Linear Algebra 2

KNU Math 232

Classnotes

Mark Siggers

v. 2019/12/05

ver. 2019/12/05

These notes are for the second semester of a second year class in Linear Algebra based on the International Student's fourth edition of Gilbert Strang's 'Linear Algebra and its applications'. This is referred to as The Text. You SHOULD buy the text. It is full of examples and explanations that we will not have time for; and, I am assigning questions out of the text.

Note:
If you have the 4th edition of the text, but not the International Students 4th edition, then the problem numbers will not be the same. You should make sure you are doing the correct problems.

5 Eigenvectors and eigenvalues

5.1 Introduction

An *eigenvalue* of a matrix M is a constant λ such that

$$Mx = \lambda x$$

for some vector x. The vector x is the corresponding *eigenvector*.

Example 5.1. For the matrix $M = \left[\begin{smallmatrix} 1 & 0 \\ 0 & 3 \end{smallmatrix}\right]$, we have

$$M\left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right] = \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right] \text{ and } M\left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right] = \left[\begin{smallmatrix} 0 \\ 3 \end{smallmatrix}\right] = 3\left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right],$$

so 1 is an eigenvalue with eigenvector $(1, 0)$ and 3 is another eigenvalue with eigenvector $(0, 1)$. There are no more eigenvalues in this example. We will see soon that **an $n \times n$ matrix can have at most n eigenvalues.**

But $M = \left[\begin{smallmatrix} 1 & 0 \\ 0 & 3 \end{smallmatrix}\right]$ does have more eigenvectors. Indeed $M\left[\begin{smallmatrix} 7 \\ 0 \end{smallmatrix}\right] = \left[\begin{smallmatrix} 7 \\ 0 \end{smallmatrix}\right]$. In general, we see that for any matrix M, if $Mx = \lambda x$ and $My = \lambda y$ then

$$M(ax + by) = aMx + bMy = a\lambda x + b\lambda y = \lambda(ax + by).$$

So **the eigenvectors corresponding to a given eigenvalue λ are a subspace**. We call this the *eigenspace* of λ.

So! Does $M = \left[\begin{smallmatrix} 1 & 0 \\ 0 & 3 \end{smallmatrix}\right]$ have any other eigenvectors? Not today. Any other $v \in \mathbb{R}^2$ can be written as $a\left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right] + b\left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right]$, for $a, b \neq 0$. So

$$Mv = M(a\left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right] + b\left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right]) = a \cdot 1\left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right] + b \cdot 3\left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right] = v + 2b\left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right].$$

This is not an eigenvector.

Practice

What are the eigenvectors of the rotation $M = \left[\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right]$?

In this example it was easy to find the eigenvectors by inspection. This is not always the case. What are the eigens (eigenvalues and eigenvectors) of $A = \left[\begin{smallmatrix} 2 & 3 \\ 1 & 4 \end{smallmatrix}\right]$? Well we want v such that $Av = \lambda v$ for some λ. We can write this as the matrix equation $Av = \lambda Iv$, and re-arrange this to get

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

ver. 2019/12/05

So we are looking for λ such that $A - \lambda I$ is singular. Its nullspace is the eigenspace of λ. As a matrix is singular if and only if its determinant is 0, we can find the eigenvalueλ for A as follows.

Example 5.2. To find the eigens of $A = \left[\begin{smallmatrix} 2 & 3 \\ 1 & 4 \end{smallmatrix}\right]$ we write out its *characteristic equation*:

$$0 = \begin{vmatrix} 2 & 3 \\ 1 & 4 \end{vmatrix} - \lambda I = \begin{vmatrix} 2-\lambda & 3 \\ 1 & 4-\lambda \end{vmatrix}.$$

Using our determinant formula to expand the *characteristic polynomial* on the right:

$$\begin{aligned} \begin{vmatrix} 2-\lambda & 3 \\ 1 & 4-\lambda \end{vmatrix} &= (2-\lambda)(4-\lambda) - 3 \\ &= \lambda^2 - 6\lambda + 5 \\ &= (\lambda - 1)(\lambda - 5) \end{aligned}$$

we must solve the quadratic equation $(\lambda - 1)(\lambda - 5) = 0$ to get $\lambda = 1$ or 5.

We usually write the eigenvalues in increasing order as $\lambda_1 = 1$ and $\lambda_2 = 5$.

Now to find the eigenspace of λ_1 we have to find the nullspace of $(A - \lambda_1 I) = \left[\begin{smallmatrix} 1 & 3 \\ 1 & 3 \end{smallmatrix}\right]$. In this simple case, solving $\left[\begin{smallmatrix} 1 & 3 \\ 1 & 3 \end{smallmatrix}\right] x = 0$ is simple by inspection, the solution is $x_1 = -3x_2$, or $x = c(-3, 1)$. So the eigenspace of λ_1 is $\langle(-3, 1)\rangle$. Generally we will find the nullspace by Gaussian Elimination.

Practice

Find the eigenspace of $\lambda_2 = 5$.

The same process works for any square matrix M:

- Find eigenvalues λ_i by solving the characteristic equation $0 = |M - \lambda I|$.
- Find the eigenspace of λ_i by finding the nullspace of $M - \lambda_i I$.

Practice

What are the eigens of the rotation matrix $R = \left[\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right]$? It rotates every vector in $\mathbb{R}^2$ by 90 degrees, so nothing is just scaled. It seems that there should be no eigenvectors, or eigenvalues. But we can still compute a characteristic polynomial. Find its eigens, and see what is happening.

When the matrix M has dimension greater than 3, the characteristic polynomial has degree greater than three, and finding its roots is usually hard to do. We will have to develop a different way to find the eigens for larger matrices. But before we go onto this, we point out a few larger matrices for which the above approach works, or for which we can use ad hoc reasoning.

ver. 2019/12/05

Example 5.3. If M is singular, so the nullspace $N(M)$ is more than just $\{0\}$, any vector v in $N(M)$ is an eigenvector with eigenvalue 0. The dimension of the eigenspace for 0 is the nullity of M.

Example 5.4. If P is the projection matrix projecting on the space $V \subset R^n$, then all vectors in V are eigenvectors with eigenvalue 1. Any vector orthogonal to V is an eigenvector with eigenvalue 0.

Example 5.5. The eigenvalues of a triangular matrix are the diagonal entries. Indeed, if say $M = \begin{bmatrix} 1 & 1 & 2 & 3 \\ & 4 & 2 & 3 \\ & & -2 & 1 \\ & & & 4 \end{bmatrix}$ then

$$|M - \lambda I| = \begin{bmatrix} 1-\lambda & 1 & 2 & 3 \\ & 4-\lambda & 2 & 3 \\ & & -2-\lambda & 1 \\ & & & 4-\lambda \end{bmatrix} = (1-\lambda)(4-\lambda)(-2-\lambda)(4-\lambda),$$

which has roots $1, 4, -2$ and 4.

In this previous example, 4 was a root of the characteristic equation of *multiplicity* two. As the characteristic polynomial of an $n \times n$ matrix has degree n there are exactly n eigenvalues, with multiplicity. (But some might not be real.) We will see that an eigenvalue having multiplicity d means that the corresponding eigenspace has dimension at most d.

As the eigenvalues of a triangular matrix are easy to determine, one might try to use Gaussian elimination to triangulise the matrix and then find the eigen values. Unfortunately, elimination does not preserve eigenvalues– if it did, then every non-singular matrix would have the same eigens as the identity. We will look at a different kind of 'diagonalisation' that preserves eigens.

Problem

Show that $|M|$ is the product of the eigenvalues of a matrix M. Show that the sum of the eigenvalues of M is the sum of the diagonal entries.

Problems from the text

5.1: 1, 3, 6, 7, 9, 10, 12, 17, 18, 24, 28

5.2 Diagonalising a matrix

Let S be an $n \times n$ matrix whose columns are eigenvectors of some $n \times n$ matrix M. We get

$$MS = M\left[\begin{array}{c|c|c|c} x_1 & x_2 & \dots & x_n \end{array}\right] = \left[\begin{array}{c|c|c|c} \lambda_1 x_1 & \lambda_2 x_2 & \dots & \lambda_n x_n \end{array}\right] = S\begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \dots & \\ & & & \lambda_n \end{bmatrix},$$

where λ_i is the eigenvalue for the vector x_i.

ver. 2019/12/05

Writing Λ for the diagonal matrix of eigenvalues on the right, we write this as

$$MS = S\Lambda.$$

If the eigenvectors are independent, then S is invertible and so we have the following decomposition

$$S^{-1}MS = \Lambda$$

which because Λ is diagonal, we call a *diagonalisation of* M. We also say that S diagonalises M (by conjugation). A fast and easy application of such a diagonalisation is that is allows us to quickly compute M^d.

Example 5.6. Where $M = \left[\begin{smallmatrix} 1 & 2 \\ 0 & 3 \end{smallmatrix}\right]$, one can show that $M = \left[\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right]\left[\begin{smallmatrix} 1 & 0 \\ 0 & 3 \end{smallmatrix}\right]\left[\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right]^{-1}$. So

$$M^5 = \left[\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right]\left[\begin{smallmatrix} 1 & 0 \\ 0 & 3 \end{smallmatrix}\right]^5\left[\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right]^{-1} = \left[\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right]\left[\begin{smallmatrix} 1^5 & 0 \\ 0 & 3^5 \end{smallmatrix}\right]\left[\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right]^{-1}.$$

The diagonalisation of a matrix, if it exists, need not be unique, Indeed, in our initial example we can re-order the columns of S or multpliy them by constants, and get a different matrix S that also diagonalises M. But as the i^{th} column of MS is λ_i times the i^{th} column of S, so the the columns of S must be eigenvalues of M.

The multiplicity d of and eigenvalue λ of M as a root of the characteristic polynomial is its *algebraic dimension* and the dimension of its eigenspace is its *geometric dimension*. For M to be diagonalisable, we need that any eigenvalue λ of M of algebraic dimension d has d independent eigenvectors, so has geometric dimension d as well.

Lets observe a couple more properties of eigenvalues. You will be asked to prove simple versions of them.

i. The eigenvectors of M^d are the same as those of M. The eigenvalues of M^d are λ_i^d where λ_i are the eigenvalues of M.

ii. Diagonalisable matrices A and B commute if and only if they are diagonalised by the same matrix S.

Practice

Assuming that M is diagonalisable, show property (i).

Practice

Show property (ii), in the case that all eigenvalues of A are distinct.

Problems from the text

5.2: 1, 4, 8, 11, 14, 20, 23, 27, 28, 31, 44

ver. 2019/12/05

5.3 Difference Equations and Powers M^k

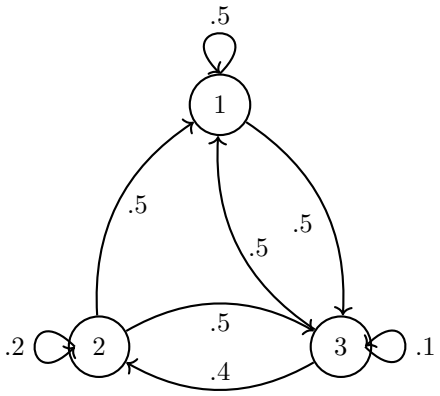
We look at two example applications of eigenvalue and eigenvectors to the field of stochastic processes. In the first we see a Markov process, in the second, the stochastic process is described by a difference equation.

5.3.1 Markov Process

In a Markov process we have a set of n positions, or states, and we let the *state vector*

$$u_t = \begin{bmatrix} p_1(t) \\ p_2(t) \\ \vdots \\ p_n(t) \end{bmatrix}$$

list the probability $p_i(t)$ of being in state i at time t. There is also a rule that tells us the probability of moving from state i to state j in any time-step. For a process with three states, it might look like the following:



But usually it is easier to represent it by a *transition matrix* T such that $Tu_t = u_{t+1}$:

$$\begin{bmatrix} .5 & .3 & .5 \\ 0 & .2 & .4 \\ .5 & .5 & .1 \end{bmatrix} \begin{bmatrix} p_1(t) \\ p_2(t) \\ p_3(t) \end{bmatrix} = \begin{bmatrix} p_1(t+1) \\ p_2(t+1) \\ p_3(t+1) \end{bmatrix}$$

Notice then that $u_t = T^t u_0$ so the whole system is described by T and u_0. The entries of T are non-negative and the columns of each sum to 1. A matrix for which this holds is called *Markov*. In applications we are interested in what happens to u_t as t changes: does the probabillity of being in a certain state go to 1, and the other states 0? Does the probability of being in a state oscillate? converge? This asymptotic behaviour of the system is related to the eigenvalues of T.

ver. 2019/12/05

Using a computer, we have computed the eigenvalues of the matrix T given above:

$$\begin{aligned}
\lambda_1 &= -2/3 & v_1 &= (1, 2, -3) \\
\lambda_2 &= 1/5 & v_2 &= (1, -1, 0) \\
\lambda_3 &= 1 & v_3 &= (1, 5/13, 10/13) = \frac{1}{28}(13, 5, 10)
\end{aligned}$$

Nice! We got an eigenvalue of 1. And what is more, the eigenvector v_3 corresponding to the eigenvalue 1, has all positive entries. These are some useful properties that we will talk about later, but occur more often than you might expect for a Markov matrix. We have scaled the eigenvector so that its entries sum to 1, so can be viewed as probabilities.

If we set this as our position vector:

$$u_t = \frac{1}{28}\begin{bmatrix} 13 \\ 5 \\ 10 \end{bmatrix}$$

then $u_{t+1} = Tu_t = u_t$. That is, the probability of being in a given state is the same for every time t. This state is called a *steady state*.

But what happens if we start at some other state $u_0 \neq v_3$? To see, we first diagonalise T:

$$T = S\Lambda S^{-1} = \begin{bmatrix} 1 & 1 & 1 \\ 5/13 & -1 & 2 \\ 10/13 & 0 & -3 \end{bmatrix} \begin{bmatrix} 1 & & \\ & 1/5 & \\ & & -2/3 \end{bmatrix} \begin{bmatrix} 13/28 & 13/28 & 13/28 \\ 5/12 & -7/12 & -1/4 \\ 5/42 & 5/42 & -3/14 \end{bmatrix}.$$

We then observe that $u_t = T^t u_0 = S\Lambda^t S^{-1} u_0$. Expanding this we get

$$\begin{aligned}
u_t &= \begin{bmatrix} 1 & 1 & 1 \\ 5/13 & -1 & 2 \\ 10/13 & 0 & -3 \end{bmatrix} \begin{bmatrix} 1 & & \\ & 1/5 & \\ & & -2/3 \end{bmatrix}^t \begin{bmatrix} 13/28 & 13/28 & 13/28 \\ 5/12 & -7/12 & -1/4 \\ 5/42 & 5/42 & -3/14 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \\
&= \underbrace{(\frac{13}{28}p_1 + \frac{13}{28}p_2 + \frac{13}{28}p_3)}_{c_1} 1^t \begin{bmatrix} 1 \\ 5/13 \\ 10/13 \end{bmatrix} \\
&+ (\frac{5}{12}p_1 - \frac{7}{12}p_2 - \frac{1}{4}p_3)(1/5)^t \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} \\
&+ (\frac{5}{42}p_1 - \frac{5}{42}p_2 - \frac{3}{14}p_3)(-2/3)^t \begin{bmatrix} 1 \\ 2 \\ -3 \end{bmatrix}.
\end{aligned}$$

Now as t gets big, the last to terms disappear, and so

$$u_t \to c_1 \begin{bmatrix} 1 \\ 5/13 \\ 10/13 \end{bmatrix},$$

ver. 2019/12/05

which is some scale of the eigenvector for $\lambda = 1$. This is for any initial position u_0, different u_0 only change the scalar c_1.

This is not the whole situation, but it is almost.

Recall a square matrix is Markov if it has non-negative entries and all columns sum to 1. The following is known as the Perron-Frobenius Theorem.

Theorem 5.7. *If T is Markov matrix then it has an eigenvalue of 1 and no eigenvalue has greater absolute value. The eigenvector of the eigenvalue 1 can be taken to have no negative entries. If T, or T^k for some positive k has all positive entries, then there is only one eigenvalue with absolute value 1 and it can be taken to have all positive entries.*

Note

The fact that T has a eigenvalue of 1 is easy: as the columns sum to 1, the columns of $T - I$ sum to 0 so are dependent. Thus $\det(T - I) = 0$. The proof of the rest of the theorem is a bit difficult– the proof in the text incomprehensible.

In the case that T has only one eigenvalue of absolute value $\lambda_1 = 1$, then $u_t \to cv_1$ for some constant c. If there are more eigenvalues with absolute value 1, then the situation is a little more complicated. We will discuss this a little more later.

Example 5.8. Each year $\frac{1}{10}$ of the people in Daegu move to Ulsan, and $\frac{2}{10}$ of the people in Ulsan move to Daegu. If there are 1000 people in Korea, (and obviously that all live in one of these two cities,) then what are the stable populations of the two cities? (It is not hard to see that $666\frac{2}{3}$ people live in Daegu and $333\frac{1}{3}$ in Ulsan, but... we use matrices.)

Where the state vector $u_t = \begin{bmatrix} D, U \end{bmatrix}$ tells us the respective proportions D and U of the population that live in Daegu and Ulsan, the transition matrix is the Markov matrix

$$T = \begin{bmatrix} .9 & .2 \\ .1 & .8 \end{bmatrix}.$$

We know it has only one eigenvalue $\lambda_1 = 1$ with absolute value 1, and the eigenvector for this is the solution x_1 to:

$$0 = \begin{bmatrix} .9 - 1 & .2 \\ .1 & .8 - 1 \end{bmatrix} x = \begin{bmatrix} -.1 & .2 \\ .1 & -.2 \end{bmatrix} x.$$

Thus $x_1 = (2, 1)$. Scaling so that the total population is 1000, we get $D = 666\frac{2}{3}$ people living in Daegu, and $333\frac{1}{3}$ in Ulsan.

ver. 2019/12/05

5.3.2 Difference Equations

Consider a recursive description of a population $p(t)$ at time t such as

$$p(0) = 100 \qquad \text{and} \qquad p(t) = p(t-1) \cdot 1.2.$$

We can solve this as

$$p(t) = 100(1.2)^t.$$

This is easy, but it gets more difficult when the recurrence is more complicated.

Example 5.9. The fibonacci recurrence for describing a population of rabbits is

$$F_0 = 0, F_1 = 1, F_2 = 1, \qquad \text{and} \qquad F_{t+2} = F_{t+1} + F_t \text{ for } t \geq 2.$$

To solve this (to find a closed formula for F_t) we look at it as a stochastic process with

$$u_{t+1} = \begin{bmatrix} F_{t+2} \\ F_{t+1} \end{bmatrix} \text{ depending on } u_t = \begin{bmatrix} F_{t+1} \\ F_t \end{bmatrix}.$$

The transmission matrix T should yield

$$T \begin{bmatrix} F_{t+1} \\ F_t \end{bmatrix} = Tu_t = u_{t+1} = \begin{bmatrix} F_{t+2} \\ F_{t+1} \end{bmatrix} = \begin{bmatrix} F_{t+1} + F_t \\ F_{t+1} \end{bmatrix},$$

and so we see that $T = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$.

Now this is not a Markov matrix, so we do not know for sure that there is an eigenvalue of 1, but lets find the eigenvalues anyways.

Solving

$$0 = \begin{vmatrix} 1-\lambda & 1 \\ 1 & -\lambda \end{vmatrix} = \lambda^2 - \lambda - 1 = (\lambda - 1/2)^2 - 5/4$$

gives $\lambda = \frac{1 \pm \sqrt{5}}{2}$. From the second row we get that the eigenvector x_i for λ_i is $(\lambda_i, 1)$. So where $u_0 = (F_1, F_0) = (1, 0)$, we have

$$\begin{aligned} u_t &= T^t u_0 = S\Lambda^t S^{-1} u_0 = \begin{bmatrix} \lambda_1 & \lambda_2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_1^t & \\ & \lambda_2^t \end{bmatrix} \begin{bmatrix} 1 & -\lambda_2 \\ -1 & \lambda_1 \end{bmatrix} \frac{(1,0)}{\lambda_1 - \lambda_2} \\ &= \frac{(1, -\lambda_2) \cdot (1,0)}{\sqrt{5}} \lambda_1^t \begin{bmatrix} \lambda_1 \\ 1 \end{bmatrix} + \frac{(-1, \lambda_1) \cdot (1,0)}{\sqrt{5}} \lambda_2^t \begin{bmatrix} \lambda_2 \\ 1 \end{bmatrix} \\ &= \frac{1}{\sqrt{5}} \lambda_1^t \begin{bmatrix} \lambda_1 \\ 1 \end{bmatrix} - \frac{1}{\sqrt{5}} \lambda_2^t \begin{bmatrix} \lambda_2 \\ 1 \end{bmatrix} \\ &= \frac{1}{\sqrt{5}} \begin{bmatrix} \lambda_1^{t+1} - \lambda_2^{t+1} \\ \lambda_1^t - \lambda_2^t \end{bmatrix}. \end{aligned}$$

Note:
$\lambda_1 - \lambda_2 = \frac{1+\sqrt{5}}{2} - \frac{1-\sqrt{5}}{2} = \sqrt{5}$

The bottom row of this is

$$F_t = \frac{\lambda_1^t - \lambda_2^t}{\sqrt{5}} = \frac{1}{\sqrt{5}} \left(\left(\frac{1+\sqrt{5}}{2} \right)^t - \left(\frac{1-\sqrt{5}}{2} \right)^t \right) \longrightarrow \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^t$$

ver. 2019/12/05

as $t \to \infty$.

Note

Actually, the absolute value $\frac{1}{\sqrt{5}}\left|\frac{1-\sqrt{5}}{2}\right|^t$ of the second term is always less than $1/2$, so F_t is just the closest integer to $\frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^t$.

In general when solving a recurrence $u_{t+1} = Tu_t$, each eigenvalue λ_1 of T yields a part

$$c_1\lambda_i^t x_i$$

of the solution. The full solution is of the form

$$u_t = c_1\lambda_1^t x_1 + c_2\lambda_2^t x_2 + \cdots + c_n\lambda_n^t x_n.$$

As t gets large, the eigenvalues of maximum absolute value dominate, and the other terms can be thrown away.

The difference equation is...

- *stable* if $|\lambda_i| < 1$ for all i, the limiting value is 0.
- *neutrally stable* if $|\lambda_i| = 1$ for some i and $|\lambda_j| < 1$ for all $j \neq i$, the state limits to $c_i x_i$,
- *unstable* if there is an eigenvalue with $|\lambda_i| > 1$, the state grows infinitly.

Note:
Often, the eigenvalues are ordered so that

$$\lambda_n \leq \cdots \leq \lambda_2 \leq \lambda_1.$$

Then λ_1 is the maximum eigenvalue and the difference equation is unstable if $|\lambda_1| > 1$.

Note

What if two eigenvalues have absolute value 1? Is the state not also stable then? Not necessarily, see the text for an example. What if two eigenvalues are 1?

Problems from the text

5.3: 1, 5, 8, 20, 27

5.4 Differential Equations

A system of differential equations is simply a system of equations involving derivatives. Solving one means finding the functions for which the equations hold.

Example 5.10. Solve the differential equation: $v'(t) = a \cdot v(t)$. Perhaps we remember that $v(t) = e^{at}$ is a solution to this equation, as $\frac{\mathrm{d}}{\mathrm{d}t}e^{at} = ae^{at}$.

ver. 2019/12/05

As $v(t) = ce^{at}$ is also a solution, there is usually and extra 'initial value' needed to determine a particular solution.

Problem

Find $v(t)$ when $v'(t) = av(t)$ and $v(0) = 4$.

This becomes more difficult with more variables.

Example 5.11. Solve

$$\begin{aligned} \frac{\mathrm{d}v}{\mathrm{d}t} &= 4v - 5w \text{ where} \qquad & v(0) &= 8 \\ \frac{\mathrm{d}w}{\mathrm{d}t} &= 2v - 3w \text{ where} \qquad & w(0) &= 5 \end{aligned}$$

To solve this, we will first guess that the solution is of the form

$$v(t) = ae^{\lambda t} \qquad w(t) = be^{\lambda t}$$

for some constants a, b and λ, and then find these constants.

Plugging our guesses into the given equations, we get

$$\begin{aligned} \lambda a e^{\lambda t} &= 4ae^{\lambda t} - 5be^{\lambda t} \\ \lambda b e^{\lambda t} &= 2ae^{\lambda t} - 3be^{\lambda t}. \end{aligned}$$

Defining the matrices

$$u(t) = \begin{bmatrix} v(t) \\ w(t) \end{bmatrix} \qquad T = \begin{bmatrix} 4 & -5 \\ 2 & -3 \end{bmatrix}$$

this the orignal system can be written as

$$u' = \frac{\mathrm{d}}{\mathrm{d}t}u = Tu \qquad u(0) = (8, 5) \tag{1}$$

where $u' = \frac{\mathrm{d}}{\mathrm{d}t}u$ is the obvious shorthand for $(v'(t), w'(t))$. With our guessed solution plugged in, this becomes

$$e^{\lambda t}\lambda \begin{bmatrix} a \\ b \end{bmatrix} = e^{\lambda t} T \begin{bmatrix} a \\ b \end{bmatrix} \tag{2}$$

which has the same solutions as $T\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right] = \lambda\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right]$. So to find a, b and λ we have to find the eigens of T.

You can do this:

$$\begin{aligned} \lambda_1 &= -1 & x_1 &= (1, 1) \\ \lambda_2 &= 2 & x_2 &= (5, 2) \end{aligned}$$

ver. 2019/12/05

Each pair (λ_i, x_i) gives a solutions to the system $u' = Tu$, so we have two:

$$u(t) = e^{\lambda_1 t} x_1 = e^{-t} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$
$$u(t) = e^{\lambda_2 t} x_2 = e^{2t} \begin{bmatrix} 5 \\ 2 \end{bmatrix}$$

and as a linear combination of solutions is a solution, we have a solution space

$$u(t) = c_1 e^{-t} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + c_2 e^{2t} \begin{bmatrix} 5 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 & 5 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} e^{-t} & \\ & e^{2t} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}.$$

Our initial conditions $u(0) = (5, 8)$ designated one solution:

$$\begin{bmatrix} 8 \\ 5 \end{bmatrix} = u(0) = \underbrace{\begin{bmatrix} 1 & 5 \\ 1 & 2 \end{bmatrix}}_{S} \underbrace{\begin{bmatrix} e^0 & \\ & e^0 \end{bmatrix}}_{e^{\Lambda t}} \underbrace{\begin{bmatrix} c_1 \\ c_2 \end{bmatrix}}_{S^{-1}u(0)} = \begin{bmatrix} 1 & 5 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$$

which we can solve:

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 1 & 5 \\ 1 & 2 \end{bmatrix}^{-1} \begin{bmatrix} 8 \\ 5 \end{bmatrix} = \frac{1}{-3} \begin{bmatrix} 2 & -1 \\ -5 & 1 \end{bmatrix} \begin{bmatrix} 8 \\ 5 \end{bmatrix} = \frac{1}{-3} \begin{bmatrix} -9 \\ -3 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}.$$

Thus $u(t) = 3e^{-t}\left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right] + e^{2t}\left[\begin{smallmatrix} 5 \\ 2 \end{smallmatrix}\right]$ is our solution.

Note

Summarising: the differential equation $\frac{\mathrm{d}}{\mathrm{d}t} u = Tu$ has the solution

$$u(t) = S e^{\Lambda t} S^{-1} u(0)$$

where T is diagonalised $T = S \Lambda S^{-1}$.

5.4.1 Matrix exponential

We cheated with the notation a bit above, writing

$$e^{\Lambda t} = \begin{bmatrix} e^{\lambda_1 t} & & \\ & e^{\lambda_2 t} & \\ & & e^{\lambda_d t} \end{bmatrix}.$$

Lets define this properly. Recall that the McClauren series for e^x is

$$e^x = 1 + x + x^2/2! + x^3/3! + \dots$$

ver. 2019/12/05

In analogue to this we defined, for a matrix M,

$$e^{Mt} = I + Mt + (Mt)^2/2! + (Mt)^3/3! + \dots.$$

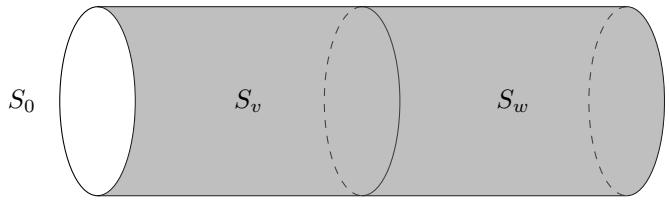
This agrees with our earlier use of the notation, and also has some properties that are envoked by the notation:

- $e^{Ms}e^{Mt} = e^{M(s+t)}$
- $e^{Mt}e^{-Mt} = I$
- $\frac{\mathrm{d}}{\mathrm{d}t}e^{MT} = Me^{Mt}$.

Again, in this last, the notation $\frac{\mathrm{d}}{\mathrm{d}t}M$ means that we take a derivative entrywise. Thus e^{Mt} is the solution to $u'(t) = Mu(t)$, and in face $e^{Mt} = Se^{\Lambda t}S^{-1}$.

5.4.2 A physical example

Differential equations can arise from looking at the movement of gas in a pipe.



In two regions S_v and S_w, we start at time $t = 0$ with concentrations $v(0)$ and $w(0)$. In the region S_0 outside the pipe, we assume the concentration of gas is essentially 0. At time t gas moves between regions proportionally to the difference in concentrations. So the flow is

$$\frac{\mathrm{d}v}{\mathrm{d}t} = (0 - v) + (w - v) \qquad \frac{\mathrm{d}w}{\mathrm{d}t} = (0 - w) + (v - w).$$

To find $v(t)$ and $w(t)$ we solve $u' = \left[\begin{smallmatrix} -2 & 1 \\ 1 & -2 \end{smallmatrix}\right]$. We start be diagonalising the transition matrix $T = \left[\begin{smallmatrix} -2 & 1 \\ 1 & -2 \end{smallmatrix}\right]$:

$$T = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} -1 & \\ & -3 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}^{-1}$$

$$\begin{aligned} u(t) &= \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} e^{-t} & \\ & e^{-3t} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}^{-1} u(0) \\ &= c_1 e^{-t} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + c_2 e^{-3t} \begin{bmatrix} 1 \\ -1 \end{bmatrix} \end{aligned}$$

where $\left[\begin{smallmatrix} c_1 \\ c_2 \end{smallmatrix}\right] = \left[\begin{smallmatrix} 1 & 1 \\ 1 & -1 \end{smallmatrix}\right]^{-1} u(0)$.

ver. 2019/12/05

5.4.3 Stability

For the equation $u' = Tu$ the solution is of the form

$$u(t) = c_1 e^{\lambda_1 t} x_1 + c_2 e^{\lambda_2 t} x_2 + \cdots + c_n e^{\lambda_n t} x_n.$$

This is stable if $e^{\lambda_i t} \to 0$ for all i and (usually) blows up if any of the $e^{\lambda_i t}$ go to infinity. Observing that

$$|e^{\lambda t}| = e^{at} \text{ where } \lambda = a + bi$$

we see that $e^{\lambda_i t}$ goes to infinity iff the real part $Re(\lambda)$ of λ is greater than 0.

Note

The system $u' = Tu$ is...

- *stable* if $Re(\lambda_i) < 0$ for all i.
- *neutrally stable* if $Re(\lambda_i) \leq 0$ for all i and $Re(\lambda_j) = 0$ for some j.
- *unstable* if $Re(\lambda_i) > 0$ for some i.

Problems from the text

5.4: 1, 7, 10, 16, 18

5.5 Complex Matrices

Okay, complex numbers are sneaking into our examples. The picture is incomplete without them. Let's look at matrices over the field $\mathbb{C}$ of complex numbers. This requires us to generalise several definitions such as length and inner (dot) product.

5.5.1 Complex Numbers

Recal lthat complex numbers are of the form

$$z = a + ib$$

where $a, b \in \mathbb{R}$ and $i^2 = -1$. So arithmetic works as expected:

- $(a_1 + ib_1) + (a_2 + ib_2) = (a_1 + a_2) + i(b_1 + b_2)$, and
- $(a_1 + ib_1)(a_2 + ib_2) = (a_1 a_2 - b_1 b_2) + i(a_1 b_2 + a_2 b_1)$.

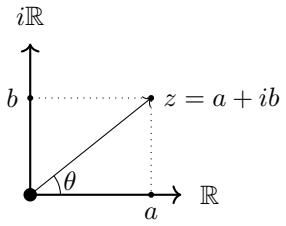
ver. 2019/12/05

For a complex number $z = a + ib$, the value $a = Re(z)$ is the *real part* of z and $b = Im(z)$ is the *imaginary part*.

Multiplication is easier to deal with if we use polar notation. Recall, in polar notation, that the complex number $z = a + ib$ can be written as

$$re^i\theta = r\sin\theta + ir\cos\theta$$

where $r = \sqrt{a^2 + b^2}$ is length of z and $\theta = \arctan(b/a)$ (for $a \geq 0$) is the angle that z makes with the real axis when we graph it in the complex plane:



Now, multiplication works as:

$$r_1e^{i\theta_1} \cdot r_2e^{i\theta_2} = (r_1 \cdot r_2)e^{i(\theta_1+\theta_2)}.$$

Moreover, this length r, this is what we use as the length $|z|$ of the z. A complex number $z = a + ib$ is real if $b = 0$, and in this case $|z| = \sqrt{a^2 + b^2} = \sqrt{a^2} = |a|$, so this generalises our notion of absolute value for real numbers.

5.5.2 Complex vectors: Length and inner product

Let $x = \begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix}$ be in $\mathbb{C}^n$. The *length* or *norm* $\|x\|$ is defined by

$$\|x\|^2 = |z_1|^2 + \cdots + |z_n|^2.$$

Problem

What is the length of the complex vector $x = (2, 1 + 2i)$?

The inner product for real vectors was

$$x \cdot y = x^T y = x_1 \cdot y_1 + \cdots + x_n \cdot y_n.$$

We could use this same formula for complex numbers, but then we would loose such properties as

$$x \cdot x = \|x\|^2.$$

ver. 2019/12/05

Problem

Give an example showing this.

To define a proper inner product, we first need to define a 'complex conjugate'. For $z = a + ib = re^{i\theta}$, the *complex conjugate* of z is the number

$$\overline{z} = a - ib = re^{-i\theta}.$$

Problem

Show that $z\overline{z} = |z|^2$.

Now, for complex (column) vectors $x = (x_1, \ldots, x_n)^T$ and $y = (y_1, \ldots, y_n)^T$ we let $\overline{x} = (\overline{x}_1, \ldots, \overline{x}_n)^T$ and define the inner product as

$$x \cdot y = \overline{x}^T y = \overline{x}_1 \cdot y_1 + \cdots + \overline{x}_n \cdot y_n.$$

It follows then that

$$\begin{aligned} x \cdot x = \overline{x}^T x &= \overline{x}_1 \cdot x_1 + \cdots + \overline{x}_n \cdot x_n \\ &= |x_1|^2 + \cdots + |x_n|^2 = \|x\|. \end{aligned}$$

5.5.3 The Hermitian

The *(complex) conjugate* of a matrix $M = [m_{ij}]$ is the matrix

$$\overline{M} = [\overline{m}_{ij}]$$

we get from M by entrywise conjugation.

So, for example

$$\overline{\begin{bmatrix} 1 & 4 \\ i+i & i \end{bmatrix}} = \begin{bmatrix} 1 & 4 \\ 1-i & -i \end{bmatrix}.$$

The *Hermitian* or *conjugate transpose* of M is

$$M^H := \overline{M}^T = \overline{M^T}.$$

Problem

What is the Hermitian of $\begin{bmatrix} 2+i & 4 \\ i & 0 \\ 1 & 2-i \end{bmatrix}$?

Observe that

$$\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}^H = [\overline{x}_1, \ldots, \overline{x}_n],$$

ver. 2019/12/05

so the inner product can now be written as $x \cdot y = x^H y$.

Though $x^T = x^H$ when x is a real vector, so $x^T x = x^H x$, this is not true for complex vectors x. In many situations, when we are considering complex vectors, we must use x^H instead of x^T in results or definition we know from the real case. One such situation is the definition of orthogonality. From now on:

Note

Two vectors x and y are *orthogonal* if $x^H y = 0$.

Observe also, that much like we had for the transpose, we have

$$(AB)^H = \overline{(AB)^T} = \overline{(B^T A^T)} = \overline{B^T}\ \overline{A^T} = B^H A^H.$$

Problem

Why does that second last equality hold?

A matrix M is *Hermitian* if

$$M^H = M.$$

For example $\left[\begin{smallmatrix} 2 & 3-i \\ 3+i & 5 \end{smallmatrix}\right]$ is Hermitian. One quickly sees that for a Hermitian matrix $M = [m_{ij}]$ we must have

- m_{ii} is real, and
- $\overline{m}_{ij} = m_{ji}$.

Hermitian matrices have three nice properties, and keep in mind, **if a matrix is real, it is Hermitian if and only if it is symmetric,** and so these properties hold for real symmetric matrices too. Let M be Hermitian.

i. For all $x \in \mathbb{C}^n$, $x^H M x$ is real.

ii. All eigenvalues of M are real.

iii. Eigenvectors of M with distinct eigenvalues are orthogonal.

When a matrix M has n distinct eigenvalues, we can diagonalise it as

$$M = S \Lambda S^T.$$

When, further, M is Hermitian, the above properties ii. and iii. allow us to (by normalising the eigenvectors) take S to satisfy $S^H S = I$. Thus $S^H = S^{-1}$. In the case that S is real we then get $S^T = S^H = S^{-1}$ and so S is orthanormal.

ver. 2019/12/05

Note

We will show in the next section that this result actually holds for any Hermitian matrix M, not only those with distinct eigenvalues. This will come down to showing that for a Hermitian matrix, every eigenvalue has the same geometric mutliplicity as algebraic multiplicity.

Proof of properties i. ii. and iii.

To see property i., observe that the 1×1 matrix $x^H Mx$ is its own conjugate:

$$(x^H Mx)^H = x^H M^H x^{HH} = x^H M^H x = x^H Mx$$

and so must be real.

To see property ii., let $Mx = \lambda x$. Then

$$\underbrace{x^H Mx}_{\text{real by i.}} = x^H \lambda x = \lambda x^H x = \lambda \|x\|$$

and $\|x\|$ is real (and non-zero) so $\lambda = \frac{x^H Mx}{\|x\|}$ is real.

To see property iii., let

$$Mx = \lambda_1 x \text{ and } My = \lambda_2 y.$$

Then

$$\lambda_1 x^H y = (Mx)^H y = x^H My = x^H \lambda_2 y = \lambda_2 x^H y,$$

so $\lambda_1 \neq \lambda_2$ gives that $x^H y = 0$, as needed.

5.5.4 Unitary Matrices

Generalising on the idea of an orthogonal matrix, a complex matrix U is *unitary* if

$$U^H U = I.$$

Using essentially the proofs as above for the more general properties of Hemitians matrices M the following properties can by shown for any unitary matrix U.

i. $(Ux)^H (Uy) = x^H y$ and in particular $\|Ux\| = \|x\|$.

ii. Every eigenvalue λ of U has $|\lambda| = 1$.

iii. Eigenvectors of different eigenvalues are orthonormal.

As we mentioned above, we will show in the next section that every Hermitian matrix will be diagonalisable by a unitary matrix. Let's see an example of this.

ver. 2019/12/05

Example 5.12. The matrix $M = \left[\begin{smallmatrix} 3 & 2+2i \\ 2-2i & 1 \end{smallmatrix}\right]$ is Hermitian. Let's diagonalise it.

$$\begin{aligned} Det(M-\lambda I) &= (3-\lambda)(1-\lambda) - (2+2i)(2-2i) = (3-\lambda)(1-\lambda) - 8 \\ &= \lambda^2 - 4\lambda - 5 = (\lambda-2)^2 - 9 \end{aligned}$$

has roots $\lambda = 2 \pm \sqrt{9} = -1, 5$. To get the eigenvector for $\lambda_1 = 5$ we find the null-vector of $\left[\begin{smallmatrix} -2 & 2+2i \\ 2-2i & -4 \end{smallmatrix}\right]$ which is $x_1 = (1+i, 1)$, and read off that the eigenvector of λ_2 is $x_2 = (1+i, -2)$. These should be orthogonal:

$$(1-i, 1) \cdot (1+i, -2) = (1-i)(1+i) + (1)(-2) = 2 - 2 = 0.$$

Normalising them to $x_1 = (1+i, 1)/\sqrt{3}$ and $x_2 = (1+i, -2)/\sqrt{6} = (1, i-1)/\sqrt{3}$ we get that $U = \frac{1}{\sqrt{3}}\left[\begin{smallmatrix} 1+i & 1 \\ 1 & -1+i \end{smallmatrix}\right]$ and so $U^{-1} = U^H = \frac{1}{\sqrt{3}}\left[\begin{smallmatrix} 1-i & 1 \\ 1 & -1-i \end{smallmatrix}\right]$.

We should check that $M = U \begin{bmatrix} -1 & \\ & 5 \end{bmatrix} U^{-1}$.

Problems from the text

5.5: 1, 2, 6, 10, 12, 15, 18, 29, 37, 42, 43, 49, 50

5.6 Similarity Transformations

A matrix B is similar to a matrix A if

$$B = MAM^{-1}$$

for some matrix M.

Similar matrics share similar properties For example, if $B = MAM^{-1}$ then A and B have the same eigenvalues, and their eigenvectors are related through M:

$$\begin{aligned} Ax = \lambda x \quad &\iff \quad M^{-1}BMx = \lambda x \\ &\iff \quad B(Mx) = \lambda(Mx) \end{aligned}$$

Problem

Show that if B is similar to a diagonalisable matrix, then B is diagonalisable.

ver. 2019/12/05

Indeed, that a matrix is diagonalisable just means that it is similar to a diagonal matrix. Recalling our long goal of computing the eigenvalues of a big matrix, this suggests where we might be going. If we can find a diagonalisation of a matrix M then we have its eigenvalues. For this goal, it enough that M is similar to a triangular matrix. But we will see in the next subsection why it is nicer if M is similar to a diagonal matrix. We will see also that while similarity is nice, similary by unitary matrices is nicer. So the ideal will be writing $M = U\Lambda U^{-1}$ for some unitary U and diagonal Λ. We will look at which matrices we can do this for. It won't be all. For the other matrices we will show that they can be triangulised, but better than this, triangulised by unitary matrices. And that they can also by 'almost diagonalised', into something we will call there Jordan form.

First, we look, from a new point of view, at why diagnolising is nicer than triangulating, if we can do it.

5.6.1 Change of Basis

Given a basis $B = \{b_1, \ldots, b_n\}$ and a vector x the notation

$$[x]_B = (a_1, \ldots, a_n)$$

tells us that $x = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$.

Example 5.13. Where $E = \{e_1, e_2\}$ is the standard basis $[(1,3)]_E = (1,3)$ but where B is the basis $B = \{b_1 = (1,1), b_2 = (-1,1)\}$ we have $[(1,3)]_B = (2,1)$ because $(1,3) = 2(1,1) + 1(-1,1)$.

Given a vector $[x]_B$ represented in the basis B, it is easy to transform it to the standard basis E, but how do we find $[x]_B$ for a given vector x, as we did in the above example? Let M_B be the matrix whose columns are the vector of B. Then viewing M_B as a transformation we have

$$M_B : \begin{cases} e_1 \mapsto b_1 \\ \ldots \\ e_n \mapsto b_n \end{cases} \qquad \text{so} \qquad M_B^{-1} : \begin{cases} b_1 \mapsto e_1 \\ \ldots \\ b_n \mapsto e_n \end{cases}$$

Thus $x = M_B[x]_B$ and so $[x]_B = M_B^{-1} x$.

Example 5.14. To represent $x = (1,3)$ in the basis $B = \{(1,1), (-1,1)\}$, we find $M_B^{-1} = \left[\begin{smallmatrix} 1 & 1 \\ -1 & 1 \end{smallmatrix}\right]/2$ and compute

$$[(1,3)]_B = M_B^{-1}[x]_E = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}.$$

ver. 2019/12/05

Note

Change of basis can be confusing. We talk of the vectors x and $M_B^{-1}x$ as being the same vector, just represented in different bases. But hey, multiplying by the transformation M_B^{-1} takes the vector $(1,3)$ to $(2,1)$! It does change. Yep, confusing. The point of view of the change of basis is that we are keeping the vectors still, but moving the paper (and the coordinate axes). Instead of rotating vectors clockwise, we rotate the paper counterclockwise. Instead of stretching the vectors, we zoom out the paper. Multiplying by M_B changes from the B frame of reference to the standard E frame of reference. (Notation can be the other way around in other books).

Example 5.15. The matrix $A = \left[\begin{smallmatrix} 3 & 1 \\ 1 & 3 \end{smallmatrix}\right]$ diagonalises to

$$A = M\Lambda M^{-1} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 4 & \\ & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} /2.$$

Applying this as a transformation to some vector $x = (1,3)$ we get

$$A : (1,3) \overset{M_B^{-1}}{\longmapsto} (2,1) \overset{\Lambda}{\mapsto} (8,2) \overset{M_B}{\longmapsto} (6,10).$$

We interpret transformation by A as consisting of three steps. The first is changing $x = (1,3)$ to $[x]_B = (2,1)$ in the basis B. Then in the basis B transforming by Λ to $(8,2)$. Transformations by diagonal matrices are easy to visualise as there is no rotation involved, just stretching in the two directions. Finally, we change back to $(6,10)$ in the original basis.

So a matrix A being similar to a diagonal matrix is pretty nice. It is even nicer if the matrix M that transforms A to diagonal is unitary (orthanormal). Not only is M^{-1} then easy to compute, but if M is unitary, then the eigenspaces of A are orthogonal, so basis we are transforming to is orthonormal. Thus the action of A decomposes into projections onto the eigenspaces:

$$A = U\Lambda U^H = \lambda_1 x_1 x_1^H + \cdots + \lambda_n x_n x_n^H.$$

Example 5.16. Where

$$\begin{aligned} A = \frac{1}{2}\begin{bmatrix} 3 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 1 \end{bmatrix} &= \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & & \\ & 1 & \\ & & 1 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= 2\begin{bmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 1/2 & 0 \\ 0 & 0 & 0 \end{bmatrix} + 1\begin{bmatrix} 1/2 & -1/2 & \\ -1/2 & 1/2 & \\ & & \end{bmatrix} + 1\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

The first matrix on the left of the last line is the projection onto the eigenspace $\langle(1,1,0)\rangle$ of $\lambda_1 = 2$ while the second two matrices are the projection onto the eigenspace $\langle(1,-1,0),(0,0,1)\rangle$ of $\lambda_2 = 1$.

ver. 2019/12/05

5.6.2 Diagonalising and Triangularising by Unitary Matrices

A matrix A is *normal* if $A^H A = AA^H$.

Theorem 5.17. *The following are equivalent.*

i. $A = U\Lambda U^H$ *for some unitary matrix* U.

ii. A *has* n *orthogonal eigenvectors.*

iii. A *is normal.*

Proof. The proof of *i.* $\iff$ *ii.* is trivial as the columns of such a U must be eigenvectors. The proof that *ii.* $\Rightarrow$ *iii.* is question 43 in the text. The proof that *iii.* $\Rightarrow$ *i.* is harder and we omit it. (It is in the text, but confusing.) $\square$

If A is not normal, then we cannot diagonalise it by unitary matrices, but we can get close.

Theorem 5.18. *Any matrix* A *can be triangularised:* $A = UTU^H$ *by a unitary matrix* U. *The eigenvalues of* A *are on the diagonal of* T.

5.6.3 Jordon Canonical Form

If we relax the restriction to conjugation by unitary matrices, we can show that a matrix is similar to something even closer to diagonal that a triangular matrix.

Theorem 5.19 (Jordon Form)**.** *Any matrix* A *can be written (uniquely) as* $A = MJM^{-1}$ *for its Jordon Form– a matrix* J *such that*

- *the eigenvalues of* A *are the diagonals of* J *(they may be assumed to be non-decreasing),*
- *all other entries are* 0 *except for* 1*s on the superdiagonals adjacent to pairs of repeated eigenvalues.*

The proof of this can be found in the appendix of the text.

Example 5.20. The matrices $A = \begin{bmatrix} 11 & -2 & 7 \\ 0 & 8 & 0 \\ 3 & -2 & 1 \end{bmatrix}$ and $B = \begin{bmatrix} 9 & -2 & -5 \\ 4 & 8 & 4 \\ 1 & -2 & 3 \end{bmatrix}$ both have eigenvalues $4, 8, 8$. For A the eigenspacess have dimensions 1 and 2, for B they have dimensions 1 and 1. So the Jordon forms are

$$J_A = \begin{bmatrix} 4 & & \\ & 8 & \\ & & 8 \end{bmatrix} \qquad \text{and} \qquad J_B = \begin{bmatrix} 4 & & \\ & 8 & 1 \\ & & 8 \end{bmatrix}$$

respectively.

ver. 2019/12/05

The Jordon Form is 'block diagonal' and each block is upper triangular with constant diagonals and ones in all superdiagonal entries. The blocks are called the *Jordon blocks* of the matrix. There may be more than one Jordon block for a given eigenvalue, as we see with A above. The number of Jordon blocks for an eigenvalue is determined by the algebraic multiplicity and the geometric multiplicity of that eigenvalue. Knowing that there are two blocks for an eigenvalue does not tell us the size of each, this is uniquely determined, We we will not go into this.

The following seems clear.

Corollary 5.21. *Two matrices are similar if and only if they have the same Jordon Form.*

Not only is the Jordon form useful in that it lets us decide if two matrices are similar, As with a diagonalisation, it is useful for taking powers.

Indeed

$$\begin{bmatrix} a & & \\ & b & 1 \\ & & b \end{bmatrix}^n = \begin{bmatrix} a^n & & \\ & b^n & nb^{n-1} \\ & & b^n \end{bmatrix}$$

and more generally we see that blocks 'power independently' and Jordon blocks 'power predictably' (according to an obvious formula):

$$\begin{bmatrix} a & 1 & \\ & a & 1 \\ & & a \end{bmatrix}^n = \begin{bmatrix} a^n & na^{n-1} & \binom{n}{2}a^{n-1} \\ & a^n & na^{n-1} \\ & & a^n \end{bmatrix}.$$

Problems from the text

5.6: 1, 3, 5, 6, 8, 11, 14, 16, 26, 36, 40

6 Positive Definite Matrices

6.1 Maxima, minima, and saddle points

We return to considering only real matrices. Recall that the eigenvalues of a square real matrix may be complex, but if the matrix is symmetric, then the eigenvalues and eigenvectors are real.

A symmetric real matrix A is *positive definite* if

$$x^T Ax > 0$$

for all non-zero (real!) vectors x.

ver. 2019/12/05

As this must hold for eighenvectors in particular:

$$x_i^T A x_i = x_i^T x_i \lambda_i = \lambda_i,$$

we get that all eigenvalues are positive.

Taking $x = e_i$ as a standard basis vector, we see that positive definite matrices must have positive diagonal entries. We would like to characterise all positive defintie matrices. A useful tool (as well as a useful application) will be the encoding of of polynomials as matrices.

In fact, any $n \times n$ matrix can be used to encode a homogeneous degree 2 polynomial in n variables, a.k.a. a *purely quadratic* polynomial.

Note:
Recall that a multivariate polynomial is of homogeneous degree 2 if it is the sum of monomials such that the sum of the degrees in each variable is 2.

Example 6.1. Where $A = \left[\begin{smallmatrix} 2 & 1 \\ 1 & 3 \end{smallmatrix}\right]$ we have

$$\begin{bmatrix} x & y \end{bmatrix} A \begin{bmatrix} x \\ y \end{bmatrix} = 2x^2 + xy + yx + 3y^2$$
$$= 2x^2 + 2xy + 3y^2$$

Calling this polynomial $f_A(x, y)$, we have that $f_A(v) = v^T A v$. For example,

$$f_A(1, 0) = \begin{bmatrix} 1 & 0 \end{bmatrix} A \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 2.$$

As f_A is homogeneous, it has no constant term, so $f_A(0, 0) = 0$. The statement that A is positive definite is equivalent to the statement that $f_A(x, y)$ is positive for all other values $(x, y) \in \mathbb{R}^2$. This is where the terminology comes from. So can we other give necessary and sufficient conditions for a symmetric real matrix to be positive definite?

We get one set of conditions from calculus. Let's start with 2×2 matrices. In general, we have

$$f(x, y) = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} a & b \\ b & c \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = ax^2 + 2bxy + cy^2.$$

The critical points occur at (x, y) such that

$$0 = \frac{\mathrm{d}}{\mathrm{d}x} f(x, y) = 2ax + 2by = 0$$
$$0 = \frac{\mathrm{d}}{\mathrm{d}y} f(x, y) = 2bx + 2cy = 0$$

or equivalently, such that

$$\begin{bmatrix} a & b \\ b & c \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

which we have observed holds for $(x, y) = (0, 0)$.

ver. 2019/12/05

It holds in a subspace of $\mathbb{R}^2$ so everywhere (if $a = b = c = 0$) in a line through $(0, 0)$ (in which case the graph of $z = f(x, y)$ is a valley) , or nowhere else (in which case $z = f(x, y)$ is a paraboloid if $(0, 0)$ is a local min, or a saddle point if it is not).

The matric is positive definite if and only if $(0, 0)$ is a local minimum. For this to be true, we also need that all directional derivatives at $(0, 0)$ are positive. In particular we need that

$$0 < \frac{\mathrm{d}^2}{\mathrm{d}x^2} f(x, y) = 2a$$
$$0 < \frac{\mathrm{d}^2}{\mathrm{d}y^2} f(x, y) = 2c$$

which is exactly the condition that the diagonals are positive.

We also need that the other directional derivatives are positive, and by the second derivative test, this is true if and only if

$$f_{xx} f_{yy} > f_{xy}^2 \qquad \text{i.e. } ac > b^2.$$

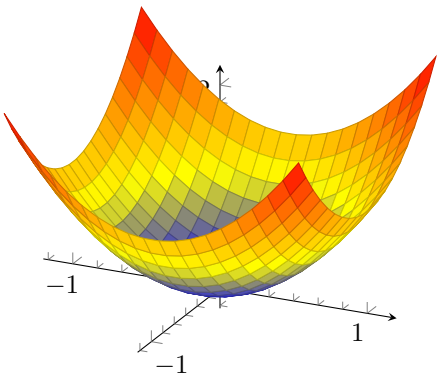
In summary,

Note

The symmetric matrix $\left[\begin{smallmatrix} a & b \\ b & c \end{smallmatrix}\right]$ is positive definite if and only if $a > 0$ and $ac > b^2$.

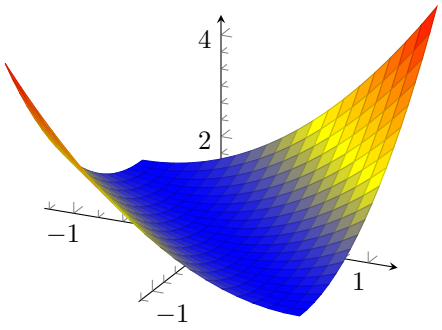
Recall from calculus what $z = f(x, y)$ looks like at $(0, 0)$ in terms of $ac - b^2$.

If $ac - b^2 > 0$ then (as we discussed) $(0, 0)$ is a local min and $z = f(x, y)$ is a upward opening parboloid.

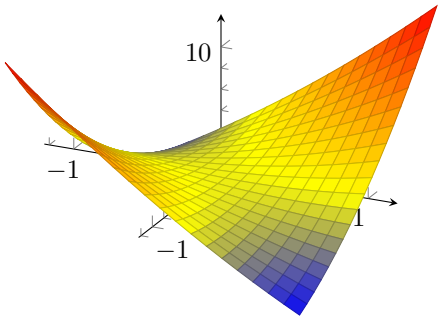


ver. 2019/12/05

If $ac = b^2$ as with the polynomial $(x + y)^2$ then $z = f(x, y)$ is a valley.



If $ac < b^2$ then $(x, y) = (0, 0)$ is a saddle point of $z = f(x, y)$.



Problems from the text

6.1: 1, 2, 3, 6, 10, 21

6.2 Tests for Positive Definiteness

We observed some parts of the following theorem last section. We prove the rest now.

Theorem 6.2. *The following are equivalent definitions for a symmetric matrix A to be positive definite.*

i. $x^T Ax > 0$ *for all* $x \neq 0$.

ii. All eigenvalues of A are positive.

iii. All upper left square submatrices of A have positive determinants.

iv. All pivots of A (without row exchange) are positive.

ver. 2019/12/05

v. There is a rank n matrix R such that $A = R^T R$.

Proof.

• We saw *i.* $\Rightarrow$ *ii.* last section.

• For *ii.* $\Rightarrow$ *i.* observe that as A is symmetric, it is diagonalisable, so any vector x can be written in a basis of orthonormal eigenvectors

$$x = c_1 x_1 + c_2 x_2 + \dots c_n x_n$$

and so

$$\begin{aligned} x^T A x &= x^T (c_1 \lambda_1 x_1 + c_2 \lambda_2 x_2 + \dots c_n \lambda_n x_n \\ &= c_1^2 \lambda_1 + c_2^2 \lambda_2 + \dots, c_n^2 \lambda_n > 0 \end{aligned}$$

• *ii.* $\Rightarrow$ *iii.*: From i. it is easy to see that all upper left square submatrices also satisfy i. by considering vectors x ending in sufficiently many zeroes. So it is enough to observe that matrices satisfying ii. have positive determinants. But this is trivial as the determinant is the product of the eigenvalues.

• *iii.* $\Rightarrow$ *iv.* The pivots are the ratios of the determinants of consecutive upper left square submatrices.

• *iv.* $\Rightarrow$ *v.* Since A is symmetric its (unique) LDU decomposition is the same as that of A^T, and so

$$LDU = A = A^T = U^T D^T L^T$$

implying that

$$A = LDL^T.$$

As the entries of D in the LDU decomposition are the pivots of A, we have that D can be written $D = \sqrt{D}\sqrt{D}$, and so

$$A = (L\sqrt{D})(\sqrt{D}L^T) = (\sqrt{(}D)^T L^T)^T (\sqrt{(}D)^T L^T) =: R^T R.$$

• *v.* $\Rightarrow$ *i.* Assuming $A = R^T R$ for R with rank n. We get

$$x^T A x = x^T R^T R x = (Rx)^T (Rx) = \|Rx\| \geq 0$$

As R has full rank, this is not 0 unless $x = 0$.

□

ver. 2019/12/05

Now, a matrix A is *positive **semi**definite* if for all vectors x

$$x^T Ax \geq 0.$$

So for $A = \left[\begin{smallmatrix} a & b \\ b & c \end{smallmatrix}\right]$ is positive semidefininte if $ac = b^2$, and the graph of $z = f_A(x, y)$ is a valley.

The characterisation theorem for positive definite matrics can be used to give a similar theorem from positive semidefinite matrics: one can essentially add small $\varepsilon > 0$ to the diagonal, apply the theorem for positive definite matrices, and then let ε go to 0. Doing this one gets:

Theorem 6.3. *For a symmetric real matrix A, the following are equivalent.*

i. $x^T Ax \geq 0$ *for all* x.

ii. All eigenvalues of A are non-negative.

iii. All principal submatrices of A have non-negative determinants.

iv. All pivots of A (without row exchange) are non-negative.

v. There is a matrix R (not necessarily rank n) such that $A = R^T R$.

Note:
The principle submatrices are those you get by removing the same columns and rows. In this case is is not enough to consider only the upper left principle submatrices, as diagonals can be 0.

6.2.1 Positive definite matrics as Ellipses

The set of solutions of the polynomial equation $x^T Ix = 1$ or

$$x_1^2 + x_2^2 + \cdots + x_n^2 = 1$$

is the surface of the unit n-sphere centered at the origin.

Changing co-efficients

$$d_1 x_1^2 + d_2 x_2^2 + \cdots + d_n x_n^2 = 1$$

shrinks the sphere in the x_i direction by a factor of $\sqrt{d_i}$. So for diagonal D $x^T Dx = 1$ is an ellipse if all entries of D are positive. (If some entries are 0 the ellipse 'becomes a cylinder' in that dimension. If they are negative we 'become hyperbolic' in that dimension.)

For a positive definite matrix A the diagonalisation

$$A = U\Lambda U^{-1}$$

by a unitary U (which doesn't change the lengths of vectors) yields that

$$1 = x^T Ax = x^T U\Lambda U^{-1} x$$

simply rotates the ellipse $x^T \Lambda x = 1$. Effectively, it is the ellipse we get from the unit sphere by stretching by a factor of $1/\sqrt{\lambda}$ in the direction of the i^{th} eigenvector.

ver. 2019/12/05

6.2.2 The Law of inertia

We look at a computational way to approximate (as closely as we like) the eigenvalues of a symmetric matrix.

Recall that for any matrix M, the matrix A shares eigenvalues with $B = M^{-1}AM$ (and has eigenvectors related through M). If M is orthogonal then B can be written $B = M^TAM$. We are going to try writing B like this for non-orthogonal M.

For symmetric A, and non-singular (but not necessarily orthogonal) C, we consider C^TAC, and call it *congruent* to A. As A is symmetric, so is C^TAC, and so its eigenvalues are still real. Indeed

$$(C^TAC)^T = (C^TA^TC)^T = (C^TAC).$$

But we have more than that its eigenvalues are real.

Theorem 6.4. *Let A be a symmetric real matrix and C be a non-singular matrix. Then A and C^TAC have the same number of positive, zero, and negative eigenvalues.*

The proof is in the book. It is not diffifult, but uses notions that our not in our scope, so we omit it. Let's see why the theorem is useful though.

For symmetric A, we saw that the LDU factorisation is indeed a congruence

$$A = LDL^T$$

with a diagonal matrix D whose entries are the pivots of A. As these are the eigenvalues of D the above theorem gives us the following.

Corollary 6.5. *For symmetric A, the eigenvalues of A have the same signs as its pivots.*

This is nice, because finding pivots is easy, it is done by elimination. This allows us to approximate the eigenvalues of a symmetric matrix A as follows.

Example 6.6. Say a symmetric matrix A of dimension 10 has 3 positive pivots, so 3 positive eigenvalues. And say that $A - 7I$ has 1 positive eigenvalue.

As for any eigenvector x_i of A we have that

$$(A - 7I)x_i = \lambda_i x_i - 7x_i = (\lambda_i - 7)x_i,$$

the eigenvalues of $A - 7I$ are those of A shifted by 7. So we know that A has two eigenvalues between 0 and 7.

So we try with $A - \frac{7}{2}I$ etc. and with a lot of computation, can find one particular, or all, eigenvalues of A.

ver. 2019/12/05

Problems from the text

6.2: 1, 2, 3, 8, 10, 12, 16, 19, 22

6.4 Minimum Principles

For a positive definite matrix A we started with the matrix equation $Ax = 0$, and multplied on the left by x^T to get the homgeneous quadratic polynomial

$$x^T Ax = x^T 0 = 0.$$

In this section we consider what happens when we multiply

$$Ax = b \qquad \text{and} \qquad Ax = \lambda x$$

by x^T.

Ax = b

Multiplying $Ax = b$ or $Ax - b = 0$ by $x^T = (x_1, \ldots, x_n)$, we get

$$Q(x) := x^T Ax - x^T b = 0.$$

For a symmetric 2×2 matrix A this is

$$0 = Q(x) \quad = \quad \underbrace{x^2 a_{11} + 2a_{12}xy + y^2 a_{22}}_{P(X)} - b_1 x - b_2 y.$$

We can write $Q(x)$ as $Q(x) = P(x) - b_1 x - b_2 y$ so view it as a shift (by a plane) of the positve definite polynomial $P(X)$. Again, we find a critical point of $Q(x)$ with calculus. It occurs at

$$0 = Q_x = 2a_{11}x + 2a_{12}y - b_1 \tag{3}$$
$$0 = Q_y = 2a_{12}x + 2a_{22}y - b_2 \tag{4}$$

Which is where

$$0 = 2A \begin{bmatrix} x \\ y \end{bmatrix} - b \qquad \text{so} \qquad 2Ax = b.$$

This is at $x = \frac{1}{2}A^{-1}b$.

Rewriting $2A$ as A we get that

$$Q(x) = \frac{1}{2}x^T Ax - x^T b + c$$

ver. 2019/12/05

has a critical point at $x_* = A^{-1}b$. Putting this into $Q(x)$ gives

$$\begin{aligned} Q(x_*) &= \frac{1}{2}x_*^T A x_* - x_*^T b + c \\ &= \frac{1}{2}(A^{-1}b)^T A(A^{-1}b) - (A^{-1}b)^T b + c \\ &= \frac{1}{2}b^T(A^{-1})^T b - b^T(A^{-1})^T b + c \\ &= -\frac{1}{2}b^T(A^{-1})^T b + c \\ &= -\frac{1}{2}(b^T A^{-1}b)^T + c \\ &= -\frac{1}{2}(b^T A^{-1}b) + c \\ &= -\frac{1}{2}(b^T x_*) + c \end{aligned}$$

We can use this to find the minimim and min value of such a Q.

Example 6.7. Minimize $Q(x) = 2x_1^2 - x_1x_2 + 3x_2^2 - x_1 + x_2$.

We can write this as $Q(x) = \frac{1}{2}x^T Ax - x^T b$ where

$$A = \begin{bmatrix} 4 & -1 \\ -1 & 6 \end{bmatrix} \text{ and } b = \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

So its minimum occurs as

$$x_* = A^{-1}b = \frac{1}{23}\begin{bmatrix} 6 & -1 \\ -1 & 4 \end{bmatrix}\begin{bmatrix} 1 \\ -1 \end{bmatrix} = \frac{1}{23}\begin{bmatrix} 7 \\ -5 \end{bmatrix}$$

and is

$$\begin{aligned} Q(x_*) &= -\frac{1}{2}(b^T A^{-1}b) + c \\ &= -\frac{1}{2}(\begin{bmatrix} 1 & -1 \end{bmatrix}\frac{1}{23}\begin{bmatrix} 7 \\ -5 \end{bmatrix}) = \frac{-12}{46} \end{aligned}$$

An application: Minimising with Constraints

What is the min of $Q(x) = x^T Ax - x^T b$ when constrained to

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} x = c_1x_1 + c_2x_2 = d?$$

Legranges method writes

$$L(x, y) = Q(x) + y(c_1x_1 + c_2x_2 - d)$$

ver. 2019/12/05

and then uses the fact that at for a critical point (x, y) we have

$$0 = L_y = c_1 x_1 + c_2 x_2 - d.$$

So our constraint is satisfied at critcal points of $L(x, y)$. Moreover, as this is satisfied, we have at critical points that

$$L(x, y) = Q(x) + y(0) = Q(x).$$

Thus any local minimum $L(x, y)$ is a local minimum of $Q(x)$ when restricted to the constraint.

Where x, b and A had dimension n, we augment them with the constraint, writing

$$x' = \begin{bmatrix} x_1 \\ \vdots \\ x_n \\ y \end{bmatrix} \qquad A' = \left[\begin{array}{ccc|c} & A & & c_1 \\ & & & \vdots \\ & & & c_n \\ \hline c_1 & \cdots & c_n & 0 \end{array}\right] \qquad b' = \begin{bmatrix} b_1 \\ \vdots \\ b_n \\ d \end{bmatrix}$$

and so get

$$L(x, y) = \frac{1}{2} x'^T A' x' - x'^T b'.$$

Our critical point is at $x'_* = A'^{-1} b'$ and has value $L(x'_*) = -\frac{1}{2} b'^T x'_*$.

The Rayleigh Quotient

Multiplying $Ax = \lambda x$ by x^T we get $x^T A x = x^T \lambda x = \lambda x^T x$. Solving for λ we see that for any eigenvector x of A, the corresponding eigenvalue is

$$\lambda = \frac{x^T A x}{x^T x}.$$

For any vector x, not necessarily an eigenvector, the *Rayleigh quotient* (with respect to a symmetric or Hermitian matrix A) is

$$R_A(x) = R(x) := \frac{x^T A x}{x^T x}.$$

The intuition for $R(x)$, which is explicit when x is an eigenvector, is that it is a measure of how much transformation by a positive definite matrix A stretches x.

ver. 2019/12/05

Note

For investigation of the Rayleigh quotient, observe that for any constant c we have

$$R(cx) = \frac{(cx)^T A(cx)}{(cx)^T(cx)} = \frac{c^2 x^T Ax}{c^2 x^T x} = R(x)$$

and so we can see all there is to see about it by considering normalized vectors x for which we have the simpler formula $R(x) = x^T Ax$.

This agrees with our intuition that $R(x)$ measure the 'stretch' of x. This should be the same for x and cx.

Diagonalising $A = Q\Lambda Q^T$ by unitary Q we get, where $x = Qy$, that

$$R(x) = x^T Ax = (Qy)^T A(Qy) = y^T \Lambda y = \lambda_1 \|y_1\|^2 + \cdots + \lambda_n \|y_n\|^2$$

Note:
For these bits we are using the convention that the eigenvalues (which are all positive real) are ordered with $\lambda_i \leq \lambda_{i+1}$.

from which it is clear that $R(x)$ is largest when $x = x_n$ and smallest when $x = x_1$. That is

$$\lambda_1 \leq R(x) \leq \lambda_n.$$

Moreover, a nice observation is that

$$R(e_i) = \frac{e_i^T A e_i}{e_i^T e_i} = a_{ii}$$

so we have the following bounds on the eigenvalues that are useful if we want to approximate them computationally:

$$\lambda_1 \leq a_{ii} \leq \lambda_n.$$

Observe also that for the matrix B which we get from A by removing the i^{th} row and column, we get that $x^T Bx = 1$ is the $n-1$ dimensional ellipsoid which is the intersection of $x^T Ax = 1$ with a hyperplane. So the principle axis of $x^T BX = 1$ is at most that of $x^T Ax = 1$, yielding that $\lambda_{n-1}(B) \leq \lambda_n(A)$. Extending this argument we get the *interlacing of eigenvalues*

$$\lambda_1(A) \leq \lambda_1(B) \leq \lambda_2(A) \leq \lambda_2(B) \leq \cdots \leq \lambda_{n-1}(B) \leq \lambda_n(A).$$

Problems from the text

6.4: 1, 2, 3, 9, 11, 14

ver. 2019/12/05

7 Computations with matrices

We can solve $Ax = b$ well enough, but for large matrices it can save time to find, rather, an approximate solution. Generally we are doing comptations by computers, and we save a LOT of time and complication by rounding irrational numbers to some number of decimal points. We should known how much this rounding effects our result.

We look at this in this chapter. The first step is to introduce a 'measure of goodness' for approximations.

7.2 Matrix Norm and Condition Number

When solving $Ax = b$ we want to known when it is okay to approximate A or b. That is, we want to know that if we change A or b by some small bit we will change the solution $x = A^{-1}b$ by only some small bit.

Let's consider first changing only b. If we solve $Ax = b$, by first replacing b with

$$b' = b + \varepsilon_b$$

for some small error ε_b. Our solution will by $A^{-1}(b+\varepsilon_b) = x + A^{-1}\varepsilon_b =: x + \varepsilon_x$, introducing some error ε_x. We want to measure ε_x compared to ε_b. So that we can say a $\%p$ change in b yields a $\%p'$ changed in x, we normalize the errors ε_b and ε_x with respect to b and x.

The *condition number* of A is the maximum c such that

$$\frac{\|\varepsilon_x\|}{\|x\|} < c\frac{\|\varepsilon_b\|}{\|b\|}, \tag{5}$$

for all x, b, ε_x, and ε_b such that $Ax = b$ and $A\varepsilon_x = \varepsilon_b$.

If A has a small condition number, then a small relative change in b results in a small relative change in x. So, given A, how do we compute this c?

7.2.1 Condition number of a positive definite A

Considering the case that A is posive definite, the min and max eigenvalue λ_m and λ_M are positive integers, and we get:

$$\lambda_m \leq \frac{\|Ax\|}{\|x\|} \leq \lambda_M.$$

So as $Ax = b$ and $A\varepsilon_x = \varepsilon_b$ we have

$$\frac{\|b\|}{\|x\|} \leq \lambda_M \qquad \text{and} \qquad \lambda_m \leq \frac{\|\varepsilon_b\|}{\|\varepsilon_x\|}.$$

ver. 2019/12/05

Plugging these into (5) yields $c \leq \lambda_M/\lambda_m$, and taking x as an eigenvector for λ_M and ε_x as one for λ_m we get equality, so

$$c = \frac{\lambda_M}{\lambda_m}$$

is the condition number for A.

Problem

If A is orthogonal, $b = (1, 3, 4, 1)$ and $\varepsilon_b = (\frac{1}{10}, 0, \frac{2}{10}, \frac{1}{10})$, and $Ax = b$ and $A\varepsilon_x = \varepsilon_b$, then what is (an upper bound for) $\|\varepsilon_x\|\|x\|$?

Problem

Do A and A^{-1} have the same condition number?

7.2.2 Condition number of more general A

When A is non-symmetric (non-positive definite) what are the maximum and minimum values (respectively) of

$$\frac{\|b\|}{\|x\|} \text{ and } \frac{\|\varepsilon_b\|}{\|\varepsilon_x\|}?$$

That is; what are the max and min values of $\|Ax\|/\|x\|$? Are they still the max and the min eigenvalues?

No! The proof on this bound used ellipses in the symmetric case. It required that there was a full complement of eigenvectors.

Example 7.1. The matrix $A = \left[\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right]$ has eigenvalues $\lambda_1 = \lambda_2 = 1$ with a single eigenvector $(1, 0)$. For a vector $v = (2, 2)$ we have

$$Av = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}.$$

So $\|Av\|/\|v\| = \frac{\sqrt{20}}{\sqrt{8}} = \sqrt{5/2} > 1$

Definition 7.2. The *norm* of a matrix A is

$$\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}.$$

Note

This immediately yields the useful identity

$$\|Ax\| \leq \|A\|\|x\|. \tag{6}$$

ver. 2019/12/05

Now we see that the condition number of A is $c = \|A\|\|A^{-1}\|$. Indeed, we have that

$$\frac{\|b\|}{\|x\|} \leq \|A\| \text{ and } \|A^{-1}\| \leq \frac{\|\varepsilon_b\|}{\|\varepsilon_x\|}$$

and so, as needed,

$$\frac{\|\varepsilon_x\|}{\|x\|} \leq \|A^{-1}\|\|A\|\frac{\|\varepsilon_b\|}{\|b\|}.$$

7.2.3 Error in A

Until now, we approximated $Ax = b$ by altering b. Commonly we will round all our numbers, and introduce error in A as well. Solving $(A+\varepsilon_A)(x+\varepsilon_x) = b$ we get

$$b = Ax + \varepsilon_A x + A\varepsilon_x + \varepsilon_A\varepsilon_x.$$

As we know that $Ax = b$ we therefore get that $A\varepsilon_x + \varepsilon_A(x + \varepsilon_x) = 0$ and rearranging this gives that

$$-\varepsilon_x = A^{-1}\varepsilon_A(x + \varepsilon_x).$$

Using (6) twice we get

$$\|\varepsilon_x\| \leq \|A^{-1}\| \cdot \|\varepsilon_A(x + \varepsilon_x)\| \leq \|A^{-1}\| \cdot \|\varepsilon_A\| \cdot \|(x + \varepsilon_x)\|,$$

which yields

$$\frac{\|\varepsilon_x\|}{\|(x + \varepsilon_x)\|} \leq \|A^{-1}\|\|\varepsilon_A\| = c\frac{\|\varepsilon_A\|}{\|A\|}.$$

So the relative error in x is again bound by the condition number of A times the relative error in A.

We want to compute norm then.

Well,

$$\|A\|^2 = \max \frac{\|Ax\|^2}{\|x\|^2} = \max \frac{x^T A^T Ax}{x^T x} = \max R(x)$$

where $R(x)$ is the Rayleigh Quotient of x with respect to $A^T A$. We saw that the Rayleigh quotient is maximized at the maximum eigenvector. So $\|A\|$ is the root of the max eigenvector of $A^T A$.

Problems from the text

7.2: 2, 5, 9, 10, 15, 17

ver. 2019/12/05

7.3 Computing Eigenvalues

There are many techniques for computing eigenvalues, and we will see two basic ones in this section: the power method and the QR-method. The power method is intuitive– it is very easy to see why it works and to see why such simple improvements as 'shifting' work. The QR-method is less transparent, but faster; it is called the QR-method because it uses a QR-decomposition of a matrix A. There are various ways to find a QR-decomposition of a matrix. We have seen one, the Gram-Schmidt method. We will see another using so-called Householder matrices.

Not only the power method and the QR-method, but other methods for computing eigenvalues are much faster for matrices with many zero entries. So the first step in computing the eigenvalues of A is usually to find a similar matrix with lots of zero entries. Finding the Jordan form of the matrix would be ideal, but we don't have that if we don't have the eigenvalues. We will see a method of putting a matrix into a Hessenberg form, using the same Householder matrix.

Though the algorithm for the Hessenberg form would be used before finding its QR-decomposition, we present the QR-decomposition via Householder matrics first as it is easier to understand than the Hessenberg algorithm, and helps to explain why the Hessenberg algorithm works.

7.3.1 The Householder Matrix

Recall that the projection matrix P_v such that for any vector x, $P_v x$ is the projection of x onto v is computed as $P_v = \frac{vv^T}{v^T v}$.

For a vector $x \in \mathbb{R}^n$ of length $\|x\| = \sigma$, let

$$H = H_x := I - 2P_{(x+\sigma z)} = I - 2\frac{(x+\sigma z)(x+\sigma z)^T}{\|x+\sigma z\|^2}$$

where z is the standard normal basis vector e_1.

Lemma 7.3. *Let H be the Householder matrix for a vector x. The following are true.*

i. $Hx = -\sigma z$

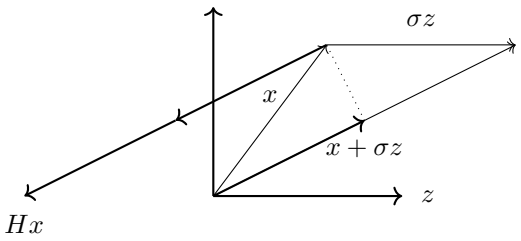
ii. H *is invertible*

iii. H *is symmetric*

iv. H *is orthogonal*

Proof. We prove *i.* with a picture:

ver. 2019/12/05



As the only vector H takes to 0 is 0, it is invertible, which gives *ii*.

To see *iii*., that H is symmetric, observe that on normalising $x + \sigma z$ to u H can be written as $H = I - 2uu^T$. So

$$H^T = (I - 2uu^T)^T = I - 2(u^T)^T u^T = I - 2uu^T = H.$$

To see *iv*., that H is orthogonal, observe that

$$H^T H = H^2 = (I - 2uu^T)^2 = I^2 - 4uu^T + 4u \overbrace{u^T u}^{=1} u^T = I^2 = I.$$

□

7.3.2 QR-decomposition with the Householder matrix

Recall the Gram-Schmidt process for finding a orthonormal basis $x_1, \ldots, x_n$ from a basis $b_1, \ldots, b_n$ was to normalise x_1, and then for $j \geq 2$ to make b_j orthogonal to the x_i for $i < j$ by subtrating of the projection $\text{proj}_{x_i} b_j$ and then normalising to get x_j. Writing these vectors as the rows of a matrix Q^T we got that $Q^T A = R$ is upper triangular R and so $A = QR$ for orthogonal Q and upper triangular R, where for $i \leq j$ R_{ij} is the length of the x_i component fo b_j.

We can do the same using Householder matrices. Let $c_1, c_2 \ldots, c_n$ be the columns of A. Where $U_1 = H_1 = H_{c_1}$ is the Householder matrix for c_1 and $\sigma_1 = |c_1|$ we get that

$$U_1 A = \left[\begin{array}{c|c} \sigma_1 & ? \\ \hline 0 & \\ 0 & ? \\ 0 & \end{array}\right]$$

Now we would like to do the same with the second column. Multiplying on the right by $H_2 = I - 2P_{(c_2 + |c_2| e_2)}$ would give us a $|c_2|$ in the second entry of the second column and zeros above and below it, but it would also change our first column. Our solution is to let the second column do what it wants above the diagonal. Let c_i^* be the i^{th} column from the diagonal down. (So it is a vector in $\mathbb{R}^{n-i+1}$.) Let H_i be the $n - i$ dimensional Householder matrix for c_i^* and let

ver. 2019/12/05

U_i be the matrix we get from the identity by replacing its lower right square submatrix with H_i:

$$U_3 = \left[\begin{array}{cc|c} 1 & 0 & \\ 0 & 1 & 0 \\ \hline & 0 & H_3 \\ & & \end{array}\right]$$

We see the following where $\sigma_2 = |c_2^*|$,

$$U_2U_1A = \left[\begin{array}{c|ccc} 1 & 0 & 0 & 0 \\ \hline 0 & & & \\ 0 & & H_2 & \\ 0 & & & \end{array}\right] \left[\begin{array}{c|c} \sigma_1 & ? \\ \hline 0 & \\ 0 & ? \\ 0 & \end{array}\right] = \left[\begin{array}{cc|c} \sigma_1 & r_{12} & \\ 0 & \sigma_2 & ? \\ 0 & 0 & \\ 0 & 0 & \end{array}\right].$$

Proceeding in this way we get that $U_nU_{n-1}\dots U_1A = R$ for some upper triangular R. As H_i is symmetric and orthogonal, so is U_i so $Q^T = U_nU_{n-1}\dots U_1$ is orthogonal (but not usually symmetric). So $A = QR$, where $Q = (Q^T)^{-1}$. Notice also that r_{12} is the length of the projection of c_2 onto the first column of Q. One can make more general such statements and conclude that the is the QR-decomposition of A we get from the Gram-Schmidt algorithm on the columns of A.

This gives us another way to compute the QR-decomposition of matrix A, and R has lots of nice zeros. Unfortunatley though it is not similar to A, so this doesn't help us in our task of finding a matrix similar to A with lots of zeros. What if we multiply the $U_i = U_i^{-1}$ on the right of A as well. This would give us something similar to A, but the first column of

$$U_1AU_1 = H_1AH_1 = \left[\begin{array}{c|c} \sigma_1 & ? \\ \hline 0 & \\ 0 & ? \\ 0 & \end{array}\right] H_1$$

is no longer mostly zeros. The first column would remain unchanged if we multiplied on the right by $\left[\begin{smallmatrix} 1 & 0 & 0 & 0 \\ 0 & & & \\ 0 & & H_1' & \\ 0 & & & \end{smallmatrix}\right]$ for some H_1' instead of H_1. Nice. Oh, but shoot, then we have to do the same on the left too. It turns out that works okay.

7.3.3 Making A tridiagonal (Hessenberg Form)

Decompose A as follows.

$$A = \left[\begin{array}{c|c} a_{11} & ? \\ \hline A_{*1} & A' \end{array}\right]$$

ver. 2019/12/05

With this, we see that

$$\underbrace{\left[\begin{array}{c|ccc} 1 & 0 & 0 & 0 \\ \hline 0 & & & \\ 0 & & H_1 & \\ 0 & & & \end{array}\right]}_{U_1^{-1}} \left[\begin{array}{c|c} a_{11} & ? \\ \hline A_{*1} & A' \end{array}\right] \underbrace{\left[\begin{array}{c|ccc} 1 & 0 & 0 & 0 \\ \hline 0 & & & \\ 0 & & H_1 & \\ 0 & & & \end{array}\right]}_{U_1} = \left[\begin{array}{c|c} a_{11} & ? \\ \hline H_1 A_{*1} & ? \end{array}\right]$$

So taking H_1 as the householder matrix for A_{*1}, this becomes

$$\left[\begin{array}{c|c} a_{11} & ? \\ \hline -\sigma & \\ 0 & ? \\ 0 & \end{array}\right]$$

As H_1 is orthogonal and symmetric so is U_1, so $U_1^{-1} = U_1^T = U_1$.

Note

Summarizing: For a matrix A, let A_{*1} be the first column of A with its first element removed, and let H_1 be the Householder matrix for A_{*1}. Letting

$$U_1 = \left[\begin{array}{c|ccc} 1 & 0 & 0 & 0 \\ \hline 0 & & & \\ 0 & & H_1 & \\ 0 & & & \end{array}\right]$$

we get that

$$U_1 A U_1 = \left[\begin{array}{c|c} a_{11} & ? \\ \hline -\sigma & \\ 0 & ? \\ 0 & \end{array}\right]$$

where $\sigma = \|A_{*1}\|$.

We have 'almost' cleared under the first diagonal. Now we attack the second. Our $U_1 A U_1$ decomposes as

$$U_1 A U_1 = \left[\begin{array}{cc|c} a_{11} & ? & ? \\ -\sigma & a'_{22} & ? \\ \hline 0 & A_{*2} & A'_2 \\ 0 & & \end{array}\right],$$

so taking

$$U_2 = \left[\begin{array}{cc|c} 1 & 0 & 0 \\ 0 & 1 & \\ \hline 0 & & H_2 \\ & & \end{array}\right],$$

where H_2 is the householder matrix projecting $x = A_{*2}$ onto $\|x\|z$ where $z = (0, 0, 1, 0 \ldots, 0)$, we will replace that A_{*2} under the a'_{22} with a column $(-\|A_{*2}\|, 0, 0 \ldots,)$. We eventually get to a

$$A' = U_{n-1} \ldots U_1 A U_1 \ldots U_{n-1}$$

which is similar to A and 'almost' upper triangular. If A is symmetric then so is A', and so it is tridiagonal.

Example 7.4. We tridiagonalise $A = \left[\begin{array}{ccc} 2 & 1 & 1 \\ 1 & 1 & 2 \\ 1 & 2 & 3 \end{array}\right]$. Writing it as $\left[\begin{array}{c|cc} 2 & 1 & 1 \\ \hline 1 & 1 & 2 \\ 1 & 2 & 3 \end{array}\right]$, our A_1^* is $x = \left[\begin{array}{c} 1 \\ 1 \end{array}\right]$ and our first z is $z = \left[\begin{array}{c} 1 \\ 0 \end{array}\right]$ so we want the Householder matrix H_1 taking

ver. 2019/12/05

x to $-\sigma z = -\|x\|(1,0,0)$. This is $I - 2P_{x+\sigma z}$ so where

$$x + \sigma z = \begin{bmatrix}1\\1\end{bmatrix} + \sqrt{2}\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}1+\sqrt{2}\\1\end{bmatrix},$$

we (use a computer to) get

$$H_1 = \begin{bmatrix}1 & \\ & 1\end{bmatrix} - \frac{1}{1+\sqrt{2}}\begin{bmatrix}3+2\sqrt{2} & 1+\sqrt{2}\\ 1+\sqrt{2} & 1\end{bmatrix} = \frac{\sqrt{2}}{2}\begin{bmatrix}-1 & -1\\ -1 & 1\end{bmatrix}.$$

This gives that

$$\begin{bmatrix}1 & 0 & 0\\ 0 & -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2}\\ 0 & \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2}\end{bmatrix} A \begin{bmatrix}1 & 0 & 0\\ 0 & -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2}\\ 0 & \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2}\end{bmatrix} = \begin{bmatrix}2 & -\sqrt{2} & 0\\ -\sqrt{2} & 4 & -1\\ 0 & -1 & 0\end{bmatrix}.$$

Now, in theory, we know how to find for any matrix A a similar tridiagonal matrix. This has a lot of zeros (even if this seems an over-statement for our 3×3 example,) so we are ready to us iterative methods to compute the eigenvalues.

Note

Recall that when $a_1, a_2, a_3, \ldots$ is a series of numbers limiting to a value a, the *rate of convergence* of the sequence is

$$\lim_{n\to\infty} \frac{|a - a_{n+1}|}{|a - a_n|}.$$

If this is less than 1 then the series converges to a. A small rate (close to 0) is better. It means that the sequence converges more quickly. If

$$\lim_{n\to\infty} \frac{|a - a_{n+1}|}{|a - a_n|^2} = c < 1,$$

then the convergence is *quadratic*. This is even better.

7.3.4 The Power Method

For this section we order the eigenvalues of a matrix A such that $|\lambda_1| \leq |\lambda_2| \leq \cdots \leq |\lambda_n|$, and call λ_1 the minimum eigenvalue and λ_n the maximum.

This method works to find the maximum eigenvalue λ_n of A as long as $|\lambda_{n-1}| \neq |\lambda_n|$. Indeed, any vector is of the form

$$u = c_1x_1 + c_2x_2 + \cdots + c_nx_n$$

and almost any random vector has $c_n \neq 0$.

If this holds, then

$$A^k u = c_1\lambda_1^k x_1 + c_2\lambda_2^k x_2 + \cdots + c_n\lambda_n^k x_n.$$

Note:
If your maximum eigenvalue is complex, for example, then the conjugate will have the same absolute value and so the power method doesn't work so well. But if A is symmetric, then this is not a problem, and most random symmetric matrices will be okay.

ver. 2019/12/05

Normalising this to

$$u_k = \frac{A^k u}{\|A^k u\|} = \frac{1}{\|A^k u\|} c_1 \lambda_1^k x_1 + c_2 \lambda_2^k x_2 + \cdots + c_n \lambda_n^k x_n$$

we get that u_k approaches $\frac{x_n}{\|x_n\|}$ which is a unit eigenvector for λ_n. Once we have this, we can compute $\lambda_n = \frac{\|Ax_n\|}{\|x_n\|}$.

Note

The rate of convergence is $\frac{|\lambda_{n-1}|}{|\lambda_n|}$, so this converges better when there is a big gap between λ_n and the next eigenvalue.

Example 7.5. We find the first eigenvalue of the matrix $A = \begin{bmatrix} 2 & 3 & 1 \\ 3 & 0 & 3 \\ 1 & 3 & 4 \end{bmatrix}$, by choosing a random vector $u = (3, 4, 5)$ and computing $u_i = A^i u$ for several i. This gives

$$u_1 = (23, 24, 35), u_2 = (153, 174, 235), u_3 = (1063, 1164, 1615) \ldots.$$

It is hard to see that this is converging to any eigenvector, but this becomes more clear when we normalise the approximations, computing rather $u_i = A^i u / \|A^i u\|$:

$$u_1 = (.476, .497, .725), u_2 = (.463, .527, .712), u_3 = (.471, .515, .715), \ldots$$

7.3.5 The Inverse Power Method

Doing the same for A^{-1} we can find λ_1. The rate of convergence of the inverse power method is $|\lambda_1|/|\lambda_2|$. There is a nice way that we can boost this considerably with a good approximation σ of λ_1, but shifting A. This is called the *shifted inverse power method*.

We simply do the inverse power method on $A - I\sigma$. This has eigenvalues $\lambda_1 - \sigma, \lambda_2 - \sigma, \ldots, \lambda_n - \sigma$, so the inverse power method converges with rate $|\lambda_1 - \sigma|/|\lambda_2 - \sigma|$.

Problem

Why might doing a 'shifted power method' be difficult?

Problem

Once you know the largest eigenvalue and vector, how could you use the power method to find the second eigenvalue?

All of these versions of the power method have their uses, but tend to be slow. In practice the QR-method works better, though it is less intuitive.

ver. 2019/12/05

7.3.6 The QR-method

We can compute the QR-decomposition of A using Gram-Schmidt or Householder matrices or in the text book there is another method based on subspace rotation that is even faster. However we do it, we can use this decomposition to iteratively triangulate A via the QR-method.

We can decompose $A_0 = A = QR$ where Q is orthogonal and R is upper triangular, so $R = Q^{-1}A$. The matrix $A_1 = RQ = Q^{-1}RQ$ is similar to A_0 and as it is the product of some Q with an upper triangular R, it tends to have most of its weigth up in the upper right traingle. This is a vague statement, that we will not prove, but lets see what we mean with an example.

Example 7.6. The matrix $A_0 = A = \begin{bmatrix} -.66 & .58 & -.52 \\ .68 & -.20 & .47 \\ .26 & -.81 & -.42 \end{bmatrix}$ has (approximate) QR decomposition $A_0 = Q_0R_0$ where

$$Q_0 = \begin{bmatrix} .69 & -.69 & -.28 \\ -.11 & -.44 & .88 \\ -.75 & -.56 & -.44 \end{bmatrix} \text{ and } R_0 = \begin{bmatrix} 1 & .75 & -.56 \\ 0 & -.75 & -.56 \\ 0 & 0 & .28 \end{bmatrix}.$$

This gives $A_1 = R_0Q_0 = \begin{bmatrix} -.38 & .62 & 1.2 \\ & .50 & .62 & -.41 \\ -.20 & -.16 & -.10 \end{bmatrix}$. Taking the QR factorisation $A_1 = Q_1R_1$ of this we can then compute $A_2 = R_1Q_1 = \begin{bmatrix} .5 & .31 & -1 \\ .-06 & .56 & .75 \\ 0 & -.19 & -.5 \end{bmatrix}$.

The entries in the top right are getting larger (in absolute value) while those in the lower left are getting smaller. Continuing in this way, we get a series of matrices similar to A that are getting closer and closer to upper triangular, so the diagonals are approaching the eigenvalues of A.

Problems from the text

7.3: 1, 2, 3, 6, 7, 8, 10

7.4 Iterative Methods for solving $Ax = b$

Unlike for finding eigenvalues, we can solve the equations $Ax = b$ algebraically. We do this using Gaussian elimination. For an $n \times n$ matrix A it takes about $n^3/3$ basic operations (additions or multiplications of matrix entries). Indeed, clearing under the i^{th} pivot takes about $(n-i)^2$ operations and $\sum_{i=1}^{n}(n-i)^2 = \sum_{i=1}^{n} i^2 \approx n^3/3$. Back substitution takes about n^2 operations.

With iterative methods we can replace elimination with calculations that take about n^2 operations. For big matrices this is important.

The techniques we look at for solving $Ax = b$ all involve sprlitting the matrix

Note:
Iterative methods of course, give approximations rather than exact solutions, but as with computing eigenvalues, the approximations can be made arbitrarily good, so for most applications this is fine. If I own you $\sqrt{2}$ dollars, you will probably settle for 1.414 dollars.

ver. 2019/12/05

A as $A = S - T$ for some invertible S. If we do this we have

$$\begin{aligned} Ax = b &\iff (S-T)x = b \\ &\iff Sx - Tx = b \\ &\iff Sx = Tx + b \\ &\iff x = S^{-1}Tx + S^{-1}b \end{aligned}$$

Starting with an approximation x_1 of x we then calculate

$$x_2 = S^{-1}Tx_1 + S^{-1}b.$$

So why do we expect x_2 to be a better approximation of x? Well, let $\varepsilon_1 = x - x_1$ be the error in our first approximation. We have

$$\begin{aligned} x &= S^{-1}Tx + S^{-1}b \\ &= S^{-1}T(x_1 + \varepsilon_1) + s^{-1}b \\ &= S^{-1}Tx_1 + S^{-1}b + S^{-1}T\varepsilon_1 \\ &= x_2 + S^{-1}T\varepsilon_1 \end{aligned}$$

As long as $S^{-1}T\varepsilon_1 < \varepsilon_1$, x_2 is closer to x than x_1 is. We can assure this if $S^{-1}T$ has norm less than 1. So the trick is to find S and T such that $\|S^{-1}T\| < 1$ is small, and such that S^{-1}, $S^{-1}b$ and $S^{-1}T$ are quick to calculate. Once we have these, computing $x^2 = S^{-1}Tx_1 + S^{-1}b$ takes about $n^2 + n$ operations, so if we iterate only constantly many times, we save time for large enough n.

7.4.1 Jacobi's Method

Jacobi's method takes S as the diagonal of A. As S is diagonal, computing S^{-1} is a quick n operations. Computing $S^{-1}b$ and $S^{-1}T$ take n and n^2 operations, so we have $n^2 + 2n$ one time operations. Computing $x_{i+1} = S^{-1}Tx_i + S^{-1}b$ takes $n^2 + n$ each time we iterate.

Example 7.7. We use Jacobi's method to approximate the solution $x \approx (-1.2, 2.9, -2.9)$ of $Ax = b$ when

$$A = \begin{bmatrix} 4 & 1 & 1 \\ -1 & 3 & 1 \\ 0 & -1 & -2 \end{bmatrix} \qquad \text{and} \qquad b = \begin{bmatrix} -5 \\ 7 \\ 3 \end{bmatrix}.$$

Decomposing

$$A = S - T = \begin{bmatrix} 4 & & \\ & 3 & \\ & & -2 \end{bmatrix} - \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

ver. 2019/12/05

we get

$$S^{-1} = \begin{bmatrix} 1/4 & & \\ & 1/3 & \\ & & -1/2 \end{bmatrix} \text{ and } S^{-1}T = \begin{bmatrix} 0 & -1/4 & -1/4 \\ 1/3 & 0 & -1/3 \\ 0 & -1/2 & 0 \end{bmatrix}.$$

The norm $\|S^{-1}T\|$ is about .587, so our error at least (approximately) half at each iteration.

Start with the guess $x_1 = (1, 1, 1)$, so we don't know it, but $\|\varepsilon_1\| \approx 4.93$. We then calculate

$$\begin{aligned} x_2 &= S^{-1}Tx_1 + S^{-1}b \approx (-1.75, 2.33, -2) & \|\varepsilon_2\| &\approx 1.22 \\ x_3 &= S^{-1}Tx_2 + S^{-1}b \approx (-1.33, 2.41, -2.66) & \|\varepsilon_3\| &\approx .57 \\ x_4 &= S^{-1}Tx_3 + S^{-1}b \approx (-1.18, 2.77, -2.70) & \|\varepsilon_4\| &\approx .28 \end{aligned}$$

We will get error of less than .01 by about x_9.

The convergence here is a bit slow for a 3×3 matrix, but on a bigger matrix an approximation of %1 in 9 iterations is okay. This method works alright, but we need this $\|S^{-1}T\| < 1$, which can be a bit limiting. It is not to hard so see though that $\|S^{-1}T\| < 1$ if A is *strongly diagonal dominant* which means that for each row, (in absolute values) the diagonal entry is at least twice the some of all the entries in the row.

7.4.2 The Gauss-Seidel method

In the Gauss-Seidel method we let S be the lower triangular part of A.

Example 7.8. With the same A and b as in the previous example we get

$$A = S - T = \begin{bmatrix} 4 & & \\ -1 & 3 & \\ 0 & -1 & -2 \end{bmatrix} - \begin{bmatrix} 0 & -1 & -1 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$S^{-1} = \frac{1}{24}\begin{bmatrix} 6 & & \\ 2 & 8 & \\ -1 & -4 & -12 \end{bmatrix} \qquad S^{-1}T = \frac{-1}{24}\begin{bmatrix} 0 & 6 & 6 \\ 0 & 2 & 10 \\ 0 & -1 & -5 \end{bmatrix}$$

We compute $x_1 = (1, 1, 1)$, $x_2 = (-1.75, 1.42, 2.21)$, $x_3 = (-1.05, 2.72, -2.85)$. It is a little bit faster here, and indeed $\|S^{-1}T\| \approx .57$ is a little smaller than for the Jacobi decomposition. But for bigger matrices the improvement is usually can be much better. And we are much more likely to get a norm below 1.

Observe that in computing

$$x_2 = \frac{-1}{24}\begin{bmatrix} 0 & 6 & 6 \\ 0 & 2 & 10 \\ 0 & -1 & -5 \end{bmatrix}\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} + \frac{1}{24}\begin{bmatrix} 6 & & \\ 2 & 8 & \\ -1 & -4 & -12 \end{bmatrix}\begin{bmatrix} -5 \\ 7 \\ 3 \end{bmatrix}$$

ver. 2019/12/05

the first coordinate is $\frac{-1}{24}(6+6)+\frac{1}{24}(-30) = -1.75$. Using this in place of the first co-oordinate 1 of x_1 gives an even better approximation of the second coordinate. Continuing in this way updating each coordinate of x_1 as we compute it, we speed up convergence significantly.

Problems from the text

7.4: 3, 6, 11, 14

ver. 2019/12/05

8 Linear Programming and Game Theory

8.1 Linear Inequalities

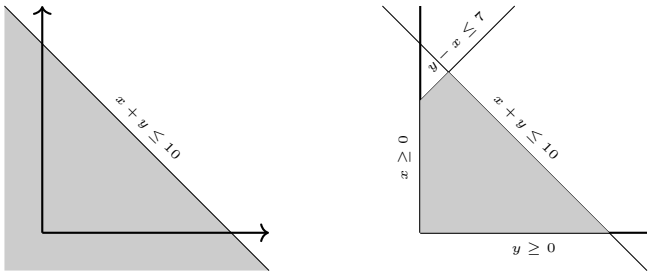
A linear programming 'problem' consists of several constraints (linear equations or linear inequalities) in several variables, and a linear objective function f in the variables. The goal is to maximise (or minimise) the objective function subject to the constraints.

We can explain the basic ideas with some simple graphical examples.

Example 8.1. A typical problem is to maximize $f(x, y) = 2x + y$ with respect to the constraints

$$\begin{aligned} x + y &\leq 10 \\ y - x &\leq 7 \\ x, y &\geq 0 \end{aligned}$$

As a linear equation would constrict us to a line in $\mathbb{R}^2$, a linear inequality restricts us to a half space.



The *feasible* points are the points that satisfy all constraints. The set of them is called the *feasible set*.

Problem

Draw the feasible points for the constraints $2x + y \geq 4$ and $2x + y \geq 3$. Draw them for $2x + y \geq 4$ and $2x - 2y \geq 4$.

Sometimes constraints can be redundant but generally each constraint 'cuts out a half plane'. The feasible set is always *convex*, which means for two points in the set, all points between them are also in the set. There are three distinct cases that can occur. The feasible set can be empty, bounded, or unbounded. In our previous example the feasible set was bounded. If we remove the constraint $x + y \leq 10$ then it would be unbounded.

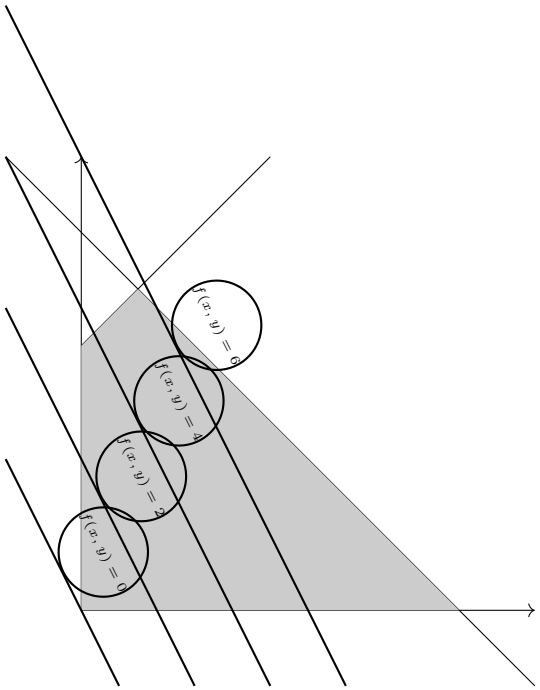
ver. 2019/12/05

Problem

Add a constraint that makes the feasible set empty.

If the feasible set is non-empty and bounded, then the objective function will have a maximum value on one (or more) of the corners (extremal points) of the feasible set. In the above example, we can simple evaluate $f(x, y) = 2x + y$ at corners $(x, y) \in \{(0, 0), (0, 7), (10, 0), (1.5, 8.5)\}$ and find that its maximum is $f(10, 0) = 20$.

A more systematic way to do this is to consider the *isotopes* of f: the lines $\{(x, y) \mid f(x, y) = c\}$. For the example above, the following are some of the isotopes:



From this is is clear that on the feasible set, $f(x, y)$ is maximum at $(10, 0)$ and minimum at $(0, 0)$. We consider the vector orthogonal to the isotopes: the isotope $2x + y = 0$ is the line $y = -2x$ or $\langle(1, -2)\rangle$. The orthogonal vector is $(2, 1)$, and f increases in the positive direction. We choose the corner 'farthest out' in that direction; ie. the corner whose projection onto $\langle(2, 1)\rangle$ is the greatest.

This seems simple enough, but finding all corners of the feasible set is a lot of work. Indeed for n variables and $M > n$ contraints, each constraint is bounded by a hyperplane, and corners occur at the intersection of n of these. There are $O(M^n)$ choices of the n hyperplanes to give such an intersection, (and very few

ver. 2019/12/05

of these are feasible). We want a better way trying all these intersection points, deciding which are feasible, and then evaluating f at them.

We want a systematic way to find the optimal corner without finding all the corners. This will come from the simplex method. The idea for the simplex method is to start at one corner, and move along some edge whose projection onto the isotope orthognal vector is positive. This increases our objective value. The fortunate thing is that if we do this until there are no more local increases, then we have the optimum corner. The simplex method thus has two main Phases: in Phase 1 we find an initial feasible point (if one exists); in Phase 2 we find the optimal feasible point.

8.1.1 Standard Form of a LP-Problem

An *LP-problem* consists of m linear inequalities in n variables and an objective function $f : \mathbb{R}^n \to \mathbb{R}$. It is in *standard form* if the following are true.

- For each variable x_i there is a positivity constraint $x_i \geq 0$.
- All other constraints are of the form $g(x_1, \ldots, x_n) \leq c$ for some c.

The example we gave above was in standard form. For a problem in standard form, if the constants c are all positive, then the origin is a feasible point. The following example, which arises from a 'natural' problem, is not in standard form. But there are several simple methods to change constraints that are non-standard into standard constraints.

Maximise $f(x) = x_1 + x_2 + x_3$ with respect to the constraints

$$\begin{aligned} x_1 + 3x_2 + x_3 &\leq 2 \\ 2x_1 - x_2 + x_3 &\geq 4 \\ x_1 \geq 0, x_2 &\leq 0 \end{aligned}$$

We have a couple of problems

- The second constraint is $\geq$ rather than $\leq$.
- The constraint for x_2 is negativity rather than positivity.
- The variable x_3 doesn't have a positivity constraint.

First we deal with the negativity of x_2 by replacing x_2 with $-x_2$. The problem becomes

Maximise $f(x) = x_1 - x_2 + x_3$ with respect to the constraints

ver. 2019/12/05

$$x_1 - 3x_2 + x_3 \leq 2$$
$$2x_1 + x_2 + x_3 \geq 4$$
$$x_1, x_2 \geq 0$$

Then we deal with the missing positivity constraint by 'splitting' x_3 into positive x_3 and negative x_4. The problem becomes

Maximise $f(x) = x_1 - x_2 + x_3 - x_4$ with respect to the constraints

$$x_1 - 3x_2 + x_3 - x_4 \leq 2$$
$$2x_1 + x_2 + x_3 - x_4 \geq 4$$
$$x_1, x_2, x_3, x_4 \geq 0$$

Finally, we flip the second constraint to:

$$-2x_1 - x_2 - x_3 + x_4 \leq -4.$$

Putting our problem into standard form, we have increased the number of variables, so the feasible space gets a little difficult to draw.

8.1.2 Matrix Form

We are in standard form. As expected, we now want to write this as a matrix equation. We might try writing

$$\begin{bmatrix} 1 & -1 & 1 & -1 \\ -2 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \leq \begin{bmatrix} 2 \\ -4 \end{bmatrix} \text{ and } \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \geq 0,$$

but this doesn't make any sense! Well, actually this is useful notation. For vectors u and v let $u \geq v$ mean that $u_i \geq v_i$ for each $i \in [n]$. But still, the techniques we know let us solve a matrix equation, not an matrix inequality. So lets add a couple more *slack* variables to make our inequalities into equalities. To replace the constraint

$$x_1 - 3x_2 + x_3 - x_4 \leq 2$$

we can simply require that the difference $s_1 = 2 - (x_1 - 3x_2 + x_3 - x_4)$ be positive. So this constraint becomes

$$x_1 - 3x_2 + x_3 - x_4 + s_1 = 2 \text{ and } s_1 \geq 0.$$

ver. 2019/12/05

Similarily replacing $-2x_1 - x_2 - x_3 + x_4 \leq 4$ with

$$-2x_1 - x_2 - x_3 + x_4 + s_2 = -4 \text{ and } s_2 \geq 0,$$

our problem is to maximize $f(x) = (1, -1, 1, -1, 0, 0) \cdot (x_1, x_2, x_3, x_4, s_1, s_2)$ over all solutions of

$$\begin{bmatrix} 1 & -1 & 1 & -1 & 1 & 0 \\ -2 & -1 & -1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ s_1 \\ s_1 \end{bmatrix} = \begin{bmatrix} 2 \\ -4 \end{bmatrix}$$

that are positive in every variable.

Problem

Make an LP-problem for the following situation, and put it into standard form and then matrix form. Solve it using goemetric intuition.

GM makes Buicks and Caddies. All cars that they make sell, and the profit per car is \$3 and \$5 respectively. It takes 2 and 3 minutes respectively to make the cars. How many of each should they make per hour to maximise their profit. (Perhaps it is clear to you here that they should only make Caddies. Lets add one more contraint.) Environmental laws say that average miles per gallon of the cars they make should be at least 18. Buicks and Caddies get respectively 17 and 20.

Problems from the text

8.1: 1 - 9

8.2 The Simplex Method

We will re-order our constraints so that the positivity constraints come first, and give them names. We solve the following LP-problem in standard form.

Example 8.2. Maximise $f(x) = 2x_1 + x_2$ with respect to the constraints

$$c_1 : x_1 \geq 0 \qquad c_2 : x_2 \geq 0 \qquad c_3 : 2x_1 + x_2 \leq 10 \qquad c_4 : x_1 - x_2 \leq 7$$

Note:
Ignore that constraint c_4 is redundant and that you know the maximum value is 10.

To prepare for the simplex method, we put this in matrix form by adding a slack variable x_i for the non-positivity constraint c_i for $i = 3, 4$:

$$x_3 = 10 - 2x_1 - x_2$$
$$x_4 = 7 - x_1 + x_2$$

ver. 2019/12/05

Setting $z = (2, 1, 0, 0)$ the problem now is to maximize $f(x) = z \cdot x$ over all positive solutions to

$$\underbrace{\begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{bmatrix}}_{A} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \underbrace{\begin{bmatrix} 10 \\ 7 \end{bmatrix}}_{b}.$$

Note

When our original standard form problem has m *original* (non-positivity) constraints, and n *original* variables we add m slack variables for a total of $n + m$ variables. So A will by an $m \times (m + n)$ matrix. The feasible set of the original problem lives in $\mathbb{R}^n$. A solution to $Ax = b$ satisfies the original constraint c_i with equality iff $x_i = 0$. A general corner is the intersection of n hyperplanes. So a feasible point is a corner if exactly n of the components x_i are 0. We call such solutions *corner solutions* and the non-zero variables are called *basic* or *active*. As we started in standard form we know that if b is positive, then we have a corner solution in which all original variables are 0 and only the slack variables are basic.

We know then we have a corner solution $s_0 = (0, 0, ?, ?)$, and from the matrix equation it is easy to see that this must be $s_0 = (0, 0, 10, 7)$. Our objective here is

$$f(0, 0, 10, 7) = z \cdot x = (2, 1, 0, 0) \cdot (0, 0, 10, 7) = 0.$$

We will get a better solution by increasing x_1 or x_2. But to maintain positive solution to

$$\begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 (= 0) \\ x_2 (= 0) \\ x_3 (= 10) \\ x_4 (= 7) \end{bmatrix} = \begin{bmatrix} 10 \\ 7 \end{bmatrix}$$

we must decrease other variables to compensate. We can only decrease the active variables x_3 or x_4. If we increase x_1 by c we see by the first row that we must decrease x_2 by $2c$, and from the second row that we must decrease x_4 by c.

This limits our c to be at most 5, increasing x_1 by 5 and decreasing x_3 by 10 and x_4 by 5 we get another solution. This gives

$$\begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 (= 5) \\ x_2 (= 0) \\ x_3 (= 0) \\ x_4 (= 2) \end{bmatrix} = \begin{bmatrix} 10 \\ 7 \end{bmatrix}.$$

And look! Exactly two components are 0 so $s_1 = (5, 0, 0, 2)$ is a corner solution. Its value is $f(5, 0, 0, 2) = 10$.

ver. 2019/12/05

Then we want to do this again. We will write it in an orderly way so that we can keep track of everything.

8.2.1 The tableaux

We are looking for the positive solutions to $Ax = b$:

$$\underbrace{\begin{bmatrix} 2 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{bmatrix}}_{A} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \underbrace{\begin{bmatrix} 10 \\ 7 \end{bmatrix}}_{b},$$

which maximizes $z \cdot x = (2, 1, 0, 0) \cdot (x_1, x_2, x_3, x_4)$, and we have a solution $s_0 = (0, 0, 10, 7)$ with $f(s_0) = 0$.

We write this in a tableax

s_0	x_1	x_2	x_3	x_4	
	A				b
	z				

$=$

s_0	x_1	x_2	x_3	x_4	
	2	1	1	0	10
	1	−1	0	1	7
	2	1	0	0	

But actually, we want to distinguish the basic and the non-basic variables. We do this by re-ordering columns. The variable labels on the top keep track of this order for us.

s_0	x_3	x_4	x_1	x_2	
x_3	1	0	2	1	10
x_4	0	1	1	−1	7
	0	0	2	1	0

The variables x_3 and x_4 that we have written on the right remind us that these are the basic variables. As they are non-basic, the variables corresponding to the columns on the right are 0, and the matrix below the basic variables is the identity, we can read off the values of the basic variables on the very right of the correspoding row: $x_3 = 10$ and $x_4 = 7$. We have also put $z \cdot s_0$ in the bottom right. (Actually, it is $-z \cdot s_0$.)

Our first step above was to make x_1 basic. Computing for x_3 that $10/2 = 5$ and for x_4 that $7/1 = 7$, we saw that x_3 is the stronger bound on how much we can increase x_1. So when x_1 'enters' as basic, x_3 'exits'. We reorder the columns to reflect this.

s_0	x_1	x_4	x_3	x_2	
x_1	2	0	1	1	10
x_4	1	1	0	−1	7
	2	0	0	1	0

ver. 2019/12/05

Now we 'eliminate' this and observe that the calculations we do are exactly those that we did to find s_1.

s_1	x_1	x_4	x_3	x_2	
x_1	1	0	1/2	1/2	5
x_4	0	1	−1/2	−3/2	2
	0	0	−1	0	−10

So how do we continue. In the table for s_0 the bottom row told us which non-basic variables would be useful. Both x_1 and X_2 had positive values 2 and 1 in the bottom row, so increasing either of the would increase $z \cdot x$. Now neither of them are positive, this tells us that neither of them will help. So we are done. **We have maximized the objective function when there are no positive values left in our objective (bottom) row**.

Problem

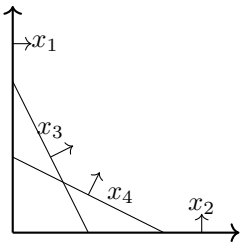
Why? What did we do to get these values on the bottom row.

8.2.2 Minimisation Problems

Consider the following *LP*-problem.

Example 8.3. Minimise $x_1 + x_2$ subject to the constraints

$$x_1, x_2 \geq 0$$
$$x_1 + 2x_2 \geq 6$$
$$2x_1 + x_2 \geq 6$$



The feasible set is unbounded, but the objective function decreases towards that middle corner, so there will be a minimum, (but no maximum). Lets discover this with the simplex method though.

Introducing slack variables $x_3 = 2x_1 + x_2 - 6$ and $x_4 = x_1 + 2x_2 - 6$ we must minimize $(x_1, x_2, x_3, x_4) \cdot (1, 1, 0, 0)$ over the positive solutions to

$$2x_1 + x_2 - x_3 = 6$$
$$x_1 + 2x_2 - x_4 = 6.$$

From the picture we see that there is a corner with $x_1 = x_3 = 0$. We have to find the x_2 and x_4 values for this to get our initial solution, but we do much the

ver. 2019/12/05

same way we update our solutions: write out our initial tableaux and eliminate it.

s_0	x_2	x_4	x_1	x_3	
x_2	1	0	2	-1	6
x_4	2	-1	1	0	6
	1	0	1	0	?

$\rightarrow$

s_0	x_2	x_4	x_1	x_3	
x_2	1	0	2	-1	6
x_4	0	1	3	-2	6
	0	0	-1	1	-6

Once we find that $(x_1, x_2) = (0, 6)$ we compute $f(x) = 6$ so can put it in the corner of the tableaux. Now we can carry on essentially as we did before. We see that the objective row co-efficient of x_1 is negative, so as we are trying to minimize the objective function, we want x_1 to enter as basic. The x_2 rows shows us that we can increase x_1 by $6/2 = 3$ and the x_4 row allows $6/3 = 2$. So x_4 exits, and we have

s_1	x_2	x_1	x_4	x_3	
x_2	1	2	0	-1	6
x_1	0	3	1	-2	6
	0	-1	0	1	-6

$\rightarrow$

s_1	x_2	x_1	x_4	x_3	
x_2	1	0	$-2/3$	$7/3$	2
x_1	0	1	$1/3$	$-2/3$	2
	0	0	$1/3$	$1/3$	-4

As no more entries in the bottom row are negative, we are done, with $f(2, 2, 0, 0) = 4$.

8.2.3 Phase 1: Finding the initial corner

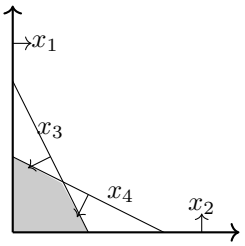
In our first example of the simplex method, Example 8.2, our problem was in standard form and our vector b was positive, so we were able to assert that there was a corner in which all non-slack variables were non-basic (ie. 0). In Example 8.3 we saw that if we knew non-basic variables of a corner, we can use a tableaux step to find the other variables. However, we have not seen how to find the non-basic variables of a corner for a problem that is not in standard form. In Example 8.3 we cheated and used a drawing of the feasible set. This is not generally possible. So how should we really find the non-basic variables of a corner?

Lets see how we should have done it for Example 8.3.

We make an auxillary problem with a dual feasible set by 'flipping' the slack variables for the non-negativity constraints.

ver. 2019/12/05

$$x_1, x_2, x_3, x_4 \geq 0$$
$$x_1 + 2x_2 + x_3 \leq 6$$
$$2x_1 + x_2 + x_4 \leq 6$$



This feasible set and the original one share a corner, and it is the corner in which $x_3 = x_4 = 0$. To find this, it is enough to solve the auxillary problem with an objective function:

$$\text{minimise } x_3 + x_4.$$

We will use the following example of the Diet Problem in the next section.

Example 8.4. In a meal we need to eat 20g of protien and 10mg of iron, and we want to do so a cheaply as possible from a menu of eggs and beans. 100g of egg costs \$2 and provides 5g of protien and 1mg of iron. 100g of beans costs \$1 and provides 4g of protien and 3mg of iron. How much of each of beans and eggs should we eat?

Letting x_1 and x_2 be the number of hundreds of grams of eggs and beans respectively, we want to minimize $(2, 1) \cdot x$ over positive solutions to

$$5x_1 + 4x_2 \geq 20$$
$$x_1 + 3x_2 \geq 10$$

Adding slack variables $x_3 = 20 - (5x_1 + 4x_2)$ and $x_4 = 10 - (x_1 + 3x_2)$ our feasible set consists of solutions to

$$\begin{bmatrix} 5 & 4 & -1 & 0 \\ 1 & 3 & 0 & -1 \end{bmatrix} x = \begin{bmatrix} 20 \\ 10 \end{bmatrix}.$$

The origin $x_1 = x_2 = 0$ is not a feasible point (in standard form b has negative coordinates,) so we use the auxillary problem to find an initial solution:

Minimise $x_3 + x_4$ over positive solutions to

$$5x_1 + 4x_2 + x_3 = 20$$
$$x_1 + 3x_2 + x_4 = 10.$$

We start with the initial solution $s_0 = (0, 0, 20, 10)$, and observe that as the slack variables have flipped, this has a slighly different matrix equation.

$$\begin{bmatrix} 5 & 4 & 1 & 0 \\ 1 & 3 & 0 & 1 \end{bmatrix} x = \begin{bmatrix} 20 \\ 10 \end{bmatrix}.$$

ver. 2019/12/05

$$\begin{array}{c|cc|cc|c} s_0 & x_3 & x_4 & x_1 & x_2 & \\ \hline x_3 & 1 & 0 & 5 & 4 & 20 \\ x_4 & 0 & 1 & 1 & 3 & 10 \\ \hline & 1 & 1 & 0 & 0 & 30 \end{array} \to \begin{array}{c|cc|cc|c} s_0 & x_3 & x_4 & x_1 & x_2 & \\ \hline x_3 & 1 & 0 & 5 & 4 & 20 \\ x_4 & 0 & 1 & 1 & 3 & 10 \\ \hline & 0 & 0 & -6 & -7 & -30 \end{array} \to \begin{array}{c|cc|cc|c} s_1 & x_1 & x_4 & x_3 & x_2 & \\ \hline x_1 & 5 & 0 & 1 & 4 & 20 \\ x_4 & 1 & 1 & 0 & 3 & 10 \\ \hline & -6 & 0 & 0 & -7 & -30 \end{array} \to$$

$$\begin{array}{c|cc|cc|c} s_1 & x_1 & x_4 & x_3 & x_2 & \\ \hline x_1 & 1 & 0 & \frac{1}{5} & \frac{4}{5} & 4 \\ x_4 & 0 & 1 & \frac{-1}{5} & \frac{11}{5} & 6 \\ \hline & 0 & 0 & \frac{6}{5} & \frac{-11}{5} & -6 \end{array} \to \begin{array}{c|cc|cc|c} s_2 & x_1 & x_2 & x_3 & x_4 & \\ \hline x_1 & 1 & \frac{4}{5} & \frac{1}{5} & 0 & 4 \\ x_2 & 0 & \frac{11}{5} & \frac{-1}{5} & 1 & 6 \\ \hline & 0 & \frac{-11}{5} & \frac{6}{5} & 0 & -6 \end{array} \to \begin{array}{c|cc|cc|c} s_2 & x_1 & x_2 & x_3 & x_4 & \\ \hline x_1 & 1 & 0 & \frac{3}{11} & \frac{-4}{11} & \frac{20}{11} \\ x_2 & 0 & 1 & \frac{-1}{11} & \frac{5}{11} & \frac{30}{11} \\ \hline & 0 & 0 & 1 & 1 & 0 \end{array}$$

The simplex method should yield an optimal corner $s_2 = (\frac{20}{11}, \frac{30}{11}, 0, 0)$. This is our starting corner for the main tableaux. Recall however that not only our cost function changes, but that that our auxillary problem had a different matrix equation that the original problem. In particular, the variables x_3 and x_4 change by a factor of -1. We have to account for this in our new initial solution, and so it is $t_0 = (\frac{20}{11}, \frac{30}{11}, -0, -0) = (\frac{20}{11}, \frac{30}{11}, 0, 0)$.

We could now restart with a fresh tableaux:

$$\begin{array}{c|cc|cc|c} t_0 & x_1 & x_2 & x_3 & x_4 & \\ \hline x_1 & 5 & 4 & -1 & 0 & 20 \\ x_2 & 1 & 3 & 0 & -1 & 10 \\ \hline & 2 & 1 & 0 & 0 & 0 \end{array}$$

and eliminate as before, but this would be redoing alot of the calculations we've already done. The cost row will be different, but our elimination of the main part of the matrix will be as before, only x_3 and x_4 have been multiplied by -1. So we start with

$$\begin{array}{c|cc|cc|c} t_0 & x_1 & x_2 & x_3 & x_4 & \\ \hline x_1 & 1 & 0 & \frac{-3}{11} & \frac{4}{11} & \frac{20}{11} \\ x_2 & 0 & 1 & \frac{1}{11} & \frac{-5}{11} & \frac{30}{11} \\ \hline & 2 & 1 & 0 & 0 & 0 \end{array} \to \begin{array}{c|cc|cc|c} t_0 & x_1 & x_2 & x_3 & x_4 & \\ \hline x_1 & 1 & 0 & \frac{-3}{11} & \frac{4}{11} & \frac{20}{11} \\ x_2 & 0 & 1 & \frac{1}{11} & \frac{-5}{11} & \frac{30}{11} \\ \hline & 0 & 0 & \frac{5}{11} & \frac{-3}{11} & \frac{-70}{11} \end{array}$$

$$\begin{array}{c|cc|cc|c} t_1 & x_4 & x_2 & x_3 & x_1 & \\ \hline x_4 & \frac{4}{11} & 0 & \frac{-3}{11} & 1 & \frac{20}{11} \\ x_2 & \frac{-5}{11} & 1 & \frac{1}{11} & 0 & \frac{30}{11} \\ \hline & \frac{-3}{11} & 0 & \frac{5}{11} & 0 & \frac{-70}{11} \end{array} \to \begin{array}{c|cc|cc|c} t_1 & x_4 & x_2 & x_3 & x_1 & \\ \hline x_4 & 1 & 0 & \frac{-3}{4} & \frac{11}{4} & 5 \\ x_2 & 0 & 1 & \frac{-1}{4} & \frac{5}{4} & 5 \\ \hline & 0 & 0 & \frac{1}{4} & \frac{3}{4} & -5 \end{array}$$

Problems from the text

8.2: 1, 2, 7, 8, 9.
Note that in #1, r is the cost row; $r > 0$ means that all entries are positive.

ver. 2019/12/05

8.3 The Dual Problem

For every (primal) LP-problem

$$(P) \text{ Minimise } c \cdot x \text{ subject to } Ax \geq b \text{ and } x \geq 0$$

there is a dual LP-problem

$$(D) \text{ Maximise } x \cdot b \text{ subject to } A^T x \leq c \text{ and } x \geq 0.$$

These two problems are closely related via the following theorem.

Theorem 8.5. *If (P) and (D) both have non-empty feasible sets then the minimum $c \cdot x$ over the feasible x for (P) is equal to the maximum $x \cdot b$ over the feasible x for (D).*

Before we prove this, lets see an example of a primal and dual problem.

Recall the example of the Diet problem we had. We had to minimise $(2, 1) \cdot x$ over the positive solutions of

$$\begin{bmatrix} 5 & 4 \\ 1 & 3 \end{bmatrix} x \geq \begin{bmatrix} 20 & 10 \end{bmatrix}.$$

If this is our primal problem (P), then the following describes our dual problem (D).

There is a chemist who makes supplements of protien and iron. She must set the prices y_1 and y_2 of one gram of each of these supplements so as to maximise the amount she sells, with the understanding that people will only buy them if they are cheaper than buying beans and eggs. So, she wants to maximise $y \cdot (20, 10)$ over positive solutions of

$$5y_1 + 1y_2 \leq 2\$$$
$$4y_1 + 3y_2 \leq 1\$$$

The first constraint comes from the eggs. If the chemist doesn't satisfy this, it is cheaper to buy eggs. If she doesn't satisfy the second, it is cheaper to buy beans.

Problem

Solve this problem with a tableaux.

Proof of Theorem 8.5. One part of the proof of this is easy. Indeed for any feasible points x_D of D and x_P or P we have

$$x_D \cdot b \leq x_D(Ax_P) = (x_D A)x_p = (A^T x_D)^T x_P \leq c \cdot x_P.$$

ver. 2019/12/05

To finish the proof it is enough to find feasible points x_D^* and x_P^* such that $x_D^* \cdot b = c \cdot x_P^*$. To do this we introduce a 'matrix' description of the tableaux steps of the simplex method.

Consider the primal problem of minimizing $c \cdot x$ subject to $Ax \leq b$ (in standard form). We set slack variables $w = b - Ax$ and then solve

$$\begin{bmatrix} A & -I \end{bmatrix} \begin{bmatrix} x \\ w \end{bmatrix} = b \text{ for } \begin{bmatrix} x \\ w \end{bmatrix} \geq 0.$$

We do this with a tableaux step

$$\begin{array}{|c|c|c|} \hline A & -I & b \\ \hline c & 0 & 0 \\ \hline \end{array} \xrightarrow{\text{rearrange columns}} \begin{array}{|c|c|c|} \hline B & N & b \\ \hline c_B & c_N & 0 \\ \hline \end{array} \xrightarrow{\text{eliminate}} \begin{array}{|c|c|c|} \hline I & B^{-1}N & B^{-1}b \\ \hline 0 & c_N - c_B B^{-1} N & -c_B B^{-1} b \\ \hline \end{array}$$

Now, we do this several times, but if we knew which variables were basic in the final solution we could choose those columns from the start and put them into B in one step. So we can assume that this final tableaux is our optimal solution. Thus $c_N - c_B B^{-1} N \geq 0$ (as this is a minimization problem) and $x^* = \begin{bmatrix} B^{-1}b \\ 0 \end{bmatrix}$ is the optimal corner, and has optimal value $\begin{bmatrix} c_B & c_N \end{bmatrix} \cdot x^* = c_B B^{-1} b$.

Now taking $y^* = c_B B^{-1}$ we get $y^* \cdot b = c \cdot X^*$ as needed, So it is enough to show that y^* is a feasible solution for the dual problem D. For this we must show $yA \geq c$ and $y \geq 0$ which we ca write as

$$y \cdot \begin{bmatrix} A & -I \end{bmatrix} \leq \begin{bmatrix} c & 0 \end{bmatrix}.$$

Doing the same reaarrangement of columns as we did for the tableaux step above, (and reordering the entries of y) this becomes

$$y \cdot \begin{bmatrix} B & N \end{bmatrix} \leq \begin{bmatrix} c_B & c_N \end{bmatrix}$$

which means we must satisfy $y \leq c_B B^{-1}$ and $yN \leq c_N$. As $y^* = c_B B^{-1}$ it satisfies the former, and as $c_N - c_B B^{-1} N \geq 0$ we have $y * N = c_B B^{-1} N < c_N$. □

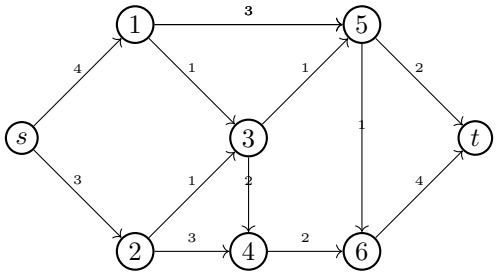
Problems from the text

8.3: 2, 3, 5, 6, 7, 8, 9

8.4 Network Models

A network flow problem consists of a graph G with a set $V = V(G)$ of vertices (or nodes) and a set $E = E(G) \subset V \times V$ of edges, each edge $e \in E$ having a non-negative capacity c. For example:

ver. 2019/12/05



Here have vertex set $\{s, t, 1, 2, 3, 4, 5, 6\}$ and for the edge e_{ij} from i to j we have a capacity c_{ij}. So, for example $c_{15} = 3$. (If there is no edge from i to j it is useful to write $c_{ij} = 0$.) A *flow* is an assignment $f : E \to \mathbb{R}$ of 'flows' such that for each edge e_{ij} we have

$$0 \leq f(e_{ij}) \leq c_{ij},$$

and for each vertex $j \in V \setminus \{s, t\}$ flow through j is preserved:

$$\sum_i f(e_{ij}) = \sum_k f(e_{jk}).$$

The *value* of a flow f is the flow $|f| := \sum_i f(e_{si}) = \sum_j f(e_{jt})$ into t or out of s. (Using the preservation of flow at vertices, it is not hard to show that these must be the same.) The *max-flow* of the network flow problem is the maximum value $|f|$ over all flows f. We can find this with a linear program in several ways. The obvious way is to introduce a variable for each edge which denotes the flow along that edge. The flow from 1 to 5 is at most 3 so we would introduce a variable x_{15} and constrain it as

$$0 \leq x_{15} \leq 3.$$

The constraint at a vertex such as 1 would be $x_{s1} = x_{13} + x_{15}$.

In the pictured example, it is easy to find a flow of 3 or 4 or even 5. But is there a flow of 6? If not, how can we show it? Well, we use duality. First we introduce the dual problem.

Looking at the source s in the above example, it is clear that no flow f can have value $|f|$ more than 7, looking at the sink t it is clear that $|f| \leq 6$ for any flow f. This are easy bounds, but we can generalise them. A *cut* is a partition of the vertices into sets $S \ni s$ and $T \ni t$. The *value* of a cut (S, T) is

$$|(S, T)| := \sum_{i \in S, j \in T} c_{ij}.$$

The *min-cut* is minimum value $|(S, T)|$ over all cuts (S, T).

Problem

Can you find a cut of value 5 in the above network flow problem?

ver. 2019/12/05

It is clear that no flow f can have more flow across a cut (S,T) than the value of the cut: for $i \in S$ and $j \in T$ we have $f(e_{ij}) \leq c_{ij}$ so

$$|f| \leq \sum_{i \in S, j \in T} f(e_{ij}) \leq \sum_{i \in S, j \in T} c_{ij} = |(S,T)|.$$

That is, the value of any flow is at most the value of any cut, and in particular, the max-flow is at most the min-cut. The following theorem is called the Max-flow min-cut theorem. It is a special case of duality.

Theorem 8.6. *The max-flow of a network is equal to the min-cut.*

Now, to observe that the min-cut problem is the dual problem to the max-flow problem in the sense we defined in the last section, and so use the duality theorem to prove this, we must reformulate the max-cut problem by introducing variables for each path from s to t. Rather than this, prove the theorem with a graph theoretic argument.

Proof. Clearly max-flow is at most min-cut. We show equality by finding a flow f and a cut (S,T) with the same value. Let f be a flow of maximum value, and let S be the set of vertices v for which there is a path (called an augmenting path)

$$s = u_0 \to u_1 \to \dots u_d = v$$

such that for each j

$$f(e_{u_i u_{i+1}}) < c_{u_i u_{i+1}} \text{ or } f(e_{u_{i+1} u_i}) > 0.$$

(The flow does not use all the capacity of the edge in the path, or it uses some but the edge is backwards in the path.)

Clearly $s \in S$ and $t \notin S$ or we could add more flow. Furthermore, the flow f must use all capacity of the cut $(S, V \setminus S)$; that is, for each $i \in S$ and $j \in T$,

$$f(e_{ij}) = c_{ij}.$$

If not, then we could extend the augmenting path from s to i into an augmenting path from s to j. But this contradicts our choice of S. Summing over all edges of the cut, we get that $|f| = |(S, V \setminus S)|$ as needed.

□

Problems from the text

8.4: 1, 4, 5

ver. 2019/12/05

References

[1] Gilbert Strang *Linear Algebra and its applications* Fourth Edition, International Student Edition. 2006 Thompson Learning.

ver. 2019/12/05