最大值, 令

$$\|A\|_v = \max_{x \neq 0} \left\{ \frac{\|Ax\|_v}{\|x\|_v} \right\} = \max_{x \neq 0} \left\{ \left\| A \frac{x}{\|x\|_v} \right\|_v \right\} = \max_{\|y\|_v=1} \{\|Ay\|_v\}$$

即

$$\|A\|_v = \max_{x \neq 0} \left\{ \frac{\|Ax\|_v}{\|x\|_v} \right\}$$

$\|A\|_v$ 称为从属于给定向量范数 $\|x\|_v$ 的矩阵范数, 简称为从属范数或算子范数.

Proof. 可以验证 $\|A\|_v$ 满足矩阵范数定义.

$$\|A\|_v \geq 0$$

$$\|\alpha A\|_v = |\alpha| \|A\|_v, \alpha \in \mathbb{R}$$

$$\begin{aligned} \|A + B\|_v &= \max_{\|y\|_v=1} \|(A + B)y\|_v \\ &\leq \max_{\|y\|_v=1} \{\|Ay\|_v + \|By\|_v\} \\ &\leq \max_{\|y\|_v=1} \|Ay\|_v + \max_{\|y\|_v=1} \|By\|_v \\ &= \|A\|_v + \|B\|_v \end{aligned}$$

■



在本书中若未明确说明, $\|A\|$ 表示的是算子范数.

由定义 $\|A\|_v = \max_{x \neq 0} \left\{ \frac{\|Ax\|_v}{\|x\|_v} \right\}$ 可得

Definition 7.1.3 — 向量范数和算子范数相容.

$$\frac{\|Ax\|_v}{\|x\|_v} \leq \|A\|_v \Rightarrow \|Ax\|_v \leq \|A\|_v \|x\|_v$$

称向量范数和算子范数相容.

Theorem 7.1.1 — 算子范数服从乘法范数相容性. 对于 $A \in \mathbb{R}^{m \times n}, B \in \mathbb{R}^{n \times p}$

$$\begin{aligned} \|AB\|_v &= \max_{x \neq 0} \left\{ \frac{\|ABx\|_v}{\|x\|_v} \right\} \\ &\leq \max_{x \neq 0} \left\{ \frac{\|A\|_v \|Bx\|_v}{\|x\|_v} \right\} \\ &\leq \|A\|_v \max_{x \neq 0} \left\{ \frac{\|B\|_v \|x\|_v}{\|x\|_v} \right\} \\ &= \|A\|_v \|B\|_v \end{aligned}$$

算子范数服从乘法范数相容性.

根据向量的常用范数可以导出矩阵 $A \in \mathbb{R}^{m \times n}$ 的算子范数

Definition 7.1.4 — A 的列范数.

$$\|A\|_1 = \max_{x \neq 0} \left(\frac{\|Ax\|_1}{\|x\|_1} \right) = \max_{1 \leq j \leq n} \sum_{i=1}^m |a_{ij}|$$

Definition 7.1.5 — A 的行范数.

$$\|A\|_\infty = \max_{x \neq 0} \left(\frac{\|Ax\|_\infty}{\|x\|_\infty} \right) = \max_{1 \leq i \leq m} \sum_{j=1}^n |a_{ij}|$$

Definition 7.1.6 — A 的 2-范数.

$$\|A\|_2 = \max_{x \neq 0} \left(\frac{\|Ax\|_2}{\|x\|_2} \right) = \sqrt{\lambda_{\max}(A^T A)} \quad (7.1)$$

Proof. For any A Choose x to be the eigenvector of $A^T A$ with largest eigenvalue λ_{\max} . The ratio in equation 7.1 is $x^T A^T A x = x^T (\lambda_{\max}) x$ divided by $x^T x$. This is λ_{\max} .

No x can give a larger ratio. The symmetric matrix $A^T A$ has eigenvalues $\lambda_1, \dots, \lambda_n$ and orthonormal eigenvectors q_1, q_2, \dots, q_n . Every x is a combination of those vectors. Try this combination in the ratio and remember that $q_i^T q_j = 0$:

$$\frac{x^T A^T A x}{x^T x} = \frac{(c_1 q_1 + \dots + c_n q_n)^T (c_1 \lambda_1 q_1 + \dots + c_n \lambda_n q_n)}{(c_1 q_1 + \dots + c_n q_n)^T (c_1 q_1 + \dots + c_n q_n)} = \frac{c_1^2 \lambda_1 + \dots + c_n^2 \lambda_n}{c_1^2 + \dots + c_n^2}$$

The maximum ratio λ_{\max} is when all c 's are zero, except the one that multiplies λ_{\max} . ■



The ratio in equation 7.1 is the Rayleigh quotient for the symmetric matrix $A^T A$. Its maximum is the largest eigenvalue $\lambda_{\max}(A^T A)$. The minimum ratio is $\lambda_{\min}(A^T A)$. If you substitute any vector x into the Rayleigh quotient $x^T A^T A x / x^T x$, you are guaranteed to get a number between $\lambda_{\min}(A^T A)$ and $\lambda_{\max}(A^T A)$.



The norm $\|A\|$ equals the largest singular value σ_{\max} of A . The singular values $\sigma_1, \dots, \sigma_r$ are the square roots of the positive eigenvalues of $A^T A$. So certainly $\sigma_{\max} = (\lambda_{\max})^{1/2}$. Since U and V are orthogonal in $A = U \Sigma V^T$, the norm is $\|A\| = \sigma_{\max}$.

■ **Example 7.2** 求矩阵 A 的各种常用范数

$$A = \begin{pmatrix} 1 & 2 & 0 \\ -1 & 2 & -1 \\ 0 & 1 & 1 \end{pmatrix}$$

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}| = \max_{1 \leq j \leq n} \{2, 5, 2\} = 5$$

$$\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}| = \max_{1 \leq i \leq n} \{3, 4, 2\} = 4$$

由于 $\|A\|_2 = \sqrt{\lambda_{\max}(A^T A)}$, 因此先求 $A^T A$ 的特征值

$$A^T A = \begin{pmatrix} 1 & -1 & 0 \\ 2 & 2 & 1 \\ 0 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 & 0 \\ -1 & 2 & -1 \\ 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 0 & 1 \\ 0 & 9 & -1 \\ 1 & -1 & 2 \end{pmatrix}$$

特征方程为

$$\det(\lambda I - A^T A) = \begin{vmatrix} \lambda - 2 & 0 & -1 \\ 0 & \lambda - 9 & 1 \\ -1 & 1 & \lambda - 2 \end{vmatrix} = 0$$

可得 $A^T A$ 的特征值

$$\lambda_1 = 9.1428, \lambda_2 = 2.9211, \lambda_3 = 0.9361$$

! 对于 $\|A\|_2$ 需要计算 $\lambda_{\max}(A^T A)$, 直接根据特征方程计算特征值的算法复杂度太高.

8. 适定问题

8.1 The Definition of Well-posed Problem

In 1923, the French mathematician Hadamard introduced the notion of well-posed (适定) problem:

- A solution for the problem exists;
- The solution is unique;
- Perturbations in the data should cause small perturbations in the solution.

One of these conditions is not satisfied, the problem is said to be ill-posed (病态) and demands a special consideration.

■ **Example 8.1** 假设 A 是非奇异矩阵

$$Ax = b$$

如果将 b 为 $b + \Delta b$, 方程新的解 $x + \Delta x$, 则有:

$$A(x + \Delta x) = b + \Delta b$$

即

$$\Delta x = A^{-1} \Delta b$$

如果小的变化 Δb 导致小变化 Δx , 则称解是稳定的. 如果小的变化 Δb 导致大变化 Δx , 则称解不稳定的.

设

$$A = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 + 10^{-10} & 1 - 10^{-10} \end{bmatrix}, \quad A^{-1} = \begin{bmatrix} 1 - 10^{10} & 10^{10} \\ 1 + 10^{10} & -10^{10} \end{bmatrix}$$

若 $b = (1, 1)$, 方程 Ax 的解 $x = (1, 1)$. 如果将 b 改为 $b + \Delta b$, 那么 x 的变化量为

$$\Delta x = A^{-1} \Delta b = \begin{bmatrix} \Delta b_1 - 10^{10} (\Delta b_1 - \Delta b_2) \\ \Delta b_1 + 10^{10} (\Delta b_1 - \Delta b_2) \end{bmatrix}$$

很小变化 Δb 会导致非常大变化 Δx ! 由矩阵 A 定义的问题, 称为适定问题或病态问题.

8.2 绝对误差的界限

假设 A 是非奇异的, 并给出定义:

Notation 8.1.

$$x = A^{-1}b$$

$$\Delta x = A^{-1}\Delta b$$

$\|\Delta x\|$ 的上界为:

$$\|\Delta x\|_2 \leq \|A^{-1}\|_2 \|\Delta b\|_2$$

矩阵范数 $\|A^{-1}\|_2$ 小时, 当 $\|\Delta b\|_2$ 变化很小, $\|\Delta x\|_2$ 也很小; $\|A^{-1}\|_2$ 大时, $\|\Delta x\|_2$ 可能很大, 即使 $\|\Delta b\|_2$ 很小.

8.3 相对误差的界限¹

假设 $b \neq 0$; 因此 $x \neq 0$

$\|\Delta x\|_2/\|x\|_2$ 的上界为:

$$\begin{aligned} \|\Delta x\|_2 &= \|A^{-1}\Delta b\|_2 \leq \|A^{-1}\|_2 \|\Delta b\|_2 \text{(向量范数和算子范数相容)} \\ \Rightarrow \frac{\|\Delta x\|_2}{\|x\|_2} &\leq \frac{\|A^{-1}\|_2 \|\Delta b\|_2}{\|x\|_2} = \frac{\|A\|_2 \|A^{-1}\|_2 \|\Delta b\|_2}{\|x\|_2 \|A\|_2} \leq \frac{\|A\|_2 \|A^{-1}\|_2 \|\Delta b\|_2}{\|b\|_2} \end{aligned}$$

由 $\|b\|_2 = \|Ax\|_2 \leq \|A\|_2 \|x\|_2$, 可得

$$\frac{\|\Delta x\|_2}{\|x\|_2} \leq \|A\|_2 \|A^{-1}\|_2 \frac{\|\Delta b\|_2}{\|b\|_2}$$

$\|A\|_2 \|A^{-1}\|_2$ 小, 当 $\frac{\|\Delta b\|_2}{\|b\|_2}$ 相对变化很小时, $\frac{\|\Delta x\|_2}{\|x\|_2}$ 也变化很小;
 $\|A\|_2 \|A^{-1}\|_2$ 大, $\frac{\|\Delta x\|_2}{\|x\|_2}$ 可远大于 $\frac{\|\Delta b\|_2}{\|b\|_2}$.

Definition 8.3.1 — 非奇异矩阵 A 的条件数 (condition number). 条件数定义

$$\kappa(A) = \|A\|_2 \|A^{-1}\|_2$$

Corollary 8.3.1 — 非奇异矩阵 A 的条件数 (condition number) 性质. 有如下性质:

- 对于所有 A , 有 $\kappa(A) \geq 1$;
- 如果 $\kappa(A)$ 比较小 (接近 1), x 的相对误差接近 b 的相对误差;
- 如果 $\kappa(A)$ 比较大 (超过 100), x 的相对误差比 b 的相对误差大得多.

¹Reference: MathWorks Blog.

9. Inverse of Matrices

9.1 Left Inverse, Right Inverse, Inverse

■ **Definition 9.1.1** — A 的左逆. 当一个矩阵 X 满足

$$XA = I$$

X 被称为 A 的左逆; 当左逆存在时, 则称 A 是可左逆的;

如果左逆矩阵存在, 则左逆矩阵有无穷多个.

■ **Example 9.1**

$$A = \begin{bmatrix} -3 & -4 \\ 4 & 6 \\ 1 & 1 \end{bmatrix}$$

矩阵 A 是可左逆的, 其左逆矩阵有两个

$$B = \frac{1}{9} \begin{bmatrix} -11 & -10 & 16 \\ 7 & 8 & -11 \end{bmatrix} \quad C = \frac{1}{2} \begin{bmatrix} 0 & -1 & 6 \\ 0 & 1 & -4 \end{bmatrix}$$

■ **Definition 9.1.2** — A 的右逆. 当左逆存在时, 则称 A 是可左逆的;

如果右逆矩阵存在, 则右逆矩阵有无穷多个.

■ **Example 9.2**

$$B = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

矩阵 B 可右逆, 以下矩阵都是 B 的右逆

$$D = \frac{1}{2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \\ 1 & 1 \end{bmatrix}, E = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, G = \begin{bmatrix} 1 & -1 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

一个大小为 $m \times n$ 的矩阵, 其左逆或右逆的维度为 $n \times m$.

Theorem 9.1.1 A 的左逆为 X 当且仅当 X^T 是 A^T 的右逆.

Proof.

$$A^T X^T = (XA)^T = I$$

Theorem 9.1.2 A 的右逆为 X 当且仅当 X^T 是 A^T 的左逆.

Proof.

$$X^T A^T = (AX)^T = I$$

Theorem 9.1.3 如果矩阵 A 存在左逆和右逆, 则左逆和右逆一定相等

Proof.

$$\begin{aligned} &XA = I, AY = I \\ \Rightarrow &X = XI = X(AY) = (XA)Y = Y \\ \Rightarrow &X = Y \end{aligned}$$

Definition 9.1.3 — 逆 A^{-1} . 如果矩阵 A 存在左逆和右逆, 此时 X 称为矩阵的逆, 记作 A^{-1} 当矩阵的逆存在时, 则称矩阵 A 可逆.

9.2 Linear Equation Systems

Definition 9.2.1 — Linear Equation Systems. 有 n 个变量的 m 个方程为

$$\left\{ \begin{array}{l} A_{11}x_1 + A_{12}x_2 + \cdots + A_{1n}x_n = b_1 \\ A_{21}x_1 + A_{22}x_2 + \cdots + A_{2n}x_n = b_2 \\ \vdots \\ A_{m1}x_1 + A_{m2}x_2 + \cdots + A_{mn}x_n = b_m \end{array} \right.$$

写成矩阵形式为: $Ax = b$. 其中 A 为系数矩阵, x 为 n 维列向量.

该方程组可能无解, 有唯一解和无穷解.

9.2.1 线性方程组求解

Theorem 9.2.1 如果矩阵 A 可逆, 假设 X 是矩阵 A 的左逆, 则至多一个解, 如有解则 $x = Xb$.

Proof.

$$Ax = b \Rightarrow x = XAx = Xb$$

列满秩时 (下面证明), 列向量线性无关, 所以其零空间中只有零解, 方程 $Ax = b$ 可能有一个唯一解 (b 在 A 的列空间中, 此特解就是全部解, 因为通常的特解可以通过零空间中的向量扩展出一组解集, 而此时零空间只有 0 向量), 也可能无解 (b 不在 A 的列空间中)

Theorem 9.2.2 如果矩阵 A 可右逆, 假设 Y 是矩阵 A 的右逆, 则至少一个解, 即 $x = Yb$

Proof. 设 $x = Yb$

$$x = Yb \Rightarrow Ax = AYb = b$$

右逆就是研究 $m \times n$ 矩阵 A 行满秩的情况, 此时 $n > m = \text{rank}(A)$. 对称的, 其左零空间中仅有零向量, 即没有行向量的线性组合能够得到零向量. ($N(A^T) = \{0\}$) ■

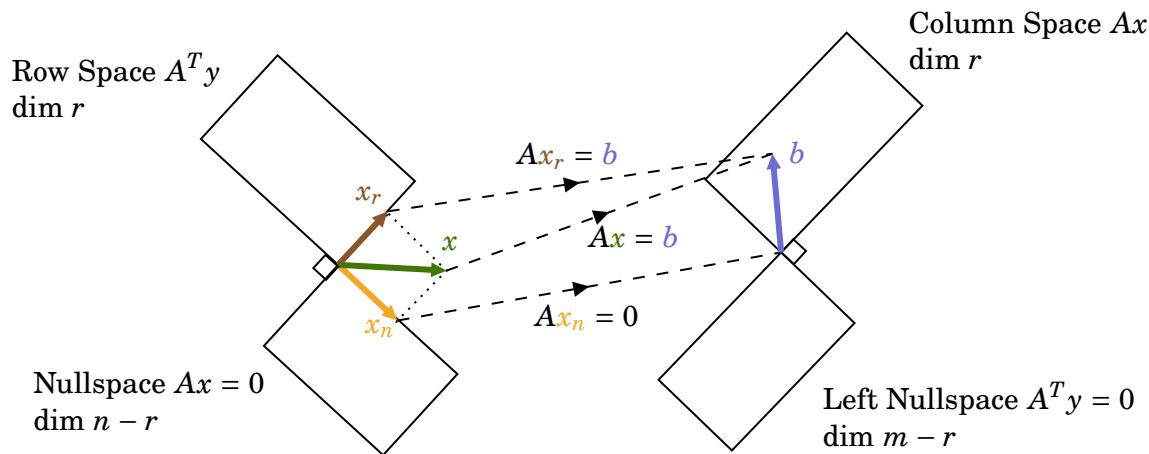
Theorem 9.2.3 如果矩阵 A 可逆的, 假设 X 是矩阵 A 的逆, 则

$$Ax = b \Rightarrow x = A^{-1}b$$

唯一解.

9.3 Fundamental Theorem of Linear Algebra

Figure 9.1: Four Subspace of Matrix A



$r = m$	$r = n$	Square and invertible	$Ax = b$ has 1 solution
$r = m$	$r < n$	Short and wide	$Ax = b$ has ∞ solutions
$r < m$	$r = n$	Tall and thin	$Ax = b$ has 0 or 1 solution
$r < m$	$r < n$	Not full rank	$Ax = b$ has 0 or ∞ solutions

The set of linear equations is called *over-determined* if $m > n$, *under-determined* if $m \leq n$, and *square* if $m = n$.

A set of equations with zero right-hand side, $Ax = 0$, is called a *homogeneous* set of equations. Any homogeneous set of equations has $x = 0$ as a solution.

9.4 Invertible Matrices

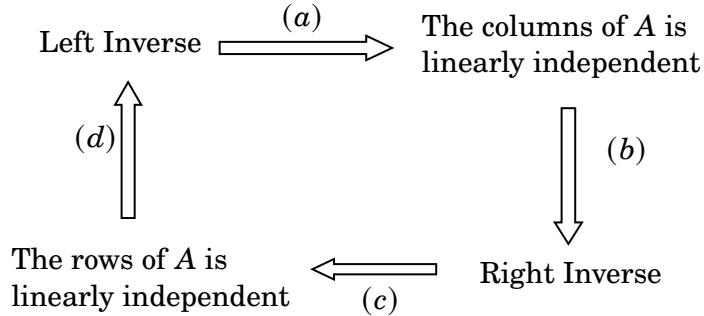
Theorem 9.4.1 对于方阵 $A \in \mathbb{R}^{n \times n}$, 以下条件都是等价的:

1. A 可左逆

2. A 的列向量线性无关
3. A 可右逆
4. A 的行向量线性无关
5. A 可逆

此时矩阵 A 为非奇异矩阵, 由条件 1 与 3, 可得 A 为可逆矩阵.

Proof. 可以通过以下方式证明:



- 性质 (a) 对任意矩阵 $A \in \mathbb{R}^{m \times n}$ 都成立
- 性质 (b) 对方阵矩阵 $A \in \mathbb{R}^{n \times n}$ 都成立
- 对于性质 (c) 与 (d), 可利用 A^T 证明

■

Theorem 9.4.2 (a): A 可左逆, 则 A 列向量线性无关.

Proof. 假设 A 的左逆是 B , 则

$$\begin{aligned} Ax = 0 &\Rightarrow BAx = 0 \\ &\Rightarrow Ix = 0 \end{aligned}$$

假设 A 的列向量 $A = [a_1, a_2, \dots, a_n]$

$$Ax = x_1a_1 + x_2a_2 + \dots + x_na_n = 0$$

则当该等式 $Ax = 0$ 成立时, 其解 $x = 0$, 则 A 的列向量线性无关.

Corollary 9.4.3 如果 $A \in \mathbb{R}^{m \times n}$ 有左逆, 则有 $m \geq n = r$.

即 A 是高或方的矩阵, 如 $A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$.

此时 A 的行向量可能线性相关, 而 A 的列向量线性无关. $N(A) = \{0\}$.

假设 A 的列向量 $A = [a_1, a_2, \dots, a_n]$

$$\begin{aligned} Ax &= x_1a_1 + x_2a_2 + \dots + x_na_n = b \\ Ay &= y_1a_1 + y_2a_2 + \dots + y_na_n = b \end{aligned}$$

$$Ax - Ay = A(x - y) = 0 \Rightarrow x = y$$

当 $b \in \mathbb{R}^m, b \notin \{y \mid y = Ax, x \in \mathbb{R}^n\}$ 时 (即 b 不在 A 的列空间, $m \geq n$ 时), 线性方程组无解. $Ax = b$ 至多一个解, 如有解则 $x = Xb$. ■

Theorem 9.4.4 矩阵的行秩等于列秩.

Proof. 令 A 是一个 $m \times n$ 的矩阵, 其列秩为 r . 因此矩阵 A 的列空间的维度是 r .

令 c_1, c_2, \dots, c_r 是 A 的列空间的一组基, 构成 $m \times r$ 矩阵 C 的列向量 $C = [c_1, c_2, \dots, c_r]$, 并使得 A 的每个列向量是 C 的 r 个列向量的线性组合.

由矩阵乘法的定义, 存在一个 $r \times n$ 矩阵 R , 使得 $A = CR$. (A 的 (i, j) 元素是 c_i 与 R 的第 j 个行向量的点积.)

现在, 由于 $A = CR$, A 的每个行向量是 R 的行向量的线性组合, 这意味着 A 的行向量空间被包含于 R 的行向量空间之中. 因此 A 的行秩 $\leq R$ 的行秩. 但 R 仅有 r 行, 所以 R 的行秩 $\leq r = A$ 的列秩. 这就证明了 A 的行秩 $\leq A$ 的列秩.

把上述证明过程中的“行”与“列”交换, 利用对偶性质同样可证 A 的列秩 $\leq A$ 的行秩. 更简单的方法是考虑 A 的转置矩阵 A^T , 则 A 的列秩 $= A^T$ 的行秩 $\leq A^T$ 的列秩 $= A$ 的行秩. 这证明了 A 的列秩等于 A 的行秩. 证毕. ■

Theorem 9.4.5 (c): 矩阵 $A \in \mathbb{R}^{m \times n}$ 有右逆 X , 则 A 行向量线性无关.

Proof.

$$X^T A^T = (AX)^T = I$$

则有 X^T 是 A^T 的左逆, A^T 的列向量线性无关.

即 $A^T \in \mathbb{R}^{n \times m}$.

Corollary 9.4.6 如果 $A \in \mathbb{R}^{m \times n}$ 有左逆, 则有 $r = m \leq n$.

即 A 是宽或方的矩阵.

此时 A 的列向量可能线性相关, 而 A 的行向量线性无关.

$N(A^T) = \{0\}$, $\dim N(A) = n - r$, $r = m$. ($Ax = b$ 有无穷解)

根据定理“矩阵的行秩等于列秩”, A^T 的列向量线性无关, 则矩阵 A 有 m 个线性无关列向量(行向量), 即通过 Gram-Schmidt 正交化可得 m 个正交基.

$\forall b \in \mathbb{R}^m$, 有 $b \in \{y \mid y = Ax, x \in \mathbb{R}^n\}$ ($m \leq n$), 方程 $Ax = b$ 有解, 其解为 $x = Xb$

Theorem 9.4.7 (b): 若方阵 A 列向量线性无关, 则 A 可右逆.

Proof. 假设 $A \in \mathbb{R}^{n \times n}$ 为方阵且列向量线性无关

$$A = [a_1, a_2, \dots, a_n]$$

则对于任意向量 $b \in \mathbb{R}^n$, 则向量组 $[a_1, a_2, \dots, a_n, b]$ 线性相关, 存在不全为 0 的系数, 使得以下等式成立

$$x_1 a_1 + x_2 a_2 + \dots + x_n a_n + x_{n+1} b = 0$$

因为 A 列向量线性无关, 则 $x_{n+1} \neq 0$ (假设 $x_{n+1} = 0$ 会推出违反线性无关假设的结论), 即 b 是 A 列向量的线性组合;

$$b = -\frac{x_1}{x_{n+1}} a_1 - \frac{x_2}{x_{n+1}} a_2 - \dots - \frac{x_n}{x_{n+1}} a_n$$

存在向量 $c_1, \dots, c_n \in \mathbb{R}^n$, 使得

$$Ac_1 = e_1$$

$$Ac_2 = e_2$$

...

$$Ac_n = e_n$$

则矩阵 $C = [c_1 c_2 \dots c_n]$ 是矩阵 A 的右逆, $AC = I$. ■

9.5 转置和共轭转置的逆

Theorem 9.5.1 — 转置 A^T 和共轭转置 A^H . 如果矩阵 A 为非奇异矩阵, 则其转置 A^T 和共轭转置 A^H 都为非奇异矩阵, 则有

$$(A^T)^{-1} = (A^{-1})^T, \quad (A^H)^{-1} = (A^{-1})^H$$

Proof.

$$(AA^{-1})^T = I \Rightarrow \underbrace{(A^{-1})^T}_{\text{the inverse of } A^T} A^T = I$$

Corollary 9.5.2 如果矩阵 A 和矩阵 B 都为非奇异矩阵, 则乘积 AB 也为非奇异矩阵

$$(AB)^{-1} = B^{-1}A^{-1}$$

Proof.

$$(AB) \underbrace{B^{-1}A^{-1}}_{\text{the inverse of } AB} = I$$

9.6 Gram Matrix 非奇异的性质

Gram 矩阵的定义见 6.4.1.

Corollary 9.6.1 — Gram Matrix 可逆等价于 A 列线性无关. 矩阵 $A \in \mathbb{R}^{m \times n}$, $G = A^T A$
矩阵 A 列向量线性无关 \Leftrightarrow Gram 矩阵 G 非奇异.

Proof. " \Rightarrow ":

假设矩阵 A 列向量线性无关, $A^T A$ 奇异. 则存在 $A^T Ax = 0, x \neq 0$, 可得 $x^T A^T Ax = \|Ax\|_2^2 = 0$, 即 $Ax = 0$ 与列向量线性无矛盾.

" \Leftarrow ":

假设 $A^T A$ 非奇异, 矩阵 A 列向量线性相关. 则有 $Ax = 0, x \neq 0$, 可得 $A^T Ax = 0$, 即 $A^T A$ 是奇异矩阵. ■

9.7 伪逆

Definition 9.7.1 — Pseudo-inverse.

$$A^\dagger = A^T (AA^T)^{-1}$$

$$A^\dagger = V\Sigma^+U^T = [v_1 \cdots v_r \cdots v_n] \begin{bmatrix} \sigma_1^{-1} & & \\ & \ddots & \\ & & \sigma_r^{-1} \end{bmatrix} [u_1 \cdots u_r \cdots u_m]^T$$

Theorem 9.7.1 伪逆 A^\dagger 为 A 的右逆

Proof.

$$AA^\dagger = AA^T (AA^T)^{-1} = (AA^T)^{-1} (AA^T) = I$$

■

Theorem 9.7.2 当 A 为方阵时，右逆等于矩阵的逆

Proof.

$$A^\dagger = A^T (AA^T)^{-1} = A^T A^{-T} A^{-1} = A^{-1}$$

■

Theorem 9.7.3

Corollary 9.7.4 以下三个结论为等价的，对于实矩阵 A

- A 是可左逆的
- A 的列向量线性无关
- $A^T A$ 为非奇异矩阵

Corollary 9.7.5 以下三个结论为等价的，对于实矩阵 A

- A 是可右逆的
- A 的行向量线性无关
- AA^T 为非奇异矩阵

By choosing good bases, A multiplies \mathbf{v}_i in the row space to give $\sigma_i \mathbf{u}_i$ in the column space. A^{-1} must do the opposite!

If $A\mathbf{v} = \sigma\mathbf{u}$ then $A^{-1}\mathbf{u} = \mathbf{v}/\sigma$. The singular values of A^{-1} are $1/\sigma$, just as the eigenvalues of A^{-1} are $1/\lambda$. The bases are reversed. The \mathbf{u} 's are in the row space of A^{-1} , the \mathbf{v} 's are in the column space.

The pseudoinverse A^+ is an n by m matrix. If A^{-1} exists, then A^+ is the same as A^{-1} . In that case $m = n = r$ and we are inverting $U\Sigma V^T$ to get $V\Sigma^{-1}U^T$.

The new symbol A^+ is needed when $r < m$ or $r < n$. Then A has no two-sided inverse, but it has a pseudoinverse A^+ with that same rank r :

$$A^+ \mathbf{u}_i = \frac{1}{\sigma_i} \mathbf{v}_i \quad \text{for } i \leq r \quad \text{and} \quad A^+ \mathbf{u}_i = \mathbf{0} \quad \text{for } i > r$$

The vectors $\mathbf{u}_1, \dots, \mathbf{u}_r$ in the column space of A go back to $\mathbf{v}_1, \dots, \mathbf{v}_r$ in the row space.

The other vectors $\mathbf{u}_{r+1}, \dots, \mathbf{u}_m$ are in the left nullspace, and A^\dagger sends them to zero. When we know what happens to all those basis vectors, we know A^\dagger .

Notice the pseudoinverse of the diagonal matrix Σ . Each σ in Σ is replaced by σ^{-1} in Σ^\dagger . The product $\Sigma^\dagger \Sigma$ is as near to the identity as we can get. It is a projection matrix, $\Sigma^\dagger \Sigma$ is partly I and otherwise zero. We can invert the σ 's, but we can't do anything about the zero rows and columns.

Figure 9.2: Ax^\dagger in the column space goes back to $A^\dagger A x^\dagger = x^\dagger$ in the row space

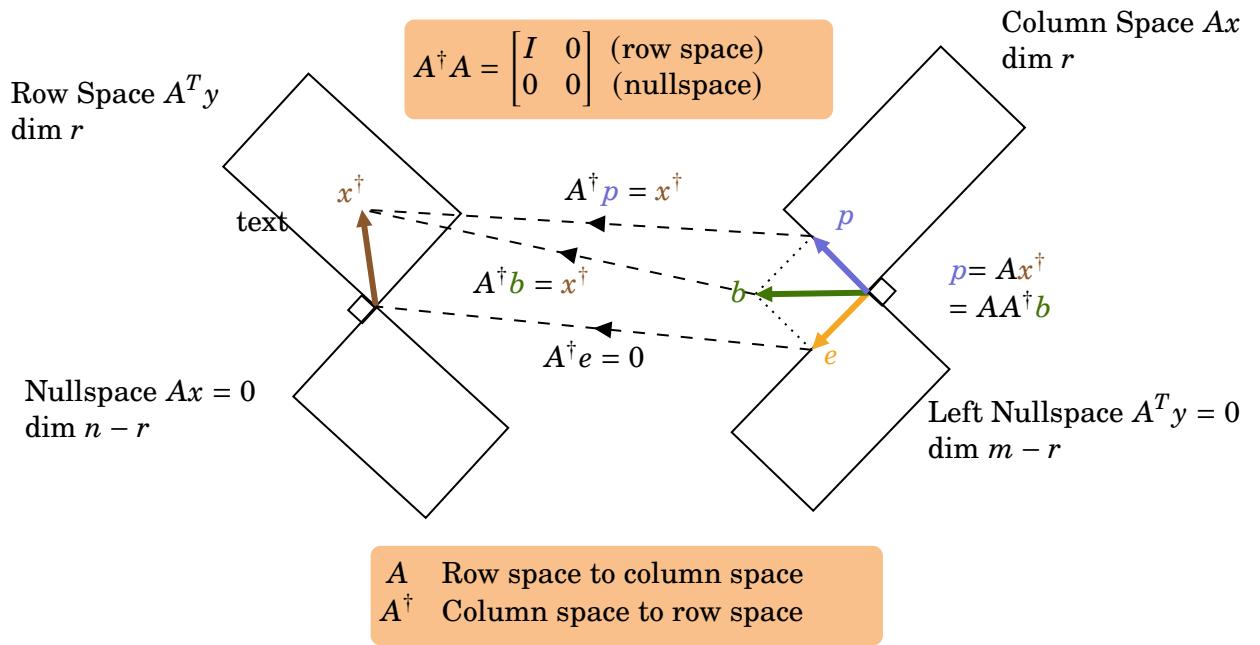
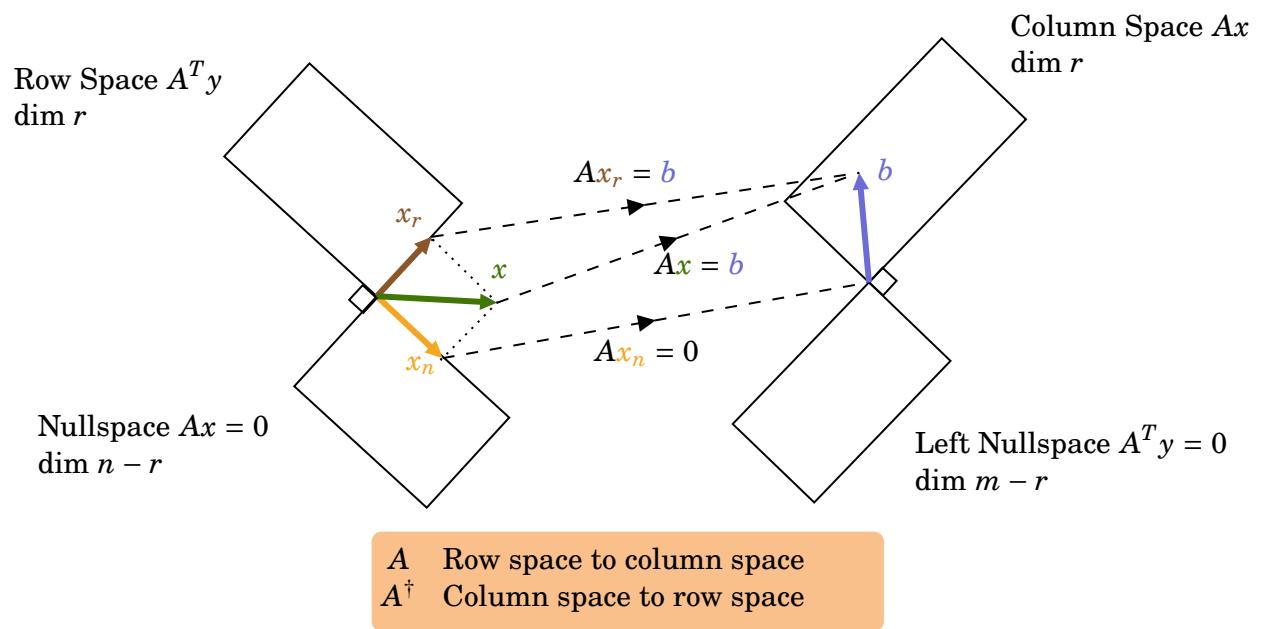


Figure 9.3: Projection from row space to column space



10. Orthogonal Matrices

10.1 预备知识

10.1.1 标准正交向量

参见 5.3.2。

10.1.2 Gram 矩阵与标准正交的关系

For the definition of Gram matrices, refer to 6.4.1. 关于它与非奇异的性质, 参见 9.6.

Theorem 10.1.1 如果 A 的 Gram 矩阵为单位矩阵, 则 $A \in \mathbb{R}^{m \times n}$ 具有标准正交列.

Proof.

$$\begin{aligned} A^T A &= [a_1 \ a_2 \ \cdots \ a_n]^T [a_1 \ a_2 \ \cdots \ a_n] \\ &= \begin{bmatrix} a_1^T a_1 & a_1^T a_2 & \cdots & a_1^T a_n \\ a_2^T a_1 & a_2^T a_2 & \cdots & a_2^T a_n \\ \vdots & \vdots & \ddots & \vdots \\ a_n^T a_1 & a_n^T a_2 & \cdots & a_n^T a_n \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \\ &= I \end{aligned}$$

■

10.1.3 矩阵-向量乘积与标准正交的关系

如果 $A \in \mathbb{R}^{m \times n}$ 具有标准正交列, 则线性函数 $f(x) = Ax$

Theorem 10.1.2 保持原内积.

$$\langle Ax, Ay \rangle = x^T y$$

Proof.

$$\langle Ax, Ay \rangle = (Ax)^T (Ay) = x^T A^T A y = x^T y$$

■

Theorem 10.1.3 保持原范数.

$$\|Ax\|_2 = \|x\|_2$$

Proof.

$$\|Ax\|_2 = \left((Ax)^T (Ax) \right)^{1/2} = \left(x^T x \right)^{1/2} = \|x\|_2$$

■

Theorem 10.1.4 保持原距离.

$$\|Ax - Ay\|_2 = \|x - y\|_2$$

Proof.

$$\|Ax - Ay\|_2 = \left((Ax - Ay)^T (Ax - Ay) \right)^{1/2} = \left((x - y)^T (x - y) \right)^{1/2} = \|x - y\|_2$$

■

Theorem 10.1.5 保持原角度.

$$\angle(Ax, Ay) = \angle(x, y)$$

Proof.

$$\angle(Ax, Ay) = \arccos \left(\frac{(Ax)^T (Ay)}{\|Ax\|_2 \|Ay\|_2} \right) = \arccos \left(\frac{x^T y}{\|x\|_2 \|y\|_2} \right) = \angle(x, y)$$

■

10.1.4 左可逆性与正交的关系

如果矩阵 $A \in \mathbb{R}^{m \times n}$ 有标准正交列，则：

Theorem 10.1.6 A 是左可逆的，其左逆为 A^T .

Proof. 根据定义：

$$A^T A = I$$

■

Theorem 10.1.7 A 有线性无关的列向量。

Proof.

$$Ax = 0 \Rightarrow A^T Ax = x = 0$$

■

Theorem 10.1.8 A 是高的或者方的，即 $m \geq n$.

Proof. 列向量 $a_1, a_2, \dots, a_n \in \mathbb{R}^m$, 由维度定理可得 $n \leq m$. ■

10.2 正交矩阵

Definition 10.2.1 — 正交矩阵. 所有列两两相互正交的方形实矩阵。

Theorem 10.2.1 — 正交矩阵满足非奇异性. 即如果矩阵 A 是正交的，则 A 是可逆的，左逆等于右逆，且它的逆为 A^T .

$$\left. \begin{array}{l} A^T A = I \\ A \text{ 是方的} \end{array} \right\} \Rightarrow AA^T = I$$

Corollary 10.2.2 A^T 也是一个正交矩阵。

Corollary 10.2.3 A 的行是标准正交的，即 a_i 范数为 1 且相互正交。

! 如果 $A \in \mathbb{R}^{m \times n}$ 有标准正交列以及 $m > n$, 则 $AA^T \neq I$ 。

10.3 Permutation Matrices

Notation 10.1. $\pi = (\pi_1, \pi_2, \dots, \pi_n)$ 为 $(1, 2, \dots, n)$ 的一个重新排序的排列。

将 π 与一个置换矩阵 $A \in \mathbb{R}^{n \times n}$ 联系起来:

$$A_{i\pi_i} = 1, \quad A_{ij} = 0 \text{ 如果 } j \neq \pi_i$$

Definition 10.3.1 — 置换. Ax 是 x 的一个置换:

$$Ax = (x_{\pi_1}, x_{\pi_2}, \dots, x_{\pi_n})$$

A 在每一行和每一列中都有一个等于 1 的元素。

Theorem 10.3.1 置换矩阵满足正交性，即所有置换矩阵都是正交的。

Corollary 10.3.2

$$A^T A = I$$

Proof. 因为 A 的每一行有一个元素等于 1

$$(A^T A)_{ij} = \sum_{k=1}^n A_{ki} A_{kj} = \begin{cases} 1 & i = j \\ 0 & \text{其它} \end{cases}$$

Corollary 10.3.3 $A^T = A^{-1}$ 是逆置换矩阵.

■ **Example 10.1** 若 $\{1, 2, 3, 4\}$ 的置换为:

$$(\pi_1, \pi_2, \pi_3, \pi_4) = (2, 4, 1, 3)$$

相应的置换矩阵及其逆矩阵为

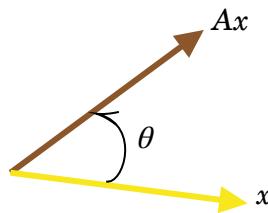
$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, A^{-1} = A^T = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

A^T 是与置换相关的置换矩阵

$$(\tilde{\pi}_1, \tilde{\pi}_2, \tilde{\pi}_3, \tilde{\pi}_4) = (3, 1, 4, 2)$$

10.4 平面旋转

Figure 10.1: An example of rotation



■ **Example 10.2 — Rotation Matrices in \mathbb{R}^2 .** 在一个 2 维平面的旋转可以用矩阵表示为

$$A = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

■ **Example 10.3 — Rotation Matrices in \mathbb{R}^3 .**

$$A = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix}$$

描述了在 \mathbb{R}^3 中 (x_1, x_3) 平面的旋转。

10.5 Householder Matrix

■ **Definition 10.5.1 — Householder matrix, reflector matrix.**

$$A = I - 2aa^T$$

其中，向量 a 满足 $\|a\|_2 = 1$ 。

Theorem 10.5.1 反射矩阵 (reflector matrix) 是对称的.

$$A^T = A$$

Theorem 10.5.2 反射矩阵 (reflector matrix) 是正交的。

Proof.

$$A^T A = (I - 2aa^T)(I - 2aa^T) = I - 4aa^T + 4aa^T aa^T = I$$

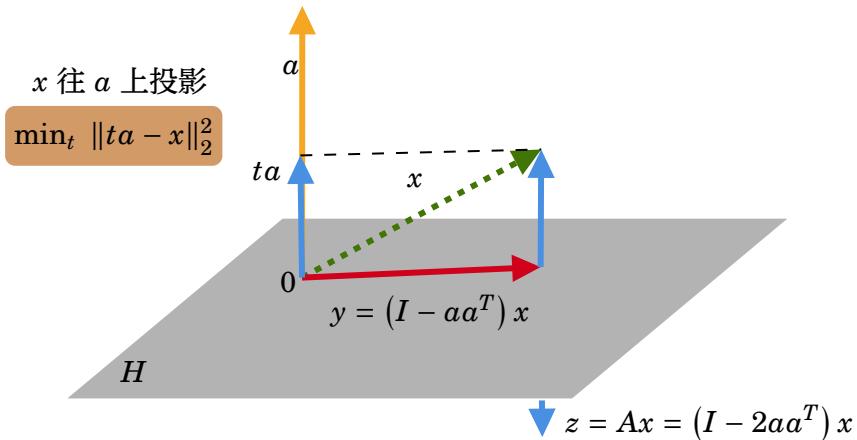
■

Theorem 10.5.3 反射矩阵是对合矩阵 (Involutory matrix)，即它的逆是它本身。

$$H = H^{-1}$$

10.5.1 The Geometry of Householder Transformation

Figure 10.2: Elementary Reflection



$H = \{u \mid a^T u = 0\}$ 是与 a 正交的向量的 (超) 平面。

Corollary 10.5.4 如果 $\|a\|_2 = 1$, x 在 H 上的投影为

$$y = x - (a^T x) a = x - a (a^T x) = (I - aa^T) x$$

Proof. 1. $y \in H$.

$$a^T y = a^T (x - a (a^T x)) = a^T x - (a^T a) (a^T x) = a^T x - a^T x = 0$$

2. 考虑任意 $z \in H (z \neq y)$, 证明 $\|x - z\| > \|x - y\|$

$$\begin{aligned}
\|x - z\|_2^2 &= \|x - y + y - z\|_2^2 \\
&= \|x - y\|_2^2 + 2(x - y)^T(y - z) + \|y - z\|_2^2 \\
&= \|x - y\|_2^2 + 2(a^T x) a^T (y - z) + \|y - z\|_2^2 \\
&= \|x - y\|_2^2 + \|y - z\|_2^2 \quad (\text{因为 } a^T y = a^T z = 0) \\
&\geq \|x - y\|_2^2
\end{aligned}$$

■

Corollary 10.5.5 x 通过超平面的反射由反射算子的乘积给出

$$z = y + (y - x) = (I - 2aa^T)x$$

10.6 正交矩阵乘积

若 $A_1, \dots, A_k \in \mathbb{R}^{n \times n}$ 是正交矩阵, 那么它们的乘积为:

$$A = A_1 A_2 \cdots A_k$$

Corollary 10.6.1 — 正交矩阵乘积的正交性.

$$\begin{aligned}
A^T A &= (A_1 A_2 \cdots A_k)^T (A_1 A_2 \cdots A_k) \\
&= A_k^T \cdots A_2^T A_1^T A_1 A_2 \cdots A_k \\
&= I
\end{aligned}$$

10.7 具有正交矩阵的线性方程

系数正交矩阵 $A \in \mathbb{R}^{n \times n}$ 的线性方程;

$$Ax = b$$

解为:

$$x = A^{-1}b = A^T b$$

10.7.1 The Complexity of the Multiplication Ax of Orthogonal Matrix A

可以在 $2n^2$ 个 flop 内计算矩阵向量乘法。

如果 A 有特殊性质, 代价将会小于 n^2 。例如,

- 置换矩阵: 0 flop。
- 反射算子 (给定 a): $4n$ flops。
- 平面旋转: $O(1)$ flop。

10.8 列标准正交的高矩阵

Theorem 10.8.1 假设矩阵 $A \in \mathbb{R}^{m \times n}$ 是高的 ($m > n$), 具有标准正交列, 则有 A^T 具有标准正交行.

Theorem 10.8.2 A^T 是 A 的一个左逆。

$$A^T A = I$$

Theorem 10.8.3 A 没有右逆。

10.9 值域范围、列空间

Definition 10.9.1 — 向量集合张成的空间。一个向量集合张成的空间是其所有线性组合的集合。

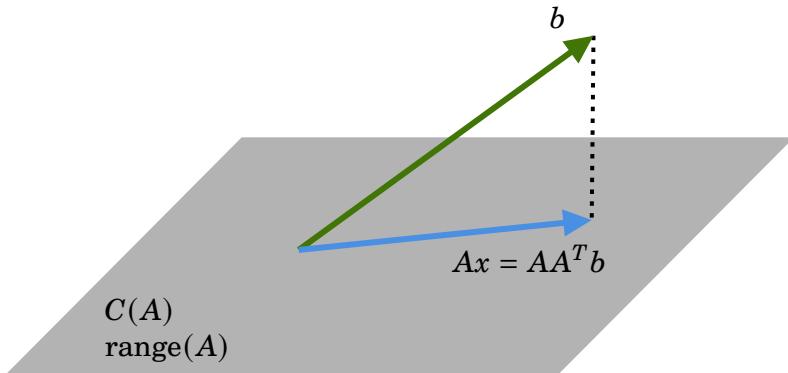
$$\text{span}(a_1, a_2, \dots, a_n) = \{x_1 a_1 + x_2 a_2 + \dots + x_n a_n \mid x \in \mathbb{R}^n\}$$

Definition 10.9.2 — A 的范围（列空间）。

$$\text{range}(A) = \{Ax \mid x \in \mathbb{R}^n\}$$

10.9.1 投影到列标准正交的矩阵 A 的列空间

Figure 10.3: Projection onto the column space of A , A has orthonormal columns



Problem 10.1 假设矩阵 $A \in \mathbb{R}^{m \times n}$ 具有标准正交列，求投影 b 在 $C(A)$ 上的投影。

即向量 Ax 与 b 有最短距离

$$\min_x \|Ax - b\|_2^2$$

$$\begin{aligned} f(x) &= \|Ax - b\|_2^2 \\ &= (Ax - b)^T (Ax - b) = x^T A^T A x - 2x^T A^T b + b^T b \\ &= x^T x - 2x^T A^T b + b^T b \quad (\because A^T A = I) \end{aligned}$$

$$\nabla f(x) = 2x - 2A^T b = 0 \Rightarrow x = A^T b$$

$AA^T b$ 称为向量 $b \in \mathbb{R}^m$ 在 $\text{range}(A)$ 上的正交投影。

Theorem 10.9.1

$$Ax = AA^T b \in \text{range}(A)$$

且是 b 在 $C(A)$ 上的投影。

Proof. 1.

$$Ax = AA^T b \in \text{range}(A)$$

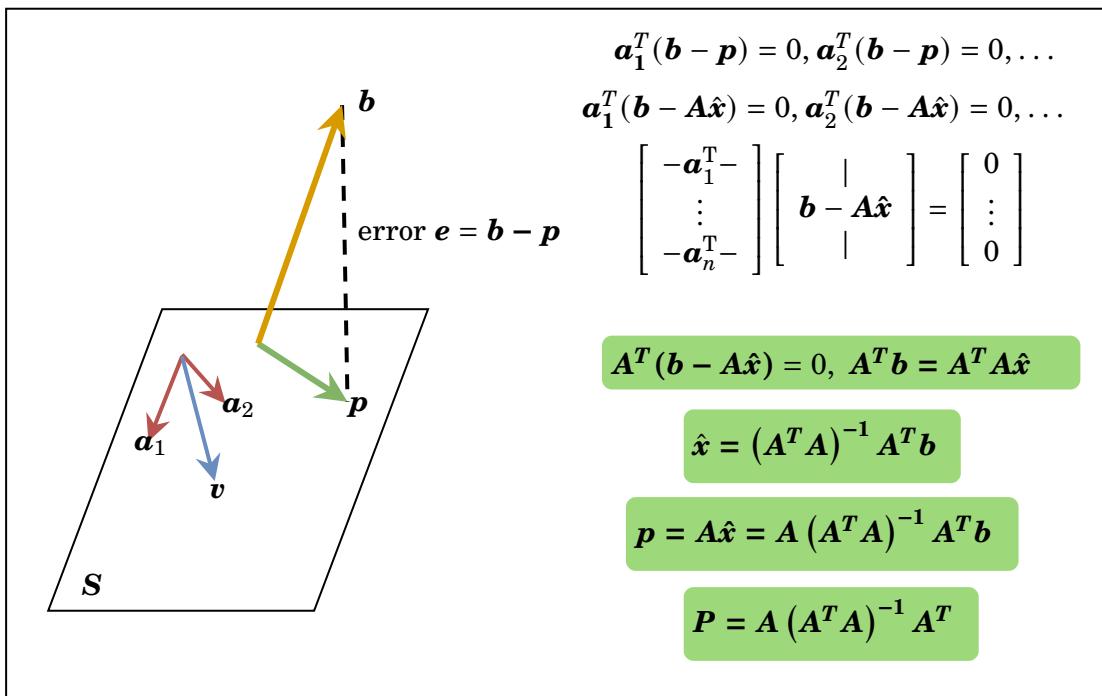
2. 可以证明 $\hat{x} = A^T b$ 满足 $\|A\hat{x} - b\| < \|Ax - b\|$, 对于所有 $x \neq \hat{x}$ 。
 b 到 $\text{range}(A)$ 内任意点 Ax 的距离的平方和为:

$$\begin{aligned}\|Ax - b\|_2^2 &= \|A(x - \hat{x}) + A\hat{x} - b\|_2^2 \quad (\text{其中 } \hat{x} = A^T b) \\ &= \|A(x - \hat{x})\|_2^2 + \|A\hat{x} - b\|_2^2 + 2(x - \hat{x})^T A^T (A\hat{x} - b) \\ &= \|A(x - \hat{x})\|_2^2 + \|A\hat{x} - b\|_2^2 \\ &= \|(x - \hat{x})\|_2^2 + \|A\hat{x} - b\|_2^2 \\ &\geq \|A\hat{x} - b\|_2^2\end{aligned}$$

当且仅当 $x = \hat{x}$, 等号成立。

第 3 行成立是因为 $A^T(A\hat{x} - b) = \hat{x} - A^T b = 0$ 。 ■

Figure 10.4: Projection of b into the column space of A , A is any matrix. Sourced from [Strang1993IntroductionTL]



11. QR 分解与 Householder 变换

11.1 Triangular Matrices

Definition 11.1.1 — Lower Triangular Matrices. 矩阵 $A \in \mathbb{R}^{n \times n}$ 为下三角 (*Lower Triangular*) 矩阵, $A_{ij} = 0, j > i$ 。

$$A = \begin{bmatrix} A_{11} & 0 & \cdots & 0 & 0 \\ A_{21} & A_{22} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{n-1,1} & A_{n-1,2} & \cdots & A_{n-1,n-1} & 0 \\ A_{n1} & A_{n2} & \cdots & A_{n,n-1} & A_{nn} \end{bmatrix}$$

Definition 11.1.2 — Upper Triangular Matrices. A^T 为上三角 (*Upper Triangular*) 矩阵。

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1,n-1} & A_{1,n} \\ 0 & A_{22} & \cdots & A_{2,n-1} & A_{2,n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & A_{n-1,n-1} & A_{n-1,n} \\ 0 & 0 & \cdots & 0 & A_{nn} \end{bmatrix}$$

Definition 11.1.3 — 单位上三角矩阵, 单位下三角矩阵. 对角元素 a_{ii} 都等于 1 的上 (下) 三角矩阵。

11.1.1 高斯消元法

Problem 11.1 当 A 是具有非零对角元素的下三角矩阵时, 解 $Ax = b$ 。

使用前向回代 (Forward Substitution) 算法求解。

时间复杂度: $1 + 3 + 5 + \dots + (2n - 1) = n^2$ flops

Problem 11.2 当 A 是具有非零对角元素的上三角矩阵, 解 $Ax = b$ 。

使用后向回代 (Back Substitution) 算法来求解。

Algorithm 4: Forward Substitution

Input: $A \in R^{n \times n}$ (A 是下三角矩阵), $b \in R^{n \times 1}$

Output: $x \in R^{n \times 1}$

$$\begin{aligned} 1 \quad x_1 &= \frac{b_1}{A_{11}} \\ 2 \quad x_2 &= \frac{b_2 - A_{21}x_1}{A_{22}} \\ 3 \quad x_3 &= \frac{b_3 - A_{31}x_1 - A_{32}x_2}{A_{33}} \\ 4 \quad \cdots \\ 5 \quad x_n &= \frac{b_n - A_{n1}x_1 - A_{n2}x_2 - \cdots - A_{n,n-1}x_{n-1}}{A_{nn}} \end{aligned}$$

Algorithm 5: Backward Substitution

Input: $A \in R^{n \times n}$ (A 是上三角矩阵), $b \in R^{n \times 1}$

Output: $x \in R^{n \times 1}$

$$\begin{aligned} 1 \quad x_n &= \frac{b_n}{A_{nn}} \\ 2 \quad x_{n-1} &= \frac{b_{n-1} - A_{n-1,n}x_n}{A_{n-1,n-1}} \\ 3 \quad x_{n-2} &= \frac{b_{n-2} - A_{n-2,n-1}x_{n-1} - A_{n-2,n}x_n}{A_{n-2,n-2}} \\ 4 \quad \cdots \\ 5 \quad x_1 &= \frac{b_1 - A_{12}x_2 - A_{13}x_3 - \cdots - A_{1n}x_n}{A_{11}} \end{aligned}$$

时间复杂度: $1 + 3 + \dots + 2n - 1 = n^2$ flops

11.1.2 The Inverses of Triangular Matrices

Theorem 11.1.1 对角元素非零的三角矩阵 A 是非奇异的, 即:

$$Ax = 0 \Rightarrow x = 0$$

Theorem 11.1.2 — 高斯消元法. A 的逆可以通过逐列解方程 $AX = I$ 来计算得到

$$A \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix} = \begin{bmatrix} e_1 & e_2 & \cdots & e_n \end{bmatrix}$$

Theorem 11.1.3 下三角矩阵的逆是下三角矩阵, 上三角矩阵的逆是上三角矩阵。

上/下三角矩阵 $A \in \mathbb{R}^{n \times n}$ 逆的复杂度

$$n^2 + (n-1)^2 + \dots + 1 \approx \frac{1}{3}n^3 \text{ flops}$$

11.2 QR Factorization

如果矩阵 $A \in \mathbb{R}^{m \times n}$ 的列向量线性无关, 则可以将其分解为

Theorem 11.2.1 — QR Factorization.

$$\begin{aligned} A_{n \times n} &= \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix} \\ &= \begin{bmatrix} q_1 & q_2 & \cdots & q_n \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1n} \\ 0 & R_{22} & \cdots & R_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_{nn} \end{bmatrix} \\ &= Q_{n \times n} R_{n \times n} \end{aligned}$$

向量 $q_1, \dots, q_n \in \mathbb{R}^m$ 是标准正交向量:

$$\|q_i\|_2 = 1, \quad q_i^T q_j = 0, \text{ if } i \neq j$$

对角元素 R_{ii} 是非零的。若 $R_{ii} < 0$, 改变 R_{ii}, \dots, R_{in} 和向量 q_i 的符号。大多数定义要求 $R_{ii} > 0$, 使得 Q 和 R 是唯一的。

Corollary 11.2.2 $Q \in \mathbb{R}^{m \times n}$ 具有标准正交列 ($Q^T Q = I$).

Corollary 11.2.3 如果 A 是方阵 ($m = n$), 则 Q 是正交的 ($Q^T Q = Q Q^T = I$).

Corollary 11.2.4 $R \in \mathbb{R}^{n \times n}$ 的上三角矩阵.

Corollary 11.2.5 R 是非奇异的 (对角元素是非零的).

Corollary 11.2.6

$$R = Q^{-1}A \Rightarrow R = Q^T A$$

QR 分解可通过 Gram-Schmidt 正交化法（参见5.4）、Householder 变换等进行。

Algorithm 6: QR Decomposition Using Gram-Schmidt Algorithm

- 1 设矩阵 A 的列向量依次为 a_1, a_2, \dots, a_n ，由于 A 为非奇异矩阵，则列向量线性无关
- 2 对列向量 a_1, a_2, \dots, a_n 按照 Gram-Schmidt 方法进行正交化，然后单位化
- 3 单位化得到的标准正交向量 q_1, q_2, \dots, q_n ，即得到标准正交矩阵 Q
- 4 根据 $R = Q^{-1}A \Rightarrow R = Q^T A$ ，得到上三角矩阵 R
- 5 QR 分解 $A = QR$

■ **Example 11.1** 矩阵 A 的 QR 分解过程

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 1 & -1 & 1 \\ 0 & 0 & 2 \end{bmatrix}$$

令 $a_1 = (1, 1, 0)^T, a_2 = (1, -1, 0)^T, a_3 = (0, 1, 2)^T$ ，由 Schmidt 方法正交单位化后，得到 $q_1 = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0\right)^T, q_2 = \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0\right)^T, q_3 = (0, 0, 1)^T$ 。

所以 $a_1 = \sqrt{2}q_1, a_2 = \sqrt{2}q_2, a_3 = \frac{1}{\sqrt{2}}q_1 - \frac{1}{\sqrt{2}}q_2 + 2q_3$ 。

$$A = QR = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 & \frac{1}{\sqrt{2}} \\ 0 & \sqrt{2} & -\frac{1}{\sqrt{2}} \\ 0 & 0 & 2 \end{bmatrix}$$

■ **Example 11.2**

$$\begin{bmatrix} -1 & -1 & 1 \\ 1 & 3 & 3 \\ -1 & -1 & 5 \\ 1 & 3 & 7 \end{bmatrix} = \begin{bmatrix} -1/2 & 1/2 & -1/2 \\ 1/2 & 1/2 & -1/2 \\ -1/2 & 1/2 & 1/2 \\ 1/2 & 1/2 & 1/2 \end{bmatrix} \begin{bmatrix} 2 & 4 & 2 \\ 0 & 2 & 8 \\ 0 & 0 & 4 \end{bmatrix}$$

$$= [q_1 \quad q_2 \quad q_3] \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ 0 & R_{22} & R_{23} \\ 0 & 0 & R_{33} \end{bmatrix}$$

$$= QR$$

11.2.1 QR 分解的存在唯一性¹

可以证明，QR 分解具有唯一性。

Theorem 11.2.7 — $m = n$ 矩阵 A 进行 QR 分解的唯一性. If $A = Q_1 R_1 = Q_2 R_2$ are two

¹Reference: SciComp, Kyle Kloster's Website

QR decompositions of full rank, square A , then

$$\begin{aligned}\mathbf{Q}_2 &= \mathbf{Q}_1 \mathbf{S} \\ \mathbf{R}_2 &= \mathbf{S} \mathbf{R}_1\end{aligned}$$

for some square diagonal \mathbf{S} with entries ± 1 .

If we require the diagonal entries of \mathbf{R} to be positive, then the decomposition is unique.

Theorem 11.2.8 — $m < n$ 矩阵 A 进行 QR 分解的唯一性. If

$$\mathbf{A} = \mathbf{Q}_1 \begin{bmatrix} \mathbf{R}_1 & \mathbf{N}_1 \end{bmatrix} = \mathbf{Q}_2 \begin{bmatrix} \mathbf{R}_2 & \mathbf{N}_2 \end{bmatrix}$$

are two QR decompositions of a full rank, $m \times n$ matrix \mathbf{A} with $m < n$, then

$$\mathbf{Q}_2 = \mathbf{Q}_1 \mathbf{S}, \quad \mathbf{R}_2 = \mathbf{S} \mathbf{R}_1, \quad \text{and} \quad \mathbf{N}_2 = \mathbf{S} \mathbf{N}_1$$

for square diagonal \mathbf{S} with entries ± 1 .

If we require the diagonal entries of \mathbf{R} to be positive, then the decomposition is unique.

Proof. Let $\mathbf{Q}_1 \begin{bmatrix} \mathbf{R}_1 & \mathbf{N}_1 \end{bmatrix} = \mathbf{Q}_2 \begin{bmatrix} \mathbf{R}_2 & \mathbf{N}_2 \end{bmatrix}$ with \mathbf{Q}_i being $m \times m$ and orthogonal, \mathbf{R}_i being $m \times m$ and upper triangular, and \mathbf{N}_i being an arbitrary $m \times (n - m)$ matrix.

Then multiplying through yields $\mathbf{Q}_1 \mathbf{R}_1 = \mathbf{Q}_2 \mathbf{R}_2$, two QR decompositions of a full rank, $m \times m$ matrix.

Using the theorem above, we get that $\mathbf{Q}_2 = \mathbf{Q}_1 \mathbf{S}$ and $\mathbf{R}_2 = \mathbf{S} \mathbf{R}_1$ for a diagonal matrix \mathbf{S} with entries ± 1 .

Looking at the right-most partition of the original product yields $\mathbf{Q}_1 \mathbf{N}_1 = \mathbf{Q}_2 \mathbf{N}_2$. But we've shown $\mathbf{Q}_2 = \mathbf{Q}_1 \mathbf{S}$, so now we have $\mathbf{Q}_1 \mathbf{N}_1 = \mathbf{Q}_1 \mathbf{S} \mathbf{N}_2$.

Left-multiplying by \mathbf{Q}_1^T and then by \mathbf{S} then proves $\mathbf{N}_2 = \mathbf{S} \mathbf{N}_1$, completing the theorem. ■

Theorem 11.2.9 — $m > n$ 矩阵 A 进行 QR 分解的唯一性. If $\mathbf{A} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{U}_1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_2 & \mathbf{U}_2 \end{bmatrix} \begin{bmatrix} \mathbf{R}_2 \\ 0 \end{bmatrix}$ are two QR decompositions of a full rank, $m \times n$ matrix \mathbf{A} with $m > n$, then

$$\mathbf{Q}_2 = \mathbf{Q}_1 \mathbf{S}, \quad \mathbf{R}_2 = \mathbf{S} \mathbf{R}_1, \quad \text{and} \quad \mathbf{U}_2 = \mathbf{U}_1 \mathbf{T}$$

for square diagonal \mathbf{S} with entries ± 1 , and square orthogonal \mathbf{T} .

If we require the diagonal entries of \mathbf{R} to be positive, then \mathbf{Q} and \mathbf{R} are unique.

Proof. Let \mathbf{A} be full rank and $m \times n$ with $m > n$. Suppose it has decompositions

$$\mathbf{A} = \tilde{\mathbf{Q}}_1 \tilde{\mathbf{R}}_1 = \tilde{\mathbf{Q}}_2 \tilde{\mathbf{R}}_2$$

for $m \times m$ orthogonal matrices $\tilde{\mathbf{Q}}_i$, $m \times n$ and upper-triangular matrices $\tilde{\mathbf{R}}_i$. (We know we can do this because the QR decomposition always exists).

Since $m > n$, we can write $\tilde{\mathbf{Q}}_i = [\mathbf{Q}_i \quad \mathbf{U}_i]$ and $\tilde{\mathbf{R}}_i = \begin{bmatrix} \mathbf{R}_i \\ 0 \end{bmatrix}$ where \mathbf{Q}_i is $m \times n$ and \mathbf{U}_i is $m \times (m - n)$. Then

$$\mathbf{A} = \tilde{\mathbf{Q}}_i \tilde{\mathbf{R}}_i = [\mathbf{Q}_i \quad \mathbf{U}_i] \begin{bmatrix} \mathbf{R}_i \\ 0 \end{bmatrix} = \mathbf{Q}_i \mathbf{R}_i$$

where \mathbf{R}_i is square, upper-triangular, invertible (because \mathbf{A} is full rank), and the columns of \mathbf{Q}_i are orthonormal so \mathbf{Q}_i satisfies $\mathbf{Q}_i^T \mathbf{Q}_i = \mathbf{I}$. Then we have

$$\mathbf{Q}_1 \mathbf{R}_1 = \mathbf{Q}_2 \mathbf{R}_2$$

and left-multiplying by \mathbf{Q}_2^T and right-multiplying by \mathbf{R}_1^{-1} yields

$$\mathbf{Q}_2^T \mathbf{Q}_1 = \mathbf{R}_2 \mathbf{R}_1^{-1}$$

Note that the right-hand side of Eqn (2) is upper-triangular (since \mathbf{R}_i is). On the other hand, left-multiplying Eqn (1) by \mathbf{Q}_1^T and right-multiplying by \mathbf{R}_2^{-1} gives $\mathbf{Q}_1^T \mathbf{Q}_2 = \mathbf{R}_1 \mathbf{R}_2^{-1}$, and taking the transpose yields a lower-triangular expression for $\mathbf{Q}_2^T \mathbf{Q}_1$. Therefore $\mathbf{Q}_1^T \mathbf{Q}_2 = \mathbf{R}_1 \mathbf{R}_2^{-1}$ is both lower- and upper-triangular, and so it is diagonal. Call it \mathbf{D} . Then right-multiplying Eqn (1) by \mathbf{R}_2^{-1} yields

$$\mathbf{Q}_2 \mathbf{R}_2 \mathbf{R}_2^{-1} = \mathbf{Q}_2 = \mathbf{Q}_1 \mathbf{R}_1 \mathbf{R}_2^{-1} = \mathbf{Q}_1 \mathbf{D}$$

and so $\mathbf{Q}_2 = \mathbf{Q}_1 \mathbf{D}$. Multiplying this by its transpose and using orthogonality of \mathbf{Q}_i we get $\mathbf{I} = \mathbf{Q}_2^T \mathbf{Q}_2 = (\mathbf{Q}_1 \mathbf{D})^T (\mathbf{Q}_1 \mathbf{D}) = \mathbf{D}^T \mathbf{Q}_1^T \mathbf{Q}_1 \mathbf{D} = \mathbf{D}^T \mathbf{D} = \mathbf{D}^2$. This proves $\mathbf{D}^2 = \mathbf{I}$, so $\mathbf{D} = \mathbf{S}$, a diagonal matrix with entries ± 1 . So $\mathbf{Q}_2 = \mathbf{Q}_1 \mathbf{S}$. Left multiplying Eqn (1) by $\mathbf{Q}_2^T = \mathbf{S} \mathbf{Q}_1^T$ then yields

$$\mathbf{S} \mathbf{Q}_1^T \mathbf{Q}_1 \mathbf{R}_1 = \mathbf{S} \mathbf{R}_1 = \mathbf{Q}_2^T \mathbf{Q}_2 \mathbf{R}_2 = \mathbf{R}_2$$

proving that $\mathbf{R}_2 = \mathbf{S} \mathbf{R}_1$. ■

11.2.2 复矩阵的 QR 分解

Theorem 11.2.10 如果 $A \in \mathbb{C}^{m \times n}$ 的列向量是线性无关的，则可以将其分解为

$$A = QR$$

$Q \in \mathbb{C}^{m \times n}$ 具有正交列。 $(Q^H Q = I)$

$R \in \mathbb{C}^{n \times n}$ 具有实非零对角元素的上三角矩阵。

大多数情况下，会优先选择对角线元素 R_{ii} 为正数。

如果没有特别说明，之后默认矩阵 A 都是实数的。

11.3 QR 分解的应用

可用 QR 分解求解以下问题：

- 线性方程
- 最小二乘问题
- 带约束的最小二乘问题

11.3.1 QR 分解和求解线性方程组 $Ax = b$

QR 分解的思想可以用于加快求逆矩阵速度。

Corollary 11.3.1

$$\begin{aligned} Ax = b &\Rightarrow x = A^{-1}b \\ QRx = b &\Rightarrow x = R^{-1}(Q^T b) \end{aligned}$$

Algorithm 8: Solving linear equations via QR factorization

Input: $n \times n$ invertible matrix A

- 1 QR factorization. Compute the QR factorization $A = QR$
- 2 Compute $Q^T b$
- 3 Back substitution. Solve the triangular equation $Rx = Q^T b$ using back substitution

对于普通矩阵使用 QR 分解求解线性方程组，正交分解需要 $2n^3$ flops，计算 $Q^T b$ 需要 $2n^2$ flops，第三步回代求解 $Rx = Q^T b$ 需要 n^2 flops。(总时间复杂度是 $O(n^3)$)。

对于稀疏矩阵求解线性方程组，时间复杂度接近 $\text{nnz}(A)$ 。内存使用和复杂度在很大程度上取决于系数矩阵的稀疏模式。内存使用量通常是 $\text{nnz}(A) + n$ 的适度倍数， $\text{nnz}(A) + n$ 是指定问题数据 A 和 b 所需的标量数量，通常远小于 $n^2 + n$ (如果 A 和 b 不是稀疏时存储它们所需的标量数)。求解稀疏线性方程的 flop 数通常也更接近 $\text{nnz}(A)$ ，而不是 n^3 (矩阵 A 不稀疏时的阶数)。

11.3.2 QR 分解和求解伪逆 A^\dagger 、逆 A^{-1}

Definition 11.3.1 — 线性无关列向量的矩阵 A 的伪逆。

$$A^\dagger = (A^T A)^{-1} A^T$$

Theorem 11.3.2

$$\begin{aligned} A^\dagger &= ((QR)^T (QR))^{-1} (QR)^T \\ &= (R^T Q^T QR)^{-1} R^T Q^T \\ &= (R^T R)^{-1} R^T Q^T \quad (Q^T Q = I) \\ &= R^{-1} R^{-T} R^T Q^T \quad (R \text{是非奇异的}) \\ &= R^{-1} Q^T \end{aligned}$$

Corollary 11.3.3 对于方阵非奇异矩阵 A , 其逆为

$$A^{-1} = (QR)^{-1} = R^{-1} Q^T$$

Corollary 11.3.4 对于方阵非奇异矩阵 $A = QR$

$$RA^{-1} = Q^T$$

Proof.

$$\begin{aligned} A &= QR \\ \Rightarrow AA^{-1} &= QRA^{-1} \\ \Rightarrow I &= Q \underbrace{RA^{-1}}_{Q^{-1}} \end{aligned}$$

■

Algorithm 9: Computing the inverse via QR factorization

Input: $n \times n$ invertible matrix A

- 1 QR factorization. Compute the QR factorization $A = QR$
- 2 **for** $i = 1, \dots, n$ **do**
- 3 | Solve the triangular equation $Rb_i = \tilde{q}_i$ using back substitution.
- 4 **end**

对于通过 QR 分解求 A^{-1} , QR 分解需要 $2n^3$ flops, n 次回代需要 n^3 flops, 时间复杂度是 $O(3n^3)$.

11.3.3 A 的列空间和 Q 的列空间相同

矩阵 $A \in \mathbb{R}^{m \times n}$ 的值域范围定义为:

$$\text{range}(A) = \{Ax \mid x \in \mathbb{R}^n\}$$

Theorem 11.3.5 假设 A 有线性无关的列向量, 且其 QR 因子为 Q, R , 则 Q 和 A 的值域范围相同 (有相同的列空间)。

即 Q 的列向量是标准正交的, 并且和 A 的列向量张成相同的空间。

Proof.

$$\begin{aligned} y \in \text{range}(A) &\Leftrightarrow y = Ax, x \in \mathbb{R}^n \\ &\Leftrightarrow y = QRx, z = Rx \\ &\Leftrightarrow y = Qz, z \in \mathbb{R}^n \\ &\Leftrightarrow y \in \text{range}(Q) \end{aligned}$$

■

11.3.4 往 A 列空间上的投影也是往 Q 列空间上的投影

结合 $A = QR$ 和 $A^\dagger = R^{-1}Q^T$, 可得:

Theorem 11.3.6

$$AA^\dagger = QRR^{-1}Q^T = QQ^T$$

Theorem 11.3.7

$$R^T R \hat{x} = R^T Q^T b \text{ or } R \hat{x} = Q^T b \text{ or } \hat{x} = R^{-1} Q^T b$$

可以用于求解最小二乘法, 而不用直接求解 $Ax = b$ (当 A 不可逆时)

注意在 AA^\dagger 中乘积的顺序与 $A^\dagger A = I$ 的差异。

$$\begin{aligned} & \min_y \|Qy - x\|_2^2 \\ \Rightarrow & Q^T(Qy - x) = 0 \\ \Rightarrow & Q^T Qy = Q^T x \\ \Rightarrow & y = Q^T x \end{aligned}$$

$QQ^T x$ 是 x 在 Q 值域 (列空间) 上的投影，也是往 A 的列空间上的投影 (见10.9.1)。

Figure 11.1: Projecting onto the column space of A is also projecting onto the column space of Q

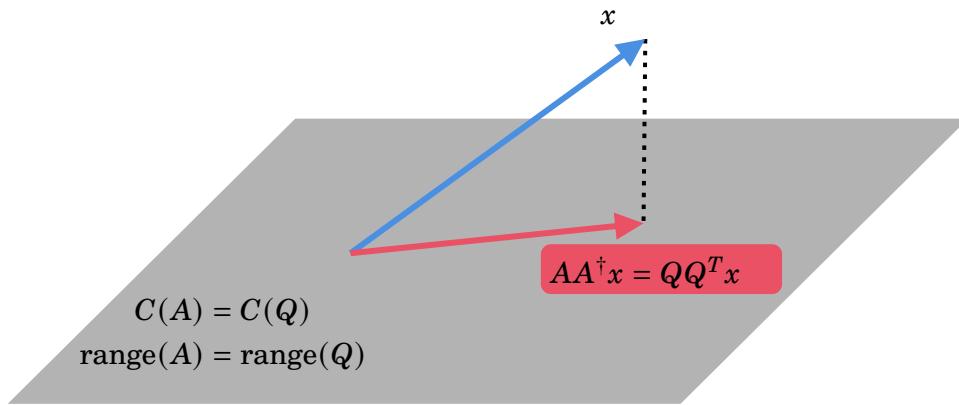
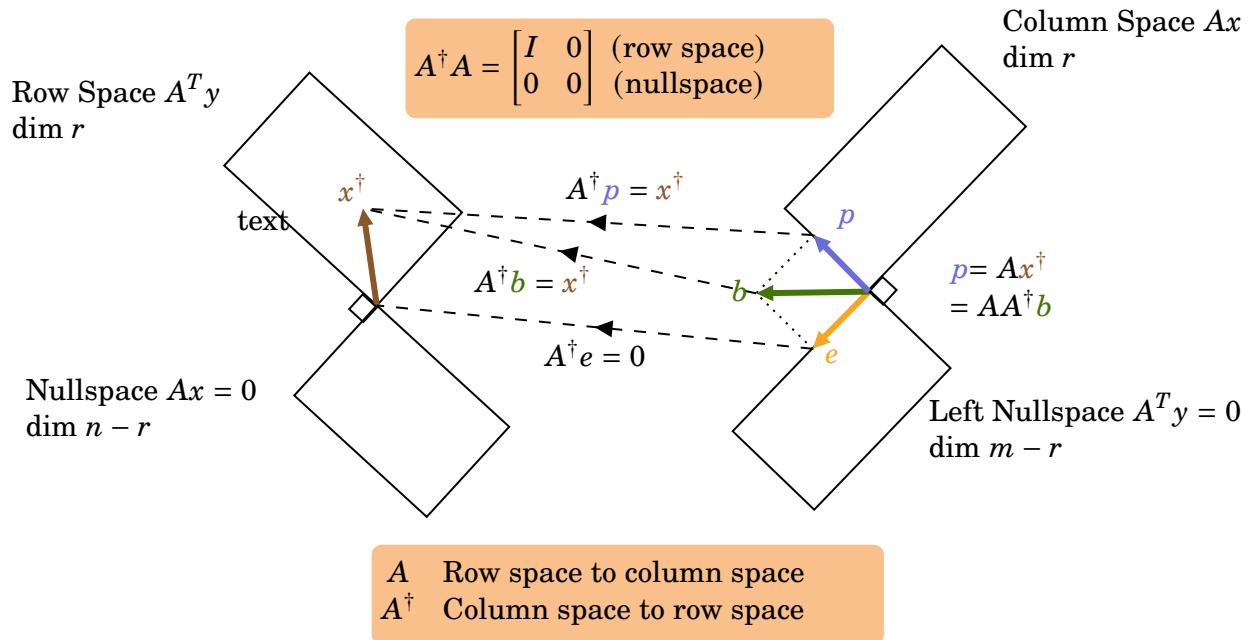


Figure 11.2: Ax^\dagger in the column space goes back to $A^\dagger Ax^\dagger = x^\dagger$ in the row space



The pseudoinverse A^\dagger is the n by m matrix that makes AA^\dagger and $A^\dagger A$ into projections.



Trying for $AA^{-1} = A^{-1}A = I$, AA^\dagger = projection matrix onto the column space of A (refer to projection onto the column space of A)
 A^+A = projection matrix onto the row space of A

11.4 QR Algorithm Using Gram-Schmidt Algorithm

Gram-Schmidt QR 算法将逐列计算 Q 和 R .

k 步后我们得到了 QR 的部分分解:

$$A = [a_1 \ a_2 \ \cdots \ a_k] = [q_1 \ q_2 \ \cdots \ q_k] \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1k} \\ 0 & R_{22} & \cdots & R_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_{kk} \end{bmatrix}$$

Corollary 11.4.1 QR 的部分分解列向量 q_1, \dots, q_k 是标准正交的。

Corollary 11.4.2 对角线元素 $R_{11}, R_{22}, \dots, R_{kk}$ 是正的。

Corollary 11.4.3 列向量 q_1, \dots, q_k 和 a_1, \dots, a_k 张成的空间相同。

Theorem 11.4.4 $A = QR$ 矩阵中的 R 矩阵为

$$R_{1k} = q_1^T a_k, R_{2k} = q_2^T a_k, \dots, R_{k-1,k} = q_{k-1}^T a_k$$



Gram-Schmidt 正交化的两条基本公式

$$\tilde{q}_i = a_i - (q_1^T a_i) q_1 - \cdots - (q_{i-1}^T a_i) q_{i-1}$$

$$\begin{aligned} a_i &= (q_1^T a_i) q_1 + \cdots + (q_{i-1}^T a_i) q_{i-1} + \underbrace{\|\tilde{q}_i\|_2}_{\tilde{q}_i} q_i \\ &= R_{1i} q_1 + \cdots + R_{ii} q_i \end{aligned}$$

Proof. 假设已经实现 $k-1$ 列的 QR 分解, 方程 $A = QR$ 的第 k 列可以计算为:

$$a_k = R_{1k} q_1 + R_{2k} q_2 + \cdots + R_{k-1,k} q_{k-1} + R_{kk} q_k$$

无论如何选择 $R_{1k}, \dots, R_{k-1,k}$, 向量

$$\tilde{q}_k = a_k - R_{1k} q_1 - R_{2k} q_2 - \cdots - R_{k-1,k} q_{k-1} \neq 0$$

都将是非零的. (由于 a_1, \dots, a_k 是线性无关的, 即 q_1, \dots, q_{i-1}, q_i 是线性无关的, 假设 Gram-Schmidt 算法过程中没有出现中途退出的状况)

因此

$$a_k \notin \text{span}\{q_1, \dots, q_{k-1}\} = \text{span}\{a_1, \dots, a_{k-1}\}$$

q_k 是 \tilde{q}_k 的单位化: 选择 $R_{kk} = \|\tilde{q}_k\|_2$, 以及 $q_k = \left(\frac{1}{R_{kk}}\right)\tilde{q}_k$ 。
 \tilde{q}_k 和 q_k 正交于 q_1, \dots, q_{k-1} , 则 $R_{1k}, \dots, R_{k-1,k}$ 为:

$$R_{1k} = q_1^T a_k, R_{2k} = q_2^T a_k, \dots, R_{k-1,k} = q_{k-1}^T a_k$$

■

11.4.1 Gram-Schmidt Algorithm

Algorithm 10: QR Decomposition Using Gram-Schmidt Algorithm

Input: 矩阵 $A \in \mathbb{R}^{m \times n}$, 列向量 a_1, \dots, a_n 线性无关

Output: 分解得到的 Q 、 R 矩阵

```

1  $R_{11} = \|a_1\|_2$ 
2  $q_1 = \frac{1}{R_{11}}a_1$ 
3 for  $k = 2$  to  $n$  do
4   for  $l = 1$  to  $k - 1$  do
5      $R_{l,k} = q_l^T a_k$ 
6   end
7    $\tilde{q}_k = a_k - (R_{1k}q_1 + R_{2k}q_2 + \dots + R_{k-1,k}q_{k-1})$ 
8    $R_{kk} = \|\tilde{q}_k\|_2$ 
9    $q_k = \frac{1}{R_{kk}}\tilde{q}_k$ 
10 end
```

■ Example 11.3

$$\begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix} = \begin{bmatrix} -1 & -1 & 1 \\ 1 & 3 & 3 \\ -1 & -1 & 5 \\ 1 & 3 & 7 \end{bmatrix} = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ 0 & R_{22} & R_{23} \\ 0 & 0 & R_{33} \end{bmatrix}$$

Q 和 R 的第一列:

$$\tilde{q}_1 = a_1 = \begin{bmatrix} -1 \\ 1 \\ -1 \\ 1 \end{bmatrix}, R_{11} = \|\tilde{q}_1\| = 2, q_1 = \frac{1}{R_{11}}\tilde{q}_1 = \begin{bmatrix} -1/2 \\ 1/2 \\ -1/2 \\ 1/2 \end{bmatrix}$$

Q 和 R 的第二列: 计算得到 $R_{12} = q_1^T a_2 = 4$ 。

正交化计算:

$$\tilde{q}_2 = a_2 - R_{12}q_1 = \begin{bmatrix} -1 \\ 3 \\ -1 \\ 3 \end{bmatrix} - 4 \begin{bmatrix} -1/2 \\ 1/2 \\ -1/2 \\ 1/2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

将其单位化得到:

$$R_{22} = \|\tilde{q}_2\| = 2, \quad q_2 = \frac{1}{R_{22}}\tilde{q}_2 = \begin{bmatrix} 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{bmatrix}$$

Q 和 R 的第三列: 计算得到 $R_{13} = q_1^T a_3 = 2$ 以及 $R_{23} = q_2^T a_3 = 8$ 。

$$\text{正交化计算: } \tilde{q}_3 = a_3 - R_{13}q_1 - R_{23}q_2 = \begin{bmatrix} 1 \\ 3 \\ 5 \\ 7 \end{bmatrix} - 2 \begin{bmatrix} -1/2 \\ 1/2 \\ -1/2 \\ 1/2 \end{bmatrix} - 8 \begin{bmatrix} 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{bmatrix} = \begin{bmatrix} -2 \\ -2 \\ 2 \\ 2 \end{bmatrix}$$

将其单位化得到:

$$R_{33} = \|\tilde{q}_3\| = 4, \quad q_3 = \frac{1}{R_{33}}\tilde{q}_3 = \begin{bmatrix} -1/2 \\ -1/2 \\ 1/2 \\ 1/2 \end{bmatrix}$$

最终结果:

$$\begin{aligned} \begin{bmatrix} -1 & -1 & 1 \\ 1 & 3 & 3 \\ -1 & -1 & 5 \\ 1 & 3 & 7 \end{bmatrix} &= [q_1 \quad q_2 \quad q_3] \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ 0 & R_{22} & R_{23} \\ 0 & 0 & R_{33} \end{bmatrix} \\ &= \begin{bmatrix} -1/2 & 1/2 & -1/2 \\ 1/2 & 1/2 & -1/2 \\ -1/2 & 1/2 & 1/2 \\ 1/2 & 1/2 & 1/2 \end{bmatrix} \begin{bmatrix} 2 & 4 & 2 \\ 0 & 2 & 8 \\ 0 & 0 & 4 \end{bmatrix} \end{aligned}$$

■

11.4.2 基于 Gram-Schmidt 方法进行 QR 分解的时间复杂度

Gram-Schmidt 方法第 k 次循环的复杂度:

- a_k 有 $k-1$ 个 $q_i^T a_k$ 内积操作: $(k-1)(2m-1)$ flops
- 计算 $\tilde{q}_k : 2(k-1)m$ flops.

$$\tilde{q}_i = a_i - (q_1^T a_i) q_1 - \cdots - (q_{i-1}^T a_i) q_{i-1}$$

- 计算 R_{kk} 和 $q_k : 3m$ flops。 $R_{kk} = \|\tilde{q}_k\|_2, q_k = \tilde{q}_k / R_{kk}$

第 k 次循环的总和: $(4m-1)(k-1) + 3m$ flops

$A \in \mathbb{R}^{m \times n}$ 分解的复杂度:

$$\begin{aligned} \sum_{k=1}^n ((4m-1)(k-1) + 3m) &= (4m-1) \frac{n(n-1)}{2} + 3mn \\ &\approx 2mn^2 \text{ flops} \end{aligned}$$

对于稀疏矩阵 (空间存储小于 $m \times n$) 的 QR 分解, 时间复杂度可以低于 $O(2mn^2)$.

11.5 The Numerical Instability of QR Decomposition based on Gram-Schmidt Algorithm

Gram-Schmidt 算法复杂度为 $2mn^2$ flops. 在实际情况中不推荐使用 (容易被舍入误差影响)。

修正 Gram-Schmidt 算法复杂度为 $2mn^2$ flops, 有更好的数值计算性能。修正之处在于求解投影时每次只减去一个投影。

Householder 算法复杂度为 $2mn^2 - (2/3)n^3$ flops。将 Q 表示为初等正交矩阵的乘积。是最广泛使用的算法(在 MATLAB 和 JULIA 中的 QR 函数使用该算法)。

本书中认为 QR 分解的复杂度为 $Q(2mn^2)$.

■ Example 11.4

```

1 [m,n]=size(A);
2 Q = zeros(m,n);
3 R = zeros(n,n);
4 for k = 1:n
5     R(1:k-1,k)=Q(:,1:k-1)'*A(:,k);
6     v= A(:,k)-Q(:,1:k-1)*R(1:k-1,k);
7     R(k,k) = norm(v);
8     Q(:,k)=v/R(k,k);
9 end
10

```

Listing 11.1: Gram Schmidt

Algorithm 11: Gram-Schmidt 的 MATLAB 算法

Input: 矩阵 A

Output: QR 分解得到的矩阵 Q 、 R

```

1 for k = 1 to n do
2      $R_{jk} = q_j^T a_k, j < k, j = 1, \dots, k - 1$ 
3      $v = \tilde{q}_k = a_k - Q_{:,1:k-1} R_{1:k-1,k}$ 
4      $R_{kk} = \|\tilde{q}_k\|_2$ 
5      $q_k = \left(\frac{1}{R_{kk}}\right) \tilde{q}_k$ 
6 end

```

构造一个矩阵 $A = USV$, 其中 U 和 V 是正交矩阵, S 是对角矩阵

$$S_{ii} = 10^{-10(i-1)/(n-1)}, \quad i = 1, \dots, n$$

把 Gram-Schmidt 算法应用到一个大小为 $m = n = 50$ 的方形矩阵 A 上。

图中显示了 q_k 与前面列之间的正交性的偏差:

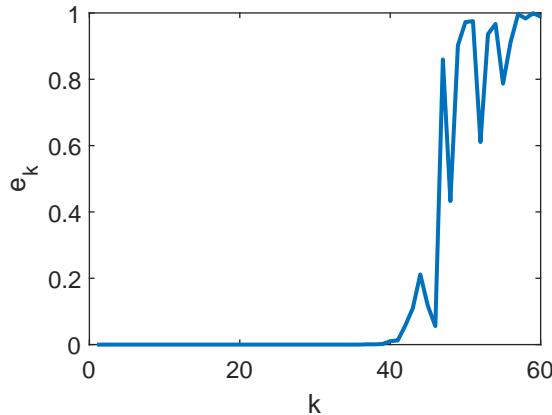
$$e_k = \max_{1 \leq i < k} |q_i^T q_k|, \quad k = 2, \dots, n$$

失去正交性是由于浮点数存储的舍入误差。

```

1 for j = 1:n
2     v=A(:,j);
3     for i=1:j-1
4         R(i,j)=Q(:,i)'*v;
5         v=v-R(i,j)*Q(:,i);
6     end
7     R(j,j)= norm(v);

```

Figure 11.3: $\max_{1 \leq i < k} |q_i^T q_k|$ (Gram-Schmidt Algorithm)

```

8   Q(:,j)=v/R(j,j);
9 end
10

```

Listing 11.2: modified Gram-Schmidt

Algorithm 12: Modified Gram-Schmidt Algorithm

Input: 矩阵 A
Output: QR 分解得到的矩阵 Q 、 R

```

1 for j = 1 to n do
2   v =  $\tilde{q}_k = a_k$ 
3   for i = 1 to j - 1 do
4      $R_{ij} = q_i^T v$ 
5      $v = v - R_{ij} q_i$ 
6   end
7    $R_{kk} = \|\tilde{q}_k\|_2$ 
8    $q_k = \left(\frac{1}{R_{kk}}\right) \tilde{q}_k$ 
9 end

```

修正 Gram-Schmidt 算法的误差更小。 ■

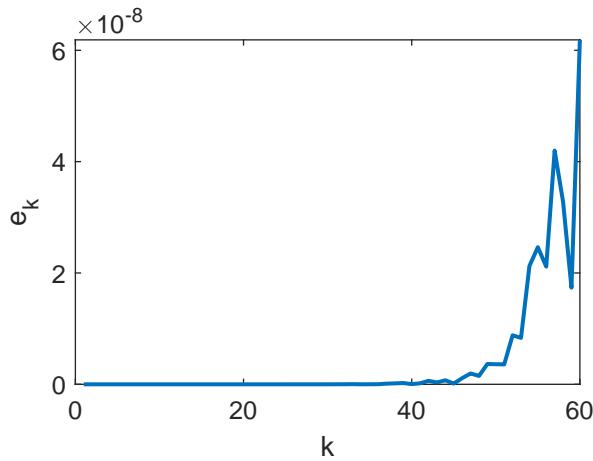
11.6 QR Decomposition Using Householder Transformation

Householder 算法是 QR 分解常用的算法 (MATLAB 和 Julia 中的 qr 函数)。与 Gram-Schmidt 相比，对舍入误差更有鲁棒性。

Householder 算法计算一个“完整的” QR 分解：

$$A_{m \times n} = [Q_{m \times n} \quad \tilde{Q}_{m \times (m-n)}] \begin{bmatrix} R_{n \times n} \\ 0_{(m-n) \times n} \end{bmatrix}, \quad [Q \quad \tilde{Q}] \text{ 是列正交的矩阵}$$

Figure 11.4: 在同一组数据中使用修正 Gram-Schmidt 算法求得的误差, $e_k = \max_{1 \leq i < k} |q_i^T q_k|, k = 2, \dots, n$



Proof.

$$\begin{aligned} A &= [Q \quad \tilde{Q}] \begin{bmatrix} R \\ 0 \end{bmatrix} \\ &= QR + \tilde{Q}0 \\ &= QR \end{aligned}$$

■

where R is an $n \times n$ upper triangular matrix, 0 is an $(m-n) \times n$ zero matrix, Q is $m \times n$, \tilde{Q} is $m \times (m-n)$, and Q and \tilde{Q} both have orthogonal columns.



$A \in \mathbb{R}^{m \times n}, m \geq n$, 当 $m < n$ 时列线性相关, 无法进行 QR 分解。

完整的 Q 因子被构造成正交矩阵的乘积:

$$[Q \quad \tilde{Q}] = H_1 H_2 \cdots H_n$$

每个 $H_i \in \mathbb{R}^{m \times m}$ 是对称正交的 Householder 矩阵。

11.6.1 Householder Matrix

Theorem 11.6.1 $H = I - 2vv^T$, 其中 $\|v\|_2 = 1$, Hx 是 x 关于超平面 $\{u \mid v^T u = 0\}$ 反对称.

H 是对称的

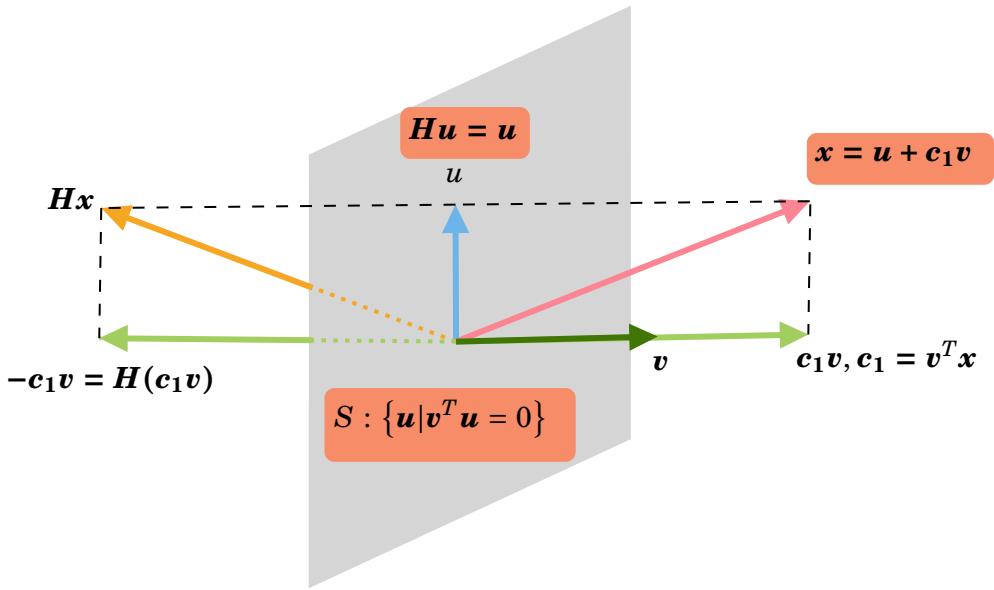
$$H^T = H$$

H 是正交的

$$H^T H = I$$

Proof.

$$\begin{aligned} Hv &= (I - 2vv^T)(c_1v) \\ &= c_1v - 2vv^T c_1v \\ &= c_1v - 2c_1vv^T v \\ &= -c_1v \end{aligned}$$

Figure 11.5: Reflection of x 

$$\begin{aligned} Hu &= (I - 2vv^T)u \\ &= u - 2vv^T u \\ &= u \quad (v^T u = 0) \end{aligned}$$

$$\begin{aligned} Hx &= H(u + c_1 v) \\ &= u - c_1 v \\ &= x - 2(v^T x)v \end{aligned}$$

■

Theorem 11.6.2 矩阵向量积 Hx 能化简为

$$Hx = x - 2(v^T x)v$$

Hx 的算法复杂度

如果 v 和 x 的长度是 p , 复杂度是 $4p$ flops。

11.6.2 构造反射算子

给定非零 p 维向量 $y = (y_1, y_2, \dots, y_p)$, 定义

Definition 11.6.1

$$w = \begin{bmatrix} y_1 + \text{sign}(y_1) \|y\|_2 \\ y_2 \\ \vdots \\ y_p \end{bmatrix}$$

$$v = \frac{1}{\|w\|_2} w$$

sign(x) 是符号函数, $\text{sign}(0) = 0$ 。

Theorem 11.6.3 向量 w 满足

$$\|w\|_2^2 = 2y^T w$$

$$\begin{aligned}\|w\|_2^2 &= w^T w \\ &= 2 \left(\|y\|_2^2 + |y_1| \|y\|_2 \right) \\ &= 2y^T (y + \text{sign}(y_1) \|y\|_2 e_1) \\ &= 2y^T w\end{aligned}$$

Definition 11.6.2 — Constructed Householder Matrix. 将 y 变换为单位基向量 $e = [1, 0, 0, \dots, 0]^T$ 乘以一个常数的 Householder 矩阵为

$$H = I - 2 \frac{ww^T}{\|w\|_2^2}$$

Proof.

$$H = I - 2vv^T = I - 2 \frac{ww^T}{\|w\|_2^2} \quad (v = \frac{1}{\|w\|_2} w)$$

■

Theorem 11.6.4 Householder 矩阵 $H = I - 2vv^T = I - 2 \frac{ww^T}{\|w\|_2^2}$ 将 y 映射为

$$Hy = \begin{bmatrix} -\text{sign}(y_1) \|y\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = r_1$$

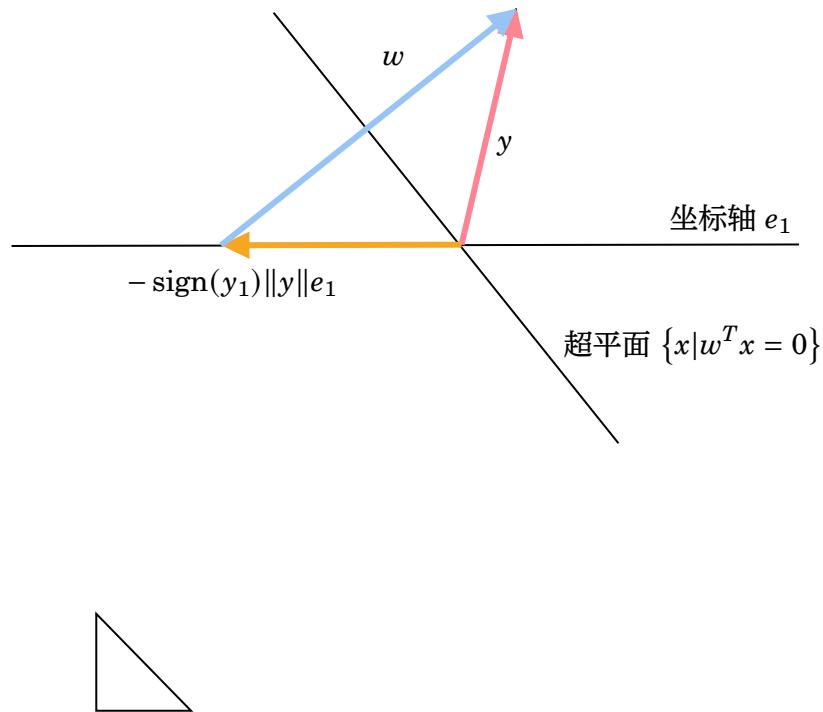
Proof.

$$\begin{aligned}Hy &= y - \frac{2(w^T y)}{\|w\|_2^2} w \\ &= y - w \\ &= -\text{sign}(y_1) \|y\|_2 e_1 \\ &= \begin{bmatrix} -\text{sign}(y_1) \|y\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}\end{aligned}$$

■

即 Householder 变换的结果 Hy 只保留第一个元素的值, 其余元素的值变为 0. 又因为 H 是正交、对称的, 它与 QR 分解求解 R 有一定联系。

Figure 11.6: 构造的 Householder 矩阵的几何意义



构造的 Householder 矩阵几何意义

关于超平面 $\{x \mid w^T x = 0\}$, 其法向量 w, v :

$$w = y + \text{sign}(y_1) \|y\|_2 e_1, v = \frac{w}{\|w\|_2}$$

反射算子 H 将 y 映射到向量 $-\text{sign}(y_1) \|y\|_2 e_1$ 。

11.6.3 Householder 三角化

计算反射算子 H_1, \dots, H_n 将 A 简化为上三角矩阵形式:

$$H_n H_{n-1} \cdots H_1 A = \begin{bmatrix} R \\ 0 \end{bmatrix}$$

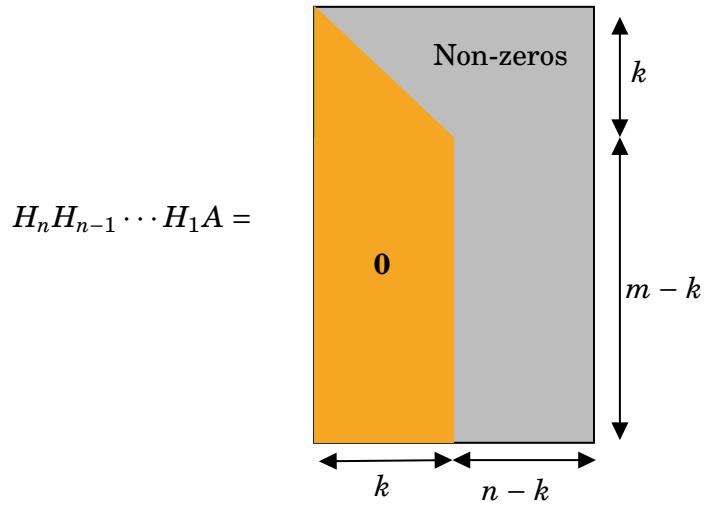
第 k 个步骤之后, 矩阵 $H_k H_{k-1} \cdots H_1 A$ 具有以下结构:

对于 $i > j$ 和 $j \leq k$, 第 i, j 个位置的元素为零。

其过程如下:

$$A = \begin{bmatrix} X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X \\ X & X & X & X & X & X & X & X \end{bmatrix}$$

在第一次处理之后, A_1 第一列只剩下第一个元素不为 0.

Figure 11.7: The structure of $H_k H_{k-1} \dots H_1 A$ 

$$H_1 A = A_1 = \begin{bmatrix} X & X & X & X & X & X & X & X \\ 0 & X & X & X & X & X & X & X \\ 0 & X & X & X & X & X & X & X \\ 0 & X & X & X & X & X & X & X \\ 0 & X & X & X & X & X & X & X \\ 0 & X & X & X & X & X & X & X \\ 0 & X & X & X & X & X & X & X \\ 0 & X & X & X & X & X & X & X \end{bmatrix}$$

第二次处理对于 $H_1 A_{2:m,1:n}$ ($A_{12:m,1:n}$) 进行处理。

$$H_2 A_1 = A_2 = \begin{bmatrix} X & X & X & X & X & X & X & X \\ 0 & X & X & X & X & X & X & X \\ 0 & 0 & X & X & X & X & X & X \\ 0 & 0 & X & X & X & X & X & X \\ 0 & 0 & X & X & X & X & X & X \\ 0 & 0 & X & X & X & X & X & X \\ 0 & 0 & X & X & X & X & X & X \\ 0 & 0 & X & X & X & X & X & X \end{bmatrix}$$

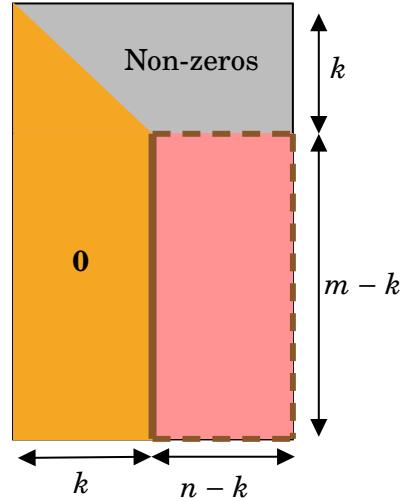
11.6.4 Household-QR Algorithm

Theorem 11.6.5 在算法步骤 2 中，将 $A_{k:m,k:n}$ 与反射算子 $I - 2v_k v_k^T$ 相乘

$$(I - 2v_k v_k^T) A_{k:m,k:n} = A_{k:m,k:n} - 2v_k (v_k^T A_{k:m,k:n})$$

等价于用 $H_k \in \mathbb{R}^{m \times m}$ 乘以 $A \in \mathbb{R}^{m \times n}$

$$H_k = \begin{bmatrix} I & 0 \\ 0 & I - 2v_k v_k^T \end{bmatrix} = I - 2 \begin{bmatrix} 0 \\ v_k \end{bmatrix} \begin{bmatrix} 0 \\ v_k \end{bmatrix}^T$$

Figure 11.8: 迭代计算过程中 A 的结构**Algorithm 13:** QR Decomposition Using Householder Transformation

Input: Matrix A

Output: $v_1, \dots, v_n, v_k \in \mathbb{R}^{m-k+1}$ (Q), $A = \begin{bmatrix} R \\ 0 \end{bmatrix}$ (R)

1 **for** k in $1 : n$ **do**

2 令 $y = A_{k:m,k} \in \mathbb{R}^{m-k+1}$, 计算向量 v_k

$$w = y + \text{sign}(y_1) \|y\| e_1$$

$$v_k = \frac{1}{\|w\|} w$$

3 将 $A_{k:m,k:n} \in \mathbb{R}^{(m-k+1) \times (n-k+1)}$ 与反射矩阵 $I - 2v_k v_k^T$ 相乘

$$A_{k:m,k:n} := A_{k:m,k:n} - 2v_k (v_k^T A_{k:m,k:n})$$

4 **end**

算法的最终结果将下列矩阵来代替 $A \in \mathbb{R}^{m \times n}$

$$\begin{bmatrix} R \\ 0 \end{bmatrix}$$

返回向量 v_1, \dots, v_n , 其中 v_k 的长度为 $m - k + 1$ 。

11.6.5 An Example for Householder Algorithm

Problem 11.3

$$A = \begin{bmatrix} -1 & -1 & 1 \\ 1 & 3 & 3 \\ -1 & -1 & 5 \\ 1 & 3 & 7 \end{bmatrix} = H_1 H_2 H_3 \begin{bmatrix} R \\ 0 \end{bmatrix}$$

计算反射算子 H_1, H_2, H_3 来将矩阵 A 三角化

$$H_3 H_2 H_1 A = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ 0 & R_{22} & R_{23} \\ 0 & 0 & R_{33} \\ 0 & 0 & 0 \end{bmatrix}$$

R 的第一列: 计算将 A 的第一列映射到 e_1 乘积的反射算子

$$y = \begin{bmatrix} -1 \\ 1 \\ -1 \\ 1 \end{bmatrix}, \quad w = y - \|y\|_2 e_1 = \begin{bmatrix} -3 \\ 1 \\ -1 \\ 1 \end{bmatrix}, \quad v_1 = \frac{1}{\|w\|_2} w = \frac{1}{2\sqrt{3}} \begin{bmatrix} -3 \\ 1 \\ -1 \\ 1 \end{bmatrix}$$

用 $I - 2v_1 v_1^T$ 和 A 的乘积代替 A :

$$A := (I - 2v_1 v_1^T) A = \begin{bmatrix} 2 & 4 & 2 \\ 0 & 4/3 & 8/3 \\ 0 & 2/3 & 16/3 \\ 0 & 4/3 & 20/3 \end{bmatrix}$$

R 的第二列: 计算将 $A_{2:4,2}$ 映射到 e_1 乘积的反射算子

$$y = \begin{bmatrix} 4/3 \\ 2/3 \\ 4/3 \end{bmatrix}, \quad w = y + \|y\|_2 e_1 = \begin{bmatrix} 10/3 \\ 2/3 \\ 4/3 \end{bmatrix}, \quad v_2 = \frac{1}{\|w\|_2} w = \frac{1}{\sqrt{30}} \begin{bmatrix} 5 \\ 1 \\ 2 \end{bmatrix}$$

用 $I - 2v_2 v_2^T$ 和 $A_{2:4,2:3}$ 的乘积代替 $A_{2:4,2:3}$:

$$A := \begin{bmatrix} 1 & 0 \\ 0 & I - 2v_2 v_2^T \end{bmatrix} A = \begin{bmatrix} 2 & 4 & 2 \\ 0 & -2 & -8 \\ 0 & 0 & 16/5 \\ 0 & 0 & 12/5 \end{bmatrix}$$

R 的第三列: 计算将 $A_{3:4,3}$ 映射到 e_1 乘积的反射算子

$$y = \begin{bmatrix} 16/5 \\ 12/5 \end{bmatrix}, \quad w = y + \|y\|_2 e_1 = \begin{bmatrix} 36/5 \\ 12/5 \end{bmatrix}, \quad v_3 = \frac{1}{\|w\|_2} w = \frac{1}{\sqrt{10}} \begin{bmatrix} 3 \\ 1 \end{bmatrix}$$

用 $I - 2v_3v_3^T$ 和 $A_{3:4,3}$ 的乘积代替 $A_{3:4,3}$:

$$A := \begin{bmatrix} I & 0 \\ 0 & I - 2v_3v_3^T \end{bmatrix} A = \begin{bmatrix} 2 & 4 & 2 \\ 0 & -2 & -8 \\ 0 & 0 & -4 \\ 0 & 0 & 0 \end{bmatrix}$$

总的求解式为

$$\begin{aligned} H_3H_2H_1A &= \begin{bmatrix} I_2 & 0 \\ 0 & I_2 - 2v_3v_3^T \end{bmatrix} \begin{bmatrix} I_1 & 0 \\ 0 & I_3 - 2v_2v_2^T \end{bmatrix} (I_4 - 2v_1v_1^T) A \\ &= \begin{bmatrix} I_2 & 0 \\ 0 & I_2 - 2v_3v_3^T \end{bmatrix} \begin{bmatrix} I_1 & 0 \\ 0 & I_3 - 2v_2v_2^T \end{bmatrix} \begin{bmatrix} 2 & 4 & 2 \\ 0 & 4/3 & 8/3 \\ 0 & 2/3 & 16/3 \\ 0 & 4/3 & 20/3 \end{bmatrix} \\ &= \begin{bmatrix} I_2 & 0 \\ 0 & I_2 - 2v_3v_3^T \end{bmatrix} \begin{bmatrix} 2 & 4 & 2 \\ 0 & -2 & -8 \\ 0 & 0 & 16/5 \\ 0 & 0 & 12/5 \end{bmatrix} \\ &= \begin{bmatrix} 2 & 4 & 2 \\ 0 & -2 & -8 \\ 0 & 0 & -4 \\ 0 & 0 & 0 \end{bmatrix} \end{aligned}$$

■

11.6.6 Complexity of Householder Algorithm

$$H_k = \begin{bmatrix} I & 0 \\ 0 & I - 2v_kv_k^T \end{bmatrix} = I - 2 \begin{bmatrix} 0 \\ v_k \end{bmatrix} \begin{bmatrix} 0 \\ v_k \end{bmatrix}^T$$

Householder 方法第 k 次循环的复杂度:

- $v_k^T A_{k:m,k:n}$ 的乘积: $(2(m-k+1)-1)(n-k+1)$ flops
- v_k 的外积: $(m-k+1)(n-k+1)$ flops
- $A_{k:m,k:n}$ 的减法: $(m-k+1)(n-k+1)$ flops

第 k 次循环的总和: $4(m-k+1)(n-k+1)$ flops

计算 R 和 v_1, \dots, v_n 的总复杂度

$$\begin{aligned} \sum_{k=1}^n 4(m-k+1)(n-k+2) &\approx \int_0^n 4(m-t)(n-t+1) dt \\ &\approx 2mn^2 - \frac{2}{3}n^3 \text{ flops} \end{aligned}$$



$\sum_{k=1}^n 4(m-k+1)(n-k+2)$ 是因为

11.7 Householder 变换进行 QR 分解的 Q 因子

Householder 算法返回向量 v_1, \dots, v_n , 其定义为:

Definition 11.7.1 — v_1, \dots, v_n 的完整表示.

$$\begin{bmatrix} Q & \tilde{Q} \end{bmatrix} = H_1 H_2 \cdots H_n$$

通常不需计算矩阵 $\begin{bmatrix} Q & \tilde{Q} \end{bmatrix}$ 。向量 v_1, \dots, v_n 是 $\begin{bmatrix} Q & \tilde{Q} \end{bmatrix}$ 简单表示 (economical representation)。

Theorem 11.7.1 $\begin{bmatrix} Q & \tilde{Q} \end{bmatrix}$ 或其转置的乘积可以计算为:

$$\begin{aligned} \begin{bmatrix} Q & \tilde{Q} \end{bmatrix} x &= H_1 H_2 \cdots H_n x \\ \begin{bmatrix} Q & \tilde{Q} \end{bmatrix}^T y &= H_n H_{n-1} \cdots H_1 y \end{aligned}$$

11.7.1 Multiplication with Q factor

Definition 11.7.2 — 矩阵-向量积 $H_k x$. 矩阵-向量积 $H_k x$ 定义为:

$$H_k x = \begin{bmatrix} I_{k-1} & 0 \\ 0 & I - 2v_k v_k^T \end{bmatrix} \begin{bmatrix} x_{1:k-1} \\ x_{k:m} \end{bmatrix} = \begin{bmatrix} x_{1:k-1} \\ x_{k:m} - 2(v_k^T x_{k:m}) v_k \end{bmatrix}$$

11.7.2 矩阵-向量积 $H_k x$ 算法复杂度

$H_k x$ 乘积的复杂度为: $4(m - k + 1)$ flops。

$H_1 H_2, \dots, H_n$ 或其转置的乘积的复杂度为:

$$\sum_{k=1}^n 4(m - k + 1) \approx 4mn - 2n^2 \text{ flops}$$

其复杂度约等于 $m \times n$ 矩阵的矩阵-向量乘积 ($2mn$ flops)。

11.8 Fast Orthogonalization (Givens and Householder)



This section is quoted from [Strang1993IntroductionTL].

There are three ways to reach the important factorization $A = QR$.

Gram-Schmidt works to find the orthonormal vectors in Q . Then R is upper triangular because of the order of Gram-Schmidt steps.

Now we look at better methods (Householder and Givens), which use a product of specially simple Q 's that we know are orthogonal.

Elimination gives $A = LU$, orthogonalization gives $A = QR$. We don't want a triangular L , we want an orthogonal Q . L is a product of E 's from elimination, with 1's on the diagonal and the multiplier ℓ_{ij} below. **Q will be a product of orthogonal matrices.**

There are two simple orthogonal matrices to take the place of the E 's. The *reflection matrices* $I - 2uu^T$ are named after *Householder*. The *plane rotation matrices* are named after *Givens*. The simple matrix that rotates the xy plane by θ is Q_{21} :

Definition 11.8.1 — Givens Rotation in the 1-2 plane.

$$Q_{21} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Use Q_{21} the way you used E_{21} , to produce a zero in the (2, 1) position. That determines the angle θ . Bill Hager gives this example in Applied Numerical Linear Algebra:

■ Example 11.5 — 使用 Givens 进行 QR 分解.

$$Q_{21}A = \begin{bmatrix} .6 & .8 & 0 \\ -.8 & .6 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 90 & -153 & 114 \\ 120 & -79 & -223 \\ 200 & -40 & 395 \end{bmatrix} = \begin{bmatrix} 150 & -155 & -110 \\ \mathbf{0} & 75 & -225 \\ 200 & -40 & 395 \end{bmatrix}$$

The zero came from $-8(90) + .6(120)$. No need to find θ , what we needed was $\cos \theta$

$$\cos \theta = \frac{90}{\sqrt{90^2 + 120^2}} \quad \text{and} \quad \sin \theta = \frac{-120}{\sqrt{90^2 + 120^2}}$$

Now we attack the (3, 1) entry. The rotation will be in rows and columns 3 and 1. The numbers $\cos \theta$ and $\sin \theta$ are determined from 150 and 200, instead of 90 and 120

$$Q_{31}Q_{21}A = \begin{bmatrix} .6 & 0 & .8 \\ 0 & 1 & 0 \\ -.8 & 0 & .6 \end{bmatrix} \begin{bmatrix} 150 & \cdot & \cdot \\ 0 & \cdot & \cdot \\ 200 & \cdot & \cdot \end{bmatrix} = \begin{bmatrix} 250 & -125 & 250 \\ \mathbf{0} & 75 & -225 \\ \mathbf{0} & 100 & 325 \end{bmatrix}$$

One more step to R . The (3, 2) entry has to go. The numbers $\cos \theta$ and $\sin \theta$ now come from 75 and 100. The rotation is now in rows and columns 2 and 3:

$$Q_{32}Q_{31}Q_{21}A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & .6 & .8 \\ 0 & -.8 & .6 \end{bmatrix} \begin{bmatrix} 250 & -125 & \cdot \\ 0 & 75 & \cdot \\ 0 & 100 & \cdot \end{bmatrix} = \begin{bmatrix} 250 & -125 & 250 \\ \mathbf{0} & 125 & 125 \\ \mathbf{0} & \mathbf{0} & 375 \end{bmatrix}$$

We have reached the upper triangular R . What is Q ? Move the plane rotations Q_{ij} to the other side to find $A = QR$ -just as you moved the elimination matrices E_{ij} to the other side to find $A = LU$:

Theorem 11.8.1

$$Q_{32}Q_{31}Q_{21}A = R \quad \text{means} \quad A = (Q_{21}^{-1}Q_{31}^{-1}Q_{32}^{-1})R = QR$$

The inverse of each Q_{ij} is Q_{ij}^T (rotation through $-\theta$). The inverse of E_{ij} was not an orthogonal matrix! **LU and QR are similar but L and Q are not the same.**

Householder reflections are faster than rotations because each one clears out a whole column below the diagonal. Watch how the first column a_1 of A becomes column r_1 of R :

■ Example 11.6 — Reflection by H_1 .

$$H_1 = I - 2u_1u_1^T$$

$$H_1a_1 = \begin{bmatrix} \|a_1\| \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} -\|a_1\| \\ 0 \\ \vdots \\ 0 \end{bmatrix} = r_1$$

The length was not changed, and u_1 is in the direction of $a_1 - r_1$. We have $n - 1$ entries in the unit vector \mathbf{u}_1 to get $n - 1$ zeros in \mathbf{r}_1 . (Rotations had one angle θ to get one zero.) When we reach column k , we have $n - k$ available choices in the unit vector u_k . This leads to $n - k$ zeros in \mathbf{r}_k . We just store the \mathbf{u} 's and \mathbf{r} 's to know the final Q and R :

Theorem 11.8.2 — H 的逆是它本身.

$$(H_{n-1} \dots H_1) A = R \quad \text{means} \quad A = (H_1 \dots H_{n-1}) R = QR$$

11.9 Recap: QR Decomposition

11.9.1 分治策略

求解线性方程组 $Ax = b$, 矩阵 $A \in \mathbb{R}^{n \times n}$ 分解成“结构简单”的矩阵相乘:

$$A = A_1 A_2 \cdots A_k$$

■ **Example 11.7** — 求解 k 个线性方程组 $A_1 A_2 \cdots A_k x = b$.

$$\begin{aligned} A_1 \underbrace{(A_2 \cdots A_k x)}_{z_1} &= b \\ A_2 \underbrace{(A_3 \cdots A_k x)}_{z_2} &= z_1 \\ &\vdots \\ A_{k-1} \underbrace{(A_k x)}_{z_{k-1}} &= z_{k-2} \\ A_k x &= z_{k-1} \end{aligned}$$

■ **Example 11.8** — QR 分解 $Ax = b$.

$$\begin{aligned} Qy &= b \\ Rx &= y \end{aligned}$$

通常分解复杂度远大于求解复杂度。

11.9.2 非奇异矩阵的 QR 分解

Theorem 11.9.1 任意非奇异矩阵 $A \in \mathbb{R}^{n \times n}$, 都可以进行 QR 分解。

Corollary 11.9.2 $Q \in \mathbb{R}^{n \times n}$ 是一个正交矩阵

Corollary 11.9.3 $R \in \mathbb{R}^{n \times n}$ 是一个上三角矩阵并且对角元素都为正数

11.9.3 使用 QR 分解求 A^{-1} 可以转换成 $R^{-1}Q^T$

Theorem 11.9.4 $A^{-1} = (QR)^{-1} = R^{-1}Q^{-1} = R^{-1}Q^T$

11.9.4 QR 分解求解 A^{-1}

计算非奇异矩阵 $A \in \mathbb{R}^{n \times n}$ 的逆 A^{-1} , $X = [x_1, x_2, \dots, x_n]$, $x_i \in \mathbb{R}^n, i = 1, \dots, n$, $I = [e_1, e_2, \dots, e_n]$, $e_i \in \mathbb{R}^n, i = 1, \dots, n$:

Theorem 11.9.5 — QR 分解求解 A^{-1} .

$$Rx_1 = Q^T e_1, Rx_2 = Q^T e_2, \dots, Rx_n = Q^T e_n$$

$$\begin{aligned} AX &= I \\ \Rightarrow QRX &= I \\ \Rightarrow RX &= Q^T I \\ \Rightarrow Rx_1 &= Q^T e_1, Rx_2 = Q^T e_2, \dots, Rx_n = Q^T e_n \end{aligned}$$

The Complexity of $QRX = I$

复杂度: $2n^3 + n^3 \approx 3n^3$ flops

- QR 分解复杂度: $2n^3$
- 回代法: 一次回代 n^2 , 则 n 次回代 n^3

11.9.5 QR 分解求解线性方程组

使用 QR 分解求解线性方程组 $Ax = b$, 矩阵 $A \in \mathbb{R}^{n \times n}$ 为非奇异矩阵

Algorithm 14: QR 分解求解线性方程组

- 1 首先对 A 进行 QR 分解, 得到 $A = QR$
- 2 计算 $y = Q^T b$
- 3 通过回代法求解 $Rx = y$

The Complexity of Solving Linear Equation Systems Using QR Decomposition

复杂度: $2n^3 + 3n^2 \approx 2n^3$ flops

- QR 分解复杂度: $2n^3$
- 矩阵向量乘法: $2n^2$
- 回代法: n^2

12. LU 分解

12.1 Solving Linear Equation Systems

12.1.1 Linear Equation Systems

- Example 12.1

$$Ax = b \Leftrightarrow \begin{array}{rcl} x+ & 2 & y+ & 3 & z = & 6 \\ 2 & x+ & 5 & y+ & 2 & z = & 4 \\ 6 & x- & 3 & y+ & & z = & 2 \end{array}$$

见 Row Picture, Column Picture 的概念。

12.1.2 Elimination

- Example 12.2 — Elimination. Before

$$\begin{array}{rcl} x- & 2 & y = & 1 \\ 3 & x+ & 2 & y = & 11 \end{array}$$

After

$$\begin{array}{rcl} x- & 2 & y = & 1 \\ 8 & y = & 8 \end{array}$$

■ Definition 12.1.1 — Pivot. The first nonzero in the row that does elimination. **Zero is not allowed as a pivot.**

■ Definition 12.1.2 — Multiplier. (Entry to eliminate) divided by (pivot)

消元法通过 elimination matrices E 进行消元操作，使得主对角线以下的元素为 0。Multiply the j^{th} equation by ℓ_{ij} and subtract from the i^{th} equation. (This eliminates x_j from equation i .) We need a lot of these simple matrices E_{ij} , one for every nonzero to be eliminated below the main diagonal.

Definition 12.1.3 — Elementary matrix, Elimination matrix. The elementary matrix or elimination matrix E_{ij} has the extra nonzero entry $-\ell$ in the i, j position. Then E_{ij} subtracts a multiple ℓ of row j from row i .

Theorem 12.1.1 消元法的本质是

$$Ax = b \Rightarrow EAx = Eb$$

Definition 12.1.4 — Permutation matrices P . P_{ij} is the identity matrix with rows i and j reversed. When this *permutation matrix* P_{ij} multiplies a matrix, it exchanges rows i and j .

Definition 12.1.5 — Augmented matrix.

$$\begin{bmatrix} A & b \end{bmatrix}$$

Computing A^{-1} by Gauss-Jordan Elimination

Multiply $[A \ I]$ by A^{-1} to get $[I \ A^{-1}]$

■ **Example 12.3**

$$\begin{aligned} \left[\begin{array}{cccc|ccccc} K & e_1 & e_2 & e_3 & 1 & 0 & 0 & 0 \\ \hline 2 & -1 & 0 & 1 & 0 & 1 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 2 & 0 & 0 & 0 & 1 & 0 \end{array} \right] &= \left[\begin{array}{cccc|ccccc} 2 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & \frac{3}{2} & -1 & \frac{1}{2} & 1 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 & 0 & 1 & 0 & 0 \end{array} \right] \quad \text{Start Gauss-Jordan on } K \\ &\rightarrow \left[\begin{array}{cccc|ccccc} 2 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & \frac{3}{2} & -1 & \frac{1}{2} & 1 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 & 0 & 1 & 0 & 0 \end{array} \right] \quad \left(\frac{1}{2} \text{ row 1 + row 2} \right) \\ &\rightarrow \left[\begin{array}{cccc|ccccc} 2 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & \frac{3}{2} & -1 & \frac{1}{2} & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{4}{3} & \frac{1}{3} & \frac{2}{3} & 1 & 0 & 0 \end{array} \right] \quad \left(\frac{2}{3} \text{ row 2 + row 3} \right) \\ \left(\begin{array}{c} \text{Zero above} \\ \text{third pivot} \end{array} \right) &\rightarrow \left[\begin{array}{cccc|ccccc} 2 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & \frac{3}{2} & 0 & \frac{3}{2} & \frac{3}{2} & \frac{3}{4} & 0 & 0 \\ 0 & 0 & \frac{4}{3} & \frac{1}{3} & \frac{2}{3} & 1 & 0 & 0 \end{array} \right] \quad \left(\frac{3}{4} \text{ row 3 + row 2} \right) \\ \left(\begin{array}{c} \text{Zero above} \\ \text{second pivot} \end{array} \right) &\rightarrow \left[\begin{array}{cccc|ccccc} 2 & 0 & 0 & \frac{3}{2} & 1 & \frac{1}{2} & 0 & 0 \\ \hline 0 & \frac{3}{2} & 0 & \frac{3}{2} & \frac{3}{2} & \frac{3}{4} & 0 & 0 \\ 0 & 0 & \frac{4}{3} & \frac{1}{3} & \frac{2}{3} & 1 & 0 & 0 \end{array} \right] \quad \left(\frac{2}{3} \text{ row 2 + row 1} \right) \end{aligned}$$

它变成 Reduced echelon form R .

$$\begin{aligned} &\left(\text{divided by 2} \right) \left[\begin{array}{cccc|ccccc} 1 & 0 & 0 & \frac{3}{4} & \frac{1}{2} & \frac{1}{4} & 0 & 0 \\ \hline 0 & 1 & 0 & \frac{1}{2} & 1 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 1 & \frac{1}{4} & \frac{1}{2} & \frac{3}{4} & 0 & 0 \end{array} \right] = \left[\begin{array}{cccc|ccccc} I & x_1 & x_2 & x_3 & 1 & 0 & 0 & 0 \\ \hline I & K^{-1} & & & & & & \end{array} \right] \end{aligned}$$

1. K is symmetric across its main diagonal. Then K^{-1} is also symmetric.
2. K is tridiagonal (only three nonzero diagonals). But K^{-1} is a dense matrix with no zeros. That is another reason we don't often compute inverse matrices. The inverse of a band matrix is generally a dense matrix.
3. The product of pivots is $2 \left(\frac{3}{2}\right) \left(\frac{4}{3}\right) = 4$. This number 4 is the determinant of K .

K^{-1} involves division by the determinant of K $K^{-1} = \frac{1}{4} \begin{bmatrix} 3 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}$.

This is why an invertible matrix cannot have a zero determinant: we need to divide.

■

12.2 LU 分解

$$(E_{32}E_{31}E_{21})A = U \quad \text{becomes} \quad A = (E_{21}^{-1}E_{31}^{-1}E_{32}^{-1})U \quad \text{which is} \quad A = LU$$

Theorem 12.2.1 When a row of A starts with zeros, so does that row of L .

When a column of A starts with zeros, so does that column of U .

■ **Example 12.4 — The key reason why A equals LU .** Ask yourself about the pivot rows that are subtracted from lower rows. Are they the original rows of A ? No, elimination probably changed them.

Are they rows of U ? Yes, the pivot rows never change again.

When computing the third row of U , we subtract multiples of earlier rows of U (not rows of A !):

$$\text{Row 3 of } U = (\text{Row 3 of } A) - \ell_{31}(\text{Row 1 of } U) - \ell_{32}(\text{Row 2 of } U)$$

Rewrite this equation to see that the row $\begin{bmatrix} \ell_{31} & \ell_{32} & 1 \end{bmatrix}$ is multiplying the matrix U :

$$(\text{Row 3 of } A) = \ell_{31}(\text{Row 1 of } U) + \ell_{32}(\text{Row 2 of } U) + 1(\text{Row 3 of } U)$$

This is exactly row 3 of $A = LU$.

That row of L holds $\ell_{31}, \ell_{32}, 1$. All rows look like this, whatever the size of A . With no row exchanges, we have $A = LU$.

Definition 12.2.1 — A 的 LU 分解.

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1k} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} & \cdots & a_{kn} \\ \vdots & & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nk} & \cdots & a_{nn} \end{pmatrix} = LU$$

$$\text{where } L = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ l_{21} & 1 & \ddots & 0 \\ \vdots & \vdots & \ddots & 0 \\ l_{n1} & l_{n2} & \cdots & 1 \end{pmatrix}, U = \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & 0 & \ddots & \vdots \\ 0 & 0 & 0 & u_{nn} \end{pmatrix}$$

所以

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1r} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ a_{r1} & \cdots & a_{rr} & \cdots & a_{rn} \\ \vdots & & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nr} & \cdots & a_{nn} \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & 0 & \ddots & 0 \\ l_{r1} & \cdots & 1 & \ddots & \vdots \\ \vdots & & \vdots & \ddots & 0 \\ l_{n1} & \cdots & l_{nr} & \cdots & 1 \end{pmatrix} \cdot \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & 0 & \ddots & \vdots \\ 0 & 0 & 0 & u_{nn} \end{pmatrix}$$

12.2.1 $A = LDU$

$A = LU$ 是不对称的。但是可以改写为对称形式。

Split U into

$$\begin{bmatrix} d_1 & & & & \\ & d_2 & & & \\ & & \ddots & & \\ & & & & d_n \end{bmatrix} \begin{bmatrix} 1 & u_{12}/d_1 & u_{13}/d_1 & \cdots \\ & 1 & u_{23}/d_2 & \cdots \\ & & \ddots & \vdots \\ & & & 1 \end{bmatrix}.$$

■ Example 12.5 ($A = LU$) $\begin{bmatrix} 1 & 0 \\ 3 & 1 \end{bmatrix} \begin{bmatrix} 2 & 8 \\ 0 & 5 \end{bmatrix}$ splits further into
 $(A = LDU) \begin{bmatrix} 1 & 0 \\ 3 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 5 \end{bmatrix} \begin{bmatrix} 1 & 4 \\ 0 & 1 \end{bmatrix}$

Theorem 12.2.2 当 A 是对称矩阵的时候, 且消元的时候不需要行交换:

$$S = LDL^T$$

■ Example 12.6

$$\begin{bmatrix} 1 & 2 \\ 2 & 7 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}$$

12.2.2 L 、 U 矩阵的性质

根据矩阵的乘法原理, 有

Theorem 12.2.3 A 的第一行元素 a_{1j} 为

$$a_{1j} = u_{1j}, j = 1, \dots, n$$

Corollary 12.2.4 U 的第一行元素 u_{1j} 为

$$u_{1j} = a_{1j}, j = 1, \dots, n$$

Proof.

$$\begin{aligned}
 A &= \begin{pmatrix} \mathbf{a}_{11} & \cdots & \mathbf{a}_{1r} & \cdots & \mathbf{a}_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ a_{r1} & \cdots & a_{rr} & \cdots & a_{rn} \\ \vdots & & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nr} & \cdots & a_{nn} \end{pmatrix} \\
 &= \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & 0 & \ddots & 0 \\ l_{r1} & \cdots & 1 & \cdots & \vdots \\ \vdots & \ddots & \vdots & \ddots & 0 \\ l_{n1} & \cdots & l_{nr} & \cdots & 1 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{u}_{11} & \cdots & \mathbf{u}_{1r} & \cdots & \mathbf{u}_{1n} \\ 0 & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & u_{rr} & \cdots & u_{rn} \\ \vdots & \ddots & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & u_{nn} \end{pmatrix}
 \end{aligned}$$

■

Corollary 12.2.5 L 的第一列元素 l_{i1} 为

$$l_{i1} = \frac{a_{i1}}{u_{11}}, i = 2, 3, \dots, n$$

Proof.

$$\begin{aligned}
 A &= \begin{pmatrix} \mathbf{a}_{11} & \cdots & a_{1r} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ \mathbf{a}_{r1} & \cdots & a_{rr} & \cdots & a_{rn} \\ \vdots & & \vdots & \ddots & \vdots \\ \mathbf{a}_{n1} & \cdots & a_{nr} & \cdots & a_{nn} \end{pmatrix} \\
 &= \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & 0 & \ddots & 0 \\ \mathbf{l}_{r1} & \cdots & 1 & \ddots & \vdots \\ \vdots & \ddots & \vdots & \ddots & 0 \\ \mathbf{l}_{n1} & \cdots & l_{nr} & \cdots & 1 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{u}_{11} & \cdots & u_{1r} & \cdots & u_{1n} \\ 0 & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & u_{rr} & \cdots & u_{rn} \\ \vdots & \ddots & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & u_{nn} \end{pmatrix}
 \end{aligned}$$

■

Theorem 12.2.6 A 的第 r 行主对角线以右元素元素 $a_{1j}(j = 1, \dots, n)$ 为

$$a_{rj} = \sum_{k=1}^r l_{rk} u_{kj}, r = 1, 2, \dots, n, j = r, \dots, n$$

Proof.

$$\begin{aligned}
 A &= \begin{pmatrix} a_{11} & \cdots & a_{1r} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ a_{r1} & \cdots & \textcolor{red}{a_{rr}} & \cdots & \textcolor{red}{a_{rn}} \\ \vdots & & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nr} & \cdots & a_{nn} \end{pmatrix} \\
 &= \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & 0 & \ddots & 0 \\ l_{r1} & \cdots & \textcolor{red}{1} & \cdots & 0 \\ \vdots & \cdots & \vdots & \ddots & 0 \\ l_{n1} & \cdots & l_{nr} & \cdots & 1 \end{pmatrix} \cdot \begin{pmatrix} u_{11} & \cdots & u_{1r} & \cdots & u_{1n} \\ 0 & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \textcolor{red}{u_{rr}} & \cdots & \textcolor{red}{u_{rn}} \\ \vdots & \ddots & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & u_{nn} \end{pmatrix}
 \end{aligned}$$

■

Corollary 12.2.7 U 第 r 行主对角线以右元素 u_{rj}

$$u_{rj} = a_{rj} - \sum_{k=1}^{r-1} l_{rk} u_{kj}, j = r, \dots, n$$

Proof.

$$\begin{aligned}
 a_{rj} &= \sum_{k=1}^r l_{rk} u_{kj}, r = 1, 2, \dots, n, j = r, \dots, n \\
 \Rightarrow a_{rj} &= \sum_{k=1}^{r-1} l_{rk} u_{kj} + l_{rr} u_{rj}, r = 1, 2, \dots, n, j = r, \dots, n \\
 \Rightarrow u_{rj} &= a_{rj} - \sum_{k=1}^{r-1} l_{rk} u_{kj}, j = r, \dots, n
 \end{aligned}$$

■

Corollary 12.2.8 U 的对角线元素 u_{rr}

$$u_{rr} = a_{rr} - \sum_{k=1}^{r-1} l_{rk} u_{kr}$$

Theorem 12.2.9 A 的第 r 列元素主对角线以下元素 a_{ir} ($i = r+1, \dots, n$) 为

$$a_{ir} = \sum_{k=1}^r l_{ik} u_{kr}, i = r+1, \dots, n, r = 1, 2, \dots, n-1$$

Proof.

$$\begin{aligned}
 A &= \begin{pmatrix} a_{11} & \cdots & a_{1r} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ a_{r1} & \cdots & \textcolor{red}{a_{rr}} & \cdots & a_{rn} \\ \vdots & & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & \textcolor{red}{a_{nr}} & \cdots & a_{nn} \end{pmatrix} \\
 &= \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & 0 & \ddots & 0 \\ l_{r1} & \cdots & \textcolor{red}{1} & \cdots & 0 \\ \vdots & \cdots & \vdots & \ddots & 0 \\ l_{n1} & \cdots & \textcolor{red}{l_{nr}} & \cdots & 1 \end{pmatrix} \cdot \begin{pmatrix} u_{11} & \cdots & u_{1r} & \cdots & u_{1n} \\ 0 & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \textcolor{red}{u_{rr}} & \cdots & u_{rn} \\ \vdots & \ddots & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & u_{nn} \end{pmatrix}
 \end{aligned}$$

■

Corollary 12.2.10 显然, $r = 1$ 时

$$a_{i1} = l_{i1}u_{11}, i = 2, 3, \dots, n$$

Corollary 12.2.11 L 第 r 列主对角线以下元素 l_{ir}

$$l_{ir} = \frac{a_{ir} - \sum_{k=1}^{r-1} l_{ik}u_{kr}}{u_{rr}}, i = r+1, \dots, n$$

12.2.3 Complexity of LU Decomposition

求解 $Ax = b$, A 为非奇异矩阵, LU 算法为求解方程组 $Ax = b$ 的标准解法, 复杂度: $\frac{2}{3}n^3 + 2n^2 \approx \frac{2}{3}n^3$ flops

1. 对矩阵 A 进行 LU 分解 ($\frac{2}{3}n^3$ flops)
2. 回代法: 求解 $Ly = b$ (n^2 flops)
3. 回代法: 求解 $Ux = y$ (n^2 flops)

12.2.4 Example of LU Decomposition

■ **Example 12.7** 对矩阵 A 进行 LU 分解

$$A = \begin{bmatrix} 8 & 2 & 9 \\ 4 & 9 & 4 \\ 6 & 7 & 9 \end{bmatrix}$$

$$A = \begin{bmatrix} 8 & 2 & 9 \\ 4 & 9 & 4 \\ 6 & 7 & 9 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

计算 U 的第一行和 L 的第一列

$$(u_{11}, u_{12}, u_{13}) = (8, 2, 9), (l_{21}, l_{31}) = \left(\frac{1}{2}, \frac{3}{4}\right)$$

然后计算 U 的第二行和 L 的第二列

$$u_{22} = a_{22} - l_{21}u_{12} = 8, u_{23} = a_{23} - l_{21}u_{13} = -\frac{1}{2}, l_{32} = \frac{a_{32} - l_{31}u_{12}}{u_{22}} = \frac{11}{16}$$

最后计算 U 的第三行

$$u_{33} = a_{33} - l_{31}u_{13} - l_{32}u_{23} = -\frac{83}{32}$$

■

12.3 Problem of LU Decomposition

■ Example 12.8

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \\ 0 & 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & L_{32} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

计算 U 的第一行和 L 的第一列

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 2 \\ 0 & 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & l_{32} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

然后计算 U 的第二行和 L 的第二列

$$u_{22} = a_{22} - l_{21}u_{12} = 0 \quad l_{32} = \frac{a_{32} - l_{31}u_{12}}{u_{22}} = \frac{1}{0}$$

即该矩阵无法 LU 分解!

■

12.4 $PA = LU$

Theorem 12.4.1 非奇异矩阵 $A \in \mathbb{R}^{n \times n}$, 则可分解为 $A = P^T LU$

P 是一个置换矩阵, L 为下三角矩阵并且对角线元素全为 1, U 为上三角矩阵

$PA = LU$ 分解方法不唯一, 随着 P 的选择不同, L 、 U 也不同。

■ Example 12.9 — $PA = LU$. $A = \begin{bmatrix} 0 & 5 & 5 \\ 2 & 9 & 0 \\ 6 & 8 & 8 \end{bmatrix}$, $P_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$, $P_2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

易知

$$P_1^T = P_1^{-1} = P_1, P_2^T = P_2^{-1} = P_2$$

计算可得

$$P_1 A = \begin{bmatrix} 6 & 8 & 8 \\ 2 & 9 & 0 \\ 0 & 5 & 5 \end{bmatrix}, P_2 A = \begin{bmatrix} 2 & 9 & 0 \\ 0 & 5 & 5 \\ 6 & 8 & 8 \end{bmatrix}$$

LU 分解不唯一:

$$P_1 A = \begin{bmatrix} 6 & 8 & 8 \\ 2 & 9 & 0 \\ 0 & 5 & 5 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{3} & 1 & 0 \\ 0 & \frac{15}{19} & 1 \end{bmatrix} \begin{bmatrix} 6 & 8 & 8 \\ 0 & \frac{19}{3} & \frac{-8}{3} \\ 0 & 0 & \frac{135}{19} \end{bmatrix} = L_1 U_1 \Rightarrow A = P_1 L_1 U_1$$

$$P_2 A = \begin{bmatrix} 2 & 9 & 0 \\ 0 & 5 & 5 \\ 6 & 8 & 8 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 3 & \frac{-19}{5} & 1 \end{bmatrix} \begin{bmatrix} 2 & 9 & 2 \\ 0 & 5 & 5 \\ 0 & 0 & 27 \end{bmatrix} = L_2 U_2 \Rightarrow A = P_2 L_2 U_2$$

Theorem 12.4.2 这个方法等价于对 A 进行初等变换然后对 PA 进行分解 $PA = LU$

The Complexity of $PA = LU$

复杂度: $\frac{2}{3}n^3$ flops

12.5 舍入误差的影响

■ Example 12.10

$$\begin{bmatrix} 10^{-5} & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

解得:

$$x_1 = -\frac{1}{1 - 10^{-5}}, x_2 = \frac{1}{1 - 10^{-5}}$$

使用 LU 分解求解上述方程，并且使用以下两个置换矩阵:

$$P_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{or} \quad P_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

计算过程中，中间结果四舍五入到小数点后四位。

选择 1: $P_1 = I$ 。

$$\begin{bmatrix} 10^{-5} & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 10^5 & 1 \end{bmatrix} \begin{bmatrix} 10^{-5} & 1 \\ 0 & 1 - 10^{-5} \end{bmatrix}$$

L 和 U 四舍五入到小数点后四位

$$L = \begin{bmatrix} 1 & 0 \\ 10^5 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 10^{-5} & 1 \\ 0 & -10^5 \end{bmatrix}$$

向前回代

$$\begin{bmatrix} 1 & 0 \\ 10^5 & 1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \Rightarrow z_1 = 1, z_2 = -10^5$$

向后回代

$$\begin{bmatrix} 10^{-5} & 1 \\ 0 & -10^5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ -10^{-5} \end{bmatrix} \Rightarrow x_1 = 0, x_2 = 1$$



x_1 的误差为 100%。

选择 2: 行进行交换。

$$\begin{bmatrix} 1 & 1 \\ 10^{-5} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 10^{-5} & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 - 10^{-5} \end{bmatrix}$$

L 和 U 四舍五入到小数点后四位

$$L = \begin{bmatrix} 1 & 0 \\ 10^{-5} & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

向前回代

$$\begin{bmatrix} 1 & 0 \\ 10^{-5} & 1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \Rightarrow z_1 = 0, z_2 = 1$$

向后回代

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \Rightarrow x_1 = -1, x_2 = 1$$



x_1, x_2 的误差约为 10^{-5} 。

不同置换矩阵 P ，算法可能导致产生不同的误差的结果；由于数值存储存在误差：第一种 P_1 行交换，算法不稳定；第二种 P_2 行交换，算法是稳定得到“准确”近似解；

在数值分析中，一些比较简单的规则去挑选置换矩阵 P ，使得算法结果比较稳定。

$$\begin{bmatrix} 10^{-5} & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 \\ 10^{-5} & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

12.6 稀疏线性方程组

Theorem 12.6.1 如果矩阵 A 是稀疏矩阵，则它一般可以被分解为

$$A = P_1 L U P_2$$

矩阵 P_1, P_2 都为置换矩阵

Corollary 12.6.2 对矩阵 A 进行行变换和列变换得到: $\tilde{A} = P_1^T A P_2^T$
然后进行分解: $\tilde{A} = LU$

P_1 和 P_2 的选择会影响 L 和 U 的稀疏度。

Least Squares

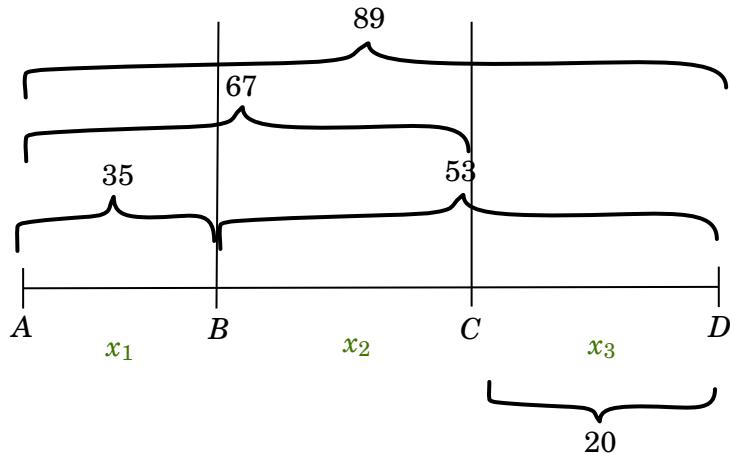
13	Least Squares	108
13.1	An Example: Measurement Problem	
13.2	求解最小二乘法	
13.3	The Geometry of Least Squares: 投影与 A 列空间的关系	
13.4	正规方程	
13.5	QR 分解求解最小二乘法	
13.6	求解正规方程可能带来的严重误差	
13.7	梯度下降法	
13.8	估计学习率 (步长) α	
14	Multi-objective Least Squares	119
14.1	Definition of Multi-objective Least Squares	
14.2	求解多目标最小二乘问题	
14.3	正则化数据拟合	
14.4	图像逆问题	
14.5	信号去噪	
15	Constrained Least Squares	124
15.1	Karush—Kuhn—Tucker Conditions	
15.2	优化问题：最小范数优化问题	
15.3	优化问题：点到线的最短距离	
15.4	最小范数优化问题推导	
15.5	QR 分解求解最小范数优化问题	
15.6	最小二乘法约束问题	
15.7	优化问题：分段多项式拟合问题	
15.8	先验假设	
15.9	优化问题：最小二乘法的带约束 KKT 条件	
15.10	KKT 最优条件	
15.11	LU 分解求解 KKT 最优条件	
15.12	QR 分解求解 KKT 最优条件	
15.13	KKT 最优条件求解复杂度：QR vs LU	
15.14	Supplement Material: Karush-Kuhn-Tucker (KKT) 条件	
15.15	Supplement Material: 浅谈最优化问题的 KKT 条件	
15.16	Supplement Material: 凸优化、拉格朗日乘子法和 KKT 条件	
15.17	Supplementary Material: KKT Conditions, Linear Programming and Nonlinear Programming	
15.18	Supplementary Material: KKT Conditions	
15.19	Karush-Kuhn-Tucker (KKT) Conditions	
16	非线性最小二乘法	162
16.1	非线性最小二乘法的定义	
16.2	梯度	
16.3	求解非线性最小二乘法：目标梯度	
16.4	Gauss-Newton Algorithm	
16.5	Levenberg—Marquardt Algorithm	
16.6	Newton Method	
16.7	Newton method for nonlinear least squares	
16.8	Unconstrained minimization problem	
16.9	Local and global optimum	

13. Least Squares

13.1 An Example: Measurement Problem

Problem 13.1 已知测量量路段长度: $AD = 89, AC = 67, BD = 53, AB = 35, CD = 20$, x_1, x_2 和 x_3 的长度是多少? ■

Figure 13.1: Measurement Problem



由 x_1, x_2 和 x_3 的关系可得方程组:

$$\left\{ \begin{array}{l} x_1 + x_2 + x_3 = 89 \\ x_1 + x_2 = 67 \\ x_2 + x_3 = 53 \Leftrightarrow Ax = b, A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, b = \begin{bmatrix} 89 \\ 67 \\ 53 \\ 35 \\ 20 \end{bmatrix} \\ x_1 = 35 \\ x_3 = 20 \end{array} \right.$$

取后三个式子求解方程组, 回代前两个式子

$$\begin{cases} x_2 + x_3 = 53 \\ x_1 = 35 \Rightarrow x_1 = 35, x_2 = 33, x_3 = 20. \\ x_3 = 20 \\ x_1 + x_2 + x_3 = 88 \neq 89 \\ x_1 + x_2 = 68 \neq 67 \end{cases}$$

由于测量存在误差，方程组之间相互矛盾，该超定方程组无解。

Problem 13.2 — 最小二乘问题. 寻找该方程组的近似解，并尽可能逼近方程组的目标 b ，即残差向量 $r = Ax - b$ 某种度量下尽可能小

$$\min_x \|Ax - b\|_2^2 = \|r\|_2^2 \quad (\ell_2 \text{范数度量残差})$$

使用 ℓ_2 、 ℓ_∞ 等也可以度量误差，但是函数在零点处不光滑，不能求导。

Problem 13.3 — 求解最小二乘解. 给定 $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, 求解 $x \in \mathbb{R}^n$ 让目标函数最小

$$\min_x \|Ax - b\|_2^2 = \min_x \sum_{i=1}^m \left(\sum_{j=1}^n A_{ij}x_j - b_i \right)^2$$

Notation 13.1 (最小二乘法的解). 最小二乘法的解为 \hat{x}

$$\hat{x} = \arg \min_x \|Ax - b\|_2^2 = \arg \min_x \sum_{i=1}^m \left(\sum_{j=1}^n A_{ij}x_j - b_i \right)^2$$

■ **Example 13.1** $f(x) = \|Ax - b\|_2^2$, $A = \begin{bmatrix} 2 & 0 \\ -1 & 1 \\ 0 & 2 \end{bmatrix}$, $b = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$

求解

$$\hat{x} = \arg \min_x \|Ax - b\|_2^2$$

解：

$$f(x) = \|Ax - b\|_2^2 = (2x_1 - 1)^2 + (-x_1 + x_2)^2 + (2x_2 + 1)^2$$

$$\frac{\partial f}{\partial x_1} = 10x_1 - 2x_2 - 4, \frac{\partial f}{\partial x_2} = -2x_1 + 10x_2 + 4$$

$$\nabla f(x) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix} = 0 \Rightarrow \hat{x} = \left(\frac{1}{3}, -\frac{1}{3} \right)^T$$

Theorem 13.1.1 设最小二乘法的解为 \hat{x} ，满足：

$$\|A\hat{x} - b\|_2^2 \leq \|Ax - b\|_2^2, \forall x \in \mathbf{R}^n$$

当残差 $\hat{r} = A\hat{x} - b = 0$ 时，则 \hat{x} 是线性方程组 $Ax = b$ 的解；否则其为误差最小平方

和意义下方程组的近似解。

13.2 求解最小二乘法

给定 $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $x \in \mathbb{R}^n$ 目标函数:

$$f(x) = \|Ax - b\|_2^2 = \sum_{i=1}^m \left(\sum_{j=1}^n A_{ij}x_j - b_i \right)^2$$

为使目标函数最小, 求最优解 \hat{x} : $\hat{x} = \arg \min_x f(x)$

Theorem 13.2.1 可微函数 $f(x)$ 的最优解 \hat{x} 满足条件: 梯度 $\nabla f(\hat{x}) = \mathbf{0}$, 即:

$$\nabla f(\hat{x}) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(\hat{x}) \\ \vdots \\ \frac{\partial f}{\partial x_n}(\hat{x}) \end{bmatrix} = 2A^T(A\hat{x} - b) = 0$$

Theorem 13.2.2 — 正规方程与最小二乘解.

$$A^T A x = A^T b$$

A 的列向量线性无关时, 则 $\hat{x} = (A^T A)^{-1} A^T b$ 。

Proof. 设函数 $g_i(x) = \sum_{j=1}^n A_{ij}x_j - b_i$, 则有

$$g_i(x) = \sum_{j=1}^n A_{ij}x_j - b_i \Rightarrow \begin{pmatrix} A_{1,1} & \cdots & A_{1,k} & \cdots & A_{1,n} \\ \vdots & & \vdots & & \vdots \\ \textcolor{red}{A_{j,1}} & \cdots & \textcolor{red}{A_{j,k}} & \cdots & \textcolor{red}{A_{j,n}} \\ \vdots & & \vdots & & \vdots \\ A_{m,1} & \cdots & A_{m,k} & \cdots & A_{m,n} \end{pmatrix} \begin{pmatrix} \textcolor{red}{x_1} \\ \vdots \\ \textcolor{red}{x_j} \\ \vdots \\ \textcolor{red}{x_n} \end{pmatrix} - \begin{pmatrix} b_1 \\ \vdots \\ \textcolor{red}{b_j} \\ \vdots \\ b_n \end{pmatrix}$$

$$f(x) = \|Ax - b\|_2^2 = \sum_{i=1}^m \left(\sum_{j=1}^n A_{ij}x_j - b_i \right)^2 = \sum_{i=1}^m (g_i(x))^2$$

函数 $f(x)$ 对变量 x_k 偏导为

$$\frac{\partial f(x)}{\partial x_k} = \sum_{i=1}^m \left((2g_i(x)) \left(\frac{\partial g_i(x)}{\partial x_k} \right) \right)$$

又因为

$$\frac{\partial g_i(x)}{\partial x_k} = A_{ik}$$

所以

$$\begin{aligned}\frac{\partial f}{\partial x_k}(x) &= \sum_{i=1}^m 2(g_i(x))(A_{ik}) \\ &= 2 \sum_{i=1}^m \left(\left(\sum_{j=1}^n A_{ij}x_j - b_i \right) (A_{ik}) \right) \\ &= 2 \sum_{i=1}^m \left(\left(\sum_{j=1}^n A_{ij}x_j \right) (A_{ik}) \right) - 2 \sum_{i=1}^m ((b_i)(A_{ik}))\end{aligned}$$

注意有

$$\sum_{j=1}^n A_{ij}x_j = \begin{pmatrix} A_{1,1} & \cdots & A_{1,k} & \cdots & A_{1,n} \\ \vdots & & \vdots & & \vdots \\ \textcolor{red}{A_{j,1}} & \cdots & \textcolor{red}{A_{j,k}} & \cdots & \textcolor{red}{A_{j,n}} \\ \vdots & & \vdots & & \vdots \\ A_{m,1} & \cdots & A_{m,k} & \cdots & A_{m,n} \end{pmatrix} \begin{pmatrix} \textcolor{red}{x_1} \\ \vdots \\ \textcolor{red}{x_j} \\ \vdots \\ \textcolor{red}{x_n} \end{pmatrix} = \begin{pmatrix} \textcolor{red}{Result}_1 \\ \vdots \\ \textcolor{red}{Result}_k \\ \vdots \\ \textcolor{red}{Result}_m \end{pmatrix} = Ax$$

$$\begin{aligned}A_{1,\textcolor{blue}{k}} \times Result_1 \\ + \\ \sum_{i=1}^m \left(\underbrace{\left(\sum_{j=1}^n A_{ij}x_j \right)}_{Result} (A_{ik}) \right) \\ + \\ \textcolor{red}{A_{i,\textcolor{blue}{k}}} \times \textcolor{red}{Result}_i \\ + \\ \vdots \\ + \\ A_{m,\textcolor{blue}{k}} \times Result_m\end{aligned} = \begin{pmatrix} A_{1,\textcolor{blue}{k}} \\ \vdots \\ A_{i,\textcolor{blue}{k}} \\ \vdots \\ A_{m,\textcolor{blue}{k}} \end{pmatrix}^T Ax = a_{\textcolor{blue}{k}}^T Ax$$

类似地，有

$$\begin{aligned}A_{1,\textcolor{blue}{k}} \times b_1 \\ + \\ \vdots \\ \sum_{i=1}^m ((b_i)(A_{ik})) = \textcolor{red}{A_{i,\textcolor{blue}{k}}} \times \textcolor{red}{b_i} \\ + \\ \vdots \\ + \\ A_{m,\textcolor{blue}{k}} \times b_m\end{aligned} = a_{\textcolor{blue}{k}}^T b$$

所以

$$\begin{aligned}\frac{\partial f}{\partial x_k}(x) &= 2a_k^T Ax - 2a_k^T b \\ &= 2a_k^T (Ax - b)\end{aligned}$$

所以函数 $f(x)$ 的梯度

$$\begin{aligned}\nabla f(x) &= \begin{bmatrix} \frac{\partial f}{\partial x_1}(x) \\ \vdots \\ \frac{\partial f}{\partial x_n}(x) \end{bmatrix} \\ &= 2 \begin{bmatrix} a_1^T(Ax - b) \\ a_2^T(Ax - b) \\ \vdots \\ a_n^T(Ax - b) \end{bmatrix} \\ &= 2 [a_1, a_2, \dots, a_n]^T (Ax - b) \\ &= 2A^T(Ax - b)\end{aligned}$$

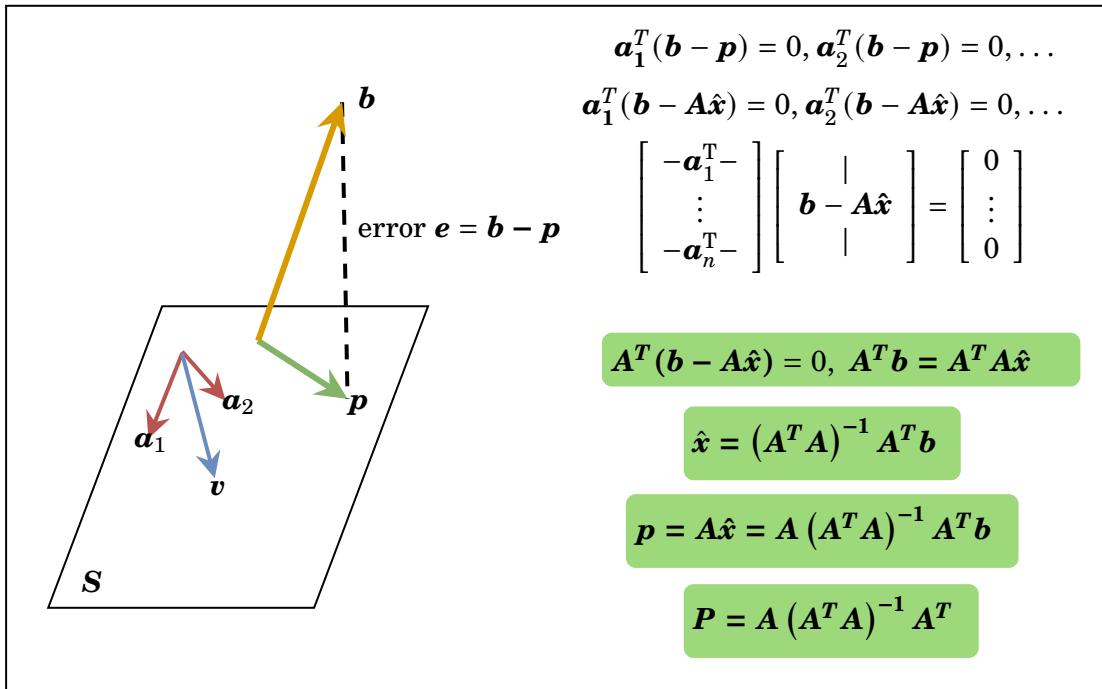
$$\nabla f(x) = 2(A^T A x - A^T b) = 0 \Rightarrow A^T A x = A^T b$$

A 的列向量无关时，则 $\hat{x} = (A^T A)^{-1} A^T b$ 。

■

13.3 The Geometry of Least Squares: 投影与 A 列空间的关系

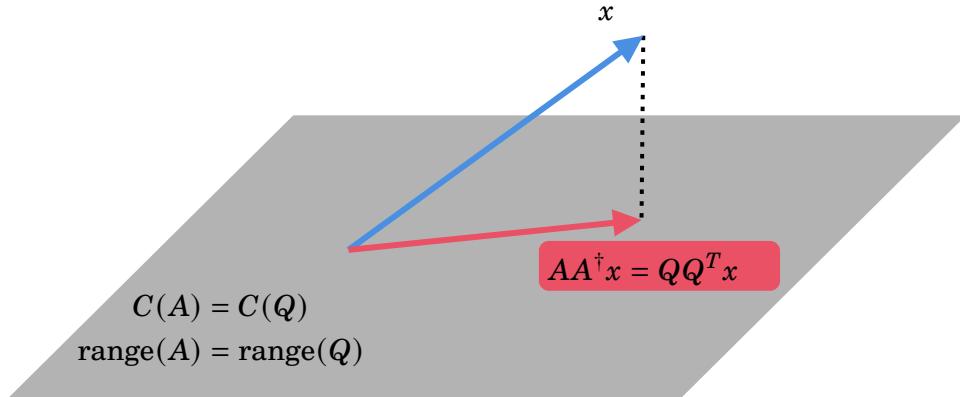
Figure 13.2: Projection of b into the column space of A , A is any matrix. Sourced from [Strang1993IntroductionTL]



矩阵 $A \in \mathbb{R}^{m \times n}$ 的列 $a_1, a_2, \dots, a_n \in \mathbb{R}^m$ 的最小二乘法问题

$$\hat{x} = \arg \min_x \|Ax - b\|_2^2, \|Ax - b\|_2^2 = \left\| \sum_{j=1}^n a_j x_j - b \right\|_2^2$$

Figure 13.3: Projecting onto the column space of A is also projecting onto the column space of Q



向量 b 在 $\text{range}(A)$ 上的投影是 $A(A^T A)^{-1} A^T b$ 。

残余向量 $\hat{r} = A\hat{x} - b$ 满足 $A^T \hat{r} = A^T(A\hat{x} - b) = 0$ 。残余向量 \hat{r} 正交于 A 的每一列, 因此正交于 $\text{range}(A)$ 。

Theorem 13.3.1 — 投影与 A 列空间的关系. $A\hat{x} \in \text{range}(A)$ 是 A 的列空间中最接近 b 的向量。

$\hat{r} = A\hat{x} - b$ 正交于 A 的列空间 (值域空间) $\text{range}(A)$ 。

13.4 正规方程

Theorem 13.4.1 — 最小二乘法问题的正规方程.

$$A^T A x = A^T b$$

等价于

$$\nabla f(x) = 0, f(x) = \|Ax - b\|_2^2$$

系数矩阵 $A^T A$ 是 A 的 Gram 矩阵, 最小二乘法问题所有的解都满足正规方程。

Theorem 13.4.2 如果 A 的列线性无关, 则
 $A^T A$ 为非奇异矩阵, 正规方程此时有唯一解。

13.5 QR 分解求解最小二乘法

Theorem 13.5.1 — QR 分解求解最小二乘法. 若 $A \in \mathbb{R}^{m \times n}$ 的列向量线性无关, 则存在 $A = QR$ 分解, $Q \in \mathbb{R}^{m \times n}$, $R \in \mathbb{R}^{n \times n}$

最小二乘法问题的解

$$\begin{aligned}\hat{x} &= \left(A^T A \right)^{-1} A^T b \\ &= \left((QR)^T (QR) \right)^{-1} (QR)^T b \\ &= \left(R^T Q^T QR \right)^{-1} R^T Q^T b \\ &= \left(R^T R \right)^{-1} R^T Q^T b \\ &= R^{-1} Q^T b\end{aligned}$$

■ Example 13.2

$$A = \begin{bmatrix} 3 & -6 \\ 4 & -8 \\ 0 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} -1 \\ 7 \\ 0 \end{bmatrix}$$

首先对 A 进行 QR 分解

$$Q = \begin{bmatrix} 3/5 & 0 \\ 4/5 & 0 \\ 0 & 1 \end{bmatrix}, \quad R = \begin{bmatrix} 5 & -10 \\ 0 & 1 \end{bmatrix}$$

计算 $d = Q^T b = (5, 2)$

求解 $Rx = d$

$$\begin{bmatrix} 5 & -10 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \end{bmatrix}$$

解得 $x_1 = 5, x_2 = 2$

13.5.1 The Complexity of Solving Least Square Problem via QR Decomposition

算法复杂度：

- 首先对 A 进行 QR 分解 $A = QR$ ($2mn^2$ flops)
- 计算矩阵向量乘积 $d = Q^T b$ ($2mn$ flops)
- 通过回代求解 $Rx = d$ (n^2 flops)
- 复杂度: $2mn^2$ flops

13.6 求解正规方程可能带来的严重误差

直接求解正规方程组求解：

$$A^T A x = A^T b$$

可能会造成严重的舍入误差。

■ Example 13.3 一个列向量“几乎”线性相关的矩阵

$$A = \begin{bmatrix} 1 & -1 \\ 0 & 10^{-5} \\ 0 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 10^{-5} \\ 1 \end{bmatrix}$$

将中间结果四舍五入到小数点后 8 位。

方法 1：通过 Gram 矩阵求解

$$A^T A = \begin{bmatrix} 1 & -1 \\ -1 & 1 + 10^{-10} \end{bmatrix} \approx \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, A^T b = \begin{bmatrix} 0 \\ 10^{-10} \end{bmatrix} \Rightarrow x = \begin{bmatrix} 10^{-10} \\ 10^{-10} \end{bmatrix}$$

经过四舍五入之后，Gram 矩阵为奇异矩阵。

方法 2：通过对 A 进行 QR 分解

$$\begin{aligned} Q &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, R = \begin{bmatrix} 1 & -1 \\ 0 & 10^{-5} \end{bmatrix} \\ \hat{x} &= (A^T A)^{-1} A^T b = R^{-1} Q^T b \\ \Rightarrow Rx &= Q^T b \\ \Rightarrow \begin{bmatrix} 1 & -1 \\ 0 & 10^{-5} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} &= \begin{bmatrix} 0 \\ 10^{-5} \end{bmatrix} \\ \Rightarrow x &= \begin{bmatrix} 1 \\ 1 \end{bmatrix} \end{aligned}$$

方法 2 比方法 1 更稳定，因为它避免构造 Gram 矩阵。 ■

13.7 梯度下降法

给定 $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $x \in \mathbb{R}^n$ 目标函数:

$$f(x) = \|Ax - b\|_2^2 = \sum_{i=1}^m \left(\sum_{j=1}^n A_{ij}x_j - b_i \right)^2$$

为使目标函数最小，可求最优解 \hat{x} : $\hat{x} = \arg \min_x f(x)$ 。

Problem 13.4 $A \in \mathbb{R}^{m \times n}$ 列向量线性相关或 n 非常大 $A^T A \in \mathbb{R}^{n \times n}$ 不可逆，无法直接代入求得最小二乘解。 ■

通过迭代求解目标的最优解过程:

$$x^{(1)}, x^{(2)}, \dots, x^{(k)} \rightarrow \hat{x}$$

设 $x^{(k)}$ 是第 k 步迭代，期望更新 $x^{(k+1)}$ ，满足 $f(x^{(k+1)}) < f(x^{(k)})$.

设函数 $f(x)$ 可微，根据泰勒公式，在 $x^{(k)}$ 的一阶公式为

$$f(x^{(k+1)}) = f(x^{(k)}) + \langle \nabla f(x^{(k)}), x^{(k+1)} - x^{(k)} \rangle + o(\|x^{(k+1)} - x^{(k)}\|)$$

如果 $\|x^{(k+1)} - x^{(k)}\|_2$ 足够小，则有

$$f(x^{(k+1)}) - f(x^{(k)}) \approx \langle \nabla f(x^{(k)}), x^{(k+1)} - x^{(k)} \rangle$$

Corollary 13.7.1 根据柯西不等式 1.5.1

$$|\langle \nabla f(x^{(k)}), x^{(k+1)} - x^{(k)} \rangle| \leq \|\nabla f(x^{(k)})\|_2 \|x^{(k+1)} - x^{(k)}\|_2$$

所以有

$$\langle \nabla f(x^{(k)}), x^{(k+1)} - x^{(k)} \rangle \geq -\|\nabla f(x^{(k)})\|_2 \|x^{(k+1)} - x^{(k)}\|_2$$

当 $x^{(k+1)} - x^{(k)} = -\alpha_k \nabla f(x^{(k)})$, $\alpha_k > 0$ 时, 等式成立。

由于 $-\|\nabla f(x^{(k)})\|_2 \|x^{(k+1)} - x^{(k)}\|_2$ 是非负的, 此时 $f(x^{(k+1)}) - f(x^{(k)}) \leq 0$ 。

迭代公式为

$$x^{(k+1)} = x^{(k)} - \alpha_k \nabla f(x^{(k)}), f(x^{(k+1)}) < f(x^{(k)})$$

Definition 13.7.1 — 梯度下降法求解最小二乘法.

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2, \quad A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$$

令

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

则 f 为凸函数, 并有 $\nabla f(x) = A^T(Ax - b)$ 。

则 $A^T A \in \mathbb{R}^{n \times n}$ 。如果列向量线性相关会导致其不可逆或 n 非常大。可以通过梯度下降法迭代

$$x^{(k+1)} = x^{(k)} - \alpha_k \nabla f(x^{(k)}), f(x^{(k+1)}) < f(x^{(k)})$$

求解

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} A^T (Ax^{(k)} - b)$$

Algorithm 15: 梯度下降法

```

1 初始化  $x^{(0)}$ ,  $k = 0$ 
2 while Not Convergent do
3    $p^{(k)} = A^T(Ax^{(k)} - b)$ 
4    $x^{(k+1)} = x^{(k)} - \alpha^{(k)} p^{(k)}$ 
5    $k \leftarrow k + 1$ 
6 end

```

13.8 估计学习率 (步长) α

Problem 13.5

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - b\|_2^2, \quad A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$$

令 $f(x) = \frac{1}{2} \|Ax - b\|_2^2$

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)} A^T (Ax^{(k)} - b)$$

需要估计 $\alpha^{(k)}$ 。

为了估计 $\alpha^{(k)}$, 通过线性搜索估计:

$$\alpha^{(k)} = \arg \min_{\alpha \in \mathfrak{R}} f(x^{(k)} - \alpha A^T (Ax^{(k)} - b))$$

即 $\alpha^{(k)}$ 是最优步长。在上面的优化式中 $x^{(k)}$ 、 A 、 b 均视为定值。

Theorem 13.8.1 — 线性搜索估计的最优步长.

$$\alpha^{(k)} = \frac{\|A^T (Ax^{(k)} - b)\|_2^2}{\|AA^T (Ax^{(k)} - b)\|_2^2}$$

Proof. 令 $g(\alpha) = f(x^{(k)} - \alpha A^T (Ax^{(k)} - b))$ 是关于 α 的凸函数, 则有

$$\min_{\alpha} g(\alpha) \Rightarrow g'(\alpha) = 0 \Rightarrow \alpha^{(k)} = \frac{\|A^T (Ax^{(k)} - b)\|_2^2}{\|AA^T (Ax^{(k)} - b)\|_2^2}$$

$$\begin{aligned} f(x) &= \frac{1}{2} \|Ax - b\|_2^2, g\left(\alpha^{(k)}\right) = f\left(x^{(k)} - \alpha^{(k)} A^T (Ax^{(k)} - b)\right) \\ \Rightarrow g\left(\alpha^{(k)}\right) &= \frac{1}{2} \left\| A \left(x^{(k)} - \alpha^{(k)} A^T (Ax^{(k)} - b) \right) - b \right\|_2^2 \\ &= \frac{1}{2} \left\| \left(Ax^{(k)} - b \right) - \left(\alpha^{(k)} A^T (Ax^{(k)} - b) \right) \right\|_2^2 \\ &= \frac{1}{2} \left(\left(Ax^{(k)} - b \right)^T \left(Ax^{(k)} - b \right) + \left(\alpha^{(k)} A^T (Ax^{(k)} - b) \right)^T \left(\alpha^{(k)} A^T (Ax^{(k)} - b) \right) \right) \\ &\quad - \left(Ax^{(k)} - b \right)^T \left(\alpha^{(k)} A^T (Ax^{(k)} - b) \right) \\ \Rightarrow g'\left(\alpha^{(k)}\right) &= \alpha^{(k)} \left(A^T (Ax^{(k)} - b) \right)^T \left(A^T (Ax^{(k)} - b) \right) - \left(Ax^{(k)} - b \right)^T \left(A^T (Ax^{(k)} - b) \right) = 0 \\ \Rightarrow \alpha^{(k)} &= \frac{\|A^T (Ax^{(k)} - b)\|_2^2}{\|AA^T (Ax^{(k)} - b)\|_2^2} \end{aligned}$$

Algorithm 16: 使用线性搜索估计步长的梯度下降法

```
1 初始化  $x^{(0)}$ ,  $k = 0$ 
2 while Not Convergent do
3    $p^{(k)} = A^T (Ax^{(k)} - b)$ 
4    $\alpha^{(k)} = \frac{\|p^{(k)}\|_2^2}{\|Ap^{(k)}\|_2^2}$ 
5    $x^{(k+1)} = x^{(k)} - \alpha^{(k)} p^{(k)}$ 
6 end
```

14. Multi-objective Least Squares

14.1 Definition of Multi-objective Least Squares

Problem 14.1 假设有以下多个目标

$$J_1(x) = \|A_1x - b_1\|_2^2, \dots, J_k(x) = \|A_kx - b_k\|_2^2$$

矩阵 $A_i \in \mathbb{R}^{m_i \times n}$, 向量 $b_i \in \mathbb{R}^{m_i}$;

寻找一个向量 $x \in \mathbb{R}^n$ 使得这 k 个目标 $J_i(x), i = 1, \dots, k$ 最小。

可以将上述多目标规划问题转换为加权最小二乘法问题。

Problem 14.2 — 加权最小二乘法问题.

$$\min_x J(x)$$

$$\begin{aligned} J(x) &= \lambda_1 \|A_1x - b_1\|_2^2 + \dots + \lambda_k \|A_kx - b_k\|_2^2 \\ &= \left\| \sqrt{\lambda_1} (A_1x - b_1) \right\|_2^2 + \dots + \left\| \sqrt{\lambda_k} (A_kx - b_k) \right\|_2^2 \end{aligned}$$

$\lambda_i > 0, i = 1, \dots, k$, 表示不同目标的相对重要程度。

利用 ℓ_2 范数平方的可加性, 目标函数 $J(x)$ 可以写成紧密形式:

Problem 14.3 — 加权最小二乘法问题紧密形式.

$$J(x) = \left\| \begin{bmatrix} \sqrt{\lambda_1} (A_1x - b_1) \\ \vdots \\ \sqrt{\lambda_k} (A_kx - b_k) \end{bmatrix} \right\|_2^2$$

进一步可简化为

Problem 14.4 — 加权最小二乘法问题矩阵形式.

$$J(x) = \|\tilde{A}x - \tilde{b}\|_2^2$$

其中

$$\tilde{A} = \begin{bmatrix} \sqrt{\lambda_1}A_1 \\ \vdots \\ \sqrt{\lambda_k}A_k \end{bmatrix}, \quad \tilde{b} = \begin{bmatrix} \sqrt{\lambda_1}b_1 \\ \vdots \\ \sqrt{\lambda_k}b_k \end{bmatrix}$$

因此将多目标问题转化为单目标问题，使用最小二乘法进行求解。

Problem 14.5 — 双目标规划问题.

$$\min_x \|A_1x - b_1\|_2^2 + \lambda \|A_2x - b_2\|_2^2, A_1, A_2 \in \mathbb{R}^{10 \times 5}$$

λ 的变化会影响解的状态。

14.2 求解多目标最小二乘问题

Problem 14.6

$$J(x) = \|\tilde{A}x - \tilde{b}\|_2^2, \quad \tilde{A} = \begin{bmatrix} \sqrt{\lambda_1}A_1 \\ \vdots \\ \sqrt{\lambda_k}A_k \end{bmatrix}, \quad \tilde{b} = \begin{bmatrix} \sqrt{\lambda_1}b_1 \\ \vdots \\ \sqrt{\lambda_k}b_k \end{bmatrix}$$

Theorem 14.2.1 如果 \tilde{A} 的列向量线性无关时，则该问题的解唯一。

$$\begin{aligned} \hat{x} &= (\tilde{A}^T \tilde{A})^{-1} \tilde{A}^T \tilde{b} \\ &= (\lambda_1 A_1^T A_1 + \cdots + \lambda_k A_k^T A_k)^{-1} (\lambda_1 A_1^T b_1 + \cdots + \lambda_k A_k^T b_k) \end{aligned}$$

可对 \tilde{A} 进行 QR 分解计算 \hat{x} 。每一个矩阵 A_i 的行向量可以线性相关。

14.3 正则化数据拟合

Problem 14.7 线性模型拟合数据 $(x^{(1)}, y^{(1)}), \dots, (x^{(N)}, y^{(N)})$

$$\hat{f}(x) = \theta_1 f_1(x) + \cdots + \theta_p f_p(x), \quad \theta = [\theta_1, \dots, \theta_p]^T$$

$f_1(x)$ 为常数函数，且恒等于 1。

较大的参数 θ_i 会让模型对 $f_i(x)$ 的变化更加敏感。让参数 $\theta_2, \dots, \theta_p$ 更小，可以避免模型过拟合。

即可引出两个目标函数：

Problem 14.8

$$J_1(\theta) = \sum_{k=1}^N \left(\hat{f}(x^{(k)}) - y^{(k)} \right)^2, \quad J_2(\theta) = \sum_{j=2}^p \theta_j^2$$

首要目标 $J_1(\theta)$ 是误差的平方和。

改写成加权最小二乘的形式

Problem 14.9

$$\min_{\theta} J_1(\theta) + \lambda J_2(\theta) = \sum_{k=1}^N \left(\hat{f}(x^{(k)}) - y^{(k)} \right)^2 + \lambda \sum_{j=2}^p \theta_j^2$$

正则化参数 $\lambda > 0$ 。

该问题等价于最小二乘法问题:

Problem 14.10

$$\min_{\theta} \left\| \begin{bmatrix} A_1 \\ \sqrt{\lambda} A_2 \end{bmatrix} \theta - \begin{bmatrix} y_{d \times 1} \\ 0 \end{bmatrix} \right\|_2^2$$

$$A_1 = \begin{bmatrix} 1 & f_2(x^{(1)}) & \dots & f_p(x^{(1)}) \\ 1 & f_2(x^{(2)}) & \dots & f_p(x^{(2)}) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & f_2(x^{(N)}) & \dots & f_p(x^{(N)}) \end{bmatrix}, A_2 = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}, y = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(N)} \end{bmatrix}$$

参数 λ 越大, 会迫使 θ 得到接近零解。

14.4 图像逆问题

Problem 14.11

$$y = Ax_{ex} + v$$

向量 $x_{ex} \in \mathbb{R}^n$ 表示未知原始信息 (需要估计), 向量 $v \in \mathbb{R}^m$ 表示未知的误差或者噪声, 向量 $y \in \mathbb{R}^m$ 为观测的已知数据, 矩阵 $A \in \mathbb{R}^{m \times n}$ 将测量值 y 和原始信息 x_{ex} 之间的关系。

使用最小二乘估计法进行估计

Problem 14.12 — 最小二乘法进行估计.

$$\min_x \|Ax - y\|_2^2$$

利用未知 x_{ex} 先验信息, 对目标进行约束, 构成多目标优化问题。

例如可以使用岭回归进行求解。

Definition 14.4.1 — Tikhonov 正则化.

$$\min_x \|Ax - y\|_2^2 + \lambda \|x\|_2^2, \lambda > 0$$

目标在于使 $\|Ax - y\|_2^2$ 足够小, 同时 x 的能量也要小。

Theorem 14.4.1 — 岭回归问题的标准方程. 该优化模型等价于求解

$$(A^T A + \lambda I) x = A^T y$$

即使矩阵 A 的列线性相关时, 也有唯一解。

14.4.1 差分矩阵平滑最小二乘解

Definition 14.4.2 — 图像转换为列向量存储. 二维图像 $X \in \mathbb{R}^{M \times N}$, 可按列存储成向量 $x \in \mathbb{R}^{MN}$:

$$x = \begin{bmatrix} X_{1:M,1} \\ X_{1:M,2} \\ \vdots \\ X_{1:M,N} \end{bmatrix}$$

■ Example 14.1 — reshape.

$$X = \begin{bmatrix} 1 & 2 \\ 4 & 5 \end{bmatrix} \Rightarrow x = \begin{bmatrix} 1 \\ 4 \\ 2 \\ 5 \end{bmatrix}$$

对于前面的优化问题

$$y = Ax_{ex} + v$$

可以假设

Proposition 14.4.2 — 图像逆问题先验假设. 图像具有光滑性: 图像相邻两个像素值之间变化不大。

- 水平方向: $X[n_1, n_2 + 1] \approx X[n_1, n_2]$
- 垂直方向: $X[n_1 + 1, n_2] \approx X[n_1, n_2]$

用差分矩阵进行平滑。

Definition 14.4.3 — 垂直差分矩阵. 垂直差分矩阵是大小为 $N \times N$ 的块矩阵, 每块大小 $(M - 1) \times M$ 。

$$D_v = \begin{bmatrix} D & 0 & \cdots & 0 \\ 0 & D & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \vdots & D \end{bmatrix}, D = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & -1 & 1 \end{bmatrix}$$

■ Example 14.2

$$X = \begin{bmatrix} 1 & 2 \\ 4 & 5 \end{bmatrix} \Rightarrow x = \begin{bmatrix} 1 \\ 4 \\ 2 \\ 5 \end{bmatrix}, D_v x \Rightarrow \begin{bmatrix} 4 - 1 \\ 5 - 2 \end{bmatrix}$$

Definition 14.4.4 水平差分矩阵: 大小为 $(N - 1) \times N$ 的块矩阵, 每块大小 $M \times M$:

$$D_h = \begin{bmatrix} -I_{M,M} & I_{M,M} & 0 & \cdots & 0 & 0 & 0 \\ 0 & -I_{M,M} & I_{M,M} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & -I_{M,M} & I_{M,M} \end{bmatrix}$$

■ Example 14.3

$$X = \begin{bmatrix} 1 & 2 \\ 4 & 5 \end{bmatrix} \Rightarrow x = \begin{bmatrix} 1 \\ 4 \\ 2 \\ 5 \end{bmatrix}, D_h x \Rightarrow \begin{bmatrix} 2 - 1 \\ 5 - 4 \end{bmatrix}$$

定义优化问题

Problem 14.13

$$\hat{x} = \arg \min_x \|Ax - y\|_2^2 + \lambda \|D_v x\|_2^2 + \lambda \|D_h x\|_2^2, \lambda > 0$$

$\|Ax - y\|_2^2$ 称为保证项: 保证 $A\hat{x} \approx y$.

$\lambda \|D_v x\|_2^2 + \lambda \|D_h x\|_2^2$ 为惩罚项, 惩罚相邻像素值的差异变化

$$\|D_h x\|_2^2 + \|D_v x\|_2^2 = \sum_{i=1}^M \sum_{j=1}^{N-1} (X_{i,j+1} - X_{i,j})^2 + \sum_{i=1}^{M-1} \sum_{j=1}^N (X_{i+1,j} - X_{i,j})^2$$

求解这个模型可以得到图像逆退化的近似变换。

14.5 信号去噪

Problem 14.14 — 信号去噪问题. 观察信号向量 $y \in \mathbb{R}^n$,

$$y = x_{ex} + v$$

$x_{ex} \in \mathbb{R}^n$ 是未知信号, $v \in \mathbb{R}^n$ 是噪声。

目标是找一个近似信号 x 变换缓慢, 信号既有光滑性, 同时逼近 y , 其优化模型为

$$\min_x \|x - y\|_2^2 + \lambda \sum_{i=1}^{n-1} (x_{i+1} - x_i)^2, \lambda > 0$$

Definition 14.5.1 — 差分矩阵. 令矩阵 $D \in \mathbb{R}^{(n-1) \times n}$ 为差分矩阵:

$$D = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & -1 & 1 \end{bmatrix}$$

则有 $\sum_{i=1}^{n-1} (x_{i+1} - x_i)^2 = \|Dx\|_2^2$
优化模型等价于

$$\min_x \left\| \begin{bmatrix} I \\ \sqrt{\lambda} D \end{bmatrix} x - \begin{bmatrix} y \\ 0 \end{bmatrix} \right\|_2^2$$

优化模型等价于求解线性方程

$$(I + \lambda D^T D) x = y$$

当 $\lambda \rightarrow 0$, $\hat{x}(\lambda) \rightarrow y$; 当 $\lambda \rightarrow \infty$, $\hat{x}(\lambda) \rightarrow avg(y)1$

15. Constrained Least Squares

15.1 Karush–Kuhn–Tucker Conditions

Problem 15.1

$$\begin{aligned} \min_x & \{f(x) = 2x_1^2 + x_2^2\} \\ \text{s.t. } & h(x) = x_1 + x_2 - 1 = 0 \end{aligned}$$

直接利用无约束优化问题求解：

$$\nabla f(x) = \begin{bmatrix} 4x_1 \\ 2x_2 \end{bmatrix} = 0 \Rightarrow x = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

显然不满足约束条件 $x_1 + x_2 - 1 = 0 + 0 - 1 \neq 0$, 不是优化问题的解。

由约束条件可得 $x_1 = 1 - x_2$ ，代入目标函数则有

$$f(x) = 3x_2^2 - 4x_2 + 2$$

即当 $\hat{x}_2 = \frac{2}{3}$ 时，目标函数值最小，并有 $\hat{x}_1 = 1 - \hat{x}_2 = \frac{1}{3}$ 。

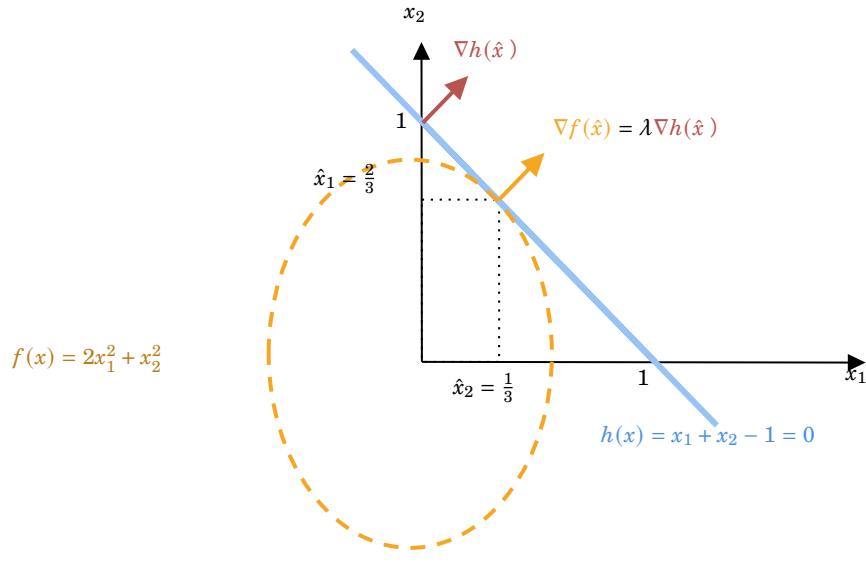
为了惩罚这个解未能满足约束条件，引入拉格朗日函数 (Lagrange Function)

$$L(x, \lambda) = f(x) - \lambda h(x) = 2x_1^2 + x_2^2 + \lambda(1 - x_1 - x_2)$$

对拉格朗日函数进行求导

$$\left. \begin{aligned} \frac{\partial L}{\partial x_1} &= 4x_1 - \lambda = 0 \\ \frac{\partial L}{\partial x_2} &= 2x_2 - \lambda = 0 \\ \frac{\partial L}{\partial \lambda} &= 1 - x_1 - x_2 = 0 \end{aligned} \right\} \Rightarrow \hat{x}_1 = \frac{1}{3}, \hat{x}_2 = \frac{2}{3}, \hat{\lambda} = \frac{4}{3}$$

Figure 15.1: The Geometry of the problem

**Definition 15.1.1 — 拉格朗日乘子法和 Lagrange Functions.**

$$\begin{aligned} & \min_x / \max_x f(x) \\ \text{s.t. } & h_i(x) = 0, i \in I \triangleq \{1, \dots, p\} \\ & g_j(x) \leq 0, j \in J \triangleq \{1, \dots, q\} \end{aligned}$$

$\lambda_i \in \mathbb{R}, i \in I, u_j \in \mathbb{R}^+, j \in J$ 称为拉格朗日乘子 (Lagrange Multipliers)。
引入拉格朗日函数

$$L(x, \lambda, u) = f(x) - \sum_{i \in I} \lambda_i h_i(x) - \sum_{j \in J} u_j g_j(x)$$

设 $\lambda = \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_p \end{bmatrix}, u = \begin{bmatrix} u_1 \\ \vdots \\ u_q \end{bmatrix}, h(x), g(x)$ 如果可以视为一个向量
则拉格朗日乘子式可写成

$$L(x, \lambda, u) = f(x) - \lambda^T h(x) - u^T g(x)$$

拉格朗日乘子法求得的解要使拉格朗日函数的梯度为 0。

Theorem 15.1.1 — 拉格朗日函数求导. 对拉格朗日函数进行求导，有

$$\begin{aligned} \nabla_x L(x, \lambda, u) &= \nabla_x f(x) - \sum_{i \in I} \lambda_i \nabla_x h_i(x) - \sum_{j \in J} u_j \nabla_x g_j(x) = 0 \\ \nabla_x f(x) &= \sum_{i \in I} \lambda_i \nabla_x h_i(x) + \sum_{j \in J} u_j \nabla_x g_j(x) \end{aligned}$$

与图15.1中的结论相符。

Theorem 15.1.2 — Karush-Kuhn-Tucker Conditions. 对于优化问题

$$\begin{aligned} & \min_x / \max_x f(x) \\ \text{s.t. } & h_i(x) = 0, i \in I \triangleq \{1, \dots, p\} \\ & g_j(x) \leq 0, j \in J \triangleq \{1, \dots, q\} \end{aligned}$$

KKT 条件包括:

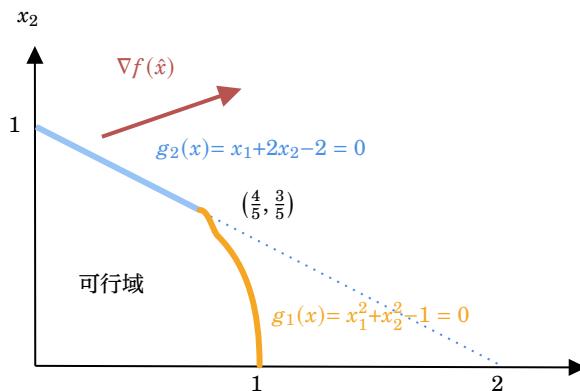
- $h_i(x) = 0, i \in I$
- $\lambda_i h_i(x) = 0, i \in I$
- $g_j(x) \leq 0, j \in J$
- $u_j g_j(x) = 0, j \in J$
- $u_j \geq 0, j \in J$

15.1.1 An Example for Karush-Kuhn-Tucker Conditions

Problem 15.2 求解

$$\begin{aligned} & \max_x \{f(x) = 20x_1 + 10x_2\} \\ \text{s.t. } & g_1(x) = x_1^2 + x_2^2 - 1 \leq 0 \\ & g_2(x) = x_1 + 2x_2 - 2 \leq 0 \\ & g_3(x) = -x_1 \leq 0 \\ & g_4(x) = -x_2 \leq 0 \end{aligned}$$

Figure 15.2: An Example for Karush-Kuhn-Tucker Conditions



对拉格朗日函数求导

$$\nabla_x L(x, u) = \nabla_x f(x) - u_1 \nabla_x g_1(x) - u_2 \nabla_x g_2(x) - u_3 \nabla_x g_3(x) - u_4 \nabla_x g_4(x) = 0, u_j \geq 0$$

计算各个部分, 可得

$$\nabla_x f(x) = \begin{bmatrix} 20 \\ 10 \end{bmatrix}, \nabla_x g_1(x) = \begin{bmatrix} 2x_1 \\ 2x_2 \end{bmatrix}, \nabla_x g_2(x) = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \nabla_x g_3(x) = \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \nabla_x g_4(x) = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

检测边界条件, 可知

$$\nabla_x f(x) = u_1 \nabla_x g_1(x), \nabla_x f(x) \neq u_2 \nabla_x g_2(x), \nabla_x f(x) \neq u_3 \nabla_x g_3(x), \nabla_x f(x) \neq u_4 \nabla_x g_4(x)$$

即 $\nabla_x f(x)$ 只可能是 $u_1 \nabla_x g_1(x)$ 的线性组合。

所以

$$u_2 = u_3 = u_4 = 0, \quad u_1 \neq 0$$

即

$$\begin{aligned}\nabla_x f(x) &= u_1 \nabla_x g_1(x) \\ \begin{bmatrix} 20 \\ 10 \end{bmatrix} &= u_1 \begin{bmatrix} 2x_1 \\ 2x_2 \end{bmatrix} \\ x_1 &= 2x_2\end{aligned}$$

由 KKT 条件得 $u_j g_j(x) = 0, j \in J$ 。由于 $u_2 = u_3 = u_4 = 0, u_1 \neq 0$, 所以 $g_1(x) = 0$ 。将上式代入

$$g_1(x) = x_1^2 + x_2^2 - 1 = 5x_2^2 - 1 = 0$$

$$\text{由于 } x_2 \geq 0, \text{ 可得 } x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \frac{\sqrt{5}}{5} \begin{bmatrix} 2 \\ 1 \end{bmatrix}, u_1 = 5\sqrt{5}.$$

15.2 优化问题：最小范数优化问题

Problem 15.3

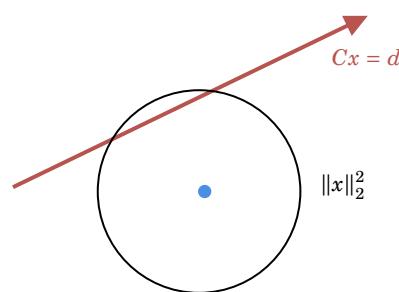
$$\begin{aligned}\min_x \|x\|_2^2 \\ \text{s.t. } Cx = d\end{aligned}$$

矩阵 $C \in \mathbb{R}^{p \times n}$, 向量 $d \in \mathbb{R}^p$.

在大多数应用中 $p < n, Cx = d$ 是一个欠定方程。这时, $Cx = d$ 表示为一条直线。它的几何意义是在直线 $Cx = d$ 中, 寻找范数最小的解。

假如 $p = n, Cx = d$ 表示一个点。

Figure 15.3: 最小范数优化问题 ($p < n$)



假设矩阵 C 行向量线性无关时, 有:

- 对任意一个 $d, Cx = d$ 至少有一个解;
- 矩阵 C 为宽的或者方的 ($p \leq n$);
- 当 $p < n$, 有无穷多个解。

15.2.1 最小范数优化问题的例子

Problem 15.4 矩阵 $C \in \mathbb{R}^{2 \times 10}$, 向量 $d \in \mathbb{R}^2$:

$$Cx = d$$

$$\underbrace{\begin{bmatrix} 19/2 & 17/2 & 15/2 & \cdots & 1/2 \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix}}_C \underbrace{x}_{\begin{bmatrix} 1 \\ 0 \end{bmatrix}} = \underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}_d$$

这个方程只有一组含有两个非 0 元素的解。

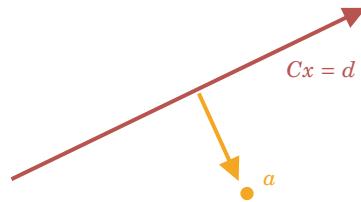
$$x = \begin{bmatrix} 1 \\ -1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad x = \begin{bmatrix} 0 \\ 1 \\ -1 \\ \vdots \\ 0 \end{bmatrix}, \dots$$

15.3 优化问题：点到线的最短距离

Problem 15.5 给定点 $a \neq 0$ ，点到线最短距离问题：

$$\begin{aligned} \min_x \quad & \|x - a\|_2^2 \\ \text{s.t.} \quad & Cx = d \end{aligned}$$

Figure 15.4: 点到线的最短距离



令 $y = x - a$, 则点到线的最短距离问题，可等价为最小范数问题：

Problem 15.6

$$\begin{aligned} \min_y \quad & \|y\|_2^2 \\ \text{s. t.} \quad & Cy = d - Ca \end{aligned}$$

则 $x = y + a$ 是点到线最短距离问题的解。

15.4 最小范数优化问题推导

Problem 15.7

$$\begin{aligned} \min_x \quad & \frac{1}{2} \|x\|_2^2 \\ \text{s.t.} \quad & Cx = d \end{aligned}$$

求其最优解。

引入拉格朗日函数

$$L(x, \lambda) = \frac{1}{2} \|x\|_2^2 - \lambda^T (Cx - d)$$

① λ 是一个向量。

对拉格朗日函数求导

$$\nabla_x L(x, \lambda) = x - C^T \lambda = 0 \Rightarrow x = C^T \lambda$$

由矩阵 C 行线性无关可得

$$Cx = CC^T \lambda = d \Rightarrow \lambda = (CC^T)^{-1} d$$

则有

$$\hat{x} = C^T \lambda = C^T (CC^T)^{-1} d = C^\dagger d$$

以上得到的是 $\hat{x} = C^\dagger d$ 是问题最优解的必要条件。还需要证明它是最优解。

Theorem 15.4.1 $\hat{x} = C^\dagger d$ 是问题最优解。

Proof. 1. 首先证明解 \hat{x} 满足等式 $\hat{x} = C^T \lambda = C^T (CC^T)^{-1} d = C^\dagger d$.

$$C\hat{x} = CC^T (CC^T)^{-1} d = d$$

2. 在 $Cx = d, x \neq \hat{x}$ 的情况下

$$\begin{aligned} \hat{x}^T (x - \hat{x}) &= d^T (CC^T)^{-1} C(x - \hat{x}) \\ &= d^T (CC^T)^{-1} (Cx - C\hat{x}) \\ &= d^T (CC^T)^{-1} (d - d) \\ &= 0 \end{aligned}$$

3. 在 $Cx = d, x \neq \hat{x}$ 的情况下, 证明 $\|x\|_2^2 > \|\hat{x}\|_2^2$.

$$\begin{aligned} \|x\|_2^2 &= \|\hat{x} + x - \hat{x}\|_2^2 \\ &= \|\hat{x}\|_2^2 + 2\hat{x}^T (x - \hat{x}) + \|x - \hat{x}\|_2^2 \\ &= \|\hat{x}\|_2^2 + \|x - \hat{x}\|_2^2 \\ &> \|\hat{x}\|_2^2 \end{aligned}$$

■

15.5 QR 分解求解最小范数优化问题

对矩阵 $C^T \in \mathbb{R}^{n \times p}$ 进行 QR 分解, $C^T = QR$

$$\begin{aligned} \hat{x} &= C^T (CC^T)^{-1} d \\ &= QR (R^T Q^T QR)^{-1} d \\ &= QR (R^T R)^{-1} d \\ &= Q (R^{-1})^T d \end{aligned}$$

Algorithm 17: QR 分解求解最小范数优化问题

Input: QR 分解得到的矩阵 Q 、 R , 向量 d

Output: \hat{x}

1 求解 $R^T z = d$

2 求解 $\hat{x} = Qz$

Proof.

$$\begin{aligned}\hat{x} &= Q \underbrace{\left(R^T \right)^{-1} d}_{\textcolor{red}{z}} \\ \textcolor{red}{z} &= \left(R^T \right)^{-1} d \Rightarrow R^T \textcolor{red}{z} = d \\ \hat{x} &= Q \textcolor{red}{z}\end{aligned}$$

■

15.5.1 QR 分解求解最小范数优化问题的复杂度

总复杂度: $\approx 2np^2$ flops

- 矩阵 $C^T = QR$ 分解, $C^T = QR$ ($2np^2$ flops)
- 回代法求解 $R^T z = d$ (p^2 flops)
- 计算 $\hat{x} = Qz$ ($2np$ flops)

15.5.2 QR 分解求解最小范数优化问题的例子

■ Example 15.1

$$C = \begin{bmatrix} 1 & -1 & 1 & 1 \\ 1 & 0 & 1/2 & 1/2 \end{bmatrix}, \quad d = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

对矩阵 C^T 进行 QR 分解, $C^T = QR$

$$\begin{bmatrix} 1 & 1 \\ -1 & 0 \\ 1 & 1/2 \\ 1 & 1/2 \end{bmatrix} = \begin{bmatrix} 1/2 & 1/\sqrt{2} \\ -1/2 & 1/\sqrt{2} \\ 1/2 & 0 \\ 1/2 & 0 \end{bmatrix} \begin{bmatrix} 2 & 1 \\ 0 & 1/\sqrt{2} \end{bmatrix}$$

通过回代法求解 $R^T Z = d$

$$\begin{bmatrix} 2 & 0 \\ 1 & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

解得 $z_1 = 0, z_2 = \sqrt{2}$ 。

可得 $\hat{x} = Qz = (1, 1, 0, 0)$

■

15.6 最小二乘法约束问题

Problem 15.8 — 分段多项式拟合问题.

$$\begin{aligned}\min_x \|Ax - b\|_2^2 \\ \text{s.t. } Cx = d\end{aligned}$$

矩阵 $A \in \mathbb{R}^{m \times n}, C \in \mathbb{R}^{p \times n}$, 向量 $b \in \mathbb{R}^m, d \in \mathbb{R}^p$

■

在大多数应用中 $p < n, Cx = d$ 是一个欠定方程。它的几何意义是在直线 $Cx = d$ 中，寻找 $\|Ax - b\|_2^2$ 最小的解。(此时 $\|Ax - b\|_2^2$ 表示一个椭圆)。

特殊情况：

- 当 $p = 0$ 时，则其转化为无约束的最小二乘法问题。
- 当 $A = I, b = 0$ 时，则其为最小范数问题。

15.7 优化问题：分段多项式拟合问题

Problem 15.9 假设样本点 $(x_1, y_1), \dots, (x_N, y_N) \in \mathbb{R}^2$ ，有

$$x_1, \dots, x_M \leq a, x_{M+1}, \dots, x_N > a$$

设 d 阶多项式

$$\begin{aligned} f(x) &= \theta_1 + \theta_2 x + \dots + \theta_d x^{d-1} \\ g(x) &= \theta_{d+1} + \theta_{d+2} x + \dots + \theta_{2d} x^{d-1} \end{aligned}$$

两个多项式 $f(x), g(x)$ 对于样本点 $(x_1, y_1), \dots, (x_N, y_N)$ 进行拟合

$$\begin{aligned} f(x_i) &\approx y_i, x_i \leq a \\ g(x_i) &\approx y_i, x_i > a \end{aligned}$$

拟合要求：函数值和导数值必须在分段位置 a 连续

$$f(a) = g(a), f'(a) = g'(a)$$

对于 $\begin{cases} f(x_i) \approx y_i, x_i \leq a \\ g(x_i) \approx y_i, x_i > a \end{cases}$ ，可以构造矩阵

Notation 15.1.

$$A = \begin{bmatrix} 1 & x_1 & \cdots & x_1^{d-1} & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 1 & x_M & \cdots & x_M^{d-1} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 1 & x_{M+1} & \cdots & x_{M+1}^{d-1} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & 1 & x_N & \cdots & x_N^{d-1} \end{bmatrix}, \theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_d \\ \theta_{d+1} \\ \vdots \\ \theta_{2d} \end{bmatrix}, b = \begin{bmatrix} y_1 \\ \vdots \\ y_M \\ y_{M+1} \\ \vdots \\ y_N \end{bmatrix}$$

则可以转化为

$$A\theta \approx b$$

对于约束条件 $f(a) = g(a), f'(a) = g'(a)$ ，可以构造矩阵

Notation 15.2.

$$C = \begin{bmatrix} 1 & a & \cdots & a^{d-1} & -1 & -a & \cdots & -a^{d-1} \\ 0 & 1 & \cdots & (d-1)a^{d-2} & 0 & -1 & \cdots & -(d-1)a^{d-2} \end{bmatrix}, \theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_d \\ \theta_{d+1} \\ \vdots \\ \theta_{2d} \end{bmatrix}, d = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

所以

$$C\theta = d$$

对于分段多项式拟合问题

Problem 15.10 — 分段多项式拟合问题. 假设样本点 $(x_1, y_1), \dots, (x_N, y_N) \in \mathbb{R}^2$, 有 $x_1, \dots, x_M \leq a, x_{M+1}, \dots, x_N > a$.

$$\begin{array}{ll} \min_{\theta} & \sum_{i=1}^M (f(x_i) - y_i)^2 + \sum_{i=M+1}^N (g(x_i) - y_i)^2 \\ \text{s.t.} & f(a) = g(a), f'(a) = g'(a) \end{array}$$

最终可以转换成矩阵形式

Problem 15.11 — 分段多项式拟合问题 (矩阵形式) .

$$\begin{array}{ll} \min_{\theta} & \|A\theta - b\|_2^2 \\ \text{s.t.} & C\theta = d \end{array}$$

15.8 先验假设

Proposition 15.8.1 堆叠矩阵的列向量线性无关

$$\left[\begin{array}{c} A \\ C \end{array} \right] \in \mathbb{R}^{(m+p) \times n}$$

Proposition 15.8.2 矩阵 $C \in \mathbb{R}^{p \times n}$ 行线性无关

假设 1 是一个比 A 可右逆更弱的条件。

假设 $p \leq n \leq m + p$ 。

15.9 优化问题：最小二乘法的带约束 KKT 条件

Problem 15.12

$$\begin{array}{ll} \min_x & \frac{1}{2} \|Ax - b\|_2^2 \\ \text{s.t.} & Cx = d \end{array}$$

引入拉格朗日函数

$$L(x, z) = \frac{1}{2} \|Ax - b\|_2^2 - z^T(d - Cx), z \in \mathbb{R}^p$$

对拉格朗日函数求导

$$\begin{aligned} \nabla_x L(x, z) &= A^T(Ax - b) + C^Tz = 0 \\ \nabla_z L(x, z) &= Cx - d = 0 \end{aligned} \tag{15.1}$$



使用矩阵导数

$$\begin{aligned}
 \frac{\partial}{\partial \mathbf{X}} \text{Tr}(\mathbf{X}^T \mathbf{B} \mathbf{X}) &= \mathbf{B} \mathbf{X} + \mathbf{B}^T \mathbf{X} \\
 \frac{\partial}{\partial \mathbf{X}} \text{Tr}(\mathbf{B} \mathbf{X} \mathbf{X}^T) &= \mathbf{B} \mathbf{X} + \mathbf{B}^T \mathbf{X} \\
 \frac{\partial}{\partial \mathbf{X}} \text{Tr}(\mathbf{X} \mathbf{X}^T \mathbf{B}) &= \mathbf{B} \mathbf{X} + \mathbf{B}^T \mathbf{X} \\
 \frac{\partial}{\partial \mathbf{X}} \text{Tr}(\mathbf{X} \mathbf{B} \mathbf{X}^T) &= \mathbf{X} \mathbf{B}^T + \mathbf{X} \mathbf{B} \\
 \frac{\partial}{\partial \mathbf{X}} \text{Tr}(\mathbf{B} \mathbf{X}^T \mathbf{X}) &= \mathbf{X} \mathbf{B}^T + \mathbf{X} \mathbf{B} \\
 \frac{\partial}{\partial \mathbf{X}} \text{Tr}(\mathbf{X}^T \mathbf{X} \mathbf{B}) &= \mathbf{X} \mathbf{B}^T + \mathbf{X} \mathbf{B}
 \end{aligned}$$

可以求得

$$\begin{aligned}
 \frac{\partial \frac{1}{2} \|Ax - b\|_2^2}{\partial x} &= \frac{\partial \frac{1}{2} (x^T A^T A x - 2x^T A^T b + b^T b)}{\partial x} \\
 &= \frac{\partial \frac{1}{2} \text{tr}(x^T A^T A x)}{\partial x} - \frac{\partial x^T A^T b}{\partial x} \\
 &= \frac{1}{2} \cdot 2A^T A x - A^T b \\
 &= A^T(Ax - b)
 \end{aligned}$$

可以转换成矩阵形式

$$\left[\begin{array}{cc} A^T A & C^T \\ C & 0 \end{array} \right] \left[\begin{array}{c} x \\ z \end{array} \right] = \left[\begin{array}{c} A^T b \\ d \end{array} \right]$$

15.10 KKT 最优条件

Problem 15.13

$$\begin{aligned}
 \min_x \frac{1}{2} \|Ax - b\|_2^2 \\
 s.t. Cx = d
 \end{aligned}$$

Theorem 15.10.1 — 优化条件的 Karush-Kuhn-Tucker(KKT) 等式. 令 \hat{x} 是上述约束优化问题的解, 则有

$$\left[\begin{array}{cc} A^T A & C^T \\ C & 0 \end{array} \right] \left[\begin{array}{c} \hat{x} \\ z \end{array} \right] = \left[\begin{array}{c} A^T b \\ d \end{array} \right], \left[\begin{array}{c} \hat{x} \\ z \end{array} \right] \in \mathbb{R}^{n+p}$$

则它是最优解。

特殊情况:

- 最小二乘法问题: 当 $p = 0$ 时, 即为正规方程 $A^T A \hat{x} = A^T b$
- 最小范数问题: 当 $A = I, b = 0$ 时, 可以推导得到 $C \hat{x} = b, \hat{x} + C^T z = 0$

Proof. 假设 x 满足 $Cx = d$, (\hat{x}, z) 满足 KKT 等式 $\left[\begin{array}{cc} A^T A & C^T \\ C & 0 \end{array} \right] \left[\begin{array}{c} x \\ z \end{array} \right] = \left[\begin{array}{c} A^T b \\ d \end{array} \right]$ 的定义, 即 x 是另一个解。

所以有

$$\begin{aligned}
\|Ax - b\|_2^2 &= \|A(x - \hat{x}) + A\hat{x} - b\|_2^2 \\
&= \|A(x - \hat{x})\|_2^2 + \|A\hat{x} - b\|_2^2 + 2(x - \hat{x})^T A^T (A\hat{x} - b) \\
&= \|A(x - \hat{x})\|_2^2 + \|A\hat{x} - b\|_2^2 - \underbrace{2(x - \hat{x})^T}_{(Cx = C\hat{x} = d)} \underbrace{C^T z}_{(A^T A\hat{x} + C^T z = A^T b)} \\
&= \|A(x - \hat{x})\|_2^2 + \|A\hat{x} - b\|_2^2 \\
&\geq \|A\hat{x} - b\|_2^2
\end{aligned}$$



x 和 \hat{x} 都是 KKT 等式的解，所以 $Cx = C\hat{x} = d$.

由拉格朗日函数求导 15.1，有

$$A^T A\hat{x} + C^T z = A^T b$$

所以 \hat{x} 是最优解。 ■

Theorem 15.10.2 — KKT 等式最优解的唯一性. 最优解 \hat{x} 是唯一的。

Proof. 假设矩阵 A 列线性无关， C 行线性无关，即

$$\begin{aligned}
A^T A(\hat{x} - x) &= 0 \Rightarrow x = \hat{x} \quad (A^T A \text{ 可逆}) \\
C^T (\hat{z} - z) &= 0 \Rightarrow \hat{z} = z
\end{aligned}$$

所以对于方程

$$A^T A\hat{x} + C^T z = A^T b$$

$\begin{bmatrix} x \\ z \end{bmatrix}$ 也是唯一的。 ■

Theorem 15.10.3 如果矩阵 A 列线性无关， C 行线性无关，则矩阵

$$\begin{bmatrix} A^T A & C^T \\ C & 0 \end{bmatrix}$$

为非奇异矩阵。

Proof.

$$\begin{aligned}
\begin{bmatrix} A^T A & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} = 0 &\Rightarrow x^T (A^T A x + C^T z) = 0, Cx = 0 \\
&\Rightarrow \|Ax\|_2^2 + (Cx)^T z = \|Ax\|_2^2 = 0, Cx = 0 \\
&\Rightarrow Ax = 0, Cx = 0 \\
&\Rightarrow x = 0 \quad (A \text{ 列线性无关})
\end{aligned}$$

由于 $A^T A x + C^T z = 0$, 则当 $x = 0$ 时, 有 $z = 0$ 。

所以 C 行线性无关。 ■

Theorem 15.10.4 如果矩阵 A 列线性无关和 C 行线性无关不同时成立，则矩阵

$$\begin{bmatrix} A^T A & C^T \\ C & 0 \end{bmatrix}$$

为奇异矩阵。

Proof. 如果 A 列线性相关，则存在 $x \neq 0$, 使得 $Ax = 0$, 则

$$\begin{bmatrix} A^T A & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ 0 \end{bmatrix} = 0$$

如果 C 行线性相关，则存在 $z \neq 0$, 使得 $C^T z = 0$, 则

$$\begin{bmatrix} A^T A & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} 0 \\ z \end{bmatrix} = 0$$

因此该矩阵为奇异矩阵。 ■

15.11 LU 分解求解 KKT 最优条件

Problem 15.14

$$\begin{aligned} & \min_x \frac{1}{2} \|Ax - b\|_2^2 \\ & s.t. Cx = d \end{aligned}$$

$$\begin{bmatrix} A^T A & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} = \begin{bmatrix} A^T b \\ d \end{bmatrix}$$

求其最优解。 ■

Algorithm 18: LU 分解求解 KKT 最优条件

1 计算 $H = A^T A$

2 计算 $c = A^T b$

3 LU 分解求解

$$\begin{bmatrix} H & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} = \begin{bmatrix} c \\ d \end{bmatrix}$$

15.11.1 LU 分解求解 KKT 最优条件的时间复杂度

Theorem 15.11.1 用 LU 分解法求解下列线性方程

$$\begin{bmatrix} H & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} = \begin{bmatrix} c \\ d \end{bmatrix}$$

需要的时间复杂度是 $\frac{2}{3}(p+n)^3$ flops。

第一步的时间复杂度是 $2mn^2$ flops, 第二步的时间复杂度是 $2mn$ flops, 总时间复杂度为 $2mn^2 + \frac{2}{3}(p+n)^3$ flops。

15.12 QR 分解求解 KKT 最优条件

Problem 15.15

$$\begin{aligned} & \min_x \frac{1}{2} \|Ax - b\|_2^2 \\ & s.t. Cx = d \end{aligned}$$

$$\begin{bmatrix} A^T A & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} = \begin{bmatrix} A^T b \\ d \end{bmatrix}$$

求其最优解。

由于 \hat{x} 满足 $C\hat{x} = d$ ，则有

$$C^T C \hat{x} = C^T d$$

列出拉格朗日函数

$$L(x, z) = \frac{1}{2} \|Ax - b\|_2^2 - z^T(d - Cx)$$

由 KKT 条件可得

$$\begin{aligned} \nabla_x L(x, z) &= A^T(A\hat{x} - b) + C^T z + C^T C \hat{x} - C^T d = 0 \\ &\Rightarrow (A^T A + C^T C) \hat{x} + C^T(z - d) = A^T b \\ \nabla_z L(x, z) &= C\hat{x} - d = 0 \end{aligned}$$

令 $w = z - d$ ，KKT 条件写成矩阵形式

$$\begin{bmatrix} A^T A + C^T C & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} \hat{x} \\ w \end{bmatrix} = \begin{bmatrix} A^T b \\ d \end{bmatrix}$$

假设 $\begin{bmatrix} A \\ C \end{bmatrix}$ 列向量无关，命题 15.8.1) 保证了 $A^T A + C^T C$ 是非奇异的，即存在以下 QR 因子分解

$$\begin{bmatrix} A \\ C \end{bmatrix} = QR = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} R = \begin{bmatrix} Q_1 R \\ Q_2 R \end{bmatrix}$$

代入 QR 分解，可得

$$\begin{bmatrix} R^T R & R^T Q_2^T \\ Q_2 R & 0 \end{bmatrix} \begin{bmatrix} \hat{x} \\ w \end{bmatrix} = \begin{bmatrix} R^T Q_1^T b \\ d \end{bmatrix}$$

将第一个方程两边乘 R^{-T} 和并令变量 $y = R\hat{x}$ ，可得

$$\begin{aligned} \begin{bmatrix} ((R^T)^{-1} R^T) R & ((R^T)^{-1} R^T) Q_2^T \\ Q_2 R & 0 \end{bmatrix} \begin{bmatrix} \hat{x} \\ w \end{bmatrix} &= \begin{bmatrix} ((R^T)^{-1} R^T) Q_1^T b \\ d \end{bmatrix} \\ \begin{bmatrix} R & Q_2^T \\ Q_2 R & 0 \end{bmatrix} \begin{bmatrix} \hat{x} \\ w \end{bmatrix} &= \begin{bmatrix} Q_1^T b \\ d \end{bmatrix} \\ \begin{bmatrix} I & Q_2^T \\ Q_2 & 0 \end{bmatrix} \begin{bmatrix} \hat{x} \\ w \end{bmatrix} &= \begin{bmatrix} Q_1^T b \\ d \end{bmatrix} \end{aligned}$$

$$\begin{bmatrix} I & Q_2^T \\ Q_2 & 0 \end{bmatrix} \begin{bmatrix} \textcolor{blue}{y} \\ w \end{bmatrix} = \begin{bmatrix} Q_1^T b \\ d \end{bmatrix}$$

矩阵 $C = Q_2 R \Rightarrow Q_2 = CR^{-1}$, Q_2 行线性无关

$$Q_2^T u = R^{-T} C^T u = 0 \Rightarrow C^T u = 0 \Rightarrow u = 0$$

因为 C 行线性无关的 (假设 2(15.8.2))。

利用 Q_2^T 的 QR 分解来求解

$$\begin{bmatrix} I & Q_2^T \\ Q_2 & 0 \end{bmatrix} \begin{bmatrix} y \\ w \end{bmatrix} = \begin{bmatrix} Q_1^T b \\ d \end{bmatrix}$$

方程第 1 行可得 $y = Q_1^T b - Q_2^T w$, 并代入第二行

$$Q_2 y = d \Rightarrow Q_2 Q_2^T w = Q_2 Q_1^T b - d$$

用 QR 分解 $Q_2^T = \tilde{Q} \tilde{R}$ 来解这个关于 w 的方程:

$$\tilde{R}^T \tilde{R} w = \tilde{R}^T \tilde{Q}^T Q_1^T b - d$$

上式可以简化为:

$$\tilde{R} w = \tilde{Q}^T Q_1^T b - \tilde{R}^{-T} d$$

$$\begin{bmatrix} A^T A + C^T C & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} \hat{x} \\ w \end{bmatrix} = \begin{bmatrix} A^T b \\ d \end{bmatrix} \quad \tilde{R} w = \tilde{Q}^T Q_1^T b - \tilde{R}^{-T} d$$

算法过程见19。

Algorithm 19: QR 分解求解 KKT 最优条件

1 计算两个 QR 分解

$$\begin{bmatrix} A \\ C \end{bmatrix} = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} R, Q_2^T = \tilde{Q} \tilde{R}$$

2 用前代法求解 $\tilde{R}^T u = d$, 计算 $c = \tilde{Q}^T Q_1^T b - u$

3 用回代法求解 $\tilde{R} w = c$, 计算 $y = Q_1^T b - Q_2^T w$

4 用回代法计算 $R \hat{x} = y$ 。

总时间复杂度: QR 分解有 $2(p+m)n^2 + 2np^2$ 次 flops。

15.13 KKT 最优条件求解复杂度: QR vs LU

假设 $p < n$:

- LU 复杂度: $2mn^2 + (2/3)(p+n)^3 < 2mn^2 + (16/3)n^3$
- QR 复杂度: $2(p+m)n^2 + 2np^2 < 2mn^2 + 4n^3$

稳定性: QR 分解避免直接计算 $A^T A$ 。

15.14 Supplement Material: Karush-Kuhn-Tucker (KKT) 条件¹

15.14.1 等式约束优化问题

Problem 15.16 — 等式约束优化问题. 给定一个目标函数 $f : \mathbb{R}^n \rightarrow \mathbb{R}$, 我们希望找到 $\mathbf{x} \in \mathbb{R}^n$, 在满足约束条件 $g(\mathbf{x}) = 0$ 的前提下, 使得 $f(\mathbf{x})$ 有最小值

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & g(\mathbf{x}) = 0 \end{aligned}$$

为方便分析, 假设 f 与 g 是连续可导函数。Lagrange 乘数法是等式约束优化问题的典型解法。定义

Definition 15.14.1 — Lagrangian 函数.

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x})$$

其中 λ 称为 Lagrange 乘数。

Theorem 15.14.1 Lagrange 乘数法将原本的约束优化问题转换成等价的无约束优化问题

$$\min_{\mathbf{x}, \lambda} L(\mathbf{x}, \lambda)$$

Theorem 15.14.2 — 拉格朗日乘子法最优解必要条件. 计算 L 对 \mathbf{x} 与 λ 的偏导数并设为零, 可得最优解的必要条件:

$$\nabla_{\mathbf{x}} L = \frac{\partial L}{\partial \mathbf{x}} = \nabla f + \lambda \nabla g = \mathbf{0}$$

$$\nabla_{\lambda} L = \frac{\partial L}{\partial \lambda} = g(\mathbf{x}) = 0$$

其中第一式为定常方程式 (*stationary equation*), 第二式为约束条件。

解开上面 $n + 1$ 个方程式可得 $L(\mathbf{x}, \lambda)$ 的驻点 (*stationary point*) \mathbf{x}^* 以及 λ 的值 (正负数皆可能)。

15.14.2 不等式约束优化问题

接下来我们将约束等式 $g(\mathbf{x}) = 0$ 推广为不等式 $g(\mathbf{x}) \leq 0$ 。考虑这个问题

Problem 15.17 — 不等式约束优化问题.

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & g(\mathbf{x}) \leq 0 \end{aligned}$$

约束不等式 $g(\mathbf{x}) \leq 0$ 称为原始可行性 (*primal feasibility*), 据此我们定义可行域 (*feasible region*) $K = \{\mathbf{x} \in \mathbb{R}^n \mid g(\mathbf{x}) \leq 0\}$ 。

假设 \mathbf{x}^* 为满足约束条件的最佳解, 分开两种情况讨论:

- $g(\mathbf{x}^*) < 0$, 最佳解位于 K 的内部, 称为内部解 (*interior solution*), 这时约束条件是无效的 (*inactive*);
- $g(\mathbf{x}^*) = 0$, 最佳解落在 K 的边界, 称为边界解 (*boundary solution*), 此时约束条件是有效的 (*active*).

¹Cited from <https://zhuanlan.zhihu.com/p/38163970>.

这两种情况的最佳解具有不同的必要条件。

- 内部解：在约束条件无效的情形下， $g(\mathbf{x})$ 不起作用，约束优化问题退化为无约束优化问题，因此驻点 \mathbf{x}^* 满足 $\nabla f = \mathbf{0}$ 且 $\lambda = 0$ 。
- 边界解：在约束条件有效的情形下，约束不等式变成等式 $g(\mathbf{x}) = 0$ ，这与前述 Lagrange 乘数法的情况相同。

对于边界解，我们可以证明

Theorem 15.14.3 驻点 \mathbf{x}^* 发生于 $\nabla f \in \text{span } \nabla g$ 。

换句话说，

Corollary 15.14.4 存在 λ 使得 $\nabla f = -\lambda \nabla g$ 。

注意这里 λ 的正负号是有其意义的。

因为我们希望最小化 f ，梯度 ∇f (函数 f 在点 \mathbf{x} 的最陡上升方向) 应该指向可行域 K 的内部 (因为最优解最小值是在边界取得的)，但 ∇g 指向 K 的外部 (即 $g(\mathbf{x}) > 0$ 的区域，因为你的约束是小于等于 0)，因此 $\lambda \geq 0$ ，称为对偶可行性 (*dual feasibility*)。

因此，不论是内部解或边界解， $\lambda g(\mathbf{x}) = 0$ 恒成立，称为互补松弛性 (*complementary slackness*)。

整合上述两种情况，

Theorem 15.14.5 — Karush-Kuhn-Tucker (KKT) 条件. 最佳解的必要条件包括：Lagrangian 函数 $L(\mathbf{x}, \lambda)$ 的定常方程式、原始可行性、对偶可行性，以及互补松弛性：

$$\begin{aligned}\nabla_{\mathbf{x}} L &= \nabla f + \lambda \nabla g = \mathbf{0} \\ g(\mathbf{x}) &\leq 0 \\ \lambda &\geq 0 \\ \lambda g(\mathbf{x}) &= 0\end{aligned}$$

这些条件合称为 *Karush-Kuhn-Tucker (KKT) 条件*。

如果我们要最大化 $f(\mathbf{x})$ 且受限于 $g(\mathbf{x}) \leq 0$ ，那么对偶可行性要改成 $\lambda \leq 0$ 。

上面结果可推广至多个约束等式与约束不等式的情况。考虑标准约束优化问题 (或称非线性规划)：

Definition 15.14.2 — 标准约束优化问题 (非线性规划).

$$\begin{array}{ll}\min & f(\mathbf{x}) \\ \text{s.t.} & g_j(\mathbf{x}) = 0, \quad j = 1, \dots, m \\ & h_k(\mathbf{x}) \leq 0, \quad k = 1, \dots, p\end{array}$$

Theorem 15.14.6 — 标准约束优化的 KKT 条件. 定义 Lagrangian 函数

$$L(\mathbf{x}, \{\lambda_j\}, \{\mu_k\}) = f(\mathbf{x}) + \sum_{j=1}^m \lambda_j g_j(\mathbf{x}) + \sum_{k=1}^p \mu_k h_k(\mathbf{x})$$

其中 λ_j 是对应 $g_j(\mathbf{x}) = 0$ 的 Lagrange 乘数， μ_k 是对应 $h_k(\mathbf{x}) \leq 0$ 的 Lagrange 乘数 (或称 KKT 乘数)。

KKT 条件包括

$$\begin{aligned}\nabla_{\mathbf{x}} L &= \mathbf{0} \\ g_j(\mathbf{x}) &= 0, \quad j = 1, \dots, m \\ h_k(\mathbf{x}) &\leq 0 \\ \mu_k &\geq 0 \\ \mu_k h_k(\mathbf{x}) &= 0, \quad k = 1, \dots, p\end{aligned}$$

15.14.3 拉格朗日乘子法的例子

Problem 15.18 考虑这个问题

$$\begin{array}{ll}\min & x_1^2 + x_2^2 \\ \text{s.t.} & x_1 + x_2 = 1 \\ & x_2 \leq \alpha\end{array}$$

其中 $(x_1, x_2) \in \mathbb{R}^2$, α 为实数。

写出 Lagrangian 函数

$$L(x_1, x_2, \lambda, \mu) = x_1^2 + x_2^2 + \lambda(1 - x_1 - x_2) + \mu(x_2 - \alpha)$$

KKT 方程组如下:

$$\begin{aligned}\frac{\partial L}{\partial x_i} &= 0, \quad i = 1, 2 \\ x_1 + x_2 &= 1 \\ x_2 - \alpha &\leq 0 \\ \mu &\geq 0 \\ \mu(x_2 - \alpha) &= 0\end{aligned}$$

求偏导可得 $\frac{\partial L}{\partial x_1} = 2x_1 - \lambda = 0$ 且 $\frac{\partial L}{\partial x_2} = 2x_2 - \lambda + \mu = 0$, 分别解出 $x_1 = \frac{\lambda}{2}$ 且 $x_2 = \frac{\lambda}{2} - \frac{\mu}{2}$ 。代入约束等式 $x_1 + x_2 = \lambda - \frac{\mu}{2} = 1$ 或 $\lambda = \frac{\mu}{2} + 1$ 。合并上面结果,

$$x_1 = \frac{\mu}{4} + \frac{1}{2}, \quad x_2 = -\frac{\mu}{4} + \frac{1}{2}$$

最后再加入约束不等式 $-\frac{\mu}{4} + \frac{1}{2} \leq \alpha$ 或 $\mu \geq 2 - 4\alpha$ 。分开三种情况讨论。

1. $\alpha > \frac{1}{2}$: 不难验证 $\mu = 0 > 2 - 4\alpha$ 满足所有的 KKT 条件, 约束不等式是无效的, $x_1^* = x_2^* = \frac{1}{2}$ 是内部解, 目标函数的极小值是 $\frac{1}{2}$ 。
2. $\alpha = \frac{1}{2}$: 如同 1, $\mu = 0 = 2 - 4\alpha$ 满足所有的 KKT 条件, $x_1^* = x_2^* = \frac{1}{2}$ 是边界解, 因为 $x_2^* = \alpha$ 。
3. $\alpha < \frac{1}{2}$: 这时约束不等式是有效的, $\mu = 2 - 4\alpha > 0$, 则 $x_1^* = 1 - \alpha$ 且 $x_2^* = \alpha$, 目标函数的极小值是 $(1 - \alpha)^2 + \alpha^2$ 。

15.15 Supplement Material: 浅谈最优化问题的 KKT 条件²

Theorem 15.15.1 — KKT 条件. 对于具有等式和不等式约束的一般优化问题

$$\begin{array}{ll}\min & f(\mathbf{x}) \\ \text{s.t.} & g_j(\mathbf{x}) \leq 0 (j = 1, 2, \dots, m) \\ & h_k(\mathbf{x}) = 0 (k = 1, 2, \dots, l)\end{array}$$

²Cited from <https://zhuanlan.zhihu.com/p/26514613>.

KKT 条件给出了判断 \mathbf{x}^* 是否为最优解的必要条件, 即:

$$\begin{cases} \frac{\partial f}{\partial x_i} + \sum_{j=1}^m \mu_j \frac{\partial g_j}{\partial x_i} + \sum_{k=1}^l \lambda_k \frac{\partial h_k}{\partial x_i} = 0, (i = 1, 2, \dots, n) \\ h_k(\mathbf{x}) = 0, (k = 1, 2, \dots, l) \\ \mu_j g_j(\mathbf{x}) = 0, (j = 1, 2, \dots, m) \\ \mu_j \geq 0 \end{cases}$$

15.15.1 等式约束优化问题

等式约束优化问题是指

Problem 15.19 — 等式约束优化问题.

$$\begin{aligned} & \min f(x_1, x_2, \dots, x_n) \\ & \text{s.t. } h_k(x_1, x_2, \dots, x_n) = 0 \end{aligned}$$

我们令 $L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \sum_{k=1}^l \lambda_k h_k(\mathbf{x})$, 函数 $L(x, y)$ 称为 *Lagrange* 函数, 参数 λ 称为 *Lagrange* 乘子.

再联立方程组:

$$\begin{cases} \frac{\partial L}{\partial x_i} = 0 (i = 1, 2, \dots, n) \\ \frac{\partial L}{\partial \lambda_k} = 0 (k = 1, 2, \dots, l) \end{cases}$$

得到的解为可能极值点, 由于我们用的是必要条件, 具体是否为极值点需根据问题本身的具体情况检验. 这个方程组称为等式约束的极值必要条件.

上式我们对 n 个 x_i 和 l 个 λ_k 分别求偏导, 回想一下在无约束优化问题

$$f(x_1, x_2, \dots, x_n) = 0$$

中, 我们根据极值的必要条件, 分别令 $\frac{\partial f}{\partial x_i} = 0$, 求出可能的极值点.

因此可以联想到: 等式约束下的 Lagrange 乘数法引入了 l 个 Lagrange 乘子, 或许我们可以把 λ_k 也看作优化变量 (x_i 就叫做优化变量). 相当于将优化变量个数增加到 $(n+l)$ 个, x_i 与 λ_k 一视同仁, 均为优化变量, 均对它们求偏导.

15.15.2 不等式约束优化问题

以上我们讨论了等式约束的情形, 接下来我们来介绍不等式约束的优化问题.

我们先给出其主要思想: 转化的思想——将不等式约束条件变成等式约束条件. 具体做法是引入松弛变量. 松弛变量也是优化变量, 也需要一视同仁求偏导.

具体而言, 我们先看一个一元函数的例子:

■ **Example 15.2**

$$\begin{aligned} & \min f(x) \\ & \text{s.t. } g_1(x) = a - x \leq 0 \\ & \quad g_2(x) = x - b \leq 0 \end{aligned}$$



优化问题中, 我们必须求得一个确定的值, 因此不妨令所有的不等式均取到等号, 即 \leq 的情况.

对于约束 g_1 和 g_2 , 我们分别引入两个松弛变量 a_1^2 和 b_1^2 , 得到 $h_1(x, a_1) = g_1 + a_1^2 = 0$ 和 $h_2(x, b_1) = g_2 + b_1^2 = 0$.



注意, 这里直接加上平方项 a_1^2, b_1^2 而非 a_1, b_1 , 是因为 g_1 和 g_2 这两个不等式的左边必须加上一个正数才能使不等式变为等式. 若只加上 a_1 和 b_1 , 又会引入新的约束 $a_1 \geq 0$ 和 $b_1 \geq 0$, 这不符合我们的意愿.

$$\begin{aligned} g_1(x) = a - x \leq 0 &\Rightarrow h_1(x, a_1) = g_1(x) + a_1^2 = a - x + a_1^2 = 0 \\ g_2(x) = x - b \leq 0 &\Rightarrow h_2(x, b_1) = g_2(x) + b_1^2 = x - b + b_1^2 = 0 \end{aligned}$$

由此我们将不等式约束转化为了等式约束, 并得到 Lagrange 函数

$$L(x, a_1, b_1, \mu_1, \mu_2) = f(x) + \mu_1(a - x + a_1^2) + \mu_2(x - b + b_1^2)$$

我们再按照等式约束优化问题 (极值必要条件) 对其求解, 联立方程

$$\left\{ \begin{array}{l} \frac{\partial F}{\partial x} = \frac{\partial f}{\partial x} + \mu_1 \frac{dg_1}{dx} + \mu_2 \frac{dg_2}{dx} = \frac{df}{dx} - \mu_1 + \mu_2 = 0 \\ \frac{\partial F}{\partial \mu_1} = g_1 + a_1^2 = 0, \\ \frac{\partial F}{\partial \mu_2} = g_2 + b_1^2 = 0 \\ \frac{\partial F}{\partial a_1} = 2\mu_1 a_1 = 0, \\ \frac{\partial F}{\partial b_1} = 2\mu_2 b_1 = 0 \\ \mu_1 \geq 0, \quad \mu_2 \geq 0 \end{array} \right.$$



对于不等式约束前的乘子, 我们要求其大于等于 0. ($\mu_1 \geq 0, \mu_2 \geq 0$)

得出方程组后, 便开始动手解它. 看到第 3 行的两式 $\mu_1 a_1 = 0$ 和 $\mu_2 b_1 = 0$ 比较简单, 我们就从它们入手吧.

对于 $\mu_1 a_1 = 0$, 我们有两种情况:

情形 1: $\mu_1 = 0, a_1 \neq 0$

此时由于乘子 $\mu_1 = 0$, 因此 g_1 与其相乘为零, 可以理解为约束 g_1 不起作用, 且有 $g_1(x) = a - x < 0$.

情形 2: $\mu_1 \geq 0, a_1 = 0$

此时 $g_1(x) = a - x = 0$ 且 $\mu_1 > 0$, 可以理解为约束 g_1 起作用, 且有 $g_1(x) = 0$.

合并情形 1 和情形 2 得: $\mu_1 g_1 = 0$, 且在约束起作用时 $\mu_1 > 0, g_1(x) = 0$; 约束不起作用时 $\mu_1 = 0, g_1(x) < 0$.

同样地, 分析 $\mu_2 b_1 = 0$, 可得出约束 g_2 起作用和不起作用的情形, 并分析得到 $\mu_2 g_2 = 0$.

由此,

Theorem 15.15.2 — 一元一次优化式的 KKT 条件. 方程组 (极值必要条件) 转化为

$$\left\{ \begin{array}{l} \frac{df}{dx} + \mu_1 \frac{dg_1}{dx} + \mu_2 \frac{dg_2}{dx} = 0 \\ \mu_1 g_1(x) = 0, \mu_2 g_2(x) = 0 \\ \mu_1 \geq 0, \mu_2 \geq 0 \end{array} \right.$$

这是一元一次的情形. 类似地,

Corollary 15.15.3 对于多元多次不等式约束问题

$$\begin{aligned} & \min f(\mathbf{x}) \\ & \text{s.t. } g_j(\mathbf{x}) \leq 0 (j = 1, 2, \dots, m) \end{aligned}$$

有

$$\begin{cases} \frac{\partial f(x^*)}{\partial x_i} + \sum_{j=1}^m \mu_j \frac{\partial g_j(x^*)}{\partial x_i} = 0 (i = 1, 2, \dots, n) \\ \mu_j g_j(x^*) = 0 (j = 1, 2, \dots, m) \\ \mu_j \geq 0 (j = 1, 2, \dots, m) \end{cases}$$

上式便称为不等式约束优化问题的 KKT (Karush-Kuhn-Tucker) 条件. μ_j 称为 KKT 乘子, 且约束起作用时 $\mu_j \geq 0, g_j(x) = 0$; 约束不起作用时 $\mu_j = 0, g_j(x) < 0$.

15.15.3 KKT 乘子必须大于等于 0

Problem 15.20 还剩最后一个问题没有解决: 为什么 KKT 乘子必须大于等于零?

我将用几何性质来解释. 由于

$$\frac{\partial f(x^*)}{\partial x_i} + \sum_{j=1}^m \mu_j \frac{\partial g_j(x^*)}{\partial x_i} = 0 (i = 1, 2, \dots, n)$$

用梯度表示

$$\nabla f(\mathbf{x}^*) + \sum_{j \in J} \mu_j \nabla g_j(\mathbf{x}^*) = 0$$

J 为起作用约束的集合.

移项可得

$$-\nabla f(\mathbf{x}^*) = \sum_{j \in J} \mu_j \nabla g_j(\mathbf{x}^*)$$

注意到梯度为向量.

Theorem 15.15.4

$$-\nabla f(\mathbf{x}^*) = \sum_{j \in J} \mu_j \nabla g_j(\mathbf{x}^*)$$

在约束极小值点 \mathbf{x}^* 处, 函数 $f(\mathbf{x}^*)$ 的负梯度一定可以表示成: 所有起作用约束在该点的梯度 (等值线的法向量) 的线性组合.

Corollary 15.15.5 — 梯度的性质. 复习课本中梯度的性质: 某点梯度的方向就是函数等值线 $f(\mathbf{x}) = C$ 。(在这点的法线方向, 等值线就是地理的等高线。)

为方便作图, 假设现在只有两个约束条件起约束作用, 我们作出图形如图15-5.

注意我们上面推导过, 约束起作用时 $g_j(\mathbf{x}) = 0$, 所以此时约束在几何上应该是一簇约束平面.

我们假设在 \mathbf{x}^* 取得极小值点, 若同时满足 $g_1(\mathbf{x}) = 0$ 和 $g_2(\mathbf{x}) = 0$, 则 \mathbf{x}^k 一定在这两个平面的交线上, 且 $-\nabla f(\mathbf{x}^*) = \sum_{j \in J} \mu_j \nabla g_j(\mathbf{x}^*)$, 即 $-\nabla f(\mathbf{x}^k)$ 、 $\nabla g_1(\mathbf{x}^k)$ 和 $\nabla g_2(\mathbf{x}^k)$ 共面.

图15-6是在点 \mathbf{x}^k 处沿 x_1Ox_2 面的截面, 过点 \mathbf{x}^k 作目标函数的负梯度 $-\nabla f(\mathbf{x}^k)$, 它垂直于目标函数的等值线 $f(\mathbf{x}) = C$, 且指向目标函数 $f(\mathbf{x})$ 的最速减小方向.

Figure 15.5

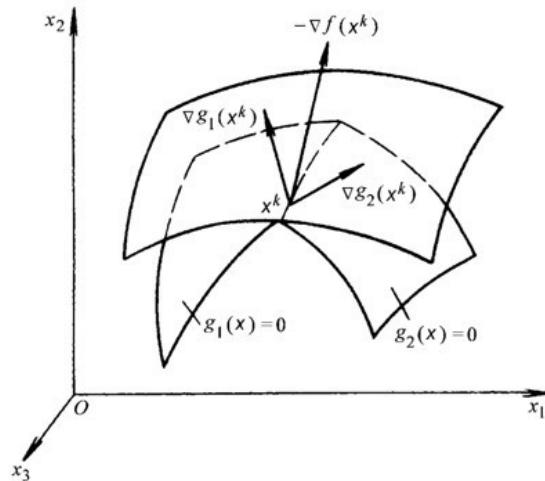
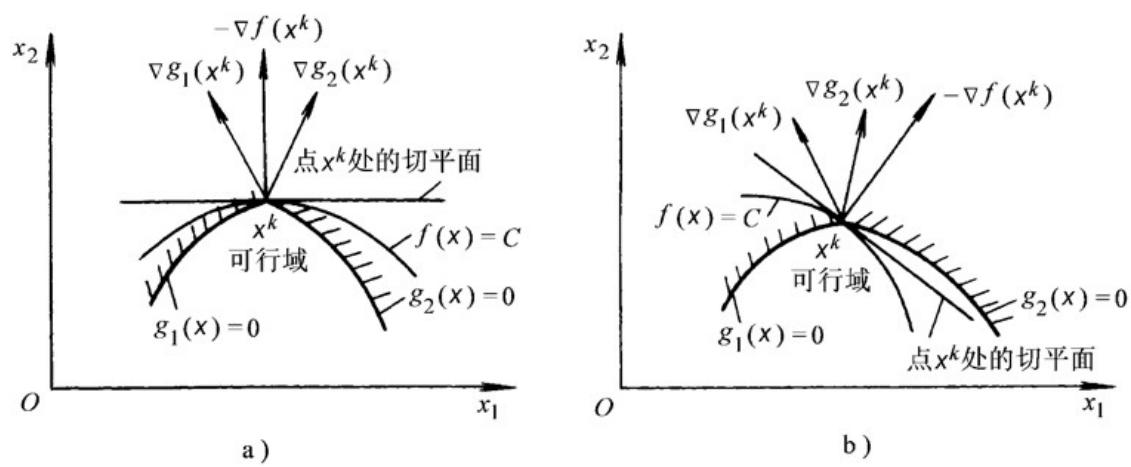


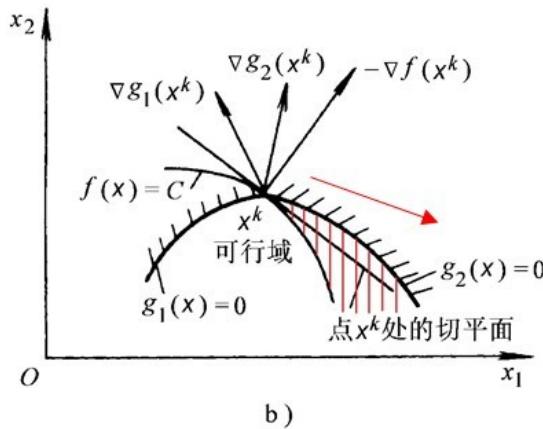
Figure 15.6



Corollary 15.15.6 一点的梯度与等值线相互垂直。

再作约束函数 $g_1(\mathbf{x}) = 0$ 和 $g_2(\mathbf{x}) = 0$ 的梯度 $\nabla g_1(\mathbf{x}^k)$ 和 $\nabla g_2(\mathbf{x}^k)$, 它们分别垂直 $g_1(\mathbf{x}) = 0$ 和 $g_2(\mathbf{x}) = 0$ 两曲面在 \mathbf{x}^k 的切平面, 并形成一个雉形夹角区域. 此时, 可能有 a、b 两种情形:

Figure 15.7



我们先来看情形 b: 若 3 个向量的位置关系如图 15.7 所示, 即 $-\nabla f$ 落在 ∇g_1 和 ∇g_2 所形成的锥角区外的一侧. 此时, 作等值面 $f(\mathbf{x}) = C$ 在点 \mathbf{x}^k 的切平面 (它与 $-\nabla f(\mathbf{x}^k)$ 垂直), 我们发现: 沿着与负梯度 $-\nabla f$ 成锐角的方向移动 (如下图红色箭头方向), 只要在红色区域取值, 目标函数 $f(\mathbf{x})$ 总能减小. 而红色区域是可行域 ($f(\mathbf{x}) = C$, C 取不同的常数能得到不同的等值线, 因此能取到红色区域), 因此既可减小目标函数值, 又不破坏约束条件. 这说明 \mathbf{x}^k 仍可沿约束曲面移动而不破坏约束条件, 且目标函数值还能够减小. 所以 \mathbf{x}^k 不是稳定的最优点, 即不是局部极值点.

反过来再看情形 a: $-\nabla f$ 落在 ∇g_1 和 ∇g_2 形成的锥角内. 此时, 同样作 $f(\mathbf{x}) = C$ 在点 \mathbf{x}^k 与 $-\nabla f$ 垂直的切平面. 当从 \mathbf{x}^k 出发沿着与负梯度 $-\nabla f$ 成锐角的方向移动时, 虽然能使目标函数值减小, 但此时任何一点都不在可行区域内. 显然, 此时 \mathbf{x}^k 就是局部最优点 \mathbf{x}^* , 再做任何移动都将破坏约束条件, 故它是稳定点.

由于 $-\nabla f(\mathbf{x}^*)$ 和 $\nabla g_1(\mathbf{x}^*)$ 、 $\nabla g_2(\mathbf{x}^*)$ 在一个平面内, 所以前者可看成是后两者的线性组合. 又由上面的几何分析知, $-\nabla f(\mathbf{x}^*)$ 在 $\nabla g_1(\mathbf{x}^*)$ 和 $\nabla g_2(\mathbf{x}^*)$ 的夹角之间, 所以线性组合的系数为正, 有

$$-\nabla f(\mathbf{x}^*) = \mu_1 \nabla g_1(\mathbf{x}^*) + \mu_2 \nabla g_2(\mathbf{x}^*), \text{ 且 } \mu_1 > 0, \mu_2 > 0.$$

这就是 $\mu_j > 0$ 的原因. 类似地, 当有多个不等式约束同时起作用时, 要求 $-\nabla f(\mathbf{x}^*)$ 处于 $\nabla g_j(\mathbf{x}^*)$ 形成的超角锥 (高维图形, 我姑且称之为“超”) 之内.

15.15.4 总结: 同时包含等式和不等式约束的一般优化问题

Theorem 15.15.7 — 同时包含等式和不等式约束的一般优化问题的 KKT 条件.

$$\begin{aligned} & \min f(\mathbf{x}) \\ & \text{s.t. } g_j(\mathbf{x}) \leq 0 (j = 1, 2, \dots, m) \\ & h_k(\mathbf{x}) = 0 (k = 1, 2, \dots, l) \end{aligned}$$

KKT 条件 (\mathbf{x}^* 是最优解的必要条件) 为

$$\begin{cases} \frac{\partial f}{\partial x_i} + \sum_{j=1}^m \mu_j \frac{\partial g_j}{\partial x_i} + \sum_{k=1}^l \lambda_k \frac{\partial h_k}{\partial x_i} = 0, (i = 1, 2, \dots, n) \\ h_k(\mathbf{x}) = 0, (k = 1, 2, \dots, l) \\ \mu_j g_j(\mathbf{x}) = 0, (j = 1, 2, \dots, m) \\ \mu_j \geq 0 \end{cases}$$

对于等式约束的 Lagrange 乘子，并没有非负的要求。

以后求其极值点，不必再引入松弛变量，直接使用 KKT 条件判断。

15.16 Supplement Material: 凸优化、拉格朗日乘子法和 KKT 条件³

15.16.1 凸集的概念

Definition 15.16.1 — 点、(直)线、线段。 $x_1 \neq x_2$ 是 \mathbb{R}^n 中的两点， $\theta \in \mathbb{R}$ ，那么 $y = x_2 + \theta(x_1 - x_2)$ 表示穿过两点的线。

当 $0 \leq \theta \leq 1$ 时， y 是 x_1 到 x_2 的线段， y 也可以表示成 $y = \theta x_1 + (1 - \theta)x_2$ 。

Definition 15.16.2 — 仿射集。一个集合 $C \subseteq \mathbb{R}^n$ 是仿射集如果其中任意两个不同的点的连线仍包含于 C 。

Definition 15.16.3 — 凸集。一个集合 $C \subseteq \mathbb{R}^n$ 是凸集，如果任意两点之间的线段仍包含于 C 。即 $\forall x_1, x_2 \in C$, 任意 $0 \leq \theta \leq 1$, 有 $\theta x_1 + (1 - \theta)x_2 \in C$ 。

Theorem 15.16.1 两个凸集的交集仍是凸集。

Definition 15.16.4 — 凸函数。一个函数 $f : \mathbb{R}^n \rightarrow \mathbb{R}$ 是凸函数如果定义域是凸集而且对任意定义域的 $x, y, 0 \leq \theta \leq 1$ 有

$$f(\theta x + (1 - \theta)y) \leq \theta f(x) + (1 - \theta)f(y)$$

二维时候的几何意义是，如果两点的线段总位于函数曲线之上，那么该函数是凸函数。

Theorem 15.16.2 f 是凸的，那么 $-f$ 是凹的。

Theorem 15.16.3 仿射函数既凸又凹。

Theorem 15.16.4 — α -sublevel 集。给定凸函数 f ，则 $\{x \in D(f) : f(x) \leq \alpha\}$ 是一个凸集。

Proof. 对任意 $x, y \in D(f)$ 使得

$$f(x) \leq \alpha, f(y) \leq \alpha$$

有

$$f(\theta x + (1 - \theta)y) \leq \theta f(x) + (1 - \theta)f(y) \leq \theta\alpha + (1 - \theta)\alpha$$

³Cited from <https://zhuanlan.zhihu.com/p/59928816>.

也就是说如果 x, y 属于 α - sublevel 集, 那么二者之间的线段上的点也可以使得 $f \leq \alpha$, 即二者之间的线段也包含于 α - sublevel 集。 ■

15.16.2 凸优化

Definition 15.16.5 — 优化问题. 优化问题有如下形式:

$$\begin{aligned} & \text{minimize} && f_0(x) \\ & \text{subject to} && f_i(x) \leq b_i, i = 1, \dots, m \end{aligned}$$

$x = (x_1, \dots, x_n)$ 是优化变量, 函数 $f_0 : \mathbb{R}^n \rightarrow \mathbb{R}$ 是目标函数, 函数 $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ 是约束函数。 b_i 是约束。使得 $f_0(x)$ 在约束条件下最小的 x^* 叫做最优点, 或问题的解。

Definition 15.16.6 — 凸优化. 凸优化问题有如下形式:

$$\begin{aligned} & \text{minimize} && f(x) \\ & \text{subject to} && x \in C \end{aligned}$$

f 是凸函数, C 是凸集。

Corollary 15.16.5 凸集可以表示成某些凸集的交集。

所以凸优化问题一般表示为

Definition 15.16.7 — 凸优化问题.

$$\begin{aligned} & \text{minimize} && f(x) \\ & \text{subject to} && g_i(x) \leq 0, i = 1, \dots, m \\ & && h_j(x) = 0, j = 1, \dots, p \end{aligned}$$

其中 g_i 是凸函数, h_j 是仿射函数。

$g_i(x) \leq 0$ 叫做 0 -sublevel 集, 是凸集。

Theorem 15.16.6 $h_j(x) = 0$ 也是凸集。

Proof.

$$h(\theta x_1 + (1 - \theta)x_2) = \theta h(x_1) + (1 - \theta)h(x_2) = 0$$

■

凸优化问题的特点是, 所有局部最优点都是全局最优点。

Definition 15.16.8 — 二次规划. 如果一个凸优化问题的 g_i 都是仿射函数, 且 f 是凸二次函数, 那么它叫做二次规划:

$$\begin{aligned} & \text{minimize} && \frac{1}{2}x^\top Px + c^\top x + d \\ & \text{subject to} && g_i(x) \leq 0, i = 1, \dots, m \\ & && h_j(x) = 0, j = 1, \dots, p \end{aligned}$$

其中 P 是一个对称半正定矩阵(使得 $\frac{1}{2}x^\top Px \geq 0$)。

15.16.3 拉格朗日对偶性

Theorem 15.16.7 对于没有限制的凸函数, 最优点 x^* 一定满足 $\nabla_x f(x^*) = 0$ 。

然而对于有限制条件的凸优化问题却不是这样。拉格朗日对偶性可以将有限制的凸优化问题转化为没有限制的问题，来求解凸优化问题。

Definition 15.16.9 — 拉格朗日函数. 给定一个凸优化问题，拉格朗日算子是一个函数 $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \rightarrow \mathbb{R}$, 定义为：

$$\mathcal{L}(x, \alpha, \beta) = f(x) + \sum_{i=1}^m \alpha_i g_i(x) + \sum_{i=1}^p \beta_i h_i(x)$$

$x \in \mathbb{R}^n$ 叫做主变量 (*primal variable*)。 $\alpha \in \mathbb{R}^m, \beta \in \mathbb{R}^p$ 统称对偶变量或拉格朗日乘子。

Theorem 15.16.8 总存在一个拉格朗日乘子，使得没有限制的拉格朗日算子相对于 x 的最小值，等于原凸优化问题的最优值

后文会证明。

主问题

为了说明拉格朗日算子和原凸优化问题的关系，需要引入主问题和对偶问题。

考虑优化问题：

Problem 15.21

$$\min_x \left[\max_{\alpha, \beta, \alpha_i \geq 0} \mathcal{L}(x, \alpha, \beta) \right] = \min_x \theta_{\mathcal{P}}(x)$$

括号内的 $\theta_{\mathcal{P}} : \mathbb{R}^n \rightarrow \mathbb{R}$ 叫做主目标 (*primal objective*)，等号右边的没有限制的最小化问题叫做主问题 (*primal problem*)。用 x^* 表示主问题的解， $p^* = \theta_{\mathcal{P}}(x^*)$ 表示主目标的最优值。 ■

Definition 15.16.10 — primal feasible. x 是主可行的 (*primal feasible*) 如果 $g_i(x) \leq 0, h_j(x) = 0$ 。

$$\begin{aligned} \theta_{\mathcal{P}}(x) &= \max_{\alpha, \beta, \alpha_i \geq 0} \mathcal{L}(x, \alpha, \beta) \\ &= \max_{\alpha, \beta, \alpha_i \geq 0} \left[f(x) + \sum_{i=1}^m \alpha_i g_i(x) + \sum_{i=1}^p \beta_i h_i(x) \right] \\ &= f(x) + \max_{\alpha, \beta, \alpha_i \geq 0} \left[\sum_{i=1}^m \alpha_i g_i(x) + \sum_{i=1}^p \beta_i h_i(x) \right] \end{aligned}$$

观察最后一个式子：

- 如果任何一个 $g_i(x) > 0$ ，那么最大化 $\theta_{\mathcal{P}}(x)$ 只需要将相应的 α_i 设置为无限大。
- 如果 $g_i(x) \leq 0$ ，因为 $\alpha_i \geq 0$ ，所以 $\theta_{\mathcal{P}}(x)$ 最大时，必然有 $\alpha_i = 0$ 。

相似地：

- 如果 $h_i \neq 0$ ，那么最大化 $\theta_{\mathcal{P}}(x)$ 只需要将相应的 β_i 设置为 $h_i(x)$ 的相同符号且绝对值无限大。
- 如果 $h_i(x) = 0$ ，那么 $\sum_{i=1}^p \beta_i h_i(x)$ 项的最大值为 0。

综上，有

$$\theta_{\mathcal{P}}(x) = \begin{cases} f(x) + 0 & x \text{ 为主可行的} \\ f(x) + \infty & x \text{ 不是主可行的} \end{cases}$$

因此当 x 主可行时，主问题15.21的最优值等于原凸优化问题15.16.7的最优值。

对偶问题

Definition 15.16.11 — 对偶问题. 对调主问题的最大最小操作:

$$\max_{\alpha, \beta, \alpha \geq 0} \left[\min_x \mathcal{L}(x, \alpha, \beta) \right] = \max_{\alpha, \beta, \alpha \geq 0} \theta_{\mathcal{D}}(\alpha, \beta)$$

$\theta_{\mathcal{D}}(\alpha, \beta) : \mathbb{R}^m \times \mathbb{R}^p \rightarrow \mathbb{R}$ 是对偶目标 (dual objective), 等号右边的有限制的最大化问题叫做对偶问题。用 (α^*, β^*) 表示对偶问题的解, $d^* = \theta_{\mathcal{D}}(\alpha^*, \beta^*)$ 表示对偶目标的最优值。

Definition 15.16.12 — 对偶可行. 如果 $\alpha_i(x) \geq 0$, (α, β) 是对偶可行的。

Theorem 15.16.9 如果 (α, β) 是对偶可行的, 那么 $\theta_{\mathcal{D}}(\alpha, \beta) \leq p^*$.

Proof.

$$\begin{aligned} \theta_{\mathcal{D}}(\alpha, \beta) &= \min_x \mathcal{L}(x, \alpha, \beta) \\ &\leq \mathcal{L}(x^*, \alpha, \beta) \\ &= f(x^*) + \sum_{i=1}^m \alpha_i g_i(x^*) + \sum_{i=1}^p \beta_i h_i(x^*) \\ &\leq f(x^*) = p^* \end{aligned}$$

■

Theorem 15.16.10 — 弱对偶性. 对任意主问题和对偶问题, 有 $d^* \leq p^*$.

Theorem 15.16.11 — 强对偶性. 对任意主问题和对偶问题, 如果满足某个条件 (constraint qualifications), 那么 $d^* = p^*$ 。最常用的 constraint qualification 是 Slater's condition: 所有的不等式限制都严格满足 (即 $g_i(x) < 0$)。

Theorem 15.16.12 — Slater 条件. 设定义在 \mathcal{D} 上的函数 $f_i(\cdot), i = 1, 2, \dots, n$ 为凸函数, $g_j(\cdot), j = 1, 2, \dots, m$ 为仿射函数, 考虑凸优化问题

$$\min_{\mathbf{x}} f_0(\mathbf{x}), \quad \text{s.t. } f_i(\mathbf{x}) \leq 0, g_i(\mathbf{x}) \leq 0$$

如果存在点 $\mathbf{x} \in \text{relint } \mathcal{D}$ (即 \mathcal{D} 的相对内点), 则强对偶性成立。

实践中, 几乎所有的凸优化问题都满足某种 constraint qualification, 所以主问题和对偶问题有相同的最优值。

Theorem 15.16.13 — 互补松弛性 (complementary slackness, KKT complementarity). 如果强对偶性满足, 那么 $\alpha_i^* g_i(x_i^*) = 0, i = 1, \dots, m$

Proof.

$$\begin{aligned}
 p^* = d^* &= \theta_{\mathcal{D}}(\alpha^*, \beta^*) = \min_x \mathcal{L}(x, \alpha^*, \beta^*) \\
 &\leq \mathcal{L}(x^*, \alpha^*, \beta^*) \\
 &= f(x^*) + \sum_{i=1}^m \alpha_i^* g_i(x^*) + \sum_{i=1}^p \beta_i^* h_i(x^*) \\
 &\leq f(x^*) = p^*
 \end{aligned}$$

因为第一个和最后一个表达式相等, 所以中间所有的小于等于号都是等号, 有

$$\sum_{i=1}^m \alpha_i^* g_i(x^*) + \sum_{i=1}^p \beta_i^* h_i(x^*) = 0$$

因为 $\alpha_i^* \geq 0$, $h_i(x^*) = 0$, 所以 α_i^* 和 $g_i(x^*)$ 至少有一个是 0。 ■

Theorem 15.16.14 设 $x^* \in \mathbb{R}^n$, $\alpha^* \in \mathbb{R}^m$, $\beta^* \in \mathbb{R}^p$, 下列条件为 KKT 条件:

1. (主可行) $g_i(x^*) \leq 0, i = 1, \dots, m, h_j(x^*) = 0, j = 1, \dots, p$
2. (对偶可行) $\alpha_i^* \geq 0, i = 1, \dots, m$
3. (互补松弛性) $\alpha_i^* g_i(x^*) = 0, i = 1, \dots, m$
4. (Lagrangian Stationary) $\nabla_x \mathcal{L}(x^*, \alpha^*, \beta^*) = 0$

Theorem 15.16.15 对于凸优化问题, 有:

x^* 原始最优, (α^*, β^*) 对偶最优, 且有

强对偶性 \Leftrightarrow 满足KKT条件

15.16.4 总结

给定一个凸优化问题, 拉格朗日算子将凸优化问题的目标函数和限制考虑进一个函数中, 在拉格朗日算子基础上可以定义主问题和对偶问题。

主问题是先调整 (α, β) 使拉格朗日算子最大(变为主目标), 再调整 x 使主目标最小。 x 主可行时, 主目标等于原凸优化问题的目标函数, 主目标的最小值等于原凸优化问题的最小值。

对偶问题是先调整 x 使拉格朗日算子最小(变为对偶目标, 因为拉格朗日算子是关于 x 的没有限制的凸函数, 所以变为对偶目标时其对 x 的偏导数为 0), 再调整使对偶目标最大。 (α, β) 对偶可行时, 对偶目标小于等于主目标的最小值。

弱对偶性指对偶目标的最大值小于等于主目标的最小值。强对偶性指对偶目标的最大值等于主目标的最小值。

对偶目标的最大值 = 主目标的最小值 = 原凸优化问题的最小值 \Leftrightarrow 满足KKT条件

因此, 当对偶问题比原凸优化问题容易求解时, 可以通过求解对偶问题来解原凸优化问题。

15.17 Supplementary Material: KKT Conditions, Linear Programming and Nonlinear Programming⁴

15.17.1 Karush-Kuhn-Tucker Theorem(s)

⁴Written by Christopher Griffine. (Linear Programming)

Theorem 15.17.1 Let $z : \mathbb{R}^n \rightarrow \mathbb{R}$ be a differentiable objective function, $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ be differentiable constraint functions for $i = 1, \dots, m$ and $h_j : \mathbb{R}^n \rightarrow \mathbb{R}$ be differentiable constraint functions for $j = 1, \dots, l$. If $\mathbf{x}^* \in \mathbb{R}^n$ is an optimal point satisfying an appropriate regularity condition for the following optimization problem:

$$P \left\{ \begin{array}{l} \max z(x_1, \dots, x_n) \\ \text{s.t. } g_1(x_1, \dots, x_n) \leq 0 \\ \quad \vdots \\ \quad g_m(x_1, \dots, x_n) \leq 0 \\ \quad h_1(x_1, \dots, x_n) = 0 \\ \quad \vdots \\ \quad h_l(x_1, \dots, x_n) = 0 \end{array} \right.$$

then there exists $\lambda_1, \dots, \lambda_m \in \mathbb{R}$ and $\mu_1, \dots, \mu_l \in \mathbb{R}$ so that:

$$\begin{aligned} \text{Primal Feasibility} &: \begin{cases} g_i(\mathbf{x}^*) \leq 0 & \text{for } i = 1, \dots, m \\ h_j(\mathbf{x}^*) = 0 & \text{for } j = 1, \dots, l \end{cases} \\ \text{Dual Feasibility} &: \begin{cases} \nabla z(\mathbf{x}^*) - \sum_{i=1}^m \lambda_i \nabla g_i(\mathbf{x}^*) - \sum_{j=1}^l \mu_j \nabla h_j(\mathbf{x}^*) = \mathbf{0} \\ \lambda_i \geq 0 \quad \text{for } i = 1, \dots, m \\ \mu_j \in \mathbb{R} \quad \text{for } j = 1, \dots, l \end{cases} \\ \text{Complementary Slackness} &: \{\lambda_i g_i(\mathbf{x}^*) = 0 \quad \text{for } i = 1, \dots, m\} \end{aligned}$$

Theorem 15.17.2 Let $z : \mathbb{R}^n \rightarrow \mathbb{R}$ be a differentiable concave function, $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ be differentiable convex functions for $i = 1, \dots, m$ and $h_j : \mathbb{R}^n \rightarrow \mathbb{R}$ be affine functions for $j = 1, \dots, l$. Suppose there are $\lambda_1, \dots, \lambda_m \in \mathbb{R}$ and $\mu_1, \dots, \mu_l \in \mathbb{R}$ so that:

$$\begin{aligned} \text{Primal Feasibility} &: \begin{cases} g_i(\mathbf{x}^*) \leq 0 & \text{for } i = 1, \dots, m \\ h_j(\mathbf{x}^*) = 0 & \text{for } j = 1, \dots, l \end{cases} \\ \text{Dual Feasibility} &: \begin{cases} \nabla z(\mathbf{x}^*) - \sum_{i=1}^m \lambda_i \nabla g_i(\mathbf{x}^*) - \sum_{j=1}^l \mu_j \nabla h_j(\mathbf{x}^*) = \mathbf{0} \\ \lambda_i \geq 0 \quad \text{for } i = 1, \dots, m \\ \mu_j \in \mathbb{R} \quad \text{for } j = 1, \dots, l \end{cases} \\ \text{Complementary Slackness} &: \{\lambda_i g_i(\mathbf{x}^*) = 0 \quad \text{for } i = 1, \dots, m\} \end{aligned}$$

then \mathbf{x}^* is a global maximizer for

$$P \left\{ \begin{array}{l} \max z(x_1, \dots, x_n) \\ \text{s.t. } g_1(x_1, \dots, x_n) \leq 0 \\ \quad \vdots \\ g_m(x_1, \dots, x_n) \leq 0 \\ h_1(x_1, \dots, x_n) = 0 \\ \quad \vdots \\ h_l(x_1, \dots, x_n) = 0 \end{array} \right.$$

The values $\lambda_1, \dots, \lambda_m$ and μ_1, \dots, μ_l are sometimes called *Lagrange multipliers* and sometimes called *dual variables*. Primal Feasibility, Dual Feasibility and Complementary Slackness are called the *Karush-Kuhn-Tucker* (KKT) conditions.



The regularity condition mentioned in Theorem 15.17.1 is sometimes called a *constraint qualification*. A common one is that the gradients of the binding constraints are all linearly independent at \mathbf{x}^* . There are many variations of constraint qualifications. We will not deal with these in these notes.

Suffice it to say, all the problems we consider will automatically satisfy a constraint qualification, meaning the KKT theorem holds.



Theorem 15.17.1 holds as a necessary condition even if $z(\mathbf{x})$ is not concave or the functions $g_i(\mathbf{x})$ ($i = 1, \dots, m$) are not convex or the functions $h_j(\mathbf{x})$ ($j = 1, \dots, l$) are not linear. In this case though, the fact that a triple: $(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^l$ does not ensure that this is an optimal solution for Problem P .

Looking more closely at the dual feasibility conditions, we see something interesting. Suppose that there are *no* equality constraints (i.e., not constraints of the form $h_j(\mathbf{x}) = 0$). Then the statements:

$$\nabla z(\mathbf{x}^*) - \sum_{i=1}^m \lambda_i \nabla g_i(\mathbf{x}^*) - \sum_{j=1}^l \mu_j \nabla h_j(\mathbf{x}^*) = \mathbf{0}$$

$$\lambda_i \geq 0 \quad \text{for } i = 1, \dots, m$$

imply that:

$$\nabla z(\mathbf{x}^*) = \sum_{i=1}^m \lambda_i \nabla g_i(\mathbf{x}^*)$$

$$\lambda_i \geq 0 \quad \text{for } i = 1, \dots, m$$

Specifically, this says that the *gradient of z at \mathbf{x}^* is a positive combination of the gradients of the constraints at \mathbf{x}^** . But more importantly, since we also have *complementary slackness*, we know that if $\mathbf{g}_i(\mathbf{x}^*) \neq \mathbf{0}$, then $\lambda_i = 0$ because $\lambda_i g_i(\mathbf{x}^*) = 0$ for $i = 1, \dots, m$. Thus, what dual feasibility is really saying is that *gradient of z at \mathbf{x}^* is a positive combination of the gradients of the **binding** constraints at \mathbf{x}^** . Remember, a constraint is binding if $g_i(\mathbf{x}^*) = 0$, in which case $\lambda_i \geq 0$.



Continuing from the previous remark, in the general case when we have some equality constraints, then dual feasibility says:

$$\begin{aligned}\nabla z(\mathbf{x}^*) &= \sum_{i=1}^m \lambda_i \nabla g_i(\mathbf{x}^*) + \sum_{j=1}^l \mu_j \nabla h_j(\mathbf{x}^*) \\ \lambda_i &\geq 0 \quad \text{for } i = 1, \dots, m \\ \mu_j &\in \mathbb{R} \quad \text{for } j = 1, \dots, l\end{aligned}$$

Since equality constraints are *always binding* this says that the *gradient of z at \mathbf{x}^* is a linear combination of the gradients of the **binding** constraints at \mathbf{x}^** .

15.17.2 Linear Programming and KKT Conditions - An Example

Consider the following linear programming problem:

$$P \left\{ \begin{array}{l} \max x_1 + x_2 \\ \text{s.t. } x_1 + 2x_2 \leq 4 \\ \quad 2x_1 + x_2 \leq 6 \\ \quad x_1, x_2 \geq 0 \end{array} \right. \quad (15.2)$$

Assuming I didn't know how to solve this problem using the Simplex method, I could look at Theorem 15.17.2 and remark that the constraints and objective are all concave (and convex) since they are linear. Therefore, it suffices to find a KKT point and this point must be optimal. We can re-write Problem P in the form of Theorem 15.17.1 as:

$$P \left\{ \begin{array}{l} \max z(x_1, x_2) \equiv x_1 + x_2 \\ \text{s.t. } g_1(x_1, x_2) \equiv x_1 + 2x_2 - 4 \leq 0 \\ \quad g_2(x_1, x_2) \equiv 2x_1 + x_2 - 6 \leq 0 \\ \quad g_3(x_1, x_2) \equiv -x_1 \leq 0 \\ \quad g_4(x_1, x_2) \equiv -x_2 \leq 0 \end{array} \right.$$

To write the KKT conditions, observe the following:

$$\nabla z = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \nabla g_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad \nabla g_2 = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad \nabla g_3 = \begin{bmatrix} -1 \\ 0 \end{bmatrix} \quad \nabla g_4 = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

We can now write the KKT conditions for this problem as:

$$\text{Primal Feasibility : } \begin{cases} x_1 + 2x_2 \leq 4 \\ 2x_1 + x_2 \leq 6 \\ x_1, x_2 \geq 0 \end{cases}$$

$$\text{Dual Feasibility : } \begin{cases} \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \lambda_1 \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \lambda_2 \begin{bmatrix} 2 \\ 1 \end{bmatrix} - \lambda_3 \begin{bmatrix} -1 \\ 0 \end{bmatrix} - \lambda_4 \begin{bmatrix} 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ \lambda_1, \lambda_2, \lambda_3, \lambda_4 \geq 0 \end{cases}$$

$$\text{Complementary Slackness : } \begin{cases} \lambda_1(x_1 + 2x_2 - 4) = 0 \\ \lambda_2(2x_1 + x_2 - 6) = 0 \\ \lambda_3(-x_1) = 0 \\ \lambda_4(-x_2) = 0 \end{cases}$$

Consider Dual Feasibility for a moment. I can expand the matrices to obtain a system of equations:

$$1 - \lambda_1 - 2\lambda_2 + \lambda_3 = 0$$

$$1 - 2\lambda_1 - \lambda_2 + \lambda_4 = 0$$

$$\lambda_1, \lambda_2, \lambda_3, \lambda_4 \geq 0$$

Or:

$$\lambda_1 + 2\lambda_2 - \lambda_3 = 1$$

$$2\lambda_1 + \lambda_2 - \lambda_4 = 1$$

$$\lambda_1, \lambda_2, \lambda_3, \lambda_4 \geq 0$$

Since $\lambda_3, \lambda_4 \geq 0$, they act like *surplus variables* and we can write the Dual Feasibility as:

$$\begin{cases} \lambda_1 + 2\lambda_2 \geq 1 \\ 2\lambda_1 + \lambda_2 \geq 1 \\ \lambda_1, \lambda_2 \geq 0 \end{cases} \quad (15.3)$$



It would now suffice to find values for $x_1, x_2, \lambda_1, \lambda_2, \lambda_3$, and λ_4 that satisfy the KKT conditions and we could solve the linear programming problem P .

We can show that the optimal point for this problem is $x = \frac{8}{3}$ and $y = \frac{2}{3}$ using a graphical method. Figure 15.8 shows the feasible region of the problem as well as the level curves of the objective function (curves for which $x_1 + x_2 = k$, where k is a fixed constant). The value k increases going left to right and bottom to top, so the optimal solution must occur at the intersection of the lines $x_1 + 2x_2 = 4$ and $2x_1 + x_2 = 6$. You can compare the value of the objective at the proposed optimal point $x = \frac{8}{3}$ and $y = \frac{2}{3}$ to the value of the objective at the other extreme points. For example, at the extreme point $x = 3, y = 0$, we see the value of the objective is only 3, as compared to $\frac{10}{3}$ at the point of optimality.

Figure 15.8: The feasible region for the example linear programming problem illustrating level curves of the objective.

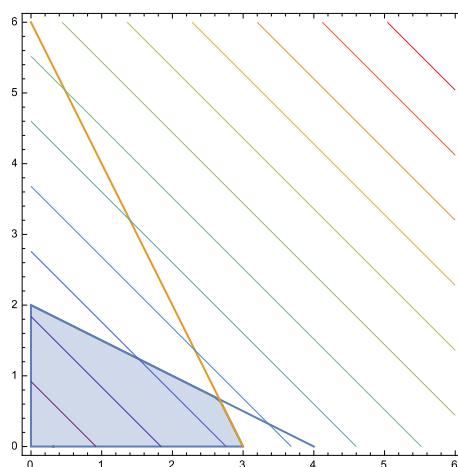
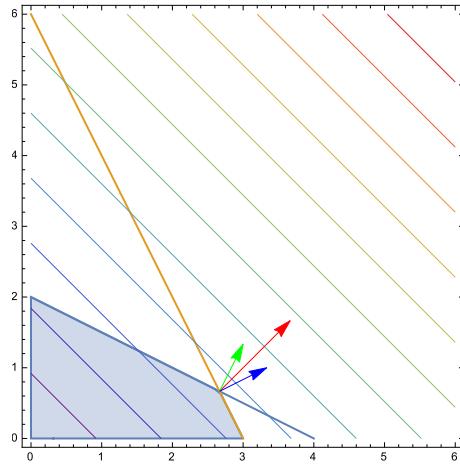


Figure 15.9: The gradients of the objective function and the binding constraints at optimality and their geometric relation are illustrated



We can now see that for the KKT conditions to hold, we must have $\lambda_3 = \lambda_4 = 0$ because $x_1 > 0$ and $x_2 > 0$ at optimality and complementary slackness requires that: $\lambda_3(-x_1) = 0$ and $\lambda_4(-x_2) = 0$ at optimality. This leaves λ_1 and λ_2 . We know these must satisfy the equations in Expression 15.3. Therefore, we see that when $\lambda_1 = \lambda_2 = \frac{1}{3}$, Expression 2 is satisfied. Thus: we have written:

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \frac{1}{3} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad (15.4)$$

That is, we have expressed the gradient of the objective function (∇z) as a positive combination of the gradients of the *binding* constraints (∇g_1 and ∇g_2). This is shown in Figure 15.9, in which we see the gradient of the objective function (red) inside the (cone of) the gradients of the binding constraints (blue and green).

15.17.3 The Dual KKT Conditions

Recall that if we are given a linear programming problem of the form:

$$P \begin{cases} \max \mathbf{c}^T \mathbf{x} \\ s.t. \mathbf{Ax} \leq \mathbf{b} \\ \mathbf{x} \geq 0 \end{cases} \quad (15.5)$$

Then the dual problem for Problem P is:

$$D \begin{cases} \min \mathbf{w}^T \mathbf{b} \\ s.t. \mathbf{w}^T \mathbf{A} \geq \mathbf{c}^T \\ \mathbf{w} \geq \mathbf{0} \end{cases} \quad (15.6)$$

Here we assume that \mathbf{w} is a *row* vector (unlike our usual assumption that all vectors are column vectors). In our example problem, we have:

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \quad \mathbf{c} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} 4 \\ 6 \end{bmatrix}$$

Let us write:

$$\mathbf{w} = [\lambda_1 \ \lambda_2]$$

Then we can write the dual problem as:

$$D \begin{cases} \min & [\lambda_1 \ \lambda_2] \begin{bmatrix} 4 \\ 6 \end{bmatrix} \\ & [\lambda_1 \ \lambda_2] \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \geq [1 \ 1] \\ & \lambda_1, \lambda_2 \geq 0 \end{cases}$$

This can be rewritten more concretely as:

$$D \begin{cases} \min & 4\lambda_1 + 6\lambda_2 \\ & \lambda_1 + 2\lambda_2 \geq 1 \\ & 2\lambda_1 + \lambda_2 \geq 1 \\ & \lambda_1, \lambda_2 \geq 0 \end{cases}$$

Immediately we see a connection. The constraints of this problem look just like the simplified dual feasibility constraints. In fact, if we add surplus variables $\lambda_3, \lambda_4 \geq 0$, we see they match exactly and the problem in standard form is:

$$D \begin{cases} \min & 4\lambda_1 + 6\lambda_2 \\ & \lambda_1 + 2\lambda_2 - \lambda_3 = 1 \\ & 2\lambda_1 + \lambda_2 - \lambda_4 = 1 \\ & \lambda_1, \lambda_2, \lambda_3, \lambda_4 \geq 0 \end{cases}$$

We can complete our exploration of the relationship between these two problems by constructing the KKT conditions for the dual problem. First, re-write the dual as:

$$\begin{aligned} \max & -4\lambda_1 - 6\lambda_2 \\ \text{s.t.} & -\lambda_1 - 2\lambda_2 + 1 \leq 0 \\ & -2\lambda_1 - \lambda_2 + 1 \leq 0 \\ & -\lambda_1 \leq 0 \\ & -\lambda_2 \leq 0 \end{aligned}$$

As before, we can write down the KKT conditions:

$$\text{Primal Feasibility : } \begin{cases} \lambda_1 + 2\lambda_2 \geq 1 \\ 2\lambda_1 + \lambda_2 \geq 1 \\ \lambda_1, \lambda_2 \geq 1 \end{cases}$$

$$\text{Dual Feasibility : } \begin{cases} \begin{bmatrix} -4 \\ -6 \end{bmatrix} - x_1 \begin{bmatrix} -1 \\ -2 \end{bmatrix} - x_2 \begin{bmatrix} -2 \\ -1 \end{bmatrix} - s_1 \begin{bmatrix} -1 \\ 0 \end{bmatrix} - s_2 \begin{bmatrix} 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ x_1, x_2, s_1, s_2 \geq 0 \end{cases}$$

$$\text{Complementary Slackness : } \begin{cases} x_1(-\lambda_1 - 2\lambda_2 + 1) = 0 \\ x_2(-2\lambda_1 - \lambda_2 + 1) = 0 \\ s_1(-\lambda_1) = 0 \\ s_2(-\lambda_2) = 0 \end{cases}$$

Again, we can re-write the dual feasibility conditions (for which we have intentionally chosen specific dual variable names) and see they become:

$$x_1 + 2x_2 + s_1 = 4$$

$$2x_1 + x_2 + s_2 = 6$$

$$x_1, x_2, s_1, s_2 \geq 0$$

Since $s_1, s_2 \geq 0$, they act like *slack variables* and thus we have:

$$x_1 + 2x_2 \leq 4$$

$$2x_1 + x_2 \leq 6$$

$$x_1, x_2 \geq 0$$

and we've recovered the constraints from the original primal problem as the dual feasibility conditions of the KKT conditions for the dual optimization problem. We can go further. Note:

$$-s_1 = x_1 + 2x_2 - 4 = -g_1(x_1, x_2)$$

$$-s_2 = 2x_1 + x_2 - 6 = -g_2(x_1, x_2)$$

while from our earlier observations about the surplus variables λ_3 and λ_4 we know:

$$\lambda_3 = -\lambda_1 - 2\lambda_2 + 1$$

$$-\lambda_4 = -2\lambda_1 - \lambda_2 + 1$$

Thus, complementary slackness from the primal problem is:

$$\lambda_3(-x_1) = 0 \iff \lambda_3(x_1) = 0 \iff x_1(\lambda_1 + 2\lambda_2 - 1) = 0$$

$$\lambda_4(-x_2) = 0 \iff \lambda_4(x_2) = 0 \iff x_2(-2\lambda_1 - \lambda_2 + 1) = 0$$

Likewise:

$$\lambda_1(x_1 + 2x_2 - 4) = 0 \iff \lambda_1(-s_1) = 0 \iff s_1(-\lambda_1) = 0$$

$$\lambda_2(2x_1 + x_2 - 6) = 0 \iff \lambda_2(-s_2) = 0 \iff s_2(-\lambda_2) = 0$$

Thus, the complementary slackness conditions of the primal problem are identical to the complementary slackness conditions of the dual problem. This fact is true for linear programming problems in general.



For an arbitrary linear programming problem and its dual, the KKT conditions for the primal and dual problems are **equivalent**, but the dual feasibility conditions for the primal problem are identical to the primal feasibility conditions for the dual problem and vice-versa. *Thus, two linear programming problems are dual to each other if and only if they share KKT conditions with the primal and dual feasibility conditions swapped.*



We will use these results on KKT conditions and linear programs to discuss the solution (i.e., Nash equilibria) of zero-sum games via linear programming. We will then generalize these results to derive a quadratic programming problem whose optimal solutions will yield Nash equilibria.

15.17.4 An Example from Nonlinear Programming

Let's recall a simple optimization problem from differential calculus (Math 140): Goats are an environmentally friendly and inexpensive way to control a lawn when there are lots of rocks or lots of hills. (Seriously, both Google and some U.S. Navy bases use goats on rocky hills instead of paying lawn mowers!)

Suppose I wish to build a pen to keep some goats. I have 100 meters of fencing and I wish to build the pen in a rectangle with the largest possible area. How long should the sides of the rectangle be? In this case, making the pen *better* means making it have the largest possible area.

The problem is illustrated in Figure 15.10. Clearly, we know that:

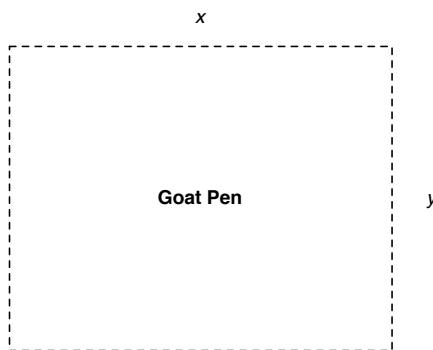


Figure 15.10: Goat pen with unknown side lengths. The objective is to identify the values of x and y that maximize the area of the pen (and thus the number of goats that can be kept).

$$2x + 2y = 100 \quad (15.7)$$

because $2x + 2y$ is the perimeter of the pen and I have 100 meters of fencing to build my pen. The area of the pen is $A(x, y) = xy$. We can use Equation 15.7 to solve for x in terms of y . Thus we have:

$$y = 50 - x \quad (15.8)$$

and $A(x) = x(50 - x)$. To maximize $A(x)$, recall we take the first derivative of $A(x)$ with respect to x , set this derivative to zero and solve for x :

$$\frac{dA}{dx} = 50 - 2x = 0; \quad (15.9)$$

Thus, $x = 25$ and $y = 50 - x = 25$. We further recall from basic calculus how to confirm that this is a maximum; note:

$$\left. \frac{d^2A}{dx^2} \right|_{x=25} = -2 < 0 \quad (15.10)$$

Which implies that $x = 25$ is a *local maximum* for this function. Another way of seeing this is to note that $A(x) = 50x - x^2$ is a concave (a frowning parabola). As we could have guessed, a square will maximize the area available for holding goats.

We can re-write the problem to be in a more common form

$$\begin{cases} \max A(x, y) = xy \\ s.t. 2x + 2y = 100 \equiv 2x + 2y - 100 = 0 \\ x \geq 0 \equiv -x \leq 0 \\ y \geq 0 \equiv -y \leq 0 \end{cases} \quad (15.11)$$

Note we've added two inequality constraints $x \geq 0$ and $y \geq 0$ because it doesn't really make any sense to have negative lengths. In our problem, we now have: $g_1(x, y) = -x$ and $g_2(x, y) = -y$ and $h(x, y) = 2x + 2y - 100$ to be consistent with the notation in Theorem 15.17.1. It is worth noting, we could have used the constraint $2x+2y-100 \leq 0$ instead and obtained the same answer.

For the point of optimality $(x = 25, y = 25)$, let us now compute the KKT conditions. Note first:

$$\nabla A = \begin{bmatrix} y \\ x \end{bmatrix} \quad \nabla h = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \quad \nabla g_1 = \begin{bmatrix} -1 \\ 0 \end{bmatrix} \quad \nabla g_2 = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

Substituting $x^* = y^* = 25$, we can see that primal feasibility is obviously satisfied. Moreover:

$$\nabla A(x^*, y^*) = \begin{bmatrix} 25 \\ 25 \end{bmatrix}$$

We consider only dual feasibility and complimentary slackness.

$$\text{Dual Feasibility : } \begin{cases} \begin{bmatrix} 25 \\ 25 \end{bmatrix} - \lambda_1 \begin{bmatrix} -1 \\ 0 \end{bmatrix} - \lambda_2 \begin{bmatrix} 0 \\ -1 \end{bmatrix} - \mu \begin{bmatrix} 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ \lambda_1, \lambda_2 \geq 0 \\ \mu \in \mathbb{R} \end{cases}$$

$$\text{Complementary Slackness : } \begin{cases} \lambda_1(-x^*) = \lambda_1 \cdot (-25) = 0 \\ \lambda_2(-y^*) = \lambda_2 \cdot (-25) = 0 \end{cases}$$

Notice we substituted x^* and y^* into the complementary slackness equations. From complementary slackness, we see at once that $\lambda_1 = \lambda_2 = 0$. This means dual feasibility reduces to:

$$\begin{bmatrix} 25 \\ 25 \end{bmatrix} - \mu \begin{bmatrix} 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The solution: $\mu = 25/2$ satisfies this equality. Notice the gradient of the objective and the gradient of the (single) binding constraint are parallel. Notice also unlike a linear programming problem, the optimal solution to this problem *does not* occur at the extreme point of the constraint set (which is really just the line segment $2x + 2y = 100$ with $x, y \geq 0$). This geometric interpretation is illustrated in Figure 15.11. The gradient of the objective is shown in red, the gradient of the binding constraint is shown in green. Note, the gradient of the binding constraint has been scaled for visual effect. As expected, the level curves shown are given by the (implicit) equation $xy = k$. As you go up and right, the value of k increases.

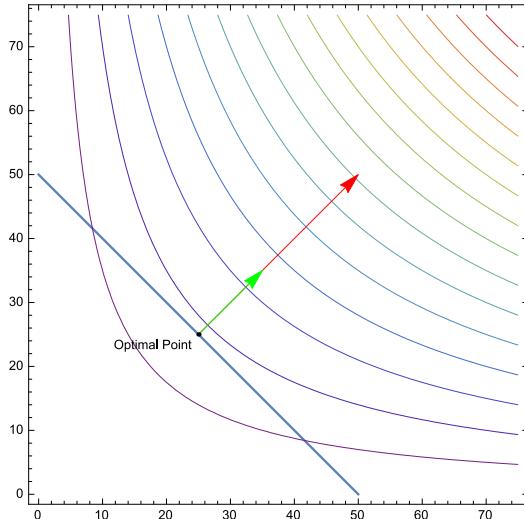


Figure 15.11: Visualization of the KKT conditions for a simple nonlinear programming problem. Notice the point of optimality is not at an extreme point. Moreover, the gradient of the binding constraint is parallel to the gradient of the objective, as expected. Note, the gradient of the binding constraint has been scaled larger for visual effect.

15.18 Supplementary Material: KKT Conditions ⁵

15.18.1 Dual Problem

Given the following convex minimization problem:

Problem 15.22 — convex minimization problem.

$$\begin{aligned} \min_x \quad & f(x) \\ \text{subject to} \quad & h_i(x) \leq 0, \quad i = 1, \dots, m \\ & \ell_j(x) = 0, \quad j = 1, \dots, r \end{aligned}$$

The Lagrangian is defined as $L(x, u, v) = f(x) + \sum_{i=1}^m u_i h_i(x) + \sum_{j=1}^r v_j \ell_j(x)$. The Lagrange dual function is defined as $g(u, v) = \min_x L(x, u, v)$. The dual problem is

$$\begin{aligned} \max_{u, v} \quad & g(u, v) \\ \text{subject to} \quad & u \geq 0 \end{aligned}$$

The Lagrange dual function $g(u, v)$ is always *concave* regardless of whether the primal problem is convex or not.

Weak duality: $f^* \geq g^*$ holds for all problems, where f^* and g^* are primal and dual optimal values, respectively.

Slater's condition, which says the primal has at least one strictly feasible point, is a sufficient condition for *strong duality* to hold. If $\exists x$ such that $h_i(x) < 0, i = 1, \dots, m$ and $\ell_j(x) = 0, j = 1, \dots, r, f^* = g^*$. This condition can be further refined to $h_i(x) < 0$ for all i such that h_i is nonaffine. As a result, Slater's condition is reduced to feasibility for LP's.

15.19 Karush-Kuhn-Tucker (KKT) Conditions

For the given problem 15.22, the KKT conditions are:

⁵These notes are taken from Machine Learning 10-725 (Convex Optimization).

1. $0 \in \partial_x \left(f(x) + \sum_{i=1}^m u_i h_i(x) + \sum_{j=1}^r v_j \ell_j(x) \right)$ (stationary)
2. $u_i \cdot h_i(x) = 0$ for all i (complementary slackness)
3. $h_i(x) \leq 0, \ell_j(x) = 0$ for all i, j (primal feasibility)
4. $u_i \geq 0$ for all i (dual feasibility)

Theorem 15.19.1 For x^* and u^*, v^* to be primal and dual solutions, KKT conditions are sufficient.

Proof. Proof:

Sufficiency: if $\exists x^*$ and u^*, v^* that satisfy the KKT conditions, $g(u^*, v^*) = f(x^*) + \sum_{i=1}^m u_i^* h_i(x^*) + \sum_{j=1}^r v_j^* \ell_j(x^*) = f(x^*)$

The first equality holds from stationarity, since $f(x) + \sum_{i=1}^m u_i h_i(x) + \sum_{j=1}^r v_j \ell_j(x)$ is convex, so any stationary point is a minimizer, and the second holds by complementary slackness.

By weak duality, x^* and u^*, v^* are optimal. It always implies that the duality gap is 0. ■

Theorem 15.19.2 For a problem with strong duality (e.g. assume Slater's condition: convex problem and there exists x strictly satisfying nonaffine inequality constraints),

x^* and u^*, v^* are primal and dual solutions $\iff x^*$ and u^*, v^* satisfy the KKT conditions

Proof. Sufficiency: Follows from Theorem 15.19.1.

Necessity: Let x^* and u^*, v^* be primal and dual solutions, and suppose we know strong duality holds. Then

$$\begin{aligned} f(x^*) &= g(u^*, v^*) \\ &= \min_x \left(f(x) + \sum_{i=1}^m u_i^* h_i(x) + \sum_{j=1}^r v_j^* \ell_j(x) \right) \\ &\leq f(x^*) + \sum_{i=1}^m u_i^* h_i(x^*) + \sum_{j=1}^r v_j^* \ell_j(x^*) \\ &\leq f(x^*) \end{aligned}$$

The LSH equals RHS, so all inequalities in the equation must be equalities. Looking at the KKT conditions one by one, primal and dual feasibility holds, by virtue of optimality. Stationarity comes from the fact that x^* minimizes $f(x) + \sum_{i=1}^m u_i^* h_i(x) + \sum_{j=1}^r v_j^* \ell_j(x)$. Since x^* is the minimizer, it must be a stationary point for this function. Complementary slackness comes from $f(x^*) + \sum_{i=1}^m u_i^* h_i(x^*) + \sum_{j=1}^r v_j^* \ell_j(x^*) = f(x^*)$, since we must have $\sum_{i=1}^m u_i^* h_i(x^*) = 0$ and they are each non-negative. ■

16. 非线性最小二乘法

16.1 非线性最小二乘法的定义

Problem 16.1 $f_1(x), \dots, f_m(x)$ 是可微函数;

优化目标:

$$\min_x \sum_{i=1}^m (f_i(x))^2$$

Problem 16.2 设函数 $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}^m$ ，其第 i 个分量为函数 $f_i(x)$

则有

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_m(x) \end{bmatrix}$$

优化目标

$$\min_x \sum_{i=1}^m (f_i(x))^2 = \|f(x)\|_2^2$$

如果 $f(x) = Ax - b$ ，问题简化为(线性)最小二乘问题。

16.1.1 例子：距离测量定位

Problem 16.3 向量 x_{ex} 表示二维或三维中的未知位置，通过测量到的已知点 a_1, \dots, a_m 的距离来估计 x_{ex}

$$\rho_i = \|x_{ex} - a_i\|_2 + v_i, \quad i = 1, \dots, m$$

其中 v_i 是测量误差。

非线性最小二乘法估计：通过最小化估计 \hat{x} 的位置

$$\min_x \sum_{i=1}^m (\|x - a_i\|_2 - \rho_i)^2 = \|f(x)\|_2^2$$

函数 $f_i(x) = \|x - a_i\|_2 - \rho_i$ 是 $f(x)$ 的 i 个分量。

16.1.2 例子：多个相机视图定位

! 这个例子与遥感卫星成像等有关。

建立一个理想的相机模型，由参数 $A \in \mathbb{R}^{2 \times 3}, b \in \mathbb{R}^2, c \in \mathbb{R}^3, d \in \mathbb{R}$ 来描述。相机及其位置和方向用 A, b, c, d 来刻画。

目标位置 $x \in \mathbb{R}^3$ 在二维平面图像投影位置 $x' \in \mathbb{R}^2$ 。

$$x' = \frac{1}{c^T x + d} (Ax + b)$$

如果物体在摄像机前面，则 $c^T x + d > 0$ 。

Problem 16.4 位于 x_{ex} 位置的物体由 l 个相机观察（由 A_i, b_i, c_i, d_i 描述），目标在相机图像平面上位置 $y_i \in \mathbb{R}^2$

$$y_i = \frac{1}{c_i^T x_{ex} + d_i} (A_i x_{ex} + b_i) + v_i$$

v_i 为测量误差或量化误差。目的是从 l 个观测点 y_1, \dots, y_l 来估计三维位置 x_{ex} 。 ■

使用非线性最小二乘法估计

$$\min_x \sum_{i=1}^l \left\| \frac{1}{c_i^T x + d_i} (A_i x + b_i) - y_i \right\|_2^2$$

这是关于 $m = 2l$ 的非线性最小二乘法问题， $(y_i)_j$ 是 y_i 的第 j 个分量：

$$f_i(x) = \frac{(A_i x + b_i)_1}{c_i^T x + d_i} - (y_i)_1, \quad f_{l+i}(x) = \frac{(A_i x + b_i)_2}{c_i^T x + d_i} - (y_i)_2$$

16.1.3 例子：模型拟合

Problem 16.5

$$\min_{\theta} \sum_{i=1}^N \left(\hat{f}(x^{(i)}, \theta) - y^{(i)} \right)^2$$

$(x^{(i)}, y^{(i)})$, $i = 1, \dots, N$ 表示样本。设函数 $\hat{f}(x, \theta)$ 的参数 $\theta = (\theta_1, \dots, \theta_p)$ 。最小化的目标是估计函数参数 θ 。 ■

假设函数 $\hat{f}(x, \theta)$ 关于 θ 的线性函数

$$\hat{f}(x, \theta) = \theta_1 f_1(x) + \dots + \theta_p f_p(x)$$

当然 $\hat{f}(x, \theta)$ 也可以是关于 θ 的非线性函数。

■ **Example 16.1** 有四个变量 $\theta_1, \theta_2, \theta_3, \theta_4$ 的非线性最小二乘问题：

$$\min_{\theta} \sum_{i=1}^N \left(\theta_1 e^{\theta_2 x^{(i)}} \cos(\theta_3 x^{(i)} + \theta_4) - y^{(i)} \right)^2$$

16.1.4 例子：正交距离回归

正交距离回归目标：最小化 $\hat{f}(x, \theta)$ 图中数据点到曲线的均方距离。

例子：三次多项式的正交距离回归：

$$\hat{f}(x, \theta) = \theta_1 + \theta_2 x + \theta_3 x^2 + \theta_4 x^3$$

16.1.5 非线性最小二乘法

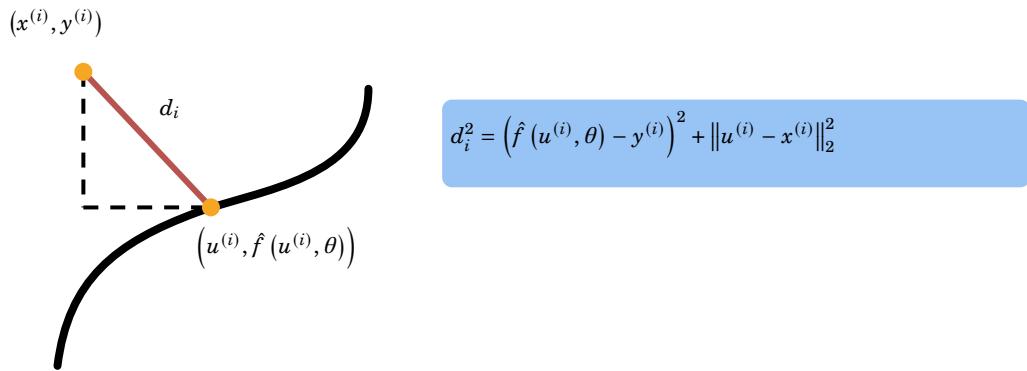
Problem 16.6

$$\min_{\theta, u^{(i)}} \sum_{i=1}^N \left((\hat{f}(u^{(i)}, \theta) - y^{(i)})^2 + \|u^{(i)} - x^{(i)}\|_2^2 \right)$$

这个模型需要优化参数 θ 和 N 个点 $u^{(i)}$ 。

第 i 项为数据点 $(x^{(i)}, y^{(i)})$ 到点 $(u^{(i)}, \hat{f}(u^{(i)}, \theta))$ 距离的平方。

Figure 16.1: 优化模型的几何意义



通过 $u^{(i)}$ 与图中点 $(x^{(i)}, y^{(i)})$ 的平方距离来最小化 d_i^2 。最小化 $\sum_i d_i^2$ ，即通过 $u^{(1)}, \dots, u^{(N)}$ 和 θ 来最小化均方距离。

16.1.6 二分类

二分类的目标函数为 $\hat{f}(x, \theta) = \text{sign}(\theta_1 f_1(x) + \theta_2 f_2(x) + \dots + \theta_p f_p(x))$

Problem 16.7 — 二分类问题.

$$\min_{\theta} \sum_{i=1}^N \left(\phi(\theta_1 f_1(x^{(i)}) + \dots + \theta_p f_p(x^{(i)})) - y^{(i)} \right)^2$$

$(x^{(i)}, y^{(i)})$ 是数据点， $y^{(i)} \in \{-1, 1\}$ 。 $\phi(u)$ 是 sigmoid 函数，它是 $\text{sign}(u)$ 函数的一个可微的近似函数。

$$\phi(u) = \frac{e^u - e^{-u}}{e^u + e^{-u}}$$

通过计算 θ 来求解非线性最小二乘问题，大概率能得到良好的结果。

16.2 梯度

16.2.1 Gradient and Directional Derivative

Definition 16.2.1 — $\mathbb{R}^n \rightarrow \mathbb{R}$ 函数 g 的梯度. 可微函数 $g : \mathbb{R}^n \rightarrow \mathbb{R}$ 在 $z \in \mathbb{R}^n$ 的梯度 (gradient) 为

$$\nabla g(z) = \begin{bmatrix} \frac{\partial g}{\partial x_1}(z) \\ \frac{\partial g}{\partial x_2}(z) \\ \vdots \\ \frac{\partial g}{\partial x_n}(z) \end{bmatrix}$$

Definition 16.2.2 — g 在 z 附近的仿射近似 (一阶泰勒公式, 线性化).

$$\begin{aligned} \hat{g}(x) &= g(z) + \frac{\partial g}{\partial x_1}(z)(x_1 - z_1) + \cdots + \frac{\partial g}{\partial x_n}(z)(x_n - z_n) \\ &= g(z) + \nabla g(z)^T(x - z) \end{aligned}$$

Definition 16.2.3 — **Directional Derivative.** For given z and nonzero v , define $h(t) = g(z + tv)$

The derivative of h at $t = 0$

$$\begin{aligned} h'(0) &= \frac{\partial g}{\partial x_1}(z)v_1 + \frac{\partial g}{\partial x_2}(z)v_2 + \cdots + \frac{\partial g}{\partial x_n}(z)v_n \\ &= \nabla g(z)^T v \end{aligned}$$

This is called the *directional derivative* of g (at z , in the direction v), v is a *descent direction* of g at z if $\nabla g(z)^T v < 0$.

16.2.2 Jacobian Matrices

Definition 16.2.4 — **Jacobian Matrices.** 可微函数 $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ 在 $z \in \mathbb{R}^n$ 的导数矩阵 (Jacobian 矩阵)

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_m(x) \end{bmatrix} \quad Df(z) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(z) & \frac{\partial f_1}{\partial x_2}(z) & \cdots & \frac{\partial f_1}{\partial x_n}(z) \\ \frac{\partial f_2}{\partial x_1}(z) & \frac{\partial f_2}{\partial x_2}(z) & \cdots & \frac{\partial f_2}{\partial x_n}(z) \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_m}{\partial x_1}(z) & \frac{\partial f_m}{\partial x_2}(z) & \cdots & \frac{\partial f_m}{\partial x_n}(z) \end{bmatrix}$$

Definition 16.2.5 — f 在 z 附近的仿射近似 (线性化).

$$\begin{aligned} \hat{f}(x) &= f(z) + Df(z)(x - z) \\ &= \begin{bmatrix} f_1(z) \\ f_2(z) \\ \vdots \\ f_m(z) \end{bmatrix} + \left(\begin{bmatrix} \frac{\partial f_1}{\partial x_1}(z) & \frac{\partial f_1}{\partial x_2}(z) & \cdots & \frac{\partial f_1}{\partial x_n}(z) \\ \frac{\partial f_2}{\partial x_1}(z) & \frac{\partial f_2}{\partial x_2}(z) & \cdots & \frac{\partial f_2}{\partial x_n}(z) \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_m}{\partial x_1}(z) & \frac{\partial f_m}{\partial x_2}(z) & \cdots & \frac{\partial f_m}{\partial x_n}(z) \end{bmatrix} \begin{bmatrix} x_1 - z_1 \\ x_2 - z_2 \\ \vdots \\ x_n - z_n \end{bmatrix} \right)_{m \times 1} \end{aligned}$$

用符号 $\hat{f}(x; z)$ 表示点 z 附近的线性化逼近。

16.2.3 Hessian

Definition 16.2.6 — Hessian Matrices. Hessian of g at z is a symmetric $n \times n$ matrix $\nabla^2 g(z)$ with elements

$$\nabla^2 g(z)_{ij} = \frac{\partial^2 g}{\partial x_i \partial x_j}(z)$$

This is also the derivative matrix $Df(z)$ of $f(x) = \nabla g(x)$ at z .

Definition 16.2.7 — Quadratic (second order) approximation of g around z .

$$g_q(x) = g(z) + \nabla g(z)^T(x - z) + \frac{1}{2}(x - z)^T \nabla^2 g(z)(x - z)$$

- **Example 16.2** Affine function: $g(x) = a^T x + b$

$$\nabla g(x) = a, \quad \nabla^2 g(x) = 0$$

Quadratic function: $g(x) = x^T P x + q^T x + r$ with P symmetric

$$\nabla g(x) = 2Px + q, \quad \nabla^2 g(x) = 2P$$

Least squares cost: $g(x) = \|Ax - b\|^2 = x^T A^T Ax - 2b^T Ax + b^T b$

$$\nabla g(x) = 2A^T Ax - 2A^T b, \quad \nabla^2 g(x) = 2A^T A$$

Theorem 16.2.1 — Linear combination properties. If $g(x) = \alpha_1 g_1(x) + \alpha_2 g_2(x)$, then

$$\begin{aligned}\nabla g(x) &= \alpha_1 \nabla g_1(x) + \alpha_2 \nabla g_2(x) \\ \nabla^2 g(x) &= \alpha_1 \nabla^2 g_1(x) + \alpha_2 \nabla^2 g_2(x)\end{aligned}$$

Theorem 16.2.2 — Composition with affine mapping properties. if $g(x) = h(Cx + d)$, then

$$\begin{aligned}\nabla g(x) &= C^T \nabla h(Cx + d) \\ \nabla^2 g(x) &= C^T \nabla^2 h(Cx + d)C\end{aligned}$$

- **Example 16.3**

$$g(x_1, x_2) = e^{x_1+x_2-1} + e^{x_1-x_2-1} + e^{-x_1-1}$$

Gradient

$$\nabla g(x) = \begin{bmatrix} e^{x_1+x_2-1} + e^{x_1-x_2-1} - e^{-x_1-1} \\ e^{x_1+x_2-1} - e^{x_1-x_2-1} \end{bmatrix}$$

Hessian

$$\nabla^2 g(x) = \begin{bmatrix} e^{x_1+x_2-1} + e^{x_1-x_2-1} + e^{-x_1-1} & e^{x_1+x_2-1} - e^{x_1-x_2-1} \\ e^{x_1+x_2-1} - e^{x_1-x_2-1} & e^{x_1+x_2-1} + e^{x_1-x_2-1} \end{bmatrix}$$

Gradient and Hessian via composition property:

express g as $g(x) = h(Cx + d)$ with $h(y_1, y_2, y_3) = e^{y_1} + e^{y_2} + e^{y_3}$ and

$$C = \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ -1 & 0 \end{bmatrix}, \quad d = \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix}$$

Gradient: $\nabla g(x) = C^T \nabla h(Cx + d)$

$$\nabla g(x) = \begin{bmatrix} 1 & 1 & -1 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} e^{x_1+x_2-1} \\ e^{x_1-x_2-1} \\ e^{-x_1-1} \end{bmatrix}$$

Hessian: $\nabla^2 g(x) = C^T \nabla h^2(Cx + d)C$

$$\nabla^2 g(x) = \begin{bmatrix} 1 & 1 & -1 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} e^{x_1+x_2-1} & 0 & 0 \\ 0 & e^{x_1-x_2-1} & 0 \\ 0 & 0 & e^{-x_1-1} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ -1 & 0 \end{bmatrix}$$

■

16.2.4 Optimality conditions for twice differentiable g

Theorem 16.2.3 — Necessary condition for twice differentiable g . If x^* is locally optimal, then $\nabla g(x^*) = 0$ and $\nabla^2 g(x^*)$ is positive semidefinite

Theorem 16.2.4 — Sufficient condition. If x^* satisfies $\nabla g(x^*) = 0$ and $\nabla^2 g(x^*)$ is positive definite then x^* is locally optimal.

Definition 16.2.8 — Convex functions. g is called *convex* if $\nabla^2 g(x)$ is positive semidefinite everywhere.

Theorem 16.2.5 — Necessary and sufficient condition for convex functions. if g is convex then x^* is optimal if and only if $\nabla g(x^*) = 0$

■ **Example 16.4** $g(x) = \log(e^x + e^{-x})$

$$g'(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \quad g''(x) = \frac{4}{(e^x + e^{-x})^2}$$

$g''(x) \geq 0$ everywhere; $x^* = 0$ is the unique optimal point

■ **Example 16.5** - $g(x) = x^4$

$$g'(x) = 4x^3, \quad g''(x) = 12x^2$$

$g''(x) \geq 0$ everywhere; $x^* = 0$ is the unique optimal point

■ **Example 16.6** - $g(x) = x^3$

$$g'(x) = 3x^2, \quad g''(x) = 6x$$

$g'(0) = 0, g''(0) = 0$ but $x = 0$ is not locally optimal

■ **Example 16.7** - $g(x) = x^T Px + q^T x + r$ (P is symmetric positive definite)

$$\nabla g(x) = 2Px + q, \quad \nabla^2 g(x) = 2P$$

$\nabla^2 g(x)$ is positive definite everywhere, hence the unique optimal point is

$$x^* = -(1/2)P^{-1}q$$

■

- Example 16.8 - $g(x) = \|Ax - b\|^2$ (A is a matrix with linearly independent columns)

$$\nabla g(x) = 2A^T Ax - 2A^T b, \quad \nabla^2 g(x) = 2A^T A$$

$\nabla^2 g(x)$ is positive definite everywhere, hence the unique optimal point is

$$x^* = (A^T A)^{-1} A^T b$$

- Example 16.9

$$g(x_1, x_2) = e^{x_1+x_2-1} + e^{x_1-x_2-1} + e^{-x_1-1}$$

Gradient

$$\nabla g(x) = \begin{bmatrix} e^{x_1+x_2-1} + e^{x_1-x_2-1} - e^{-x_1-1} \\ e^{x_1+x_2-1} - e^{x_1-x_2-1} \end{bmatrix}$$

Hessian

$$\nabla^2 g(x) = \begin{bmatrix} e^{x_1+x_2-1} + e^{x_1-x_2-1} + e^{-x_1-1} & e^{x_1+x_2-1} - e^{x_1-x_2-1} \\ e^{x_1+x_2-1} - e^{x_1-x_2-1} & e^{x_1+x_2-1} + e^{x_1-x_2-1} \end{bmatrix}$$

We can express $\nabla^2 g(x)$ as

$$\nabla^2 g(x) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} e^{x_1+x_2-1} & 0 & 0 \\ 0 & e^{x_1-x_2-1} & 0 \\ 0 & 0 & e^{-x_1-1} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & 0 \end{bmatrix}$$

this shows that $\nabla^2 g(x)$ is positive definite for all x therefore x^* is optimal if and only if

$$\nabla g(x^*) = \begin{bmatrix} e^{x_1^*+x_2^*-1} + e^{x_1^*-x_2^*-1} - e^{-x_1^*-1} \\ e^{x_1^*+x_2^*-1} - e^{x_1^*-x_2^*-1} \end{bmatrix} = 0$$

two nonlinear equations in two variables

16.3 求解非线性最小二乘法：目标梯度

对于优化问题 16.2

$$g(x) = \|f(x)\|_2^2 = \sum_{i=1}^m (f_i(x))^2, f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_m(x) \end{bmatrix}$$

g 对 x_j 的一阶导数为

$$\begin{aligned} \frac{\partial g}{\partial x_j}(z) &= 2 \sum_{i=1}^m f_i(z) \frac{\partial f_i}{\partial x_j}(z) \\ &= 2 \left[\frac{\partial f_1}{\partial x_j}(z), \frac{\partial f_2}{\partial x_j}(z), \dots, \frac{\partial f_m}{\partial x_j}(z) \right] \begin{bmatrix} f_1(z) \\ f_2(z) \\ \vdots \\ f_m(z) \end{bmatrix} \end{aligned}$$

注意到

$$(Df(z))^T = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(z) & \frac{\partial f_1}{\partial x_2}(z) & \cdots & \frac{\partial f_1}{\partial x_n}(z) \\ \frac{\partial f_2}{\partial x_1}(z) & \frac{\partial f_2}{\partial x_2}(z) & \cdots & \frac{\partial f_2}{\partial x_n}(z) \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_m}{\partial x_1}(z) & \frac{\partial f_m}{\partial x_2}(z) & \cdots & \frac{\partial f_m}{\partial x_n}(z) \end{bmatrix} = [\nabla f_1(z), \nabla f_2(z), \dots, \nabla f_m(z)]$$

g 在 z 的梯度为

$$\nabla g(z) = \begin{bmatrix} \frac{\partial g}{\partial x_1}(z) \\ \vdots \\ \frac{\partial g}{\partial x_n}(z) \end{bmatrix} = 2 \sum_{i=1}^m f_i(z) \nabla f_i(z) = 2(Df(z))^T f(z)$$

Theorem 16.3.1 — 非线性最小二乘问题的最优必要条件.

$$\min_x g(x) = \sum_{i=1}^m f_i(x)^2 \quad f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_m(x) \end{bmatrix}$$

如果要 x 使 $g(x)$ 最小, 则必须满足:

$$\nabla g(x) = 2Df(x)^T f(x) = 0$$

Corollary 16.3.2 — 正规方程的最优必要条件. 推广到正规方程, 如果 $f(x) = Ax - b$, 那么 $Df(x) = A$ 和

$$\nabla g(x) = 2A^T(Ax - b)$$



对于一般函数 $f(x)$, $\nabla g(x) = 0$ 不是最优解的充分条件。

16.4 Gauss-Newton Algorithm

Problem 16.8

$$\min_x g(x) = \|f(x)\|_2^2 = \sum_{i=1}^m f_i(x)^2$$

从某个初始值 $x^{(1)}$ 开始, 当迭代 $x^{(2)}, x^{(3)}, \dots$
函数 f 在 $x^{(k)}$ 附近的泰勒展开式是

$$\hat{f}(x; x^{(k)}) = f(x^{(k)}) + Df(x^{(k)})(x - x^{(k)})$$

用仿射近似 $\hat{f}(x; x^{(k)})$ 代替最小二乘法问题中函数 $f(x)$

$$x^{(k+1)} = \arg \min_x \|\hat{f}(x; x^{(k)})\|_2^2$$

第 k 次迭代问题：

$$\begin{aligned} x^{(k+1)} &= \arg \min_x \|\hat{f}(x; x^{(k)})\|_2^2 \\ x^{(k+1)} &= \arg \min_x h(x) = \|f(x^{(k)}) + Df(x^{(k)})(x - x^{(k)})\|_2^2 \end{aligned}$$

函数 $h(x)$ 的梯度：

$$\nabla h(x) = 2Df(x^{(k)})^T (f(x^{(k)}) + Df(x^{(k)})(x - x^{(k)})) = 0$$

则有

$$Df(x^{(k)})^T Df(x^{(k)}) x^{(k)} - Df(x^{(k)})^T f(x^{(k)}) = Df(x^{(k)})^T Df(x^{(k)}) x$$

$$Df(x^{(k)})^T Df(x^{(k)}) x^{(k)} - Df(x^{(k)})^T f(x^{(k)}) = Df(x^{(k)})^T Df(x^{(k)}) x$$

如果 $Df(x^{(k)})$ 的列是线性无关的，则其解为：

$$x^{(k+1)} = x^{(k)} - \left(Df(x^{(k)})^T Df(x^{(k)}) \right)^{-1} Df(x^{(k)})^T f(x^{(k)})$$

高斯-牛顿法步骤 $\Delta x^{(k)} = x^{(k+1)} - x^{(k)}$ ：

$$\begin{aligned} \Delta x^{(k)} &= - \left(Df(x^{(k)})^T Df(x^{(k)}) \right)^{-1} Df(x^{(k)})^T f(x^{(k)}) \\ &= -\frac{1}{2} \left(Df(x^{(k)})^T Df(x^{(k)}) \right)^{-1} \nabla g(x^{(k)}) \end{aligned}$$

(利用第 14.10 的 $\nabla g(x)$ 表达式)。

逼近函数 $\hat{f}(x; x^{(k)})$ 的关于 $x^{(k+1)}$ 的代价：

$$\begin{aligned} \|\hat{f}(x^{(k+1)}; x^{(k)})\|_2^2 &= \|f(x^{(k)}) + Df(x^{(k)}) \Delta x^{(k)}\|_2^2 \\ &= \|f(x^{(k)})\|_2^2 + 2f(x^{(k)})^T Df(x^{(k)}) \Delta x^{(k)} + \|Df(x^{(k)}) \Delta x^{(k)}\|_2^2 \end{aligned}$$

$$2Df(x^{(k)})^T (f(x^{(k)}) + Df(x^{(k)})(x^{(k+1)} - x^{(k)})) = 0 \text{ 目标函数梯度等于 } 0 \Rightarrow Df(x^{(k)})^T f(x^{(k)}) = -Df(x^{(k)})^T Df(x^{(k)}) \Delta x^{(k)} \Rightarrow (\Delta x^{(k)})^T Df(x^{(k)})^T f(x^{(k)}) = -(\Delta x^{(k)})^T Df(x^{(k)})^T Df(x^{(k)}) \Delta x^{(k)} = -\|Df(x^{(k)}) \Delta x^{(k)}\|_2^2$$

$$\|\hat{f}(x^{(k+1)}; x^{(k)})\|_2^2 = \|f(x^{(k)})\|_2^2 - \|Df(x^{(k)}) \Delta x^{(k)}\|_2^2$$

如果 $Df(x^{(k)})$ 的列向量线性无关，且 $\Delta x^{(k)} \neq 0$ ：

$$\|Df(x^{(k)}) \Delta x^{(k)}\|_2^2 > 0 \Rightarrow \|\hat{f}(x^{(k+1)}; x^{(k)})\|_2^2 < \|f(x^{(k)})\|_2^2$$

当 $m = n$ 时， $Df(x^{(k)})$ 的列向量线性无关，高斯-牛顿法可简化为

$$\begin{aligned}
x^{(k+1)} &= x^{(k)} - \left(Df(x^{(k)})^T Df(x^{(k)}) \right)^{-1} Df(x^{(k)})^T f(x^{(k)}) \\
&= x^{(k)} - \left(Df(x^{(k)}) \right)^{-1} \left(Df(x^{(k)})^T \right)^{-1} Df(x^{(k)})^T f(x^{(k)}) \\
&= x^{(k)} - \left(Df(x^{(k)}) \right)^{-1} f(x^{(k)}).
\end{aligned}$$

如果 $m = n = 1$ 时, 迭代更新可进一步简化为

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}, f'(x^{(k)}) \neq 0$$

然而, $\hat{f}(x; x^{(k)})$ 只是 $f(x)$ 的局部近似, 仍然存在:

$$\|f(x^{(k+1)})\|_2^2 > \|f(x^{(k)})\|_2^2$$

Jacobian 矩阵 $Df(x^{(k)})$ 的可能"列向量是线性相关" $(Df(x^{(k)})^T Df(x^{(k)}))$ 奇异, 不可逆

16.5 Levenberg—Marquardt Algorithm

Levenberg-Marquardt(LM) 算法解决高斯-牛顿法的两个难点:

- 1.
- 1. 当列 $Df(x^{(k)})$ 是线性相关时, 如何更新 $x^{(k)}$? 2. 当高斯-牛顿法的更新过程, $\|f(x)\|_2^2$ 并没有减少。

LM 算法通过求解正则化最小二乘法更新 $x^{(k+1)}$:

$$x^{(k+1)} = \arg \min_x \|\hat{f}(x; x^{(k)})\|_2^2 + \lambda^{(k)} \|x - x^{(k)}\|_2^2$$

要求 x 靠近 $x^{(k)}$, 即 $\hat{f}(x; x^{(k)}) \approx f(x)$. The second term forces x to be close to $x^{(k)}$ where $\hat{f}(x; x^{(k)}) \approx f(x)$. 当 $\lambda^{(k)} > 0$ 时, 总有唯一解(无需对 $Df(x^{(k)})$ 限制)。

第 k 次迭代求解的最小二乘正则化问题:

$$x^{(k+1)} = \arg \min_x \|f(x^{(k)}) + Df(x^{(k)})(x - x^{(k)})\|_2^2 + \lambda^{(k)} \|x - x^{(k)}\|_2^2$$

令目标函数导数等于 0:

$$\begin{aligned}
&2Df(x^{(k)})^T (f(x^{(k)}) + Df(x^{(k)})(x - x^{(k)})) + 2\lambda^{(k)}(x - x^{(k)}) = 0 \\
&\Rightarrow \left(Df(x^{(k)})^T Df(x^{(k)}) + \lambda^{(k)} I \right) (x - x^{(k)}) = -Df(x^{(k)})^T f(x^{(k)})
\end{aligned}$$

$x^{(k+1)}$ 更新:

$$x^{(k+1)} = x^{(k)} - \left(Df(x^{(k)})^T Df(x^{(k)}) + \lambda^{(k)} I \right)^{-1} Df(x^{(k)})^T f(x^{(k)})$$

$$x^{(k+1)} = x^{(k)} - \left(Df(x^{(k)})^T Df(x^{(k)}) + \lambda^{(k)} I \right)^{-1} Df(x^{(k)})^T f(x^{(k)})$$

LM 算法步骤 $\Delta x^{(k)} = x^{(k+1)} - x^{(k)}$ 是:

$$\begin{aligned}\Delta x^{(k)} &= - \left(Df(x^{(k)})^T Df(x^{(k)}) + \lambda^{(k)} I \right)^{-1} Df(x^{(k)})^T f(x^{(k)}) \\ &= -\frac{1}{2} \left(Df(x^{(k)})^T Df(x^{(k)}) + \lambda^{(k)} I \right)^{-1} \nabla g(x^{(k)})\end{aligned}$$

对于 $\lambda^{(k)} = 0$ ，简化为高斯-牛顿法；对于较大的 $\lambda^{(k)}$ ：

$$\Delta x^{(k)} \approx -\frac{1}{2\lambda^{(k)}} \nabla g(x^{(k)})$$

16.5.1 正则化参数

几种调整 $\lambda^{(k)}$ 的策略 在第 k 次迭代时，求解：

$$\hat{x} = \arg \min_x \left\| \hat{f}(x; x^{(k)}) \right\|_2^2 + \lambda^{(k)} \|x - x^{(k)}\|_2^2$$

如果 $\|f(\hat{x})\|_2^2 < \|f(x^{(k)})\|_2^2$ ，令 $x^{(k+1)} = \hat{x}$ ，并降低 λ 。否则，无需更新 x （取 $x^{(k+1)} = x^{(k)}$ ），而是增加 λ 。

一些变化：比较预期成本降低的情况和实际成本降低的情况。具有“置信区间”的最小二乘法问题: $\min_x \|\hat{f}(x; x^{(k)})\|_2^2$ subject to $\|x - x^{(k)}\|_2^2 \leq \gamma$

Algorithm 20: Levenberg Marquardt Algorithm

- 1 选择 $x^{(1)}$ 和 $\lambda^{(1)}$ 开始，循环 $k = 1, 2, \dots$:
- 2 1. 求 $f(x^{(k)})$ ，然后令 $A = Df(x^{(k)})$ 。
- 3 2. 计算最小二乘法正则化问题的解:

$$\hat{x} = x^{(k)} - \left(A^T A + \lambda^{(k)} I \right)^{-1} A^T f(x^{(k)})$$

- 4 3. 定义 $x^{(k+1)}$ 和 $\lambda^{(k+1)}$ 如下:

$$\begin{cases} x^{(k+1)} = \hat{x}, \lambda^{(k+1)} = \beta_1 \lambda^{(k)} & \text{if } \|f(\hat{x})\|_2^2 < \|f(x^{(k)})\|_2^2 \\ x^{(k+1)} = x^{(k)}, \lambda^{(k+1)} = \beta_2 \lambda^{(k)} & \text{others} \end{cases}$$

其中 β_1, β_2 是常数， $0 < \beta_1 < 1 < \beta_2$ 。在步骤 2 中，可以使用 QR 分解来计算 \hat{x} 。迭代会在当 $\nabla g(x^{(k)}) = 2A^T f(x^{(k)})$ 足够小时终止。

16.6 Newton Method

Assume $f : \mathbf{R}^n \rightarrow \mathbf{R}^n$ is differentiable.

$Df(x^{(k)})$ is the derivative matrix of f at $x^{(k)}$.

Each iteration requires one evaluation of $f(x)$ and $Df(x)$, each iteration requires factorization of the $n \times n$ matrix $Df(x)$.

We assume $Df(x)$ is nonsingular.

```

1 choose  $x^{(1)}$ 
2 while  $k = 1, 2, \dots$  do
3   
$$x^{(k+1)} = x^{(k)} - Df(x^{(k)})^{-1} f(x^{(k)})$$

4 end

```

16.6.1 Interpretation of Newton Method

$$x^{(k+1)} = x^{(k)} - Df(x^{(k)})^{-1} f(x^{(k)})$$

linearize f (i.e., make affine approximation) around current iterate $x^{(k)}$

$$\hat{f}(x; x^{(k)}) = f(x^{(k)}) + Df(x^{(k)}) (x - x^{(k)})$$

solve the linearized equation $\hat{f}(x; x^{(k)}) = 0$; the solution is

$$x = x^{(k)} - Df(x^{(k)})^{-1} f(x^{(k)})$$

take the solution x of the linearized equation as the next iterate $x^{(k+1)}$

16.6.2 One Variable

affine approximation of f around $x^{(k)}$ is

$$\hat{f}(x; x^{(k)}) = f(x^{(k)}) + f'(x^{(k)}) (x - x^{(k)})$$

solve the linearized equation $\hat{f}(x; x^{(k)}) = 0$ and take the solution as $x^{(k+1)}$:

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}$$

16.6.3 Relation to Gauss–Newton method

recall Gauss–Newton method for nonlinear least squares problem

Problem 16.9 — nonlinear least squares problem.

$$\text{minimize } \|f(x)\|^2$$

where f is a differentiable function from \mathbf{R}^n to \mathbf{R}^m .

Gauss–Newton update

$$x^{(k+1)} = x^{(k)} - \left(Df(x^{(k)})^T Df(x^{(k)}) \right)^{-1} Df(x^{(k)})^T f(x^{(k)})$$

If $m = n$, then $Df(x)$ is square and this is the Newton update

$$x^{(k+1)} = x^{(k)} - Df(x^{(k)})^{-1} f(x^{(k)})$$

16.6.4 Examples

■ Example 16.10

$$f_1(x_1, x_2) = \log(x_1^2 + 2x_2^2 + 1) - 0.5 = 0$$

$$f_2(x_1, x_2) = x_2 - x_1^2 + 0.2 = 0$$

two equations in two variables; two solutions $(0.70, 0.29), (-0.70, 0.29)$.

Newton iteration

Algorithm 21: Newton iteration

- 1 evaluate $g = f(x)$ and

$$H = Df(x) = \begin{bmatrix} 2x_1/(x_1^2 + 2x_2^2 + 1) & 4x_2/(x_1^2 + 2x_2^2 + 1) \\ -2x_1 & 1 \end{bmatrix}$$

- 2 solve $Hv = -g$ (two linear equations in two variables)

- 3 update $x := x + v$

Results:

- $x^{(1)} = (1, 1)$: converges to $x^* = (0.70, 0.29)$ in about 4 iterations
- $x^{(1)} = (-1, 1)$: converges to $x^* = (-0.70, 0.29)$ in about 4 iterations
- $x^{(1)} = (1, -1)$ or $x^{(0)} = (-1, -1)$: does not converge

- Newton's method works very well if started near a solution, may not work otherwise
- can converge to different solutions depending on the starting point
- does not necessarily find the solution closest to the starting point

16.6.5 Convergence of Newton's method

Quadratic convergence explains fast convergence when started near solution.

Theorem 16.6.1 — Quadratic Convergence. if $f(x^*) = 0$ and $Df(x^*)$ is nonsingular, and $x^{(1)}$ is sufficiently close to x^* , then

$$x^{(k)} \rightarrow x^*, \quad \|x^{(k+1)} - x^*\| \leq c \|x^{(k)} - x^*\|^2$$

for some $c > 0$

16.6.6 Newton's method for minimizing a convex function

if $\nabla^2 g(x)$ is positive definite everywhere, we can minimize $g(x)$ by solving

$$\nabla g(x) = 0$$

Algorithm:

$v = -\nabla^2 g(x)^{-1} \nabla g(x)$ is called the Newton step at x .

It converges if started sufficiently close to the solution. Newton step is computed by a Cholesky factorization of the Hessian.

Algorithm 22: Newton's method for minimizing a convex function

```

1 choose  $x^{(1)}$ 
2 while  $k = 1, 2, \dots$  do
3   
$$x^{(k+1)} = x^{(k)} - \nabla^2 g(x^{(k)})^{-1} \nabla g(x^{(k)})$$

4 end

```

Interpretations of Newton step

Definition 16.6.1 — affine approximation of $f(x) = \nabla g(x)$ around $x^{(k)}$.

$$\hat{f}(x; x^{(k)}) = \nabla g(x^{(k)}) + \nabla^2 g(x^{(k)}) (x - x^{(k)})$$

is

Newton update $x^{(k+1)}$ is solution of linear equation $\hat{f}(x; x^{(k)}) = 0$.

Definition 16.6.2 — quadratic approximation of $g(x)$ around $x^{(k)}$.

$$g_q(x; x^{(k)}) = g(x^{(k)}) + \nabla g(x^{(k)})^T (x - x^{(k)}) + \frac{1}{2} (x - x^{(k)})^T \nabla^2 g(x^{(k)}) (x - x^{(k)})$$

Newton update $x^{(k+1)}$ minimizes $g_q(x; x^{(k)})$ (satisfies $\nabla g_q(x; x^{(k)}) = 0$)

16.6.7 Damped Newton method**Algorithm 23:** Damped Newton method

```

1 choose  $x^{(1)}$ 
2 while  $k = 1, 2, \dots$  do
3   compute Newton step  $v = -\nabla^2 g(x^{(k)})^{-1} \nabla g(x^{(k)})$ 
4   (Line Search) find largest  $t$  in  $\{1, 0.5, 0.5^2, 0.5^3, \dots\}$  that satisfies
      
$$g(x^{(k)} + tv) < g(x^{(k)})$$

5   
$$x^{(k+1)} = x^{(k)} + tv$$

6 end

```

Positive scalar t is called the *step size*, step 2 in algorithm is called *line search*.

Interpretation of line search

to determine a suitable step size, consider the function $h : \mathbf{R} \rightarrow \mathbf{R}$

$$h(t) = g(x^{(k)} + tv)$$

$h'(0) = \nabla g(x^{(k)})^T v$ is the directional derivative at $x^{(k)}$ in the direction v
line search terminates with positive t if $h'(0) < 0$ (v is a descent direction)

if $\nabla^2 g(x^{(k)})$ is positive definite, the Newton step is a descent direction

$$h'(0) = \nabla g(x^{(k)})^T v = -v^T \nabla^2 g(x^{(k)}) v < 0$$

Newton method for nonconvex functions

if $\nabla^2 g(x^{(k)})$ is not positive definite, it is possible that Newton step v satisfies

$$\nabla g(x^{(k)})^T v = -\nabla g(x^{(k)})^T \nabla^2 g(x^{(k)})^{-1} \nabla g(x^{(k)}) > 0$$

- if Newton step is not descent direction, replace it with descent direction - simplest choice is $v = -\nabla g(x^{(k)})$; practical methods make other choices

16.7 Newton method for nonlinear least squares

16.7.1 Hessian of nonlinear least squares cost

$$g(x) = \|f(x)\|^2 = \sum_{i=1}^m f_i(x)^2$$

Theorem 16.7.1 gradient:

$$\nabla g(x) = 2 \sum_{i=1}^m f_i(x) \nabla f_i(x) = 2 Df(x)^T f(x)$$

Theorem 16.7.2 second derivatives:

$$\frac{\partial^2 g}{\partial x_j \partial x_k}(x) = 2 \sum_{i=1}^m \left(\frac{\partial f_i}{\partial x_j}(x) \frac{\partial f_i}{\partial x_k}(x) + f_i(x) \frac{\partial^2 f_i}{\partial x_j \partial x_k}(x) \right)$$

Theorem 16.7.3 Hessian

$$\nabla^2 g(x) = 2 Df(x)^T Df(x) + 2 \sum_{i=1}^m f_i(x) \nabla^2 f_i(x)$$

Theorem 16.7.4 (Undamped) Newton step at $x = x^{(k)}$:

$$\begin{aligned} v_{\text{nt}} &= -\nabla^2 g(x)^{-1} \nabla g(x) \\ &= - \left(Df(x)^T Df(x) + \sum_{i=1}^m f_i(x) \nabla^2 f_i(x) \right)^{-1} Df(x)^T f(x) \end{aligned}$$

Theorem 16.7.5 Gauss-Newton step at $x = x^{(k)}$

$$v_{\text{gn}} = - \left(Df(x)^T Df(x) \right)^{-1} Df(x)^T f(x)$$

Corollary 16.7.6 can be written as $v_{\text{gn}} = -H_{\text{gn}}^{-1}\nabla g(x)$ where $H_{\text{gn}} = 2Df(x)^T Df(x)$
 H_{gn} is the Hessian without the term $\sum_i f_i(x)\nabla^2 f_i(x)$.

16.7.2 Comparison

Newton step:

- requires second derivatives of f
- not always a descent direction ($\nabla^2 g(x)$ is not necessarily positive definite)
- fast convergence near local minimum

Gauss-Newton step:

- does not require second derivatives
- a descent direction (if columns of $Df(x)$ are linearly independent):

$$\nabla g(x)^T v_{\text{gn}} = -2v_{\text{gn}}^T Df(x)^T Df(x) v_{\text{gn}} < 0 \quad \text{if } v_{\text{gn}} \neq 0$$

- local convergence to x^* is similar to Newton method if

$$\sum_{i=1}^m f_i(x^*) \nabla^2 f_i(x^*)$$

is small (e.g., $f(x^*)$ is small, or f is nearly affine around x^*)

16.8 Unconstrained minimization problem

Problem 16.10 Assume that g is twice differentiable

minimize $g(x_1, x_2, \dots, x_n)$

g is a function from \mathbf{R}^n to \mathbf{R} (the cost function or objective function), $x = (x_1, x_2, \dots, x_n)$ is n -vector of optimization variables.

to solve a maximization problem (i.e., maximize $g(x)$), minimize $-g(x)$. ■

16.9 Local and global optimum

Definition 16.9.1 — globally optimal. x^* is an optimal point (or a minimum) if

$$g(x^*) \leq g(x) \quad \text{for all } x$$

It is also called globally optimal.

Definition 16.9.2 — locally optimal. x^* is a locally optimal point (local minimum) if for some $R > 0$ $g(x^*) \leq g(x)$ for all x with $\|x - x^*\| \leq R$

It is also called locally optimal.

IV

Extensive Reading

17	Fourier Series, Fourier Transform	179
17.1	基本概念	
17.2	Fourier Series	
17.3	Fourier Transform	
17.4	Discrete Fourier Transform	
18	Factorization of Matrices	187
18.1	主要的矩阵分解	

17. Fourier Series, Fourier Transform

17.1 基本概念

Definition 17.1.1 — 波. 波的基本属性：频率、振幅、相位。

Definition 17.1.2 — Complex Numbers.

$$C = R + jI$$

where R and I are real numbers and $j = \sqrt{-1}$. Here, R denotes the real part of the complex number and I its imaginary part.

Real numbers are a subset of complex numbers in which $I = 0$.

Definition 17.1.3 — Complex Number in Polar Coordinates.

$$C = |C|(\cos \theta + j \sin \theta) = |C|e^{j\theta}$$

where $|C| = \sqrt{R^2 + I^2}$ is the length of the vector extending from the origin of the complex plane to point (R, I) , and θ is the angle between the vector and the real axis.

上式使用了欧拉公式

Theorem 17.1.1 — Euler's Formular.

$$e^{j\theta} = \cos \theta + j \sin \theta$$

Corollary 17.1.2 $e^{2\pi i t}$ 可以表示每秒 1 圈的旋转。

Definition 17.1.4 — 正弦函数.

$$y = A \sin(\omega t + \varphi)$$

就是一个以 $\frac{2\pi}{\omega}$ 为周期的正弦函数, 其中 y 表示动点的位置, t 表示时间, A 为振幅, ω 为角频率, φ 为初相.

如何深入研究非正弦周期函数呢? 将周期函数展开成由简单的周期函数例如三角函数组成的级数. 具体地说, 将周期为 $T (= \frac{2\pi}{\omega})$ 的周期函数用一系列以 T 为周期的正弦函数 $A_n \sin(n\omega t + \varphi_n)$ 组成的级数来表示, 记为

$$f(t) = A_0 + \sum_{n=1}^{\infty} A_n \sin(n\omega t + \varphi_n)$$

其中 $A_0, A_n, \varphi_n (n = 1, 2, 3, \dots)$ 都是常数.

将周期函数按上述方式展开, 它的物理意义是很明确的, 这就是把一个比较复杂的周期运动看成是许多不同频率的简谐振动的叠加. 在电工学上, 这种展开称为谐波分析, 其中常数项 A_0 称为 $f(t)$ 的直流分量, $A_1 \sin(\omega t + \varphi_1)$ 称为一次谐波(又叫做基波), $A_2 \sin(2\omega t + \varphi_2), A_3 \sin(3\omega t + \varphi_3), \dots$ 依次称为二次谐波, 三次谐波, 等等.

Definition 17.1.5 — 三角级数. 令 $\frac{a_0}{2} = A_0, a_n = A_n \sin \varphi_n, b_n = A_n \cos \varphi_n, \omega = \frac{\pi}{l}$ (即 $T = 2l$), 则 $f(t) = A_0 + \sum_{n=1}^{\infty} A_n \sin(n\omega t + \varphi_n)$ 可以改写为

$$\frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos \frac{n\pi t}{l} + b_n \sin \frac{n\pi t}{l} \right)$$

其中 $a_0, a_n, b_n (n = 1, 2, 3, \dots)$ 都是常数.

Definition 17.1.6 — 三角函数系(基波).

$$1, \cos x, \sin x, \cos 2x, \sin 2x, \dots, \cos nx, \sin nx, \dots$$

在区间 $[-\pi, \pi]$ 上正交, 就是指在三角函数系(7-4)中任何不同的两个函数的乘积在区间 $[-\pi, \pi]$ 上的积分等于零, 即

$$\begin{aligned} \int_{-\pi}^{\pi} \cos nx \, dx &= 0 \quad (n = 1, 2, 3, \dots), \\ \int_{-\pi}^{\pi} \sin nx \, dx &= 0 \quad (n = 1, 2, 3, \dots), \\ \dots \end{aligned}$$

Definition 17.1.7 — 复数域的 Gram 矩阵. 如果 $A \in \mathbb{C}^{m \times n}$ 的 Gram 矩阵为单位矩阵, 则 A 具有正交列:

$$\begin{aligned} A^H A &= \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix}^H \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix} \\ &= \begin{bmatrix} a_1^H a_1 & a_1^H a_2 & \cdots & a_1^H a_n \\ a_2^H a_1 & a_2^H a_2 & \cdots & a_2^H a_n \\ \vdots & \vdots & & \vdots \\ a_n^H a_1 & a_n^H a_2 & \cdots & a_n^H a_n \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \\ &= I \end{aligned}$$

Gram 矩阵列有单位范数: $\|a_i\|_2^2 = a_i^H a_i = 1$.

Gram 矩阵列是相互正交的：对于 $i \neq j$, $a_i^H a_j = 0$ 。

Definition 17.1.8 — 酉矩阵. 列正交的方形复数矩阵称为酉矩阵。

Definition 17.1.9 — 酉矩阵的逆.

$$\left. \begin{array}{l} A^H A = I \\ A \text{ 是方的} \end{array} \right\} \Rightarrow A A^H = I$$

酉矩阵是具有逆 A^H 的非奇异矩阵。如果 A 是酉矩阵，那么 A^H 也是酉矩阵。

17.2 Fourier Series

Definition 17.2.1 — 傅里叶级数.

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sin kx).$$

where

$$\begin{cases} a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos nx \, dx & (n = 0, 1, 2, 3, \dots), \\ b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin nx \, dx & (n = 1, 2, 3, \dots). \end{cases}$$

Proof. 设 $f(x)$ 是周期为 2π 的周期函数，且能展开成三角级数

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sin kx).$$

先求 a_0 . 上式一从 $-\pi$ 到 π 积分，由于假设式右端级数可逐项积分，因此有

$$\int_{-\pi}^{\pi} f(x) \, dx = \int_{-\pi}^{\pi} \frac{a_0}{2} \, dx + \sum_{k=1}^{\infty} \left[a_k \int_{-\pi}^{\pi} \cos kx \, dx + b_k \int_{-\pi}^{\pi} \sin kx \, dx \right]$$

根据三角函数系的正交性，等式右端除第一项外，其余各项均为零，所以

$$\int_{-\pi}^{\pi} f(x) \, dx = \frac{a_0}{2} \cdot 2\pi$$

于是得

$$a_0 = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \, dx$$

其次求 a_n . 用 $\cos nx$ 乘式两端，再从 $-\pi$ 到 π 积分，得到

$$\begin{aligned} & \int_{-\pi}^{\pi} f(x) \cos nx \, dx \\ &= \frac{a_0}{2} \int_{-\pi}^{\pi} \cos nx \, dx + \sum_{k=1}^{\infty} \left[a_k \int_{-\pi}^{\pi} \cos kx \cos nx \, dx + b_k \int_{-\pi}^{\pi} \sin kx \cos nx \, dx \right] \end{aligned}$$

根据三角函数系的正交性，等式右端除 $k = n$ 的一项外，其余各项均为 0. 所以

$$\int_{-\pi}^{\pi} f(x) \cos nx \, dx = a_n \int_{-\pi}^{\pi} \cos^2 nx \, dx = a_n \pi$$

于是得

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos nx \, dx \quad (n = 1, 2, 3, \dots)$$

类似地, 用 $\sin nx$ 乘 (7-5) 式的两端, 再从 $-\pi$ 到 π 积分, 可得

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin nx \, dx \quad (n = 1, 2, 3, \dots)$$

■

Theorem 17.2.1 — 收敛定理, 狄利克雷 (Dirichlet) 充分条件. 设 $f(x)$ 是周期为 2π 的周期函数, 如果它满足:

1. 在一个周期内连续或只有有限个第一类间断点,

2. 在一个周期内至多只有有限个极值点

那么 $f(x)$ 的傅里叶级数收敛, 并且

- 当 x 是 $f(x)$ 的连续点时, 级数收敛于 $f(x)$;

- 当 x 是 $f(x)$ 的间断点时, 级数收敛于 $\frac{1}{2}[f(x^-) + f(x^+)]$.

收敛定理告诉我们: 只要函数在 $[-\pi, \pi]$ 上至多有有限个第一类间断点, 并且不作无限次振动, 函数的傅里叶级数在连续点处就收敛于该点的函数值, 在间断点处收敛于该点左极限与右极限的算术平均值. 可见, 函数展开成傅里叶级数的条件比展开成幂级数的条件低得多.

Corollary 17.2.2 记 $C = \{x \mid f(x) = \frac{1}{2}[f(x^-) + f(x^+)]\}$

在 C 上就成立 $f(x)$ 的傅里叶级数展开式

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx), \quad x \in C$$

Definition 17.2.2 — 正弦级数. 当 $f(x)$ 为奇函数时, $f(x) \cos nx$ 是奇函数, $f(x) \sin nx$ 是偶函数, 故

$$\left. \begin{aligned} a_n &= 0 \quad (n = 0, 1, 2, \dots), \\ b_n &= \frac{2}{\pi} \int_0^{\pi} f(x) \sin nx \, dx \quad (n = 1, 2, 3, \dots). \end{aligned} \right\}$$

即知奇函数的傅里叶级数是只含有正弦项的正弦级数.

$$\sum_{n=1}^{\infty} b_n \sin nx$$

Definition 17.2.3 — 余弦级数. 当 $f(x)$ 为偶函数时, $f(x) \cos nx$ 是偶函数, $f(x) \sin nx$ 是奇函数, 故

$$\left. \begin{aligned} a_n &= \frac{2}{\pi} \int_0^{\pi} f(x) \cos nx \, dx \quad (n = 0, 1, 2, \dots) \\ b_n &= 0 \quad (n = 1, 2, 3, \dots) \end{aligned} \right\}$$

即知偶函数的傅里叶级数是只含常数项和余弦项的余弦级数

$$\frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos nx$$

在实际应用(如研究某种波动问题,热的传导、扩散问题)中,有时还需要把定义在区间 $[0, \pi]$ 上的函数 $f(x)$ 展开成正弦级数或余弦级数.

根据前面讨论的结果,这类展开问题可以按如下的方法解决:设函数 $f(x)$ 定义在区间 $[0, \pi]$ 上并且满足收敛定理的条件,我们在开区间 $(-\pi, 0)$ 内补充函数 $f(x)$ 的定义,得到定义在 $(-\pi, \pi)$ 上的函数 $F(x)$,使它在 $(-\pi, \pi)$ 上成为奇函数(\equiv 偶函数).按这种方式拓广函数定义域的过程称为奇延拓(偶延拓).然后将奇延拓(偶延拓)后的函数展开成傅里叶级数,这个级数必定是正弦级数(余弦级数).再限制 x 在 $(0, \pi]$ 上,此时 $F(x) \equiv f(x)$,这样便得到 $f(x)$ 的正弦级数(余弦级数)展开式.

Theorem 17.2.3 设周期为 $2l$ 的周期函数 $f(x)$ 满足收敛定理的条件,则它的傅里叶级数展开式为

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos \frac{n\pi x}{l} + b_n \sin \frac{n\pi x}{l} \right) (x \in C), \quad (8-1)$$

其中

$$\left. \begin{aligned} a_n &= \frac{1}{l} \int_{-l}^l f(x) \cos \frac{n\pi x}{l} dx \quad (n = 0, 1, 2, \dots) \\ b_n &= \frac{1}{l} \int_{-l}^l f(x) \sin \frac{n\pi x}{l} dx \quad (n = 1, 2, 3, \dots), \\ C &= \{x \mid f(x) = \frac{1}{2} [f(x^-) + f(x^+)]\} \end{aligned} \right\}$$

当 $f(x)$ 为奇函数时,

$$f(x) = \sum_{n=1}^{\infty} b_n \sin \frac{n\pi x}{l} \quad (x \in C)$$

其中

$$b_n = \frac{2}{l} \int_0^l f(x) \sin \frac{n\pi x}{l} dx \quad (n = 1, 2, 3, \dots)$$

当 $f(x)$ 为偶函数时,

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos \frac{n\pi x}{l} \quad (x \in C)$$

其中

$$a_n = \frac{2}{l} \int_0^l f(x) \cos \frac{n\pi x}{l} dx \quad (n = 0, 1, 2, \dots)$$

Definition 17.2.4 — 傅里叶级数的复数形式.

$$\sum_{n=-\infty}^{\infty} c_n e^{\frac{n\pi x}{l}} j$$

$$\text{where } c_n = \frac{1}{2l} \int_{-l}^l f(x) e^{-\frac{n\pi x}{l}} j dx \quad (n = 0, \pm 1, \pm 2, \dots).$$

Proof. 设周期为 $2l$ 的周期函数 $f(x)$ 的傅里叶级数为

$$\frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos \frac{n\pi x}{l} + b_n \sin \frac{n\pi x}{l} \right)$$

其中系数 a_n 与 b_n 为

$$\left. \begin{aligned} a_n &= \frac{1}{l} \int_{-l}^l f(x) \cos \frac{n\pi x}{l} dx \quad (n = 0, 1, 2, \dots), \\ b_n &= \frac{1}{l} \int_{-l}^l f(x) \sin \frac{n\pi x}{l} dx \quad (n = 1, 2, 3, \dots). \end{aligned} \right\}$$

利用欧拉公式

$$\cos t = \frac{e^{ti} + e^{-ti}}{2}, \quad \sin t = \frac{e^{ti} - e^{-ti}}{2i}$$

记

$$\frac{a_0}{2} = c_0, \quad \frac{a_n - b_n i}{2} = c_n, \quad \frac{a_n + b_n i}{2} = c_{-n} \quad (n = 1, 2, 3, \dots)$$

则表示为

$$\begin{aligned} c_0 + \sum_{n=1}^{\infty} \left(c_n e^{\frac{n\pi x}{l} i} + c_{-n} e^{-\frac{n\pi x}{l} i} \right) &= \left(c_n e^{\frac{n\pi x}{l} i} \right)_{n=0} + \sum_{n=1}^{\infty} \left(c_n e^{\frac{n\pi x}{l} i} + c_{-n} e^{-\frac{n\pi x}{l} i} \right) \\ c_0 &= \frac{a_0}{2} = \frac{1}{2l} \int_{-l}^l f(x) dx \\ c_n &= \frac{a_n - b_n i}{2} \\ &= \frac{1}{2} \left[\frac{1}{l} \int_{-l}^l f(x) \cos \frac{n\pi x}{l} dx - \frac{i}{l} \int_{-l}^l f(x) \sin \frac{n\pi x}{l} dx \right] \\ &= \frac{1}{2l} \int_{-l}^l f(x) \left(\cos \frac{n\pi x}{l} - i \sin \frac{n\pi x}{l} \right) dx \\ &= \frac{1}{2l} \int_{-l}^l f(x) e^{-\frac{n\pi}{l} x} dx \quad (n = 1, 2, 3, \dots); \\ c_{-n} &= \frac{a_n + b_n i}{2} = \frac{1}{2l} \int_{-l}^l f(x) e^{\frac{n\pi x}{l} i} dx \quad (n = 1, 2, 3, \dots) \end{aligned}$$

将已得的结果合并写为

$$c_n = \frac{1}{2l} \int_{-l}^l f(x) e^{-\frac{n\pi x}{l}} dx \quad (n = 0, \pm 1, \pm 2, \dots)$$

■

17.3 Fourier Transform

Definition 17.3.1 — The Fourier transform of a continuous function $f(t)$ of a continuous variable t .

$$F(\mu) = \mathcal{F}\{f(t)\} = \int_{-\infty}^{\infty} f(t) e^{-j2\pi\mu t} dt$$

Using Euler's formula, we can write as

$$F(\mu) = \int_{-\infty}^{\infty} f(t)[\cos(2\pi\mu t) - j \sin(2\pi\mu t)] dt$$

傅里叶变换将一个复杂的波变换成不同的正弦波（基波），时频图（时域、频域、某一刻相位）。对于任意一个时域的曲线，均可以分解成为不同的基波的

$$\text{时域 } f(t) \Leftrightarrow F(u, v) \text{ 频域}$$

Definition 17.3.2 — inverse Fourier transform.

$$f(t) = \int_{-\infty}^{\infty} F(\mu) e^{j2\pi\mu t} d\mu$$

17.4 Discrete Fourier Transform

Definition 17.4.1 — Discrete Fourier Transform.

$$F_m = \sum_{n=0}^{M-1} f_n e^{-j2\pi mn/M} \quad m = 0, 1, 2, \dots, M-1$$

Definition 17.4.2 — inverse discrete Fourier transform (IDFT).

$$f_n = \frac{1}{M} \sum_{m=0}^{M-1} F_m e^{j2\pi mn/M} \quad n = 0, 1, 2, \dots, M-1$$

Definition 17.4.3 — 离散傅里叶变换矩阵 W .

$$W = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega^{-1} & \omega^{-2} & \cdots & \omega^{-(n-1)} \\ 1 & \omega^{-2} & \omega^{-4} & \cdots & \omega^{-2(n-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \omega^{-(n-1)} & \omega^{-2(n-1)} & \cdots & \omega^{-(n-1)(n-1)} \end{bmatrix}$$

where $\omega = e^{2\pi j/n}$, $j = \sqrt{-1}$

Corollary 17.4.1 矩阵 $(1/\sqrt{n})W$ 是酉矩阵:

$$\frac{1}{n} W^H W = \frac{1}{n} W W^H = I$$

Corollary 17.4.2 W 的逆 $W^{-1} = (1/n)W^H$ 。

Corollary 17.4.3 W 的共轭转置为

$$W^H = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega^1 & \omega^2 & \cdots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(n-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \cdots & \omega^{(n-1)(n-1)} \end{bmatrix}$$

Corollary 17.4.4 Gram 矩阵的第 i, j 个元素为

$$(W^H W)_{ij} = 1 + \omega^{i-j} + \omega^{2(i-j)} + \cdots + \omega^{(n-1)(i-j)}$$

$$(W^H W)_{ii} = n, (W^H W)_{ij} = \frac{\omega^{n(i-j)} - 1}{\omega^{i-j} - 1} = 0 \text{ 如果 } i \neq j$$

最后一步因为 $\omega^n = 1$ 。

Definition 17.4.4 — 离散傅里叶反变换. n 维向量 x 的离散傅里叶反变换是

$$W^{-1}x = (1/n)W^H x$$



18. Factorization of Matrices

18.1 主要的矩阵分解

$$A = LU$$

$$A = LPU$$

$$A = QR$$

$$A = X \Lambda X^{-1}$$

$$S = Q \Lambda Q^T$$

$$A = U \Sigma V^T$$

$$A = CMR$$



Statistics

19	Random Variables	189
19.1	随机变量的引入	
19.2	随机变量的特点	
19.3	优势	
19.4	常见随机变量	
19.5	随机变量的分布函数	
19.6	随机变量的分布函数性质	
19.7	连续型随机变量及其概率密度	
19.8	随机变量的数字特征	
20	大数定律和中心极限定理	194
20.1	辛钦大数定律	
20.2	伯努利大数定理	
20.3	中心极限定理	
21	Sampling Distribution	196
21.1	统计量	
21.2	常见统计量	
21.3	卡方分布	
21.4	t 分布	
21.5	F 分布	
21.6	正态总体的样本均值和样本方差的分布	
22	Hypothesis Testing	203
22.1	正态总体均值方差的检验法	
22.2	经验分布函数	
22.3	Q-Q 图 (Quantile-quantile Plot)	
22.4	卡方拟合优度检验	
22.5	柯尔莫哥洛夫 (Kolmogorov-Smirnov) 检验	
22.6	秩和检验	
22.7	方差分析 (Analysis of Variance, ANOVA)	
22.8	多元线性回归	
22.9	逐步回归	
23	Bootstrap	216
23.1	估计量的标准误差的 Bootstrap 估计	
23.2	估计量的均方误差的 Bootstrap 估计	
23.3	Bootstrap 置信区间	
23.4	参数 Bootstrap 方法	
24	List Of Definitions	219



19. Random Variables

19.1 随机变量的引入

在实际问题中, 随机试验的结果可以用数量来表示, 由此就产生了随机变量的概念.

有些试验结果本身与数值有关 (本身就是一个数); 在有些试验中, 试验结果看来与数值无关, 但我们可以引进一个变量来表示它的各种结果. 也就是说, 把试验结果数值化. 这种对应关系在数学上理解为定义了一种实值单值函数

19.2 随机变量的特点

它随试验结果的不同而取不同的值, 因而在试验之前只知道它可能取值的范围, 而不能预先肯定它将取哪个值.

由于试验结果的出现具有一定的概率, 于是这种实值函数取每个值和每个确定范围内的值也有一定的概率, 所以称这种定义在样本空间 S 上的实值单值函数 $X = X(e)$ 为随机变量

19.3 优势

1. 引入随机变量后, 对随机现象统计规律的研究, 就由对事件及事件概率的研究扩大为对随机变量及其取值规律的研究
2. 易于表示
3. 和原来集合的表示等价

19.4 常见随机变量

19.4.1 离散型随机变量

公式法

$$P \{X = x_k\} = p_k, k = 1, 2, \dots$$

列表法

$$X \sim \begin{pmatrix} x_1 & x_2 & \cdots & x_n & \cdots \\ p_1 & p_2 & \cdots & p_n & \cdots \end{pmatrix}$$

X	x_1	x_2	\cdots	x_n	\cdots
p_k	p_1	p_2	\cdots	p_n	\cdots

19.4.2 两点分布

特殊的二项分布 ($n = 1$)

X	0	1
p_k	$1 - p$	p

19.4.3 (n 重) 伯努利试验, 二项分布

$$P\{X = k\} = \binom{n}{k} p^k (1-p)^{n-k} \quad k = 0, 1, \dots, n$$

X	0	1	\cdots	k	\cdots	n
p_k	q^n	$\binom{n}{1} pq^{n-1}$	\cdots	$\binom{n}{k} p^k q^{n-k}$	\cdots	p^n

二项分布性质

- 将伯努利试验 E 独立地重复地进行 n 次, 则称这一串重复的独立试验为 n 重伯努利试验. 伯努利试验对试验结果没有等可能的要求“重复”是指这 n 次试验中 $P(A) = p$ 保持不变.“独立”是指各次试验的结果互不影响.
- 每次试验条件相同
- 每次试验只考虑两个互逆结果 A 或 \bar{A}
- 各次试验相互独立
- 二项分布描述的是 n 重伯努利试验中事件 A 出现的次数 X 的分布律

二项分布单峰性质

若在 k_0 处, 概率 $P_{X=k}$ 达到最大 (称 k_0 为随机变量 X 的最可能值)

$$\begin{cases} \frac{P\{X=k_0\}}{P\{X=k_0+1\}} \geq 1 \\ \frac{P\{X=k_0\}}{P\{X=k_0-1\}} \geq 1 \end{cases}$$

得

$$(n+1)p - 1 \leq k_0 \leq (n+1)p$$

$$k_0 = \begin{cases} (n+1)p \text{ 和 } (n+1)p - 1, & \text{当 } (n+1)p \text{ 为整数,} \\ [(n+1)p], & \text{其它,} \end{cases}$$

19.4.4 泊松分布

$$P\{X = k\} = \frac{\lambda^k e^{-\lambda}}{k!}, \quad k = 0, 1, 2, \dots$$

其中 $\lambda > 0$ 是常数, 则称 X 服从参数为 λ 的泊松分布, 记作

$$X \sim \pi(\lambda)$$

Theorem 19.4.1 — 泊松定理. 泊松分布是作为二项分布的近似, 于 1837 年由法国数学家泊松引入的.

$$\lim_{n \rightarrow \infty} C_n^k p_n^k (1 - p_n)^{n-k} = e^{-\lambda} \frac{\lambda^k}{k!} \quad (\lambda = np)$$

19.4.5 连续型随机变量

19.4.6 均匀分布

$$f(x) = \begin{cases} \frac{1}{b-a}, & a < x < b \\ 0, & \text{其它} \end{cases}$$

$$X \sim U(a, b)$$

$$F(x) = P\{X \leq x\} = \begin{cases} 0, & x < a \\ \frac{x-a}{b-a}, & a \leq x < b \\ 1, & x \geq b \end{cases}$$

19.4.7 指数分布

$$f(x) = \begin{cases} \frac{1}{\theta} e^{-x/\theta}, & x > 0 \\ 0, & \text{其它,} \end{cases}$$

$$F(x) = P\{X \leq x\} = \begin{cases} 1 - e^{-x/\theta}, & x > 0 \\ 0, & \text{其它} \end{cases}$$

19.4.8 正态分布

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad -\infty < x < \infty$$

19.4.9 标准正态分布

$$\text{若 } X \sim N(\mu, \sigma^2), \text{ 则 } Z = \frac{X - \mu}{\sigma} \sim N(0, 1)$$

19.5 随机变量的分布函数

19.6 随机变量的分布函数性质

对任意实数 $x_1 < x_2$, 随机点落在区间 $(x_1, x_2]$ 内的概率为:

$$P\{x_1 < X \leq x_2\} = P\{X \leq x_2\} - P\{X \leq x_1\} = F(x_2) - F(x_1)$$

$$\begin{aligned} F(-\infty) &= \lim_{x \rightarrow -\infty} F(x) = 0 \\ F(+\infty) &= \lim_{x \rightarrow +\infty} F(x) = 1 \end{aligned}$$

19.7 连续型随机变量及其概率密度

$$f(x) \geq 0$$

$$\int_{-\infty}^{\infty} f(x) dx = 1$$

19.8 随机变量的数字特征

19.8.1 方差

$$D(X) = E\{[x - E(X)]^2\}$$

$$D(X) = E(X^2) - [E(X)]^2$$

方差性质

$$D(C) = 0$$

$$D(CX) = C^2 D(X)$$

$$D(X + C) = D(X)$$

X 和 Y 相互独立时,

$$D(X + Y) = D(X) + D(Y)$$

$D(X) = 0$ 等价于 $P[X = E(X)] = 1$

Theorem 19.8.1 — 切比雪夫不等式.

$$P(|X - \mu| \geq \varepsilon) \leq \frac{\sigma^2}{\varepsilon^2}$$

$$P(|X - \mu| < \varepsilon) \geq 1 - \frac{\sigma^2}{\varepsilon^2}$$

19.8.2 协方差

$$Cov(X, Y) = E\{[X - E(X)][Y - E(Y)]\}$$

$$Cov(X, Y) = E(XY) - E(X)E(Y)$$

协方差性质

$$D(X + Y) = D(X) + D(Y) + 2Cov(X, Y)$$

$$Cov(X, Y) = Cov(Y, X)$$

$$Cov(X, X) = D(X)$$

$$Cov(X, c) = 0$$

$$Cov(aX + b, Y) = aCov(X, Y)$$

$$Cov(aX, bY) = abCov(X, Y)$$

$$Cov(X_1 + X_2, Y) = Cov(X_1, Y) + Cov(X_2, Y)$$

对于随机变量序列 X_1, \dots, X_n 与 Y_1, \dots, Y_m , 有

$$\text{cov} \left(\sum_{i=1}^n X_i, \sum_{j=1}^m Y_j \right) = \sum_{i=1}^n \sum_{j=1}^m \text{cov}(X_i, Y_j)$$

对于随机变量序列 X_1, \dots, X_n , 有

$$\text{var} \left(\sum_{i=1}^n X_i \right) = \sum_{i=1}^n \text{var}(X_i) + 2 \sum_{i,j:i < j} \text{cov}(X_i, X_j)$$

19.8.3 相关系数

$$\rho_{XY} = \frac{\text{Cov}(X, Y)}{\sqrt{D(X)}\sqrt{D(Y)}}$$

相关系数是刻画两个变量间线性相关程度的一个重要的数字特征. 相关系数也可以看成协方差：一种剔除了两个变量量纲影响、标准化后的特殊协方差.

19.8.4 X, Y 不相关时候的性质

若 $\rho_{XY} = 0$, 称 X 和 Y 不(线性)相关.

Corollary 19.8.2 独立一定是不相关, 不相关不一定独立.

Theorem 19.8.3 若随机变量 X 与 Y 的方差都存在, 且均不为零; 则下列四个命题等价.

1. $\rho_{XY} = 0$
2. $\text{Cov}(X, Y) = 0$
3. $E(XY) = E(X)E(Y)$
4. $D(X \pm Y) = DX + DY$.

20. 大数定律和中心极限定理

20.1 辛钦大数定律

Theorem 20.1.1 — 辛钦大数定律. 设随机变量序列 X_1, X_2, \dots 相互独立, 服从同一分布, 具有数学期 $E(X_i) = \mu, i = 1, 2, \dots$, 则对于任意正数 ε , 有

$$\lim_{n \rightarrow \infty} P \left\{ \left| \frac{1}{n} \sum_{i=1}^n X_i - \mu \right| < \varepsilon \right\} = 1$$

则序列 $\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$ 依概率收敛于 μ (很可能接近于 μ) .

另一表述:

$$\bar{X} \xrightarrow{P} \mu$$

20.1.1 辛钦大数定律条件

- X_1, X_2, \dots 相互独立
- X_1, X_2, \dots 服从同一分布
- 不要求方差存在.

Corollary 20.1.2 设 $X_n \xrightarrow{P} a, Y_n \xrightarrow{P} b$ 则

$$g(X_n, Y_n) \xrightarrow{P} g(a, b)$$

20.2 伯努利大数定理

Theorem 20.2.1 — 伯努利大数定理. 设 f_A 是 n 次独立重复试验中事件 A 的发生次数, p 是每次试验中发生的概率, 则对于任意的正数 ε .

$$\lim_{n \rightarrow \infty} \{ |f_A/n - p| < \varepsilon \} = 1$$

贝努里大数定律表明, 当重复试验次数 n 充分大时, 事件 A 发生的频率 nA/n 与事件 A 的概率 p 有较大偏差的概率很小.

20.3 中心极限定理

Theorem 20.3.1 — Lyapunov 中心极限定理. 设 X_1, X_2, \dots 相互独立, 他们拥有数学期望和方差

$$E(X_k) = \mu_k$$

$$D(X_k) = \sigma_k^2$$

则有

$$\begin{aligned} \lim_{n \rightarrow \infty} F_n(x) &= \lim_{n \rightarrow \infty} P \left\{ \frac{\sum_{k=1}^n X_k - \sum_{k=1}^n \mu_k}{B_n} \leq x \right\} \\ &= \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt \\ &= \Phi(x) \end{aligned}$$

$$\text{其中 } B_n = \sqrt{D(\sum_{k=1}^n X_k)}$$

Theorem 20.3.2 设 X_1, X_2, \dots 相互独立, 他们服从同一分布, 拥有数学期望和方差

$$E(X_k) = \mu$$

$$D(X_k) = \sigma^2$$

则有

$$\begin{aligned} \lim_{n \rightarrow \infty} F_n(x) &= \lim_{n \rightarrow \infty} P \left\{ \frac{\sum_{i=1}^n X_i - n\mu}{\sigma\sqrt{n}} \leq x \right\} \\ &= \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt \\ &= \Phi(x) \end{aligned}$$

Theorem 20.3.3 — De Moivre-Laplace 定理. 设 η_n 服从参数为 n, p 的二项分布, 则对于任意 x , 有

$$\lim_{n \rightarrow \infty} P \left\{ \frac{\eta_n - np}{\sqrt{np(1-p)}} \leq x \right\} = \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt = \Phi(x)$$

Proof. 将 η_n 拆分成 n 个相互独立, 服从同一分布的随机变量 X_1, X_2, \dots 之和

■



21. Sampling Distribution

21.1 统计量

不含任何未知参数的样本的函数称为统计量. 它是完全由样本决定的量.

21.2 常见统计量

Definition 21.2.1 — 样本平均值.

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

Definition 21.2.2 — 样本方差.

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2$$

Definition 21.2.3 — 样本标准差.

$$S = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2}$$

Definition 21.2.4 — 样本 k 阶原点矩.

$$A_k = \frac{1}{n} \sum_{i=1}^n X_i^k$$

Definition 21.2.5 — 样本 k 阶中心矩.

$$B_k = \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X})^k$$

Definition 21.2.6 — X 和 Y 的 $k+p$ 阶混合原点矩. 若

$$E \left\{ (X^k) (Y^p) \right\}, k, p = 1, 2, \dots$$

存在, 则称它为 X 和 Y 的 $k+p$ 阶混合原点矩

Definition 21.2.7 — X 和 Y 的 $k+l$ 阶混合中心矩. 若

$$E \left\{ [X - E(X)]^k [Y - E(Y)]^l \right\}, k, l = 1, 2, \dots$$

存在, 则称它为 X 和 Y 的 $k+l$ 阶混合中心矩

21.3 卡方分布

Definition 21.3.1 — 卡方分布. 设 X_1, X_2, \dots, X_n 相互独立, 都服从正态分布 $N(0, 1)$ (都是来自总体 $N(0, 1)$ 的样本), 则称随机变量:

$$\chi^2 = X_1^2 + X_2^2 + \dots + X_n^2$$

所服从的分布为自由度为 n 的 χ^2 分布. 记作 $\chi^2 \sim \chi^2(n)$

21.3.1 卡方分布性质

Corollary 21.3.1 $\chi^2(1) \sim X^2(1)$

Corollary 21.3.2 — χ^2 分布的可加性. 若 $X_1 \sim \chi^2(n_1), X_2 \sim \chi^2(n_2)$, 且 X_1 与 X_2 相互独立, 则

$$X_1 + X_2 \sim \chi^2(n_1 + n_2)$$

Proof. 事实上, 卡方分布是 Gamma 分布的特殊情况. 自由度为 n 的卡方分布 χ_n^2 其实就是 $\Gamma(\frac{n}{2}, \frac{1}{2})$.

根据 Gamma 分布的矩生成函数 (Moment Generating Function, MGF), 若 $X_1 \sim \chi_{n_1}^2, X_2 \sim \chi_{n_2}^2$, 那么 X_1, X_2 对应的 MGF 分别为

$$M_{X_1}(t) = (1 - 2t)^{-n_1/2}$$

$$M_{X_2}(t) = (1 - 2t)^{-n_2/2}$$

因为 X_1, X_2 相互独立, 那么 $Y = X_1 + X_2$ 的 MGF 为

$$M_Y(t) = M_{X_1}(t)M_{X_2}(t) = (1 - 2t)^{-(n_1+n_2)/2}$$

由 MGF 的唯一性,

$$Y \sim \Gamma \left(\frac{n_1 + n_2}{2}, \frac{1}{2} \right) = \chi_{n_1+n_2}^2$$

■

Corollary 21.3.3 — χ^2 分布的数学期望和方差.

$$E(\chi^2) = n$$

$$D(\chi^2) = 2n$$

Proof.

$$E(X) = 0, D(X) = 1, E(X_i^2) = 1, D(X_i^2) = 2$$

■

Corollary 21.3.4 — 卡方分布中心极限定理. 设 X_1, X_2, \dots, X_n 相互独立, 都服从正态分布 $N(0, 1)$, $E(X) = \mu$, $D(X) = \sigma^2$, 则有

$$P\left(\frac{\chi^2(n) - n}{\sqrt{2n}} \leq x\right) \sim \Phi(x)$$

Proof.

$$E(X) = 0, D(X) = 1, E(X_i^2) = 1, D(X_i^2) = 2$$

■

Corollary 21.3.5

$$\frac{\chi^2(n)}{n} \leftarrow \frac{1}{n} \sum_{i=1}^n X_i^2 = 1$$

Proof.

$$E(X_i^2) = 1$$

■

Definition 21.3.2 — 卡方分布的上分位点.

$$p(\chi^2 > \chi_\alpha^2(n)) = \alpha$$

21.4 t 分布

Definition 21.4.1 — 自由度为 t 的 t 分布. 设 $X \sim N(0, 1)$, $Y \sim \chi^2(n)$, 且 X, Y 相互独立, 则称随机变量

$$t = \frac{X}{\sqrt{Y/n}}$$

为自由度为 n 的 t 分布, 记作 $t \sim t(n)$

21.4.1 t 分布性质

Corollary 21.4.1 — t 分布数学期望和方差.

$$E(t) = 0$$

$$D(t(n)) = \frac{n}{n-2}$$

Corollary 21.4.2 — t 分布的概率密度函数. $n = 1$ 时

$$f(t) = \frac{1}{\pi(1+t^2)} \text{(柯西密度)}$$

数学期望不存在. $n > 1$ 时, $h(t)$ 的图形关于 $t = 0$ 对称.

Corollary 21.4.3 n 足够大时 t 分布近似于 $N(0, 1)$ 分布. 由于卡方分布

$$n \rightarrow \infty \text{ 时}, \frac{\chi^2(n)}{n} \rightarrow 1$$

所以

$$t(n) = \frac{N(0, 1)}{\sqrt{\frac{\chi^2(n)}{n}}} \rightarrow n \rightarrow \infty N(0, 1)$$

Corollary 21.4.4 — 与 F 分布的关系.

$$\begin{aligned} t^2(n) &= \frac{N(0, 1)^2}{\chi^2(n)/n} \\ &= \frac{\chi^2(1)/1}{\chi^2(n)/n} \sim F(1, n) \\ \frac{1}{t^2(n)} &\sim F(n, 1) \end{aligned}$$

Definition 21.4.2 — t 分布的上分位点.

$$t_{1-\alpha}(n) = -t_\alpha(n)$$

21.5 F 分布

Definition 21.5.1 — F 分布. 设 $U \sim \chi^2(n_1), V \sim \chi^2(n_2)$, 且 U, V 相互独立, 则称随机变量

$$F = \frac{U/n_1}{V/n_2}$$

服从自由度为 n_1, n_2 的 F 分布. 记作 $F \sim F(n_1, n_2)$

21.5.1 F 分布性质

Corollary 21.5.1 — F 分布数学期望.

$$E(F) = \frac{n_2}{n_2 - 2}$$

即与 n_1 无关.

Corollary 21.5.2 — $F(n_1, n_2), F(n_2, n_1)$ 的关系.

$$\frac{1}{F(n_1, n_2)} \sim F(n_2, n_1)$$

Corollary 21.5.3 — 与 t 分布的关系.

$$\begin{aligned} t^2(n) &= \frac{N(0, 1)^2}{\chi^2(n)/n} \\ &= \frac{\chi^2(1)/1}{\chi^2(n)/n} \sim F(1, n) \\ \frac{1}{t^2(n)} &\sim F(n, 1) \end{aligned}$$

Corollary 21.5.4 — F 分布上 分位点的性质.

$$F_{1-\alpha}(n_1, n_2) = \frac{1}{F_\alpha(n_2, n_1)}$$

21.6 正态总体的样本均值和样本方差的分布

Definition 21.6.1 — 正态总体的样本均值的数学期望、方差和样本方差的数学期望. 设总体 $X \sim N(\mu, \sigma^2)$ 的均值为 μ , 方差为 σ^2 .

X_1, X_2, \dots, X_n 是来自 X 的一个样本, 样本均值是 \bar{X} , 样本方差是 S^2 , 则有

$$E(\bar{X}) = \mu$$

$$D(\bar{X}) = \frac{\sigma^2}{n}$$

$$E(S^2) = \sigma^2$$

Corollary 21.6.1 — 矩估计法原理.

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i \rightarrow E(X)$$

$$\frac{1}{n} \sum X_i^2 - \bar{X}^2 \rightarrow D(X) = E(X^2) - E(X)^2$$

Theorem 21.6.2 — 样本均值的分布. 设总体 $X \sim N(\mu, \sigma^2)$ 的均值为 μ , 方差为 σ^2 , X_1, X_2, \dots, X_n 是来自 X 的一个样本, 样本均值是 \bar{X} , 则有

$$\bar{X} \sim N(\mu, \sigma^2/n)$$

Proof.

$$\sum_{i=1}^n X_i = X_1 + X_2 + \dots \sim N(n\mu, n\sigma^2)$$

■

Corollary 21.6.3

$$\frac{\bar{X} - \mu}{\sigma^2/n} = \frac{\sqrt{n}(\bar{X} - \mu)}{\sigma} \sim N(0, 1)$$

Theorem 21.6.4 — 样本方差的分布. 设总体 $X \sim N(\mu, \sigma^2)$ 的均值为 μ , 方差为 σ^2 , X_1, X_2, \dots, X_n 是来自 X 的一个样本, 样本均值是 \bar{X} , 样本方差是 S^2 , 则有

$$\frac{(n-1)S^2}{\sigma^2} \sim \chi^2(n-1)$$

而且 \bar{X} 与 S^2 相互独立.

Theorem 21.6.5 — 样本均值和样本方差的关系. 设 X_1, X_2, \dots, X_n 是总体 $X \sim N(\mu, \sigma^2)$ 的样本, 样本均值是 \bar{X} , 样本方差是 S^2 , 则有

$$\frac{\bar{X} - \mu}{S/\sqrt{n}} = \frac{\sqrt{n}(\bar{X} - \mu)}{\sigma} \sim t(n-1)$$

对比

$$\frac{\bar{X} - \mu}{\sigma^2/n} = \frac{\sqrt{n}(\bar{X} - \mu)}{\sigma} \sim N(0, 1)$$

但是 n 很大时

$$S^2 = \frac{1}{n} \sum_{i=1}^n X_i - (\bar{X})^2$$

$$S^2 \rightarrow E(X^2) - E(X)^2 = \sigma^2$$

$$\therefore S \rightarrow \sigma$$

Theorem 21.6.6 — 两总体样本均值差、样本方差比的分布. 设 $X \sim N(\mu_1, \sigma_1^2)$, $Y \sim N(\mu_2, \sigma_2^2)$, 且 X 与 Y 独立,

X_1, X_2, \dots, X_n 是来自 X 的样本, Y_1, Y_2, \dots, Y_{n_2} 是取自 Y 的样本,

\bar{X} 和 \bar{Y} 分别是这两个样本的样本均值, S_1^2 和 S_2^2 分别是这两个样本的样本方差, 则有

1. $\frac{S_1^2/S_2^2}{\sigma_1^2/\sigma_2^2}$ 的分布

$$\frac{S_1^2/S_2^2}{\sigma_1^2/\sigma_2^2} \sim F(n_1 - 1, n_2 - 1)$$

2.

$$\frac{\bar{X} - \bar{Y} - (\mu_1 - \mu_2)}{\sqrt{\frac{(n_1-1)S_1^2 + (n_2-1)S_2^2}{\sigma^2(n_1+n_2-2)}}} \sim t(n_1 + n_2 - 2)$$

Proof. 1. $\frac{S_1^2/S_2^2}{\sigma_1^2/\sigma_2^2}$ 的分布

$$\frac{S_1^2/S_2^2}{\sigma_1^2/\sigma_2^2} \sim F(n_1 - 1, n_2 - 1)$$

当 $\sigma_1^2 = \sigma_2^2 = \sigma^2$ 时

$$\bar{X} - \bar{Y} = N(\mu_1 - \mu_2, \sigma^2/n_1 + \sigma^2/n_2)$$

$$\therefore U = \frac{\bar{X} - \bar{Y} - (\mu_1 - \mu_2)}{\sigma \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} \sim N(0, 1)$$

又因为

$$\frac{n_1 - 1}{\sigma^2} S_1^2 \sim \chi^2(n_1 - 1)$$

$$\frac{n_2 - 1}{\sigma^2} S_2^2 \sim \chi^2(n_2 - 1)$$

$$\therefore V = \frac{n_1 - 1}{\sigma^2} S_1^2 + \frac{n_2 - 1}{\sigma^2} S_2^2 \sim \chi^2(n_1 + n_2 - 2)$$

2 和 3 相互独立, 根据 t 分布定义, 因此有

$$\frac{\bar{X} - \bar{Y} - (\mu_1 - \mu_2)}{S_w \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} \sim t(n_1 + n_2 - 2)$$

$$\text{代入 } S_w^2 = \frac{(n_1-1)S_1^2 + (n_2-1)S_2^2}{(n_1+n_2-2)}$$

$$\frac{\bar{X} - \bar{Y} - (\mu_1 - \mu_2)}{\sqrt{\frac{(n_1-1)S_1^2 + (n_2-1)S_2^2}{\sigma^2(n_1+n_2-2)}}} \sim t(n_1 + n_2 - 2)$$

■

22. Hypothesis Testing

22.1 正态总体均值方差的检验法

假设显著性水平为 α . 参阅22.1.

22.2 经验分布函数

设 X_1, X_2, \dots, X_n 是总体 F 的一个样本, 用 $S(x)$ ($-\infty < x < \infty$) 表示 X_1, X_2, \dots, X_n 中不大于 x 的随机变量

$$F_n(x) = \frac{1}{n} S(x), -\infty < x < \infty.$$

容易得到的 ($F_n(x)$ 的观察值仍以 $F_n(x)$ 表示). 一般地, 设 x_1, x_2, \dots, x_n 是总体 F 的一个容量为 n 的样本值先将 x_1, x_2, \dots, x_n 按从小到大的次序排列, 并重新编号.

设为

$$x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(n)}$$

则经验分布函数 $F_n(x)$ 的观察值为

$$F_n(x) = \begin{cases} 0, & \text{若 } x < x_{(1)} \\ \frac{k}{n}, & \text{若 } x_{(k)} \leq x < x_{(k+1)} \\ 1, & \text{若 } x \geq x_{(n)} \end{cases}$$

经验分布函数的任一个观察值 $F_n(x)$ 与总体分布函数 $F(x)$ 只有微小的差别, 从而在实际上可当作 $F(x)$ 来使用.

22.3 Q-Q 图 (Quantile-quantile Plot)

Q-Q 图是 Quantile-Quantile Plot 的简称, 是检验拟合优度的好方法, 目前在国外被广泛使用, 它的图示方法简单直观, 易于使用.

现在我们希望知道观测数据与分布模型的拟合效果如何. 如果拟合效果好, 观测数据的经验分布就应当非常接近分布模型的理论分布, 而经验分布函数的分位数自然也应当与分布模型的理论分位数近似相等.

Table 22.1: 正态总体均值方差的检验法

	原假设 H_0	检验统计量	备择假设 H_1	拒绝域
1	$\mu \leq \mu_0, \mu \geq \mu_0, \mu = \mu_0 (\sigma^2 \text{ 已知})$	$Z = \frac{\bar{X} - \mu_0}{\sigma/\sqrt{n}}$	$\mu > \mu_0, \mu < \mu_0, \mu \neq \mu_0$	$z \geq z_\alpha, z \leq -z_\alpha, z \geq z_{\alpha/2}$
2	$\mu \leq \mu_0, \mu \geq \mu_0, \mu = \mu_0 (\sigma^2 \text{ 未知})$	$t = \frac{\bar{X} - \mu_0}{S/\sqrt{n}}$	$\mu > \mu_0, \mu < \mu_0, \mu \neq \mu_0$	$t \geq t_\alpha(n-1), t \leq -t_\alpha(n-1), t \geq t_{\alpha/2}(n-1)$
3	$\mu_1 - \mu_2 \leq \delta, \mu_1 - \mu_2 \geq \delta, \mu_1 - \mu_2 = \delta (\sigma_1^2, \sigma_2^2 \text{ 已知})$	$Z = \frac{\bar{X} - \bar{Y} - \delta}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$	$\mu_1 - \mu_2 > \delta, \mu_1 - \mu_2 < \delta, \mu_1 - \mu_2 \neq \delta$	$z \geq z_\alpha, z \leq -z_\alpha, z \geq z_{\alpha/2}$
4	$\mu_1 - \mu_2 \leq \delta, \mu_1 - \mu_2 \geq \delta, \mu_1 - \mu_2 = \delta (\sigma_1^2 = \sigma_2^2 = \sigma^2 \text{ 未知})$	$t = \frac{\bar{X} - \bar{Y} - \delta}{S_w \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}, S_w^2 = \frac{(n_1-1)S_1^2 + (n_2-2)S_2^2}{n_1+n_2-2}$	$\mu_1 - \mu_2 > \delta, \mu_1 - \mu_2 < \delta, \mu_1 - \mu_2 \neq \delta$	$t \geq t_\alpha(n_1+n_2-2), t \leq -t_\alpha(n_1+n_2-2), t \geq t_{\alpha/2}(n_1+n_2-1)$
5	$\sigma^2 \leq \sigma_0^2, \sigma^2 \geq \sigma_0^2, \sigma^2 = \sigma_0^2 (\mu \text{ 未知})$	$\chi^2 = \frac{(n-1)S^2}{\sigma_0^2}$	$\sigma^2 > \sigma_0^2, \sigma^2 < \sigma_0^2, \sigma^2 \neq \sigma_0^2$	$\chi^2 \geq \chi_\alpha^2(n-1), \chi^2 \leq \chi_{1-\alpha}^2(n-1), \chi^2 \geq \chi_{\alpha/2}^2(n-1) \text{ 或 } \chi^2 \leq \chi_{1-\alpha/2}^2(n-1)$
6	$\sigma_1^2 \leq \sigma_2^2, \sigma_1^2 \geq \sigma_2^2, \sigma_1^2 = \sigma_2^2 (\mu_1, \mu_2 \text{ 未知})$	$F = \frac{S_1^2}{S_2^2}$	$\sigma_1^2 > \sigma_2^2, \sigma_1^2 < \sigma_2^2, \sigma_1^2 \neq \sigma_2^2$	$F \geq F_\alpha(n_1-1, n_2-1), F \leq F_{1-\alpha}(n_1-1, n_2-1), F \geq F_{\alpha/2}(n_1-1, n_2-1) \text{ 或 } F \geq F_{1-\alpha/2}(n_1-1, n_2-1)$
7	$\mu_D \leq 0, \mu_D \geq 0, \mu_D = 0 \text{ (成对数据)}$	$t = \frac{\bar{D}-0}{S_D/\sqrt{n}}$	$\mu_D > 0, \mu_D < 0, \mu_D \neq 0$	$t \geq t_\alpha(n-1), t \leq -t_\alpha(n-1), t \geq t_{\alpha/2}(n-1)$

Algorithm 24: 作 Q-Q 图

Input: 观测数据 x_1, x_2, \dots, x_n

1 将 x_1, x_2, \dots, x_n 依大小顺序排列成: $x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(n)}$

2 取 $y_i = F^{-1}((i-1/2)/n), i = 1, 2, \dots, n$

3 将 $(y_i, x_{(i)}), i = 1, 2, \dots, n$, 这 n 个点画在直角坐标图上

4 如果这 n 个点看起来呈一条 45° 角的直线, 从 $(0, 0)$ 到 $(1, 1)$ 分布, 我们就相信 x_1, x_2, \dots, x_n 拟合分布 $F(x)$ 的效果很好.

22.4 卡方拟合优度检验

可按照下面的五个步骤进行检验：

Algorithm 25: χ^2 拟合优度检验

- 1 建立待检假设 H_0 : 总体 X 的分布函数为 $F(x)$
- 2 在数轴上选取 $k - 1$ 个分点 t_1, t_2, \dots, t_{k-1} , 将数轴分成 k 个区间:
 $(-\infty, t_1), [t_1, t_2), \dots, [t_{k-2}, t_{k-1}), [t_{k-1}, +\infty)$, 令 p_i 为分布函数 $F(x)$ 的总体 X 在第 i 个区间内取值的概率, 设 m_i 为 n 个样本观察值中落入第 i 个区间上的个数, 也称为组频数
- 3 选取统计量 $\chi^2 = \sum_{i=1}^k \frac{(m_i - np_i)^2}{np_i} = \sum_{i=1}^k \frac{m_i^2}{np_i} - n$, 如果 H_0 为真, 则 $\chi^2 \sim \chi^2(k - 1 - r)$, 其中 r 为分布函数 $F(x)$ 中未知参数的个数
- 4 对于给定的显著性水平 α , 确定 χ^2_α , 使其满足 $P\{\chi^2(k - 1 - r) > \chi^2_\alpha\} = \alpha$ 。
- 5 依据样本计算统计量 χ^2 的观察值, 作出判为总体 X 的分布函数为 $F(x)$

22.5 柯尔莫哥洛夫 (Kolmogorov-Smirnov) 检验

χ^2 拟合优度检验实际上是检验 $p_i = F_0(a_i) - F_0(a_{i-1}) = p_{i0}$ ($i = 1, 2, \dots, k$) 的正确性, 并未直接检验原假设的分布函数 $F_0(x)$ 的正确性, 柯尔莫哥洛夫检验直接针对原假设 $H_0 : F(x) = F_0(x)$, 这里分布函数 $F_0(x)$ 必须是连续型分布. 柯尔莫哥洛夫检验基于经验分布函数 (或称样本分布函数) 作为检验统计量, 检验理论分布函数与样本分布函数的拟合优度.

设总体 X 服从连续分布, X_1, X_2, \dots, X_n 是来自总体 X 的简单随机样本, F_n 为经验分布函数. 当 H_0 为真时, 根据大数定律, 当 n 趋于无穷大时, 经验分布函数 $F_n(x)$ 依概率收敛总体分布函数 $F_0(x)$. 定义 $F_n(x)$ 到 $F_0(x)$ 的距离为

$$D_n = \sup_{-\infty < x < +\infty} |F_n(x) - F_0(x)|$$

, 当 n 趋于无穷大时, D_n 依概率收敛到 0.

Theorem 22.5.1 — Kolmogorov 定理. 在 $F_0(x)$ 为连续分布的假定下, 当原假设为真时, $\sqrt{n}D_n$ 的极限分布为

$$\lim_{n \rightarrow \infty} P\{\sqrt{n}D_n \leq t\} = 1 - 2 \sum_{i=1}^{\infty} (-1)^{i-1} e^{-2i^2 t^2}, t > 0$$

在显著性水平 α 下, 一个合理的检验是: 如果 $\sqrt{n}D_n > k$, 则拒绝原假设, 其中 k 是合适的常数.

注: 对于固定的 α 值, 我们需要知道该 α 值下检验的临界值. 常用的是在统计量为 D_n 时, 各个 α 值所对应的临界值如下: 在 $\alpha = 0.1$ 的显著性水平下, 检验的临界值是 $1.22/\sqrt{n}$; 在 $\alpha = 0.05$ 的显著性水平下, 检验的临界值是 $1.36/\sqrt{n}$; 在 $\alpha = 0.01$ 的显著性水平下, 检验的临界值是 $1.63/\sqrt{n}$. 这里 n 为样本的个数. 当由样本计算出来的 D_n 值小于临界值时, 说明不能拒绝原假设, 所假设的分布是可以接受的; 当由样本计算出来的 D_n 值大于临界值时, 拒绝原假设, 即所假设的分布是不能接受的.

Algorithm 26: 柯尔莫哥洛夫检验

- 1 (1) 原假设和备择假设

$$H_0 : F(x) = F_0(x), H_1 : F(x) \neq F_0(x)$$

- 2 (2) 选取检验统计量

$$D_n = \sup_{-\infty < x < +\infty} |F_n(x) - F_0(x)|$$

, 当 H_0 为真时, D_n 有偏小趋势, 则拟合得越好; 当 H_0 不真时, D_n 有偏大趋势, 则拟合得越差.

- 3 (3) 确定拒绝域给定显著性水平 α , 查 D_n 极限分布表, 求出 t_α 满足
 $P\sqrt{n}D_n \geq t_\alpha = \alpha$, 作为临界值, 即拒绝域为 $[t_\alpha, +\infty)$.
- 4 (4) 作判断计算统计量的观察值, 如果检验统计量 $\sqrt{n}D_n$ 的观察值落在拒绝域中, 则拒绝原假设, 否则不拒绝原假设.

Algorithm 27: 秩和检验

Input: 设分别从 X 、 Y 两总体中独立抽取大小为 n_1 和 n_2 的样本, 设 $n_1 \leq n_2$

- 1 将两个样本混合起来, 按照数值大小统一编序由小到大, 每个数据对应的序数称为秩.
- 2 计算取自总体 X 的样本所对应的秩之和, 用 T 表示
- 3 根据 n_1, n_2 与水平 α , 查秩和检验表, 得秩和下限 T_1 与上限 T_2
- 4 两总体分布有显著差异. 否则认为 X 、 Y 两总体分布在水平 α 下无显著差异

Table 22.2: 所有试验数据

A_1	X_{11}	X_{12}	\dots	X_{1n_1}
A_2	X_{21}	X_{22}	\dots	X_{2n_2}
\dots	\dots	\dots		\dots
\ddot{A}_r	X_{r1}	X_{r2}	\dots	X_{rn_r}

22.6 秩和检验

秩和检验的依据是, 如果两总体分布无显著差异, 那么 T 不应太大或太小, 以 T_1 和 T_2 为上、下界的话, 则 T 应在这两者之间, 如果 T 太大或太小, 则认为两总体的分布有显著差异.

22.7 方差分析 (Analysis of Variance, ANOVA)

在现实问题中, 经常会遇到类似考察两台机床生产的零件尺寸是否相等, 病人和正常人的某个生理指标是否一样, 采用两种不同的治疗方案对同一类病人的治疗效果比较等问题. 这类问题通常会归纳为检验两个不同总体的均值是否相等, 对这类问题的解决可以采用两个总体的均值检验方法. 但若检验总体多于两个, 仍采用多总体均值检验方法会遇到困难.

22.7.1 单因素方差分析

只考虑一个因素 A 所关心的指标的影响, A 取几个水平在每个水平上作若干个试验, 假定试验过程中除因素自身外其他影响指标的因素都保持不变 (只有随机因素存在). 我们的任务是从试验结果推断, 因素 A 对指标有无显著影响, 即当 A 取不同水平时指标有无显著差异 A 取某个水平下的指标视为随机变量, 判断 A 取不同水平时指标有无显著差别, 相当于检验若干总体的均值是否相等.

不妨设 A 取 r 个水平, 分别记为 A_1, A_2, \dots, A_r . 若在水平 A_i 下总体 $X_i \sim N(\mu_i, \sigma^2)$, $i = 1, 2, \dots, r$, 这里 μ_i, σ^2 未知, μ_i 可以互不相同, 但假定 X_i 有相同的方差. 设在水平 A_i 下作了 n_i 次独立试验, 即从总体 X_i 中抽取样本容量为 n_i 的样本, 记作

$$X_{ij}, j = 1, 2, \dots, n_i$$

其中, $X_{ij} \sim N(\mu_i, \sigma^2)$, $i = 1, 2, \dots, r, j = 1, 2, \dots, n_i$, 且相互独立.

将所有试验数据列成表格

表 table 22.2 中对应 A_i 行的数据称为第 i 组数据. 判断 A 的 r 个水平对指标有无显著影响, 相当于作以下的假设检验:

原假设 $H_0: \mu_1 = \mu_2 = \dots = \mu_r$; 备择假设 $H_1: \mu_1, \mu_2, \dots, \mu_r$ 不全相等.

由于 X_{ij} 的取值既受不同水平 A_i 的影响, 又受 A_i 固定下随机因素的影响, 所以将它分解为

$$X_{ij} = \mu_i + \varepsilon_{ij}, i = 1, 2, \dots, r, j = 1, 2, \dots, n_i$$

其中 $\varepsilon_{ij} \sim N(0, \sigma^2)$, 且相互独立. 引入记号

$$\mu = \frac{1}{n} \sum_{i=1}^r n_i \mu_i, \quad n = \sum_{i=1}^r n_i, \quad a_i = \mu_i - \mu, \quad i = 1, \dots, r$$

称 μ 为总均值, a_i 是水平 A_i 下总体的平均值 μ_i 与总评均值 μ 的差异, 习惯上称为指标 A_i 的效应.

Definition 22.7.1 原假设 $H_0 : \mu_1 = \mu_2 = \cdots = \mu_r$; 备择假设 $H_1 : \mu_1, \mu_2, \dots, \mu_r$ 不全相等.

为检验 H_0 , 给定显著性水平 α , 记

$$F = \frac{S_A/(r-1)}{S_E/(n-r)} \sim F(r-1, n-r)$$

分布的上 α 分位数为 $F_\alpha(r-1, n-r)$, 检验规则为

$F < F_\alpha(r-1, n-r)$ 时接受 H_0 , 否则拒绝.

Proof. 由 section 22.7.1 式, 模型可表为

$$\begin{cases} X_{ij} = \mu + \alpha_i + \varepsilon_{ij} \\ \sum_{i=1}^r n_i \alpha_i = 0 \\ \varepsilon_{ij} \sim N(0, \sigma^2), i = 1, \dots, r, j = 1, \dots, n_i \end{cases}$$

原假设是

$$H_0 : \alpha_1 = \alpha_2 = \cdots = \alpha_r = 0$$

记

$$\bar{X}_{i\cdot} = \frac{1}{n_i} \sum_{j=1}^{n_i} X_{ij}, \bar{X} = \frac{1}{n} \sum_{i=1}^r \sum_{j=1}^{n_i} X_{ij}$$

$\bar{X}_{i\cdot}$ 是第 i 组数据的组平均值, \bar{X} 是全体数据的总平均值. 考察全体数据对 \bar{X} 的偏差平方和

$$S_T = \sum_{i=1}^r \sum_{j=1}^{n_i} (X_{ij} - \bar{X})^2$$

$$S_T = \sum_{i=1}^r n_i (\bar{X}_{i\cdot} - \bar{X})^2 + \sum_{i=1}^r \sum_{j=1}^{n_i} (X_{ij} - \bar{X}_{i\cdot})^2.$$

$$\text{记 } S_A = \sum_{i=1}^r n_i (\bar{X}_{i\cdot} - \bar{X})^2,$$

$$S_E = \sum_{i=1}^r \sum_{j=1}^{n_i} (X_{ij} - \bar{X}_{i\cdot})^2,$$

$$\text{则 } S_T = S_A + S_E,$$

S_A 是各组均值对总平均值的偏差平方和, 反映 A 不同水平间的差异, 称为组间平方和; S_E 是各组内的数据对样本均值偏差平方和的总和, 反映了样本观测值与样本均值的差异, 称为组内平方和, 而这种差异认为是由随机误差引起的, 因此也称为误差平方和.

注意到 $\sum_{j=1}^{n_i} (X_{ij} - \bar{X}_{i0})^2$ 是总体 $N(\mu_i, \sigma^2)$ 的样本方差的 $n_i - 1$ 倍, 于是有

$$\sum_{j=1}^{n_i} (X_{ij} - \bar{X}_{i\cdot})^2 / \sigma^2 \sim \chi^2(n_i - 1)$$

由 χ^2 分布的可加性知

$$S_E / \sigma^2 \sim \chi^2 \left(\sum_{i=1}^r (n_i - 1) \right),$$

方差来源	离差平方和	自由度	均方	F 值	概率
因素 A (组间)	S_A	$r - 1$	$S_A/(r - 1)$	$F = \frac{S_A/(r-1)}{S_E/(n-r)} p$	
误差 (组内)	S_E	$n - r$	$S_E/(n - r)$		
总和	S_T	$n - 1$			

即

$$S_E/\sigma^2 \sim \chi^2(n - r),$$

且有

$$ES_E = (n - r)\sigma^2$$

对 S_A 作进一步分析可得

$$ES_A = (r - 1)\sigma^2 + \sum_{i=1}^r n_i \alpha_i^2. \quad (7.10)$$

$$ES_A = (r - 1)\sigma^2$$

可知若 H_0 成立, S_A 只反映随机波动, 而若 H_0 不成立, 那它就还反映了 A 的不同水平的效应 α_i . 单从数值上

$$\frac{S_A/(r - 1)}{S_E/(n - r)} \approx 1$$

该比值服从自由度 $n_1 = r - 1, n_2 = (n - r)$ 的 F 分布, 即

$$F = \frac{S_A/(r - 1)}{S_E/(n - r)} \sim F(r - 1, n - r)$$

以上对

$$S_A, S_E$$

的分析相当于对组间、组内方差的分析. ■

若由实验数据算得结果有 $F > F_\alpha(r - 1, n - r)$, 则拒绝 H_0 , 即认为因素 A 对试验结果有显著影响; 若 $F < F_\alpha(r - 1, n - r)$, 则接受 H_0 , 即认为因素 A 对试验结果没有显著影响.

$F > F_{0.01}(r - 1, n - r)$, 则称因素 A 的影响高度显著.

如果取 $\alpha = 0.01$ 时拒绝 H_0 如果取 $\alpha = 0.05$ 时拒绝 H_0 , 但取 $\alpha = 0.01$ 时不拒绝 H_0 , 即

$$F_{0.01}(r - 1, n - r) \geq F > F_{0.05}(r - 1, n - r)$$

则称因素 A 的影响显著.

22.7.2 双因素方差分析方法

如果要考虑两个因素对指标的影响, 就要采用双因素方差分析. 它的基本思想是: 对每个因素各取几个水平然后对各因素不同水平的每个组合作一次或若干次试验对所得数据进行方差分析. 对双因素方差分析可分为无重复和等重复试验两种情况, 无重复试验只需检验两因素是否分别对指标有显著影响; 而对等重复试验还要进一步检验两因素是否对指标有显著的交互影响.

设 A 取 s 个水平 A_1, A_2, \dots, A_s , B 取 r 个水平 B_1, B_2, \dots, B_r , 在水平组合 (B_i, A_j) 下总体 X_{ij} 服从正态分布 $N(\mu_{ij}, \sigma^2)$, $i = 1, \dots, r, j = 1, \dots, s$. 又设在水平组合 (B_i, A_j) 下作了 t 个试验, 所得结果记作 X_{ijk}, X_{ijk} 服从 $N(\mu_{ij}, \sigma^2)$, $i = 1, \dots, r, j = 1, \dots, s, k = 1, \dots, t$, 且相互独立

Table 22.3: 双因素试验数据表

	A_1	A_2	\dots	A_s
B_1	X_{111}, \dots, X_{11t}	X_{121}, \dots, X_{12t}	\dots	X_{1s1}, \dots, X_{1st}
B_2	X_{211}, \dots, X_{21t}	X_{221}, \dots, X_{22t}	\dots	X_{2s1}, \dots, X_{2st}
\vdots	\vdots	\vdots	\vdots	\vdots
B_r	X_{r11}, \dots, X_{r1t}	X_{r21}, \dots, X_{r2t}	\dots	X_{rs1}, \dots, X_{rst}

将 X_{ijk} 分解为

$$X_{ijk} = \mu_{ij} + \varepsilon_{ijk}, \quad i = 1, \dots, r, \quad j = 1, \dots, s, \\ k = 1, \dots, t,$$

其中 $\varepsilon_{ijk} \sim N(0, \sigma^2)$, 且相互独立. 记

$$\begin{aligned} \mu &= \frac{1}{rs} \sum_{i=1}^r \sum_{j=1}^s \mu_{ij}, \\ \mu_{\cdot j} &= \frac{1}{r} \sum_{i=1}^r \mu_{ij}, \\ \alpha_j &= \mu_{\cdot j} - \mu, \quad \gamma_{ij} = (\mu_{ij} - \mu) - \alpha_i - \beta_j \\ \mu_{i\cdot} &= \frac{1}{s} \sum_{j=1}^s \mu_{ij}, \\ \beta_i &= \mu_{i\cdot} - \mu \end{aligned}$$

μ 是总均值, α_j 是水平 A_j 对指标的效应, β_i 是水平 B_i 对指标的效应, γ_{ij} 是水平 B_i 与 A_j 对指标的交互效应.

模型为

$$\begin{cases} X_{ijk} = \mu + \alpha_j + \beta_i + \gamma_{ij} + \varepsilon_{ijk} \\ \sum_{j=1}^s \alpha_j = 0, \sum_{i=1}^r \beta_i = 0, \sum_{i=1}^r \gamma_{ij} = \sum_{j=1}^s \gamma_{ij} = 0 \\ \varepsilon_{ijk} \sim N(0, \sigma^2), i = 1, \dots, r, j = 1, \dots, s, k = 1, \dots, t \end{cases}$$

原假设为

$$\begin{aligned} H_{01} : \alpha_j = 0 &(j = 1, \dots, s) \\ H_{02} : \beta_i = 0 &(i = 1, \dots, r) \\ H_{03} : \gamma_{ij} = 0 &(i = 1, \dots, r; j = 1, \dots, s). \end{aligned}$$

22.7.3 无交互影响的双因素方差分析

没有交互影响, 每组试验就不必重复, 即可令 $t = 1$, 过程大为简化.

$$\mu_{ij} = \mu + \alpha_j + \beta_i, \quad i = 1, \dots, r, \quad j = 1, \dots, s$$

$$\begin{cases} X_{ij} = \mu + \alpha_j + \beta_i + \varepsilon_{ij} \\ \sum_{j=1}^s \alpha_j = 0, \sum_{i=1}^r \beta_i = 0 \\ \varepsilon_{ij} \sim N(0, \sigma^2), i = 1, \dots, r, j = 1, \dots, s \end{cases}$$

采用与单因素方差分析模型类似的方法导出检验统计量. 记

$$\begin{aligned} \bar{X} &= \frac{1}{rs} \sum_{i=1}^r \sum_{j=1}^s X_{ij}, \bar{X}_{i\cdot} = \frac{1}{s} \sum_{j=1}^s X_{ij}, \bar{X}_{\cdot j} = \frac{1}{r} \sum_{i=1}^r X_{ij} \\ S_T &= \sum_{i=1}^r \sum_{j=1}^s (X_{ij} - \bar{X})^2 \end{aligned}$$

其中 S_T 为全部试验数据的总变差, 称为总平方和

方差来源	离差平方和	自由度	均方	F 值
因素 A	S_A	$s - 1$	$\frac{S_A}{s-1}$	$F_A = \frac{S_A/(s-1)}{S_E/[(r-1)(s-1)]}$
因素 B	S_B	$r - 1$	$\frac{S_B}{r-1}$	$F_B = \frac{S_B/(r-1)}{S_E/[(r-1)(s-1)]}$
误差	S_E	$(r - 1)(s - 1)$	$\frac{S_E}{(r-1)(s-1)}$	
总和	S_T	$rs - 1$		

对其进行分解

$$\begin{aligned}
S_T &= \sum_{i=1}^r \sum_{j=1}^s (X_{ij} - \bar{X})^2 \\
&= \sum_{i=1}^r \sum_{j=1}^s (X_{ij} - \bar{X}_{i\bullet} - \bar{X}_{\cdot j} + \bar{X})^2 + s \sum_{i=1}^r (\bar{X}_{i\bullet} - \bar{X})^2 + r \sum_{j=1}^s (\bar{X}_{\cdot j} - \bar{X})^2 \\
&= S_E + S_A + S_B
\end{aligned}$$

可以验证, 在上述平方和分解中交叉项均为 0, 其中

$$\begin{aligned}
S_E &= \sum_{i=1}^r \sum_{j=1}^s (X_{ij} - \bar{X}_{i\bullet} - \bar{X}_{\cdot j} + \bar{X})^2 \\
S_A &= r \sum_{j=1}^s (\bar{X}_{\cdot j} - \bar{X})^2, \\
S_B &= s \sum_{i=1}^r (\bar{X}_{i\bullet} - \bar{X})^2
\end{aligned}$$

我们先来看看 S_A 的统计意义. 因为 $\bar{X}_{\cdot j}$ 是水平 A_j 下所有观测值的平均, 所以 $\sum_{j=1}^s (\bar{X}_{\cdot j} - \bar{X})^2$ 反映了 $\bar{X}_{\cdot 1}, \bar{X}_{\cdot 2}, \dots, \bar{X}_{\cdot r}$ 差异的程度。这种差异是由于因素 A 的不同水平所引起的, 因此 S_A 称为因素 A 的平方和. 类似地, S_B 称为因素 B 的平方和. 至于 S_E 的意义不甚明显, 我们可以这样来理解:

因为 $S_E = S_T - S_A - S_B$, 在我们所考虑的两因素问题中, 除了因素 A 和 B 之外, 剩余的再没有其它系统性因素的影响, 因此从总平方和中减去 S_A 和 S_B 之后, 剩下的数据变差只能归入随机误差, 故 S_E 反映了试验的随机误差.

有了总平方和的分解式 $S_T = S_E + S_A + S_B$, 以及各个验统计量应取为 S_A 与 S_E 的比.

$$F_A = \frac{\frac{S_A}{s-1}}{\frac{S_E}{(r-1)(s-1)}} \sim F(s - 1, (r - 1)(s - 1))$$

$$F_B = \frac{\frac{S_B}{r-1}}{\frac{S_E}{(r-1)(s-1)}} \sim F(r - 1, (r - 1)(s - 1))$$

检验规则为

$F_A < F_\alpha(s - 1, (r - 1)(s - 1))$ 时接受 H_{01} , 否则拒绝 H_{01} ; $F_B < F_\alpha(r - 1, (r - 1)(s - 1))$ 时接受 H_{02} , 否则拒绝 H_{02} ;

22.7.4 关于交互效应的双因素方差分析

与前面方法类似, 记

$$\begin{aligned}
\bar{X} &= \frac{1}{rst} \sum_{i=1}^r \sum_{j=1}^s \sum_{k=1}^t X_{ijk}, \bar{X}_{ij\cdot} = \frac{1}{t} \sum_{k=1}^t X_{ijk}, \\
\bar{X}_{i\cdot\cdot} &= \frac{1}{st} \sum_{j=1}^s \sum_{k=1}^t X_{ijk}, \bar{X}_{\cdot j\cdot} = \frac{1}{rt} \sum_{i=1}^r \sum_{k=1}^t X_{ijk}
\end{aligned}$$

将全体数据对 \bar{X} 的偏差平方和

$$S_T = \sum_{i=1}^r \sum_{j=1}^s \sum_{k=1}^t (X_{ijk} - \bar{X})^2$$

进行分解, 可得

$$S_T = S_E + S_A + S_B + S_{AB},$$

其中

$$\begin{aligned} S_E &= \sum_{i=1}^r \sum_{j=1}^s \sum_{k=1}^t (X_{ijk} - \bar{X}_{ij\cdot})^2 \\ S_A &= rt \sum_{j=1}^s (\bar{X}_{\cdot j\cdot} - \bar{X})^2 \\ S_B &= st \sum_{i=1}^r (\bar{X}_{i\cdot\cdot} - \bar{X})^2 \\ S_{AB} &= t \sum_{i=1}^r \sum_{j=1}^s (\bar{X}_{ij\cdot} - \bar{X}_{i\cdot 0} - \bar{X}_{\cdot j\cdot} + \bar{X})^2 \end{aligned}$$

称 S_E 为误差平方和, S_A 为因素 A 的平方和 (或列间平方和), S_B 为因素 B 的平方和 (或行间平方和), S_{AB} 为交互作用的平方和 (或格间平方和).

$$F_{AB} = \frac{\frac{S_{AB}}{(r-1)(s-1)}}{\frac{S_E}{rs(t-1)}} \sim F((r-1)(s-1), rs(t-1))$$

据此统计量, 可以检验 H_{03} .

水平 α , 检验的结论为: 为交互作用显著. 将试验数据按上述分析、计算的结果排成表 7.20 的形式, 称为双因素方差分析表.

方差来源	离差平方和	自由度	均方	F 值
因素 A	S_A	$s - 1$	$\frac{S_A}{s-1}$	$F_A = \frac{S_A/(s-1)}{S_E/[rs(t-1)]}$
因素 B	S_B	$r - 1$	$\frac{S_B}{r-1}$	$F_B = \frac{S_B/(r-1)}{S_E/[rs(t-1)]}$
交互效应	S_{AB}	$(r-1)(s-1)$	$\frac{S_{AB}}{(r-1)(s-1)}$	$F_{AB} = \frac{S_{AB}/[(r-1)(s-1)]}{S_E/[rs(t-1)]}$
误差	S_E	$rs(t-1)$		
总和	$\frac{S_E}{rs(t-1)}, S_T$	$rst - 1$		

22.8 多元线性回归

多元线性回归分析的模型为

$$\begin{cases} y = \beta_0 + \beta_1 x_1 + \cdots + \beta_m x_m + \varepsilon \\ \varepsilon \sim N(0, \sigma^2) \end{cases}$$

式中 $\beta_0, \beta_1, \dots, \beta_m, \sigma^2$ 都是与 x_1, x_2, \dots, x_m 无关的未知参数其中 $\beta_0, \beta_1, \dots, \beta_m$ 称为回归系数.

现得到 n 个独立观测数据 $[b_i, a_{i1}, \dots, a_{im}]$, 其中 b_i 为 y 的观察值, a_{i1}, \dots, a_{im} 分别为 x_1, x_2, \dots, x_m 的观察值, $i = 1, \dots, n, n > m$, 可得

$$\begin{cases} b_i = \beta_0 + \beta_1 a_{i1} + \cdots + \beta_m a_{im} + \varepsilon_i \\ \varepsilon_i \sim N(0, \sigma^2), \quad i = 1, \dots, n. \end{cases}$$

记 $X = \begin{bmatrix} 1 & a_{11} & \cdots & a_{1m} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & a_{n1} & \cdots & a_{nm} \end{bmatrix}$, $Y = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$, 7.21 $\varepsilon = [\varepsilon_1, \dots, \varepsilon_n]^T$, $\beta = [\beta_0, \beta_1, \dots, \beta_m]^T$ (7.19) 表示为

$$\begin{cases} Y = X\beta + \varepsilon \\ \varepsilon \sim N(0, \sigma^2 E_n) \end{cases}$$

22.8.1 回归模型的假设检验

因变量 y 与自变量 x_1, \dots, x_m 之间是否存在如模型式 (7.19) 所示的线性关系是需要检验的, 显然, 如果所有的 $|\hat{\beta}_j| (j = 1, \dots, m)$ 都很小, y 与 x_1, \dots, x_m 的线性关

$$H_0: \beta_j = 0, \quad j = 1, \dots, m$$

$$F = \frac{U/m}{Q/(n-m-1)} \sim F(m, n-m-1),$$

在显著性水平 α 下, 对于上 α 分位数 $F_\alpha(m, n-m-1)$, 注: 接受 H_0 只说明 y 与 x_1, \dots, x_m 的线性关系不明显, 可能存在非线性关系, 如平方关系. 还有一些衡量 y 与 x_1, \dots, x_m 相关程度的指标, 如用回归平方和在总平方和中的比值定义复判定系数

$$R^2 = \frac{U}{SST}$$

$R = \sqrt{R^2}$ 称为复相关系数, R 越大, y 与 x_1, \dots, x_m 相关关系越密切, 通常, R 大于 0.8 (或 0.9) 才认为相关关系成立.

22.8.2 回归系数的假设检验和区间估计

其中若干个等于零. 所以应进一步作如下 $m+1$ 个检验

$$H_0^{(j)}: \beta_j = 0, \quad j = 0, 1, \dots, m$$

由式 (7.30), $\hat{\beta}_j \sim N(\beta_j, \sigma^2 c_{jj})$, c_{jj} 是 $(X^T X)^{-1}$ 中的第 (j, j) 元素, 用 s^2 代替 σ^2 $H_0^{(j)}$ 成立时

$$t_j = \frac{\hat{\beta}_j / \sqrt{c_{jj}}}{\sqrt{Q/(n-m-1)}} \sim t(n-m-1)$$

对给定的 α , 若 $|t_j| < t_{\frac{\alpha}{2}}(n-m-1)$, 接受 $H_0^{(j)}$; 否则拒绝. 式也可用于对 β_j 作区间估计, 在置信水平 $1-\alpha$ 下, β_j 的置信区间为

$$\left[\hat{\beta}_j - t_{\frac{\alpha}{2}}(n-m-1)s\sqrt{c_{jj}}, \hat{\beta}_j + t_{\frac{\alpha}{2}}(n-m-1)s\sqrt{c_{jj}} \right]$$

式中, $s = \sqrt{\frac{Q}{n-m-1}}$

22.8.3 利用回归模型进行预测

$[x_1, \dots, x_m]$ 的取值 $[a_{01}, \dots, a_{0m}]$ 预测 y 的取值 b_0 , b_0 是随机的, 显然其预测值(点估计)为

$$\hat{b}_0 = \hat{\beta}_0 + \hat{\beta}_1 a_{01} + \dots + \hat{\beta}_m a_{0m}$$

给定 α 可以算出 b_0 的预测区间(区间估计), 结果较复简化为

$$\left[\hat{b}_0 - z_{\frac{\alpha}{2}} s, \hat{b}_0 + z_{\frac{\alpha}{2}} s \right]$$

式中, z_α 是标准正态分布的上 $\frac{\alpha}{2}$ 分位数. 对 b_0 的区间估计方法可用于给出已知数据残差 $e_i = b_i - \hat{b}_i (i = 1, \dots, n)$ 的置信区间, e_i 服从均值为零的正态分布, 所以若某个 e_i 的置信区间不包含零点, 则认为这个数据是异常的, 可予以剔除.

22.9 逐步回归

实际问题中影响因变量的因素可能很多, 有些可能关联性强一些, 而有些可能影响弱一些. 人们总希望从中挑选出对因变量影响显著的自变量来建立回归模型, 逐步回归是一种从众多变量中有效地选择重要变量的方法以下只讨论多元线性回归模型的情形

简单地说, 就是所有对因变量影响显著的变量都应选入模型, 而影响不显著的变量都不应选入模型; 从便于应用的角度, 变量的选择应使模型中变量个数尽可能少

基本思想: 记 $S = \{x_1, x_2, \dots, x_m\}$ 为候选的自变量集合 $S_1 \subset S$ 是从集合 S 中选出的一个子集. 设 S_1 中有 l 个自变量 ($1 \leq l \leq m$), 由 S_1 和因变量 y 构造的回归模型的残差平方和为 Q , 则模型的残差方差 $s^2 = Q/(n-l-1)$ n 为数据样本容量. 所选子集 S_1 应使 s^2 尽量小. 通常回若模型中包含有对 y 影响很小的变量, 那么 Q 不会由于包含这些变量在内减少多少, 却因 l 的增加可能使 s^2 反而增大, 同时这些对 y 影响不显著的变量也会影响模型的稳定性, 因此可将残差方差 s^2 最小作为衡量变量选择的一个数量标准.

可以从另外一个角度考虑自变量 x_j 的显著性. y 对自变量 x_1, x_2, \dots, x_m 线性回归的残差平方和为 Q , 回归平方和为 U , 在剔除掉 x_j 后, 用 y 对其余的 $m-1$ 个自变量做回归, 记所得的残差平方和为 $Q_{(j)}$, 回归平方和为 $U_{(j)}$, 则自变量 x_j 对回归的贡献为

$$\Delta U_{(j)} = U - U_{(j)}$$

称为 x_j 的偏回归平方和. 由此构造偏 F 统计量

$$F_j = \frac{\Delta U_{(j)}/1}{Q/(n-m-1)}$$

F_j 服从自由度为 $(1, n-m-1)$ 的 F 分布, 此 F 检验与式方程中剔除变元时, 回归平方和减少, 残差平方和增加. 根据平方和分解式可知

$$\Delta U_{(j)} = \Delta Q_{(j)} = Q_{(j)} - Q$$

残差平方和减少, 两者的增减量同样相等.

当自变量的个数较多时, 求出所有可能的回归方程是非常困难的. 为此, 人们提出了一些较为简便、实用、快速的选择自变量的方法. 这些方法各有优缺点, 至今还没有绝对最优的方法, 目前常用的方法有前进法、后退法、逐步回归法, 而逐步回归法最受推崇.

22.9.1 前进法

前进法的思想是变量由少到多, 每次增加一个, 直至没有可引入的变量为止. 具体做法是首先将全部 m 个自变量分别对因变量 y 建立一元线性回归方程, 利用归系数的 F 检验值, 记为 $\{F_1^1, F_2^1, \dots, F_m^1\}$, 选其最大者记为

$$F_j^1 = \max \{F_1^1, F_2^1, \dots, F_m^1\}$$

给定显著性水平 α , 若 $F_j^1 \geq F_\alpha(1, n - 2)$, 则首先将 x_j 引入回归方程, 为了方便, 不妨设 x_j 就是 x_1 .

接下来因变量 y 分别与 $(x_1, x_2), (x_1, x_3), \dots, (x_1, x_m)$ 建立二元线性回归方程, 对这 $m - 1$ 个回归方程中 x_2, x_3, \dots, x_m 的回归系数进行 F 检验, 利用 (7.43) 式计算 F 值, 记为 $\{F_2^2, F_3^2, \dots, F_m^2\}$, 选其最大者记为

$$F_j^2 = \max \{F_2^2, F_3^2, \dots, F_m^2\}$$

若 $F_j^2 \geq F_\alpha(1, n - 3)$, 则接着将 x_j 引入回归方程.

依上述方法接着做下去, 直至所有未被引入方程的入变量的个数. 这时, 得到的回归方程就是确定的方程. 有关, 在用软件计算时, 我们实际是使用显著性 P 值做检验.

22.9.2 后退法

后退法易于掌握, 我们使用 t 统计量做检验, 与 F 统计量做检验是等价的. 具体步骤如下:

Algorithm 28: 后退法

- 1 以全部自变量作为解释变量拟合方程
- 2 每一步都在未通过 t 检验的自变量中选择一个 t_j 值最小的变量 x_j , 将它从模型中删除
- 3 直至所有的自变量均通过 t 检验, 则算法终止

23. Bootstrap

设总体的分布 F 未知, 但已知有一个容量为 n 的来自分布 F 的数据样本, 自这一样本按放回抽样的方法抽取一个容量为 n 的样本, 这种样本称为 bootstrap 样本或称为自助样本. 相继地, 独立地自原始样本中取很多个 Bootstrap 样本, 利用这些样本对总体 F 进行统计推断, 这种方法称为非参数 Bootstrap 方法, 又称自助法. 这一方法可以用于当人们对总体知之甚少的情况, 它是近代统计中的一种用于数据处理的重要实用方法. 这种方法的实现需要在计算机上作大量的计算, 随着计算机威力的增长, 它已成为一种流行的方法.

23.1 估计量的标准误差的 Bootstrap 估计

在估计总体未知参数 θ 时, 人们不但要给出 θ 的估计 $\hat{\theta}$, 还需指出这一估计 $\hat{\theta}$ 的精度. 通常我们用估计量 $\hat{\theta}$ 的标准差 $\sqrt{D(\hat{\theta})}$ 来度量估计的精度. 估计量 $\hat{\theta}$ 的标准差 $\sigma_{\hat{\theta}} = \sqrt{D(\hat{\theta})}$ 也称为估计量 $\hat{\theta}$ 的标准误差.

设 X_1, X_2, \dots, X_n 是来自 $\mathcal{F}(x)$ 为分布函数的总体的样本, θ 是我们感兴趣的未知参数, 用 $\hat{\theta} = \hat{\theta}(X_1, X_2, \dots, X_n)$ 作为 θ 的估计量, 在应用中 $\hat{\theta}$ 的抽样分布常是很难处理的, 这样, $\sqrt{D(\hat{\theta})}$ 常没有一个简单的表达式, 不过我们可以用计算机模拟的方法来求得 $\sqrt{D(\hat{\theta})}$ 的估计.

为此, 自 F 产生很多容量为 n 的样本 (例如 B 个), 对于每一个样本计算 $\hat{\theta}$ 的值, 得 $\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_B$, 则 $\sqrt{D(\hat{\theta})}$ 可以用

$$\hat{\sigma}_{\hat{\theta}} = \sqrt{\frac{1}{B-1} \sum_{i=1}^B (\hat{\theta}_i - \bar{\theta})^2}$$

来估计, 其中 $\bar{\theta} = \frac{1}{B} \sum_{i=1}^B \hat{\theta}_i$.

然而 F 常常是未知的, 这样就无法产生模拟样本, 需要另外的方法.

设分布 F 未知, x_1, x_2, \dots, x_n 是来自 F 的样本值, F_n 是相应的经验分布函数. 当 n 很大时, F_n 接近 F . 用 F_n 代替上一段中的 F , 在 F_n 中抽样. 在 F_n 中抽样, 就是在原始样本 x_1, x_2, \dots, x_n 中每次随机地取一个个体作放回抽样. 如此得到一个容量为 n 的样

本 $x_1^*, x_2^*, \dots, x_n^*$, 这就是第一段中所说的 Bootstrap 样本. 用 Bootstrap 样本按上一段中计算估计 $\hat{\theta}(x_1, x_2, \dots, x_n)$ 那样求出 θ 的估计 $\hat{\theta}^* = \hat{\theta}(x_1^*, x_2^*, \dots, x_n^*)$ 估计 $\hat{\theta}^*$ 称为 θ 的 Bootstrap 估计.

相应地、独立地抽得 B 个 Bootstrap 样本, 以这些样本分别求出 θ 的相应的 Bootstrap 估计如下:

Bootstrap 样本 1 $x_1^{*1}, x_2^{*1}, \dots, x_n^{*1}$, Bootstrap 估计 $\hat{\theta}_1^*$;

Bootstrap 样本 2 $x_1^{*2}, x_2^{*2}, \dots, x_n^{*2}$, Bootstrap 估计 $\hat{\theta}_2^*$;

Bootstrap 样本 B $x_1^{*B}, x_2^{*B}, \dots, x_n^{*B}$, Bootstrap 估计 $\hat{\theta}_B^*$.

则 $\hat{\theta}$ 的标准误差 $\sqrt{D(\hat{\theta})}$, 就以

$$\hat{\sigma}_{\hat{\theta}} = \sqrt{\frac{1}{B-1} \sum_{i=1}^B (\hat{\theta}_i^* - \bar{\theta}^*)^2}$$

来估计, 其中 $\bar{\theta}^* = \frac{1}{B} \sum_{i=1}^B \hat{\theta}_i^*$, 上式就是 $\sqrt{D(\hat{\theta})}$ 的 Bootstrap 估计.

Algorithm 29: 求 $\sqrt{D(\hat{\theta})}$ 的 Bootstrap 估计

- 1 自原始数据样本 x_1, x_2, \dots, x_n 按放回抽样的方法, 抽得容量为 n 的样本 $x_1^*, x_2^*, \dots, x_n^*$ (称为 Bootstrap 样本)
- 2 相继地、独立地求出 $B (B \geq 1000)$ 个容量为 n 的 Bootstrap 样本,
 $x_1^{*i}, x_2^{*i}, \dots, x_n^{*i}, i = 1, 2, \dots, B$. 对于第 i 个 Bootstrap 样本, 计算
 $\hat{\theta}_i^* = \hat{\theta}(x_1^{*i}, x_2^{*i}, \dots, x_n^{*i}), i = 1, 2, \dots, B$ ($\hat{\theta}_i^*$ 称为 θ 的第 i 个 Bootstrap 估计)
- 3 计算 $\hat{\sigma}_{\hat{\theta}} = \sqrt{\frac{1}{B-1} \sum_{i=1}^B (\hat{\theta}_i^* - \bar{\theta}^*)^2}$, 其中 $\bar{\theta}^* = \frac{1}{B} \sum_{i=1}^B \hat{\theta}_i^*$

23.2 估计量的均方误差的 Bootstrap 估计

设 $X = (X_1, X_2, \dots, X_n)$ 是来自总体 F 的样本, F 未知, $R = R(X)$ 是感兴趣的随机变量, 它依赖于样本 X . 假设我们希望去估计 R 的分布的某些特征. 例如 R 的数学期望 $E_F(R)$, 就可以按照上面所说的三个步骤 1°, 2°, 3° 进行, 只是在 2° 中对于第 i 个 Bootstrap 样本 $x_i^* = (x_1^{*i}, x_2^{*i}, \dots, x_n^{*i})$, 计算 $R_i^* = R_i^*(x_i^*)$ 代替计算 θ_i^* , 且在 3° 中计算感兴趣的 R 的特征. 例如如果希望估计 $E_F(R)$ 就计算

$$E_*(R^*) = \frac{1}{B} \sum_{i=1}^B R_i^*$$

23.3 Bootstrap 置信区间

设 $X = (X_1, X_2, \dots, X_n)$ 是来自总体 F 容量为 n 的样本 $x = (x_1, x_2, \dots, x_n)$ 是一个已知的样本值. F 中含有未知参数 θ , $\hat{\theta} = \hat{\theta}(X_1, X_2, \dots, X_n)$ 是 θ 的估计量. 现在来求 θ 的置信水平为 $1 - \alpha$ 的置信区间.

相继地, 独立地从样本 $x = (x_1, x_2, \dots, x_n)$ 中抽出 B 个容量为 n 的 Bootstrap 样本, 对于每个 Bootstrap 样本求出 θ 的 Bootstrap 估计: $\hat{\theta}_1^*, \hat{\theta}_2^*, \dots, \hat{\theta}_B^*$. 将它们自小到大排序, 得

$$\hat{\theta}_{(1)}^* \leq \hat{\theta}_{(2)}^* \leq \cdots \leq \hat{\theta}_{(B)}^*$$

取 $R(X) = \hat{\theta}$, 用对应的 $R(X^*) = \hat{\theta}^*$ 的分布作为 $R(X)$ 的分布的近似, 求出 $R(X^*)$ 的分布的近似分位数 $\hat{\theta}_{\alpha/2}^*$ 和 $\hat{\theta}_{1-\alpha/2}^*$ 使

$$P \left\{ \hat{\theta}_{\alpha/2}^* < \hat{\theta}^* < \hat{\theta}_{1-\alpha/2}^* \right\} = 1 - \alpha$$

于是近似地有

$$P \left\{ \hat{\theta}_{\alpha/2}^* < \theta < \hat{\theta}_{1-\alpha/2}^* \right\} = 1 - \alpha.$$

记 $k_1 = \lceil B \times \frac{\alpha}{2} \rceil$, $k_2 = \lceil B \times (1 - \frac{\alpha}{2}) \rceil$, 在上式中以 $\hat{\theta}_{(k_1)}^*$ 和 $\hat{\theta}_{(k_2)}^*$ 分别作为分位数 $\hat{\theta}_{\alpha/2}^*$ 和 $\hat{\theta}_{1-\alpha/2}^*$ 的估计, 得到近似等式

$$P \left\{ \hat{\theta}_{(k_1)}^* < \theta < \hat{\theta}_{(k_2)}^* \right\} = 1 - \alpha$$

于是由上式就得到 θ 的置信水平为 $1 - \alpha$ 的近似置信区间 $(\hat{\theta}_{(k_1)}^*, \hat{\theta}_{(k_2)}^*)$, 这一区间称为 θ 的置信水平为 $1 - \alpha$ 的 Bootstrap 置信区间. 这种求置信区间的方法称为分位数法.

23.4 参数 Bootstrap 方法

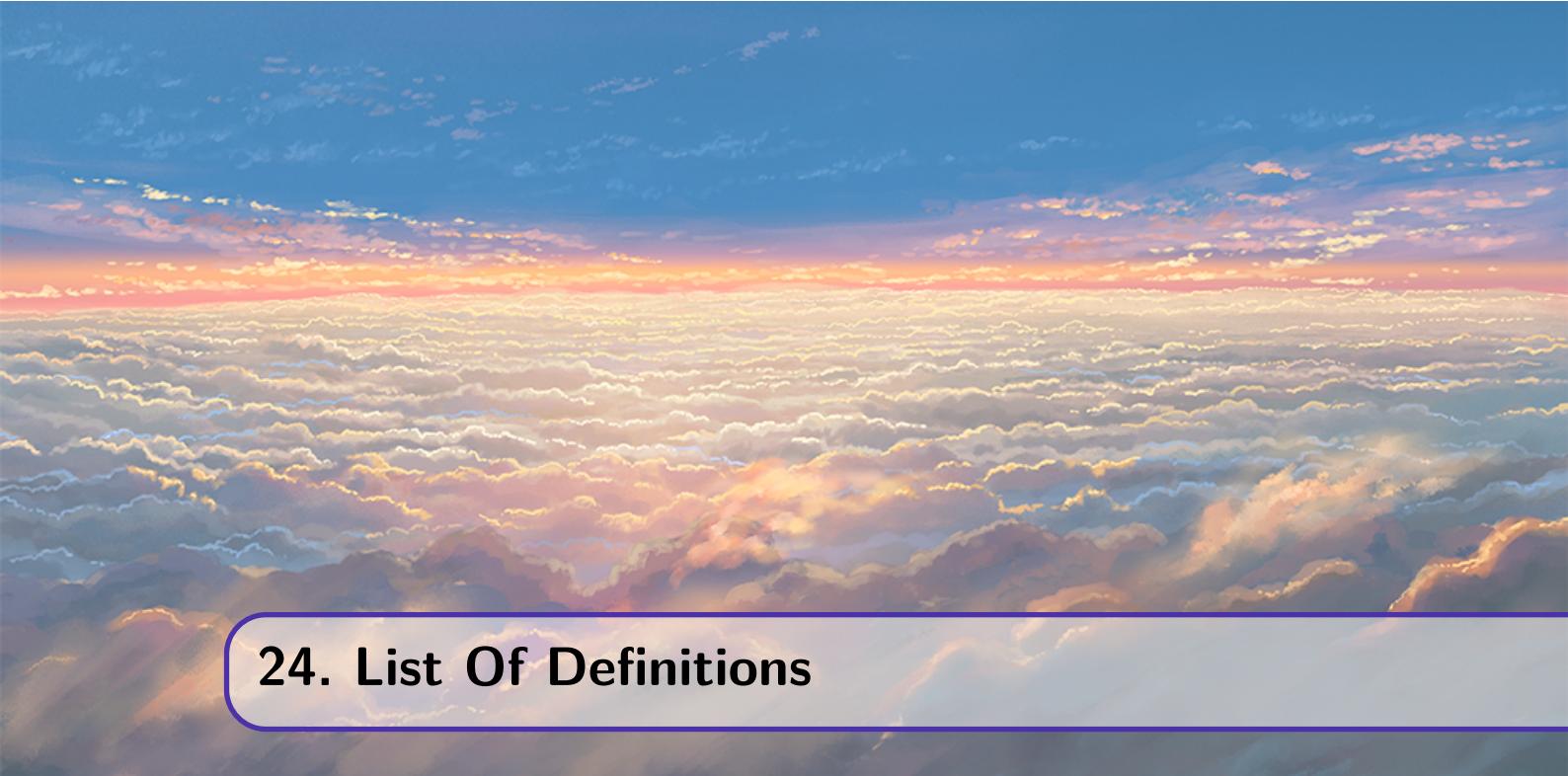
假设所研究的总体的分布函数 $F(x; \beta)$ 的形式已知, 但其中包含未知参数 β (β 可以是向量). 现在已知有一个来自 $F(x; \beta)$ 的样本

$$X_1, X_2, \dots, X_n$$

利用这一样本求出 β 的最大似然估计 $\hat{\beta}$. 在 $F(x; \beta)$ 中以 $\hat{\beta}$ 代替 β 得到 $F(x; \hat{\beta})$, 接着在 $F(x; \hat{\beta})$ 中产生容量为 n 的样本

$$X_1^*, X_2^*, \dots, X_n^* \sim F(x; \hat{\beta})$$

这种样本可以产生很多个, 例如产生 B ($B \geq 1000$) 个, 就可以利用这些样本对总体进行统计推断, 其做法与非参数 Bootstrap 方法一样. 这种方法称为参数 Bootstrap 方法.



24. List Of Definitions

Chapter 1 对向量的介绍	6
1.1.1 (Definition) Vector	6
1.1.2 (Definition) n 维向量 a 的第 i 元素	6
1.1.3 (Definition) $a = b$	6
1.1.4 (Definition) stacked vector	6
1.1.5 (Definition) 零向量	6
1.1.6 (Definition) 全一向量	7
1.1.7 (Definition) 单位向量	7
1.1.8 (Definition) 稀疏向量	7
1.2.1 (Definition) 向量空间 V	7
1.2.1 (Corollary)	7
1.2.2 (Corollary)	7
1.2.2 (Definition) 数域	7
1.3.1 (Definition) 向量加法	7
1.3.1 (Corollary) 向量位移相加	8
1.3.2 (Definition) 标量与向量的乘法	8
1.3.3 (Definition) 线性组合	8
1.4.1 (Definition) 内积	8
1.4.1 (Corollary) 选出第 i 项	9
1.4.2 (Corollary) 向量每一项之和	9
1.4.3 (Corollary) 向量每一项的平方和	9
1.4.4 (Corollary) 向量元素的平均值	9
1.4.5 (Corollary) Selective sum	9

1.4.2 (Definition) The sum of block vectors	9
1.5.1 (Theorem) Cauchy-Schwartz Inequality	10
1.5.2 (Theorem) Cauchy-Schwarz 不等式的矩阵元素形式	11
1.5.3 (Corollary) Cauchy-Schwarz 不等式变体	11
1.6.1 (Corollary)	11
Chapter 2 Linear Function	12
2.1.1 (Definition) Linear Function	12
2.1.1 (Theorem)	12
2.1.2 (Definition) 内积函数 (inner product function)	13
2.1.2 (Corollary)	13
2.1.3 (Corollary)	13
2.1.3 (Definition) 仿射函数 (affine function)	13
2.1.4 (Theorem)	13
2.2.1 (Definition) 函数 f 第 i 个分量的一阶偏导数	13
2.2.2 (Definition) f 在点 z 的梯度	14
2.2.3 (Definition) Taylor's Approximation	14
2.2.4 (Definition) 一阶泰勒公式	14
2.2.1 (Corollary) 一阶泰勒公式的内积形式	14
2.3.1 (Corollary) n 阶泰勒多项式	15
2.3.2 (Corollary) 对于高阶余项的公式	15
2.3.3 (Corollary) 麦克劳林 (Maclaurin) 公式	15
2.4.1 (Definition) Regression Model	16
Chapter 3 Norm and Distance	17
3.1.1 (Definition) Vector Norm	17
3.1.1 (Corollary) 柯西—施瓦茨不等式	18
3.1.2 (Definition) ℓ_∞ -范数	18
3.1.3 (Definition) ℓ_p -范数	18
3.1.2 (Theorem) Minkowski Inequality	18
3.1.3 (Theorem) Hölder Inequality	18
3.2.1 (Definition) 向量 x 的均方值 (mean-square value)	19
3.2.2 (Definition) n 维向量 x 的均方根 (root-mean-square value, RMS)	19
3.3.1 (Theorem) Chebyshev's Inequality	19
3.3.2 (Theorem) Chebyshev's Inequality	19
3.3.3 (Corollary) Chebyshev's Inequality Using RMS	19
3.4.1 (Definition) Euclidean distance	20
3.4.2 (Definition) RMS deviation	20
3.4.1 (Theorem) Triangular Inequality	20
3.4.3 (Definition) Feature Distance	20
3.4.4 (Definition) Nearest Neighbor	20

3.5.1 (Definition)	算术平均值	20
3.5.2 (Definition)	De-meaned Vector	20
3.5.3 (Definition)	x 的标准差	20
3.5.1 (Theorem)		20
3.6.1 (Definition)	两个非零向量 a 和 b 之间的角 (angle)	20
3.6.2 (Definition)	球面的距离	21
3.6.3 (Definition)	a 和 b 的相关系数	21
4.1 (Problem)		22
Chapter 4 优化问题初步		22
4.1.1 (Definition)	高阶无穷小记号 o	22
4.1.1 (Corollary)		22
4.1.2 (Theorem)	优化求解的必要条件	22
4.2.1 (Definition)	凸集	23
4.2.2 (Definition)	凸函数	23
4.2.1 (Theorem)	可微函数 f 是凸函数的充要条件	24
4.2.2 (Theorem)		24
4.3.1 (Definition)	向量对向量的导数	25
4.2 (Problem)		26
4.3 (Problem)		27
Chapter 5 Linear Independence		29
5.1.1 (Definition)	线性相关 (linearly dependent)	29
5.1.1 (Corollary)		29
5.1.2 (Corollary)		29
5.1.2 (Definition)	线性独立 (linearly independent)	29
5.1.3 (Corollary)		29
5.1.4 (Theorem)		30
5.2.1 (Definition)	基 (Basis)	30
5.2.2 (Definition)	向量 b 在基底 a_1, \dots, a_n 下的分解	30
5.3.1 (Definition)	Orthogonal Vectors	30
5.3.2 (Definition)	Orthonormal Vectors	31
5.3.1 (Corollary)		31
5.3.2 (Corollary)		31
5.3.3 (Definition)	n 维向量的一个标准正交基	32
5.3.4 (Definition)	x 在标准正交基下的标准正交分解	32
5.4.1 (Theorem)		32
5.4.2 (Corollary)		33
5.4.3 (Corollary)		33
5.4.4 (Corollary)		33
5.4.5 (Corollary)		34

5.4.6 (Corollary)	34
Chapter 6 Matrices	36
6.1.1 (Definition) 矩阵	36
6.1.2 (Definition) 标量	36
6.1.3 (Definition) 向量	36
6.1.4 (Definition) 行向量, 列向量	36
6.1.5 (Definition) 高形, 宽形和方形矩阵	36
6.1.6 (Definition) 分块矩阵	36
6.1.7 (Definition) 矩阵的列向量表示	37
6.1.8 (Definition) 矩阵的行向量表示	37
6.2.1 (Definition) 矩阵数乘	37
6.2.2 (Definition) 矩阵加法	37
6.2.3 (Definition) Transpose	37
6.2.1 (Corollary) 转置的性质	37
6.2.4 (Definition) 共轭转置	37
6.2.2 (Corollary) 共轭转置的性质	38
6.2.5 (Definition) 矩阵乘法	38
6.2.3 (Corollary) 矩阵乘法性质	38
6.2.6 (Definition) 分块矩阵乘法	38
6.2.7 (Definition) 矩阵-向量乘积 Ax	38
6.2.4 (Corollary)	38
6.2.8 (Definition) 矩阵-向量乘积函数 $f(x) = Ax$	40
6.2.9 (Definition) Matrix Power	40
6.3.1 (Definition) Zero Matrix	42
6.3.2 (Definition) 单位矩阵	42
6.3.1 (Corollary)	42
6.3.3 (Definition) Symmetric Matrices	42
6.3.4 (Definition) Hermitian Matrices	42
6.3.5 (Definition) Diagonal Matrices	42
6.3.6 (Definition) 下三角矩阵	42
6.3.7 (Definition) 上三角矩阵	42
6.3.8 (Definition) Selector matrices	44
6.3.9 (Definition) Downsampling	44
6.3.10 (Definition) 关联矩阵	44
6.3.11 (Definition) 一维卷积	44
6.3.2 (Corollary)	44
6.3.3 (Corollary) 卷积性质	45
6.3.4 (Corollary)	45
6.3.12 (Definition) 多项式	45
6.3.13 (Definition) Vandermonde Matrices	45

6.3.14 (Definition)	Discrete Fourier Transform (DFT)	45
6.3.15 (Definition)	Discrete Inverse Fourier Transform	46
6.3.16 (Definition)	半正定矩阵	46
6.3.17 (Definition)	正定矩阵	46
6.3.18 (Definition)	二次型	46
6.3.5 (Corollary)		46
6.3.6 (Theorem)	正定矩阵对角元素性质	47
6.3.7 (Theorem)	半正定矩阵对角元素性质	47
6.4.1 (Definition)	实矩阵 A 的 Gram 矩阵	47
6.4.2 (Definition)	复矩阵的 A 的 Gram 矩阵	47
6.4.1 (Theorem)		47
6.4.2 (Theorem)		47
6.4.3 (Corollary)		47
Chapter 7 Matrices Norms		48
7.1.1 (Definition)	Matrix Norm	48
7.1.2 (Definition)	从属于给定向量范数 $\ x\ _v$ 的矩阵范数	49
7.1.3 (Definition)	向量范数和算子范数相容	49
7.1.1 (Theorem)	算子范数服从乘法范数相容性	49
7.1.4 (Definition)	A 的列范数	50
7.1.5 (Definition)	A 的行范数	50
7.1.6 (Definition)	A 的 2-范数	50
Chapter 8 适定问题		53
8.3.1 (Definition)	非奇异矩阵 A 的条件数 (condition number)	53
8.3.1 (Corollary)	非奇异矩阵 A 的条件数 (condition number) 性质	53
Chapter 9 Inverse of Matrices		54
9.1.1 (Definition)	A 的左逆	54
9.1.2 (Definition)	A 的右逆	54
9.1.1 (Theorem)		55
9.1.2 (Theorem)		55
9.1.3 (Theorem)		55
9.1.3 (Definition)	逆 A^{-1}	55
9.2.1 (Definition)	Linear Equation Systems	55
9.2.1 (Theorem)		55
9.2.2 (Theorem)		56
9.2.3 (Theorem)		56
9.4.1 (Theorem)		56
9.4.2 (Theorem)		57
9.4.3 (Corollary)		57

9.4.4 (Theorem)	58
9.4.5 (Theorem)	58
9.4.6 (Corollary)	58
9.4.7 (Theorem)	58
9.5.1 (Theorem) 转置 A^T 和共轭转置 A^H	59
9.5.2 (Corollary)	59
9.6.1 (Corollary) Gram Matrix 可逆等价于 A 列线性无关	59
9.7.1 (Definition) Pseudo-inverse	60
9.7.1 (Theorem)	60
9.7.2 (Theorem)	60
9.7.3 (Theorem)	60
9.7.4 (Corollary)	60
9.7.5 (Corollary)	60
10.1.1 (Theorem)	63
10.1.2 (Theorem)	64
10.1.3 (Theorem)	64
10.1.4 (Theorem)	64
10.1.5 (Theorem)	64
10.1.6 (Theorem)	64
10.1.7 (Theorem)	65
10.1.8 (Theorem)	65
Chapter 10 Orthogonal Matrices	65
10.2.1 (Definition) 正交矩阵	65
10.2.1 (Theorem) 正交矩阵满足非奇异性	65
10.2.2 (Corollary)	65
10.2.3 (Corollary)	65
10.3.1 (Definition) 置换	65
10.3.1 (Theorem)	65
10.3.2 (Corollary)	65
10.3.3 (Corollary)	66
10.5.1 (Definition) Householder matrix, refletor matrix	66
10.5.1 (Theorem)	67
10.5.2 (Theorem)	67
10.5.3 (Theorem)	67
10.5.4 (Corollary)	67
10.5.5 (Corollary)	68
10.6.1 (Corollary) 正交矩阵乘积的正交性	68
10.8.1 (Theorem)	68
10.8.2 (Theorem)	69
10.8.3 (Theorem)	69

10.9.1 (Definition)	向量集合张成的空间	69
10.9.2 (Definition)	A 的范围 (列空间)	69
10.1 (Problem)		69
10.9.1 (Theorem)		70
Chapter 11 QR 分解与 Householder 变换		71
11.1.1 (Definition)	Lower Triangular Matrices	71
11.1.2 (Definition)	Upper Triangular Matrices	71
11.1.3 (Definition)	单位上三角矩阵, 单位下三角矩阵	71
11.1 (Problem)		71
11.2 (Problem)		71
11.1.1 (Theorem)		73
11.1.2 (Theorem)	高斯消元法	73
11.1.3 (Theorem)		73
11.2.1 (Theorem)	QR Factorization	73
11.2.2 (Corollary)		73
11.2.3 (Corollary)		73
11.2.4 (Corollary)		73
11.2.5 (Corollary)		73
11.2.6 (Corollary)		74
11.2.7 (Theorem)	$m = n$ 矩阵 A 进行 QR 分解的唯一性	74
11.2.8 (Theorem)	$m < n$ 矩阵 A 进行 QR 分解的唯一性	75
11.2.9 (Theorem)	$m > n$ 矩阵 A 进行 QR 分解的唯一性	75
11.2.10 (Theorem)		76
11.3.1 (Corollary)		77
11.3.1 (Definition)	线性无关列向量的矩阵 A 的伪逆	77
11.3.2 (Theorem)		77
11.3.3 (Corollary)		77
11.3.4 (Corollary)		77
11.3.5 (Theorem)		78
11.3.6 (Theorem)		78
11.3.7 (Theorem)		78
11.4.1 (Corollary)		80
11.4.2 (Corollary)		80
11.4.3 (Corollary)		80
11.4.4 (Theorem)		80
11.6.1 (Theorem)		85
11.6.2 (Theorem)		86
11.6.1 (Definition)		86
11.6.3 (Theorem)		87
11.6.2 (Definition)	Constructed Householder Matrix	87

11.6.4 (Theorem)	87
11.6.5 (Theorem)	89
11.3 (Problem)	91
11.7.1 (Definition) v_1, \dots, v_n 的完整表示	93
11.7.1 (Theorem)	93
11.7.2 (Definition) 矩阵-向量积 $H_k x$	93
11.8.1 (Definition) Givens Rotation in the 1-2 plane	93
11.8.1 (Theorem)	94
11.8.2 (Theorem) H 的逆是它本身	95
11.9.1 (Theorem)	95
11.9.2 (Corollary)	95
11.9.3 (Corollary)	95
11.9.4 (Theorem)	95
11.9.5 (Theorem) QR 分解求解 A^{-1}	96
Chapter 12 LU 分解	97
12.1.1 (Definition) Pivot	97
12.1.2 (Definition) Multiplier	97
12.1.3 (Definition) Elementary matrix, Elimination matrix	98
12.1.1 (Theorem)	98
12.1.4 (Definition) Permutation matrices P	98
12.1.5 (Definition) Augmented matrix	98
12.2.1 (Theorem)	99
12.2.1 (Definition) A 的 LU 分解	99
12.2.2 (Theorem)	100
12.2.3 (Theorem)	100
12.2.4 (Corollary)	100
12.2.5 (Corollary)	101
12.2.6 (Theorem)	101
12.2.7 (Corollary)	102
12.2.8 (Corollary)	102
12.2.9 (Theorem)	102
12.2.10(Corollary)	103
12.2.11(Corollary)	103
12.4.1 (Theorem)	104
12.4.2 (Theorem)	105
12.6.1 (Theorem)	106
12.6.2 (Corollary)	106
13.1 (Problem)	108
13.2 (Problem) 最小二乘问题	109
13.3 (Problem) 求解最小二乘解	109

13.1.1 (Theorem)	109
13.2.1 (Theorem)	110
13.2.2 (Theorem) 正规方程与最小二乘解	110
13.3.1 (Theorem) 投影与 A 列空间的关系	113
13.4.1 (Theorem) 最小二乘法问题的正规方程	113
13.4.2 (Theorem)	113
13.5.1 (Theorem) QR 分解求解最小二乘法	113
13.4 (Problem)	115
13.7.1 (Corollary)	116
Chapter 13 Least Squares	116
13.7.1 (Definition) 梯度下降法求解最小二乘法	116
13.5 (Problem)	116
13.8.1 (Theorem) 线性搜索估计的最优步长	117
14.1 (Problem)	119
14.2 (Problem) 加权最小二乘法问题	119
14.3 (Problem) 加权最小二乘法问题紧密形式	119
14.4 (Problem) 加权最小二乘法问题矩阵形式	119
14.5 (Problem) 双目标规划问题	120
14.6 (Problem)	120
14.2.1 (Theorem)	120
14.7 (Problem)	120
14.8 (Problem)	120
14.9 (Problem)	120
14.10 (Problem)	121
14.11 (Problem)	121
14.12 (Problem) 最小二乘法进行估计	121
Chapter 14 Multi-objective Least Squares	121
14.4.1 (Definition) Tikhonov 正则化	121
14.4.1 (Theorem) 岭回归问题的标准方程	121
14.4.2 (Definition) 图像转换为列向量存储	121
14.4.3 (Definition) 垂直差分矩阵	122
14.4.4 (Definition)	122
14.13 (Problem)	122
14.14 (Problem) 信号去噪问题	123
14.5.1 (Definition) 差分矩阵	123
15.1 (Problem)	124
Chapter 15 Constrained Least Squares	124
15.1.1 (Definition) 拉格朗日乘子法和 Lagrange Functions	125
15.1.1 (Theorem) 拉格朗日函数求导	125

15.1.2 (Theorem)	Karush-Kuhn-Tucker Conditions	126
15.2 (Problem)		126
15.3 (Problem)		127
15.4 (Problem)		127
15.5 (Problem)		128
15.6 (Problem)		128
15.7 (Problem)		128
15.4.1 (Theorem)		129
15.8 (Problem)	分段多项式拟合问题	130
15.9 (Problem)		131
15.10 (Problem)	分段多项式拟合问题	132
15.11 (Problem)	分段多项式拟合问题 (矩阵形式)	132
15.12 (Problem)		132
15.13 (Problem)		133
15.10.1(Theorem)	优化条件的 Karush-Kuhn-Tucker(KKT) 等式	133
15.10.2(Theorem)	KKT 等式最优解的唯一性	134
15.10.3(Theorem)		134
15.10.4(Theorem)		135
15.14 (Problem)		135
15.11.1(Theorem)		135
15.15 (Problem)		136
15.16 (Problem)	等式约束优化问题	138
15.14.1(Definition)	Lagrangian 函数	138
15.14.1(Theorem)		138
15.14.2(Theorem)	拉格朗日乘子法最优解必要条件	138
15.17 (Problem)	不等式约束优化问题	138
15.14.3(Theorem)		139
15.14.4(Corollary)		139
15.14.5(Theorem)	Karush-Kuhn-Tucker (KKT) 条件	139
15.14.2(Definition)	标准约束优化问题 (非线性规划)	139
15.14.6(Theorem)	标准约束优化的 KKT 条件	139
15.18 (Problem)		140
15.15.1(Theorem)	KKT 条件	140
15.19 (Problem)	等式约束优化问题	141
15.15.2(Theorem)	一元一次优化式的 KKT 条件	142
15.15.3(Corollary)		143
15.20 (Problem)		143
15.15.4(Theorem)		143
15.15.5(Corollary)	梯度的性质	143
15.15.6(Corollary)		145

15.15.7(Theorem)	同时包含等式和不等式约束的一般优化问题的 KKT 条件	145
15.16.1(Definition)	点、(直) 线、线段	146
15.16.2(Definition)	仿射集	146
15.16.3(Definition)	凸集	146
15.16.1(Theorem)		146
15.16.4(Definition)	凸函数	146
15.16.2(Theorem)		146
15.16.3(Theorem)		146
15.16.4(Theorem)	α -sublevel 集	146
15.16.5(Definition)	优化问题	147
15.16.6(Definition)	凸优化	147
15.16.5(Corollary)		147
15.16.7(Definition)	凸优化问题	147
15.16.6(Theorem)		147
15.16.8(Definition)	二次规划	147
15.16.7(Theorem)		147
15.16.9(Definition)	拉格朗日函数	148
15.16.8(Theorem)		148
15.21 (Problem)		148
15.16.1(Definition)	primal feasible	148
15.16.10(Definition)	对偶问题	149
15.16.11(Definition)	对偶可行	149
15.16.9(Theorem)		149
15.16.10(Theorem)	弱对偶性	149
15.16.11(Theorem)	强对偶性	149
15.16.12(Theorem)	Slater 条件	149
15.16.13(Theorem)	互补松弛性 (complementary slackness, KKT complementarity)	
	149	
15.16.14(Theorem)		150
15.16.15(Theorem)		150
15.17.1(Theorem)		151
15.17.2(Theorem)		151
15.22 (Problem)	convex minimization problem	160
15.19.1(Theorem)		161
15.19.2(Theorem)		161
16.1 (Problem)		162
16.2 (Problem)		162
16.3 (Problem)		162
16.4 (Problem)		163
16.5 (Problem)		163
16.6 (Problem)		164

16.7 (Problem) 二分类问题	164
Chapter 16 非线性最小二乘法	164
16.2.1 (Definition) $\mathbb{R}^n \rightarrow \mathbb{R}$ 函数 g 的梯度	165
16.2.2 (Definition) g 在 z 附近的仿射近似 (一阶泰勒公式, 线性化)	165
16.2.3 (Definition) Directional Derivative	165
16.2.4 (Definition) Jacobian Matrices	165
16.2.5 (Definition) f 在 z 附近的仿射近似 (线性化)	165
16.2.6 (Definition) Hessian Matrices	166
16.2.7 (Definition) Quadratic (second order) approximation of g around z	166
16.2.1 (Theorem) Linear combination properties	166
16.2.2 (Theorem) Composition with affine mapping properties	166
16.2.3 (Theorem) Necessary condition for twice differentiable g	167
16.2.4 (Theorem) Sufficient condition	167
16.2.8 (Definition) Convex functions	167
16.2.5 (Theorem) Necessary and sufficient condition for convex functions	167
16.3.1 (Theorem) 非线性最小二乘问题的最优必要条件	169
16.3.2 (Corollary) 正规方程的最优必要条件	169
16.8 (Problem)	169
16.9 (Problem) nonlinear least squares problem	173
16.6.1 (Theorem) Quadratic Convergence	174
16.6.1 (Definition) affine approximation of $f(x) = \nabla g(x)$ around $x^{(k)}$	175
16.6.2 (Definition) quadratic approximation of $g(x)$ around $x^{(k)}$	175
16.7.1 (Theorem)	176
16.7.2 (Theorem)	176
16.7.3 (Theorem)	176
16.7.4 (Theorem)	176
16.7.5 (Theorem)	176
16.7.6 (Corollary)	177
16.10 (Problem)	177
16.9.1 (Definition) globally optimal	177
16.9.2 (Definition) locally optimal	177
Chapter 17 Fourier Series, Fourier Transform	179
17.1.1 (Definition) 波	179
17.1.2 (Definition) Complex Numbers	179
17.1.3 (Definition) Complex Number in Polar Coordinates	179
17.1.1 (Theorem) Euler's Formular	179
17.1.2 (Corollary)	179
17.1.4 (Definition) 正弦函数	179
17.1.5 (Definition) 三角级数	180

17.1.6 (Definition) 三角函数系 (基波)	180
17.1.7 (Definition) 复数域的 Gram 矩阵	180
17.1.8 (Definition)酉矩阵	181
17.1.9 (Definition) 酉矩阵的逆	181
17.2.1 (Definition) 傅里叶级数	181
17.2.1 (Theorem) 收敛定理, 狄利克雷 (Dirichlet) 充分条件	182
17.2.2 (Corollary)	182
17.2.2 (Definition) 正弦级数	182
17.2.3 (Definition) 余弦级数	182
17.2.3 (Theorem)	183
17.2.4 (Definition) 傅里叶级数的复数形式	183
17.3.1 (Definition) The Fourier transform of a continuous function $f(t)$ of a continuous variable t	184
17.3.2 (Definition) inverse Fourier transform	185
17.4.1 (Definition) Discrete Fourier Transform	185
17.4.2 (Definition) inverse discrete Fourier transform (IDFT)	185
17.4.3 (Definition) 离散傅里叶变换矩阵 W	185
17.4.1 (Corollary)	185
17.4.2 (Corollary)	185
17.4.3 (Corollary)	185
17.4.4 (Corollary)	186
17.4.4 (Definition) 离散傅里叶反变换	186
19.4.1 (Theorem) 泊松定理	191
19.8.1 (Theorem) 切比雪夫不等式	192
19.8.2 (Corollary)	193
19.8.3 (Theorem)	193
20.1.1 (Theorem) 辛钦大数定律	194
20.1.2 (Corollary)	194
20.2.1 (Theorem) 伯努利大数定理	194
20.3.1 (Theorem) Lyapunov 中心极限定理	195
20.3.2 (Theorem)	195
20.3.3 (Theorem) De Moivre-Laplace 定理	195
Chapter 21 Sampling Distribution	196
21.2.1 (Definition) 样本平均值	196
21.2.2 (Definition) 样本方差	196
21.2.3 (Definition) 样本标准差	196
21.2.4 (Definition) 样本 k 阶原点矩	196
21.2.5 (Definition) 样本 k 阶中心矩	197
21.2.6 (Definition) X 和 Y 的 $k+p$ 阶混合原点矩	197
21.2.7 (Definition) X 和 Y 的 $k+l$ 阶混合中心矩	197

21.3.1 (Definition) 卡方分布	197
21.3.1 (Corollary)	197
21.3.2 (Corollary) χ^2 分布的可加性	197
21.3.3 (Corollary) χ^2 分布的数学期望和方差	198
21.3.4 (Corollary) 卡方分布中心极限定理	198
21.3.5 (Corollary)	198
21.3.2 (Definition) 卡方分布的上分位点	198
21.4.1 (Definition) 自由度为 t 的 t 分布	198
21.4.1 (Corollary) t 分布数学期望和方差	198
21.4.2 (Corollary) t 分布的概率密度函数	199
21.4.3 (Corollary)	199
21.4.4 (Corollary) 与 F 分布的关系	199
21.4.2 (Definition) t 分布的上分位点	199
21.5.1 (Definition) F 分布	199
21.5.1 (Corollary) F 分布数学期望	199
21.5.2 (Corollary) $F(n_1, n_2), F(n_2, n_1)$ 的关系	200
21.5.3 (Corollary) 与 t 分布的关系	200
21.5.4 (Corollary) F 分布上 分位点的性质	200
21.6.1 (Definition) 正态总体的样本均值的数学期望、方差和样本方差的数学期望	200
21.6.1 (Corollary) 矩估计法原理	200
21.6.2 (Theorem) 样本均值的分布	200
21.6.3 (Corollary)	201
21.6.4 (Theorem) 样本方差的分布	201
21.6.5 (Theorem) 样本均值和样本方差的关系	201
21.6.6 (Theorem) 两总体样本均值差、样本方差比的分布	201
22.5.1 (Theorem) Kolmogorov 定理	205
Chapter 22 Hypothesis Testing	207
22.7.1 (Definition)	208