Linear Algebra Review and Reference

Zico Kolter

October 16, 2007

1 Basic Concepts and Notation

基本概念和符号

Linear algebra provides a way of compactly representing and operating on sets of linear equations. For example, consider the following system of equations:

$$\begin{array}{rcrcr} 4x_1 & - & 5x_2 & = & -13 \\ -2x_1 & + & 3x_2 & = & 9 \end{array}.$$

This is two equations and two variables, so as you know from high school algebra, you can find a unique solution for x_1 and x_2 (unless the equations are somehow degenerate, for example if the second equation is simply a multiple of the first, but in the case above there is in fact a unique solution). In matrix notation, we can write the system more compactly as:

$$Ax = b$$

$$\text{with } A = \begin{bmatrix} 4 & -5 \\ -2 & 3 \end{bmatrix}, \; b = \begin{bmatrix} 13 \\ -9 \end{bmatrix}.$$

As we will see shortly, there are many advantages (including the obvious space savings) to analyzing linear equations in this form.

1.1 Basic Notation

We use the following notation:

通常 A 代表一个矩阵，x 代表列向量，xT 代表行向量

- By $A \in \mathbb{R}^{m \times n}$ we denote a matrix with m rows and n columns, where the entries of A are real numbers.

- By $x \in \mathbb{R}^n$, we denote a vector with n entries. Usually a vector x will denote a ***column vector*** — i.e., a matrix with n rows and 1 column. If we want to explicitly represent a ***row vector*** — a matrix with 1 row and n columns — we typically write x^T (here x^T denotes the transpose of x, which we will define shortly).

- The ith element of a vector x is denoted x_i:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}.$$

- We use the notation a_{ij} (or A_{ij}, $A_{i,j}$, etc) to denote the entry of A in the ith row and jth column:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}.$$

- We denote the jth column of A by a_j or $A_{:,j}$: 怎么定义矩阵的第 j 列的

$$A = \begin{bmatrix} | & | & & | \\ a_1 & a_2 & \cdots & a_n \\ | & | & & | \end{bmatrix}.$$

- We denote the ith row of A by a_i^T or $A_{i,:}$: 怎么定义矩阵的第 i 行的

$$A = \begin{bmatrix} — & a_1^T & — \\ — & a_2^T & — \\ & \vdots & \\ — & a_m^T & — \end{bmatrix}.$$

- Note that these definitions are ambiguous (for example, the a_1 and a_1^T in the previous two definitions are *not* the same vector). Usually the meaning of the notation should be obvious from its use.

2 Matrix Multiplication

矩阵乘法

The product of two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{n\times p}$ is the matrix

$$C = AB \in \mathbb{R}^{m\times p},$$

where

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

这里给的是点积的定义

Note that in order for the matrix product to exist, the number of columns in A must equal the number of rows in B. There are many ways of looking at matrix multiplication, and we'll start by examining a few special cases.

2.1 Vector-Vector Products 向量与向量相乘，点积

Given two vectors $x, y \in \mathbb{R}^n$, the quantity $x^T y$, sometimes called the ***inner product*** or ***dot product*** of the vectors, is a real number given by

$$x^T y \in \mathbb{R} = \sum_{i=1}^{n} x_i y_i.$$

Note that it is always the case that $x^T y = y^T x$.

Given vectors $x \in \mathbb{R}^m$, $y \in \mathbb{R}^n$ (they no longer have to be the same size), xy^T is called the ***outer product*** of the vectors. It is a matrix whose entries are given by $(xy^T)_{ij} = x_i y_j$, i.e.,

$$xy^T \in \mathbb{R}^{m \times n} = \begin{bmatrix} x_1 y_1 & x_1 y_2 & \cdots & x_1 y_n \\ x_2 y_1 & x_2 y_2 & \cdots & x_2 y_n \\ \vdots & \vdots & \ddots & \vdots \\ x_m y_1 & x_m y_2 & \cdots & x_m y_n \end{bmatrix}.$$

2.2 Matrix-Vector Products 矩阵乘上向量，左乘，右乘，左乘是指矩阵写在左边，右乘是指矩阵写在右边

Given a matrix $A \in \mathbb{R}^{m \times n}$ and a vector $x \in \mathbb{R}^n$, their product is a vector $y = Ax \in \mathbb{R}^m$. There are a couple ways of looking at matrix-vector multiplication, and we will look at them both.

If we write A by rows, then we can express Ax as,

左乘：对矩阵中的列向量进行线性组合
右乘：对矩阵中的行向量进行线性组合

$$y = \begin{bmatrix} — & a_1^T & — \\ — & a_2^T & — \\ & \vdots & \\ — & a_m^T & — \end{bmatrix} x = \begin{bmatrix} a_1^T x \\ a_2^T x \\ \vdots \\ a_m^T x \end{bmatrix}.$$

In other words, the ith entry of y is equal to the inner product of the ith *row* of A and x, $y_i = a_i^T x$.

Alternatively, lets write A in column form. In this case we see that,

$$y = \begin{bmatrix} | & | & & | \\ a_1 & a_2 & \cdots & a_n \\ | & | & & | \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} a_1 \end{bmatrix} x_1 + \begin{bmatrix} a_2 \end{bmatrix} x_2 + \ldots + \begin{bmatrix} a_n \end{bmatrix} x_n .$$

线性组合

In other words, y is a ***linear combination*** of the *columns* of A, where the coefficients of the linear combination are given by the entries of x.

So far we have been multiplying on the right by a column vector, but it is also possible to multiply on the left by a row vector. This is written, $y^T = x^T A$ for $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^m$, and $y \in \mathbb{R}^n$. As before, we can express y^T in two obvious ways, depending on whether we express A in terms on its rows or columns. In the first case we express A in terms of its columns, which gives

$$y^T = x^T \begin{bmatrix} | & | & & | \\ a_1 & a_2 & \cdots & a_n \\ | & | & & | \end{bmatrix} = \begin{bmatrix} x^T a_1 & x^T a_2 & \cdots & x^T a_n \end{bmatrix}$$

which demonstrates that the ith entry of y^T is equal to the inner product of x and the ith *column* of A.

Finally, expressing A in terms of rows we get the final representation of the vector-matrix product,

$$\begin{aligned} y^T &= \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} — & a_1^T & — \\ — & a_2^T & — \\ & \vdots & \\ — & a_m^T & — \end{bmatrix} \\ &= x_1 \begin{bmatrix} — & a_1^T & — \end{bmatrix} + x_2 \begin{bmatrix} — & a_2^T & — \end{bmatrix} + ... + x_n \begin{bmatrix} — & a_n^T & — \end{bmatrix} \end{aligned}$$

so we see that y^T is a linear combination of the *rows* of A, where the coefficients for the linear combination are given by the entries of x.

2.3 Matrix-Matrix Products 两个矩阵相乘

Armed with this knowledge, we can now look at four different (but, of course, equivalent) ways of viewing the matrix-matrix multiplication $C = AB$ as defined at the beginning of this section. First we can view matrix-matrix multiplication as a set of vector-vector products. The most obvious viewpoint, which follows immediately from the definition, is that the i, j entry of C is equal to the inner product of the ith row of A and the jth row of B. Symbolically, this looks like the following,

$$C = AB = \begin{bmatrix} — & a_1^T & — \\ — & a_2^T & — \\ & \vdots & \\ — & a_m^T & — \end{bmatrix} \begin{bmatrix} | & | & & | \\ b_1 & b_2 & \cdots & b_p \\ | & | & & | \end{bmatrix} = \begin{bmatrix} a_1^T b_1 & a_1^T b_2 & \cdots & a_1^T b_p \\ a_2^T b_1 & a_2^T b_2 & \cdots & a_2^T b_p \\ \vdots & \vdots & \ddots & \vdots \\ a_m^T b_1 & a_m^T b_2 & \cdots & a_m^T b_p \end{bmatrix}.$$

Remember that since $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{n \times p}$, $a_i \in \mathbb{R}^n$ and $b_j \in \mathbb{R}^n$, so these inner products all make sense. This is the most "natural" representation when we represent A by rows and B by columns. Alternatively, we can represent A by columns, and B by rows, which leads to the interpretation of AB as a sum of outer products. Symbolically,

$$C = AB = \begin{bmatrix} | & | & & | \\ a_1 & a_2 & \cdots & a_n \\ | & | & & | \end{bmatrix} \begin{bmatrix} — & b_1^T & — \\ — & b_2^T & — \\ & \vdots & \\ — & b_n^T & — \end{bmatrix} = \sum_{i=1}^{n} a_i b_i^T \ .$$

Put another way, AB is equal to the sum, over all i, of the outer product of the ith column of A and the ith row of B. Since, in this case, $a_i \in \mathbb{R}^m$ and $b_i \in \mathbb{R}^p$, the dimension of the outer product $a_i b_i^T$ is $m \times p$, which coincides with the dimension of C.

Second, we can also view matrix-matrix multiplication as a set of matrix-vector products. Specifically, if we represent B by columns, we can view the columns of C as matrix-vector products between A and the columns of B. Symbolically,

$$C = AB = A \begin{bmatrix} | & | & & | \\ b_1 & b_2 & \cdots & b_p \\ | & | & & | \end{bmatrix} = \begin{bmatrix} | & | & & | \\ Ab_1 & Ab_2 & \cdots & Ab_p \\ | & | & & | \end{bmatrix}.$$

Here the ith column of C is given by the matrix-vector product with the vector on the right, $c_i = Ab_i$. These matrix-vector products can in turn be interpreted using both viewpoints given in the previous subsection. Finally, we have the analogous viewpoint, where we represent A by rows, and view the rows of C as the matrix-vector product between the rows of A and C. Symbolically,

$$C = AB = \begin{bmatrix} \text{—} & a_1^T & \text{—} \\ \text{—} & a_2^T & \text{—} \\ & \vdots & \\ \text{—} & a_m^T & \text{—} \end{bmatrix} B = \begin{bmatrix} \text{—} & a_1^T B & \text{—} \\ \text{—} & a_2^T B & \text{—} \\ & \vdots & \\ \text{—} & a_m^T B & \text{—} \end{bmatrix}.$$

Here the ith row of C is given by the matrix-vector product with the vector on the left, $c_i^T = a_i^T B$.

It may seem like overkill to dissect matrix multiplication to such a large degree, especially when all these viewpoints follow immediately from the initial definition we gave (in about a line of math) at the beginning of this section. However, virtually all of linear algebra deals with matrix multiplications of some kind, and it is worthwhile to spend some time trying to develop an intuitive understanding of the viewpoints presented here.

In addition to this, it is useful to know a few basic properties of matrix multiplication at a higher level:

- Matrix multiplication is associative: $(AB)C = A(BC)$.
- Matrix multiplication is distributive: $A(B + C) = AB + AC$.
- Matrix multiplication is, in general, *not* commutative; that is, it can be the case that $AB \neq BA$. 注意不满足交换律，将矩阵乘法看成矩阵函数的话，ABx 蕴含着函数的复合顺序，先复合谁是一定的，不能颠倒。

3 Operations and Properties

In this section we present several operations and properties of matrices and vectors. Hopefully a great deal of this will be review for you, so the notes can just serve as a reference for these topics.

3.1 The Identity Matrix and Diagonal Matrices 单位阵，对角阵

The ***identity matrix***, denoted $I \in \mathbb{R}^{n\times n}$, is a square matrix with ones on the diagonal and zeros everywhere else. That is,

$$I_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

It has the property that for all $A \in \mathbb{R}^{m\times n}$,

$$AI = A = IA$$

where the size of I is determined by the dimensions of A so that matrix multiplication is possible.

A ***diagonal matrix*** is a matrix where all non-diagonal elements are 0. This is typically denoted $D = \mathrm{diag}(d_1, d_2, \ldots, d_n)$, with

$$D_{ij} = \begin{cases} d_i & i = j \\ 0 & i \neq j \end{cases}$$

Clearly, $I = \mathrm{diag}(1, 1, \ldots, 1)$.

3.2 The Transpose 转置

The ***transpose*** of a matrix results from "flipping" the rows and columns. Given a matrix $A \in \mathbb{R}^{m\times n}$, is transpose, written A^T, is defined as

$$A^T \in \mathbb{R}^{n\times m},\ (A^T)_{ij} = A_{ji}\ \ .$$

We have in fact already been using the transpose when describing row vectors, since the transpose of a column vector is naturally a row vector.

The following properties of transposes are easily verified:

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

3.3 Symmetric Matrices 对称阵

反对称阵：对角线上的元素肯定全为 0

A square matrix $A \in \mathbb{R}^{n\times n}$ is ***symmetric*** if $A = A^T$. It is ***anti-symmetric*** if $A = -A^T$. It is easy to show that for any matrix $A \in \mathbb{R}^{n\times n}$, the matrix $A + A^T$ is symmetric and the matrix $A - A^T$ is anti-symmetric. From this it follows that any square matrix $A \in \mathbb{R}^{n\times n}$ can be represented as a sum of a symmetric matrix and an anti-symmetric matrix, since

$$A = \frac{1}{2}(A + A^T) + \frac{1}{2}(A - A^T)$$

and the first matrix on the right is symmetric, while the second is anti-symmetric. It turns out that symmetric matrices occur a great deal in practice, and they have many nice properties which we will look at shortly. It is common to denote the set of all symmetric matrices of size n as $\mathbb{S}^n$, so that $A \in \mathbb{S}^n$ means that A is a symmetric $n \times n$ matrix;

3.4 The Trace 迹

对角线元素的和

The ***trace*** of a square matrix $A \in \mathbb{R}^{n\times n}$, denoted $\mathrm{tr}(A)$ (or just $\mathrm{tr}A$ if the parentheses are obviously implied), is the sum of diagonal elements in the matrix:

$$\mathrm{tr}A = \sum_{i=1}^{n} A_{ii}.$$

As described in the CS229 lecture notes, the trace has the following properties (included here for the sake of completeness):

- For $A \in \mathbb{R}^{n\times n}$, $\mathrm{tr}A = \mathrm{tr}A^T$. 转置以后对角线元素不变
- For $A, B \in \mathbb{R}^{n\times n}$, $\mathrm{tr}(A + B) = \mathrm{tr}A + \mathrm{tr}B$. A + B，矩阵加法
- For $A \in \mathbb{R}^{n\times n}$, $t \in \mathbb{R}$, $\mathrm{tr}(tA) = t\,\mathrm{tr}A$. 矩阵数乘
- For A, B such that AB is square, $\mathrm{tr}AB = \mathrm{tr}BA$. 矩阵乘法
- For A, B, C such that ABC is square, $\mathrm{tr}ABC = \mathrm{tr}BCA = \mathrm{tr}CAB$, and so on for the product of more matrices.

3.5 Norms 范数

A ***norm*** of a vector $\|x\|$ is informally measure of the "length" of the vector. For example, we have the commonly-used Euclidean or ℓ_2 norm,

https://www.zhihu.com/question/20473040

$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}.$$

二范数，也可以不写下标 ||x||

Note that $\|x\|_2^2 = x^T x$.

More formally, a norm is any function $f : \mathbb{R}^n \to \mathbb{R}$ that satisfies 4 properties:

1. For all $x \in \mathbb{R}^n$, $f(x) \geq 0$ (non-negativity).
2. $f(x) = 0$ if and only if $x = 0$ (definiteness). 有且只有 0 向量的范数为0
3. For all $x \in \mathbb{R}^n$, $t \in \mathbb{R}$, $f(tx) = |t|f(x)$ (homogeneity).
4. For all $x, y \in \mathbb{R}^n$, $f(x + y) \leq f(x) + f(y)$ (triangle inequality).

Other examples of norms are the ℓ_1 norm,

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|$$ L1 范数

and the ℓ_∞ norm,

$$\|x\|_\infty = \max_i |x_i|.$$ 无穷范数

In fact, all three norms presented so far are examples of the family of ℓ_p norms, which are parameterized by a real number $p \geq 1$, and defined as

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

Norms can also be defined for matrices, such as the Frobenius norm,

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} A_{ij}^2} = \sqrt{\operatorname{tr}(A^T A)}.$$ 矩阵范数的定义

Many other norms exist, but they are beyond the scope of this review.

3.6 Linear Independence and Rank 线性无关 秩

A set of vectors $\{x_1, x_2, \ldots x_n\}$ is said to be ***(linearly) independent*** if no vector can be represented as a linear combination of the remaining vectors. Conversely, a vector which *can* be represented as a linear combination of the remaining vectors is said to be ***(linearly) dependent***. For example, if

xn 可以用 x1,x2,...,xn-1 表示出来，所以这组向量是线性相关的

$$x_n = \sum_{i=1}^{n-1} \alpha_i x_i$$

一组向量中，任何一个向量都不能用其他向量表示出来，那就可以说这组向量线性无关

for some $\{\alpha_1, \ldots, \alpha_{n-1}\}$ then x_n is dependent on $\{x_1, \ldots, x_{n-1}\}$; otherwise, it is independent of $\{x_1, \ldots, x_{n-1}\}$. 列秩 行秩 = 列秩 = 秩

The ***column rank*** of a matrix A is the largest number of columns of A that constitute linearly independent set. This is often referred to simply as the number of linearly independent columns, but this terminology is a little sloppy, since it is possible that any vector in some set $\{x_1, \ldots x_n\}$ can be expressed as a linear combination of the remaining vectors, even though some subset of the vectors might be independent. In the same way, the ***row rank*** 行秩 is the largest number of rows of A that constitute a linearly independent set.

It is a basic fact of linear algebra, that for any matrix A, $\operatorname{columnrank}(A) = \operatorname{rowrank}(A)$, and so this quantity is simply refereed to as the ***rank*** of A, denoted as $\operatorname{rank}(A)$. The following are some basic properties of the rank:

矩阵的秩的一些性质

- For $A \in \mathbb{R}^{m \times n}$, $\operatorname{rank}(A) \leq \min(m, n)$. If $\operatorname{rank}(A) = \min(m, n)$, then A is said to be ***full rank***. 满秩矩阵

- For $A \in \mathbb{R}^{m\times n}$, $\text{rank}(A) = \text{rank}(A^T)$. AB = A * B（将 AB 看成一个整体，则说明AB可以用A中的向量线性表示，也可以用B中的向量线性表示，所以AB的秩不可能比A 或 B 大）
- For $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{n\times p}$, $\text{rank}(AB) \le \min(\text{rank}(A), \text{rank}(B))$.
- For $A, B \in \mathbb{R}^{m\times n}$, $\text{rank}(A + B) \le \text{rank}(A) + \text{rank}(B)$.

3.7 The Inverse 逆矩阵

The ***inverse*** of a square matrix $A \in \mathbb{R}^{n\times n}$ is denoted A^{-1}, and is the unique matrix such that

$$A^{-1}A = I = AA^{-1}.$$

可逆阵，非奇异阵　非可逆阵，奇异阵

It turns out that A^{-1} may not exist for some matrices A; we say A is ***invertible*** or ***non-singular*** if A^{-1} exists and ***non-invertible*** or ***singular*** otherwise. One condition for invertibility we already know: it is possible to show that A^{-1} exists if and only if A is full rank. We will soon see that there are many alternative sufficient and necessary conditions, in addition to full rank, for invertibility. The following are properties of the inverse; all assume that $A, B \in \mathbb{R}^{n\times n}$ are non-singular:

如果一个方阵是满秩矩阵，那么这个方阵是可逆的

- $(A^{-1})^{-1} = A$
- If $Ax = b$, we can multiply by A^{-1} on both sides to obtain $x = A^{-1}b$. This demonstrates the inverse with respect to the original system of linear equalities we began this review with.
- $(AB)^{-1} = B^{-1}A^{-1}$
- $(A^{-1})^T = (A^T)^{-1}$. For this reason this matrix is often denoted A^{-T}.

3.8 Orthogonal Matrices 正交矩阵

方阵 A 为正交阵的充分必要条件是 A 的列向量都是单位向量，且两两正交

两向量正交
标准化向量（归一化的）

Two vectors $x, y \in \mathbb{R}^n$ are ***orthogonal*** if $x^Ty = 0$. A vector $x \in \mathbb{R}^n$ is ***normalized*** if $\|x\|_2 = 1$. A square matrix $U \in \mathbb{R}^{n\times n}$ is ***orthogonal*** (note the different meanings when talking about vectors versus matrices) if all its columns are orthogonal to each other and are normalized (the columns are then referred to as being ***orthonormal***).

It follows immediately from the definition of orthogonality and normality that

$$U^TU = I = UU^T.$$

如果一个矩阵是正交阵，那么这个矩阵的逆矩阵是他的转置

In other words, the inverse of an orthogonal matrix is its transpose. Note that if U is not square — i.e., $U \in \mathbb{R}^{m\times n}$, $n < m$ — but its columns are still orthonormal, then $U^TU = I$, but $UU^T \ne I$. We generally only use the term orthogonal to describe the previous case, where U is square.

Another nice property of orthogonal matrices is that operating on a vector with an orthogonal matrix will not change its Euclidean norm, i.e.,

$$\|Ux\|_2 = \|x\|_2$$

for any $x \in \mathbb{R}^n$, $U \in \mathbb{R}^{n\times n}$ orthogonal.

3.9 Range and Nullspace of a Matrix 向量空间和 0 向量空间

向量空间

The ***span*** of a set of vectors $\{x_1, x_2, \ldots x_n\}$ is the set of all vectors that can be expressed as a linear combination of $\{x_1, \ldots, x_n\}$. That is,

$$\operatorname{span}(\{x_1, \ldots x_n\}) = \left\{ v : v = \sum_{i=1}^{n} \alpha_i x_i, \quad \alpha_i \in \mathbb{R} \right\}.$$

It can be shown that if $\{x_1, \ldots, x_n\}$ is a set of n linearly independent vectors, where each $x_i \in \mathbb{R}^n$, then $\operatorname{span}(\{x_1, \ldots x_n\}) = \mathbb{R}^n$. In other words, *any* vector $v \in \mathbb{R}^n$ can be written as a linear combination of x_1 through x_n. The ***projection*** of a vector $y \in \mathbb{R}^m$ onto the span of $\{x_1, \ldots, x_n\}$ (here we assume $x_i \in \mathbb{R}^m$) is the vector $v \in \operatorname{span}(\{x_1, \ldots x_n\})$, such that v as close as possible to y, as measured by the Euclidean norm $\|v - y\|_2$. We denote the projection as $\operatorname{Proj}(y; \{x_1, \ldots, x_n\})$ and can define it formally as,

$$\operatorname{Proj}(y; \{x_1, \ldots x_n\}) = \operatorname{argmin}_{v \in \operatorname{span}(\{x_1, \ldots, x_n\})} \|y - v\|_2.$$

The ***range*** (sometimes also called the columnspace) of a matrix $A \in \mathbb{R}^{m \times n}$, denoted $\mathcal{R}(A)$, is the the span of the columns of A. In other words,

$$\mathcal{R}(A) = \{v \in \mathbb{R}^m : v = Ax, x \in \mathbb{R}^n\}.$$

Making a few technical assumptions (namely that A is full rank and that $n < m$), the projection of a vector $y \in \mathbb{R}^m$ onto the range of A is given by,

$$\operatorname{Proj}(y; A) = \operatorname{argmin}_{v \in \mathcal{R}(A)} \|v - y\|_2 = A(A^T A)^{-1} A^T y \ .$$

This last equation should look extremely familiar, since it is almost the same formula we derived in class (and which we will soon derive again) for the least squares estimation of parameters. Looking at the definition for the projection, it should not be too hard to convince yourself that this is in fact the same objective that we minimized in our least squares problem (except for a squaring of the norm, which doesn't affect the optimal point) and so these problems are naturally very connected. When A contains only a single column, $a \in \mathbb{R}^m$, this gives the special case for a projection of a vector on to a line:

$$\operatorname{Proj}(y; a) = \frac{a a^T}{a^T a} y \ .$$

The ***nullspace*** of a matrix $A \in \mathbb{R}^{m \times n}$, denoted $\mathcal{N}(A)$ is the set of all vectors that equal 0 when multiplied by A, i.e.,

$$\mathcal{N}(A) = \{x \in \mathbb{R}^n : Ax = 0\}.$$

Note that vectors in $\mathcal{R}(A)$ are of size m, while vectors in the $\mathcal{N}(A)$ are of size n, so vectors in $\mathcal{R}(A^T)$ and $\mathcal{N}(A)$ are both in $\mathbb{R}^n$. In fact, we can say much more. It turns out that

$$\left\{ w : w = u + v, u \in \mathcal{R}(A^T), v \in \mathcal{N}(A) \right\} = \mathbb{R}^n \text{ and } \mathcal{R}(A^T) \cap \mathcal{N}(A) = \emptyset \ .$$

In other words, $\mathcal{R}(A^T)$ and $\mathcal{N}(A)$ are disjoint subsets that together span the entire space of $\mathbb{R}^n$. Sets of this type are called ***orthogonal complements***, and we denote this $\mathcal{R}(A^T) = \mathcal{N}(A)^\perp$.

3.10 The Determinant 行列式

A 是一个 n 阶方阵，行列式是把 n 阶方阵映射到实数集的一个函数

The ***determinant*** of a square matrix $A \in \mathbb{R}^{n\times n}$, is a function $\det : \mathbb{R}^{n\times n} \to \mathbb{R}$, and is denoted $|A|$ or $\det A$ (like the trace operator, we usually omit parentheses). The full formula for the determinant gives little intuition about its meaning, so we instead first give three defining properties of the determinant, from which all the rest follow (including the general formula):

1. The determinant of the identity is 1, $|I| = 1$.

2. Given a matrix $A \in \mathbb{R}^{n\times n}$, if we multiply a single row in A by a scalar $t \in \mathbb{R}$, then the determinant of the new matrix is $t|A|$,

   倍乘

$$\left| \begin{bmatrix} — & t\ a_1^T & — \\ — & a_2^T & — \\ & \vdots & \\ — & a_m^T & — \end{bmatrix} \right| = t|A| \quad .$$

3. If we exchange any two rows a_i^T and a_j^T of A, then the determinant of the new matrix is $-|A|$, for example

$$\left| \begin{bmatrix} — & a_2^T & — \\ — & a_1^T & — \\ & \vdots & \\ — & a_m^T & — \end{bmatrix} \right| = -|A| \quad .$$

   互换

These properties, however, also give very little intuition about the nature of the determinant, so we now list several properties that follow from the three properties above:

- For $A \in \mathbb{R}^{n\times n}$, $|A| = |A^T|$.
- For $A, B \in \mathbb{R}^{n\times n}$, $|AB| = |A||B|$.

  行列式为0，奇异阵
- For $A \in \mathbb{R}^{n\times n}$, $|A| = 0$ if and only if A is singular (i.e., non-invertible).
- For $A \in \mathbb{R}^{n\times n}$ and A non-singular, $|A|^{-1} = 1/|A|$.

Before given the general definition for the determinant, we define, for $A \in \mathbb{R}^{n\times n}$, $A_{\backslash i,\backslash j} \in \mathbb{R}^{(n-1)\times(n-1)}$ to be the *matrix* that results from deleting the ith row and jth column from A. The general (recursive) formula for the determinant is 计算方式

$$\begin{aligned} |A| &= \sum_{i=1}^{n} (-1)^{i+j} a_{ij} |A_{\backslash i,\backslash j}| \qquad (\text{for any } j \in 1,\ldots,n) \\ &= \sum_{j=1}^{n} (-1)^{i+j} a_{ij} |A_{\backslash i,\backslash j}| \qquad (\text{for any } i \in 1,\ldots,n) \end{aligned}$$

with the initial case that $|A| = a_{11}$ for $A \in \mathbb{R}^{1\times 1}$. If we were to expand this formula completely for $A \in \mathbb{R}^{n\times n}$, there would be a total of $n!$ (n factorial) different terms. For this reason, we hardly even explicitly write the complete equation of the determinant for matrices bigger than 3×3. However, the equations for determinants of matrices up to size 3×3 are fairly common, and it is good to know them:

$$\begin{aligned}
|[a_{11}]| &= a_{11} \\
\left|\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}\right| &= a_{11}a_{22} - a_{12}a_{21} \\
\left|\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}\right| &= \begin{array}{l} a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} \\ \quad -a_{11}a_{23}a_{32} - a_{12}a_{21}a_{33} - a_{13}a_{22}a_{31} \end{array}
\end{aligned}$$

伴随矩阵

The ***classical adjoint*** (often just called the adjoint) of a matrix $A \in \mathbb{R}^{n\times n}$, is denoted $\mathrm{adj}(A)$, and defined as

基本计算方式：去掉对应行列，得到余子式，计算正负号

$$\mathrm{adj}(A) \in \mathbb{R}^{n\times n}, \quad (\mathrm{adj}(A))_{ij} = (-1)^{i+j}|A_{\setminus j,\setminus i}|$$

(note the switch in the indices $A_{\setminus j,\setminus i}$). It can be shown that for any nonsingular $A \in \mathbb{R}^{n\times n}$,

$$A^{-1} = \frac{1}{|A|}\mathrm{adj}(A) \ .$$

用伴随矩阵求逆矩阵
或者说逆矩阵和伴随矩阵的关系（如果逆矩阵存在的话）
逆矩阵 = 伴随阵 / 行列式，行列式是一个数

While this is a nice "explicit" formula for the inverse of matrix, we should note that, numerically, there are in fact much more efficient ways of computing the inverse.

3.11 Quadratic Forms and Positive Semidefinite Matrices

Given a matrix square $A \in \mathbb{R}^{n\times n}$ and a vector $x \in \mathbb{R}$, the scalar value x^TAx is called a ***quadratic form***. Written explicitly, we see that

二次型，给你一个矩阵 A 和一个向量 x， x T Ax 被称为二次型

任给一个二次型就唯一确定一个对称阵
任给一个对称阵也唯一确定一个二次型

$$x^TAx = \sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}x_ix_j \ .$$

Note that,

$$x^TAx = (x^TAx)^T = x^TA^Tx = x^T(\frac{1}{2}A + \frac{1}{2}A^T)x$$

1/2(xT Ax + xT AT x)

i.e., only the symmetric part of A contributes to the quadratic form. For this reason, we often implicitly assume that the matrices appearing in a quadratic form are symmetric.

We give the following definitions:

正定阵

- A symmetric matrix $A \in \mathbb{S}^n$ is ***positive definite*** (PD) if for all non-zero vectors $x \in \mathbb{R}^n$, $x^TAx > 0$. This is usually denoted $A \succ 0$ (or just $A > 0$), and often times the set of all positive definite matrices is denoted $\mathbb{S}^n_{++}$.

半正定阵

- A symmetric matrix $A \in \mathbb{S}^n$ is ***position semidefinite*** (PSD) if for all vectors $x^T Ax \geq 0$. This is written $A \succeq 0$ (or just $A \geq 0$), and the set of all positive semidefinite matrices is often denoted $\mathbb{S}^n_+$.

负定阵

- Likewise, a symmetric matrix $A \in \mathbb{S}^n$ is ***negative definite*** (ND), denoted $A \prec 0$ (or just $A < 0$) if for all non-zero $x \in \mathbb{R}^n$, $x^T Ax < 0$.

半负定阵

- Similarly, a symmetric matrix $A \in \mathbb{S}^n$ is ***negative semidefinite*** (NSD), denoted $A \succeq 0$ (or just $A \leq 0$) if for all $x \in \mathbb{R}^n$, $x^T Ax \leq 0$.

- Finally, a symmetric matrix $A \in \mathbb{S}^n$ is ***indefinite***, if it is neither positive semidefinite nor negative semidefinite — i.e., if there exists $x_1, x_2 \in \mathbb{R}^n$ such that $x_1^T Ax_1 > 0$ and $x_2^T Ax_2 < 0$.

It should be obvious that if A is positive definite, then $-A$ is negative definite and vice versa. Likewise, if A is positive semidefinite then $-A$ is negative semidefinite and vice versa. If A is indefinite, then so is $-A$. It can also be shown that positive definite and negative definite matrices are always invertible.

Finally, there is one type of positive definite matrix that comes up frequently, and so deserves some special mention. Given any matrix $A \in \mathbb{R}^{m\times n}$ (not necessarily symmetric or even square), the matrix $G = A^T A$ (sometimes called a ***Gram matrix***) is always positive semidefinite. Further, if $m \geq n$ (and we assume for convenience that A is full rank), then $G = A^T A$ is positive definite.

3.12 Eigenvalues and Eigenvectors 特征值和特征向量

Given a square matrix $A \in \mathbb{R}^{n\times n}$, we say that $\lambda \in \mathbb{C}$ is an ***eigenvalue*** of A and $x \in \mathbb{C}^n$ is the corresponding ***eigenvector***[1] if

$$Ax = \lambda x, \quad x \neq 0 \ .$$

Intuitively, this definition means that multiplying A by the vector x results in a new vector that points in the same direction as x, but scaled by a factor λ. Also note that for any eigenvector $x \in \mathbb{C}^n$, and scalar $t \in \mathbb{C}$, $A(cx) = cAx = c\lambda x = \lambda(cx)$, so cx is also an eigenvector. For this reason when we talk about "the" eigenvector associated with λ, we usually assume that the eigenvector is normalized to have length 1 (this still creates some ambiguity, since x and $-x$ will both be eigenvectors, but we will have to live with this).

We can rewrite the equation above to state that (λ, x) is an eigenvalue-eigenvector pair of A if,

$$(\lambda I - A)x = 0, \quad x \neq 0 \ .$$

[1]Note that λ and the entries of x are actually in $\mathbb{C}$, the set of complex numbers, not just the reals; we will see shortly why this is necessary. Don't worry about this technicality for now, you can think of complex vectors in the same way as real vectors.

But $(\lambda I - A)x = 0$ has a non-zero solution to x if and only if $(\lambda I - A)$ has a non-empty nullspace, which is only the case if $(\lambda I - A)$ is singular, i.e.,

$$|(\lambda I - A)| = 0 \ .$$

We can now use the previous definition of the determinant to expand this expression into a (very large) polynomial in λ, where λ will have maximum degree n. We then find the n (possibly complex) roots of this polynomial to find the n eigenvalues $\lambda_1, \ldots, \lambda_n$. To find the eigenvector corresponding to the eigenvalue λ_i, we simply solve the linear equation $(\lambda_i I - A)x = 0$. It should be noted that this is not the method which is actually used in practice to numerically compute the eigenvalues and eigenvectors (remember that the complete expansion of the determinant has $n!$ terms); it is rather a mathematical argument.

The following are properties of eigenvalues and eigenvectors (in all cases assume $A \in \mathbb{R}^{n\times n}$ has eigenvalues $\lambda_i, \ldots, \lambda_n$ and associated eigenvectors $x_1, \ldots x_n$):

- The trace of a A is equal to the sum of its eigenvalues, 一个方阵的迹就是特征值之和

$$\mathrm{tr}A = \sum_{i=1}^{n} \lambda_i \ .$$

- The determinant of A is equal to the product of its eigenvalues,

$$|A| = \prod_{i=1}^{n} \lambda_i \ .$$

方阵对应的行列式就是特征值的乘积

- The rank of A is equal to the number of non-zero eigenvalues of A. 方阵的秩等于非0特征值的个数
- If A is non-singular then $1/\lambda_i$ is an eigenvalue of A^{-1} with associated eigenvector x_i, i.e., $A^{-1}x_i = (1/\lambda_i)x_i$.
- The eigenvalues of a diagonal matrix $D = \mathrm{diag}(d_1, \ldots d_n)$ are just the diagonal entries $d_1, \ldots d_n$. 方阵的对角阵 不是所有方阵都能对角化，一个n阶方阵能否对角化，取决于这个方阵是不是有 n 个线性无关的特征向量，特别的如果 A 有 n 个特征值互不相等，那么A一定可以对角化

We can write all the eigenvector equations simultaneously as

对称真一定存在一个正交矩阵可以将其对角化

$$AX = X\Lambda$$

where the columns of $X \in \mathbb{R}^{n\times n}$ are the eigenvectors of A and Λ is a diagonal matrix whose entries are the eigenvalues of A, i.e.,

$$X \in \mathbb{R}^{n\times n} = \begin{bmatrix} | & | & & | \\ x_1 & x_2 & \cdots & x_n \\ | & | & & | \end{bmatrix}, \quad \Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n) \ .$$

If the eigenvectors of A are linearly independent, then the matrix X will be invertible, so $A = X\Lambda X^{-1}$. A matrix that can be written in this form is called ***diagonalizable***.

对称矩阵的特征值和特征向量

3.13 Eigenvalues and Eigenvectors of Symmetric Matrices

Two remarkable properties come about when we look at the eigenvalues and eigenvectors of a symmetric matrix $A \in \mathbb{S}^n$. First, it can be shown that all the eigenvalues of A are real. Secondly, the eigenvectors of A are orthonormal, i.e., the matrix X defined above is an orthogonal matrix (for this reason, we denote the matrix of eigenvectors as U in this case). We can therefore represent A as $A = U\Lambda U^T$, remembering from above that the inverse of an orthogonal matrix is just its transpose.

Using this, we can show that the definiteness of a matrix depends entirely on the sign of its eigenvalues. Suppose $A \in \mathbb{S}^n = U\Lambda U^T$. Then

$$x^T A x = x^T U \Lambda U^T x = y^T \Lambda y = \sum_{i=1}^{n} \lambda_i y_i^2$$

where $y = U^T x$ (and since U is full rank, any vector $y \in \mathbb{R}^n$ can be represented in this form). Because y_i^2 is always positive, the sign of this expression depends entirely on the λ_i's. If all $\lambda_i > 0$, then the matrix is positive definite; if all $\lambda_i \geq 0$, it is positive semidefinite. Likewise, if all $\lambda_i < 0$ or $\lambda_i \leq 0$, then A is negative definite or negative semidefinite respectively. Finally, if A has both positive and negative eigenvalues, it is indefinite.

An application where eigenvalues and eigenvectors come up frequently is in maximizing some function of a matrix. In particular, for a matrix $A \in \mathbb{S}^n$, consider the following maximization problem,

$$\max_{x \in \mathbb{R}^n} \ x^T A x \quad \text{subject to } \|x\|_2^2 = 1$$

i.e., we want to find the vector (of norm 1) which maximizes the quadratic form. Assuming the eigenvalues are ordered as $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$, the optimal x for this optimization problem is x_1, the eigenvector corresponding to λ_1. In this case the maximal value of the quadratic form is λ_1. Similarly, the optimal solution to the minimization problem,

$$\min_{x \in \mathbb{R}^n} \ x^T A x \quad \text{subject to } \|x\|_2^2 = 1$$

is x_n, the eigenvector corresponding to λ_n, and the minimal value is λ_n. This can be proved by appealing to the eigenvector-eigenvalue form of A and the properties of orthogonal matrices. However, in the next section we will see a way of showing it directly using matrix calculus.

4 Matrix Calculus

While the topics in the previous sections are typically covered in a standard course on linear algebra, one topic that does not seem to be covered very often (and which we will use extensively) is the extension of calculus to the vector setting. Despite the fact that all the actual calculus we use is relatively trivial, the notation can often make things look much more difficult than they are. In this section we present some basic definitions of matrix calculus and provide a few examples.

4.1 The Gradient 矩阵求梯度

Suppose that $f : \mathbb{R}^{m\times n} \to \mathbb{R}$ is a function that takes as input a matrix A of size $m \times n$ and returns a real value. Then the ***gradient*** of f (with respect to $A \in \mathbb{R}^{m\times n}$) is the matrix of partial derivatives, defined as:

$$\nabla_A f(A) \in \mathbb{R}^{m\times n} = \begin{bmatrix} \frac{\partial f(A)}{\partial A_{11}} & \frac{\partial f(A)}{\partial A_{12}} & \cdots & \frac{\partial f(A)}{\partial A_{1n}} \\ \frac{\partial f(A)}{\partial A_{21}} & \frac{\partial f(A)}{\partial A_{22}} & \cdots & \frac{\partial f(A)}{\partial A_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f(A)}{\partial A_{m1}} & \frac{\partial f(A)}{\partial A_{m2}} & \cdots & \frac{\partial f(A)}{\partial A_{mn}} \end{bmatrix}$$

注意 f 的定义，输入是一个矩阵，输出一个实数，对矩阵的各个位置上的值求偏导

i.e., an $m \times n$ matrix with

$$(\nabla_A f(A))_{ij} = \frac{\partial f(A)}{\partial A_{ij}}.$$

Note that the size of $\nabla_A f(A)$ is always the same as the size of A. So if, in particular, A is just a vector $x \in \mathbb{R}^n$,

$$\nabla_x f(x) = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{bmatrix}.$$

It is very important to remember that the gradient of a function is *only* defined if the function is real-valued, that is, if it returns a scalar value. We can not, for example, take the gradient of $Ax, A \in \mathbb{R}^{n\times n}$ with respect to x, since this quantity is vector-valued.

It follows directly from the equivalent properties of partial derivatives that:

- $\nabla_x(f(x) + g(x)) = \nabla_x f(x) + \nabla_x g(x)$.
- For $t \in \mathbb{R}$, $\nabla_x(t\ f(x)) = t\nabla_x f(x)$.

It is a little bit trickier to determine what the proper expression is for $\nabla_x f(Ax)$, $A \in \mathbb{R}^{n\times n}$, but this is doable as well (if fact, you'll have to work this out for a homework problem).

4.2 The Hessian 海森矩阵

Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ is a function that takes a vector in $\mathbb{R}^n$ and returns a real number. Then the ***Hessian*** matrix with respect to x, written $\nabla_x^2 f(x)$ or simply as H is the $n \times n$ matrix of partial derivatives,

注意 f 的定义，输入是一个向量，输出是一个实数

$$\nabla_x^2 f(x) \in \mathbb{R}^{n\times n} = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \cdots & \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix}.$$

二阶偏导

海森矩阵是对称矩阵

x1x1 x1x2 x1x3 …… x1xn
x2x1 ……

In other words, $\nabla_x^2 f(x) \in \mathbb{R}^{n\times n}$, with

$$(\nabla_x^2 f(x))_{ij} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}.$$

Note that the Hessian is always symmetric, since

$$\frac{\partial^2 f(x)}{\partial x_i \partial x_j} = \frac{\partial^2 f(x)}{\partial x_j \partial x_i}.$$

Similar to the gradient, the Hessian is defined only when $f(x)$ is real-valued.

It is natural to think of the gradient as the analogue of the first derivative for functions of vectors, and the Hessian as the analogue of the second derivative (and the symbols we use also suggest this relation). This intuition is generally correct, but there a few caveats to keep in mind.

First, for real-valued functions of one variable $f : \mathbb{R} \to \mathbb{R}$, it is a basic definition that the second derivative is the derivative of the first derivative, i.e.,

$$\frac{\partial^2 f(x)}{\partial x^2} = \frac{\partial}{\partial x}\frac{\partial}{\partial x} f(x).$$

However, for functions of a vector, the gradient of the function is a vector, and we cannot take the gradient of a vector — i.e.,

$$\nabla_x \nabla_x f(x) = \nabla_x \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \\ \vdots \\ \frac{\partial f(x)}{\partial x_1} \end{bmatrix}$$

and this expression is not defined. Therefore, it is *not* the case that the Hessian is the gradient of the gradient. However, this is *almost* true, in the following sense: If we look at the ith entry of the gradient $(\nabla_x f(x))_i = \partial f(x)/\partial x_i$, and take the gradient with respect to x we get

$$\nabla_x \frac{\partial f(x)}{\partial x_i} = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_i \partial x_1} \\ \frac{\partial^2 f(x)}{\partial x_i \partial x_2} \\ \vdots \\ \frac{\partial f(x)}{\partial x_i \partial x_n} \end{bmatrix}$$

which is the ith column (or row) of the Hessian. Therefore,

$$\nabla_x^2 f(x) = \begin{bmatrix} \nabla_x(\nabla_x f(x))_1 & \nabla_x(\nabla_x f(x))_2 & \cdots & \nabla_x(\nabla_x f(x))_n \end{bmatrix}.$$

If we don't mind being a little bit sloppy we can say that (essentially) $\nabla_x^2 f(x) = \nabla_x(\nabla_x f(x))^T$, so long as we understand that this really means taking the gradient of each entry of $(\nabla_x f(x))^T$, not the gradient of the whole vector.

Finally, note that while we can take the gradient with respect to a matrix $A \in \mathbb{R}^n$, for the purposes of this class we will only consider taking the Hessian with respect to a vector $x \in \mathbb{R}^n$. This is simply a matter of convenience (and the fact that none of the calculations we do require us to find the Hessian with respect to a matrix), since the Hessian with respect to a matrix would have to represent all the partial derivatives $\partial^2 f(A)/(\partial A_{ij} \partial A_{k\ell})$, and it is rather cumbersome to represent this as a matrix.

4.3 Gradients and Hessians of Quadratic and Linear Functions

Now let's try to determine the gradient and Hessian matrices for a few simple functions. It should be noted that all the gradients given here are special cases of the gradients given in the CS229 lecture notes.

For $x \in \mathbb{R}^n$, let $f(x) = b^T x$ for some known vector $b \in \mathbb{R}^n$. Then

$$f(x) = \sum_{i=1}^{n} b_i x_i$$

so

$$\frac{\partial f(x)}{\partial x_k} = \frac{\partial}{\partial x_k} \sum_{i=1}^{n} b_i x_i = b_k.$$

From this we can easily see that $\nabla_x b^T x = b$. This should be compared to the analogous situation in single variable calculus, where $\partial/(\partial x)\ ax = a$.

Now consider the quadratic function $f(x) = x^T A x$ for $A \in \mathbb{S}^n$. Remember that

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} x_i x_j$$

so

$$\frac{\partial f(x)}{\partial x_k} = \frac{\partial}{\partial x_k} \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} x_i x_j = \sum_{i=1}^{n} A_{ik} x_i + \sum_{j=1}^{n} A_{kj} x_j = 2 \sum_{i=1}^{n} A_{ki} x_i$$

对称阵

where the last equality follows since A is symmetric (which we can safely assume, since it is appearing in a quadratic form). Note that the kth entry of $\nabla_x f(x)$ is just the inner product of the kth row of A and x. Therefore, $\nabla_x x^T A x = 2Ax$. Again, this should remind you of the analogous fact in single-variable calculus, that $\partial/(\partial x)\ ax^2 = 2ax$.

Finally, lets look at the Hessian of the quadratic function $f(x) = x^T A x$ (it should be obvious that the Hessian of a linear function $b^T x$ is zero). This is even easier than determining the gradient of the function, since

$$\frac{\partial^2 f(x)}{\partial x_k \partial x_\ell} = \frac{\partial^2}{\partial x_k \partial x_\ell} \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} x_i x_j = A_{k\ell} + A_{\ell k} = 2A_{k\ell}.$$

Therefore, it should be clear that $\nabla_x^2 x^T A x = 2A$, which should be entirely expected (and again analogous to the single-variable fact that $\partial^2/(\partial x^2)\ ax^2 = 2a$).

To recap,

- $\nabla_x b^T x = b$
- $\nabla_x x^T A x = 2Ax$ (if A symmetric)
- $\nabla_x^2 x^T A x = 2A$ (if A symmetric)

A 是对称阵

4.4 Least Squares

最小二乘

Lets apply the equations we obtained in the last section to derive the least squares equations. Suppose we are given matrices $A \in \mathbb{R}^{m\times n}$ (for simplicity we assume A is full rank) and a vector $b \in \mathbb{R}^m$ such that $b \notin \mathcal{R}(A)$. In this situation we will not be able to find a vector $x \in \mathbb{R}^n$, such that $Ax = b$, so instead we want to find a vector x such that Ax is as close as possible to b, as measured by the square of the Euclidean norm $\|Ax - b\|_2^2$.

计算两个向量的欧式距离

Using the fact that $\|x\|_2^2 = x^T x$, we have

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

Taking the gradient with respect to x we have, and using the properties we derived in the previous section

$$\begin{aligned} \nabla_x (x^T A^T A x - 2b^T A x + b^T b) &= \nabla_x x^T A^T A x - \nabla_x 2b^T A x + \nabla_x b^T b \\ &= 2A^T A x - 2A^T b \end{aligned}$$

Setting this last expression equal to zero and solving for x gives the normal equations

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

which is the same as what we derived in class.

4.5 Gradients of the Determinant

Now lets consider a situation where we find the gradient of a function with respect to a matrix, namely for $A \in \mathbb{R}^{n\times n}$, we want to find $\nabla_A |A|$. Recall from our discussion of determinants that

$$|A| = \sum_{i=1}^n (-1)^{i+j} A_{ij} |A_{\backslash i, \backslash j}| \quad \text{(for any } j \in 1, \ldots, n)$$

so

$$\frac{\partial}{\partial A_{k\ell}} |A| = \frac{\partial}{\partial A_{k\ell}} \sum_{i=1}^n (-1)^{i+j} A_{ij} |A_{\backslash i, \backslash j}| = (-1)^{k+\ell} |A_{\backslash k, \backslash \ell}| = (\operatorname{adj}(A))_{\ell k}.$$

From this it immediately follows from the properties of the adjoint that

$$\nabla_A |A| = (\operatorname{adj}(A))^T = |A| A^{-T}.$$

Now lets consider the function $f : \mathbb{S}^n_{++} \to \mathbb{R}$, $f(A) = \log |A|$. Note that we have to restrict the domain of f to be the positive definite matrices, since this ensures that $|A| > 0$, so that the log of $|A|$ is a real number. In this case we can use the chain rule (nothing fancy, just the ordinary chain rule from single-variable calculus) to see that

$$\frac{\partial \log |A|}{\partial A_{ij}} = \frac{\partial \log |A|}{\partial |A|} \frac{\partial |A|}{\partial A_{ij}} = \frac{1}{|A|} \frac{\partial |A|}{\partial A_{ij}}.$$

From this is should be obvious that

$$\nabla_A \log |A| = \frac{1}{|A|} \nabla_A |A| = A^{-1},$$

where we can drop the transpose in the last expression because A is symmetric. Note the similarity to the single-valued case, where $\partial/(\partial x) \log x = 1/x$.

4.6 Eigenvalues as Optimization

Finally, we use matrix calculus to solve an optimization problem in a way that leads directly to eigenvalue/eigenvector analysis. Consider the following, equality constrained optimization problem:

$$\max_{x \in \mathbb{R}^n} \ x^T A x \quad \text{subject to } \|x\|_2^2 = 1$$

for a symmetric matrix $A \in \mathbb{S}^n$. A standard way of solving optimization problems with equality constraints is by forming the ***Lagrangian***, an objective function that includes the equality constraints.[2] The Lagrangian in this case can be given by

$$\mathcal{L}(x, \lambda) = x^T A x - \lambda x^T x$$

where λ is called the Lagrange multiplier associated with the equality constraint. It can be established that for x^* to be a optimal point to the problem, the gradient of the Lagrangian has to be zero at x^* (this is not the only condition, but it is required). That is,

$$\nabla_x \mathcal{L}(x, \lambda) = \nabla_x (x^T A x - \lambda x^T x) = 2A^T x - 2\lambda x = 0.$$

Notice that this is just the linear equation $Ax = \lambda x$. This shows that the only points which can possibly maximize (or minimize) $x^T A x$ assuming $x^T x = 1$ are the eigenvectors of A.

[2] Don't worry if you haven't seen Lagrangians before, as we will cover them in greater detail later in CS229.