Numerical Methods I.

Csaba J. Hegedüs

ELTE, Faculty of Informatics
Budapest, 2015 November

A jegyzet az ELTE Informatikai Kar 2015 évi Jegyzetpályázatának támogatásával készült
Lektorálta: Dr. Abaffy József, Óbudai Egyetem

Contents

Machine numbers, errors in computation

Here we review some characteristics of computer arithmetic and investigate errors that may occur in the course of numerical computations.

1.1. Machine numbers

In the majority of cases, machine numbers are normalized binary numbers supplied with a sign so that we shall deal primarily with them here. They have the form

$$\pm.101\ldots01 \cdot 2^k = \pm m \cdot 2^k \tag{1.1}$$

sign, t binary digits, ↑exponent

The mantissa m, if nonzero, is supposed to begin with digit 1, such that $0.5 \le m < 1,\ m \ne 0$. If the base is not equal to 2, then the bases 10 and 16 (hexadecimal) may still appear in practice.

Denote by $M(t, k^-, k^+)$ the set of binary machine numbers, where t is the mantissa length, k^- is the smallest and k^+ the largest exponent.. If we want to indicate that the base may be other than two, we can apply the notation: $M(b, t, k^-, k^+)$, where the additional parameter b is supposed to denote the base of the number system. The single precision numbers in the ordinary PC's have four bytes = 32 *bits* and the following bits are allocated to the various parts of machine numbers:

1	8	23

1 bit is for sign, 8 bits are for the exponent (observe that it is also a signed number) and the mantissa has 23 bits. The precision of these numbers is about 7 decimal digits: $23\log_{10} 2 \approx 6.923$, such that we should multiply roughly the number of bits by 0.3 to get the precision in decimal. The order of magnitude is between 10^{-38} and 10^{38}. The double precision numbers have twice as much, 8 bytes = 64 bits:

1	11	52

The sign: 1 bit, exponent: 11 bits and the mantissa has: 52 bits. Now the precision is about 15 decimal digits, and the numbers may have orders of magnitude between 10^{-307} and 10^{307}. There are programming languages that allow machine numbers of quadruple precision.

One can exploit in the actual realisation of machine numbers that the first bit of the mantissa is always one – with the exception of number zero – that can be omitted. With this trick one gets an additional bit that can be exploited for improving the quality of machine arithmetic. There may also be very large numbers that can not be represented by machine numbers. We shall use the sign ∞ for such numbers. One can also find the notation NaN – „not-a-number". We may get such an answer if we try to divide by zero. Using this 'value' in subsequent arithmetic operations, we always get NaN for result, even if multiplying by zero.

1.2. Special machine numbers

The smallest positive mantissa is ½. The largest mantissa: $\overbrace{.11\ldots1}^{t\text{-times }1}=1-2^{-t}$. The smallest positive number in $M(t,k^-,k^+)$ is: $\varepsilon_0=.10\ldots0\cdot\ 2^{k^-}=1/2\cdot\ 2^{k^-}$.

The other remarkable number is ε_M, called the *precision unit machine number*. It is the distance between 1 and the neighbouring machine number less than one: $.11\ldots11\cdot\ 2^{+0}+\varepsilon_M=1$, from here $\varepsilon_M=2^{-t}$. The largest machine number is: $M_\infty=(.11\ldots1\cdot\ 2^{k^+})=(1-2^{-t})2^{k^+}$. Supplying this with the minus sign yields the smallest machine number.

Example. Let the set of machine numbers be $M(5,-4,3)$. Then the largest mantissa is $.11111=1-2^{-5}$, the smallest mantissa is always equal to ½. The first positive machine number is $\varepsilon_0=1/2\cdot2^{-4}=2^{-5}$. The precision unit is $\varepsilon_M=2^{-t}=2^{-5}$ and we have the largest number as: $M_\infty=(1-2^{-t})\cdot2^{k^+}=(1-2^{-5})2^3=8-1/4$.

1.3. Conversion to machine numbers

Now it is a natural need to give an algorithm to convert numbers into the set of machine numbers. The function that realizes the task will be denoted by fl (from *floating point number*), $\mathrm{fl}:\ \mathbb{R}\to M$. It is given by:

$$\mathrm{fl}(x)=\begin{cases}\infty, & \text{if}\quad |x|>M_\infty\\ 0, & \text{if}\quad |x|<\varepsilon_0\\ \text{closest machine number to } x, \text{ if } \varepsilon_0\le|x|\le M_\infty\end{cases}, \qquad (1.2)$$

where the closest machine number is given by rounding x to t binary digits.

For instance, convert 10.87 to 8-digit binary number. Now the integer and fractional part of the number will need different approaches. The integer part should be divided successively by 2, and the remainders are recorded. Then the binary digits are given in reverse order as we have got them. The fraction part is to be multiplied by 2 . The resulting integer ones are kept, they do not take part in the process any more. It is not needed to multiply for the last digit, it is zero if the current fraction is less than 0.5, otherwise it is equal to 1.

$$\begin{array}{r|l}10&0\\5&1\\2&0\\1&1\end{array}\to 10_2=1010 \qquad\qquad \begin{array}{r|l}.&87\\1&74\\1&48\\0&96\end{array}\to 0.87_2=.1101\ldots$$

We have got: $10.87_2=1010.1101\ldots$. But this result comes from chopping, not from rounding. For rounding, an additional digit is necessary. If the next digit is 1, then we add 1 to the last digit, otherwise we leave it unchanged. At the present case the next (ninth) digit is 1, such that the rounded number is: 1010.1110. If we want to convert 10.87 into the set $M(5,-4,3)$ of the previous example with the aid of the function fl, then $\mathrm{fl}(11.87)=\infty$, because of $M_\infty<10.87$.

Now assume that x is known exactly. Then the upper bound for the error of $\mathrm{fl}(x)$ can be given by:

$$|x-\mathrm{fl}(x)| \le \begin{cases} \infty, & \text{if} \quad |x| > M_\infty \\ \varepsilon_0, & \text{if} \quad |x| < \varepsilon_0 \\ \varepsilon_M\,|x|, & \text{if} \quad \varepsilon_0 \le |x| \le M_\infty \end{cases}, \tag{1.3}$$

where $\varepsilon_M = 2^{-t}$ is the *machine epsilon*, giving the relative error of the numbers between ε_0 and M_∞ when converted to machine numbers. The first line here is only the indication of not representable numbers. The second line is also clear, only the third line needs some explanation. It tells that the error of the represented number is not greater than the error in the t-th binary digit. We get an upper bound for the relative error by the rearrangement

$$\frac{|x-\mathrm{fl}(x)|}{|x|} \le \varepsilon_M \,. \tag{1.4}$$

To justify this statement, it is enough to consider the relative error of mantissa because the exponent is dropped by division. The error of mantissa is at most 2^{-t-1} in the case of rounding. We get the upper bound of the relative error if we divide by the possible smallest mantissa ½, that leads to the result $\varepsilon_M = 2^{-t}$.

1.3.1 The IEEE Standard for floating point arithmetic (IEEE 754)

The first version was released in 1985 and it got into general use from the 90s. The current releaase is from 2008 and it has additional features. Here the mantissa – or *significand*, a newer term for it – is normed such that $1 \le m < 2$ and the first digit – number one – is omitted to have a hidden bit for rounding. For the integer in the exponent the two's complement system is used, that is, the negative of a positive binary number is given by interchanging 0's and 1's and +1 is added. In order to distinguish between 0 and 1, the smallest negative integer in the exponent is chosen for the number 0. More detailed information can be found in

https://en.wikipedia.org/wiki/Floating_point and

https://en.wikipedia.org/wiki/IEEE_floating_point

1.4. Anomalies in computer arithmetic

Now having machine numbers at hand the next question is, what properties of the computer arithmetic will have when doing calculations with floating point numbers. We shall use decimal numbers in the following examples. We assume four decimal digits and use decimal floating point arithmetic, where the exponent can be a signed two-digit number. The set of such machine numbers will be denoted by M. For simplicity we also use the notation: $0.2543 \cdot 10^2 = 0.2543 \ \ +02$.

Then we have to realize that not everything is true in computer arithmetic that is considered natural in the field of real numbers. Here are some examples for such differences:

- There may exist nonzero $a, b \in M$, for which $a + b = a$ in computer arithmetic. That may be possible because of the big difference in the order of magnitude of numbers. For instance, add numbers 0.3460 +02 and 0.4524 –03. The final step is rounding, where all digits of the small number have to be dropped.

$$\begin{array}{l} 0.3460\ +02 \\ \underline{0.000004524\ +02} \\ 0.3460\ +02 \end{array}$$

- There may exist nonzero numbers $a,b,c \in M$, for which $(a+b)+c \neq a+(b+c)$. For instance,

$$\begin{array}{l} 0.3460\ +02 \\ \underline{0.00004524\ +02} \\ 0.3460\ +02 \end{array} \qquad \begin{array}{l} 0.3460\ +02 \\ \underline{0.00003872\ +02} \\ 0.3460\ +02 \end{array}$$

  But adding the small numbers first, we get a different result

$$\begin{array}{l} 0.3872\ \text{-}02 \\ \underline{0.4524\ \text{-}02} \\ 0.8386\ \text{-}02 \end{array} \qquad \begin{array}{l} 0.3460\ +02 \\ \underline{0.00008386\ +02} \\ 0.3461\ +02 \end{array} \cdot$$

  That would give us a hint that summing up a lot many numbers, it is better to begin with small numbers (in absolute value) first.

- There may exist nonzero $a,b,c \in M$, for which $(ab)c \neq a(bc)$. An example is

$$(0.1245\ +62 \times 0.4314\ -58) \times 0.4362\ -54 = .5371\ +03 \times 0.4362\ -54 = .2343\ \text{-}51,$$

  while choosing the second bracketing leads to zero as the product of the two last numbers is less then ε_0, the smallest positive number. If we have to multiply a lot many numbers, we have to be careful, because it may easily happen that some intermediate products get out the domain of nonzero machine numbers. If the result is too large or too small, then it is a possibility to reduce problems by computing the logarithm of it.

- Adding two numbers close in magnitude but with different signs may result in the growth of the relative error of the result. For instance

$$\begin{array}{r} 0.4693\ +02 \\ \underline{-0.4682\ +02} \\ 0.0011+02 \end{array}$$

  That is equal to 0.1100 +00. The last two digits are uncertain, only the first two digits can be considered exact. This phenomenon is called *cancellation error*. We may assume that there were further digits that can not be seen here because of the length of the mantissa. On the other hand, if the incoming machine numbers can be considered exact, then the result can be taken accurate up to machine precision. Sometimes one can apply tricks to avoid cancellations errors. As an example, we can compute $\sqrt{3472}-\sqrt{3471}$ by exploiting the fact that the numbers under square roots are integers:

$$\frac{(\sqrt{3472}-\sqrt{3471})(\sqrt{3472}+\sqrt{3471})}{\sqrt{3472}+\sqrt{3471}} = \frac{1}{\sqrt{3472}+\sqrt{3471}}.$$

- The safe computation of the roots of second order polynomials can be done as follows:

$$\text{Roots of } x^2-2px+q=0\text{:}\quad x_1 = p+\operatorname{sign}(p)\sqrt{p^2-q}, \quad x_2 = q/x_1.$$

- There may happen cases when an intermediate result gets larger than M_∞ (arithmetic overflow) although the result is in the domain of nonzero machine numbers. As an example, let us have $a = 0.3265 + 60$, $b = 0.5671\ +02$ and compute $\sqrt{a^2 + b^2}$. When squaring, the first number has exponent 120, that results in overflow. But if we compute $s\sqrt{(a/s)^2 + (b/s)^2}$, where $s = \max(|a|,|b|)$, then that may not happen.
- Another example is the computation of the binomial coefficient $\binom{n}{k}$. For not too large n the computation of $n!$ may get larger than the largest machine number, however, the result can be represented as a machine number. A safer way of computation is

  $$\frac{n}{1}\frac{n-1}{2}\frac{n-2}{3}\cdots\frac{n-k+1}{k}.$$

  When collecting the product, add a next fraction by multiplying with the numerator first and after that apply division.
- One may find that a function does not return a value with the same relative precision than that of the input data. For instance, consider the sine function. If the argument is small, then there is no problem. But if x has a large value, say $x = 2356$, then for computing $\sin(2356)$ we have to take $2356 \bmod (2\pi)$. But then the remainder will have only one precise digit using the above arithmetic such that we may not expect better accuracy for the output value.

The shown examples tell us that the undesirable phenomena of computer arithmetic happen primarily in cases when the numbers have very different orders of magnitude or in the case when difference of very close numbers is computed.

1.5. Estimating the number of lost digits in cancellation

Cancellation happens if two numbers are nearly the same and they are subtracted from each other. For example, assume a 6-digit decimal arithmetic and compute: $126.426 - 126.411 = 0.015$. It is seen, the first four digits are lost, and the result, if normalized, has the form: $0.150000 \cdot 10^{-1}$. Now the question is, how we can interpret the accuracy of the result. If there were 10 digits and the further 4 digits – which are not seen here – are the same, then the result is accurate to 6 decimals. If the missing four digits were not the same, then we have accuracy only for two figures. As seen, the number of accurate digits may range now from 2 to 6. We shall adopt the pessimistic picture now such that there is no more accurate digits than 2.

Let the scalars α, β be nonzero and nearly the same. When subtracting, the cancellation can be characterized by the ratio

$$\eta = \frac{|\alpha - \beta|}{\max(|\alpha|,|\beta|)}. \tag{1.5}$$

If $\eta > 0.5$ we may say that there is no cancellation of binary digits, while in the case of $\eta < 10^{-\rho}$, where ρ is the number of accurate decimal digits – we say that the two numbers are the same to computational accuracy. Although 15 decimal digits are assumed in double precision computation, we should take into account that usually the last 2-3 digits are uncertain due to rounding errors. Therefore a practical choice for ρ is ρ=12. Then we may loose digits by cancellation if the condition

$$\eta_{\min} = 10^{-\rho} \le \eta < \eta_{\max} \tag{1.6}$$

holds, where $\eta_{\max} = 1/2$ may be chosen. The worst case is always assumed, therefore the number of lost decimals is estimated by $-\log_{10}\eta$. This value is 4.06... in the above example.

As a consequence, the number of accurate digits is

$$\gamma = \rho + \log_{10}\eta \tag{1.7}$$

after cancellation and the error of the difference $|\alpha - \beta|$ is $10^{-\gamma}|\alpha - \beta|$. Similarly, the error of η can be given by $10^{-\gamma}\eta$.

1.6. Errors

One is also interested in quality computations about the accuracy of the result. For that we have to consider the posssible sources of errors. We may have errors already in the initial data: They may be called *data errors* or *inherited errors*. When doing computations, we may also commit errors that should be discovered and corrected by ourselves through surveying our activity carefully. We may also use a mathematical formula that has error because of being an approximation. Such errors belong to the applied method and usually they are assumed to be larger than the error of the numerical computation. *Rounding errors* may show up also in the data preparation phase but characteristically we always have to face the presence of rounding errors in the course of computation. When analyzing errors, we have to recognize, which kind of errors are essential from the point of view of our problem. There are many cases when data errors or formula errors give the main contribution to the error of computation. Many times we have to accept data errors, but formula errors may be decreased by using a more precise approximation.

According to the basic model of error computation we consider all computed values subject to errors. We are primarily interested in the bound of errors.

Notations. The *accurate value* of x will be denoted by x^*, and its error is given by: $\Delta x = x - x^*$, where Δx is a signed number. The *relative error* is defined by $\delta x = \Delta x / x \approx \Delta x / x^*$. It should be remarked at this point that other authors define the relative error by dividing with the accurate value x^*, as in the second formula here. Our choice admits the fact that we do not know the accurate value. The *error bound* Δ_x is a nonnegative number that gives an upper bound of the absolute value of the error: $|\Delta x| \le \Delta_x$. Similarly, δ_x is the *relative error bound*, for which $|\delta x| \le \delta_x$ holds.

Remark. The difference between the two ways of defining relative errors has a magnitude of second order: $\Delta x / x^* - \delta x = \delta x / (1 - \delta x) - \delta x = (\delta x)^2 / (1 - \delta x)$.

In reality Δx is not known, only the upper bound to that. What we know at start is that x^* is in a neighborhood of x with radius Δ_x.

1.6.1 Error propagation

It is important to know the error propagation formulas of the basic arithmetic operations. $+,-,*,/$. Below the formulas on the left refer to the errors and in the same line to the right, one can find the error bound relation:

$$\begin{aligned} &\Delta(x\pm y)=\Delta x\pm\Delta y, && \Delta_{x\pm y}=\Delta_x+\Delta_y,\\ &\Delta(xy)=x\Delta y+y\Delta x, && \Delta_{xy}=|x|\Delta_y+|y|\Delta_x, \\ &\Delta(x/y)=\frac{y\Delta x-x\Delta y}{y^2}, && \Delta_{x/y}=\frac{|y|\Delta_x+|x|\Delta_y}{\left|y^2\right|}. \end{aligned} \tag{1.8}$$

The error formulas on the left are usually derived in mathematical analysis for getting the differentiation rules for the sum, product and ratio of functions. It is also seen from here that the formulas can be considered right only if the errors are really small such that the second order terms can be neglected. The formulas to the right are consequences of the left ones, the same arrangement can be seen for the relative errors below:

$$\begin{aligned} &\delta(x\pm y)=\frac{x\delta x\pm y\delta y}{x\pm y}, && \delta_{x\pm y}=\frac{|x|\delta_x+|y|\delta_y}{|x\pm y|},\\ &\delta(xy)=\delta y+\delta x, && \delta_{xy}=\delta_y+\delta_x, \\ &\delta(x/y)=\delta x-\delta y, && \delta_{x/y}=\delta_x+\delta_y. \end{aligned} \tag{1.9}$$

1.6.2 Error of functions

Let $f:\mathbb{R}\to\mathbb{R}$ be a continuously differentiable function. Then according to the mean value theorem of Lagrange, there exist $\xi\in[x,x^*]$, for which

$$f(x)=f(x^*)+f'(\xi)\Delta x,\ \ \Delta x=x-x^*$$

holds. From here the *error of the function value* is expressible as

$$f(x)-f(x^*)=\Delta f=f'(\xi)\Delta x\,. \tag{1.10}$$

Let $\max\limits_{x\in[x-\Delta_x,x+\Delta_x]}|f'(x)|=M_1$, then we get the error bound

$$\Delta_f\le M_1\Delta_x\,, \tag{1.11}$$

where the estimate is restricted to a neighborhood of x^* with radius Δ_x.

This relation suggests a definition to the *stability of an algorithm*: We attribute the mapping of the algorithm to a function $f(x)$ and say the algorithm stable if for two input values x_1, x_2 we have the relation

$$|f(x_2)-f(x_1)|\le C|x_1-x_2|,\ \ x_1,x_2\in X\,, \tag{1.12}$$

where C is a not very large constant. The reason is that for such an algorithm we can keep the error in our control. We are then able to demand a necessary precision for the input values in order to get an output value of specified accuracy. Observe that C can not be arbitrarily large because $|f(x_2)-f(x_1)|/C\le\Delta_f/C$ may not be smaller than the largest gap between two machine numbers in the region, where x_1, x_2 can be found.

It is still important to define the concept of *inverse stability*. A mapping – or algorithm – is inverse stable if the result of computation can be obtained exactly from a slightly perturbed input value.

1.6.3 The condition number

For the relative error of the function we get the relation

$$\delta f = \frac{\Delta f}{f(x)} \approx \frac{xf'(x)}{f(x)}\frac{\Delta x}{x} = \frac{xf'(x)}{f(x)}\delta x.$$

Taking the absolute value:

$$\left|\delta f\right| \approx c(f,x)\left|\delta x\right|, \tag{1.13}$$

where number $c(f,x) = \left|xf'(x)/f(x)\right|$ is called the *condition number* of f at point x. If this number is large then the function is called *instabile* or *ill-conditioned*, because a small relative change in the value of x leads to a large relative change in the value of the function. If the condition number is too big then small rounding errors in the domain of x may lead to unacceptably large errors of the output value.

1.7. The postulate of error analysis

Denote by $\circ$ any of the four arithmetic operations. Then the basic assumption in error analysis is the following:

$$fl(a \circ b) = a \circ b(1+\varepsilon), \quad \left|\varepsilon\right| \le \varepsilon_M, \quad a,b \in M(t,k^-,k^+)\,. \tag{1.14}$$

Seemingly ε is the relative error. If we check the relative error bounds in (1.9), then we find that $0 \le \left|\varepsilon\right| \le 2\varepsilon_M$ holds in the case of multiplication and division. For numbers in the IEEE Standard the smallest mantissa is 1, therefore $\varepsilon_M = 2^{-t}$ can be considered as a doubled value of the actual precision bound such that the assumption (1.14) is good. But for the case of +, – we may get very far values from the assumption if a catastrophic cancellation error occurs, i.e. the number in the denominator is very small. However, if we restrict ourselves to machine numbers, and consider the machine numbers accurate then the assumption remains true if we still have saved an additional bit for rounding, see Sect. 1.5. But considering machine numbers accurate may be controversial with respect to real life situations. Therefore the programmer is advised to avoid cancellation errors in the program as far as possible. In case of serious cancellation problems one has to keep count of it and one should investigate its effects on the final result.

1.8. Problems

1.1. Let the set of machine numbers be $M(5,-4,4)$. Identify the special machine numbers! Map the following numbers: 1/50, 0.37, 3.67, 7.2, 21.78 into this set!

1.2. How should we convert 10.87 into a ternary number of base 3?

1.3. How the machine epsilon is modified, if chopping is applied instead of rounding?

Norms, inequalitites

We introduce distance functions for vectors and matrices in this section. At first some notational conventions will be given.

Matrices are denoted by capital letters: $A, B, C, \ldots$ vectors with lower case letters: $a, b, c, \ldots$, with the exception of i, j, k, l, m, n : they will be used for indices as a rule. We use greek lower case letters for scalars. If matrix A is built of column vectors $a_1, a_2, \ldots$, then it is given by $A = [a_1 a_2 \ldots a_n]$. Another form of giving matrices is $A = [a_{ij}]$, where the ij-th element is given generally. The unit matrix of order n is $I_n = [e_1 \, e_2 \ldots e_n]$, that has columns $e_1, e_2, \ldots, e_n$, the Cartesian unit vectors. The transpose of a vector is indicated by: a^T , in the complex case the conjugate transpose is a^H , where complex conjugation is also done. The notation for the transpose or conjugate transpose of matrices is given similarly. Usually we consider real matrices.

1.9. Metric space

Let $\mathcal{X}$ be a set, and introduce the function $\delta : (\mathcal{X} \times \mathcal{X}) \to \mathbb{R}$ between two elements of the set. In order that it be a distance function, it is natural to demand the following assumptions for $a, b \in \mathcal{X}$:

i) $\delta(a,b) = \delta(b,a)$, that is, the distance of a from b should be the same as the distance of b from a (symmetry).

ii) $\delta(a,b) = 0 \iff a = b$, the distance is zero only if the two elements are identical.

iii) $\delta(a,c) \le \delta(a,b) + \delta(b,c)$, the triangle inequality. It reflects the fact that the shortest path between two elements is along the connecting line.

We call the pair $(\delta, \mathcal{X})$ a *metric space* if the above three assumptions are fulfilled. For set $\mathcal{X}$ we shall consider here the sets $\mathbb{R}^n$ and $\mathbb{R}^{m \times n}$ and we shall find distance functions for vectors and matrices.

T2.1 Theorem on non-negativeness of a metric

$$0 \le \delta(a,b) \tag{2.1}$$

Proof. $0 = \delta(a,a) \le \delta(a,b) + \delta(b,a) = 2\delta(a,b)$ applying properties *i)*, *iii)* and finally *ii)*. ■

1.10. Vector norms, the power norm

The vector norm $\|x\|$: $\mathbb{R}^n \to \mathbb{R}$ is a scalar valued function and it has the following properties:

$$\begin{aligned} &i) && \|x\| = 0 \iff x = 0, \\ &ii) && \|\lambda x\| = |\lambda| \|x\|, \\ &iii) && \|x + y\| \le \|x\| + \|y\|. \end{aligned} \tag{2.2}$$

Then the choice $\delta(x,y)=\|x-y\|$ gives a metric, because the necessary conditions are fulfilled. The first two conditions are trivially satisfied by the power norm:

$$\|x\|_p=\left(\sum_{i=1}^{n}|x_i|^p\right)^{1/p}, \quad 1\le p\le\infty. \tag{2.3}$$

Later on we shall see that the third condition is also fulfilled.

1.11. Hölder's inequality

For power norms the Hölder inequality is

$$|y^Tx|\le\sum_{i=1}^{n}|x_i||y_i|\le\|x\|_p\,\|y\|_q, \qquad \frac{1}{p}+\frac{1}{q}=1, \tag{2.4}$$

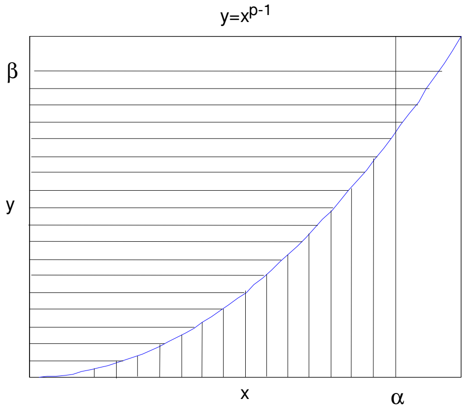

Figure 1.

that reduces to the well-known Cauchy-Bunyakovsky inequality for $p=q=2$. The connection between p and q can be rearranged into the form $p-1=1/(q-1)$, that should be kept in mind when deriving the next inequality. The applied function is $y=x^{p-1}$ and the first integral is the area shaded vertically in the shown figure, while the second one belongs to the area shaded horizontally:

$$\alpha\beta\le\int_0^{\alpha}x^{p-1}dx+\int_0^{\beta}y^{q-1}dy=\frac{\alpha^p}{p}+\frac{\beta^q}{q}.$$

Then apply the substitutions

$$\alpha_i=\frac{|x_i|}{\|x\|_p}, \qquad \beta_i=\frac{|y_i|}{\|y\|_q}$$

and summing for i results in the right inequality of (2.4).

The third relation of (2.2), the triangle inequality can be shown by using the relation $p/q=p-1$ in

$$\|x+y\|_p^p=\sum_{i=1}^{n}|x_i+y_i|^p\le\sum_{i=1}^{n}\{|x_i|+|y_i|\}|x_i+y_i|^{p-1}$$

such that the Hölder inequality is applied for both terms on the right side. Now the first term yields the result:

$$\sum_{i=1}^{n}|x_i||x_i+y_i|^{p-1}\le\|x\|_p\left\{\sum_{i=1}^{n}|x_i+y_i|^{(p-1)q}\right\}^{1/q}=\|x\|_p\|x+y\|_p^{p/q}.$$

The other term yields similar result and ordering the two leads to the desired inequality, which is called the *Minkowski inequality* for a general power c$1\le p$.

1.12. Some properties of power norms

We have the inequality:

$$\|x\|_{p+s}\le\|x\|_p,\quad 1\le p,\quad 0\le s. \tag{2.5}$$

It can be rearranged into the form

$$\sum_{i=1}^{n}\left|\frac{x_i}{x_k}\right|^{p+s}\le\left\{\sum_{i=1}^{n}\left|\frac{x_i}{x_k}\right|^{p}\right\}\left\{\sum_{i=1}^{n}\left|\frac{x_i}{x_k}\right|^{p}\right\}^{s/p},\quad x_k\ne 0.$$

Choose $|x_k|=\max_i|x_i|$, then the first factor on the right is greater term by term as compared to the sum on the left, moreover, the second factor is surely not less than 1.

We list the frequently used power norms. The first one is:

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

It is called the 1-norm, octahedron norm or sometimes Manhattan norm. The 3-dimensional unit sphere - vectors x for which .$\|x\|_1=1$. - will be the octahedron that has vertices $\{(\pm1,0,0),(0,\pm1,0),(0,0,\pm1)\}$. The next norm

$$\|x\|_2=\left\{\sum_{i=1}^{n}|x_i|^2\right\}^{1/2}$$

is the 2-norm, Euclidean norm or sphere norm of vector x. The unit sphere in this norm is the unit ball. One gets the third important norm in the limiting case $p\to\infty$

$$\|x\|_\infty=\max_j|x_j|\cdot\lim_{p\to\infty}\left\{\sum_{i=1}^{n}\left|\frac{x_i}{\max_j|x_j|}\right|^{p}\right\}^{1/p}=\max_j|x_j|$$

that is called the maximum norm, ∞-norm or Chebyshev-norm. It is seen from (2.5) that we have here the largest and smallest power norms moreover, the 2-norm which is invariant under orthogonal transformations, see Exercise 2.6. One can derive the following inequalities for these norms:

$$\begin{aligned}&\|x\|_\infty\le\|x\|_1\le n\|x\|_\infty,\\&\|x\|_\infty\le\|x\|_2\le\sqrt{n}\|x\|_\infty,\\&\frac{1}{\sqrt{n}}\|x\|_1\le\|x\|_2\le\|x\|_1.\end{aligned} \tag{2.6}$$

1.13. Convergence in norm. The norm equivalence theorem

The norms can be used to define the convergence of vector sequences. By writing $x^{(k)} \to x$ we mean that $\exists x \in \mathbb{R}^n,\ \lim\limits_{k\to\infty} \left\| x^{(k)} - x \right\| = 0$.

Two norms $\|x\|_{(1)}$ and $\|x\|_{(2)}$ are called *equivalent* if there exist numbers $c_1, c_2 > 0$ such that the inequalities

$$c_1 \|x\|_{(1)} \le \|x\|_{(2)} \le c_2 \|x\|_{(1)}$$

hold. From here one can easily get the other form

$$\frac{\|x\|_{(2)}}{c_2} \le \|x\|_{(1)} \le \frac{\|x\|_{(2)}}{c_1}$$

indicating that norm equivalence is a symmetric relation.

T2.2 The norm equivalence theorem

Any two norms are equivalent in finite dimensional spaces. In other words, the norms may not differ from each other arbitrarily. As a consequence, it will be all the same, which norm we are using when investigating convergence. (Proof omitted.)

1.14. Matrix norms

The matrix norm $\|A\|:\ \mathbb{R}^{m\times n} \to \mathbb{R}$ has the following properties:

$$\begin{array}{ll} i) & \|A\| = 0 \iff A = 0, \\ ii) & \|\lambda A\| = |\lambda| \|A\|, \\ iii) & \|A+B\| \le \|A\| + \|B\|, \\ iv) & \|AB\| \le \|A\| \|B\|. \end{array} \tag{2.7}$$

For the last two properties we need that the two matrices have appropriate sizes such that they can be added and multiplied. As vectors can be considered special matrices, all matrix norms give a vector norm, which we call c*ompatible* with that matrix norm. But this approach can be followed also in the reverse direction, as all vector norms *induce a matrix norm* by the relation:

$$\|A\| = \sup_{x\ne 0} \frac{\|Ax\|}{\|x\|} = \sup_{\|x\|=1} \|Ax\|, \tag{2.8}$$

where $\|.\|$ denotes a vector norm. We remark at this point that in a more general definition of induced matrix norms, it is allowable to use different vector norms for Ax and x. For induced norms a dircct consequence of the definition of (2.8) is the inequality

$$\|Ax\| \le \|A\| \|x\|. \tag{2.9}$$

We call the matrix norm $\|A\|$ and vector norm $\|x\|$ *consistent,* if for any x (2.9) is satisfied. This definition has importance in the case when the matrix norm is not an induced norm from a vector norm.

T2.3 Theorem on induced norms

The induced matrix norm satisfies the conditions of (2.7).

Proof. Ad 1. $A=0\to\|A\|=0.\quad \|A\|=0\to Ax=0\ \ \forall x-\text{re}\to A=0.$

Ad 2. $\|\lambda A\|=\sup\limits_{\|x\|=1}\|\lambda Ax\|=|\lambda|\sup\limits_{\|x\|=1}\|Ax\|=|\lambda|\|A\|.$

Ad 3. $\|A+B\|=\sup\limits_{\|x\|=1}\|(A+B)x\|\le\sup\limits_{\|x\|=1}\{\|Ax\|+\|Bx\|\}\le\|A\|+\|B\|.$

Ad 4. $\exists x_0\in\mathbb{R}^n,\ \|x_0\|=1:\quad \|AB\|=\|ABx_0\|\le\|A\|\|Bx_0\|\le\|A\|\|B\|.$ ■

1.15. Determination of some matrix norms

The ***column norm,*** $p=1$*:*

$$\|A\|_1=\max_{(j)}\left\|Ae_j\right\|_1=\max_{(j)}\sum\nolimits_{i=1}^{m}|a_{ij}|. \tag{2.10}$$

Choose $\|x\|_1=\sum\nolimits_{j=1}^{n}|x_j|=1$, then using norm properties, we find

$$\|Ax\|_1=\left\|\sum\nolimits_{j=1}^{m}{}_j Ae_je_j^Tx\right\|_1\le\sum\nolimits_{j=1}^{n}\left\|Ae_j\right\|_1|x_j|\le\|Ae_k\|_1\sum\nolimits_{j=1}^{n}|x_j|=\|Ae_k\|_1,$$

where $\|Ae_k\|_1=\max\limits_{(j)}\left\|Ae_j\right\|_1$. This upper bound is reached for $x=e_k$, hence the maximum is found and the result is the induced 1-norm.

The ***row norm,*** $p=\infty$:

$$\|A\|_\infty=\max_{(i)}\left\|e_i^TA\right\|_\infty=\max_{(i)}\left\|A^Te_i\right\|_1=\max_{(i)}\sum\nolimits_{j=1}^{n}|a_{ij}|. \tag{2.11}$$

Choose $\|x\|_\infty=1$, then

$$\|Ax\|_\infty=\max_{(i)}\left|\sum\nolimits_{j=1}^{n}a_{ij}x_j\right|\le\max_{(i)}\sum\nolimits_{j=1}^{n}|a_{ij}||x_j|\le\max_{(i)}\sum\nolimits_{j=1}^{n}|a_{ij}|.$$

Assume the maximum is found for the k-th row. Then for the elements of vector $\|x\|_\infty=1$, we can take

$$x_j=\begin{cases}\{\bar{a}_{kj}/|a_{kj}|, & \text{if } a_{kj}\ne 0\\ \quad 0 & \text{, otherwise}\end{cases},$$

giving the value of the just found upper bound for $\|Ax\|_\infty$. Here $\bar{a}_{kj}$ denotes the complex conjugate of a_{kj} such that our result is also good for complex matrices.

The ***spectral norm,*** $p=2$*:*

$$\|A\|_2=\max_{(k)}\left(\lambda_k(A^TA)\right)^{1/2}. \tag{2.12}$$

The following maximum is sought:

$$\|A\|_2^2 = \max \frac{\|Ax\|_2^2}{\|x\|_2^2} = \max \frac{x^T A^T Ax}{x^T x}.$$

The quotient here is called the *Rayleigh-quotient* with respect to matrix $A^T A$. If u_k is an eigenvector of $A^T A$ with eigenvalue λ_k, then applying the choice $x = u_k$ the Rayleigh-quotient takes the value of λ_k. From here it is clear that the largest value of the Rayleigh-quotient is at least $\lambda_{\max} = \max_k \lambda_k$. We show it may not be larger. It is known from linear algebra that the eigenvectors of a symmetric matrix form a full orthonormal set such that any vector x can be expanded in the form $x = \sum_{j=1}^n \alpha_j u_j$. If this is substituted into the Rayleigh-quotient, we get:

$$\lambda_{\max} - \frac{x^T A^T Ax}{x^T x} = \lambda_{\max} - \frac{\sum_{j=1}^n \lambda_j \alpha_j^2}{\sum_{j=1}^n \alpha_j^2} = \frac{\sum_{j=1}^n (\lambda_{\max} - \lambda_j)\alpha_j^2}{\sum_{j=1}^n \alpha_j^2} \geq 0,$$

showing that the maximum is found. Observe that all eigenvalues of $A^T A$ is non-negative because of $\lambda_j(A^T A) = \|Au_j\|_2 / \|u_j\|_2$.

Remark. For complex A we take A^H, the conjugate transpose of A instead of A^T and we can proceed similarly. The nonzero eigenvalues of $A^H A$ and AA^H are the same. This can be shown for the matrix product AB in general, for let u be an eigenvector of AB then

$$ABu = \lambda u \rightarrow BA(Bu) = \lambda(Bu), \quad A, B^T \in \mathbb{R}^{m,n}$$

follows if multiplied by matrix B from the left.

The ***Frobenius norm*** is defined by

$$\|A\|_F = \left(\sum_{i,j} |a_{ij}|^2 \right)^{1/2} = \sqrt{\operatorname{tr}(A^H A)} = \sqrt{\operatorname{tr}(AA^H)}, \tag{2.13}$$

where $\operatorname{trace}(A) = \operatorname{tr}(A) = \sum_{i=1}^n a_{ii}$. To show that it is also a matrix norm, introduce the vec function that arranges the columns of matrix $A = [a_1 \; a_2 \; \dots \; a_n]$ into one column:

$$\operatorname{vec}(A) = \left[a_1^T \; a_2^T \; \dots \; a_n^T \right]^T.$$

Then $\|A\|_F = \|\operatorname{vec}(A)\|_2$ holds and the first three norm conditions are automatically fulfilled. To show the fourth norm condition, consider the Cauchy-Bunyakovsky inequality for an element of the matrix product AB in squared form: $|e_i^T ABe_j|^2 \leq \|e_i^T A\|_2^2 \|Be_j\|_2^2$. Summing up for i and j gives the squared fourth condition.

1.16. A relation between the spectral radius and matrix norms

We define the *spectral radius* of matrix A by the following number:

$$\rho(A) = \max_k |\lambda_k(A)|, \tag{2.14}$$

where $\lambda_k(A)$ is an eigenvalue of A.

T2.4 Theorem on spectral radius

The inequality

$$\rho(A) \le \|A\| \tag{2.15}$$

holds, where $\|A\|$ is an arbitrary matrix norm.

Proof. Let u_k be an eigenvector, then $Au_k = \lambda_k u_k$ holds. Multiply both sides by u^T from the right then there are matrices on both sides. Applying the matrix norm properties, we get

$$|\lambda_k| \|u_k u_k^T\| = \|Au_k u_k^T\| \le \|A\| \|u_k u_k^T\|, \text{ for } \forall k .$$

Dividing by $\|u_k u_k^T\|$ on both sides yields the result as the inequality is also true for the absolute largest eigenvalue. ■

1.17. Perturbing the solutions of linear systems

We shall consider two cases. First, the right vector b of the linear system is perturbed by a small vector δb, and in the other case, we consider the perturbation of the coefficient matrix.

Now assume δb is added to vector b such that the system $Ax = b$ is changed to $A(x+\delta x) = b + \delta b$ from which $A\delta x = \delta b$ follows and for consistent vector and matrix norms we get the lower and upper bound estimate:

$$\frac{1}{\|A\| \|A^{-1}\|} \frac{\|\delta b\|}{\|b\|} \le \frac{\|\delta x\|}{\|x\|} \le \|A\| \|A^{-1}\| \frac{\|\delta b\|}{\|b\|}. \tag{2.16}$$

For derivation we start from the equations and take norms as shown:

$$\begin{array}{cc} b = Ax, & \delta x = A^{-1}\delta b, \\ \downarrow & \downarrow \\ \|b\| \le \|A\| \|x\|, & \|\delta x\| \le \|A^{-1}\| \|\delta b\|. \end{array}$$

Multiplying the same sides of the inequalities and arranging terms yields the right side of (2.16). For the left side we start from

$$\begin{array}{cc} x = A^{-1}b, & \delta b = A\delta x, \\ \downarrow & \downarrow \\ \|x\| \le \|A^{-1}\| \|b\|, & \|\delta b\| \le \|A\| \|\delta x\| \end{array}$$

and proceed in a similar way.

L2.1 Lemma.

If $\|B\| < 1$ holds then $I + B$ is invertible and one has the inequality for induced norms:

$$\left\|(I+B)^{-1}\right\| \le \frac{1}{1-\|B\|}. \tag{2.17}$$

Proof. Applying Theorem T2.4, we see that all absolute eigenvalues of B are less than 1, consequently no eigenvalues of $I+B$ may be zero such that $I+B$ is invertible. By a simple rearrangement,

$$(I+B)^{-1}=(I+B-B)(I+B)^{-1}=I-B(I+B)^{-1}.$$

Taking norm of both sides and applying the triangle inequality on the right gives $\left\|(I+B)^{-1}\right\|\le 1+\|B\|\left\|(I+B)^{-1}\right\|$ from where the statement follows. ■

Now we turn to the second case, where the coefficient matrix is perturbed by a small δA: $(A+\delta A)(x+\delta x)=b\ \rightarrow (A+\delta A)\delta x=-\delta Ax\ \rightarrow\ \delta x=-(I+A^{-1}\delta A)^{-1}A^{-1}\delta Ax$ and one gets the estimate:

$$0\le\frac{\|\delta x\|}{\|x\|}\le\left\|\left(I+A^{-1}\delta A\right)^{-1}\right\|\left\|A^{-1}\right\|\|A\|\frac{\|\delta A\|}{\|A\|}\le\|A\|\left\|A^{-1}\right\|\frac{\|\delta A\|}{\|A\|}\ \frac{1}{1-\left\|A^{-1}\delta A\right\|}. \tag{2.18}$$

Observe that Lemma L2.1 was used in the last step.

1.18. The matrix condition number

The previous estimates show that the relative change of the solution is proportional with the number $\mathrm{cond}(A)=\|A\|\left\|A^{-1}\right\|$. This is called the *condition number* of the matrix that is frequently denoted also by $\kappa(A)$. If this number is large, then we call the linear system $Ax=b$ *ill-conditioned.*

1.19. The relative residual

The number $\|\delta x\|/\|x\|$ does not characterize well the stability of the solution method because it may be large – independently of the method of solution – if $\mathrm{cond}(A)$ is large. For that purpose the residual vector is more appropriate. Assume, we have the approximate solution $\tilde{x}$, then the *residual vector* is defined by: $r=b-A\tilde{x}$. The *relative residual* is given by:

$$\eta=\frac{\|r\|}{\|A\|\|\tilde{x}\|}. \tag{2.19}$$

According to inverse stability, the method of solution is backward stable if the result is the accurate result of a slightly perturbed initial problem: $(A+\delta A)\tilde{x}=b$, where $\|\delta A\|/\|A\|$ is small.

T2.5 Theorem on the relative residual

If η is large then $\|\delta A\|/\|A\|$ is also large. But if η is small then the relative change of the matrix is also small in 2-norm at least.

Proof. From the relation $0=b-(A+\delta A)\tilde{x}=r-\delta A\tilde{x}$ we get the estimate $\|r\|\le\|\delta A\|\|\tilde{x}\|$. Substitute this into η of (2.19):

$$\eta=\frac{\|r\|}{\|A\|\|\tilde{x}\|}\le\frac{\|\delta A\|}{\|A\|},$$

that is, if η is large, then the relative change of the coefficient matrix may be even larger.

On the other hand, if η is small then we use an explicit form of δA:

$$\delta A = \frac{r\tilde{x}^T}{\tilde{x}^T\tilde{x}}, \text{ because } b - \left(A + \frac{r\tilde{x}^T}{\tilde{x}^T\tilde{x}}\right)\tilde{x} = b - A\tilde{x} - r = 0. \tag{2.20}$$

Choosing 2-norm, $\left\|r\tilde{x}^T\right\|_2 = \|r\|_2 \left\|\tilde{x}^T\right\|_2$ (see exercise 2.5), and substitution gives $\dfrac{\|\delta A\|_2}{\|A\|_2} = \dfrac{\|r\|_2}{\|A\|_2\|\tilde{x}\|_2}$. ■

Remark. If η is large then the relative error of the solution with respect to $\tilde{x}$ is also large, because $\|r\| = \|A(x-\tilde{x}\|) \le \|A\|\|x-\tilde{x}\| = \|A\|\|\delta x\|$ and substituting into (2.19) gives

$$\eta \le \frac{\|r\|}{\|A\|\|\tilde{x}\|} \le \frac{\|A\|\|\delta x\|}{\|A\|\|\tilde{x}\|} = \frac{\|\delta x\|}{\|\tilde{x}\|}. \tag{2.21}$$

1.20. Distance of the nearest singular matrix

Assume matrix A is invertible and B is a nearby matrix and it is singular. Then there exists a nonzero vector x such that $Bx = 0$ moreover, we can write

$$\|Ax\| = \|(A-B)x\| \le \|A-B\|\|x\|$$

and

$$\|x\| = \left\|A^{-1}Ax\right\| \le \left\|A^{-1}\right\|\|Ax\|.$$

Multiply the same sides of these inequalities, then after simplification one gets the inequality

$$\frac{1}{\|A-B\|} \le \left\|A^{-1}\right\|. \tag{2.22}$$

It shows that the reciprocal of the distance to the set of singular matrices gives a lower bound for the norm of the inverse. We may find various modifications of A to get such estimates.

For instance, choose $B = A - e_i e_i^T A$, then $A-B$ will have only one nonzero row and taking the 1-norm, gives

$$\frac{1}{\max\limits_j |a_{ij}|} \le \left\|A^{-1}\right\|_1,$$

but we may take the row, for which the left denominator is minimal:

$$\frac{1}{\min\limits_i \max\limits_j |a_{ij}|} \le \left\|A^{-1}\right\|_1. \tag{2.23}$$

Another relation can be obtained by choosing $B = A - Ae_i e_i^T$ and applying the infinity norm, we get:

$$\frac{1}{\min\limits_{j}\max\limits_{i}\left|a_{ij}\right|} \le \left\|A^{-1}\right\|_{\infty}. \tag{2.24}$$

Similar approach can be applied when $PA = LU$, where P is a permutation matrix, L is a unit lower triangular matrix (it has 1's in the diagonal) and U is upper triangular. Then the permutation does not change the 1,2 and ∞ norms of A, (see Exercise 2.17). Choose $B = LU - Le_n e_n^T U$ to find

$$\frac{1}{\left|u_{nn}\right|} \le \left\|A^{-1}P^T\right\|_p = \left\|A^{-1}\right\|_p, \quad p = 1,2,\infty. \tag{2.25}$$

Such estimates help to get lower bounds for the condition number.

1.21. Problems

2.1. Show that for all induced norm $\|I\| = 1$ holds. May the Frobenius norm be an induced norm?

2.2. If A is invertible then $\|x\|_A = \|Ax\|$ is also a vector norm.

2.3. A matrix condition number may not be less then 1 for induced norms.

2.4. Using the 2-norm, the condition number of orthogonal or unitary matrices is equal to 1.

2.5. $\left\|ab^T\right\|_2 = \|a\|_2 \|b\|_2$. $\left\|ab^T\right\|_1 = \|a\|_1 \|b\|_\infty$. $\left\|ab^T\right\|_\infty = \|a\|_\infty \|b\|_1$.

2.6. $U^T U = I$ (orthogonal) $\to \|AU\|_2 = \|A\|_2$.

2.7. $\left|\|A\| - \|B\|\right| \le \|A \pm B\|$.

2.8. $A = \begin{bmatrix} 2 & -3 & 1 \\ -4 & -2 & 1 \end{bmatrix}$, $\|A\|_1 = ?$ $\|A\|_\infty = ?$ $\|A\|_2 = ?$

2.9. $\|A\|_2 \le \sqrt{\|A\|_1 \|A\|_\infty}$.

2.10. Check the inequality $\|Ax\|_2 \le \|A\|_F \|x\|_2$. (It is consistent with the 2-norm.)

2.11. If $A = A^T$ then $\|A\|_2 = \rho(A)$ = spectral radius, that is, the spectral norm is the minimal norm for symmetric matrices. ($\|\,.\,\|_2$ = spectral norm.)

2.12. If $A = A^T$ then $\|A\|_2 \le \|A\|_p$, $p = 1,\infty$.

2.13. $U^T U = I$ (orthogonal) $\to \|AU\|_F = \|A\|_F$.

2.14. $\|AB\|_2 = \|BA\|_2$ if $A = A^T$ and $B = B^T$.

2.15. $\operatorname{cond}_2(A^T A) = \operatorname{cond}_2^2(A)$.

2.16. $\|PA\|_p = \|A\|_p = \|AP\|_p$, $p = 1,2,\infty$, where P is a permutation matrix.

Elementary transforms in numerical linear algebra

Here we recall some elementary facts in linear algebra and review some elementary transforms that will be needed later.

1.22. Multiplication of matrices

Multiplying matrices $A=[a_{ij}]\in\mathbb{R}^{m\times n}$ and $B=[b_{jk}]\in\mathbb{R}^{n\times l}$ results in the matrix $C=AB=\left[c_{ik}\right] =\left[\sum_{j=1}^{n}a_{ij}b_{jk}\right]\in\mathbb{R}^{m\times l}$. Vectors can be considered special matrices consisting of one row or column, their multiplication can be done according to the rules of matrix multiplication. We can do two kinds of vector multiplication: One is the *scalar* product, for instance a^Tb, where the result is a scalar value. The other one is the *dyadic* multiplication, an example is ab^T, the result now is a rank-one matrix or dyad. Observe that vectors should have the same length in the first case, while it is irrelevant in the other case.

Next we turn to some special matrices.

1.23. Permutation matrix

If we permute the rows or columns of a unit matrix, we get a permutation matrix. As a result, this matrix has one nonzero entry 1 in each row and column, the other elements are zero. To give a permutation matrix, it is not necessary to use a 2-dimensional array. It is enough to give a vector, where the k-th element has value i_k. A possible interpretation is that the i_k-th row gets into the k-th row. Another interpretation may refer to columns in the same sense.

Assume we are interchanging rows of a matrix and we should like to record the performed operations in a vector that would represent a permutation matrix. Initialize that vector by entering k for the k-th element. Then perform the same interchanges for the elements of that vector as done for the rows of the matrix (as if we have attached this vector to the matrix as a column). Then we shall be able to identify which vector gets to what position at the end. For instance, if the first element is equal to 5, then the fifth row was chosen for the first row.

1.24. Unit matrix plus a rank-one matrix

Matrices having the simple form of a unit matrix plus a rank- one matrix play special roles in numerical linear algebra:

$$F=I+ab^T \tag{3.1}$$

The vector product here is also called *outer product* as contrast to *inner product* of vectors, e.g. b^Ta. With their help we can perform various linear algebraic transformations easily by choosing vectors a and b appropriately, according to the task that should be done.

It is easy to find the inverse to this matrix. If we assume that the inverse is also a unit plus rank-one matrix, then the relation $FF^{-1}=I$ results in the form $F^{-1}=I+\alpha ab^T$, and we conclude $\alpha=-1/(1+b^Ta)$ such that

$$F^{-1}=I-\frac{ab^T}{1+b^Ta}. \tag{3.2}$$

The inverse does not exist if $1+b^Ta=0$, from this fact we may conjecture that we have the determinant of F in the denominator.

E3.1 Example

Replace the *i*-th column of the unit matrix with vector a. The result can be expressed as:

$$F = I + (a - e_i)e_i^T.$$

Its inverse can be given by:

$$F^{-1} = I - \frac{(a-e_i)e_i^T}{1+e_i^T(a-e_i)} = I - \frac{(a-e_i)e_i^T}{e_i^T a}. \tag{3.3}$$

Matrices of this kind have importance in algorithms for solving linear systems of equations.

D3.1 Gauss-Jordan transform

Matrices of the type

$$T_{GJ}(a, e_i) = I - \frac{(a-e_i)e_i^T}{e_i^T a} \tag{3.4}$$

will be called Gauss-Jordan transform unless $e_i^T a \neq 0$. As it is seen in Exercise 3.3, it transforms vector a into e_i. Observe that e_i may be replaced by vector b if it is not orthogonal to a, then a will be transformed into b.

E3.2 Example

We do the following operations: the i-th column of matrix A is multiplied by α and the result is added to the k-th column. Find the matrix which exactly performs this task!

Solution.

$$A + \alpha A e_i e_k^T = A(I + \alpha e_i e_k^T). \tag{3.5}$$

E3.3 Example

Show the determinant identity $\left|I + ab^T\right| = 1 + b^T a$!

Solution. Assume vectors a and b are not zero, otherwise the problem is trivial. Let the i-th element of a be $e_i^T a = a_i \neq 0$, and consider matrix $I - (a/a_i - e_i)e_i^T$. This has all diagonal elements 1 and it still has nonzero elements in the i-th column. But these non-diagonal elements can be brought to zero by adding the appropriate multiple of the i-th row, consequently it has determinant 1. Now multiply matrix $I + ab^T$ by $I - (a/a_i - e_i)e_i^T$ from the left. This will move vector a into $a_i e_i$ such that the result is: $I - (a/a_i - e_i)e_i^T + a_i e_i b^T$, which differs from the unit matrix only in the i-th row and column. Now multiply the k-th column by a_k / a_i and add to the i-th $(i \neq k)$ column (see Example 3.6):

$$\left(I - (\frac{a}{a_i} - e_i)e_i^T + a_i e_i b^T\right)\left(I + \frac{a_k}{a_i} e_k e_i^T\right) = I - \left(\frac{a - a_k e_k}{a_i} - e_i\right)e_i^T + a_i e_i b^T + a_k b_k e_i e_i^T.$$

As seen, the k-th element of vector a has turned into zero and the i-th diagonal element got the value $1 + a_i b_i + a_k b_k$. After performing this operation for all $k \neq i$, all non-diagonal elements of the vector a/a_i are zero, the i-th diagonal element is $1 + b^T a$, and the other diagonal elements are 1's. Using the row vectors e_k^T, $k \neq i$ in the next phase, the non-diagonal elements of $a_i e_i b^T$ can be brought to zero without changing the value of the determinant.

Remark. It is much easier to show this relation with the aid of Schur complements that will be seen in the next Chapter.

1.25. Sums of rank-1 matrices, expansions

The unit matrix of order n can be given by the sum of rank-1 matrices of the Cartesian unit vectors: $I_n = \sum_{i=1}^n e_i e_i^T$. Substituting this between two matrices in a matrix product yields a sum of rank-1 matrices, where the columns of A and the rows of B are applied:

$$AB = \sum_{i=1}^n A e_i e_i^T B \ .$$

As it is known, vector x of order n can be expanded by the unit vectors as $x = \sum_{i=1}^n e_i (e_i^T x)$. Similar relation exists for the system of orthonormal vectors $\{q_i\}_{i=1}^n$. To see that, introduce matrix $Q = [q_1 q_2 \ldots q_n]$, then we have $Q^T Q = I = QQ^T$ because of the orthonormal property, consequently one can write $x = QQ^T x = \sum_{i=1}^n q_i (q_i^T x)$. Such Q matrices are called *orthogonal* (or *unitary* in complex).

D3.2 Biorthogonal systems

The systems of vectors $\{a_i\}_{i=1}^n$ and $\{b_i\}_{i=1}^n$ form a *biorthogonal system of vectors*, if $a_i^T b_j = \alpha_i \delta_{ij}$, $\alpha_i \neq 0$ holds for any indices, where δ_{ij} is the *Kronecker delta.* If n is the dimension of the vectors, then the system is said *complete or full.*

T3.1 Theorem, simple product form of a matrix

If the columns of matrix $A \in \mathbb{R}^{m \times n}$ are linearly independent then A can be given by a product of n simple matrices, where a factor consists of a permutation and a Gauss-Jordan transformation of the type $T_i = I - (A_i e_i - e_i) e_i^T / e_i^T A_i e_i$, where the permutation is optional.

Proof. The procedure will be given explicitly. Consider the first column of matrix A at the first step. If the first entry is not zero, $a_{11} = e_1^T A e_1 \neq 0$, then no interchange of rows is needed. If this entry is zero, then we look for a nonzero element in the first column and the row having that element is interchanged with the first row. If all entries in the column are zero, then the matrix is not invertible and contradicts our hypothesis. Denote the first permutation by Π_1 and the row-permuted matrix by $A_1 = \Pi_1 A$.

Multiply A_1 with matrix $T_1 = I - (A_1 e_1 - e_1) e_1^T / e_1^T A_1 e_1$. As we have seen it before, the result of this multiplication in the first column is vector e_1 moreover, we have $T_1^{-1} = I + (A_1 e_1 - e_1) e_1^T$.

We bring the second column into e_2 in the next step. At first we look for a nonzero entry in the second column below the main diagonal and if necessary, apply row permutation Π_2 such that we have nonzero at position 22 of matrix $A_2 = \Pi_2 T_1 A_1$. Now multiplication by

$T_2 = I - (A_2e_2 - e_2)e_2^T / e_2^T A_2 e_2$ moves the second column into e_2. Observe that Π_2 and T_2 leave the first column e_1 unchanged.

Continuing the process in a similar way, we have $A_i = \Pi_i T_{i-1} A_{i-1}$ in the i-th step, where the element in the ii position is nonzero. (If in the i-th column $a_{ji} = 0,\ j \ge i$ would hold, then again, we should get in contradiction with the assumption that the columns are linearly independent.) Multiplication with matrix $T_i = I - (A_i e_i - e_i)e_i^T / e_i^T A_i e_i$ gives e_i in the i-th column and all unit vectors are unchanged in the previous columns. After the n-th step we have

$$T_n \Pi_n T_{i-1} \Pi_n \ldots T_1 \Pi_1 A = \begin{pmatrix} e_1 & e_2 & \ldots & e_n \end{pmatrix},$$

from which we have:

$$\Pi_1^T T_1^{-1} \Pi_2^T T_2^{-1} \ldots T_n^{-1} \begin{pmatrix} e_1 & e_2 & \ldots & e_n \end{pmatrix} = A .$$

Observe that to give T_i^{-1}, it is enough to have index i and vector $a_i = A_i e_i$. ■

1.26. Simple product forms of triangular matrices

Matrix L is said *lower triangular* if all elements above the main diagonal are zero. In a similar way, matrix U is said *upper triangular*, if there are only zeros below the main diagonal. To give the simple product form of triangular matrices is especially simple. If applying the previous theorem, we get the product form of L as:

$$L = \left(I + (L - I)e_1e_1^T\right)\left(I + (L - I)e_2e_2^T\right)\ldots\left(I + (L - I)e_ne_n^T\right),$$

where L has size n. We can use the concise notation

$$L = \prod_{i=1}^{n} \left(I + (L - I)e_ie_i^T\right),$$

if we add that the index of the factors should grow from left to right. This expression can also be shown directly by looking for the j-th column. Multiply by e_j from the right, the first resulting column different from e_j comes from the factor having index j: $e_j + Le_j - e_j = Le_j$ and that is the j-th column of L. The other factors have $e_k,\ k < j$ and multiplying with Le_j gives no further contribution because $e_k^T Le_j = 0,\ k < j$ holds and thus the final result is Le_j. One can also write the product form with row vectors:

$$L = \prod_{i=1}^{n} \left(I + e_ie_i^T (L - I)\right).$$

Its proof is left to the reader.

We have similar relations for the upper triangular matrix U:

$$U = \prod_{i=n\ (-1)}^{1} \left(I + (U - I)e_ie_i^T\right) = \prod_{i=n\ (-1)}^{1} \left(I + e_ie_i^T (U - I)\right),$$

where the factors from left to right should be read in decreasing order of indices.

1.27. Projection matrices

Consider the matrix

$$P = I - ab^T, \tag{3.6}$$

where $b^T a = 1$. It has determinant 0, such that it has no inverse. But it has the interesting property that multiplying with itself results in the same matrix:

$$\left(I - ab^T\right)\left(I - ab^T\right) = I - 2ab^T + ab^T ab^T = I - ab^T.$$

D3.3 Projection matrices

Matrices having the property $P^2 = P$ are said projections or projection matrices.

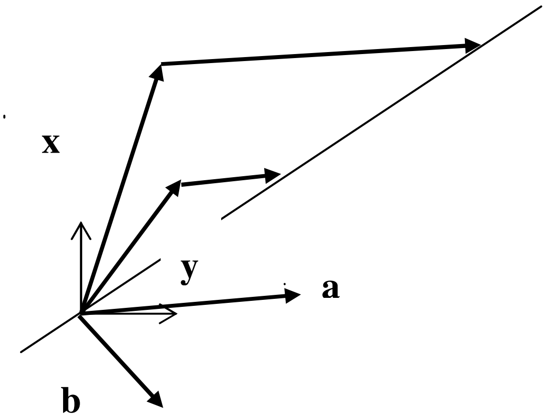

Figure 2.

For $a = b$ the matrix is symmetric. Symmetric projections are also called *orthogonal*, because then the projected vectors Px and $(I - P)x$ are orthogonal to each other. If a and b do not have the same direction, then the projection is said *oblique*. Traditionally projections are also called *idempotent* matrices due to the fact that all positive powers of the matrix are equal. Notice from (3.6): $Pa = 0$ and $b^T P = 0$ hold.

Figure 2. shows, how the projection in (3.6) is actually projecting vectors x and y along direction a into the plane having normal vector b. If the directions of a and b would be the same, then the projection into the plane is done perpendicularly.

1.28. Involutory matrices

Matrix A is said *involutory*, if it satisfies the equation $A^2 = I$. All projections P defines an involutory matrix of the form $I - 2P$:

$$(I - 2P)(I - 2P) = I - 4P + 4P = I,$$

and all involutory matrices defines a projection in the form $(I - A)/2$:

$$(I - A)(I - A)/4 = (2I - 2A)/4 = (I - A)/2.$$

From here it can be seen, that there are infinitely many square roots of the unit matrix of size larger than 1.

The projection $ab^T / b^T a,\ b^T a \neq 0$ can be used to give the involutory matrix: $I - 2ab^T / b^T a$. We can conclude from Figure 2, that such a matrix performs an "oblique" reflection to the plane having normal vector b. In other words, we get along vector a to the plane, then crossing it, we continue our path of the same length on the other side. The reflection is perpendicular to the plane if $a = b$.

1.29. Block matrices

We can build matrices not only from scalars but also from smaller matrices. Such smaller matrices are called *blocks.* And if we form blocks in a matrix then we apply *partitioning*. Partitioning can be done as follows: The unit matrix is sliced into k parts: $I = [E_1, E_2, \ldots, E_k]$. Similarly for the rows we can introduce $I = [F_1, F_2, \ldots, F_j]^T$. Then the ij-th block is $A_{ij} = F_i^T A E_j$ and our matrix is

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1k} \\ A_{21} & A_{22} & \cdots & A_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ A_{j1} & A_{k2} & \cdots & A_{jk} \end{bmatrix}.$$

1.30. Diagonals of a matrix

One can also subdivide matrices along diagonals. The element a_{ij} is located on the k-th diagonal if $k = j - i$. Obviously with this concept the main diagonal is the 0-th diagonal. The *codiagonal* elements reside on the diagonals -1 and 1. It is easy to check that $-n+1 \leq k \leq n-1$.

Sometimes it is useful to introduce the matrix functions *tril* and *triu* as it is done in Matlab. The function $\text{tril}(A)$ returns the lower triangular part of the matrix, in other words, it keeps the diagonals $-n+1 \leq k \leq 0$ of A, the other elements are set to zero. More generally, $\text{tril}(A, \ell)$ keeps the diagonals $-n+1 \leq k \leq \ell$ unchanged and the other elements are turned into zero.

In a similar fashion, $\text{triu}(A, \ell)$ keeps the diagonals $\ell \leq k \leq n-1$ unchanged forming an upper triangular part of the matrix, while the lower diagonals are set to zero. The short form $\text{triu}(A, 0) = \text{triu}(A)$ also exists.

1.31. Problems

3.1. Perform a dyadic multiplication with two vectors. Explain that it should have rank 1. Which method is simpler to multiply by a dyad? *a)* Form $A = ab^T$ then compute Ax. *b)* Compute $b^T x$ first and then multiply vector a with that scalar.

3.2. Consider the permutation matrix $\Pi = [e_2, e_4, e_3, e_1]$. Check that its transpose gives the inverse. Prove this fact in general! How can we store this permutation matrix in a vector?

3.3. Check: $F^{-1} a = e_i$ of (3.3).

3.4. With the aid of formula (3.5), show that the determinant of a matrix will not change if a scalar multiple of a column is added to another column of the matrix. Apply the theorem on the determinant of the product of two matrices!

3.5. Form the rank-1 sum of ADB, where $D=[d_i\delta_{ij}]$ is a diagonal matrix, (only the diagonal elements are nonzero).

3.6. Applying the scalar product and the dyadic product forms of matrix multiplication, show that $\mathrm{tr}(AB)=\mathrm{tr}(BA),\ A,B^T\in\mathbb{R}^{m,n}$!

3.7. Let matrix A be invertible. Give the expansion of vector x in terms of the columns of A.

3.8. Collect the vectors of a biorthogonal system into matrices $A=[a_1,a_2,\dots,a_n]$ and $B=[b_1,b_2,\dots,b_n]$. Check that A^TB is a diagonal matrix! How can we give vector x as a linear combination of vectors a_i? And how can we give the expansion with the aid of vectors b_i?

3.9 Check: if P is a projection, then $I-P$ is also a projection.

3.10. A plane has normal vector s and its defining equation is $s^Tx=\sigma$. Introduce the projection $P=I-ss^T/s^Ts$. Show that for all vectors y the operation $Py+\sigma s/s^Ts$ produces a vector in the plane.

3.11. Show with the previous matrix P: $Py\perp s$, in other words Py is perpendicular to s. Give the vector that connects $Py+\sigma s/s^Ts$ and y !

3.12. Show that the *backward identity* $J=[e_ne_{n-1}\dots e_1]$, where the columns of the unit matrix are given in reverse order, is involutory. What projection will it define for $n=2,3$?

3.13. Show that matrix $I-2(x-y)(x-y)^T/(x-y)^T(x-y)$ will reflect vectors x and y into each other, if they are different and have the same length: $x^Tx=y^Ty$.

3.14. We have the possibility to reflect vector x with the previous matrix into vector $y=\pm\sigma e_1$, where $\sigma^2=x^Tx$. How should we choose the sign of σ to avoid cancellation error in the denominator?

3.15. Introduce $F=I+UV^T$, where the unit matrix is modified by the $n\times l$ matrices U and V, that is, they have $l<n$ columns. If F is invertible, show that $F^{-1}=I-U(I_l+V^TU)^{-1}V^T$ (*Sherman-Morrison-Woodbury formula*) holds, where I_l is a unit matrix of size $l\times l$.

LU-decomposition of matrices, Gauss-Jordan algorithm

The *LU*-decomposition can be considered as another form of Gaussian elimination aimed to store intermediate results. This is done by factorizing matrix A into the product of a lower L and an upper U triangular matrix.

1.32. The Gauss transform

As it is known, Gasussian elimination interchanges rows if necessary and adds them up to achieve an upper triangular form of the linear system of equations. Having that form, it will be easy to solve the linear system by back substitution.

It is possible to give an identity plus rank-1 matrix that brings the column elements $a_{ij},\ j < i \le n$ into zero, it is called the *Gauss transform:*

$$T_G(A, j) = I - \left(\text{tril}(A)e_j / a_{jj} - e_j\right)e_j^T, \tag{4.1}$$

where $\text{tril}(A)$ is defined in Sect. 3.9. If it is applied to the *j*-th column, the result is: $T_G(A, j)Ae_j = Ae_j - \text{tril}(A)e_j + a_{jj}e_j$ such that the diagonal element remains a_{jj} and the other elements below the diagonal are zero. The other numbers above the diagonal are unchanged. The Gauss transformation is a simple lower triangular matrix that differs from the unit matrix only in the *j*-th row, where the vector $-\text{tril}(A,-1)e_j / a_{jj}$ is still added. (Observe that the *j*-th element is canceling in .)

T4.1 LU-decomposition

If matrix $A \in \mathbb{R}^{n\times n}$ is nonsingular, then its rows can be reordered by a permutation matrix P into $P^T A$ such that it can be decomposed into a product of lower L and upper U triangular matrices. The decomposition of $P^T A$ is unique if the diagonal elements of L are chosen to be 1's.

Proof. Having seen the algorithm in Sect. 3.5, it is possible to give the *i*-th step in general. At first, look for a nonzero in column *i* and move it with the row changing permutation P_i^T into the diagonal position *ii*. Denote the resulting matrix by A_i. Then apply Gauss transformation $T_G(A_i, i)A_i$ to get zeros below the diagonal in the *i*-th column. We can write $T_G(A_i, i)$ as L_i^{-1} such that

$$L_i e_i = \text{tril}(A_i) / e_i^T A_i e_i \tag{4.2}$$

because L_i^{-1}, as a Gauss transformation moves $\text{tril}(A_i) / e_i^T A_i e_i$ into e_i. The divisor $e_i^T A_i e_i$ is called the *pivot* and we can identify the *i*-th column of L_i from (4.2).

If there are no permutations, we have the following picture after the first step:

$$L_1^{-1}A_1 = \begin{bmatrix} a_{11} & * & \dots & * \\ 0 & * & \dots & * \\ \vdots & \vdots & \ddots & \vdots \\ 0 & * & \dots & * \end{bmatrix}, \tag{4.3}$$

where the asterisk * indicates nonzero elements. It is seen, the first column of the upper triangular matrix has shown up and $L_1 = I + (Ae_1 / a_{11} - e_1)e_1^T$ is the first multiplier of matrix L in the LU-decomposition, from where we can identify the first column of L: Ae_1 / a_{11}.

In the second step we repeat the same procedure for the right lower $(n-1)\times(n-1)$ block:

$$A_2 = \begin{pmatrix} a_{11} & \begin{matrix} * & \cdots & * \end{matrix} \\ \begin{matrix} 0 \\ \vdots \\ 0 \end{matrix} & \boxed{\begin{matrix} \boxed{*} & \cdots & * \\ \vdots & \ddots & \vdots \\ * & \cdots & * \end{matrix}} \end{pmatrix},$$

thus first element is 0 and second element is 1 in the second column of L_2. Continuing the process, we get matrices L and U as

$$L = L_1 L_2 \ldots L_{n-1}, \qquad U = L_{n-1}^{-1} L_{n-2}^{-1} \ldots L_1^{-1} A = \begin{pmatrix} * & * & \cdots & * \\ & * & \cdots & * \\ & & \ddots & \vdots \\ & & & * \end{pmatrix}. \tag{4.4}$$

If permutations are needed to get nonzero pivots then we may assume, the necessary row changes are performed beforehand such that $P^T A$ has an LU-decomposition such that

$$A = PLU\,. \tag{4.5}$$

■

If our task is to solve the linear system $Ax = b$ then it is practical to attach vector b to matrix A: $[A, b]$, because the same transformations also act on b. For instance, let the linear system be:

$$\begin{bmatrix} 2 & 0 & 3 \\ -4 & 5 & -2 \\ 6 & -5 & 4 \end{bmatrix} x = \begin{bmatrix} -1 \\ 3 \\ -3 \end{bmatrix}.$$

Observe that multiplying with L_1^{-1} will not change the first row of the matrix: $e_1^T L_1^{-1} = e_1^T$. And the ik element in the right lower block of order $(n-1)$ will change to:

$$e_i^T \left(I - \left(\frac{Ae_1}{a_{11}} - e_1 \right) e_1^T \right) Ae_k = a_{ik} - \frac{a_{i1} a_{1k}}{a_{11}} = a_{ik} - \left(\frac{a_{i1}}{a_{11}} \right) a_{1k}, \quad k, i > 1\,.$$

That shows, for the right lower block a rank-1 matrix is subtracted: $A - \dfrac{Ae_1}{e_1^T A e_1} e_1^T A$. Here the column vector is just the first column of L_1, so that the practical approach is the following:

1. mark the pivot element,
2. divide column elements below the pivot,
3. and keep the row of the pivot unchanged.

4. Form the rank-1 matrix of the pivot column and pivot row and subtract it from the rest of A:

$$\begin{bmatrix} 2 & 0 & 3 & -1 \\ -4 & 5 & -2 & 3 \\ 6 & -5 & 4 & -3 \end{bmatrix} \to \begin{bmatrix} \boxed{2} & 0 & 3 & -1 \\ -2 & 5 & 4 & 1 \\ 3 & -5 & -5 & 0 \end{bmatrix} \to \begin{bmatrix} 2 & 0 & 3 & -1 \\ -2 & \boxed{5} & 4 & 1 \\ 3 & -1 & -1 & 1 \end{bmatrix}$$

$$L = \begin{bmatrix} 1 & & \\ -2 & 1 & \\ 3 & -1 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 2 & 0 & 3 \\ & 5 & 4 \\ & & -1 \end{bmatrix}.$$

At the end of the process, the linear system is reduced to an upper triangular system

$$Ux = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix},$$

that is easy to solve for vector $x = [1 \quad 1 \quad -1]^T$.

Pivoting is always done in numerical programs. In all columns the largest element in absolute value is found and moved to the diagonal position. It is done for a numerically more stable algorithm. Another variant is when the position of the pivot is recorded and no row interchanges are done.

1.33. Operation count of LU-decomposition

There are $n-1$ divisions in the first step by dividing the column elements. Subtracting the rank-1 matrix needs $(n-1)^2$ multiplications and additions. The time count for the arithmetic operations are taken the same, therefore the operation count of the first step is: $(n-1)(2n-1)$ flops (= *fl*oating point *op*eration). We get $(n-2)(2n-3)$ flops in the next step and the total count is $\sum_{k=1}^{n-1}(k-1)(2k-1)$ flops. The result is approximated by taking the integral of the highest order term from 0 to n: $2n^3/3$. The additional terms are lower powers of n, we do not find them because the highest order term is the dominant.

1.34. Block LU-decomposition

Sometimes it may be practical to apply block form in the decomposition or inversion of the matrix. That is the case if one can separate an easily invertible block in the matrix, for instance, it is diagonal or triangular. Here we shall consider block LU -decomposition for a block 2×2 matrix. The pivot now is a block and it is assumed that it has an inverse. Let the block form of the unit matrix be $I = [E_1, E_2]$, $A_{ij} = E_i^T A E_j$, then matrix L below is the block counterpart of L_1 that can be seen in (4.2) (see also *Exercise* 3.14)

$$L^{-1} = I - \left(AE_1A_{11}^{-1} - E_1\right)E_1^T \tag{4.6}$$

and one step of the block LU-decomposition is:

$$\begin{bmatrix} \boxed{A_{11}} & A_{12} \\ A_{21}A_{11}^{-1} & A_{22} - A_{21}A_{11}^{-1}A_{12} \end{bmatrix}, \text{ where } L = \begin{bmatrix} I_1 & 0 \\ A_{21}A_{11}^{-1} & I_2 \end{bmatrix}, \; U = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} - A_{21}A_{11}^{-1}A_{12} \end{bmatrix}. \tag{4.7}$$

1.35. Schur-complement

The right lower block in the decomposition is called the Schur-complement of matrix A with respect to the block A_{11}, in notation: $\left(A\middle|A_{11}\right)=A_{22}-A_{21}A_{11}^{-1}A_{12}$. Of course, there also exists the Schur-complement with respect to the block A_{22}. Formally it can be found by interchanging indices $1\leftrightarrow 2$.

1.36. Inversion by partitioning

We can write in (4.7):

$$A=\begin{bmatrix} I_1 & 0 \\ A_{21}A_{11}^{-1} & I_2 \end{bmatrix}\begin{bmatrix} A_{11} & A_{12} \\ 0 & \left(A\middle|A_{11}\right) \end{bmatrix}=\begin{bmatrix} I_1 & 0 \\ A_{21}A_{11}^{-1} & I_2 \end{bmatrix}\begin{bmatrix} A_{11} & \\ & \left(A\middle|A_{11}\right) \end{bmatrix}\begin{bmatrix} I_1 & A_{11}^{-1}A_{12} \\ 0 & I_2 \end{bmatrix},$$

from where

$$A^{-1}=\begin{bmatrix} I_1 & -A_{11}^{-1}A_{12} \\ 0 & I_2 \end{bmatrix}\begin{bmatrix} A_{11}^{-1} & \\ & \left(A\middle|A_{11}\right)^{-1} \end{bmatrix}\begin{bmatrix} I_1 & 0 \\ -A_{21}A_{11}^{-1} & I_2 \end{bmatrix}. \tag{4.8}$$

Changing the product to block outer product form (see *Problem 3.5*) this can still be expanded into

$$A^{-1}=\begin{bmatrix} A_{11}^{-1} & 0 \\ 0 & 0 \end{bmatrix}+\begin{bmatrix} -A_{11}^{-1}A_{12} \\ I_2 \end{bmatrix}\left(A\middle|A_{11}\right)^{-1}\begin{bmatrix} -A_{21}A_{11}^{-1} & I_2 \end{bmatrix}. \tag{4.9}$$

T4.2 Theorem on the Schur-complement in LU-decomposition

The emerging right lower block in standard *LU*-decomposition is the Schur-complement with respect to the left upper block.

Proof. An intermediate phase can be represented by the partitioning:

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}=\begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix}\begin{bmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{bmatrix}. \tag{4.10}$$

For block A_{11} the *LU*-decomposition is already done. The right lower block will lead to L_{22} and U_{22}, in fact, before decomposition at this stage we have $L_{22}U_{22}$. After having done multiplication, we get

$$A_{11}=L_{11}U_{11},\quad A_{12}=L_{11}U_{12},\quad A_{21}=L_{21}U_{11} \tag{4.11}$$

and

$$L_{22}U_{22}=A_{22}-L_{21}U_{12}=A_{22}-A_{21}U_{11}^{-1}L_{11}^{-1}A_{12}=A_{22}-A_{21}A_{11}^{-1}A_{12}, \tag{4.12}$$

and that is equal to the Schur-complement $\left(A \mid A_{11}\right)$. ■

1.37. The Gauss-Jordan algorithm for linear systems

It was shown in the previous chapter (*Theorem T3.1*) that an invertible matrix can be brought to the unit matrix by a series of Gauss-Jordan transforms. Applying the same series of transforms to the right hand side of the linear system of equations $Ax=b$ gives another way of solution. However, LU-decomposition is preferred as the amount of work for factorizing is less in that case. But the solution phase needs the same amount of operations: $2n^2$ flops.

1.38. The Gauss-Jordan method for matrix inversion

Though it is not suggested for solving linear system of equations, the Gauss-Jordan transforms need the same amount of operations for computing the inverse of A. Now one may get the idea to complement A with the unit matrix: $A \to [A, I]$ and apply the series of transformations for the extended matrix: $[TA, T]=[I, T]$. Clearly, we have $T = A^{-1}$. Actually this type of matrix inversion can be done by using no more additional computer memory than a vector for recording permutations.

Assume in the i-th step that we have matrix A_i with the necessary row change already done. Then the i-th multiplication is:

$$\left(I - \frac{A_i e_i - e_i}{e_i^T A_i e_i} e_i^T \right) A_i = A_i - \frac{A_i e_i e_i^T A_i}{e_i^T A_i e_i} + \frac{e_i e_i^T A_i}{e_i^T A_i e_i}.$$

The first two terms on the right hand side is the rank-1 update that was already seen at LU-decomposition. But now it should be applied for the whole matrix with the exception of the i-th row and column. The third term shows that the i-th row should be divided by the pivot, - observe that the i-th row from the first two term is zero.

What has been said will be explained through an example. We should like to compute the inverse of the matrix:

$$\begin{bmatrix} 0 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 3 & 6 \end{bmatrix}$$

The first step is swapping the first two rows in the extended matrix:

$$\begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 2 & 3 & 0 & 1 & 0 \\ 1 & 3 & 6 & 0 & 0 & 1 \end{bmatrix} \underset{\text{sorcsere}}{1\leftrightarrow 2} \begin{bmatrix} \boxed{1} & 2 & 3 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 3 & 6 & 0 & 0 & 1 \end{bmatrix} \xrightarrow{Tr1}$$

After the first transformation step the first column goes into e_1, the first row is divieded by the pivot and the subtraction of the rank-1 matrix is done at the other places. The further steps are done similarly:

$$\to \begin{bmatrix} 1 & 2 & 3 & 0 & 1 & 0 \\ 0 & \boxed{1} & 1 & 1 & 0 & 0 \\ 0 & 1 & 3 & 0 & -1 & 1 \end{bmatrix} \xrightarrow{Tr2} \begin{bmatrix} 1 & 0 & 1 & -2 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & \boxed{2} & -1 & -1 & 1 \end{bmatrix} \xrightarrow{Tr3} \begin{bmatrix} 1 & 0 & 0 & -3/2 & 3/2 & -1/2 \\ 0 & 1 & 0 & 3/2 & 1/2 & -1/2 \\ 0 & 0 & 1 & -1/2 & -1/2 & 1/2 \end{bmatrix}$$

.

After the last step the inverse can be found at the place of the starting unit matrix.

Of course, we may modify the algorithm such that no more memory is needed. We just have to observe that one can collect here a unit matrix in all phases of computation that need not be stored. After all transformation steps a new vector appears in the field of the right 3×3 matrix (initially the unit matrix) and a unit vector comes into the place of a column in the left 3×3 matrix field (initially the starting matrix). We proceed in the concise algorithm as follows: write the incoming vector on the right side into the place of the incoming unit vector at the left side. Then the new vector from right at the place of the i th unit vector will be

$$\left(I-\frac{A_ie_i-e_i}{e_i^TA_ie_i}e_i^T\right)e_i=e_i-\frac{A_ie_i-e_i}{e_i^TA_ie_i}=\begin{cases}1/e_i^TA_ie_i, & j=i,\\ -a_{ji}^{(i)}/a_{ii}^{(i)}, & j\neq i\end{cases}$$

Now the pivot should be replaced by its reciprocal, the other elements of the column get negative sign and are divided by the pivot. The rank one updating is the same as before. The marked element shows the pivot and that defines the row and column of the rank-1 matrix. Now the concise algorithm is:

$$\begin{bmatrix}0&1&1\\1&2&3\\1&3&6\end{bmatrix}\overset{\substack{1\leftrightarrow 2\\ \text{row change}}}{\longrightarrow}\begin{bmatrix}\boxed{1}&2&3\\0&1&1\\1&3&6\end{bmatrix}\overset{Tr1}{\longrightarrow}\begin{bmatrix}1&2&3\\0&\boxed{1}&1\\-1&1&3\end{bmatrix}\overset{Tr2}{\longrightarrow}\begin{bmatrix}1&-2&1\\0&1&1\\-1&-1&\boxed{2}\end{bmatrix}\overset{Tr3}{\longrightarrow}$$

$$\rightarrow\begin{bmatrix}3/2&-3/2&-1/2\\1/2&3/2&-1/2\\-1/2&-1/2&1/2\end{bmatrix}\overset{\substack{1\leftrightarrow 2\\ \text{column change}}}{\longrightarrow}\begin{bmatrix}-3/2&3/2&-1/2\\3/2&1/2&-1/2\\-1/2&-1/2&1/2\end{bmatrix}.$$

Because of the initial row change we have inverted matrix ΠA, where Π is a permutation matrix. Its inverse is $A^{-1}\Pi^T$, because $\Pi^{-1}=\Pi^T$. Consequently, the result still should be multiplied by Π^T from the right, that is the $1\leftrightarrow 2$ column change.

1.39. Problems

4.1. Using LU-decomposition, solve the following linear system:

$$\begin{bmatrix}2&2&3\\4&3&7\\6&7&5\end{bmatrix}x=\begin{bmatrix}1\\5\\-3\end{bmatrix}.$$

4.2. Find the operation count for Ax, LUx, $U^{-1}L^{-1}x$. The factorizations given in Section 3.11 may be applied for the last case.

4.3. Using Problem 3.15, show that the matrix of (4.6) can be inverted by taking the negative of the block 21. Similar result for the upper triangular case can be found by transposition.

4.4. Let L_{11} be a lower triangular matrix, which is complemented by a block row $\begin{bmatrix}L_{21} & L_{22}\end{bmatrix}$ to a larger lower triangular matrix. Assuming that the diagonal blocks are invertible, apply the partitioned inverse to get

$$\begin{bmatrix}L_{11}&\\L_{21}&L_{22}\end{bmatrix}^{-1}=\begin{bmatrix}L_{11}^{-1}&\\-L_{22}^{-1}L_{21}L_{11}^{-1}&L_{22}^{-1}\end{bmatrix}.$$

4.5. By using the block partitioned form, check the determinant identity $|A|=|A_{11}|\left|\left(A|A_{11}\right)\right|$.

4.6. With the aid of the previous problem, check the identities

$$\left\|\begin{bmatrix}1&-b^T\\a&I\end{bmatrix}\right\|=1+b^Ta=\left|I+ab^T\right|.$$

Compare it with the approach given in *Example E3.3*!

4.7. What is the dominant term in the operation count of the Gauss-Jordan factorization?

4.8. $A = \begin{bmatrix} 2 & -2 & 1 & 0 \\ 4 & -1 & 3 & -1 \\ -2 & -1 & 0 & 2 \\ 6 & -3 & 2 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} -3 \\ -2 \\ -2 \\ 0 \end{bmatrix}, \quad A = LU, \quad Ax = b. \qquad L = ?, \quad U = ? \quad x = ?$

Some properties of *LU*-decomposition, special inverses

1.40. Symmetric positive definite matrices

A real symmetric matrix A is said *positive definite*, if $x^T Ax > 0$ holds for all vectors $x \neq 0$. The matrix is *positive semidefinite*, if $x^T Ax \geq 0$ holds. The concept of *negative definiteness* and *negative semidefiniteness* can be introduced similarly: $x^T Ax < 0$ or $x^T Ax \leq 0$ hold respectively. In the case of *indefiniteness* the inner product may assume negative and positive values.

One can give two other equivalent definitions for the property of positive definiteness. One is that all eigenvalues of the matrix are positive, the other has the condition that all principal minors (determinant of left upper symmetric blocks of increasing order) are positive. A semidefinite matrix has zero eigenvalue and zero principal minor.

We say a nonsymmetric matrix positive definite, if its symmetric part is positive definite. The symmetrix part of a matrix is $A_+ = (A + A^T)/2$ and the anti-symmetric part is $A_- = (A - A^T)/2$, $A = A_+ + A_-$. Observe, that the inner product of the anti-symmetric part is always zero: $x^T A_- x = 0$.

If we choose $x = e_i$, then it follows from the definition that $a_{ii} > 0$ holds for all i at real symmetric positive definite matrices and in case of $x = e_i \pm e_j$, it can be checked easily that $a_{ii} + a_{jj} \pm 2a_{ij} > 0$ must fulfill. Sometimes these simple conditions are useful to decide quickly if a matrix is positive definite at all. For instance, if all the diagonal elements are equal to zero, and if there are nonzero nondiagonal elements, then it can be seen at once that the matrix is indefinite.

T5.1 Theorem for sufficiency for positive definiteness

If the matrix has the form $A = V^T V$, where the columns of V are linearly independent, then A is positive definite.

Proof. By definition, we must have $x^T Ax = x^T V^T Vx = \|Vx\|_2^2 > 0$ for all nonzero x, and $Vx \neq 0$, if V has independent columns. ■

T5.2 Theorem on preserving positive definiteness in LU-decomposition

If matrix A is positive definite then this property is preserved in *LU*-decomposition, in other words: after each step positive definiteness is inherited in the remaining lower right block. The statement is also true for block *LU*-decomposition.

Proof. Let a block form of A be

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad (A \mid A_{11}) = A_{22} - A_{21}A_{11}^{-1}A_{12}$$

and after one step of block *LU*-decomposition the remaining block is the Schur-komplement $\left(A\,|\,A_{11}\right)$. It has to be shown that for all nonzero vectors x_2 the relation $x_2^T\left(A\,|\,A_{11}\right)x_2>0$ holds. To show the statement, we choose a partitioned vector $x^T=\left(x_1^T,x_2^T\right)$ such that $x^TAx=x_2^T\left(A\,|\,A_{11}\right)x_2$. We can achieve this task by choosing x_1 to x_2 such that the first row gives zero: $A_{11}x_1+A_{12}x_2=0$. With this $x_1=-A_{11}^{-1}A_{12}x_2$ and

$$0<x^TAx=\begin{bmatrix}x_1^T & x_2^T\end{bmatrix}\begin{bmatrix}0\\ A_{21}x_1+A_{22}x_2\end{bmatrix}=x_2^T(A_{22}-A_{21}A_{11}^{-1}A_{12})x_2\ .$$

■

Remarks. That positive semidefiniteness is also inherited can be shown in a similar way.

T5.3 Theorem on decomposition of positive semidefinite matrices

If matrix A is positive semidefinite , then it can be decomposed in the form $A=PLL^TP^T$, where P is a permutation.

Proof. We have seen, all diagonal elements of A may only be nonnegative. If $a_{11}>0$ holds, then compute

$$A_2=A-\frac{Ae_1e_1^TA}{e_1^TAe_1}, \tag{6.1}$$

which has zero first row and column. Choose $Le_1=Ae_1/\sqrt{a_{11}}$ for the first column of L then we may write $A_2=A-Le_1e_1^TL$.

If the first diagonal element is zero, then look for a nonzero in the diagonal and move it to the first position by interchanging rows and columns with the same indices. That makes possible to proceed as done in (6.1).

Continue procedure for the remaining smaller right lower block until we can find a nonzero diagonal element. Thus we gain a new column in matrix L at all steps. If we arrive to a stage where all diagonal elements are zero then the whole block has to be zero. That can be seen from the remark just before Theorem 5.1.1 where it was stated that a symmetric matrix having a zero diagonal and nonzero elements at other positions should be indefinite, and that would contradict to the preservation of semidefiniteness.

Observe that no interchanges of rows and columns are needed if the matrix is positive definite. If P is not unity then we may write $A=\tilde{L}\tilde{L}^T$, where $\tilde{L}=PL$. ■

For symmetric, positive definite matrices, we call the decomposition $A=LL^T$ *Cholesky-decomposition.* Now the diagonal elements of L are not 1's, e.g. the first element of $Le_1=Ae_1/\sqrt{a_{11}}$ is $\sqrt{a_{11}}$. The Cholesky-decomposition can be done similarly to *LU*-decomposition, the difference is that we have to take square root and the corresponding row and column should be divided with it. In a computer algorithm we can exploit the fact that the upper triangle need not be computed hence the operation count is roughly halved.

E5.1 Example for Cholesky-decomposition

$$\begin{bmatrix} 4 & -2 & 2 \\ -2 & 10 & -7 \\ 2 & -7 & 21 \end{bmatrix} \to \begin{bmatrix} \boxed{2} & -1 & 1 \\ -1 & 9 & -6 \\ 1 & -6 & 20 \end{bmatrix} \to \begin{bmatrix} 2 & & \\ -1 & \boxed{3} & -2 \\ 1 & -2 & 16 \end{bmatrix} \to \begin{bmatrix} 2 & & \\ -1 & 3 & \\ 1 & -2 & \boxed{4} \end{bmatrix},$$

$$L = \begin{bmatrix} 2 & & \\ -1 & 3 & \\ 1 & -2 & 4 \end{bmatrix}, \quad L^T = \begin{bmatrix} 2 & -1 & 1 \\ & 3 & -2 \\ & & 4 \end{bmatrix}.$$

As seen, subtraction of rank-1 matrices are done similarly to LU-decomposition.

1.41. Diagonal-dominance

A matrix is said *diagonally dominant by rows* if in each row the absolute sum of nondiagonal row elements is less than the absolute value of the diagonal element:

$$|a_{ii}| > \left\| e_i^T (A - \mathrm{diag}(A)) \right\|_\infty ,$$

where $\mathrm{diag}(A) = D$ denotes the diagonal matrix formed from the diagonal elements of A. The matrix is said *essentially diagonally dominant* if the sign $\geq$ is allowed in some nonzero rows but not in all rows. The definition of *diagonal dominance with respect to columns* can be done similarly.

T5.4 Theorem, diagonal-dominance is inherited in LU-decomposition

If A is diagonally dominant by rows, then in all steps of LU -decomposition the right lower – still not decomposed – block will preserve diagonal dominance. In other words, the Schur-complement inherits diagonal dominance.

Proof. It is enough to check the first step because all subsequent steps are similar. Introduce the partitioned form

$$A = \begin{bmatrix} a_{11} & b^T \\ c & A_{22} \end{bmatrix}.$$

After the first step of LU-decomposition, the right lower block can be expressed as

$$A_{22} - \frac{cb^T}{a_{11}}, \quad k\text{-th row: } e_k^T \begin{bmatrix} b^T \\ A_{22} \end{bmatrix} - a_{k1} b^T / a_{11}.$$

Initially the k-th row was diagonally dominant. Now the first element a_{k1} is left out from this row – this still improves diagonal dominance but the row $a_{k1} b^T / a_{11}$ is added. Observe that

$$|a_{k1}| > |a_{k1}| \left\| b^T \right\|_\infty / |a_{11}|$$

holds because of the diagonal dominance of the first row, that is, the absolut sum of the elements of b divided by $|a_{11}|$ is less than 1. Therefore the added row has smaller absolute sum than the absolute value of the cancelled element a_{k1}, hence the diagonal dominance still improves unless a_{k1} is zero. The situation is similar in the following steps. ■

A consequence of the theorem is that the diagonal element can always be chosen as a pivot, hence no pivoting will be needed in the LU-decomposition.

1.42. Bi- and tridiagonal matrices

1.42.1 Special matrices

We have a bidiagonal matrix if there are nonzero elements only in the diagonal and one of the codiagonals (a diagonal next to the main diagonal): $a_{ij} \neq 0,\ j-i \in \{0,1\}$, or $j-i \in \{0,-1\}$. A special representative is the difference matrix:

$$K = \begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}, \quad K^{-1} = S = \begin{bmatrix} 1 & & & \\ 1 & 1 & & \\ \vdots & \vdots & \ddots & \\ 1 & 1 & \cdots & 1 \end{bmatrix}.$$

Its inverse is just the summation matrix. With the help of these two matrices, it is easy to give the inverse of the frequently used matrix

$$T = \begin{bmatrix} 2 & -1 & & \\ -1 & 2 & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{bmatrix}. \tag{6.2}$$

In fact, we have:

$$T^{-1} = \left(K+K^T\right)^{-1} = \left[K\left(S+S^T\right)K^T\right]^{-1} = K^{-T}\left(I+ee^T\right)^{-1}K^{-1} = K^{-T}\left(I - \frac{ee^T}{1+n}\right)K^{-1}, \tag{6.3}$$

where e is a vector having all elements 1. Now it is possible to give an algorithm for the computation of $T^{-1}x$ with operation count $4n$ for n-dimensional problems.

1.42.2 Diagonally dominant tridiagonal matrices

As we have seen it before, this time pivoting is not needed in the LU-decomposition. If it is done in a standard way then solving a linear system of order n has operation count essentially $9n$ flops. However, for tridiagonal matrices there exist two methods, where $8n$ flops are enough. We give these two algorithms in the sequel. The first method may be called *fast LU-decomposition*. Let us choose the form of the tridiagonal system as:

$$Tx = \begin{bmatrix} d_1 & c_1 & & \\ a_1 & d_2 & \ddots & \\ & \ddots & \ddots & c_{n-1} \\ & & a_{n-1} & d_n \end{bmatrix} x = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}. \tag{6.4}$$

The first LU step will introduce changes only in the second row:

$$\left[a_1/d_1 \quad d_2 - a_1c_1/d_1 \quad c_2 \quad \ldots \quad 0\right]x = b_2 - b_1a_1/d_1.$$

The result is a new tridiagonal matrix of size one less, for which the procedure may be applied repeatedly. Continuing, finally the pivots and right vector elements can be given as:

$$\begin{aligned} d_1' = d_1; \; d_i' &= d_i - a_{i-1} c_{i-1} / d_{i-1}', \quad i = 2,3,\ldots,n, \\ b_1' = b_1; \; b_i' &= b_i - a_{i-1} b_{i-1}' / d_{i-1}', \quad i = 2,3,\ldots,n. \end{aligned} \tag{6.5}$$

Matrix U is upper bidiagonal and the system to solve is:

$$\begin{bmatrix} d_1 & c_1 & & \\ 0 & d_2' & \ddots & \\ & \ddots & \ddots & c_{n-1} \\ & & 0 & d_n' \end{bmatrix} x = \begin{bmatrix} b_1 \\ b_2' \\ \vdots \\ b_n' \end{bmatrix}, \quad x_n = b_n' / d_n'; \; x_i = (b_i' - c_i x_{i+1}) / d_i', \; i = n-1, n-2, \ldots, 1.$$

As we see, matrix L is not needed for the solution, on the other hand, we have a_{i-1} / d_{i-1}' in both lines of (6.5) and that can be computed only once. The algorithm:

```
Start: $d_1' = d_1$; $b_1' = b_1$;
for $i = 2,3,\ldots,n$
    $s := a_{i-1} / d_{i-1}'$; $d_i' := d_i - c_{i-1} * s$;  $b_i' := b_i - b_{i-1}' * s$;
$x_n := b_n' / d_n'$;
for $i = n-1, n-2, \ldots, 1$
    $x_i := (b_i' - c_i * x_{i+1}) / d_i'$;
```

The other way – passage method – starts from the recursion of the second phase:

$$x_i = f_i - g_i x_{i+1}.$$

From the first row: $x_1 = (b_1 - c_1 x_2) / d_1$, from here $f_1 = b_1 / d_1$ and $g_1 = c_1 / d_1$. Substituting the formula for x_{i-1} into the i-th row gives

$$a_{i-1}(f_{i-1} - g_{i-1} x_i) + d_i x_i + c_i x_{i+1} = b_i,$$

from here

$$x_i = \frac{b_i - a_{i-1} f_{i-1}}{d_i - a_{i-1} g_{i-1}} - \frac{c_i}{d_i - a_{i-1} g_{i-1}} x_{i+1} = f_i - g_i x_{i+1},$$

and that helps to identify f_i and g_i. We have the following algorithm:

```
Start: $f_1 = b_1 / d_1$; $g_1 := c_1 / d_1$;
for $i = 2,3,\ldots,n$
    $s := d_i - a_{i-1} g_{i-1}$; $f_i := (b_i - a_{i-1} f_{i-1}) / s$;  $g_i := c_i / s$;
$x_n := f_n$;
for $i = n-1, n-2, \ldots, 1$
    $x_i := f_i - g_i * x_{i+1}$;
```

1.43. Problems

5.1. We have the Cholesky-decomposition $A = LL^T$. Give the operation count for computing $x^T A x$ if matrix A is used in the computation! How can we decrease the number of operations if $x^T L L^T x$ is used?

5..2. We can avoid square roots, if we use the form $A = LDL^T$, where L has unit diagonal and D is a diagonal matrix. Elaborate the steps of this decomposition! This method can also be used for indefinite matrices if the pivot elements in D happen to be large enough.

5.3. Show that the row diagonal dominance is preserved if the matrix is multiplied from the left by a nonsingular diagonal matrix. Also, it is preserved if two rows and columns with the same row and column numbers are interchanged.

5.4. Show that for the LU-decomposition of essentially diagonally dominant matrices (by row): strict diagonally dominance takes place in the jth step for the k-th row, if there was strict diagonal dominance in the j-th row and the element $a_{jk}^{(j)}$, $j < k$ was not zero.

5.5. Show that diagonal dominance by columns is also inherited in LU-decomposition.

5.6. If we are given a new right vector b, which data should be preserved and which data should be recomputed in both algorithms (fast LU and passage)?

5.7. Prove that the tridiagonal matrix in (6.2) is positive definite, because it has a LL^T-decomposition.

5.8. $$\begin{bmatrix} 4 & -2 & 4 & -4 \\ -2 & 10 & -5 & 5 \\ 4 & -5 & 9 & -3 \\ -4 & 5 & -3 & 22 \end{bmatrix} = LL^T, \ L = ?$$

Gram-Schmidt orthogonalisation, QR-decomposition

Among simple transformations of linear algebra we have already seen projection matrices in Chapter 3. Such matrices are capable of generating orthogonal vectors from a set of vectors. Arranging these vectors into columns of matrices will lead us to a newer, called *QR*-decomposition of matrices.

1.44. Gram-Schmidt orthogonalisation

Assume we have a set of linearly independent vectors: $\{a_i\}_{i=1}^k$, $a_i \in \mathbb{R}^m$. We should like to generate an orthogonal system by using these vectors such that they form a base for expanding all vectors a_i. Then we can proceed as follows. Denote the new orthogonal vectors by q.

Choose $q_1 = a_1$ in the first step. The next vector is prepared so that vector a_2 is orthogonalised to q_1 with the aid of an orthogonal projection:

$$\left(I - \frac{q_1 q_1^T}{q_1^T q_1} \right) a_2 = q_2 . \tag{6.1}$$

It easy to check: $q_1^T q_2 = 0$. Then the next vector a_3 is orthogonalised to q_1 és q_2:

$$\left(I - \frac{q_2 q_2^T}{q_2^T q_2} \right) \left(I - \frac{q_1 q_1^T}{q_1^T q_1} \right) a_3 = q_3 . \tag{6.2}$$

Once again, checking orthogonality by computing the scalar product, gives the result: $q_1 \perp q_3$ and $q_2 \perp q_3$ hold. Now introduce the projection matrix

$$P_i = I - \frac{q_i q_i^T}{q_i^T q_i} \tag{6.3}$$

for the i-th vector. As we have seen it before, if we multiply a vector with this matrix, the result is a vector orthogonal to q_i.

We conclude that the $i+1$-st orthogonal vector can be generated from a_{i+1} by the series of projections:

$$P_i P_{i-1} \dots P_1 a_{i+1} = q_{i+1} . \tag{6.4}$$

Observe that the order of the projection matrices here is arbitrary because of the orthogonality of the applied vectors. In fact, the identites

$$P_i P_{i-1} \dots P_1 = \prod_{j=1}^{i} \left(I - \frac{q_j q_j^T}{q_j^T q_j} \right) = I - \sum_{j=1}^{i} \frac{q_j q_j^T}{q_j^T q_j} , \tag{6.5}$$

hold, the proof of which is left to an exercise. It shows that there are two possibilities numerically for performing orthogonalisation. In the first one, we use the summation formula of the above equation. Then all q_j will form a scalar product with vector a_{i+1} and (6.4), (6.5) will lead to:

$$q_{i+1} = a_{i+1} - \sum_{j=1}^{i} \frac{q_j q_j^T a_{i+1}}{q_j^T q_j}, \quad \rightarrow \quad a_{i+1} = q_{i+1} + \sum_{j=1}^{i} \frac{q_j q_j^T a_{i+1}}{q_j^T q_j}, \tag{6.6}$$

that is, every a_{i+1} can be expanded with the aid of the orthogonal vectors, where the expansion coefficients are

$$r_{j,i+1} = \frac{q_j^T a_{i+1}}{q_j^T q_j}. \tag{6.7}$$

3If we apply in (6.4) the matrix product form, then we compute the following series of vectors:

$$z_1 = a_{i+1},\ z_2 = P_1 z_1,\ \dots\ ,\ z_{j+1} = P_j z_j,\ q_{i+1} = z_{i+1}.$$

Expanding the projection matrices leads to the different formula

$$r_{j,i+1} = \frac{q_j^T z_j}{q_j^T q_j}. \tag{6.8}$$

The first approach with summation is called the *classic Gram-Schmidt (GS) orthogonalisation,* and the second one with the matrix product form is said the *modified Gram-Schmidt orthogonalisation.* Björck (1964) had shown the modified GS method has better numerical properties. According to more recent results, both methods are equally good if orthogonalization is done twice at all steps. Then the resulting normed vectors are orthogonal to machine accuracy.

T6.1 Theorem, QR-decomposition

Assume $A \in \mathbb{R}^{m \times n}$, where the columns of A are linearly independent. Then A can always be decomposed as

$$A = QR, \tag{6.9}$$

where the columns of Q are mutually orthogonal and R is an upper triangular matrix. The columns of Q and R can be built up recursively beginning with the first columns.

Proof. The condition $n \le m$ is necessary, otherwise the columns of A may not be linearly independent. Assume matrix A is composed of the column vectors $a_1, a_2, \dots, a_n$ and apply the GS orthogonalisation given in the previous Section. Comparing (6.6) and (6.7), we find that:

$$a_{i+1} = \sum_{j=1}^{i+1} q_j r_{j,i+1}, \quad \text{where } r_{i+1,i+1} = 1.$$

Having matrices $A = [a_1, a_2, \dots, a_n]$, $Q = [q_1, q_2, \dots, q_n]$ this is nothing else than the $i+1$-st column of (6.9), where $R = [r_{ij}]$. In GS orthogonalisation the elements r_{ij}, $i > j$ were not defined. They are not needed anyway, such that taking them zero, (6.9) holds exactly. ■

E6.1 Example for QR-decomposition

The non-normalized version of *QR*-decomposition will be done for

$$A=\begin{bmatrix}1 & 1 & -1\\ 2 & 1 & 3\\ 1 & 1 & -2\\ 2 & 1 & 1\end{bmatrix}=\begin{bmatrix}a_1 & a_2 & a_3\end{bmatrix}.$$

At start $q_1 = a_1$ and $q_1^T q_1 = 10$. For the next vector $q_1^T a_2 = 6$ and

$$q_2 = a_2 - q_1\frac{q_1^T a_2}{q_1^T q_1} = \begin{bmatrix}1\\1\\1\\1\end{bmatrix} - \frac{6}{10}\begin{bmatrix}1\\2\\1\\2\end{bmatrix} = \frac{1}{5}\begin{bmatrix}2\\-1\\2\\-1\end{bmatrix}.$$

The next divisor is $q_2^T q_2 = \frac{10}{25} = \frac{2}{5}$, but this result can also be computed observing that $q_2^T q_2 = \left(a_2^T - \frac{q_1^T a_2}{q_1^T q_1} q_1^T\right)\left(a_2 - q_1\frac{q_1^T a_2}{q_1^T q_1}\right) = a_2^T a_2 - \frac{\left(q_1^T a_2\right)^2}{q_1^T q_1} = 4 - \frac{36}{10} = \frac{2}{5}$. For the third vector $q_1^T a_3 = 5$ and $q_2^T a_3 = -2$, with these the third vector and the QR-decomposition are:

$$q_3 = a_3 - q_1\frac{q_1^T a_3}{q_1^T q_1} - q_2\frac{q_2^T a_3}{q_2^T q_2} = \begin{bmatrix}-1\\3\\-2\\1\end{bmatrix} - \frac{1}{2}\begin{bmatrix}1\\2\\1\\2\end{bmatrix} + \frac{2\cdot 5}{2\cdot 5}\begin{bmatrix}2\\-1\\2\\-1\end{bmatrix} = \frac{1}{2}\begin{bmatrix}1\\2\\-1\\-2\end{bmatrix},$$

$$A=\begin{pmatrix}1 & 2/5 & 1/2\\ 2 & -1/5 & 1\\ 1 & 2/5 & -1/2\\ 2 & -1/5 & -1\end{pmatrix}\begin{pmatrix}1 & 3/5 & 1/2\\ 0 & 1 & -5\\ 0 & 0 & 1\end{pmatrix}=\begin{pmatrix}1 & 2 & 1\\ 2 & -1 & 2\\ 1 & 2 & -1\\ 2 & -1 & -2\end{pmatrix}\begin{pmatrix}1 & 3/5 & 1/2\\ 0 & 1/5 & -1\\ 0 & 0 & 1/2\end{pmatrix}.$$

1.45. The Arnoldi method

It is GS orthogonalisation for the vectors of the *Krylov-base*. The vectors of the *Krylov-base* are $x, Ax, A^2x, \ldots$, where $x \neq 0$, otherwise arbitrary starting vector. Then the first vector in the Arnoldi method is $q_1 = x/\|x\|_2$, and the next vector q_2 is generated by orthogonalizing Aq_1 to q_1. In general, q_{i+1} comes from the orthogonalisation of Aq_i to vectors $q_j, j = 1,2,\ldots,i$, finally the result is normed to 1. It can be shown that vectors q_j will span the same subspace than that of the vectors in the Krylov-base. This procedure will lead to the QR-decomposition:

$$\begin{bmatrix}q_1 & Aq_1 & Aq_2 & \ldots & Aq_i\end{bmatrix} = \begin{bmatrix}q_1 & q_2 & \ldots & q_{i+1}\end{bmatrix}R, \qquad (6.10)$$

where R is an upper triangular matrix. Usually the first q_1 vector is omitted in this scheme on the left side. That means to leave out the first column of R on the right side. But in order to have a square matrix in place of R, its last row will be written separately. Denote the remainder matrix by H. Then after collecting vectors q_j into matrix Q, we get

$$Q = \begin{bmatrix} q_1 & q_2 & \dots & q_i \end{bmatrix}, \quad AQ = QH + h_{i+1,i} q_{i+1} e_i^T , \tag{6.11}$$

where H is called an *upper Hessenberg* matrix. Such matrices are close to triangular matrices with the difference that subdiagonal elements are also not zeros. Multiplying the system by e_i from the right, we get a recursion for the computation of q_{i+1}:

$$Aq_i = \sum_{j=1}^{i} h_{ji} q_j + h_{i+1,i} q_{i+1}, \quad h_{ji} = q_j^T A q_i , \tag{6.12}$$

where $h_{i+1,i}$ comes from the condition that q_{i+1} is normed. If $h_{i+1,i} = 0$ holds, then the recursion breaks down. In case of $i < n$, the columns of Q will then give an invariant subspace of A.

1.46. QR -decomposition of A with Householder reflections

We have seen in Problems 3.13 and 3.14 that elementary reflections can be used to reflect two vectors of the same length into each other. It will be shown here that QR-decomposition can also be realized by the series of orthogonal reflections. Such an elementary reflection can be given by the matrix

$$R(r-s) = I - 2\frac{(r-s)(r-s)^T}{(r-s)^T(r-s)} = I - \frac{(r-s)(r-s)^T}{\|r\|_2^2 - r^T s} \tag{6.13}$$

where $\|r\|_2 = \|s\|_2$, $R(r-s)s = r$ and $R(r-s)r = s$. According to Householder's idea, the first column a_1 of A is reflected into $\sigma_1 e_1$ in the first step, where $|\sigma_1| = \|a_1\|_2$. In order to avoid cancellation, the sign of σ_1 is chosen such that in the denominator of $R(a_1 - \sigma_1 e_1)$, the number $\sigma_1 - e_1^T a_1 = \sigma_1 - a_{11}$ be larger, that is, $\sigma_1 = -\text{sign}(a_{11})\|a_1\|_2$ holds. If eventually a_1 is zero then there is no reflection.

The *algorithm*: The first column of $A_2 = R(Ae_1 - \sigma_1 e_1)A$ after the first step is $\sigma_1 e_1$. Next reflect $\text{tril}(A_2)e_2$ into – $\sigma_2 e_2$ in a similar way. In other words, we do the same as in the first step, but for the right lower block only. The tril function was introduced in Sect. 3.9. Now the first two columns of the upper triangular matrix is ready. Generally in the i-th step the reflection for A_i can be given by

$$A_{i+1} = R\left(\text{tril}(A_i)e_i - \sigma_i e_i\right) A_i , \tag{6.14}$$

where $\text{tril}(A_i)e_i$ is reflected into $\sigma_i e_i$. Denote by Q^T the product of the elementary reflections – the product of orthogonal matrices is also orthogonal – then one gets

$$Q^T A = R . \tag{6.15}$$

Now $R \in \mathbb{R}^{n,k}$ is an upper triangular matrix, where the diagonal elements are not necessarily positive, moreover, in case of $k < n$, there is a zero block below the upper triangular part. Observe that now $Q \in \mathbb{R}^{n,n}$, that is, the columns of Q form a base .

1.47. QR -decomposition with 2×2 rotations

It is known that the counterclockwise rotation in the plane with an angle α can be given by the matrix

$$\begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix},$$

which is also orthogonal. Now all vectors in the plane can be rotated into the direction of the x-axis, in other words, in the direction of vector $\begin{bmatrix}1 & 0\end{bmatrix}^T$. If we have vector $x^T = [x_1 \quad x_2]$, then

$$\begin{aligned} F &= \frac{1}{\sqrt{x_1^2+x_2^2}}\begin{bmatrix} x_1 & x_2 \\ -x_2 & x_1 \end{bmatrix} = \frac{1}{\sqrt{1+(x_2/x_1)^2}}\begin{bmatrix} 1 & x_2/x_1 \\ -x_2/x_1 & 1 \end{bmatrix} \\ &= \frac{1}{\sqrt{1+(x_1/x_2)^2}}\begin{bmatrix} x_1/x_2 & 1 \\ -1 & x_1/x_2 \end{bmatrix} \end{aligned} \qquad (6.16)$$

Is such a matrix. We have given more options. The first one is the simplest for checking that the rotation is right. The other two options can be applied in numerical computations. The choice is that the ratio in the formulae be less that 1 There exist arrangements in numerical algorithms, where square root multipliers are collected in a diagonal matrix in order to have a better operation count.

Applying more 2×2-es rotations, it is possible to bring an n-dimensional vector into the scalar multiple of e_1. Now the 2×2 rotation matrix is imbedded in a unit matrix of order n. Let $F(i,j)$ be such that the elements in the positions ii, ij, ji, jj are overwritten by the elements of the 2×2 rotation matrix. Then the series of rotations can be given so that

$$F(1,2)F(1,3)\ldots F(1,n)x = \|x\|_2\, e_1 \qquad (6.17)$$

holds. In this way it is possible to bring $A\in\mathbb{R}^{n\times k}$ into an upper triangular matrix as it was seen for Householder reflections.

1.48. Problems

6.1. Prove formula (6.5)!

6.2. Show that $r_{j,i+1}$'s of (6.7) and (6.8) are equal!

6.3. Collect the orthogonal vectors into matrix $Q=[q_1 q_2 \ldots q_i]$. Derive the formula: $P_i P_{i-1}\ldots P_1 = \; = I - Q\left(Q^T Q\right)^{-1} Q^T$.

6.4. Let matrix $A\in\mathbb{R}^{m\times n}$ have linearly independent columns. Check that $I - A\left(A^T A\right)^{-1} A^T$ is also a projection and applying it to a vector, the resulting vector will be orthogonal to all columns of A.

6.5. One can elaborate the variant of GS orthogonalisation, when the q_j's are normed vectors, $\|q_j\|_2 = 1$. Rewrite formulas for that case!

6.6. Having a QR-decomposition of A, how can we solve the linear system $Ax=b$?

6.7. Make the QR-decomposition of $\begin{bmatrix} 2 & 6 & 5 \\ -1 & -4 & 1 \\ -1 & -2 & -3 \end{bmatrix}$.

6.8. Let the starting vector x be the sum of three eigenvectors of A having different eigenvalues. How many new vectors can be generated by the Arnoldi method?

The eigenproblem of matrices

In this problem we are looking for an *eigentriple* (λ, y, x), for which

$$Ax = \lambda x, \qquad y^T A = \lambda y^T \tag{7.1}$$

hold, where λ is the *eigenvalue* of matrix $A \in \mathbb{R}^{n \times n}$, x is the *right* and y is the *left eigenvector*. The eigenvalues are the roots of the *characteristic polynomial* $|\lambda I - A|$ and $\lambda I - A$ is singular if λ is an eigenvalue. It can be seen from the determinant that the similarity transform of the matrix has the same characteristic polynomial: $|\lambda I - S^{-1}AS| = |S^{-1}(\lambda I - A)S| = |S^{-1}||\lambda I - A||S| = |\lambda I - A|$, therefore the eigenvalues are invariant under the similarity transform.

We usually consider real matrices. But real matrices may also have complex eigenvalues and eigenvectors, so that complex cases should also be considered.

1.49. Some properties

Below we recall some basic knowledge on eigenproblems.

T7.1 Existence of eigenvectors

To each eigenvalue λ_i, there exist at least one left and one right eigenverctor.

It is because the null spaces of $\lambda_i I - A$ and $\lambda_i I - A^T$ have at least dimension 1 due to singularity of the matrices. ■

T7.2 Linear independence

The eigenvectors belonging to different eigenvalues are linearly independent.

If having an eigenvector u, the Rayleigh quotient

$$H(u) = \frac{u^H A u}{u^H u} = \lambda \tag{7.2}$$

returns the belonging eigenvalue. Observe that multiplying the eigenvector with a nonzero scalar does not change the eigenvalue. Now assume indirectly that there are two different eigenvalues and they have eigenvectors which are linearly dependent. But then they have the same direction such that the Rayleigh quotient should return the same eigenvalue and that is contradiction. Having one eigenvector, now one concludes that it is linearly independent from all other eigenvectors of different eigenvalues and that is true for all eigenvectors. ■

T7.3 Orthogonality of left and right eigenvectors

Let v_i be the left eigenvector to λ_i and let u_j be the right eigenvector to λ_j, $i \neq j$. Then $v_i^T u_j = 0$ holds.

Proof. Consider the following equations: $v_i^T A u_j = \lambda_i v_i^T u_j = \lambda_j v_i^T u_j$, where the left and right eigenvector relations were applied. As a consequence $(\lambda_i - \lambda_j) v_i^T u_j = 0$ holds, and the statement follows because of $\lambda_i \neq \lambda_j$. ■

C7.1 Corollary

If all eigenvalues are different, then arranging the eigenvectors into matrices as $X = [x_1 x_2 \dots x_n]$ and $Y = [y_1 y_2 \dots y_n]$, we have the following relations:

$$AX = X\Lambda, \quad Y^H A = \Lambda Y^H. \tag{7.3}$$

Because of linear independence of the columns, X and Y are invertible, and $X^{-1}AX = \Lambda = Y^H A Y^{-H}$ hold, where the inverse of the adjungate is denoted by $-H$ in the exponent. We have $DY^H = X^{-1}$, where D is a nonsingular diagonal matrix for scaling the length of the y_i vectors. Without restriction of generality, we may write: $Y^T = X^{-1}$, as the length of the eigenvectors is arbitrary. The form given in (7.3) shows that now the matrix can be brought to diagonal form by similarity transformation.

T7.4 Schur's theorem

Square matrices can be brought to upper triangular forms by unitary similarity transform.

Proof. Denote by $R(u) = I - 2uu^H / u^H u$ a Householder reflection matrix. Let $x \neq e_1$ be a normed eigenvector: $\|x\|_2 = 1$, that can be scaled so that the first element is a real non-positive number. (It can always be done if the vector is multiplied with the nonzero number $-\overline{x}_1 / |x_1|$.) Then e_1 and x are reflected into each other: $R(x - e_1)e_1 = x$ and $R(x - e_1)x = e_1$. Assuming $Ax = \lambda x$, we get

$$R(x - e_1)AR(x - e_1)e_1 = R(x - e_1)Ax = R(x - e_1)\lambda x = \lambda e_1.$$

The matrix $R(x - e_1)$ is involutory (its inverse is the same), hence a unitary similarity transform was done, where the first column went into λe_1. In other words, the upper triangular form appeared in the first row and column. Continuing the procedure for the right lower blocks, finally we end up with the desired triangular form. ■

Remarks. If $x = e_1$, then the first column of A already has the λe_1 form. The scaling applied for x ensures that the reflection matrix exists. Having an eigenvector at hand, the above method also serves as a *method of deflation*, because we get a matrix of size 1 less in the right lower corner, that has the remaining eigenvalues of the starting matrix.

With the help of Schur's theorem, some further important statements can be proved.

T7.5 Theorem, diagonalisability with unitary similarity transform

Matrix A is normal $\Leftrightarrow$ A can be diagonalised by a unitary similarity transform.

Proof. Recall that $A \in \mathbb{C}^{n \times n}$ is said *normal*, if $AA^H = A^H A$ holds.

$\Leftarrow$: Assume $A = U\Lambda U^H$, then $AA^H = U\Lambda U^H U\overline{\Lambda} U^H = U\Lambda\overline{\Lambda} U^H = U\overline{\Lambda} U^H U\Lambda U^H = A^H A$.

$\Rightarrow$: It is easy to check: if A is normal, then any unitary similarity transform of it is also normal. From Schur's theorem let $B=U^H AU$ be upper triangular, then normality is preserved: $BB^H = B^H B$. Now consider here the element in the 1,1 position:

$$e_1^T BB^H e_1 = \left\|B^H e_1\right\|_2^2 = \sum_{j=1}^n \left|b_{1j}\right|^2 = e_1^T B^H Be_1 = \left\|Be_1\right\|_2^2 = \left|b_{11}\right|^2,$$

That is, the 2-norms of the first row and first column in matrix B are equal. But this is possible only if $b_{1j}=0,\ j=2,\ldots,n$ hold. Continuing the procedure for the one less right lower block, we arrive to a diagonal matrix B. ■

A corollary of the theorem is that real symmetric and complex Hermitian matrices can be diagonalized by orthogonal or unitary similarity transform respectively. It is straightforward by taking the complex conjugate of the Rayleigh quotient to check that the eigenvalues of such matrices are always real.

T7.6 Theorem

Let y and x be the left and right eigenvectors belonging to a single eigenvalue. Then their scalar product must be nonzero: $y^H x \neq 0$.

Proof. Bring the matrix by unitary similarity transform to upper triangular form: $B=U^H AU$. Then the eigenvectors go into vectors $y \to U^H y$ and $x \to U^H x$, from where it is seen that their scalar product remains the same. Without restricting generality, assume that B has the form

$$B=\begin{bmatrix} \lambda & b^T \\ 0 & C \end{bmatrix},$$

where λ is the eigenvalue to eigenvectors x and y. After having been transformed, x goes into e_1, and for the left eigenvector choose the form $y^H U = \begin{bmatrix} \bar{\eta} & y_2^H \end{bmatrix}$. If multiplying B with this from the left, then the eigenvalue equation is

$$\begin{bmatrix} \bar{\eta} & y_2^H \end{bmatrix} \begin{bmatrix} \lambda & b^T \\ 0 & C \end{bmatrix} = \begin{bmatrix} \bar{\eta}\lambda & \bar{\eta} b^T + y_2^H C \end{bmatrix} = \lambda \begin{bmatrix} \bar{\eta} & y_2^H \end{bmatrix},$$

from where $\bar{\eta} b^T + y_2^H C = \lambda y_2^H$. If $\bar{\eta} = y^H x$ would be zero, then the right lower block C would have λ as an eigenvalue. And that contradicts the fact that λ is a single eigenvalue. ■

D7.1 Jordan-blocks

The matrix of the form

$$J(\mu)=\begin{bmatrix} \mu & 1 & & \\ & \mu & \ddots & \\ & & \ddots & 1 \\ & & & \mu \end{bmatrix} \in \mathbb{C}^{k\times k} \qquad (7.4)$$

is called a *Jordan block*. It is the protptype of matrices that is not diagonalizable by similarity transform. Its characteristic polynomial can easily be found as $|\lambda I - J(\mu)| = (\lambda - \mu)^k$, that is, $\lambda = \mu$ is a root of multiplicity k. The characteristic matrix $\lambda I - J(\mu)$ has rank loss only 1 at $\lambda = \mu$, because the subdeterminant belonging to the left lower corner element is just the product of -1's in the superdiagonal, therefore there exist a full rank submatrix of order $k-1$. As a consequence, there exists only one right and left eigenvector e_1 and e_k^T, such that their scalar product is 0, if $k > 1$.

The multiplicity of the eigenvalue in the characteristic polynomial is said *algebraic multiplicity* and it is denoted by m_A. The eigenvectors belonging to an eigenvalue form a subspace. That subspace is called the *eigenspace* and its dimension is said the *geometric multiplicity* m_G of the eigenvalue. The Jordan block above has $m_A = k$ and $m_G = 1$.

It is mentioned here without proof that every matrix can be brought to *Jordan canonical form* by similarity transform, where one or more Jordan blocks belong to each eigenvalue and these Jordan blocks are in the diagonal. A Jordan block may have size 1, for instance, single eigenvalues have 1×1 blocks. Now it is easy to see that the characteristic matrix has rank loss m_G for an eigenvalue and it is equal to the number of Jordan blocks. But the algebraic multiplicity m_A is equal to the sum of the belonging Jordan block sizes.

1.50. Localization of eigenvalaues

Even real matrices may have complex eigenvalues. Because of that one has to give domains in the complex plane, where eigenvalues may be located. We have already seen such an estimate in Sec. 2.8 with respect to matrix norms. According to that the spectral radius may not be larger than any norm of the matrix. So that no eigenvalue may be larger in absolute evalue than $\|A\|_1$ vagy $\|A\|_\infty$. But even more accurate estimate is possible with the aid of

T7.7 Gershgorin's theorem

Let K_i denote the i-th Gershgorin circle. Its center is located in the complex plane at a_{ii} having radius $r_i = \|e_i^T(A - a_{ii}I)\|_\infty$, that is the absolute sum of the i-th row elements with the exception of the diagonal element. According to Gershgorin's theorem, the eigenvalues of matrix A lie in the united set of Gershgorin circles.

Proof. Consider the i-th row of equation $Ax = \lambda x$, where x is an eigenvector with eigenvalue λ and $|x_i| = \|x\|_\infty$. After rearrangement

$$\lambda - a_{ii} = \sum_{j=1,\, j\neq i}^{n} \frac{a_{ij}x_j}{x_i},$$

from where

$$|\lambda - a_{ii}| \le \sum_{j=1,\, j\neq i}^{n} \left| \frac{a_{ij}x_j}{x_i} \right| \le \sum_{j=1,\, j\neq i}^{n} |a_{ij}| = r_i .$$

We can write similar relations for each eigenvalue and that gives the statement. ■

T7.8 Gershgorin's second theorem

If there exists a disjunct subset of the Gershgorin circles, then one can find as many eigenvalues in this subset as is the number of Gershgorin circles.

Proof. We apply the fact here that the eigenvalues of the matrix are continouos functions of the matrix elements. Now split matrix into two parts and define matrix $A(\varepsilon) = D + \varepsilon A_1$, where D is the diagonal part of A and A_1 is the nondiagonal part. If $\varepsilon = 0$, then all circles have zero radius. If ε is tending to 1, then all eigenvalues may leave the center, but because of continuity, they may not jump out of their own circle. ■

C7.2 Corollary

The transposed matrix A^T has the same eigenvalues and Gershgorin's theorems can also be applied. Then we get a new set of circles. If we take the intersection of this set with the Gershgorin circles of A, then all eigenvalues should lie in this common part of circle sets.

E7.1 Example

Gershgorin's theorems may be combined with diagonal similarity transformations. With this trick we can change the radius of the circles and we can do a purposed estimate. For instance, show that matrix

$$A = \begin{bmatrix} 8 & 5 & 3 \\ 1 & 4 & 1 \\ 1 & 2 & 5 \end{bmatrix}$$

has no zero eigenvalue!

The first Gershgorin's circle has center 8, radius 8, such that it contains zero. The other circles not. Apply the diagonal similarity transform $D^{-1}AD$, where $D = \text{diag}(2 \quad 1 \quad 1)$:

$$D^{-1}AD = \begin{bmatrix} 8 & 5/2 & 3/2 \\ 2 & 4 & 1 \\ 2 & 2 & 5 \end{bmatrix}.$$

Now the radius of the first circle is diminished to 4 and the other two circles still do not contain zero so that our aim is achieved. Observe the change in row and column if only one element is different from 1 in the diagonal matrix!

1.51. Computation of the characteristic polynomial

Consider the so-called *Frobenius companion matrix*:

$$F = \begin{bmatrix} 0 & 0 & \dots & 0 & -a_0 \\ 1 & 0 & \dots & 0 & -a_1 \\ & 1 & \ddots & \vdots & \vdots \\ & & \ddots & 0 & -a_{n-2} \\ & & & 1 & -a_{n-1} \end{bmatrix}. \qquad (7.5)$$

If expanding along the last column, one can show the formula: $\det(\lambda I - F) = \lambda^n + \sum_{i=0}^{n-1} a_i \lambda^i$. Therefore it is easy to compute the coefficients of the characteristic polynomial if we can apply a similarity transform that brings matrix A to this form. *Danilevsky* suggested Gauss-Jordan transform such that the first column of the matrix is transformed not into e_1, but into e_2. Thererfore we choose the first matrix as $T_1 = I + (Ae_1 - e_2)e_2^T$ and the result of the similarity transform $A_2 = T_1^{-1} A T_1$ will be e_2 in the first column:

$$A_2 e_1 = T_1^{-1} A T_1 e_1 = \left(I - \frac{(Ae_1 - e_2)e_2^T}{e_2^T A e_1} \right) A \left(I + (Ae_1 - e_2)e_2^T \right) e_1 = \left(I - \frac{(Ae_1 - e_2)e_2^T}{e_2^T A e_1} \right) A e_1 = e_2 .$$

Generally we apply $T_k = I + (A_k e_k - e_{k+1}) e_{k+1}^T$ in the k-th step, and the former unit vectors will not be affected because of

$$\left(I - \frac{(A_k e_k - e_{k+1}) e_{k+1}^T}{e_{k+1}^T A_k e_k} \right) A_k \left(I + (A_k e_k - e_{k+1}) e_{k+1}^T \right) e_\ell = e_{\ell+1}, \quad \ell \le k .$$

The condition for performing a step is that the subdiagonal element is not zero. Otherwise interchange the same rows and columns to move a nonzero into that position. If we find nonzeros in all steps, then we arrive to the form of (7.5) in the $n-1$-st step. If we can not find a nonzero in a column, then we turn to the next column and start a new Frobenius block. This time the result is an upper block tridiagonal matrix having Frobenius blocks in the diagonal.

If the matrix is tridiagonal, a simple recursion can be found to get the characteristic polynomial. For instance, the recursion of the determinant

$$\begin{vmatrix} \lambda - d_1 & -c_1 & & \\ -a_1 & \lambda - d_2 & \ddots & \\ & \ddots & \ddots & -c_{n-1} \\ & & -a_{n-1} & \lambda - d_n \end{vmatrix}$$

is

$$p_{i+1}(\lambda) = (\lambda - d_{i+1}) p_i(\lambda) - a_i c_i p_{i-1}(\lambda), \; p_0 = 1, \; p_1 = \lambda - d_1, \tag{7.6}$$

where $p_i(\lambda)$ is the determinant of the left upper block of order i. The determinant of $i+1$-st order can be found by expanding along the $i+1$-st column and the resulting recursion can be seen in (7.6). This recursion can also be used to compute the numerical value of the polynomial at various places.

Similar recursion can also be given for upper Hessenberg matrices. For example, choose $p_3 = 1$ and solve the following system from below:

$$\begin{bmatrix} \lambda - 2 & 1 & 3 \\ 2 & \lambda + 1 & 2 \\ 0 & 2 & \lambda - 1 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} = \begin{bmatrix} p(\lambda) \\ 0 \\ 0 \end{bmatrix} .$$

From the last row we get $p_2(\lambda) = (1-\lambda)/2$, the second row gives $2p_1(\lambda) + (\lambda+1) p_2(\lambda) + 2 = 0$, from where p_1 can be expressed. With these $p(\lambda)$ can be found from the first row. The determinant of the matrix is zero if $p(\lambda)$ is zero, therefore the roots of $p(\lambda)$ are equal to the eigevalues of the matrix. Observe that in case of $p(\lambda)=0$, the triple $p_1(\lambda), p_2(\lambda), p_3(\lambda)$ gives the elements of the eigenvector belonging to the eigenvalue λ.

T7.9 Theorem

Every matrix can be brought to an upper Hessenberg form by a unitary similarity transform.

Proof. Choose $u_1 = \text{tril}(A,-1)e_1$, $\|u_1\|_2 = |\sigma_1|$ in the first step, that is, the diagonal element is cancelled from the first column of A. The similarity transform will be done with the reflection matrix $R(u_1 - \sigma_1 e_2)$, where the sign of σ_1 is chosen such that the real part in the second element of u_1 / σ_1 is negative. Then $R(u_1 - \sigma_1 e_2)A$ maps the first column into $a_{11}e_1 + \sigma_1 e_2$, (first entry is unchanged, the other column elements were reflected into $\sigma_1 e_2$). Applying the same reflection from the right will not change the first column any more, because the first row and column of $R(u_1 - \sigma_1 e_2)$ are e_1^T and e_1. Now the first column of $A_2 = R(u_1 - \sigma_1 e_2)AR(u_1 - \sigma_1 e_2)$ already shows the Hessenberg form. In the next step we apply the same approach for the right lower block of A_2 that has size one less. Continuing the process, finally we arrive to the Hessenberg form of the whole matrix. ■

1.52. Iteration methods

1.52.1 Power iteration

This method is based on the observation, that with increasing k the component belonging to the eigenvector of the largest eigenvalue will grow up in $A^k x_0$. Convergence is stated in the next theorem:

Let A be a real or complex matrix of order n and its eigenvalues can be ordered as

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_n|.$$

Moreover, the matrix has simple structure that is, the number of eigenvectors is n and the spectral decomposition is $A = \sum_{i=1}^{n} \lambda_k u_k v_k^T$, where v_k, u_k are the left and right eigenvectors and vector x_0 can be expanded as $x_0 = \sum_{k=1}^{n} \alpha_k u_k$. Then

$$\lim_{m\to\infty} \frac{1}{\lambda_1^m} A^m x_0 = \alpha_1 u_1. \tag{7.7}$$

Proof. According to the method, we compute vectors $x_m = Ax_{m-1} = \sum_{k=1}^{n} \alpha_k \lambda_k^m u_k$ from where we have $\frac{A^m x_0}{\lambda_1^m} = \sum_{k=1}^{n} \alpha_k \left(\frac{\lambda_k}{\lambda_1} \right)^m u_k$. The statement follows by taking the limit $m \to \infty$, as the multiplier of the other vectors tend to zero. ■

It is seen, the speed of convergence is essentially given by the ratio λ_2 / λ_1 in case of $\alpha_1 \neq 0$. The algorithm is:

$$\begin{aligned} &m = 1, 2, \ldots - re: \\ &y_{m+1} = Ax_m, \\ &x_{m+1} = \frac{y_{m+1}}{\|y_{m+1}\|}. \end{aligned}$$

A practical choice for the norm is the infinity norm. The eigenvalue can be estimated by the Rayleigh-quotient as

$$\lambda_1^{(m)} = \frac{x_m^T y_{m+1}}{x_m^T x_m} = \frac{x_m^T A x_m}{x_m^T x_m}.$$

The power iteration is applicable for finding eigenvalues at the ends of the *spectrum* (= the set of eigenvalues). But we can also find eigenvalues within the spectrum by *inverse power iteration*. This time the vector

$$x_{m+1} = (\lambda I - A)^{-1} x_m$$

is computed in one step of the iteration. The eigenvalues of $(\lambda I - A)^{-1}$ are $1/(\lambda - \lambda_k),\ k = 1, \ldots, k$. As seen, we have another power iteration that converges to the closest eigenvalue to λ and the belonging eigenvector.

1.52.2 The *QR*-algorithm

For solving eigenproblems, one of the best methods is considered to be the *QR*-algorithm. It begins with forming a *QR*-decomposition of $A = Q_1 R_1$. The next matrix is $A_2 = R_1 Q_1 = Q_1^T A Q_1$, that is the result of an orthogonal similarity transform and in general the k-th matrix is $A_k = R_{k-1} Q_{k-1}$. If the matrix has a simple structure (all eigenvalues are simple) and the modal matrix (the matrix of eigenvectors) has an LU-decomposition, then it can be shown that the QR-algorithm converges to an upper triangular matrix. The convergence can further be accelerated if the decompositions are combined with a shift by $-\kappa I$, where κ is an estimate of an eigenvalue. In that case the QR-algorithm has convergence of second order and it is even faster – of third order – for symmetric matrices.

1.53. Some inequalities related to eigenproblems

We have already seen such relations when discussing Gershgorin-discs. Here we continue our investigations and we are interested in characterizing the goodness of an approximate eigenpair (λ, u). Another problem is to give the variation of an eigenpair if the matrix elements are changed a little – i.e. perturbed.

We shall apply the following notations: $M = \max\limits_{(i)} |\lambda_i(A)|,\quad m = \min\limits_{(i)} |\lambda_i(A)|$ and assume that the matrix is invertible. Always induced matrix norms will be used here.

From Theorem T2.4 on the spectral radius, the relations $M \le \|A\|,\ 1/m \le \|A^{-1}\|$ hold and multiplying the same sides gives:

$$\frac{M}{m} \le \operatorname{cond}(A) = \|A\| \|A^{-1}\|. \tag{7.8}$$

This relation shows that the condition number of the matrix is large if the ratio of the absolute largest and smallest eigenvalues is large.

L7.1 Lemma

Let $D = \mathrm{diag}(d_1,\ldots,d_n)$ be a diagonal matrix, then $\|D\|_p = \max\limits_{(i)} |d_i|,\ \ 1 \le p \le \infty$.

Proof. Assume $|d_k| \ge |d_i|$ for all i-s. From the definition of induced norms it follows that

$$\|D\|_p^p = \sup_{(x\neq 0)} \frac{\sum_{i=1}^n |d_i x_i|^p}{\sum_{i=1}^n |x_i|^p} = |d_k|^p \sup_{(x\neq 0)} \frac{\sum_{i=1}^n |x_i d_i / d_k|^p}{\sum_{i=1}^n |x_i|^p} = |d_k|^p ,$$

because the denominator is larger than the numerator if there exists a nonzero x_i, for which $|d_i / d_k| < 1$. ■

T7.10 Theorem on the goodness of the eigenpair

Assume A has simple structure: $AU = U\Lambda$, where U is the matrix of eigenvectors and the diagonal matrix Λ has the eigenvalues, moreover (λ, x) is an approximate eigenpair. Then using the notation $r = Ax - \lambda x$, we have the inequality

$$\min_{(i)} |\lambda_i - \lambda| \le \frac{\|r\|}{\|x\|} \mathrm{cond}(U), \quad \|\ .\ \|\ p\text{-norm} \tag{7.9}$$

Proof. If $\lambda_i = \lambda$ holds for some i, then the statement is true. Now choose $\lambda_i \neq \lambda$ such that $A - \lambda I$ is invertible then we may write $x = (A - \lambda I)^{-1} r = U(\Lambda - \lambda I)^{-1} U^{-1} r$. After taking the norms and applying the previous lemma, we have

$$\|x\| \le \frac{\mathrm{cond}(U)}{\min\limits_i |\lambda_i - \lambda|} \|r\|$$

that leads to the statement by reordering. ■

C7.3 Corollary

In the case of Hermitian matrices U is unitary, hence $\mathrm{cond}_2(U) = 1$ and $\min\limits_{(i)} |\lambda_i - \lambda| \le \|r\|_2 / \|x\|_2$ follows, an easily computable quantity.

T7.11 Theorem, lower bounds for cond(U)

If A is invertible and has simple structure, we have

$$\frac{\|A\|}{M} \le \mathrm{cond}(U), \quad \|A^{-1}\| m \le \mathrm{cond}(U), \quad \sqrt{\frac{\mathrm{cond}(A)}{\mathrm{cond}(\Lambda)}} \le \mathrm{cond}(U). \tag{7.10}$$

Proof. The third relation comes by multiplying the corresponding sides of the first two. The first inequality comes by taking the norm of $A = U\Lambda U^{-1}$, and the second one follows from $A^{-1} = U\Lambda^{-1}U^{-1}$ similarly. ■

T7.12 Theorem, (Bauer, Fike)

Let A have simple structure and let E be a square matrix of the same size. If μ is an eigenvalue of $A + E \in \mathbb{C}^{nxn}$ and $AU = U\Lambda$ then

$$\min_{(i)} |\lambda_i - \mu| \le \|E\|_p \operatorname{cond}_p(U) . \tag{7.11}$$

Proof. Suppose $\mu \notin \{\lambda_i(A)\}$, otherwise the statement is true. As μ is an eigenvalue, it follows that $U^{-1}(A+E-\mu I)U = \Lambda - \mu I + U^{-1}EU$ is singular, which can still be reordered into $I+(\Lambda-\mu I)^{-1}U^{-1}EU$. But this last matrix may be singular only if $(\Lambda-\mu I)^{-1}U^{-1}EU$ has an eigenvalue with absolute value 1. Applying Theorem T2.4 on the spectral radius results in the inequality $1 \le \left\|(\Lambda-\mu I)^{-1}U^{-1}EU\right\|_p \le \left\|(\Lambda-\mu I)^{-1}\right\|_p \|E\|_p \operatorname{cond}_p(U)$. The statement follows from here by using Lemma L7.1 and moving the eigenvalue multiplier on the other side. ■

T7.13 Theorem, inverse perturbation

Let (λ, x) be an approximate eigenpair and introduce $r = Ax - \lambda x$. Then adding matrix $E = -\frac{rx^H}{\|x\|_2^2}$, $\|E\|_2 = \frac{\|r\|_p}{\|x\|_p}$, $p = 2, F$ (F indicates Frobenius-norm) to A, the eigenpair (λ, x) is an accurate solution:

$$(A+E)x = \lambda x . \tag{7.12}$$

Proof. $\left(A - \frac{rx^H}{x^H x}\right)x = Ax - r = \lambda x$. ■

For instance, if $\|r\|_2 / \|x\|_2 \approx 10^{-9}$ holds and A has elements around 1, then (λ, x) is an accurate solution of a problem, where the matrix differs from A only in the ninth figure. If A has elements accurate for 7 figures, then there is no point to continue iteration for a more accurate solution.

T7.14 Theorem, perturbation of a single eigenvalue

Let (λ, x, y) be an eigentriple that belongs to A and λ is a single eigenvalue. Then the first order change in the eigenvalue of $A+E$ is

$$\tilde{\lambda} = \lambda + \frac{y^T E x}{y^T x} + \mathcal{O}(\|E\|_2^2) . \tag{7.13}$$

Consequently, we have

$$\left|\tilde{\lambda} - \lambda\right| \le \frac{\|y\|_2 \|x\|_2}{\left|y^T x\right|} \|E\|_2 + \mathcal{O}(\|E\|_2^2) . \tag{7.14}$$

Proof. The second relation follows form the first one by taking norms. To prove the first one, let μ denote the change in the eigenvalue, and h in the eigenvector:

$$(A+E)(x+h) = (\lambda+\mu)(x+h) .$$

Assume that in case of $E \to 0$ the change will also tend to zero: $\mu \to 0$ és $h \to 0$. After performing multiplications, omit second order terms:

$$Ah + Ex \approx \lambda h + \mu x .$$

If multiplying with the left eigenvector y^T from the left, the contribution of the first vectors will cancel from both sides of the equation and the first statement follows:

$$\mu \approx \frac{y^T E x}{y^T x}. \tag{7.15}$$

Here the divisor may not be zero because of Theorem T7.6. The multiplier of $\|E\|_2$ in (7.14) is nothing else then the reciprocal cosine of the included angle between vectors x and y: $\sec\angle(x,y) = \|x\|_2 \|y\|_2 / |y^T x| y$. This value is usually said the *condition number of the eigenvalue* λ. ■

1.54. Singular value decomposition

T7.14 Theorem, singular value decomposition of matrices

Let $A \in \mathbb{C}^{m\times n}$. Then there exist unitary matrices $U \in \mathbb{C}^{n\times n}$ and $V \in \mathbb{C}^{m\times m}$, for which

$$V^H A U = \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix}, \quad \Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_r) \tag{7.16}$$

holds, where $\sigma_1 \ge \sigma_2 \ge \ldots \ge \sigma_r > 0$ are the nonzero singular values of the matrix, (V^H denotes the conjugate transpose of the matrix).

Proof. Matrix $A^H A$ is positive semidefinite that is, all eigenvalues are nonnegative. Let the nonzero eigenvalues be ordered as $\sigma_1^2 \ge \sigma_2^2 \ge \ldots \ge \sigma_r^2 > 0$. Then there exist unitary matrix U, such that

$$U^H A^H A U = \begin{pmatrix} \Sigma^2 & 0 \\ 0 & 0 \end{pmatrix}.$$

The columns of matrix U are the eigenvectors. Partition U into two parts: $U = \begin{pmatrix} U_1 & U_2 \end{pmatrix}$, where U_1 has the columns with nonzero eigenvalues and the columns of U_2 have zero eigenvalues. Then $AU_2 = 0$ and

$$U_1^H A^H A U_1 = \Sigma^2 \;\rightarrow\; \Sigma^{-1} U_1^H A^H A U_1 \Sigma^{-1} = I.$$

Now if introducing matrix $V_1 = AU_1\Sigma^{-1}$, the last equality here shows $V_1^H V_1 = I$ that is, the columns of V_1 are orthonormal vectors. Complement V_1 with matrix V_2 such that $V = \begin{pmatrix} V_1 & V_2 \end{pmatrix}$ is unitary: $V^H V = I$, then

$$V^H A U = \begin{pmatrix} V_1^H A U_1 & V_1^H A U_2 \\ V_2^H A U_1 & V_2^H A U_2 \end{pmatrix} = \begin{pmatrix} V_1^H V_1 \Sigma & 0 \\ V_2^H V_1 \Sigma & 0 \end{pmatrix} = \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix},$$

where we have used the facts that $AU_2 = 0$ and $V_2^H V_1 = 0$. ■

1.55. Problems

7.1. Let A be an upper Hessenberg matrix such that all subdiagonal elements are nonzero. Show that there is only one Jordan block to each eigenvalue.

7.2. Show if the eigenvalues of A are λ_i's, then A^{-1} has eigenvalues $1/\lambda_i$'s.

7.3. Check: $\|A\|_2 = \sigma_1$. $\|A\|_F = \left(\sum_{i=1}^{r} \sigma_i^2\right)^{1/2}$.

7.4. The matrix is diagonally dominant, if the Gershgorin disks do not have zero.

7.5. Diagonally dominant matrices are invertible.

7.6. Interchanging the i-th and j-th rows and columns will not affect diagonal dominance.

7.7. The rank of the matrix is at least as large as the number of those Gershgorin disks, which do not contain zero.

7.8. Gershgorin disks can also be found with respect to columns if using left eigenvectors in the derivation .

7.9. By using Gershgorin's theorem and diagonal similarity transform, decide if matrix A is invertible: $A = \begin{bmatrix} 7 & 6 & -3 \\ 1 & 5 & 1 \\ 4 & -2 & 6 \end{bmatrix}$.

7.10. Show that the eigenvalues of a 2×2 matrix A are: $\lambda_{1,2} = \frac{a_{11}+a_{22}}{2} \pm \sqrt{\left(\frac{a_{11}-a_{22}}{2}\right)^2 + a_{12}a_{21}}$.

Prove that

7.11. $\|U\| \le \|AU\| / m$, where $AU = U\Lambda$.

7.12. $\|U^{-1}\| \le \|U^{-1}A^{-1}\| M$.

7.13. $\mathrm{cond}(U) \le \mathrm{cond}(AU)\,\mathrm{cond}(\Lambda)$.

Linear least squares

1.56. A practical problem of fitting functions to data.

Many times in pactice, a typical problem is the following: Given the pairs of data (t_i, y_i), $i = 0,1,\ldots,n$, where y_i is a measured value at t_i. The measured values are subject to random errors. We would like to approximate this series of points – or a part of it – with a function of the form $f(t) = \sum_{j=0}^{n} c_j \varphi_j(t)$ (e.g. $\varphi_j(t) = t^j$):

$$\sum_{j=0}^{n} c_j \varphi_j(t_i) \approx y_i . \tag{8.1}$$

To find the unknown parameters c_i, it seems reasonable to look for the minimum of the sum of squares of the deviations:

$$\sum_{i=0}^{m} (y_i - \sum_{j=0}^{n} c_j t_i^j)^2 = \min \tag{8.2}$$

that is, the linear combination coefficients c_j in (8.1) are sought by this minimization condition.

We may reformulate problem (8.1) for the unknowns c_j with the aid of a linear system of equations, so that consider

$$Ax = b,\ A \in \mathbb{R}^{m,n},\ x \in \mathbb{R}^n, b \in \mathbb{R}^m, \tag{8.3}$$

where the coefficient matrix is *rectangular* - not a square matrix - and one may not take it for sure that the system has a solution. To find the least squares solution is equivalent to finding a solution for which $\|b - Ax\|_2^2 = \min$ is fulfilled. To achieve this goal, we consider projection matrices first.

1.57. Projection matrices

We have already introduced projection matrices P in Ch. 3. They have the property $P^2 = P$. That means: the second application of P leaves vector x unchanged: $P(Px) = Px$ and it is the case at repeated applications of P, exactly what we have in our mind about projections. As any power of P is equal to itself, it is also called an *idempotent matrix.*

A projection example: Let $A \in \mathbb{R}^{m,n}$, $B \in \mathbb{R}^{m,n}$, $n < m$ and denote by ^{T} the transpose and let $B^T A$ be invertible. Then

$$P = P(A,B) = A(B^T A)^{-1} B^T$$

is a projection into the range space of A (the subspace of columns vectors of A):

$$P(A,B):\ \mathbb{R}^m \to \mathrm{Im}(A)$$

and from $(I - P)P = 0$ one has $\{I - P^T(A,B)\}z \perp \mathrm{Im}(A)$ for any vector z.

T8.1 Theorem on least angle

Let $\mathcal{P}(\mathcal{A})$ be the set of projections into subspace $\mathcal{A} \subset \mathbb{R}^m$. Then for any vector $x \notin \mathcal{A}$, $Px \neq 0$

$$\max_{P \in \mathcal{P}(\mathcal{A})} \frac{x^{\mathrm{T}} Px}{\|Px\|_2} = \|P_s x\|_2 ,$$

where P_s is the symmetric (orthogonal) projection into $\mathcal{A}$.

Proof. This theorem suggests the following picture: the angle of vector x with a vector from $\mathcal{A}$ is minimal for $P_s x$ (or for a vector in the same direction). For, let $P_s, P \in \mathcal{P}(\mathcal{A})$, then

$$P_s P = P$$

because the columns of P are in $\mathcal{A}$, and they are left unchanged by any projection onto the same subspace. Applying Cauchy's inequality yields

$$\frac{x^{\mathrm{T}} Px}{\|Px\|_2} = \frac{x^{\mathrm{T}} P_s Px}{\|Px\|_2} \leq \frac{\|P_s x\|_2 \|Px\|_2}{\|Px\|_2} = \|P_s x\|_2 , \tag{8.4}$$

where maximum is still attained for projections $\tilde{P}$ satisfying $\tilde{P}x = \lambda P_s x$, $\lambda > 0$. ■

T8.2 Theorem on uniqueness of orthogonal projections

Among projections onto the same subspace, the orthogonal (symmetric) projection is unique.

Proof. Indirect. Assume P_1 and P_2 are two different orthogonal projections onto the same subspace, then

$$P_1 P_2 = P_2 \qquad \Rightarrow \quad P_2 = P_2^{\mathrm{T}} = P_2^{\mathrm{T}} P_1^{\mathrm{T}} = P_2 P_1 = P_1$$

that is a contradiction. ■

T8.3 Theorem on distance from a subspace in two-norm

The distance of vector x from subspace $\mathcal{A}$ in two-norm can be given by:

$$\mathrm{dist}_2(x, \mathcal{A}) = \|(I - P_s)x\|_2 , \quad P_s \in \mathcal{P}(\mathcal{A}) .$$

Proof. As $P_s x$ has the smallest angle with x , the nearest point of A can be found along the direction $P_s x$. Therefore we look for a λ, for which $x - \lambda P_s x$ has minimal norm:

$$\|x - \lambda P_s x\|_2^2 = x^{\mathrm{T}} x - 2\lambda x^{\mathrm{T}} P_s x + \lambda^2 x^{\mathrm{T}} P_s x .$$

The derivative is zero at $\lambda = 1$ and after substitution, it leads to the statement.

■

Remark 1. We have for the distance vector: $(I - P_s)x \perp \mathcal{A}$.

Remark 2. One can also find the answer by taking the distance of x from a general vector $P_s x + y \in \mathcal{A}$, $y \in \mathcal{A}$. The result is $\|x - P_s x - y\|_2^2 = \|(I - P_s)x\|_2^2 + \|y\|_2^2$ because of the orthogonality of $(I - P_s)x$ and y. This expression has minimum for $y = 0$.

1.58. Generalized inverses, pseudoinverse

Generalized inverse to a matrix is introduced if ordinary inverse does not exist. Among generalized inverses the pseudoinverse has the special property that it returns a solution for the linear system for which the deviate vector – in other word: residual vector $r = b - Ax$ has minimal two-norm. If there are many minimal residual norm solutions, it returns the solution vector of smallest two-norm. We recall at first two simple facts of linear algebra.

L8.1 Lemma

Let the columns of matrix L be linearly independent. Then from $LB = LC$ the equality $B = C$ follows.

Proof. After rearranging one gets $L(B-C) = 0$. Because of the linear independence of the columns in L, all columns of $B-C$ has to be zero. ■

L8.2 Lemma

Let $A \in \mathbb{R}^{m,n}$. Then $A^T A$ is positive semidefinite. If A has linearly independent columns, i.e. A is of column rank, then $A^T A$ is positive definite.

Proof. Let $y = Ax$, then $x^T A^T Ax = y^T y \geq 0$. If A is of column rank, from $y = 0$ follows $x = 0$ by the previous lemma such that $A^T A$ is positive definite. ■

1.58.1 The pseudoinverse

We show in the following that there exists pseudoniverse or Moore-Penrose generalized inverse A^+ to every matrix A, which is the ordinary inverse if it exists. It returns the minimal residual solution (in two-norm) for the general case. Such a matrix is defined by the four Penrose conditions (complex numbers are supposed):

1. $AA^+A = A$, 2. $A^+AA^+ = A^+$,

3. AA^+ is Hermitian, 4. A^+A is Hermitian.

As we are dealing here with real matrices, we shall demand symmetricity for the last two conditions.

One can make at once simple observations. Multiply first equation by A^+ from the right or from the left and observe conditions 3 and 4. Then it follows that AA^+ and A^+A are symmetric projections. AA^+ projects into $\operatorname{Im}(A)$, and we see from $A(I - A^+A) = 0$ that $I - A^+A$ is the symmetric projection into $\operatorname{Null}(A)$, the null space of A.

D8.1 Definition, rank factorization

We say $A = LU$ is a rank factorization if $r = \operatorname{rank}(A) = \operatorname{rank}(L) = \operatorname{rank}(U)$, $L \in \mathbb{R}^{m,r}$, $U \in \mathbb{R}^{r,n}$.

Having a rank factorization LU at hand makes it possible to give the pseudoinverse.

T8.4 Theorem, construction of pseudoinverse

Let $A = LU$ be a rank factorizátion. Then the pseudoinverse is given by $A^+ = U^+L^+$ and it is unique, where $L^+ = (L^T L)^{-1} L^T$ and $U^+ = U^T (UU^T)^{-1}$.

Proof. Uniqueness can be proven indirectly. Assume there are two: A_1^+ and A_2^+. Then because of the uniqueness of symmetric projections follows $A_1^+A = A_2^+A$ and $AA_1^+ = AA_2^+$. Further applying the Penrose conditions

$$A_1^+ = A_1^+AA_1^+ = A_2^+AA_1^+ = A_2^+AA_2^+ = A_2^+$$

that is a contradiction. Next we construct the pseudoinverse.

Observe that $\mathrm{Im}(A) = \mathrm{Im}(L)$, hence the unique symmetric projection into this subspace is $AA^+ = LL^+ = L(L^TL)^{-1}L^T$ and $L^+ = (L^TL)^{-1}L^T$ follows from Lemma L 8.1. Similarly, the unique symmetric projection into $\mathrm{Im}(A^T) = \mathrm{Im}(U^T)$ can be given by $A^+A = A^T(A^+)^T = U^T(U^+)^T = U^+U = U^T(UU^T)^{-1}U$, from where $U^+ = U^T(UU^T)^{-1}$. Now it follows that $L^+L = UU^+ = I_r$, the unit matrix of order r and we can write the relations

$$AA^+ = LL^+ = LUU^+L^+ \text{ és } A^+A = U^+U = U^+L^+LU\,.$$

Both relations are true for $A^+ = U^+L^+$. ■

Remarks

If A is of column rank then $L = A$ and $U = I_n$ is an appropriate choice and $A^+ = (A^TA)^{-1}A^T$ follows. For such a matrix having $A = QR$, the pseudoinverse formula is $A^+ = R^{-1}Q^{\mathrm{T}}$. If A is of row rank, then choosing $L = I_m$ and $U = A$ yields the formula $A^+ = A^{\mathrm{T}}(AA^{\mathrm{T}})^{-1}$. If now $A^{\mathrm{T}} = QR$, then $A^+ = Q(R^{\mathrm{T}})^{-1}$. Finally, if the rank is less than the smaller size of A that is, there are linearly dependent rows or columns, then applying orthogonalisation for both sides leads to the form $A = Q_1BQ_2$, where Q_1 has orthonormal columns, Q_2 has orthonormal rows and B is full rank upper diagonal, then $A^+ = Q_2^{\mathrm{T}}(B)^{-1}Q_1^{\mathrm{T}}$ is the result. To determine rank of matrices can be a very delicate task sometimes.

T8.5 Theorem, solvability of a linear system

Let P be a projection onto $\mathrm{Im}(A)$. Then the linear sytem $Ax = b$ is solvable if and only if $Pb = b$.

Proof. Necessity. If the system is solvable, then $b \in \mathrm{Im}(A)$ and $Pb = b$ must hold. For sufficiency choose the projection AA^+ onto $\mathrm{Im}(A)$, for which $AA^+b = b$ holds. But then $x = A^+b$ is a solution. ■

T8.6 Theorem, properties of the solution with pseudoinverse

The general solution of the linear system $Ax = b$ with the pseudoinverse can be given by:

$$x = x_p + x_h = A^+b + (I - A^+A)t, \quad t \in \mathbb{R}^n, \tag{8.5}$$

where x_p is a particular solution and x_h is the general solution of the homogenous system.

If the system is solvable, then A^+b is a particular solution and $(I - A^+A)t$ is the general solution of the homogenous system. If the system is inconsistent, then A^+b is that least squares solution, for which the two-norm of $b - AA^+b$ is minimal. In every case A^+b is a least squares solution.

Proof. $\operatorname{dist}_2(b, \operatorname{Im}(A)) = \|b - AA^+b\|_2$ according to Theorem T8.2, hence A^+b is a least squares solution. Moreover, observe that the two vectors in (8.5) are orthogonal to each other, because A^+A leaves the first vector unchanged due to the pseudoinverse conditions, while it brings the second one into zero. One may write then: $\|x\|_2^2 = \|A^+b\|_2^2 + \|(I - A^+A)t\|_2^2$, that has the smallest value, if $t = 0$, or $I - A^+A = 0$. In this latter case the pseudosolution is unique. ■

1.59. Problems

8.1. Let $A = LU$ be a rank-factorisation. What is the orthogonal projection to $\operatorname{Im}(A)$?

8.2. What is the orthogonal projection to the null space of A? Give the distance of x to $\operatorname{Nul}(A)$ in two-norm!

8.3. A line passes points r_0 and r_1. Give the distance of vector x from this line!

8.4. Show that if a matrix is invertible then its inverse and pseudoinverse are equal.

8.5. $A^T = \begin{bmatrix} 2 & -4 & 6 \\ 0 & 5 & -5 \end{bmatrix}$. Give the orthogonal projection into $\operatorname{Im}(A)$!

8.6. The row vectors of A^T in Problem 8.5 are the normal vectors of two planes. Give the orthogonal projection into the intersection of the two planes!

8.7. $r^T = [1 \quad -1 \quad 1]$. What is the distance of vector r from the intersection of the two planes in the previous problem?

8.8. $A = \begin{bmatrix} 2 & 0 & 3 \\ -4 & 5 & -2 \\ 6 & -5 & 5 \end{bmatrix}$, $\operatorname{rank}(A) = 2$. $A^+ = ?$ (Use LU-decomposition!)

8.9. What is the pseudosolution of $Ax = b$ if $b^T = [1 \quad -1 \quad 1]$ and matrix A comes from Problem 8.8.

8.10. Show $I - A^+A = 0$, if the columns of A are linearly independent.

8.11. Derive the relation $(A^+)^T = (A^T)^+$ from the four Pendrose conditions.

8.12. Matrix A has an approximate eigenvector x. Find the belonging approximate eigenvalue λ from the condition that $\|Ax - \lambda x\|_2$ is minimal. Give formula for λ!

Orthogonal polynomials

Many times least squares fitting is needed for polynomials. A feasible way of solving such problems is using orthogonal polynomials. Later on such polynomials will be needed in quadrature methods such that a short introduction is given here.

1.60. Scalar product of functions

The salar product of functions f and g is defined by

$$\left(f,g\right)=\int_a^b f(x)g(x)w(x)dx,\quad \mathrm{w}(x)>0 \tag{9.1}$$

where $w(x)$ is called the weight function and assume that the integral exists. But we shall use a simpler scalar product now in connection with orthogonal polynomials, namely:

$$\left(f,g\right)=\sum_{i=1}^{m} f(x_i)g(x_i)w_i \tag{9.2}$$

where $x_i,\ i=0,1,\ldots,m$ are the base points of the fitting problem moreover, w_i are the belonging weights. Frequently $w_i=1$ for all i.

The reader is asked here to check the properties of the scalar products given. It is true of course, that these scalar products define a norm:

$$\|f\|^2=\left(f,f\right). \tag{9.3}$$

Now there is no difficulty in applying Gram-Schmidt orthogonalization to generate an orthogonal system from the set of linearly independent functions $x^i,\ i=0,1,\ldots$. This process leads us *orthogonal polynomials*.

D9.1 Definition

We say a polynomial *monic* if the coefficient of the largest power is equal to 1.

T9.1 Theorem

Let the polynomials $p_i(x),\ i=0,1,\ldots$ be monic and of degree i. Then any polynomial $q(x)$ can be given uniquely as the linear combination of polynomials p_i:

$$q(x)=\sum_{j=0}^{n} b_j p_j(x). \tag{9.4}$$

Proof. Let the polynomial be given as $p_i(x)=x^i+p_{i.i-1}x^{i-1}+\ldots+p_{i,0}$, then the coefficients b_j are determined by the linear system:

$$\begin{bmatrix} 1 & p_{10} & p_{20} & \cdots & p_{n0} \\ & 1 & p_{21} & \cdots & p_{n1} \\ & & 1 & \dots & \vdots \\ & \bigcirc & & \ddots & \vdots \\ & & & & 1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ \vdots \\ b_n \end{bmatrix} = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ \vdots \\ a_n \end{bmatrix} \tag{9.5}$$

.

than can be solved from the bottom to upward. ■

C9.1 Corollary

Let p_i-s be orthogonal polynomials. Then p_{n+1} is orthogonal to every polynomial having degree smaller than n, as it can be expanded by lower degree orthogonal polynomials.

1.61. Recursion for orthogonal polynomials

The monic orthogonal polynomials can be generated recursively starting with $p_0(x)$ and $p_1(x)$ by using the formula:

$$p_{n+1} = (x - \alpha_{n+1}) p_n - \beta_n p_{n-1}. \tag{9.6}$$

To prove the statement, consider the scalar product

$$(xp_k, p_n) = (p_k, xp_n).$$

The result is zero, if

$$k+1 < n \text{ és } n+1 < k$$

because of Corollary C 9.1. It is nonzero, if

$$n-1 \le k \le n+1,$$

hence only the polynomials p_{n-1}, p_n, p_{n+1} will have nonzero expansion coefficients for xp_n:

$$xp_n = p_{n+1} + \alpha_{n+1} p_n + \beta_n p_{n-1}.$$

Now expressing for p_{n+1} gives (9.6). ■

T9.2 Theorem

We have for the expansion coefficients α_{n+1} and β_n

$$\alpha_{n+1} = \frac{(xp_n, p_n)}{(p_n, p_n)}, \tag{9.7}$$

$$\beta_n = \frac{(xp_n, p_{n-1})}{(p_{n-1}, p_{n-1})} = \frac{(p_n, p_n)}{(p_{n-1}, p_{n-1})}. \tag{9.8}$$

Proof. Express xp_n from (9.6) and take the scalar product with p_n. Because of orthogonality α_{n+1} follows. To find the expression for β_n, the procedure is similar. We move x to p_{n-1} at first, then substitute the recursion for $n-1$: $xp_{n-1} = p_n + \alpha_n p_{n-1} + \beta_{n-1} p_{n-2}$:

$$\beta_n = \frac{(xp_n, p_{n-1})}{(p_{n-1}, p_{n-1})} = \frac{(p_n, xp_{n-1})}{(p_{n-1}, p_{n-1})} = \frac{(p_n, p_n)}{(p_{n-1}, p_{n-1})}. \qquad \blacksquare$$

1.62. Polynomial least squares

We apply the scalar product in (9.2) where $w_i = 1$ is chosen for all i and the first starting polynomial is

$$p_0 \equiv 1, \quad \|p_0\|^2 = \sum_{j=0}^{m} 1 = m+1. \tag{9.9}$$

The next polynomial p_1 is sought in the form of $x - \alpha_1$. Because of orthogonality, we have

$$(p_0, p_1) = 0 = (p_0, x - \alpha_1) \;\to\; (p_0, x) = \alpha_1 (p_0, p_0,)$$

yielding

$$\alpha_1 = \frac{1}{m+1} \sum_{j=0}^{m} x_j, \tag{9.10}$$

and β_0 may be taken zero.

The other polynomials can be computed from the recursion. A function given by the function values y_i, $i = 0,1,\ldots,m$ at base points x_i can be approximated in the least squares sense as follows:

$$y \approx \sum_{j=0}^{k} p_j(x) \frac{(p_j, y)}{(p_j, p_j)}, \quad (p_j, y) = \sum_{i=0}^{m} p_j(x_i) y_i. \tag{9.11}$$

This form looks like the expansion of vector y in linear algebra with respect to an orthogonal system $\{q_j\}$:

$$y = \sum_{j=1}^{k} \frac{q_j q_j^T y}{q_j^T q_j}. \tag{9.12}$$

The expression $P_k = \sum_{j=0}^{k} p_j(x) p_j(t) / (p_j, p_j)$ in (9.11) is a symmetric projection, thus (9.11) shows the least squares property: $\|(I - P_k) y\|$ is the distance of y from the subspace spanned by the polynomials p_j, $j = 0,\ldots,k$.

E9.1 Example

Using orthogonal polynomials, find the first degree polynomial that approximates the series of points in the least squares sense:

x_i	-1	0	1	2
y_i	1	2	2	4

Solution. First we find the orthogonal polynomials. $p_0(x)=1$, such that $(p_0,p_0)=4$. It follows that $p_1(x)=x-\alpha_1$, where $\alpha_1=(xp_0,p_0)/(p_0,p_0)=\sum_{j=0}^{3}x_j/(p_0,p_0)=1/2$. Yet we have to compute the square norm of $p_1(x)$: $(p_1,p_1)=\sum_{j=0}^{3}(x_j-\alpha_1)^2=\frac{1}{4}(9+1+1+9)=5$. Now the first degree least squares polynomial is given by:

$$P_1(x)=\frac{(p_0,y)}{(p_0,p_0)}p_0+\frac{(p_1,y)}{(p_1,p_1)}p_1=\frac{9}{4}+\frac{1}{5}\cdot\frac{1}{2}(-3-2+2+12)(x-\frac{1}{2})=\frac{9}{4}+\frac{9}{10}(x-\frac{1}{2}).$$

1.63. Problems

9.1. Check if the base points are located symmetrically to $x=0$, then $\alpha_j=0,\ i=1,2,\ldots$ hold and the polynomials are alternating even and odd functions.

9.2. Find the orthogonal polynomials p_0,p_1,p_2 for the base points $\{-2,-1,0,1,2\}$!

9.3. The Chebyshev polynomials are also orthogonal and they can be generated by the following recursion: $T_0=1,\ T_1=x,\ T_{n+1}=2xT_n-T_{n-1}$. Although they are not monic now, yet it is the familiar form. Expand $4x^2-3x+2$ with Chebyshev polynomials!

9.4. $P(x)=\sum_{j=0}^{k}(2j+1)T_j(x)$. Give a skillful way of computing the sum at the point x_0!

9.5. Show that $(p_i,p_i)=\mu_0\beta_1\beta_2\ldots\beta_i$, where $\mu_0=(p_0,p_0)\left[=\int_a^b\alpha(x)dx\right]$ is the 0-th moment.

9.6. Show that the principal minors of the tridiagonal matrix

$$\begin{pmatrix} x-\alpha_1 & -\beta_1 & & \\ -\beta_1 & x-\alpha_2 & \ddots & \\ & \ddots & \ddots & -\beta_{n-1} \\ & & -\beta_{n-1} & x-\alpha_n \end{pmatrix}$$

have the same recursions as orthogonal polynomials with parameters α_i and β_i^2.

Iterative solution of linear systems

It is not always straightforward to solve linear systems with direct methods such as *LU*-decomposition. If the matrix is very large and sparse – that is, there are only few nonzero elements in rows or columns – then it is a disadvantage of *LU*-decomposition that when performing decomposition, the number of nonzero elements will grow up – the decomposed matrices get dense – and that may cause storage problems on one hand, on the other, the operation count will grow because of the increasing number of nonzeroes. Such problems will not arise in iteration methods but slow convergence may be a problem.

1.64. Fix point iteration

Let

$$A = M - N \tag{10.1}$$

be a splitting of matrix $A \in \mathbb{R}^{n\times n}$. If M is invertible, then we may start the following iteration: $Ax = (M-N)x = b \quad \rightarrow \quad x = M^{-1}(b+Nx)$, that is

$$x^{(k+1)} = M^{-1}Nx^{(k)} + M^{-1}b = Bx^{(k)} + c, \tag{10.2}$$

where k denotes the iteration number in the upper index. We call matrix B an *iteration matrix*. Then it is an important question, when will such a *fix point iteration* converge and how fast convergence may be expected?

1.64.1 Convergence of fix point iteration

The mapping $F:\ \mathbb{R}^n \to \mathbb{R}^n$ is said a contraction, if there exists a number $0 \le q < 1$, such that for all $\forall x, y \in \mathbb{R}^n$ the inequality

$$\|F(x) - F(y)\| \le q\|x - y\| \tag{10.3}$$

holds. Here q is the c*ontraction number.* Observe that the mapped vectors get closer to each other because of $q < 1$.

1.64.2 Banach's fix point theorem

Let $F:\ \mathbb{R}^n \to \mathbb{R}^n$ be mapping with contraction number $q < 1$. Then

1) $\exists x^* \in \mathbb{R}^n:\ x^* = F(x^*)$, that is, there exists a fix point of the iterations an it is unique.

2) For all initial vectors $x^{(0)} \in \mathbb{R}^n$ the series $x^{(k+1)} = F(x^{(k)})$ is convergent and $\lim\limits_{k\to\infty} x^{(k)} \to x^*$.

3) We have the error estimate: $\left\|x^{(k)} - x^*\right\| \le \dfrac{q^k}{1-q}\left\|x^{(1)} - x^{(0)}\right\|$.

Proof. The series $x^{(k+1)} = F(x^{(k)})$ is a Cauchy series: $\left\|x^{(k+1)} - x^{(k)}\right\| = \left\|F(x^{(k)}) - F(x^{(k-1)})\right\| \le$ $\le q\left\|x^{(k)} - x^{(k-1)}\right\| \le q^2\left\|x^{(k-1)} - x^{(k-2)}\right\| \le \ldots \le q^k\left\|x^{(1)} - x^{(0)}\right\|$. Thus the subsequent elements in the series are getting closer to each other such that there exists a limit. Assume that $m \ge k \ge 1$. Then forming the telescopic sum and tending with $m \to \infty$ leads to statement 3):

$$\left\|x^{(m)}-x^{(k)}\right\|=\left\|x^{(m)}-x^{(m-1)}+x^{(m-1)}-x^{(m-2)}+\ldots+x^{(k+1)}-x^{(k)}\right\|\le(q^{m-1}+q^{m-2}+\ldots+q^{k})\left\|x^{(1)}-x^{(0)}\right\|=$$

$$=\frac{q^k-q^m}{1-q}\left\|x^{(1)}-x^{(0)}\right\|<\frac{q^k}{1-q}\left\|x^{(1)}-x^{(0)}\right\|.$$

To prove uniqueness, assume indirectly that there exist two fix points: x_1^* and x_2^*. But then using the contraction property, we get contradiction because $\left\|x_1^*-x_2^*\right\|=\left\|F(x_1^*)-F(x_2^*)\right\|\le q\left\|x_1^*-x_2^*\right\|$, $q<1$ follows. Here the right hand side is definitely smaller whereas larger or equal would follow. ■

Applying the theorem for the iteration $x^{(k+1)}=Bx^{(k)}+c$, it is convergent if the mapping $F(x)=Bx+c$ is contractive:

$$\left\|F(x)-F(y)\right\|=\left\|Bx+c-By-c\right\|=\left\|B(x-y)\right\|\le\left\|B\right\|\left\|x-y\right\|.$$

There is contraction if we find a norm for which $\left\|B\right\|<1$ holds. We remark without proof: the spectral radius is the infimum of induced norms, such that we may say: $Bx+c$ is convergent, if the spectral radius of B is less than 1: $\rho(B)<1$. We say the splitting in (10.1) *regular*, if M is invertible and $\rho(M^{-1}N)<1$ holds.

1.65. Jacobi iteration

Define the splitting of A as $A=L+D+U$, where $D=\mathrm{diag}(A)$, $L=\mathrm{tril}(A,-1)$ and $U=\mathrm{triu}(A,1)$ that is, they are strictly lower and upper triangular parts of A.

The choice for Jacobi iteration is $M=D$ and $N=-L-U$, thus

$$B_J=-D^{-1}(L+U),\;\; c_J=D^{-1}b. \tag{10.4}$$

The component-wise form is $x_i^{(k+1)}=-\frac{1}{a_{ii}}\left(\sum_{\substack{j=1\\ j\ne i}}^{n}a_{ij}x_j^{(k)}-b_i\right)$.

Storage is needed for $A,b,x^{(k)},x^{(k+1)}$. A practical starting vector may be: $x^{(0)}=c_J$.

1.65.1 Theorem

If A is diagonally dominant by rows, then the Jacobi iteration is convergent.

Proof. $\left\|B_J\right\|_\infty=\max_{(k)}\left\|e_k^TD^{-1}(L+U)\right\|_\infty=\max_{(k)}\sum_{j\ne k}\left|\frac{a_{kj}}{a_{kk}}\right|<1$, that shows contraction.

1.66. Gauss-Seidel iteration

For Gauss-Seidel iteration the splitting is given by $M=L+D,\;N=-U$, such that

$$B_{GS}=-(L+D)^{-1}U,\;\; c_{GS}=(L+D)^{-1}b. \tag{10.5}$$

The component-wise form of Gauss-Seidel iteration is found from the i-th row of $(L+D)x^{(k+1)}=-Ux^{(k)}+b$:

$$x_i^{(k+1)} = -\frac{1}{a_{ii}}\left(\sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} + \sum_{j=i+1}^{n} a_{ij}x_j^{(k)} - b_i\right), \quad i = 1, 2, \ldots, n. \tag{10.6}$$

Now the order of operations make it possible that the value of $x_i^{(k)}$ can be overwritten by $x_i^{(k+1)}$. Therefore it is necessary to store only A, b, x, it is better as compared to Jacobi iteration. Suggested starting vector if there is no any better: $x^{(0)} = c_{GS}$.

1.66.1 Theorem. Norm estimate for splitting

Let A_1, A_2, D be $n \times n$ real matrices, where D is diagonal: $e_i^T D e_i = d_i$ and $\left\|e_i^T A_1\right\|_\infty < |d_i| \ \forall \ i$. Then one has the estimate:

$$\left\|(A_1 + D)^{-1} A_2\right\|_\infty \le \max_{(i)} \frac{\left\|e_i^T A_2\right\|_\infty}{|d_i| - \left\|e_i^T A_1\right\|_\infty}, \tag{10.7}$$

where maximum search should be done only for nonzero numerators.

Proof. As $A_1 + D$ is diagonally dominant, it is invertible. According to the definition of induced norms: $\left\|(A_1 + D)^{-1} A_2\right\|_\infty = \max_{\|x\|_\infty = 1} \left\|(A_1 + D)^{-1} A_2 x\right\|_\infty$. Introduce vector $y = (A_1 + D)^{-1} A_2 x$ and assume that the maximum takes its value at the i-th index: $\|y\|_\infty = |y_i|$. After reordering $A_2 x = (A_1 + D)y \to Dy = A_2 x - A_1 y$, from where $\left\|e_i^T D y\right\|_\infty = |d_i y_i| \le \left\|e_i^T A_2\right\|_\infty \|x\|_\infty + \left\|e_i^T A_1\right\|_\infty |y_i|$ follows for the i-th row. Observing $\|x\|_\infty = 1$ gives the rerquired inequality. As it is not known, which i index gives $\|y\|_\infty$, therefore the maximum is chosen. If the i-th row of A_2 is zero, then $|d_i||y_i| \le \left\|e_i^T A_1\right\| |y_i|$ follows, that is contradicting for nonzero $|y_i|$, so that such rows should be omitted. ■

1.66.2 Theorem

Let A be diagonally dominant by rows. Let $D = \mathrm{diag}(A)$ and choose the elements of A_1 and A_2 arbitrarily from the off-diagonal part of A such that $A = A_1 + D + A_2$. Then the choice $M = A_1 + D$, $N = -A_2$ gives a regular splitting.

Proof. We apply the previous theorem, where $d_i = e_i^T D e_i = a_{ii}$. Introduce

$$\alpha_i = \frac{1}{|a_{ii}|}\left\|e_i^T A_1\right\|_\infty, \quad \text{and} \quad \beta_i = \frac{1}{|a_{ii}|}\left\|e_i^T A_2\right\|_\infty \tag{10.8}$$

Then we have the estimate from (10.7):

$$\left\|(A_1 + D)^{-1} A_2\right\|_\infty \le \max_{(i)} \frac{\beta_i}{1 - \alpha_i}, \tag{10.9}$$

where $\alpha_i + \beta_i < 1$ because of diagonal dominance. Now $\beta_i < 1 - \alpha_i$ follows so that the norm estimate above gives a value less than 1. ■

A consequence of this theorem that Gauss-Seidel iteration is convergent for diagonally dominant matrices. For Jacobi iteration the similar result is $\max_i(\alpha_i+\beta_i)$ such that Gauss-Seidel iteration may have even faster convergence.

1.67. Gauss-Seidel (GS-) relaxation

In hoping acceleration of convergence, we share the role of D between L and U :

$$(L+D)x=-Ux+b \qquad \text{/ multiply by } \omega$$

$$Dx=Dx \qquad \text{/ multiply by } (1-\omega)$$

Adding the equations gives:

$$(D+\omega L)x^{(k+1)}=(1-\omega)Dx^{(k)}-\omega Ux^{(k)}+\omega b \tag{10.10}$$

$$x^{(k+1)}=(D+\omega L)^{-1}\left[(1-\omega)D-\omega U\right]x^{(k)}+(D+\omega L)^{-1}\omega b.$$

Now the iteration mtrix is

$$B_{GS}(\omega)=(D+\omega L)^{-1}\left[(1-\omega)D-\omega U\right]. \tag{10.11}$$

If $\omega=1$, then we have the Gauss-Seidel iteration. The i-th row of (10.10) gives:

$$x_i^{(k+1)}=-\frac{\omega}{a_{ii}}\left(\sum_{j=1}^{i-1}a_{ij}x_j^{(k+1)}+\sum_{j=i+1}^{n}a_{ij}x_j^{(k)}-b_i\right)+(1-\omega)x_i^{(k)},\quad i=1,2,\dots,n. \tag{10.12}$$

Now we have the following picture: The next result of a Gauss-Seidel step is multiplied by ω and the $(1-\omega)$-multiple of the k-th vector is added.

1.68. Some theorems on relaxation methods

1. If A has nonzero diagonal elements, otherwise it is arbitrary, then $\rho(B_{GS}(\omega))\ge|\omega-1|$, therefore convergence may be expected only if ω falls between 0 and 2.
2. Let $A\in\mathbb{R}^{n\times n}$ be symmetric, positive definite and let $0<\omega<2$ hold. Then $\rho(B_{GS}(\omega))<1$, that is, GS-relaxation is convergent for all such ω.

The next two theorems refer to block-tridiagonal matrices. Of course, in case of 1×1-es blocks, we get back simple tridiagonal matrices.

3. Let $A\in\mathbb{R}^{n\times n}$ be block-tridiagonal matrix. Then for the corresponding matrices of block Jacobi (J) and block GS-iteration

$$\rho(B_{GS}^b)=\left[\rho(B_J^b)\right]^2.$$

That means, the two are convergent or divergent at the same time and in case of convergence GS- iteration is twice as fast.

4. Let A be block-tridiagonal, symmetric and positive definite. Then the block Jacobi iteration, and block GS relaxation at $0<\omega<2$ are convergent. The optimal relaxation parameter for the latter is

$$\omega_0=2/\left(1+\sqrt{1-\left(\rho(B_J^b)\right)^2}\right)\in(0,2)$$

and for this optimal parameter the spectral radius is

$$\rho(B_{GS}^b(\omega_0)) = |\omega_0 - 1| < \rho(B_{GS}^b) = \left(\rho(B_J^b)\right)^2 .$$

1.69. Optimal ω for one step

We have seen in (10.12) that starting from x_k the vector

$$x_{k+1}^{\omega} = \omega x_{k+1} + (1-\omega)x_k = x_k + \omega(x_{k+1} - x_k) \tag{10.13}$$

is computed instead of x_{k+1} of the Gauss-Seidel method. We modify the relaxation method a little and introduce notations $y_k = x_{k+1} - x_k$, $r_k = b - Ax_k$ and determine parameter ω generally for the splitting $A = M - N$. We get from (10.13):

$$r_{k+1} = b - Ax_{k+1}^{\omega} = r_k - \omega A y_k . \tag{10.14}$$

In the next step we get ω_k of the k-th step from the condition that $\|r_{k+1}\|_2$ is minimal. For that we have to do no more than solve the "equation" $Ay_k\omega = r_k$ with the pseudo-inverse for ω:

$$\omega_k = \left(Ay_k\right)^+ r_k = \frac{y_k^T A^T r_k}{\|Ay_k\|_2^2} = \frac{r_k^T A y_k}{\|Ay_k\|_2^2} . \tag{10.15}$$

In order that x_{k+1} should not be computed explicitly, y_k is expressed from the non relaxed form:

$$x_{k+1} = M^{-1}(Nx_k + b) = x_k + M^{-1}\left(b - (M-N)x_k\right) = x_k + M^{-1}r_k . \tag{10.16}$$

From here

$$y_k = M^{-1} r_k . \tag{10.17}$$

We introduce a newer vector for the determination of ω_k:

$$c_k = Ay_k = (M-N)M^{-1}r_k = r_k - Ny_k \tag{10.18}$$

and then we arrive at the following algorithm:

```
Start: $r_0 = b - Ax_0$;
For $k = 1, 2, 3, \ldots,$ compute
    $y_k = M^{-1} r_k$;
    $c_k = r_k - N y_k$;
    $\omega_k = \frac{r_k^T c_k}{c_k^T c_k}$;
    $x_{k+1} = x_k + \omega_k * y_k$;
    $r_{k+1} = r_k - \omega_k * c_k \quad (= b - Ax_{k+1})$;
```

There are here two possibilities for computing r_{k+1}. The first one is cheaper, of course. But with the advance of iteration, it may happen that the second method will substantially differ from the first one. Then it is suggested to improve the value of r_{k+1} with the aid of the second method. The vectors were indexed in the algorithm, although it is not necessary because the new vectors may overwrite the previous ones.

1.70. Richardson iteration

If the eigenvalues of the matrix are real, positive numbers, then we may start iteration according to the following observation:

$$(I-pA)r_i = r_{i+1} = b - A(x_i + pr_i) = b - Ax_{i+1},\ \ x_{i+1} = x_i + pr_i, \tag{10.19}$$

where number p is chosen so that the spectral radius of $I-pA$ be as small as possible. Now the eigenvalues of $I-pA$ are $1-p\lambda_i$-s. Now the eigenvalues are mapped by the linear function $1-px$. It intersects the point (0,l) on the horizontal axis and for positive p's it has a negative slope. Let the smallest eigenvalue be m and let the largest be M. In the Richardson iteration the optimal p is chosen from the condition that the smallest and largest eigenvalue should be mapped into numbers of the same absolute value:

$$1-pm = -(1-pM) \ \ \rightarrow \ \ p = 2/(m+M). \tag{10.20}$$

With this choice the spectral radius of $I-pA$ will be $(M-m)/(M+m)$.

If the eigenvalues of the matrix are not known, but we know that the eigenvalues are positive, e.g. because A is symmetric, positive definite, the number p can be found from the condition that $\|r_{i+1}\|_2$ be minimal. Then the pseudo-solution of the equation $r_i = pAr_i$ is

$$p = \frac{r_i^T A^T r_i}{\|Ar_i\|_2^2} = \frac{r_i^T Ar_i}{\|Ar_i\|_2^2}. \tag{10.21}$$

It will be enough to compute p for a few times in the course of the iteration, because it will oscillate around the previously stated optimal value.

1.71. Problems

10.1. How should we modify Jacobi iteration, if the matrix is diagonally dominant with respect to columns?

10.2. Show that Theorem 10.3.1 can be reformulated for the case when the matrix is diagonally dominant with respect to columns.

10.3. Elaborate estimate (10.9) for the GS-iteration! What happens to Jacobi and GS iteration if instead of diagonal dominance we have equality in some equations? And if equality takes place in the last row?

10.4. $A = \begin{pmatrix} 5 & -1 & 2 & 1 \\ -3 & 7 & -2 & 0 \\ 3 & 0 & 5 & -1 \\ 0 & 2 & -4 & 6 \end{pmatrix}$. $\|B_J\|_\infty = ?$ $\|B_{GS}\|_\infty \le ?$

10.5. Applying Theorem 10.3.1 show: $\|A^{-1}\|_\infty \le \max_i \frac{1}{|a_{ii}|(1-\alpha_i-\beta_i)}$, see also (10.8), if A is strictly diagonally dominant by rows. How can we modify statement for diagonal dominance with respect to columns?

10.6. Assuming $D+\omega L$ is diagonally dominant by rows, prove $\|B_{GS}(\omega)\|_\infty \le \max\limits_{(j)} \dfrac{|1-\omega|+\omega\beta_j}{1-\omega\alpha_j}$ by applying Theorem 10.3.1.

10.7. If $\|D^{-1}A_1\|<1$ holds, then we can derive an inequality similar to that of Theorem 10.3.1 by using (2.15), because of the equality $(A_1+D)^{-1}A_2=(I+D^{-1}A_1)^{-1}D^{-1}A_2$. Show that $\|(A_1+D)^{-1}A_2\| \le \dfrac{\|D^{-1}A_2\|}{1-\|D^{-1}A_1\|}$ holds for induced norms. Is that necessary that D be a diagonal matrix? For the matrix of Example 4 which method gives a better estimate?