0 Fundamental concepts

This is a short summary of certain important fundamental techniques, which are not a part of the course but will be used several times.

0.1 Orthogonal matrices and QR factorization

The QR-factorization is a factorization involving an orthogonal matrix Q and an upper triangular matrix R.

Definition 0.1.1 (Orthogonal matrix)**.** *$Q \in \mathbb{R}^{n\times m}$ is called an orthogonal matrix if*

$$Q^TQ = I.$$

For complex matrices, the corresponding property is called unitary: $Q^*Q = I$.

Orthogonal matrices are important in matrix computations, most importantly when the matrix represents a basis of a vector space. Many properties of the vector space are not robust with respect to rounding errors if the basis is not orthogonal.

The identity matrix will be denoted $I \in \mathbb{R}^{k\times k}$, where k is given by the context we use it.

Properties:

(i) The columns of an orthogonal matrix are orthonormal.

(ii) If $n = m$, then $Q^T = Q^{-1}$.

(iii) If $n = m$, then $QQ^T = I$.

Note that (iii) is not satisfied if Q is a rectangular matrix ($n \neq m$). For instance

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

is an orthogonal matrix since $Q^TQ = I$, but

$$QQ^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Theorem 0.1.2 (Uniqueness of QR-factorization)**.** *For any matrix $A \in \mathbb{R}^{n\times m}$, $n \leq m$ there exists an orthogonal matrix $Q \in \mathbb{R} \in \mathbb{R}^{n\times m}$ and an upper triangular matrix $R \in \mathbb{R}^{m\times m}$ such that*

$$A = QR.$$

A QR-factorization can be computed with the matlab command `[Q,R]=qr(A)`. It will however in general not return the solution with positive diagonal elements.

Moreover, if A is non-singular, the diagonal elements of R can be chosen positive, and the decomposition where the diagonal elements of R are positive is unique.

The QR-decomposition is the underlying method to solve overdetermined linear systems of equations, for instance with the backslash operator in matlab when the matrices are rectangular. The decomposition can be computed in a finite number of operations with for instance Householder reflectors or Givens rotations, which will be used in this course.

0.2 Jordan canonical form (JCF)

The Jordan canonical form, also sometimes the Jordan form or the Jordan decomposition, is a transformation that brings the matrix to a certain block diagonal form.

The Jordan canonical form is typically treated in basic linear algebra courses, but also used in the study of stability theory in for instance systems theory and differential equations.

Definition 0.2.1 (Jordan canonical form)**.** *The Jordan decomposition of a matrix $A \in \mathbb{R}^{n\times n}$ is an invertible matrix $X \in \mathbb{R}^{n\times n}$ and a block diagonal matrix*

$$D = \begin{bmatrix} J_1 & & \\ & \ddots & \\ & & J_q \end{bmatrix}$$

where

$$J_i = \begin{bmatrix} \lambda_i & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_i \end{bmatrix} \in \mathbb{R}^{n_i\times n_i}$$

such that

$$A = XDX^{-1}.$$

The matrix J_i is called a Jordan block, or sometimes a Jordan matrix.

The Jordan decomposition is a *eigenvalue revealing* decomposition, since the eigenvalues of A are the diagonal of the J-matrix.

Example of Jordan decomposition

The Jordan decomposition of

$$A = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 1 & 0 & 3 \end{bmatrix}$$

is represented by

$$D = \begin{bmatrix} 3 & 1 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3 \end{bmatrix}, \quad X = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}$$

since $A = XDX^{-1}$. Note that the eigenvalue $\lambda = 3$ has two Jordan blocks, one of size $n_1 = 2$ and one of size $n_2 = 1$.

Theorem 0.2.2 (Existance and uniqueness of Jordan decomposition)**.** *All matrices $A \in \mathbb{R}^{n\times n}$ have a Jordan decomposition. The (unordered) set of Jordan blocks is unique for any given matrix.*

Definition 0.2.3. *If a matrix A has a Jordan decomposition where all Jordan blocks have size one (such that D is a diagonal matrix) the matrix A is called a diagonalizable matrix.*

Lemma 0.2.4. *Suppose A is symmetric. Then A is diagonalizable and the X matrix in the Jordan decomposition can be chosen as an orthogonal matrix.*

We say that the eigenvalues of a matrix A denoted $\lambda_1, \ldots, \lambda_n$ are distinct if they are different from each other, that is

$$\lambda_i \neq \lambda_j \quad \text{when } i \neq j$$

This is a sufficient condition for diagonalizability.

Lemma 0.2.5. *If $A \in \mathbb{R}^{n\times n}$ has n distinct eigenvalues, then A is diagonalizable.*

0.3 Linear least squares problems

Let and $c \in \mathbb{R}^n$ and let $B \in \mathbb{R}^{n\times m}$ be a tall skinny matrix ($n > m$), where the columns of B are linearly independent. The problem to determine a vector $z \in \mathbb{R}^m$ which minimizes the problem

$$\min_{z\in\mathbb{R}^m} \|Bz - c\|_2 \tag{$\star$}$$

is usually referred to as the linear least squares problem. We will commonly use the corresponding minimizer

$$z_* = \operatorname*{argmin}_{z\in\mathbb{R}^n} \|Bz - c\|_2 \quad \Rightarrow \quad \min_{z\in\mathbb{R}^m} \|Bz - c\|_2 = \|Bz_* - c\|_2.$$

The problem ($\star$) is sometimes denoted

$$Bz \approx c.$$

The solution to linear least squares problem are given by the normal equations

$$B^T Bz = B^T c, \tag{$\star\star$}$$

which is a linear system of equation which has a unique solution.

Although ($\star\star$) gives a direct procedure how to compute a solution to ($\star$) by solving a linear system of equations, it is not the best way to compute the solution numerically. In MATLAB the backslash operator can compute solutions to linear least squares problems `z=B\c`

The method behind backslash for tall and skinny matrices is based on computing a QR-factorization.

0.4 *Convergence order and convergence factor*

Suppose $v_k \in \mathbb{R}^n$, $k = 1, \ldots$ is a sequence of vectors such that that v_k converges to some vector $v_* \in \mathbb{R}^n$ as $k \to \infty$. In our context, k corresponds to an approximation parameter in a numerical method.

If we define the error as

$$e_k = \|v_k - v_*\|$$

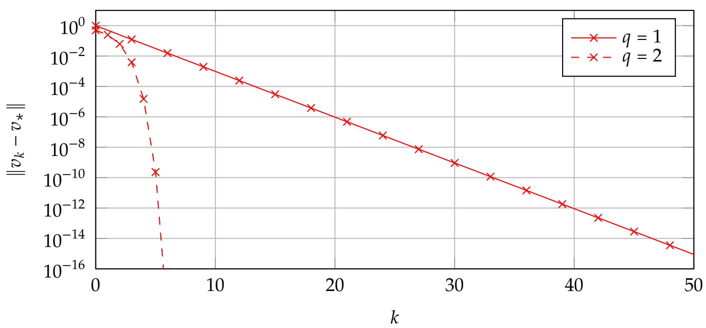
the convergence order is the a value p such that

$$e_{k+1} \leq \alpha e_k^p. \tag{1}$$

Errors of the type (1) commonly occur in iterative methods as in this course. Note that this is different from the error typically occurring in the discretization of differential equations, where the second order approximation leads to error of the type $h^2 = 1/k^2$. That type of convergence is sometimes called algebraic converge, and is better visualized in a log-log plot.

Note that the error at iteration $k+1$ is given as (a constant) times the previous error, and we write $e_{k+1} = \mathcal{O}(e_k^p)$. For $p > 1$ the error often goes to zero very quickly and we need to use a logarithmic y-axis in order capture the behaviour.

The figures shows $e_{k+1} = \frac{1}{2} e_k^p$.



When $q = 2$ we say that we have *quadratic convergence*, and $q = 3$ is *cubic convergence*.

When $q = 1$, we sometimes say that the method is linearly convergent, although we prefer the term "convergence order one" since linear convergence can sometimes refer $e_k \leq 1/k$ which is much slower.

If we have convergence order one, we can then also express the relation $e_{k+1} \leq \alpha e_k$ as

$$e_k \leq \alpha^k e_0 = \mathcal{O}(\alpha^k)$$

Note that we here express the error in terms of k directly rather than as a function of the previous iterate. The value α is called *convergence factor* or *convergence rate*.

The use of convergence factors when $q > 1$ is rare. It matters less when the convergence is usually fast anyway.

Examples in this course:

- The power method error is convergent with convergence order $q = 1$
- The convergence of the power method is $|\lambda_1|/|\lambda_2|$
- The Rayleigh quotient iteration is convergent with convergence order two or three.

