Part III

Nonlinear optimization

Chapter 9

The basics and applications

The problem of minimizing a function of several variables, possibly subject to constraints on these variables, is what optimization is about. So the main problem is easy to state! And, more importantly, such problems arise in many applications in natural science, engineering, economics and business as well as in mathematics itself.

Nonlinear optimization differs from Fourier analysis and wavelet theory in that classical multivariate analysis also is an important ingredient. A recommended book on this, used here at the University of Oslo, is [8] (in Norwegian). It contains a significant amount of fixed point theory, nonlinear equations, and optimization.

There are many excellent books on nonlinear optimization (or nonlinear programming, as it is also called). Some of these books that have influenced these notes are [1, 2, 9, 5, 13, 11]. These are all recommended books for those who want to go deeper into the subject. These lecture notes are particularly influenced by the presentations in [1, 2].

Optimization has its mathematical foundation in linear algebra and multivariate calculus. In analysis the area of convexity is especially important. For the brief presentation of convexity given here the author's own lecture notes [4] (originally from 2001), and the very nice book [14], have been useful sources. But, of course, anyone who wants to learn convexity should study the work by R.T. Rockafellar, see e.g. the classic text [12].

Linear optimization (LP, linear programming) is a special case of nonlinear optimization, but we do not discuss this in any detail here. The reason for this is that we, at the University of Oslo, have a separate course in linear optimization which covers many parts of that subject in some detail.

This first chapter introduces some of the basic concepts in optimization and discusses some applications. Many of the ideas and results that you will find in these lecture notes may be extended to more general linear spaces, even infinite-dimensional. However, to keep life a bit easier and still cover most applications, we will only be working in $\mathbb{R}^n$.

Due to its character this chapter is a "proof-free zone", but in the remaining text we usually give full proofs of the main results.

Notation: For $\boldsymbol{z} \in \mathbb{R}^n$ and $\delta > 0$ define the (closed) *ball* $\bar{B}(\boldsymbol{z}; \epsilon) = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x} - \boldsymbol{z}\| \leq \epsilon\}$. It consists of all points with distance at most ϵ from $\boldsymbol{z}$. Similarly, define the open ball $B(\boldsymbol{z}; \epsilon) = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x} - \boldsymbol{z}\| < \epsilon\}$. A *neighborhood* of $\boldsymbol{z}$ is a set N containing $B(\boldsymbol{z}; \epsilon)$ for some $\epsilon > 0$. Vectors are treated as column vectors and they are identified with the corresponding n-tuple, denoted by $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$. A statement like

$$P(\boldsymbol{x}) \quad (\boldsymbol{x} \in H)$$

means that the statement $P(\boldsymbol{x})$ is true for all $\boldsymbol{x} \in H$.

9.1 The basic concepts

Optimization deals with finding optimal solutions! So we need to define what this is.

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a real-valued function in n variables. The function value is written as $f(\boldsymbol{x})$, for $\boldsymbol{x} \in \mathbb{R}^n$, or $f(x_1, x_2, \ldots, x_n)$. This is the function we want to minimize (or maximize) and it is often called the *objective function*. Let $\boldsymbol{x}^* \in \mathbb{R}^n$. Then $\boldsymbol{x}^*$ is a *local minimum* (or local minimizer) of f if there is an $\epsilon > 0$ such that

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in B(\boldsymbol{x}^*; \epsilon).$$

So, no point "sufficiently near" $\boldsymbol{x}^*$ has smaller f-value than $\boldsymbol{x}^*$. A *local maximum* is defined similarly, but with the inequality reversed. A stronger notion is that $\boldsymbol{x}^*$ is a *global minimum* of f which means that

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in \mathbb{R}^n.$$

A *global maximum* satisfies the opposite inequality.

The definition of local minimum has a "variational character"; it concerns the behavior of f near $\boldsymbol{x}^*$. Due to this it is perhaps natural that Taylor's formula, which gives an approximation of f in such a neighborhood, becomes a main tool for characterizing and finding local minima. We present Taylor's formula, in different versions, in Section 9.3.

An extension of the notion of minimum and maximum is for *constrained* problems where we want, for instance, to minimize $f(\boldsymbol{x})$ over all $\boldsymbol{x}$ lying in a given set C. Then $\boldsymbol{x}^* \in C$ is a *local minimum* of f over the set C, or subject to $\boldsymbol{x} \in C$ as we shall say, provided no point in C in some neighborhood of $\boldsymbol{x}^*$ has smaller f-value than $\boldsymbol{x}^*$. A similar extension holds for global minimum over C, and for maxima.

Example 9.1. To make these things concrete, consider an example from plane geometry. Consider the point set $C = \{(z_1, z_2) : z_1 \geq 0,\ z_2 \geq 0,\ z_1 + z_2 \leq 1\}$ in the plane. We want to find a point $\boldsymbol{x} = (x_1, x_2) \in C$ which is closest possible to the point $\boldsymbol{a} = (3, 2)$. This can be formulated as the minimization problem

$$\begin{aligned} &\text{minimize} && (x_1 - 3)^2 + (x_2 - 2)^2 \\ &\text{subject to} && \\ & && x_1 + x_2 \leq 1 \\ & && x_1 \geq 0,\ x_2 \geq 0. \end{aligned}$$

The function we want to minimize is $f(\boldsymbol{x}) = (x_1 - 3)^2 + (x_2 - 2)^2$ which is a quadratic function. This is the square of the distance between $\boldsymbol{x}$ and $\boldsymbol{a}$; and minimizing the distance or the square of the distance is equivalent (why?). A minimum here is $\boldsymbol{x^*} = (1, 0)$. If we instead minimize this function f over $\mathbb{R}^2$, the unique global minimum is $\boldsymbol{x^*} = \boldsymbol{a} = (3, 2)$. It is useful to study this example and try to solve it geometrically as well as analytically.

In optimization one considers minimization and maximization problems. As

$$\max\{f(\boldsymbol{x}) : \boldsymbol{x} \in S\} = -\min\{-f(\boldsymbol{x}) : \boldsymbol{x} \in S\}$$

it is clear how to convert a maximization problem into a minimization problem (or vise versa). This transformation may, however, change the properties of the function you work with. For instance, if f is convex (definitions come later!), then $-f$ is not convex (unless f is linear), so rewriting between minimization and maximization may take you out of a class of "good problems". Note that a minimum or maximum may not exist. A main tool one uses to establish that optimal solutions really exist is the *extreme value theorem* as stated next. You may want to look these notions up in [8].

Theorem 9.2. Let C be a subset of $\mathbb{R}^n$ which is closed and bounded, and let $f : C \to \mathbb{R}$ be a continuous function.

Then f attains both its (global) minimum and maximum, so these are points $\boldsymbol{x^1}, \boldsymbol{x^2} \in C$ with

$$f(\boldsymbol{x^1}) \leq f(\boldsymbol{x}) \leq f(\boldsymbol{x^2}) \qquad (\boldsymbol{x} \in C).$$

9.2 Some applications

It is useful to see some application areas for optimization. They are many, and here we mention a few in some detail.

9.2.1 Portfolio optimization

The following optimization problem was introduced by Markowitz in order to find an optimal portfolio in a financial market; he later received the Nobel prize in economics[1] (in 1990) for his contributions in this area:

$$\begin{array}{ll} \text{minimize} & \alpha \sum_{i,j\le n} c_{ij}x_ix_j - \sum_{j=1}^n \mu_j x_j \\ \text{subject to} & \\ & \sum_{j=1}^n x_j = 1 \\ & x_j \ge 0 \qquad (j \le n). \end{array}$$

The model may be understood as follows. The decision variables are $x_1, x_2, \ldots, x_n$ where x_i is the fraction of a total investment that is made in (say) stock i. Thus one has available a set of stocks in different companies (Statoil, IBM, Apple etc.) or bonds. The fractions x_i must be nonnegative (so we consider no short sale) and add up to 1. The function f to be minimized is

$$f(\boldsymbol{x}) = \alpha \sum_{i,j\le n} c_{ij}x_ix_j - \sum_{j=1}^n \mu_j x_j.$$

It can be explained in terms of random variables. Let R_j be the return on stock j, this is a random variable, and let $\mu_j = \mathbb{E}R_j$ be the expectation of R_j. So if X denotes the random variable $X = \sum_{j=1}^n x_j R_j$, which is the return on our portfolio (= mix among investments), then $\mathbb{E}X = \sum_{j=1}^n \mu_j x_j$ which is the second term in f. The minus sign in front explains that we really want to maximize the expected return. The first term in f is there because just looking at expected return is too simple. We want to spread our investments to reduce the risk. The first term in f is the variance of X multiplied by a weight factor α; the constant c_{ij} is the covariance of R_i and R_j and c_{ii} is the variance of R_i. The covariance of R_i and R_j is defined as $\mathbb{E}(R_i - \mu_i)(R_j - \mu_j)$.

So f is a weighted difference of variance and expected return. This is what we want to minimize. The optimization problem is to minimize a quadratic function subject to linear constraints. We shall discuss theory and methods for such problems later.

In order to use such a model one needs to find good values for all the parameters μ_j and c_{ij}; this is done using historical data from the stock markets. The weight parameter α is often varied and the optimization problem is solved for each such "interesting" value. This makes it possible to find a so-called efficient frontier of expectation versus variance for optimal solutions.

The Markowitz model is a useful tool for financial investments, and now extensions and variations of the model exist, e.g., by using different ways of measuring risk. All such models involve a balance between risk and expected return.

[1] The precise term is "Sveriges Riksbank Prize in Economic Sciences in Memory of Alfred Nobel"

9.2.2 Fitting a model

In many applications one has a mathematical model of some phenomenon where the model has some parameters. These parameters represent a flexibility of the model, and they may be adjusted so that the model explains the phenomenon best possible.

To be more specific consider a model

$$y = F_\alpha(\boldsymbol{x})$$

for some function $F_\alpha : \mathbb{R}^m \to \mathbb{R}$. Here $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n) \in \mathbb{R}^n$ is a parameter vector (so we may have several parameters). Perhaps there are natural constraints on the parameter, say $\alpha \in A$ for a given set A in $\mathbb{R}^n$.

For instance, consider

$$y = \alpha_1 \cos x_1 + x_2^{\alpha_2}$$

so here $n = m = 2$, $\alpha = (\alpha_1, \alpha_2)$ and $F_\alpha(\boldsymbol{x}) = \alpha_1 \cos x_1 + x_2^{\alpha_2}$ where (say) $\alpha_1 \in \mathbb{R}$ and $\alpha_2 \in [1, 2]$.

The general model may also be thought of as

$$y = F_\alpha(\boldsymbol{x}) + \text{error}$$

since it is usually a simplification of the system one considers. In statistics one specifies this error term as a random variable with some (partially) known distribution. Sometimes one calls y the *dependent variable* and $\boldsymbol{x}$ the *explaining variable*. The goal is to understand how y depends on $\boldsymbol{x}$.

To proceed, assume we are given a number of observations of the phenomenon given by points

$$(\boldsymbol{x^i}, y^i) \ \ (i = 1, 2, \ldots, m).$$

meaning that one has observed y^i corresponding to $\boldsymbol{x} = \boldsymbol{x^i}$. We have m such observations. Usually (but not always) we have $m \geq n$. The *model fit problem* is to adjust the parameter α so that the model fits the given data as good as possible. This leads to the optimization problem

$$\text{minimize} \ \sum_{i=1}^{m} (y^i - F_\alpha(\boldsymbol{x^i}))^2 \quad \text{subject to} \quad \alpha \in A.$$

The optimization variable is the parameter α. Here the model error is quadratic (corresponding to the Euclidean norm), but other norms are also used.

This optimization problem above is a constrained nonlinear optimization problem. When the function F_α depends linearly on α, which often is the case in practice, the problem becomes the classical *least squares approximation problem* which is treated in basic linear algebra courses. The solution is then characterized by a certain linear system of equations, the so-called normal equations.

9.2.3 Maximum likelihood

A very important problem in statistics, arising in many applications, is parameter estimation and, in particular, *maximum likelihood estimation*. It leads to optimization.

Let Y be a "continuous" real-valued random variable with probability densisty $p_x(y)$. Here x is a parameter (often one uses other symbols for the parameter, like ξ, θ etc.). For instance, if Y is a normal (Gaussian) variable with expectation x and variance 1, then $p_x(y) = \frac{1}{\sqrt{2\pi}} e^{-(y-x)^2/2}$ and

$$\mathbb{P}(a \leq Y \leq b) = \int_a^b \frac{1}{\sqrt{2\pi}} e^{-(y-x)^2/2} dy$$

where $\mathbb{P}$ denotes probability.

Assume Y is the outcome of an experiment, and that we have observed $Y = y$ (so y is a known real number or a vector, if several observations were made). On the basis of y we want to *estimate* the value of the parameter x which "explains" best possible our observation $Y = y$. We have now available the probability density $p_x(\cdot)$. The function $x \to p_x(y)$, for fixed y, is called the *likelihood* function. It gives the "probability mass" in y as a function of the parameter x. The *maximum likelihood* problem is to find a parameter value x which maximizes the likelihood, i.e., which maximizes the probability of getting precisely y. This is an optimization problem

$$\max_x p_x(y)$$

where y is fixed and the optimization variable is x. We may here add a constraint on x, say $x \in C$ for some set C, which may incorporate possible knowledge of x and assure that $p_x(y)$ is positive for $x \in C$. Often it is easier to solve the equivalent optimization problem of maximizing the logarithm of the likelihood function

$$\max_x \ln p_x(y)$$

This is a nonlinear optimization problem. Often, in statistics, there are several parameters, so $\boldsymbol{x} \in \mathbb{R}^n$ for some n, and we need to solve a nonlinear optimization problem in several variables, possibly with constraints on these variables. If the likelihood function, or its logarithm, is a concave function, we have (after multiplying by -1) a convex optimization problem. Such problems are easier to solve than general optimization problems. This will be discussed later.

As a specific example assume we have the linear statistical model

$$y = A\boldsymbol{x} + \boldsymbol{w}$$

where A is a given $m \times n$ matrix, $\boldsymbol{x} \in \mathbb{R}^n$ is an unknown parameter, $\boldsymbol{w} \in \mathbb{R}^m$ is a random variable (the "noise"), and $\boldsymbol{y} \in \mathbb{R}^n$ is the observed quantity. We assume that the components of $\boldsymbol{w}$, i.e., $w_1, w_2, \ldots, w_m$ are independent and identically

distributed with common density function p on $\mathbb{R}$. This leads to the likelihood function

$$p_{\boldsymbol{x}}(y) = \prod_{i=1}^{m} p(y_i - \boldsymbol{a}_i \boldsymbol{x})$$

where $\boldsymbol{a}_i$ is the i'th row in A. Taking the logarithm we obtain the maximum likelihood problem

$$\max \sum_{i=1}^{m} \ln p(y_i - \boldsymbol{a}_i \boldsymbol{x}).$$

In many applications of statistics is is central to solve this optimization problem numerically.

9.2.4 Optimal control problems

Recall that a discrete dynamical system is an equation

$$\boldsymbol{x}_{t+1} = h_t(\boldsymbol{x}_t) \quad (t = 0, 1, \ldots)$$

where $\boldsymbol{x}_t \in \mathbb{R}^n$, $\boldsymbol{x}_0$ is the initial solution, and h_t is a given function for each t. We here think of t as time and $\boldsymbol{x}_t$ is the state of the process at time t. For instance, let $n = 1$ and consider $h_t(\boldsymbol{x}) = a\boldsymbol{x}$ $(t = 0, 1, \ldots)$ for some $a \in \mathbb{R}$. Then the solution is $\boldsymbol{x}_t = a^t \boldsymbol{x}_0$. Another example is when A is an $n \times n$ matrix, $\boldsymbol{x}_t \in \mathbb{R}^n$ and $h_t(\boldsymbol{x}) = A\boldsymbol{x}$ for each t. Then the solution is $\boldsymbol{x}_t = A^t \boldsymbol{x}_0$. For the more general situation, where the system functions h_t may be different, it may be difficult to find an explicit solution for $\boldsymbol{x}_t$. Numerically, however, we compute $\boldsymbol{x}_t$ simply in a for-loop by computing $\boldsymbol{x}_0$, then $\boldsymbol{x}_1 = f_1(\boldsymbol{x}_0)$ and then $\boldsymbol{x}_2 = f_2(\boldsymbol{x}_1)$ etc.

Now, consider a dynamical system where we may "control" the system in each time step. We restrict the attention to a finite time span, $t = 0, 1, \ldots, T$. A proper model is then

$$\boldsymbol{x}_{t+1} = h_t(\boldsymbol{x}_t, \boldsymbol{u}_t) \quad (t = 0, 1, \ldots, T-1)$$

where $\boldsymbol{x}_t$ is the state of the system at time t and the new variable $\boldsymbol{u}_t$ is the control at time t. We assume $\boldsymbol{x}_t \in \mathbb{R}^n$ and $\boldsymbol{u}_t \in \mathbb{R}^m$ for each t (but these things also work if these vectors lie in spaces of different dimensions). Thus, when we choose the controls $\boldsymbol{u}_0, \boldsymbol{u}_1, \ldots, \boldsymbol{u}_{T-1}$ and $\boldsymbol{x}_0$ is known, the sequence $\{\boldsymbol{x}_t\}$ of states is uniquely determined. Next, assume there are given functions $f_t : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ that we call cost functions. We think of $f_t(\boldsymbol{x}_t, \boldsymbol{u}_t)$ as the "cost" at time t when the system is in state $\boldsymbol{x}_t$ and we choose control $\boldsymbol{u}_t$. The *optimal control* problem is

$$\begin{array}{lll} \text{minimize} & f_T(\boldsymbol{x}_T) + \sum_{t=0}^{T-1} f_t(\boldsymbol{x}_t, \boldsymbol{u}_t) & \\ \text{subject to} & & \\ & \boldsymbol{x}_{t+1} = h_t(\boldsymbol{x}_t, \boldsymbol{u}_t) & (t = 0, 1, \ldots, T-1) \end{array} \tag{9.1}$$

where the control is the sequence $(\boldsymbol{u}_0, \boldsymbol{u}_1, \ldots, \boldsymbol{u}_{T-1})$ to be determined. This problem arises an many applications, in engineering, finance, economics etc. We now rewrite this problem. First, let $\boldsymbol{u} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_T) \in \mathbb{R}^N$ where $N = Tn$. Since, as we noted, $\boldsymbol{x}_t$ is uniquely determined by $\boldsymbol{u}$, there is a function $\boldsymbol{v}_t$ such that $\boldsymbol{x}_t = \boldsymbol{v}_t(\boldsymbol{u})$ $(t = 1, 2, \ldots, T)$; $\boldsymbol{x}_0$ is given. Therefore the total cost may be written

$$f_T(\boldsymbol{x}_T) + \sum_{t=0}^{T-1} f_t(\boldsymbol{x}_t, \boldsymbol{u}_t) = f_T(\boldsymbol{v}_T(\boldsymbol{u})) + \sum_{t=0}^{T-1} f_t(\boldsymbol{v}_t(\boldsymbol{u}), \boldsymbol{u}_t) := f(\boldsymbol{u})$$

which is a function of $\boldsymbol{u}$. Thus, we see that the optimal control problem may be transformed to the unconstrained optimization problem

$$\min_{\boldsymbol{u} \in \mathbb{R}^N} f(\boldsymbol{u})$$

Sometimes there may be constraints on the control variables, for instance that they each lie in some interval, and then the transformation above results in a constrained optimization problem.

9.2.5 Linear optimization

This is not an application, but rather a special case of the general nonlinear optimization problem where all functions are linear. A *linear optimization* problem, also called *linear programming*, has the form

$$\begin{aligned} &\text{minimize} && \boldsymbol{c}^T \boldsymbol{x} \\ &\text{subject to} && \\ & && A\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \geq \boldsymbol{0}. \end{aligned} \tag{9.2}$$

Here A is an $m \times n$ matrix, $\boldsymbol{b} \in \mathbb{R}^m$ and $\boldsymbol{x} \geq \boldsymbol{0}$ means that $x_i \geq 0$ for each $i \leq n$. So in linear optimization one minimizes (or maximizes) a linear function subject to linear equations and nonnegativity on the variables. Actually, one can show any problem with constraints that are linear equations and/or linear inequalities may be transformed into the form above. Such problems have a wide range of application in science, engineering, economics, business etc. Applications include portfolio optimization and many planning problems for e.g. production, transportation etc. Some of these problems are of a combinatorial nature, but linear optimization is a main tool here as well.

We shall not treat linear optimization in detail here since this is the topic of a separate course, INF-MAT3370 Linear optimization. In that course one presents some powerful methods for such problems, the simplex algorithm and interior point methods. In addition one considers applications in network flow models and game theory.

9.3 Multivariate calculus and linear algebra

We first recall some useful facts from linear algebra.

The *spectral theorem* says that if A is a real symmetric matrix, then there is an orthogonal matrix V (i.e., its columns are orthonormal) and a diagonal matrix D such that

$$A = VDV^T.$$

The diagonal of D contains the eigenvalues of A, and A has an orthonormal set of eigenvectors (the columns of V).

A real symmetric matrix is *positive semidefinite*[2] if $\boldsymbol{x}^T A\boldsymbol{x} \geq 0$ for all $\boldsymbol{x} \in \mathbb{R}^n$. The following statements are equivalent

(i) A is positive semidefinite,

(ii) all eigenvalues of A are nonnegative,

(iii) $A = W^T W$ for some matrix W.

Similarly, a real symmetric matrix is *positive definite* if $\boldsymbol{x}^T A\boldsymbol{x} > 0$ for all nonzero $\boldsymbol{x} \in \mathbb{R}^n$. The following statements are equivalent

(i) A is positive definite,

(ii) all eigenvalues of A are positive,

(iii) $A = W^T W$ for some invertible matrix W.

Every positive definite matrix is therefore invertible.

We also recall some central facts from multivariate calculus. They will be used repeatedly in these notes. Let $f : \mathbb{R}^n \to \mathbb{R}$ be a real-valued function defined on $\mathbb{R}^n$. The *gradient* of f at $\boldsymbol{x}$ is the n-tuple

$$\nabla f(\boldsymbol{x}) = \left(\frac{\partial f(\boldsymbol{x})}{\partial x_1}, \frac{\partial f(\boldsymbol{x})}{\partial x_2}, \ldots, \frac{\partial f(\boldsymbol{x})}{\partial x_n} \right).$$

We will always identify an n-tuple with the corresponding column vector[3]. Of course, the gradient only exists if all the partial derivatives exist. Second order information is contained in a matrix: assuming f has second order partial derivatives we define the *Hessian matrix*[4] $\nabla^2 f(\boldsymbol{x})$ as the $n \times n$ matrix whose (i, j)'th entry is

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

If these second order partial derivatives are continuous, then we may switch the order in the derivations, and $\nabla^2 f(x)$ is a symmetric matrix.

[2] See Section 7.2 in [7]

[3] This is somewhat different from [8], since the gradient there is always considered as a row vector

[4] See Section 5.9 in [8]

For vector-valued functions we also need the derivative. Consider the vector-valued function F given by

$$\boldsymbol{F}(\boldsymbol{x}) = \begin{bmatrix} F_1(\boldsymbol{x}) \\ F_2(\boldsymbol{x}) \\ \vdots \\ F_n(\boldsymbol{x}) \end{bmatrix}$$

so $F_i : \mathbb{R}^n \to \mathbb{R}$ is the ith component function of $\boldsymbol{F}$. $\boldsymbol{F}'$ denotes the *Jacobi matrix*[5], or simply the *derivative*, of $\boldsymbol{F}$

$$\boldsymbol{F}'(x) = \begin{bmatrix} \frac{\partial F_1(\boldsymbol{x})}{\partial x_1} & \frac{\partial F_1(\boldsymbol{x})}{\partial x_2} & \dots & \frac{\partial F_1(\boldsymbol{x})}{\partial x_n} \\ \frac{\partial F_2(\boldsymbol{x})}{\partial x_1} & \frac{\partial F_2(\boldsymbol{x})}{\partial x_2} & \dots & \frac{\partial F_1(\boldsymbol{x})}{\partial x_n} \\ & & \vdots & \\ \frac{\partial F_n(\boldsymbol{x})}{\partial x_1} & \frac{\partial F_n(\boldsymbol{x})}{\partial x_2} & \dots & \frac{\partial F_n(\boldsymbol{x})}{\partial x_n} \end{bmatrix}$$

The ith row of this matrix is therefore the gradient of F_i, now viewed as a row vector.

Next we recall Taylor's theorems from multivariate calculus [6]:

Theorem 9.3 (First order Taylor theorem)**.** Let $f : \mathbb{R}^n \to \mathbb{R}$ be a function having continuous partial derivatives in some ball $B(\boldsymbol{x}; r)$. Then, for each $\boldsymbol{h} \in \mathbb{R}^n$ with $\|\boldsymbol{h}\| < r$ there is some $t \in (0, 1)$ such that

$$f(\boldsymbol{x} + \boldsymbol{h}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x} + t\boldsymbol{h})^T \boldsymbol{h}.$$

The next one is known as Taylor's formula, or the second order Taylor's theorem[7]:

Theorem 9.4 (Second order Taylor theorem)**.** Let $f : \mathbb{R}^n \to \mathbb{R}$ be a function having second order partial derivatives that are continuous in some ball $B(\boldsymbol{x}; r)$. Then, for each $\boldsymbol{h} \in \mathbb{R}^n$ with $\|\boldsymbol{h}\| < r$ there is some $t \in (0, 1)$ such that

$$f(\boldsymbol{x} + \boldsymbol{h}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T \boldsymbol{h} + \frac{1}{2} \boldsymbol{h}^T \nabla^2 f(\boldsymbol{x} + t\boldsymbol{h}) \boldsymbol{h}.$$

This may be shown by considering the one-variable function $g(t) = f(\boldsymbol{x} + t\boldsymbol{h})$ and applying the chain rule and Taylor's formula in one variable.

[5] See Section 2.6 in [8]

[6] This theorem is also the mean value theorem of functions in several variables, see Section 5.5 in [8]

[7] See Section 5.9 in [8]

There is another version of the second order Taylor theorem in which the Hessian is evaluated in $\boldsymbol{x}$ and, as a result, we get an error term. This theorem shows how f may be approximated by a quadratic polynomial in n variables[8]:

Theorem 9.5 (Second order Taylor theorem, version 2)**.** Let $f : \mathbb{R}^n \to \mathbb{R}$ be a function having second order partial derivatives that are continuous in some ball $B(\boldsymbol{x}; r)$. Then there is a function $\epsilon : \mathbb{R}^n \to \mathbb{R}$ such that, for each $\boldsymbol{h} \in \mathbb{R}^n$ with $\|\boldsymbol{h}\| < r$,

$$f(\boldsymbol{x}+\boldsymbol{h}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T\boldsymbol{h} + \frac{1}{2}\boldsymbol{h}^T\nabla^2 f(\boldsymbol{x})\boldsymbol{h} + \epsilon(\boldsymbol{h})\|\boldsymbol{h}\|^2.$$

Here $\epsilon(\boldsymbol{y}) \to \boldsymbol{0}$ when $\boldsymbol{y} \to \boldsymbol{0}$.

Using the O-notation from Definition 4.6, the very useful approximations we get from Taylor' theorems can thus be summarized as follows:

Taylor approximations:

First order:	$f(\boldsymbol{x}+\boldsymbol{h})$	$= f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T\boldsymbol{h} + O(\|\boldsymbol{h}\|)$
		$\approx f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T\boldsymbol{h}.$
Second order:	$f(\boldsymbol{x}+\boldsymbol{h})$	$= f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T\boldsymbol{h} + \frac{1}{2}\boldsymbol{h}^T\nabla^2 f(\boldsymbol{x})\boldsymbol{h} + O(\|\boldsymbol{h}\|^2)$
		$\approx f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T\boldsymbol{h} + \frac{1}{2}\boldsymbol{h}^T\nabla^2 f(\boldsymbol{x})\boldsymbol{h}.$

We introduce notation for these approximations

$$\begin{aligned} T_f^1(\boldsymbol{x}; \boldsymbol{x}+\boldsymbol{h}) &= f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T\boldsymbol{h} \\ T_f^2(\boldsymbol{x}; \boldsymbol{x}+\boldsymbol{h}) &= f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T\boldsymbol{h} + \tfrac{1}{2}\boldsymbol{h}^T\nabla^2 f(\boldsymbol{x})\boldsymbol{h} \end{aligned}$$

As we shall see, one can get a lot of optimization out of these approximations!

We also need a Taylor theorem for vector-valued functions, which follows by applying Taylor' theorem above to each component function:

Theorem 9.6 (First order Taylor theorem for vector-valued functions)**.** Let $\boldsymbol{F} : \mathbb{R}^n \to \mathbb{R}^m$ be a vector-valued function which is continuously differentiable in a neighborhood N of $\boldsymbol{x}$. Then

$$\boldsymbol{F}(\boldsymbol{x}+\boldsymbol{h}) = \boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{F}'(\boldsymbol{x})\boldsymbol{h} + O(\|\boldsymbol{h}\|)$$

when $\boldsymbol{x}+\boldsymbol{h} \in N$.

[8] See Section 5.9 in [8]

Finally, if $\boldsymbol{F} : \mathbb{R}^n \to \mathbb{R}^m$ and $\boldsymbol{G} : \mathbb{R}^k \to \mathbb{R}^n$ we define the *composition* $\boldsymbol{H} = \boldsymbol{F} \circ \boldsymbol{G}$ as the function $\boldsymbol{H} : \mathbb{R}^k \to \mathbb{R}^m$ by $\boldsymbol{H}(\boldsymbol{x}) = \boldsymbol{F}(\boldsymbol{G}(\boldsymbol{x}))$. Then, under natural differentiability assumption the following *chain rule*[9] holds:

$$\boldsymbol{H}'(\boldsymbol{x}) = \boldsymbol{F}'(\boldsymbol{G}(\boldsymbol{x}))\boldsymbol{G}'(\boldsymbol{x}).$$

Here the right-hand side is a product of two matrices, the respective Jacobi matrices evaluated in the right points.

Finally, we discuss some notions concerning the convergence of sequences.

Definition 9.7 (Linear convergence)**.** We say that a sequence $\{\boldsymbol{x}_k\}_{k=1}^{\infty}$ converges to $\boldsymbol{x}^*$ *linearly* (or that the convergence speed in linear) if there is a $\gamma < 1$ such that

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \leq \gamma \|\boldsymbol{x}_k - \boldsymbol{x}^*\| \quad (k = 0, 1, \ldots).$$

A faster convergence rate is *superlinear convergence* which means that

$$\lim_{k\to\infty} \|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| / \|\boldsymbol{x}_k - \boldsymbol{x}^*\| = 0$$

A special type of superlinear convergence is *quadratic convergence* where

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \leq \gamma \|\boldsymbol{x}_k - \boldsymbol{x}^*\|^2 \quad (k = 0, 1, \ldots)$$

for some $\gamma < 1$.

Exercises for Section 9.3

Ex. 1 — Give an example of a function $f : \mathbb{R} \to \mathbb{R}$ with 10 global minima.

Ex. 2 — Consider the function $f(x) = x \sin(1/x)$ defined for $x > 0$. Find its local minima. What about global minimum?

Ex. 3 — Let $f : X \to \mathbb{R}_+$ be a function (with nonnegative function values). Explain why it is equivalent to minimize f over $x \in X$ or minimize $f^2(x)$ over X.

Ex. 4 — Consider $f : \mathbb{R}^2 \to \mathbb{R}$ given by $f(\boldsymbol{x}) = (x_1 - 3)^2 + (x_2 - 2)^2$. How would you explain to *anyone* that $\boldsymbol{x^* = (3, 2)}$ is a minimum point?

[9] See Section 2.7 in [8]

Ex. 5 — The *level sets* of a function $f : \mathbb{R}^2 \to \mathbb{R}$ are sets of the form $L_\alpha = \boldsymbol{x} \in \mathbb{R}^2 : f(\boldsymbol{x}) = \alpha\}$. Let $f(\boldsymbol{x}) = (1/4)(x-1)^2 + (x-3)^2$. Draw the level sets in the plane for $\alpha = 10, 5, 1, 0.1$.

Ex. 6 — The *sublevel set* of a function $f : \mathbb{R}^n \to \mathbb{R}$ is the set $S_\alpha(f) = \{\boldsymbol{x} \in \mathbb{R}^2 : f(\boldsymbol{x}) \leq \alpha\}$, where $\alpha \in \mathbb{R}$. Assume that $\inf\{f(\boldsymbol{x}) : \boldsymbol{x} \in \mathbb{R}^n\} = \eta$ exists.

a. What happens to the sublevel sets S_α as α decreases? Give an example.

b. Show that if f is continuous and there is an $\boldsymbol{x'}$ such that with $\alpha = f(\boldsymbol{x'})$ the sublevel set $S_\alpha(f)$ is bounded, then f attains its minimum.

Ex. 7 — Consider the portfolio optimization problem in Subsection 9.2.1.

a. Assume that $c_{ij} = 0$ for each $i \neq j$. Find, analytically, an optimal solution. Describe the set of all optimal solutions.

b. Consider the special case where $n = 2$. Solve the problem (hint: eliminate one variable) and discuss how minimum point depends on α.

Ex. 8 — Later in these notes we will need the expression for the gradient of functions which are expressed in terms of matrices.

a. Let $f : \mathbb{R}^n \to \mathbb{R}$ be defined by $f(\boldsymbol{x}) = \boldsymbol{q}^T\boldsymbol{x} = \boldsymbol{x}^T\boldsymbol{q}$, where $\boldsymbol{q}$ is a vector. Show that $\nabla f(\boldsymbol{x}) = \boldsymbol{q}$, and that $\nabla^2 f(\boldsymbol{x}) = \boldsymbol{0}$.

b. Let $f : \mathbb{R}^n \to \mathbb{R}$ be the quadratic function $f(\boldsymbol{x}) = (1/2)\boldsymbol{x}^T A\boldsymbol{x}$. Show that $\nabla f(\boldsymbol{x}) = A\boldsymbol{x}$, and that $\nabla^2 f(\boldsymbol{x}) = A$.

Ex. 9 — Consider $f(\boldsymbol{x}) = f(x_1, x_2) = x_1^2 + 3x_1x_2 - 5x_2^2 + 3$. Determine the first order Taylor approximation to f at each of the points $(0, 0)$ and $(2, 1)$.

Ex. 10 — Let $A = \begin{bmatrix} 1 & 2 \\ 2 & 8 \end{bmatrix}$. Show that A is positive definite. (Try to give two different proofs.)

Ex. 11 — Show that is A is positive definite, then its inverse is also positive definite.

Chapter 10

A crash course in convexity

Convexity is a branch of mathematical analysis dealing with convex sets and convex functions. It also represents a foundation for optimization.

We just summarize concepts and some results. For proofs one may consult [4] or [14], see also [1].

10.1 Convex sets

A set $C \subseteq \mathbb{R}^n$ is called *convex* if $(1-\lambda)\boldsymbol{x} + \lambda\boldsymbol{y} \in C$ whenever $\boldsymbol{x}, \boldsymbol{y} \in C$ and $0 \leq \lambda \leq 1$. Geometrically, this means that C contains the line segment between each pair of points in C, so, loosely speaking, a convex set contains no "holes".

For instance, the ball $B(\boldsymbol{a}; \delta) = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x} - \boldsymbol{a}\| \leq \delta\}$ is a convex set. Let us show this. Recall the triangle inequality which says that $\|\boldsymbol{u}+\boldsymbol{v}\| \leq \|\boldsymbol{u}\|+\|\boldsymbol{v}\|$ whenever $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$. Let $\boldsymbol{x}, \boldsymbol{y} \in B(\boldsymbol{a}; \delta)$ and $\lambda \in [0, 1]$. Then

$$\begin{aligned} \|((1-\lambda)\boldsymbol{x} + \lambda\boldsymbol{y}) - \boldsymbol{a}\| &= \|(1-\lambda)(\boldsymbol{x} - \boldsymbol{a}) + \lambda(\boldsymbol{y} - \boldsymbol{a})\| \\ &\leq \|(1-\lambda)(\boldsymbol{x} - \boldsymbol{a})\| + \|\lambda(\boldsymbol{y} - \boldsymbol{a})\| \\ &= (1-\lambda)\|\boldsymbol{x} - \boldsymbol{a}\| + \lambda\|\boldsymbol{y} - \boldsymbol{a}\| \\ &\leq (1-\lambda)\delta + \lambda\delta = \delta. \end{aligned}$$

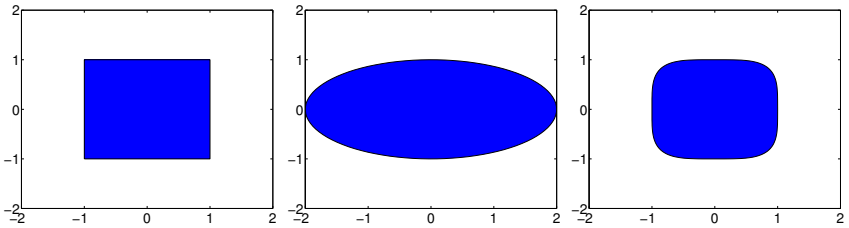
Therefore $B(\boldsymbol{a}; \delta)$ is convex.

Every linear subspace is also a convex set, as well as the translate of every subspace (which is called an affine set). Some other examples of convex sets in $\mathbb{R}^2$ are shown in Figure 10.1. We will come back to why each of these sets are convex later. Another important property is that the intersection of a family of convex sets is a convex set.

By a *linear system* we mean a finite system of linear equations and/or linear inequalities involving n variables. For example

$$x_1 + x_2 = 3,\ x_1 \geq 0,\ x_2 \geq 0$$

is a linear system in the variables x_1, x_2. The solution set is the set of points $(x_1, 3 - x_1)$ where $0 \leq x_1 \leq 3$. The set of solutions of a linear system is called



(a) A square (b) The ellipse $\frac{x^2}{4} + y^2 \leq 1$ (c) The area $x^4 + y^4 \leq 1$

Figure 10.1: Examples of some convex sets.

a *polyhedron*. These sets often occur in optimization. Thus, a polyhedron has the form

$$P = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} \leq \boldsymbol{b}\}$$

where $A \in \mathbb{R}^{m,n}$ and $\boldsymbol{b} \in \mathbb{R}^m$ (m is arbitrary, but finite) and $\leq$ means componentwise inequality. There are simple techniques for rewriting any linear system in the form $A\boldsymbol{x} \leq \boldsymbol{b}$.

Proposition 10.1. Every polyhedron is a convex set.

The square from Figure 10.1(a) is defined by the inequalities $-1 \leq x, y \leq 1$. It is therefore a polyhedron, and therefore convex. The next result shows that convex sets are preserved under linear maps.

Proposition 10.2. If $T : \mathbb{R}^n \to \mathbb{R}^m$ is a linear transformation, and $C \subseteq \mathbb{R}^n$ is a convex set, then the image $T(C)$ of this set is also convex.

10.2 Convex functions

The notion of a convex function also makes sense for real-valued functions of several variables. Consider a real-valued function $f : C \to \mathbb{R}$ where $C \subseteq \mathbb{R}^n$ is a convex set. We say that f is *convex* provided that

$$f((1-\lambda)\boldsymbol{x} + \lambda\boldsymbol{y}) \leq (1-\lambda)f(\boldsymbol{x}) + \lambda f(\boldsymbol{y}) \quad (\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n,\, 0 \leq \lambda \leq 1) \qquad (10.1)$$

(This inequality holds for all $\boldsymbol{x}, \boldsymbol{y}$ and λ as specified). Due to the convexity of C, the point $(1-\lambda)\boldsymbol{x} + \lambda\boldsymbol{y}$ lies in C, so the inequality is well-defined. The geometrical interpretation in one dimension is that whenever you take two points on the graph of f, say $(x, f(x))$ and $(y, f(y))$, the graph of f restricted to the line segment $[x, y]$ lies below the line segment in $\mathbb{R}^{n+1}$ between the two chosen points. A function g is called *concave* if $-g$ is convex.

Every linear function is convex. Some other examples of convex functions in n variables are

- $f(\boldsymbol{x}) = L(\boldsymbol{x}) + \alpha$ where L is a linear function from $\mathbb{R}^n$ into $\mathbb{R}$ (a linear functional) and α is a real number. Such a function is called an *affine function* and it may be written $f(\boldsymbol{x}) = \boldsymbol{c}^T\boldsymbol{x} + \alpha$ for a suitable vector $\boldsymbol{c}$.
- $f(\boldsymbol{x}) = \|\boldsymbol{x}\|$ (Euclidean norm). That this is convex can be proved by writing $\|(1-\lambda)\boldsymbol{x} + \lambda\boldsymbol{y}\| \leq \|(1-\lambda)\boldsymbol{x}\| + \|\lambda\boldsymbol{y}\| = (1-\lambda)\|\boldsymbol{x}\| + \lambda\|\boldsymbol{y}\|$. In fact, the same argument can be used to show that *every* norm defines a convex function. Such an example is the l_1-norm, also called the *sum norm*, defined by $\|\boldsymbol{x}\|_1 = \sum_{j=1}^n |x_j|$.
- $f(\boldsymbol{x}) = e^{\sum_{j=1}^n x_j}$ (see Exercise 7).
- $f(\boldsymbol{x}) = e^{h(\boldsymbol{x})}$ where $h : \mathbb{R}^n \to \mathbb{R}$ is a convex function.
- $f(\boldsymbol{x}) = \max_i g_i(\boldsymbol{x})$ where $g_i : \mathbb{R}^n \to \mathbb{R}$ is an affine function ($i \leq m$). This means that the pointwise maximum of affine functions is a convex function. Note that such convex functions are typically not differentiable everywhere. A more general result is that the pointwise supremum of an arbitrary family of affine functions (or even convex functions) is convex. This is a very useful fact in convexity and its applications.

The following result is an exercise to prove, and it gives a method for proving convexity of a function.

Proposition 10.3. Assume that $f : \mathbb{R}^n \to \mathbb{R}$ is convex and $\boldsymbol{H} : \mathbb{R}^m \to \mathbb{R}^n$ is affine. Then the composition $f \circ \boldsymbol{H}$ is convex, where $(f \circ \boldsymbol{H})(\boldsymbol{x}) := f(\boldsymbol{H}(\boldsymbol{x}))$.

The next result is often used, and is called *Jensen's inequality.*. It can be shown using induction.

Theorem 10.4 (Jensen's inequality)**.** Let $f : C \to \mathbb{R}$ be a convex function defined on a convex set $C \subseteq \mathbb{R}^n$. If $\boldsymbol{x}^1, \boldsymbol{x}^2 \ldots, \boldsymbol{x}^r \in C$ and $\lambda_1, \ldots, \lambda_r \geq 0$ satisfy $\sum_{j=1}^r \lambda_j = 1$, then

$$f(\sum_{j=1}^r \lambda_j \boldsymbol{x}^j) \leq \sum_{j=1}^r \lambda_j f(\boldsymbol{x}^j). \tag{10.2}$$

A point of the form $\sum_{j=1}^r \lambda_j \boldsymbol{x}^j$, where the λ_j's are nonnegative and sum to 1, is called a *convex combination* of the points $\boldsymbol{x}^1, \boldsymbol{x}^2 \ldots, \boldsymbol{x}^r$. One can show that a set is convex if and only if it contains all convex combinations of its points.

Finally, one connection between convex sets and convex functions is the following fact whose proof is an exercise.

Proposition 10.5. Let $C \subseteq \mathbb{R}^n$ be a convex set and consider a convex function $f : C \to \mathbb{R}$. Let $\alpha \in \mathbb{R}$. Then the "sublevel" set

$$\{\boldsymbol{x} \in C : f(\boldsymbol{x}) \leq \alpha\}$$

is a convex set.

10.3 Properties of convex functions

A convex function may not be differentiable in every point. However, one can show that a convex function always has one-sided directional derivatives at any point. But what about continuity?

Theorem 10.6. Let $f : C \to \mathbb{R}$ be a convex function defined on an open convex set $C \subseteq \mathbb{R}^n$. Then f is continuous on C.

However, a convex function may be discontinuous in points on the *boundary* of its domain. For instance, the function $f : [0, 1] \to \mathbb{R}$ given by $f(0) = 1$ and $f(x) = 0$ for $x \in (0, 1]$ is convex, but discontinuous at $x = 0$. Next we give a useful technique for checking that a function is convex.

Theorem 10.7. Let f be a real-valued function defined on an open convex set $C \subseteq \mathbb{R}^n$ and assume that f has continuous second-order partial derivatives on C.

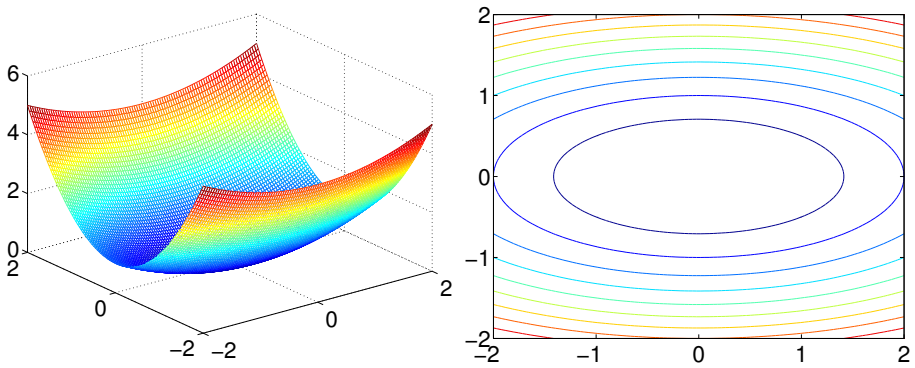
Then f is convex if and only if the Hessian matrix $\nabla^2 f(\boldsymbol{x})$ is positive semidefinite for each $\boldsymbol{x} \in C$.

With this result it is straightforward to prove that the remaining sets from Figure 10.1 are convex. They can be written as sublevel sets of the functions $f(x, y) = \frac{x^2}{4} + y^2$, and $f(x, y) = x^4 + y^4$. For the first of these the level sets are ellipses, and are shown in Figure 10.2, together with f itself. One can quickly verify that the Hessian matrices of these functions are positive semidefinite. It follows from Proposition 10.5 that the corresponding sets are convex.

An important class of convex functions consists of (certain) quadratic functions. Let $A \in \mathbb{R}^{n \times n}$ be a symmetric matrix which is positive semidefinite and consider the quadratic function $f : \mathbb{R}^n \to \mathbb{R}$ given by

$$f(\boldsymbol{x}) = (1/2)\, \boldsymbol{x}^T A \boldsymbol{x} - \boldsymbol{b}^T \boldsymbol{x} = (1/2) \sum_{i,j} a_{ij} x_i x_j - \sum_{j=1}^{n} b_j x_j.$$

(If $A = \mathbf{0}$, then the function is linear, and it may be strange to call it quadratic. But we still do this, for simplicity.) Then (Exercise 9.3.8) the Hessian matrix of f is A, i.e., $\nabla^2 f(\boldsymbol{x}) = A$ for each $\boldsymbol{x} \in \mathbb{R}^n$. Therefore, by Theorem 10.7 is a convex function.



(a) The function $f(x, y) = \frac{x^2}{4} + y^2$ (b) Some level curves of f

Figure 10.2: A function and its level curves.

We remark that sometimes it may be easy to check that a symmetric matrix A is positive semidefinite. A (real) symmetric $n \times n$ matrix A is called *diagonally dominant* if $|a_{ii}| \geq \sum_{j \neq i} |a_{ij}|$ for $i = 1, \ldots, n$. These matrices arise in many applications, e.g. splines and differential equations. It can be shown that every symmetric diagonally dominant matrix is positive semidefinite. For a simple proof of this fact using convexity, see [3]. Thus, we get a simple criterion for convexity of a function: check if the Hessian matrix $\nabla^2 f(\boldsymbol{x})$ is diagonally dominant for each $\boldsymbol{x}$. Be careful here: this matrix may be positive semidefinite without being diagonally dominant!

We now look at differentiability properties of convex functions.

Theorem 10.8. Let f be a real-valued convex function defined on an open convex set $C \subseteq \mathbb{R}^n$. Assume that all the partial derivatives $\partial f(\boldsymbol{x})/\partial x_1, \ldots, \partial f(\boldsymbol{x})/\partial x_n$ exist at a point $\boldsymbol{x} \in C$. Then f is differentiable at $\boldsymbol{x}$.

A convex function may not be differentiable everywhere, but it is differentiable "almost everywhere". More precisely, for a convex function defined on an open convex set in $\mathbb{R}^n$, the set of points for which f is not differentiable has Lebesgue measure zero. We do not go into further details on this here, but refer to e.g. [5] for a proof and a discussion.

Another characterization of convex functions that involves the gradient may now be presented.

Theorem 10.9. Let $f : C \to \mathbb{R}$ be a differentiable function defined on an open convex set $C \subseteq \mathbb{R}^n$. Then the following conditions are equivalent:

(i)	f is convex.	
(ii)	$f(\boldsymbol{x}) \geq f(\boldsymbol{x}_0) + \nabla f(\boldsymbol{x}_0)^T(\boldsymbol{x} - \boldsymbol{x}_0)$	for all $\boldsymbol{x}, \boldsymbol{x}_0 \in C$.
(iii)	$(\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{x}_0))^T(\boldsymbol{x} - \boldsymbol{x}_0) \geq 0$	for all $\boldsymbol{x}, \boldsymbol{x}_0 \in C$.

This theorem is important. Property (ii) says that the first-order Taylor approximation of f at x_0 (which is the right-hand side of the inequality) always underestimates f. This result has interesting consequences for optimization as we shall see later.

Exercises for section 10.3

Ex. 1 — Let $S = \{(x, y, z) : z \geq x^2 + y^2\} \subset \mathbb{R}^3$. Sketch the set and verify that it is a convex set.

Ex. 2 — Let $f : S \to \mathbb{R}$ be a differentiable function, where S is an open set in $\mathbb{R}$. Check that f is convex if and only if $f''(x) \geq 0$ for all $x \in S$.

Ex. 3 — Prove Proposition 10.3.

Ex. 4 — Prove Proposition 10.5.

Ex. 5 — Explain how you can write the LP problem $\max\{\boldsymbol{c}^T\boldsymbol{x} : A\boldsymbol{x} \geq \boldsymbol{b},\ B\boldsymbol{x} = \boldsymbol{d},\ \boldsymbol{x} \geq \boldsymbol{0}\}$ as an LP problem of the form

$$\max\{\boldsymbol{c}^T\boldsymbol{x} : H\boldsymbol{x} \leq \boldsymbol{h},\ \boldsymbol{x} \geq \boldsymbol{0}\}$$

for suitable matrix H and vector $\boldsymbol{h}$.

Ex. 6 — Let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t \in \mathbb{R}^n$ and let C be the set of vectors of the form

$$\sum_{j=1}^{t} \lambda_j \boldsymbol{x}_j$$

where $\lambda_j \geq 0$ for each $j = 1, \ldots, t$. Show that C is convex. Make a sketch of such a set in $\mathbb{R}^3$.

Ex. 7 — Show that $f(\boldsymbol{x}) = e^{\sum_{j=1}^{n} x_j}$ is a convex function.

Ex. 8 — Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex function and let $\alpha \in \mathbb{R}$. Show that the sublevel set $S_\alpha(f) = \{\boldsymbol{x} \in \mathbb{R}^n : f(\boldsymbol{x}) \leq \alpha\}$ is a convex set.

Ex. 9 — Assume that f and g are convex functions defined on an interval I. Determine which of the functions following functions that are convex or concave:

a. λf where $\lambda \in \mathbb{R}$,
b. $\min\{f, g\}$,
c. $|f|$.

Ex. 10 — Let $f : [a, b] \to \mathbb{R}$ be a convex function. Show that

$$\max\{f(x) : x \in [a, b]\} = \max\{f(a), f(b)\}$$

i.e., a convex function defined on closed real interval attains its maximum in one of the endpoints.

Ex. 11 — Let $f : \langle 0, \infty\rangle \to \mathbb{R}$ and define the function $g : \langle 0, \infty\rangle \to \mathbb{R}$ by $g(x) = xf(1/x)$. Why is the function $x \to xe^{1/x}$ convex?

Ex. 12 — Let $C \subseteq \mathbb{R}^n$ be a convex set and consider the distance function d_C defined by $d_C(x) = \inf\{\|x - y\| : y \in C\}$. Show that d_C is a convex function.

Chapter 11

Nonlinear equations

A basic mathematical problem is to solve a system of equations in several unknowns (variables). There are numerical methods that can solve such equations, at least within a small error tolerance. We shall briefly discuss such methods here; for further details, see [6, 11].

11.1 Equations and fixed points

In linear algebra one works a lot with linear equations in several variables, and Gaussian elimination is a central method for solving such equations. There are also other faster methods, so-called iterative methods, for linear equations. But what about *nonlinear equations*? For instance, consider the system in two variables x_1 and x_2:

$$\begin{aligned} x_1^2 - x_1 x_2^{-3} + \cos x_1 &= 1 \\ 5x_1^4 + 2x_1^3 - \tan(x_1 x_2^8) &= 3 \end{aligned}$$

Clearly, such equations can be very hard to solve. The general problem is to solve the equation

$$\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0} \tag{11.1}$$

for a given function $\boldsymbol{F} : \mathbb{R}^n \to \mathbb{R}^n$. If $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}$ we call $\boldsymbol{x}$ a *root* of $\boldsymbol{F}$ (or of the equation). The example above is equivalent to finding roots in $\boldsymbol{F}(\boldsymbol{x}) = (F_1(\boldsymbol{x}), F_2(\boldsymbol{x}))$ where

$$\begin{aligned} F_1(\boldsymbol{x}) &= x_1^2 - x_1 x_2^{-3} + \cos x_1 - 1 \\ F_2(\boldsymbol{x}) &= 5x_1^4 + 2x_1^3 - \tan(x_1 x_2^8) - 3 \end{aligned}$$

In particular, if $\boldsymbol{F}(\boldsymbol{x}) = A\boldsymbol{x} - \boldsymbol{b}$ where A is an $n \times n$ matrix and $\boldsymbol{b} \in \mathbb{R}^n$, then we are back to linear equations (a square system). More generally one may consider equations $\boldsymbol{G}(\boldsymbol{x}) = \boldsymbol{0}$ where $\boldsymbol{G} : \mathbb{R}^n \to \mathbb{R}^m$, but we here only discuss the case $m = n$.

Often the problem $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}$ has the following form, or may be rewritten to it:

$$\boldsymbol{K}(\boldsymbol{x}) = \boldsymbol{x}. \tag{11.2}$$

for some function $\boldsymbol{K} : \mathbb{R}^n \to \mathbb{R}^n$. This corresponds to the special choice $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{K}(\boldsymbol{x}) - \boldsymbol{x}$. A point $\boldsymbol{x} \in \mathbb{R}^n$ such that $\boldsymbol{x} = \boldsymbol{K}(\boldsymbol{x})$ is called a *fixed point* of the function $\boldsymbol{K}$. In finding such a fixed point it is tempting to use the following iterative method: choose a starting point $\boldsymbol{x}_0$ and repeat the following iteration

$$\boldsymbol{x}_{k+1} = \boldsymbol{K}(\boldsymbol{x}_k) \quad \text{for } k = 1, 2, \ldots \tag{11.3}$$

This is called a *fixed-point iteration*. We note that if $\boldsymbol{K}$ is continuous and this procedure converges to some point $\boldsymbol{x}^*$, then $\boldsymbol{x}^*$ must be a fixed point. The fixed-point iteration is an extremely simple algorithm, and very easy to implement. Perhaps surprisingly, it also works very well for many such problems. Let $\epsilon > 0$ denote a small error tolerance used for stopping the process, e.g. 10^{-6}.

Fixed-point algorithm:
1. Choose an initial point $\boldsymbol{x}_0$, let $\boldsymbol{x} = \boldsymbol{x}_0$ and err $= 1$.
2. while err $> \epsilon$ do
   - (i) Compute $\boldsymbol{x}_1 = \boldsymbol{K}(\boldsymbol{x})$
   - (ii) Compute err $= \|\boldsymbol{x}_1 - \boldsymbol{x}\|$
   - (iii) Update $\boldsymbol{x} := \boldsymbol{x}_1$

When does the fixed-point iteration work? Let $\|\cdot\|$ be a fixed norm, e.g. the Eulidean norm, on $\mathbb{R}^n$. We say that the function $\boldsymbol{K} : \mathbb{R}^n \to \mathbb{R}^n$ is a *contraction* if there is a constant $0 \leq c < 1$ such that

$$\|\boldsymbol{K}(\boldsymbol{x}) - \boldsymbol{K}(\boldsymbol{y})\| \leq c\|\boldsymbol{x} - \boldsymbol{y}\| \quad (\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n).$$

We also say that $\boldsymbol{K}$ is *c-Lipschitz* in this case. The following theorem is called the *Banach contraction principle*. It also holds in Banach spaces, i.e., complete normed vector spaces (possibly infinite-dimensional).

Theorem 11.1. Assume that $\boldsymbol{K}$ is c-Lipschitz with $0 < c < 1$. Then $\boldsymbol{K}$ has a unique fixed point $\boldsymbol{x}^*$. For any starting point $\boldsymbol{x}_0$ the fixed-point iteration (11.3) generates a sequence $\{\boldsymbol{x}_k\}_{k=0}^{\infty}$ that converges to $\boldsymbol{x}^*$. Moreover

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| \leq c\|\boldsymbol{x}_k - \boldsymbol{x}^*\| \text{ for } k = 0, 1, \ldots \tag{11.4}$$

so that

$$\|\boldsymbol{x}_k - \boldsymbol{x}^*\| \leq c^k\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|.$$

Proof. First, note that if both $\boldsymbol{x}$ and $\boldsymbol{y}$ are fixed points of $\boldsymbol{K}$, then

$$\|\boldsymbol{x} - \boldsymbol{y}\| = \|\boldsymbol{K}(\boldsymbol{x}) - \boldsymbol{K}(\boldsymbol{y})\| \leq c\|\boldsymbol{x} - \boldsymbol{y}\|$$

which means that $\boldsymbol{x} = \boldsymbol{y}$ (as $c < 1$); therefore $\boldsymbol{K}$ has at most one fixed point. Next, we compute

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\| = \|\boldsymbol{K}(\boldsymbol{x}_k) - \boldsymbol{K}(\boldsymbol{x}_{k-1})\| \leq c\|\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\| = \cdots \leq c^k\|\boldsymbol{x}_1 - \boldsymbol{x}_0\|$$

so

$$\begin{aligned}\|\boldsymbol{x}_m - \boldsymbol{x}_0\| &= \|\textstyle\sum_{k=0}^{m-1}(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k)\| \leq \sum_{k=0}^{m-1}\|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k\| \\ &\leq (\textstyle\sum_{k=0}^{n-1} c^k)\|\boldsymbol{x}_1 - \boldsymbol{x}_0\| \leq (1/(1-c))\|\boldsymbol{x}_1 - \boldsymbol{x}_0\|\end{aligned}$$

From this we derive that $\{\boldsymbol{x}_k\}$ is a Cauchy sequence; as we have

$$\begin{aligned}\|\boldsymbol{x}_{s+m} - \boldsymbol{x}_s\| &= \|\boldsymbol{K}(\boldsymbol{x}_{s+m-1}) - \boldsymbol{K}(\boldsymbol{x}_{s-1})\| \leq c\|\boldsymbol{x}_{s+m-1} - \boldsymbol{x}_{s-1}\| = \cdots \\ &\leq c^s\|\boldsymbol{x}_m - \boldsymbol{x}_0\| \leq (c^s/(1-c))\|\boldsymbol{x}_1 - \boldsymbol{x}_0\|.\end{aligned}$$

and $0 < c < 1$. Any Cauchy sequence in $\mathbb{R}^n$ has a limit point, so $\boldsymbol{x}_m \to \boldsymbol{x}^*$ for some $\boldsymbol{x}^* \in \mathbb{R}^n$. We now prove that the limit point $\boldsymbol{x}^*$ is a (actually, the) fixed point:

$$\begin{aligned}\|\boldsymbol{x}^* - \boldsymbol{K}(\boldsymbol{x}^*)\| &\leq \|\boldsymbol{x}^* - \boldsymbol{x}_m\| + \|\boldsymbol{x}_m - \boldsymbol{K}(\boldsymbol{x}^*)\| \\ &= \|\boldsymbol{x}^* - \boldsymbol{x}_m\| + \|\boldsymbol{K}(\boldsymbol{x}_{m-1}) - \boldsymbol{K}(\boldsymbol{x}^*)\| \\ &\leq \|\boldsymbol{x}^* - \boldsymbol{x}_m\| + c\|\boldsymbol{x}_{m-1} - \boldsymbol{x}^*\|\end{aligned}$$

and letting $m \to \infty$ here gives $\|\boldsymbol{x}^* - \boldsymbol{K}(\boldsymbol{x}^*)\| \leq 0$ so $\boldsymbol{x}^* = \boldsymbol{K}(\boldsymbol{x}^*)$ as desired.

Finally,

$$\|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\| = \|\boldsymbol{K}(\boldsymbol{x}_k) - \boldsymbol{K}(\boldsymbol{x}^*)\| \leq c\|\boldsymbol{x}_k - \boldsymbol{x}^*\| \leq c^{k+1}\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|$$

which completes the proof. □

We see that $\boldsymbol{x}_k \to \boldsymbol{x}^*$ *linearly*, and that Equation (11.4) gives an estimate on the convergence speed.

11.2 Newton's method

We return to the main problem (11.1). Our goal is to present Newton's method, a highly efficient iterative method for solving this equation. The method constructs a sequence

$$\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$$

in $\mathbb{R}^n$ which, hopefully, converges to a root $\boldsymbol{x}^*$ of $\boldsymbol{F}$, so $\boldsymbol{F}(\boldsymbol{x}^*) = \boldsymbol{0}$. The idea is to linearize $\boldsymbol{F}$ at the current iterate $\boldsymbol{x}_k$ and choose the next iterate $\boldsymbol{x}_{k+1}$ as a zero of this linearized function. The first order Taylor approximation of $\boldsymbol{F}$ at $\boldsymbol{x}_k$ is

$$T^1_{\boldsymbol{F}}(\boldsymbol{x}_k; \boldsymbol{x}) = \boldsymbol{F}(\boldsymbol{x}_k) + \boldsymbol{F}'(\boldsymbol{x}_k)(\boldsymbol{x} - \boldsymbol{x}_k).$$

We solve $T_{\boldsymbol{F}}^1(\boldsymbol{x}_k;\boldsymbol{x}) = \boldsymbol{0}$ for $\boldsymbol{x}$ and define the next iterate as $\boldsymbol{x}_{k+1} = \boldsymbol{x}$. This gives

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \boldsymbol{F}'(\boldsymbol{x}_k)^{-1}\boldsymbol{F}(\boldsymbol{x}_k) \tag{11.5}$$

which leads to Newton's method. One here assumes that the derivative $\boldsymbol{F}'$ is known analytically. Note that we do not (and hardly ever do!) compute the inverse of the matrix $\boldsymbol{F}'$.

Newton's method for nonlinear equations:
1. Choose an initial point $\boldsymbol{x}_0$.
2. For $k = 0, 1, \ldots$ do
   (i) Find the direction $\boldsymbol{p}$ by solving $\boldsymbol{F}'(\boldsymbol{x}_k)\boldsymbol{p} = -\boldsymbol{F}(\boldsymbol{x}_k)$
   (ii) Update: $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{p}$

In the main step, which is to compute $\boldsymbol{p}$, one needs to solve an $n \times n$ linear system of equations where the coefficient matrix is the Jacobi matrix of $\boldsymbol{F}$, evaluated at $\boldsymbol{x}_k$. In MAT1110 [8] we implemented the following code for Newton's method for nonlinear equations:

```
function x=newtonmult(x0,F,J)
  % Performs Newtons method in many variables
  % x: column vector which contains the start point
  % F: computes the values of F
  % J: computes the Jacobi matrix
  epsilon=0.0000001; N=30; n=0;
  x=x0;
  while norm(F(x)) > epsilon && n<=N
    x=x-J(x)\F(x);
    fval = F(x);
    fprintf('itnr=%2d x=[%13.10f,%13.10f] F(x)=[%13.10f,%13.10f]\n',...
            n,x(1),x(2),fval(1),fval(2))
    n = n + 1;
  end
```

This code also terminates after a given number of iterations, and when a given accuracy is obtained. Note that this function should work for any function $\boldsymbol{F}$, since it is a parameter to the function.

The convergence of Newton's method may be analyzed using fixed point theory since one may view Newton's method as a fixed point iteration. Observe that the Newton iteration (11.5) may be written

$$\boldsymbol{x}_{k+1} = \boldsymbol{G}(\boldsymbol{x}_k)$$

where $\boldsymbol{G}$ is the function

$$\boldsymbol{G}(\boldsymbol{x}) = \boldsymbol{x} - \boldsymbol{F}'(\boldsymbol{x})^{-1}\boldsymbol{F}(\boldsymbol{x})$$

From this it is possible to show that if the starting point is sufficiently close to the root, then Newton's method will converge to this root at a linear convergence rate. With more clever arguments one may show that the convergence rate of Newton's method is even faster: it has superlinear convergence. Actually, for many functions one even has quadratic convergence rate. The proof of the following convergence theorem relies purely on Taylor's theorem.

Theorem 11.2. Assume that Newton's method with initial point $\boldsymbol{x}_0$ produces a sequence $\{\boldsymbol{x}_k\}_{k=0}^{\infty}$ which converges to a solution $\boldsymbol{x}^*$ of (11.1). Then the convergence rate is superlinear.

Proof. From Taylor's theorem for vector-valued functions, Theorem 9.6, in the point $\boldsymbol{x}_k$ we have

$$\boldsymbol{0} = \boldsymbol{F}(\boldsymbol{x}^*) = \boldsymbol{F}(\boldsymbol{x}_k + (\boldsymbol{x}^* - \boldsymbol{x}_k)) = \boldsymbol{F}(\boldsymbol{x}_k) + \boldsymbol{F}'(\boldsymbol{x}_k)(\boldsymbol{x}^* - \boldsymbol{x}_k) + O(\|\boldsymbol{x}_k - \boldsymbol{x}^*\|)$$

Multiplying this equation by $\boldsymbol{F}'(\boldsymbol{x}_k)^{-1}$ (which is assumed to exist!) gives

$$\boldsymbol{x}_k - \boldsymbol{x}^* - \boldsymbol{F}'(\boldsymbol{x}_k)^{-1}\boldsymbol{F}(\boldsymbol{x}_k) = O(\|\boldsymbol{x}_k - \boldsymbol{x}^*\|)$$

Combining this with the Newton iteration $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \boldsymbol{F}'(\boldsymbol{x}_k)^{-1}\boldsymbol{F}(\boldsymbol{x}_k)$ we get

$$\boldsymbol{x}_{k+1} - \boldsymbol{x}^* = O(\|\boldsymbol{x}_k - \boldsymbol{x}^*\|).$$

So

$$\lim_{k\to\infty} \|\boldsymbol{x}_{k+1} - \boldsymbol{x}^*\|/\|\boldsymbol{x}_k - \boldsymbol{x}^*\| = 0$$

This shows the superlinear convergence. □

The previous result is interesting, but it does not say *how near* to the root the starting point need to be in order to get convergence. This is the next topic. Let $\boldsymbol{F} : U \to \mathbb{R}^n$ where U is an open, convex set in $\mathbb{R}^n$. Consider the conditions on the derivative $\boldsymbol{F}'$

$$\begin{array}{lll} (i) & \|\boldsymbol{F}'(\boldsymbol{x}) - \boldsymbol{F}'(\boldsymbol{y})\| \le M\|\boldsymbol{x} - \boldsymbol{y}\| & \text{for all } \boldsymbol{x}, \boldsymbol{y} \in U \\ (ii) & \|\boldsymbol{F}'(\boldsymbol{x}_0)\| \le K & \text{for some } \boldsymbol{x}_0 \in U \end{array} \tag{11.6}$$

where K and L are some constants. Here $\|\boldsymbol{F}'(\boldsymbol{x}_0)\|$ denotes the *operator norm* of the square matrix $\boldsymbol{F}'(x_0)$ which is defined as

$$\|\boldsymbol{F}'(\boldsymbol{x}_0)\| = \sup_{\|\boldsymbol{x}\|=1} \|\boldsymbol{F}'(\boldsymbol{x}_0)\boldsymbol{x}\|$$

and it measures how much the operator $\boldsymbol{F}'(\boldsymbol{x}_0)$ may increase the size of vectors. The following convergence result for Newton's method is known as *Kantorovich' theorem*.

Theorem 11.3 (Kantorovich' theorem)**.** Let $\boldsymbol{F} : U \to \mathbb{R}^n$ be a differentiable function satisfying (11.6). Assume that $\bar{B}(\boldsymbol{x}_0; 1/(KL)) \subseteq U$ and that

$$\|\boldsymbol{F}'(\boldsymbol{x}_0)^{-1}\boldsymbol{F}(\boldsymbol{x}_0)\| \leq 1/(2KL).$$

Then $\boldsymbol{F}'(\boldsymbol{x})$ is invertible for all $\boldsymbol{x} \in B(\boldsymbol{x}_0; 1/(KL))$ and Newton's method with initial point $\boldsymbol{x}_0$ will produce a sequence $\{\boldsymbol{x}_k\}_{k=0}^{\infty}$ contained in $B(\boldsymbol{x}_0; 1/(KL))$ and $\lim_{k\to\infty} \boldsymbol{x}_k = \boldsymbol{x}^*$ for some limit point $\boldsymbol{x}^* \in \bar{B}(\boldsymbol{x}_0; 1/(KL))$ with

$$\boldsymbol{F}(\boldsymbol{x}^*) = \boldsymbol{0}.$$

A proof of this theorem is quite long (but not very difficult to understand) [8].

One disadvantage with Newton's method is that one needs to know the Jacobi matrix $\boldsymbol{F}'$ explicitly. For complicated functions, or functions being the output of a simulation, the derivative may be hard or impossible to find. The *quasi-Newton method*, also called the *secant-method*, is then a good alternative. The idea is to approximate $\boldsymbol{F}'(\boldsymbol{x}_k)$ by some matrix B_k and to compute the new search direction from

$$B_k\boldsymbol{p} = -\boldsymbol{F}(\boldsymbol{x}_k)$$

A practical method for finding these approximations $B_1, B_2, \ldots$ is *Broyden's method*. Provided that the previous iteration gave $\boldsymbol{x}_k$, with Broyden's method we compute $\boldsymbol{x}_{k+1}$ by following the search direction, define $\boldsymbol{s}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$ and $\boldsymbol{y}_k = \boldsymbol{F}(\boldsymbol{x}_{k+1}) - \boldsymbol{F}(\boldsymbol{x}_k)$, and compute B_{k+1} from B_k by the formula

$$B_{k+1} = B_k + (1/\boldsymbol{s}_k^T\boldsymbol{s}_k)(\boldsymbol{y}_k - B_k\boldsymbol{s}_k)\boldsymbol{s}_k^T. \tag{11.7}$$

It can be shown that B_k approximates the Jacobi matrix $\boldsymbol{F}'(x_k)$ well in each iteration. Moreover, the update given in (11.7) can be done efficiently (it is a rank one update of B_k).

Algorithm: Broyden's method:
1. Choose an initial point $\boldsymbol{x}_0$, and an initial B_0.
2. For $k = 0, 1, \ldots$ do
   (i) Find direction $\boldsymbol{p}_k$ by solving $B_k\boldsymbol{p} = -\boldsymbol{F}(\boldsymbol{x}_k)$
   (ii) Use line search (see Section 12.2) along direction $\boldsymbol{p}_k$ to find α_k
   (iii) Update: $\boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \alpha_k\boldsymbol{p}_k$
   $\boldsymbol{s}_k := \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$
   $\boldsymbol{y}_k := \boldsymbol{F}(\boldsymbol{x}_{k+1}) - \boldsymbol{F}(\boldsymbol{x}_k)$
   compute B_{k+1} from (11.7).

Note that this algorithm also computes an α through what we call a *line search*, to attempt to find the optimal distance to follow the search direction.

We do not here specify how this line search can be performed. Also, we do not specify how the initial values can be chosen. For B_0, any approximation of the Jacobian of $\boldsymbol{F}$ at $\boldsymbol{x}_0$ can be used, using a numerical differentiation method of your own choosing. One can show that Broyden's method, under certain assumptions, also converges superlinearly, see [11].

Exercises for Section 11.2

Ex. 1 — Show that the problem of solving nonlinear equations (11.1) may be transformed into a nonlinear optimization problem. (Hint: Square each component function and sum these up!)

Ex. 2 — Let $T : \mathbb{R} \to \mathbb{R}$ be given by $T(x) = (3/2)(x - x^3)$. Draw the graph of this function, and determine its fixed points. Let x^* denote the largest fixed point. Find, using your graph, an interval I containing x^* such that the fixed point algorithm with an initial point in I will guaranteed converge towards x^*. Then try the fixed point algorithm with starting point $x_0 = \sqrt{5/3}$.

Ex. 3 — Let $\alpha \in \mathbb{R}_+$ be fixed, and consider $f(x) = x^2 - \alpha$. Then the zeros are $\pm\sqrt{\alpha}$. Write down the Newton's iteration for this problem. Let $\alpha = 2$ and compute the first three iterates in Newton's method when $x_0 = 1$.

Ex. 4 — For any vector norm $\|\cdot\|$ on $\mathbb{R}^n$, we can more generally define a corresponding *operator norm* for $n \times n$ matrices by

$$\|A\| = \sup_{\|\boldsymbol{x}\|=1} \|A\boldsymbol{x}\|.$$

a. Explain why this supremum is attained.
b. Consider the vector norm $\|\boldsymbol{x}\| = \|\boldsymbol{x}\|_1 = \sum_{j=1}^n |x_j|$ on $\mathbb{R}^n$. For $n = 2$, draw the sublevel set $\{\boldsymbol{x} \in \mathbb{R}^2 : \|\boldsymbol{x}\|_1 \leq 1\}$. Compute the corresponding operator norm $\|A\|$ where A is an $n \times n$ matrix.

Ex. 5 — Consider a linear map $T : \mathbb{R}^n \to \mathbb{R}^m$ given by $T(\boldsymbol{x}) = A\boldsymbol{x}$ where A is an $n \times n$ matrix. When is T a contraction, using the operator norm defined in the previous exercise?

Ex. 6 — Test the function `newtonmult` on the equations given initially in Section 11.1.

Ex. 7 — In this exercise we will implement Broyden's method with Matlab.

a. Given a value $\boldsymbol{x}_0$, implement a function which computes an estimate of $\boldsymbol{F}'(\boldsymbol{x}_0)$ by estimating the partial derivatives of $\boldsymbol{F}$, using a numerical differentiation method and step size of you own choosing.

b. Implement a function

```
function x=broyden(x0,F)
```

which returns an estimate of a zero of $\boldsymbol{F}$ using Broyden's method. Your method should set B_0 to be the matrix obtained from the function in a. Just indicate where line search along the search direction should be performed in your function, without implementing it. The function should work as `newtonmult` in that it terminates after a given number of iterations, or after precision of a given accuracy has been obtained.

Chapter 12

Unconstrained optimization

How can we know whether a given point $\boldsymbol{x}^*$ is a minimum, local or global, of some given function $f : \mathbb{R}^n \to \mathbb{R}$? And how can we find such a point $\boldsymbol{x}^*$?

These are, of course, some main questions in optimization. In order to give good answers to these questions we need *optimality conditions*. They provide tests for optimality, and serve as the basis for algorithms. We here focus on differentiable functions; the corresponding results for the nondifferentiable case are more difficult (but they exist, and are based on convexity, see [5, 13]).

For unconstrained problems it is not difficult to find powerful optimality conditions from Taylor's theorem for functions in several variables.

12.1 Optimality conditions

In order to establish optimality conditions in unconstrained optimization, Taylor's theorem is the starting point, see Section 9.3. We only consider minimization problems, as maximization problems are turned into minimization problems by multiplying the function f by -1.

First we look at some necessary optimality conditions.

Theorem 12.1. Assume that $f : \mathbb{R}^n \to \mathbb{R}$ has continuous partial derivatives, and assume that $\boldsymbol{x}^*$ is a local minimum of f. Then

$$\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}. \tag{12.1}$$

If, moreover, f has continuous second order partial derivatives, then $\nabla^2 f(\boldsymbol{x}^*)$ is positive semidefinite.

Proof. Assume that $\boldsymbol{x}^*$ is a local minimum of f and that $\nabla f(\boldsymbol{x}^*) \neq \boldsymbol{0}$. Let $\boldsymbol{h} = -\alpha \nabla f(\boldsymbol{x}^*)$ where $\alpha > 0$. Then $\nabla f(\boldsymbol{x}^*)^T \boldsymbol{h} = -\alpha \|\nabla f(\boldsymbol{x}^*)\|^2 < 0$ and by continuity of the partial derivatives of f, $\nabla f(\boldsymbol{x})^T \boldsymbol{h} < 0$ for all $\boldsymbol{x}$ in some

neighborhood of $\boldsymbol{x}^*$. From Theorem 9.3 (first order Taylor) we obtain

$$f(\boldsymbol{x}^* + \boldsymbol{h}) - f(\boldsymbol{x}^*) = \nabla f(\boldsymbol{x}^* + t\boldsymbol{h})^T \boldsymbol{h} \tag{12.2}$$

for some $t \in (0, 1)$ (depending on α). By choosing α small enough, the right-hand side of (12.2) is negative (as just said), and so $f(\boldsymbol{x}^* + \boldsymbol{h}) < f(\boldsymbol{x}^*)$, contradicting that $\boldsymbol{x}^*$ is a local minimum. This proves that $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$.

To prove the second statement, we get from Theorem 9.4 (second order Taylor)

$$\begin{aligned} f(\boldsymbol{x}^* + \boldsymbol{h}) &= f(\boldsymbol{x}^*) + \nabla f(\boldsymbol{x}^*)^T \boldsymbol{h} + \frac{1}{2} \boldsymbol{h}^T \nabla^2 f(\boldsymbol{x}^* + t\boldsymbol{h}) \boldsymbol{h} \\ &= f(\boldsymbol{x}^*) + \frac{1}{2} \boldsymbol{h}^T \nabla^2 f(\boldsymbol{x} + t\boldsymbol{h}) \boldsymbol{h} \end{aligned} \tag{12.3}$$

If $\nabla^2 f(\boldsymbol{x}^*)$ is not positive semidefinite, there is an $\boldsymbol{h}$ such that $\boldsymbol{h}^T \nabla^2 f(\boldsymbol{x}^*) \boldsymbol{h} < 0$ and, by continuity of the second order partial derivatives, $\boldsymbol{h}^T \nabla^2 f(\boldsymbol{x}) \boldsymbol{h} < 0$ for all $\boldsymbol{x}$ in some neighborhood of $\boldsymbol{x}^*$. But then (12.3) gives $f(\boldsymbol{x}^* + \boldsymbol{h}) - f(\boldsymbol{x}^*) < 0$; a contradiction. This proves that $\nabla^2 f(\boldsymbol{x})$ is positive semidefinite. □

The two necessary optimality conditions in Theorem 12.1 are called the *first-order* and the *second-order* conditions, respectively. The first-order condition says that the gradient must be zero at $\boldsymbol{x}^*$, and such a point if often called a *stationary* point. The second-order condition may be interpreted by f being "convex locally" at $\boldsymbol{x}^*$, although this is not a precise term. A stationary point which is neither a local minimum or a local maximum is called a *saddle point*. So, every neighborhood of a saddle point contains points with larger and points with smaller f-value.

Theorem 12.1 gives a connection to nonlinear equations. In order to find a stationary point we may solve $\nabla f(\boldsymbol{x}) = \boldsymbol{0}$, which is a $n \times n$ (usually nonlinear) system of equations. (The system is linear whenever f is a quadratic function.) One may solve this equation, for instance, by Newton's method and thereby get a candidate for a local minimum. Sometimes this approach works well, in particular if f has a unique local minimum and we have an initial point "sufficiently close". However, there are other better methods which we discuss later.

It is important to point out that any algorithm for finding a minimum of f *has to* be able to find a stationary point. Therefore algorithms in this area are typically iterative and move to gradually better points where the norm of the gradient becomes smaller, and eventually almost equal to zero.

As an example consider a convex quadratic function

$$f(\boldsymbol{x}) = (1/2)\, \boldsymbol{x}^T A \boldsymbol{x} - \boldsymbol{b}^T \boldsymbol{x}$$

where A is the (symmetric) Hessian matrix is (constant equal to) A and this matrix is positive semidefinite. Then $\nabla f(\boldsymbol{x}) = A\boldsymbol{x} - \boldsymbol{b}$ so the first-order necessary optimality condition is

$$A\boldsymbol{x} = \boldsymbol{b}$$

which is a linear system of equations. If f is *strictly convex*, which happens when A is positive definite, then A is invertible and the unique solution is $\boldsymbol{x}^* = A^{-1}\boldsymbol{b}$. Thus, there is only one candidate for a local (and global) minimum, namely $\boldsymbol{x}^* = A^{-1}\boldsymbol{b}$. Actually, this is indeed a unique global minimum, but to verify this we need a suitable argument. One way is to use convexity (with results presented later) or an alternative is to use *sufficient* optimality conditions which we discuss next. The linear system $A\boldsymbol{x} = \boldsymbol{b}$, when A is positive definite, may be solved by several methods. A popular, and very fast, method is the *conjugate gradient method*. This method, and related methods, are discussed in detail in the course INF-MAT4360 *Numerical linear algebra* [10].

In order to present a sufficient optimality condition we need a result from linear algebra. Recall from linear algebra that a symmetric positive definite matrix has only real eigenvalues and all these are positive.

Proposition 12.2. Let A be an $n \times n$ symmetric positive definite matrix, and let $\lambda_n > 0$ denote its smallest eigenvalue. Then

$$\boldsymbol{h}^T A\boldsymbol{h} \geq \lambda_n \|\boldsymbol{h}\|^2 \quad (\boldsymbol{h} \in \mathbb{R}^n).$$

Proof. By the spectral theorem there is an orthogonal matrix V (containing the orthonormal eigenvectors as its columns) such that

$$A = VDV^T$$

where D is the diagonal matrix with the eigenvalues $\lambda_1, \ldots, \lambda_n$ on the diagonal. Let $\boldsymbol{h} \in \mathbb{R}^n$ and define $\boldsymbol{y} = V^T\boldsymbol{h}$. Then $\|\boldsymbol{y}\| = \|\boldsymbol{h}\|$ and

$$\boldsymbol{h}^T A\boldsymbol{h} = \boldsymbol{h}^T VDV^T\boldsymbol{h} = \boldsymbol{y}^T D\boldsymbol{y} = \sum_{j=1}^{n} \lambda_i y_i^2 \geq \lambda_n \sum_{i=1}^{n} y_i^2 = \lambda_n \|\boldsymbol{y}\|^2 = \lambda_n \|\boldsymbol{h}\|^2.$$

□

Next we consider *sufficient* optimality conditions in the general differentiable case. These conditions are used to prove that a candidate point (say, found by an algorithm) is really a local minimum.

Theorem 12.3. Assume that $f : \mathbb{R}^n \to \mathbb{R}$ has continuous second order partial derivatives in some neighborhood of a point $\boldsymbol{x}^*$. Assume that $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$ and $\nabla^2 f(\boldsymbol{x}^*)$ is positive definite. Then $\boldsymbol{x}^*$ is a local minimum of f.

Proof. From Theorem 9.5 (second order Taylor) and Proposition 12.2 we get

$$\begin{aligned} f(\boldsymbol{x}^* + \boldsymbol{h}) &= f(\boldsymbol{x}^*) + \nabla f(\boldsymbol{x}^*)^T\boldsymbol{h} + \tfrac{1}{2}\boldsymbol{h}^T\nabla^2 f(\boldsymbol{x}^*)\boldsymbol{h} + \epsilon(\boldsymbol{h})\|\boldsymbol{h}\|^2 \\ &\geq f(\boldsymbol{x}^*) + \tfrac{1}{2}\lambda_n\|\boldsymbol{h}\|^2 + \epsilon(\boldsymbol{h})\|\boldsymbol{h}\|^2 \end{aligned}$$

where $\lambda_n > 0$ is the smallest eigenvalue of $\nabla^2 f(\boldsymbol{x}^*)$. Dividing here by $\|\boldsymbol{h}\|^2$ gives

$$(f(\boldsymbol{x}^* + \boldsymbol{h}) - f(\boldsymbol{x}^*))/|\boldsymbol{h}\|^2 = \frac{1}{2}\lambda_n + \epsilon(\boldsymbol{h})$$

Since $\lim_{\boldsymbol{h}\to\boldsymbol{0}} \epsilon(\boldsymbol{h}) = 0$, there is an r such that for $\|\boldsymbol{h}\| < r$, $|\epsilon(\boldsymbol{h})| < \lambda_n/4$. This implies that

$$(f(\boldsymbol{x}^* + \boldsymbol{h}) - f(\boldsymbol{x}^*))/|\boldsymbol{h}\|^2 \geq \lambda_n/4$$

for all $\boldsymbol{h}$ with $\|\boldsymbol{h}\| < r$. This proves that $\boldsymbol{x}^*$ is a local minimum of f. □

We remark that the proof of the previous theorem actually shows that $\boldsymbol{x}^*$ is a *strict* local minimum of f meaning that $f(\boldsymbol{x}^*)$ is strictly smaller than $f(\boldsymbol{x})$ for all other points $\boldsymbol{x}$ in some neighborhood of $\boldsymbol{x}^*$. Note the difference between the necessary and the sufficient optimality conditions: a necessary condition is that $\nabla^2 f(\boldsymbol{x})$ is positive semidefinite, while a part of the sufficient condition is the stronger property that $\nabla^2 f(\boldsymbol{x})$ is positive definite.

Let us see what happens when we work with a convex function.

Theorem 12.4. Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex function. Then a local minimum is also a global minimum. If, in addition, f is differentiable, then a point $\boldsymbol{x}^*$ is a local (and then global) minimum of f if and only if

$$\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}.$$

Proof. Let $\boldsymbol{x}_1$ be a local minimum. If $\boldsymbol{x}_1$ is not a global minimum, there is an $\boldsymbol{x}_2 \neq \boldsymbol{x}_1$ with $f(\boldsymbol{x}_2) < f(\boldsymbol{x}_1)$. Then for $0 < \lambda < 1$

$$f((1-\lambda)\boldsymbol{x}_1 + \lambda\boldsymbol{x}_2) \leq (1-\lambda)f(\boldsymbol{x}_1) + \lambda f(\boldsymbol{x}_2) < f(\boldsymbol{x}_1)$$

and this contradicts that $f(\boldsymbol{x}) \geq f(\boldsymbol{x}_1)$ for all $\boldsymbol{x}$ in a neighborhood of $\boldsymbol{x}^*$. Therefore $\boldsymbol{x}_1$ must be a global minimum.

Assume f is convex and differentiable. Due to Theorem 12.1 we only need to show that if $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$, then $\boldsymbol{x}^*$ is a local and global minimum. So assume that $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$. Then, from Theorem 10.9 we have

$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) + \nabla f(\boldsymbol{x}^*)^T(\boldsymbol{x} - \boldsymbol{x}^*)$$

for all $\boldsymbol{x} \in \mathbb{R}^n$. If $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$, this directly shows that $\boldsymbol{x}^*$ is a global minimum. □

12.2 Methods

Algorithms for unconstrained optimization are iterative methods that generate a sequence of points with gradually smaller values on the function f which is to be minimized. There are two main types of algorithms in unconstrained optimization:

- *Line search methods*: Here one first chooses a *search direction* $\boldsymbol{d}_k$ from the current point $\boldsymbol{x}_k$, using information about the function f. Then one chooses a *step length* α_k so that the new point

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k$$

  has a small, perhaps smallest possible, value on the halfline $\{\boldsymbol{x}_k + \alpha \boldsymbol{d}_k : \alpha \geq 0\}$. α_k describes how far one should go along the search direction. The problem of choosing α_k is a one-dimensional optimization problem. Sometimes we can find α_k exactly, and in such cases we refer to the method as *exact line search*. In cases where α_k can not be found analytically, algorithms can be used to approximate how we can get close to the minimum on the halfline. The method is then refered to as *backtracking line search*.

- *Trust region methods*: In these methods one chooses an approximation $\hat{f}_k$ to the function in some neighborhood of the current point $\boldsymbol{x}_k$. The function $\hat{f}_k$ is simpler than f and one minimizes $\hat{f}_k$ (in the mentioned neighborhood) and let the next iterate $\boldsymbol{x}_{k+1}$ be this minimizer.

These types are typically both based on quadratic approximation of f, but they differ in the order in which one chooses search direction and step size. In the following we only discuss the first type, the line search methods.

A very natural choice for search direction at a point $\boldsymbol{x}_k$ is the negative gradient, $\boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k)$. Recall that the direction of maximum increase of a (differentiable) function f at a point $\boldsymbol{x}$ is $\nabla f(\boldsymbol{x})$, and the direction of maximum decrease is $-\nabla f(\boldsymbol{x})$. To verify this, Taylor's theorem gives

$$f(\boldsymbol{x} + \boldsymbol{h}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x}) \cdot \boldsymbol{h} + \frac{1}{2} \boldsymbol{h}^T \nabla^2 f(\boldsymbol{x} + t\boldsymbol{h}) \boldsymbol{h}.$$

So, for small $\boldsymbol{h}$, the first order term dominates and we would like to make this term small. By the Cauchy-Schwarz inequality[1]

$$\nabla f(\boldsymbol{x}) \cdot \boldsymbol{h} \geq -\|\nabla f(\boldsymbol{x})\| \, \|\boldsymbol{h}\|$$

and equality holds for $\boldsymbol{h} = -\alpha \nabla f(\boldsymbol{x})$ for some $\alpha \geq 0$. In general, we call $\boldsymbol{h}$ a *descent direction* at $\boldsymbol{x}$ if $\nabla f(\boldsymbol{x}) \cdot \boldsymbol{h} < 0$. Thus, if we move in a descent direction from $\boldsymbol{x}$ and make a sufficiently small step, the new point has a smaller f-value. With this background we shall in the following focus on *gradient methods* given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k \tag{12.4}$$

where the direction $\boldsymbol{d}_k$ satisfies

$$\nabla f(\boldsymbol{x}_k) \cdot \boldsymbol{d}_k < 0 \tag{12.5}$$

There are two gradient methods we shall discuss:

[1] The Cauchy-Schwarz' inequality says: $|\boldsymbol{u} \cdot \boldsymbol{v}| \leq \|\boldsymbol{u}\| \, \|\boldsymbol{v}\|$ for $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$.

- If we choose the search direction $\boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k)$, we get the *steepest descent method*

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \nabla f(\boldsymbol{x}_k).$$

  In each step it moves in the direction of the negative gradient. Sometimes this gives slow convergence, so other methods have been developed where other choices of direction $\boldsymbol{d}_k$ are made.

- An important method is *Newton's method*

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \nabla^2 f(\boldsymbol{x}_k)^{-1} \nabla f(\boldsymbol{x}_k). \tag{12.6}$$

  This is the gradient method with $\boldsymbol{d}_k = -\nabla^2 f(\boldsymbol{x}_k)^{-1} \nabla f(\boldsymbol{x}_k)$; this vector $\boldsymbol{d}_k$ is called the *Newton step*. The so-called *pure Newton method* is when one simply chooses step size $\alpha_k = 1$ for each k. To interpret this method consider the second order Taylor approximation of f in $\boldsymbol{x}_k$

$$f(\boldsymbol{x}_k + \boldsymbol{h}) \approx T_f^2(\boldsymbol{x}_k; \boldsymbol{x}_k + \boldsymbol{h}) = f(\boldsymbol{x}_k) + \nabla f(\boldsymbol{x}_k)^T \boldsymbol{h} + (1/2)\boldsymbol{h}^T \nabla^2 f(\boldsymbol{x}_k) \boldsymbol{h}$$

  If we minimize this quadratic function w.r.t. $\boldsymbol{h}$, assuming $\nabla^2 f(\boldsymbol{x}_k)$ is positive definite, we get (see Exercise 7)

$$\boldsymbol{h} = -\nabla^2 f(\boldsymbol{x}_k)^{-1} \nabla f(\boldsymbol{x}_k)$$

  which explains the Newton step.

In the following we follow the presentation in [1]. In a gradient method we need to *choose the step length*. This is the one-dimensional optimization problem

$$\min\{f(\boldsymbol{x} + \alpha \boldsymbol{d}) : \alpha \geq 0\}.$$

Sometimes (maybe not too often) we may solve this problem exactly. Most practical methods try some candidate α's and pick the one with smallest f-value. Note that it is not necessary to compute the exact minimum (this may take too much time). The main thing is to assure that we get a sufficiently large decrease in f without making a too small step.

A popular step size rule is the *Armijo Rule*. Here one chooses (in advance) parameters s, a reduction factor β satisfying $0 < \beta < 1$, and $0 < \sigma < 1$. Define the integer

$$m_k = \min\{m : m \geq 0,\ f(\boldsymbol{x}_k) - f(\boldsymbol{x}_k + \beta^m s \boldsymbol{d}_k) \geq -\sigma \beta^m s \nabla f(\boldsymbol{x}_k)^T \boldsymbol{d}_k\} \tag{12.7}$$

and choose step length $\alpha_k = \beta^{m_k} s$. Here σ is typically chosen very small, e.g. $\sigma = 10^{-3}$. The parameter s fixes the search for step size to lie within the interval $[0, s]$. This can be important: for instance, we can set s so small that the initial step size we try is within the domain of definition for f. According to [1] β is usually chosen in $[1/10, 1/2]$. In the literature one may find a lot more information about step size rules and how they may be adjusted to the methods for finding search direction, see [1], [11].

Now, we return to the *choice of search direction* in the gradient method (12.4). A main question is whether it generates a sequence $\{\boldsymbol{x}_k\}_{k=1}^{\infty}$ which converges to a stationary point $\boldsymbol{x}^*$, i.e., where $\nabla f(\boldsymbol{x}^*) = \mathbf{0}$. It turns out that this may not be the case; one needs to be careful about the choice of $\boldsymbol{d}_k$ to assure this convergence. The problem is that if $\boldsymbol{d}_k$ tends to be nearly orthogonal to $\nabla f(\boldsymbol{x}_k)$ one may get into trouble. For this reason one introduces the following notion:

Definition 12.5 (Gradient related)**.** $\{\boldsymbol{d}_k\}$ is called *gradient related* to $\{\boldsymbol{x}_k\}$ if for any subsequence $\{\boldsymbol{x}_{k_p}\}_{p=1}^{\infty}$ of $\{\boldsymbol{x}_k\}$ converging to a nonstationary point, then the corresponding subsequence $\{\boldsymbol{d}_{k_p}\}_{p=1}^{\infty}$ of $\{\boldsymbol{d}_k\}$ is bounded and $\limsup_{p\to\infty} \nabla f(\boldsymbol{x}_k)^T \boldsymbol{d}_k < 0$.

What this condition assures is that $\|\boldsymbol{d}_k\|$ is not too small or large compared to $\|\nabla f(\boldsymbol{x}_k)\|$ and that the angle between the vectors $\boldsymbol{d}_k$ and $\nabla f(\boldsymbol{x}_k)$ is not too close to 90°. The proof of the following theorem may be found in [1].

Theorem 12.6. Let $\{\boldsymbol{x}_k\}_{k=0}^{\infty}$ be generated by the gradient method (12.4), where $\{\boldsymbol{d}_k\}_{k=0}^{\infty}$ is gradient related to $\{\boldsymbol{x}_k\}_{k=0}^{\infty}$ and the step size α_k is chosen using the Armijo rule. Then every limit point of $\{\boldsymbol{x}_k\}_{k=0}^{\infty}$ is a stationary point.

We remark that in Theorem 12.6 the same conclusion holds if we use exact minimization as step size rule, i.e., $f(\boldsymbol{x}_k+\alpha\boldsymbol{d}_k)$ is minimized exactly with respect to α.

A very important property of a numerical algorithm is its convergence speed. Let us consider the steepest descent method first. It turns out that the convergence speed for this algorithm is very well explained by its performance on minimizing a quadratic function, so therefore the following result is important. In this theorem A is a symmetric positive definite matrix with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n > 0$.

Theorem 12.7. If the steepest descent method $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \nabla f(\boldsymbol{x}_k)$ using exact line search is applied to the quadratic function $f(\boldsymbol{x}) = \boldsymbol{x}^T A \boldsymbol{x}$ where A is positive definite, then (the minimum value is 0 and)

$$f(\boldsymbol{x}_{k+1}) \leq m_A f(\boldsymbol{x}_k)$$

where $m_A = ((\lambda_1 - \lambda_n)/(\lambda_1 + \lambda_n))^2$.

The proof may be found in [1]. Thus, if the largest eigenvalue is much larger than the smallest one, m_A will be nearly 1 and one typically have slow convergence. In this case we have $m_A \approx \text{cond}(A)$ where $\text{cond}(A) = \lambda_1/\lambda_n$ is the *condition number* of the matrix A. So the rule is: if the condition number of A is small we get fast convergence, but if $\text{cond}(A)$ is large, there will be

slow convergence. A similar behavior holds for most functions f because locally near a minimum point the function is very close to its second order Taylor approximation in $\boldsymbol{x}^*$ which is a quadratic function with $A = \nabla^2 f(\boldsymbol{x}^*)$.

Thus, Theorem 12.7 says that the sequence obtained in the steepest descent method converges linearly to a stationary point (at least for quadratic functions).

We now turn to Newton's method.

Newton's method for unconstrained optimization:
1. Choose an initial point $\boldsymbol{x}_0$.
2. For $k = 1, 2, \ldots$ do
   (i) (Newton step) $\boldsymbol{d}_k := -\nabla^2 f(\boldsymbol{x})^{-1} \nabla f(\boldsymbol{x})$; $\eta = -\nabla f(\boldsymbol{x})^T \boldsymbol{d}_k$
   (ii) (Stopping criterion) If $\eta/2 < \epsilon$: stop.
   (iii) (Line search) Use backtracking line search to find step size α_k
   (iv) (Update) $\boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k$

Recall that the pure Newton step minimizes the second order Taylor approximation of f at the current iterate $\boldsymbol{x}_k$. Thus, if the function we minimize is quadratic, we are done in one step. Similarly, if the function can be well approximated by a quadratic function, then one would expect fast convergence.

We shall give a result on the convergence of Newton's method (see [2] for further details). When A is symmetric, we let $\lambda_{min}(A)$ denote that smallest eigenvalue of A.

For the convergence result we need a lemma on strictly convex functions. Assume that $\boldsymbol{x}_0$ is a starting point for Newton's method and let $S = \{\boldsymbol{x} \in \mathbb{R}^n : f(\boldsymbol{x}) \leq f(\boldsymbol{x}_0)\}$. We shall assume that f is continuous and convex, and this implies that S is a closed convex set. We also assume that f has a minimum point $\boldsymbol{x}^*$ which then must be a global minimum. Moreover the minimum point will be unique due to a strict convexity assumption on f. Let $f^* = f(\boldsymbol{x}^*)$ be the optimal value.

The following lemma says that for a convex functions as just described, a point is nearly a minimum point (in terms of the f-value) whenever the gradient is small in that point.

Lemma 12.8. Assume that f is convex as above and that $\lambda_{min}(\nabla^2 f(\boldsymbol{x})) \geq m$ for all $\boldsymbol{x} \in S$. Then

$$f(\boldsymbol{x}) - f^* \leq \frac{1}{2m} \|\nabla f(\boldsymbol{x})\|^2. \tag{12.8}$$

Proof. From Theorem 9.4, the second order Taylor' theorem, we have for each $\boldsymbol{x}, \boldsymbol{y} \in S$

$$f(\boldsymbol{y}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T (\boldsymbol{y} - \boldsymbol{x}) + (1/2)(\boldsymbol{y} - \boldsymbol{x})^T \nabla^2 f(\boldsymbol{z})(\boldsymbol{y} - \boldsymbol{x})$$

for suitable $\boldsymbol{z}$ on the line segment between $\boldsymbol{x}$ and $\boldsymbol{y}$. Here a lower bound for the quadratic term is $(m/2)\|\boldsymbol{y}-\boldsymbol{x}\|^2$, due to Proposition 12.2. Therefore

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{y}-\boldsymbol{x}) + (m/2)\|\boldsymbol{y}-\boldsymbol{x}\|^2.$$

Now, fix $\boldsymbol{x}$ and view the expression on the right-hand side as a quadratic function of $\boldsymbol{y}$. This function is minimized for $\boldsymbol{y}^* = \boldsymbol{x} - (1/m)\nabla f(\boldsymbol{x})$. So, by inserting $\boldsymbol{y} = \boldsymbol{y}^*$ above we get

$$\begin{aligned} f(\boldsymbol{y}) &\geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{y}^*-\boldsymbol{x}) + (m/2)\|\boldsymbol{y}^*-\boldsymbol{x}\|^2 \\ &= f(\boldsymbol{x}) - \tfrac{1}{2m}\|\nabla f(\boldsymbol{x})\|^2 \end{aligned}$$

This holds for every $\boldsymbol{y} \in S$ so letting $\boldsymbol{y} = \boldsymbol{x}^*$ gives

$$f^* = f(\boldsymbol{x}^*) \geq f(\boldsymbol{x}) - \frac{1}{2m}\|\nabla f(\boldsymbol{x})\|^2$$

which proves the desired inequality. □

In the following convergence result we consider a function f as in Lemma 12.8. Moreover, we assume that the Hessian matrix is Lipschitz continuous over S; this is essentially a bound on the third derivatives of f. We do not give the complete proof (it is quite long), but consider some of the main ideas. Recall the definition of the set S from above. Recall that the *spectral norm* of a square matrix A is defined by

$$\|A\|_2 = \max_{\|\boldsymbol{x}\|=1} \|A\boldsymbol{x}\|.$$

It is a fact that $\|A\|_2$ is equal to the largest singular value of A.

Theorem 12.9. Let f be convex and twice continuously differentiable and assume that

(i) $\lambda_{min}(\nabla^2 f(\boldsymbol{x})) \geq m$ for all $\boldsymbol{x} \in S$.

(ii) $\|\nabla^2 f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{y})\|_2 \leq L\|\boldsymbol{x}-\boldsymbol{y}\|$ for all $\boldsymbol{x} \in S$.

Moreover, assume that f has a minimum point $\boldsymbol{x}^*$. Then Newton's method generates a sequence $\{\boldsymbol{x}_k\}_{k=0}^{\infty}$ that converges to $\boldsymbol{x}^*$. From a certain k' the convergence speed is quadratic.

Proof. Define $f^* = f(\boldsymbol{x}^*)$. It is possible to show that there are numbers η and $\gamma > 0$ with $0 < \eta \leq m^2/L$ such that the following holds for each k:

(i) If $\|\nabla f(\boldsymbol{x}_k)\| \geq \eta$, then

$$f(\boldsymbol{x}_{k+1}) \leq f(\boldsymbol{x}_k) - \gamma. \tag{12.9}$$

(ii) If $\|\nabla f(\boldsymbol{x}_k)\| < \eta$, then backtracking line search gives $\alpha_k = 1$ and

$$\frac{L}{2m^2}\|\nabla f(\boldsymbol{x}_{k+1})\| \leq \left(\frac{L}{2m^2}\|\nabla f(\boldsymbol{x}_k)\|\right)^2. \tag{12.10}$$

We omit the proof of this fact; it may be found in [2].

We may now prove that if $\|\nabla f(\boldsymbol{x}_k)\| < \eta$, then also $\|\nabla f(\boldsymbol{x}_{k+1})\| < \eta$. This follows from (ii) above and the fact (assumption) $\eta \leq m^2/L$. Therefore, as soon as case (ii) occurs in the iterative process, in all the remaining iterations case (ii) will occur. Actually, as soon as case (ii) "kicks in" quadratic convergence starts as we shall see now. So assume that case (ii) occurs from a certain k. (Below we show that such k must exist.)

Define $\mu_l = \frac{L}{2m^2}\|\nabla f(\boldsymbol{x}_l)\|$ for each $l \geq k$. Then $0 \leq \mu_k < 1/2$ as $\eta \leq m^2/L$. From what we just saw and (12.10)

$$\mu_{l+1} \leq \mu_l^2 \quad (l \geq k).$$

So (by induction)

$$\mu_l \leq \mu_k^{2^{l-k}} \leq (1/2)^{2^{l-k}} \quad (l = l, k+1, \ldots).$$

Next, from Lemma 12.8

$$f(\boldsymbol{x}_k) - f^* \leq \frac{1}{2m}\|\nabla f(\boldsymbol{x}_l)\|^2 \leq \frac{2m^3}{L^2}(1/2)^{2^{l-k+1}} \quad (l \geq k).$$

This inequality shows that $f(\boldsymbol{x}_l) \to f^*$, and since the minimum point is unique, we must have $\boldsymbol{x}_l \to \boldsymbol{x}^*$. Moreover, it follows that the convergence is quadratic.

It only remains to explain why case (ii) above indeed occurs for some k. In each iteration of type (i) f is decreased by at least γ, as seen from equation (12.10), so the number of such iterations must be bounded by

$$(f(\boldsymbol{x}_0) - f^*)/\gamma$$

which is a finite number. Finally, the proof of the statements in connection with (i) and (ii) above is quite long and one derives several inequalities using the convexity properties of f. □

From the proof it is also possible to say something about haw many iterations that are needed to reach a certain accuracy. In fact, if $\epsilon > 0$ a bound on the number of iterations until $f(\boldsymbol{x}_k) \leq f^* + \epsilon$ is

$$(f(\boldsymbol{x}_0) - f^*)/\gamma + \log_2 \log_2 \frac{2m^3}{\epsilon L^2}.$$

Here γ is the parameter introduced in the proof above. The second term in this expression (the logarithmic term) grows very slowly as ϵ is decreased, and it may roughly be replaced by the constant 6. So, whenever the second stage (case (ii) in the proof) occurs, the convergence is extremely fast, it takes about 6 more Newton iterations. Note that quadratic convergence means, roughly, that the number of correct digits in the answer doubles for every iteration.

Exercises for Section 12.2

Ex. 1 — Consider the function $f(x_1, x_2) = x_1^2 + ax_2^2$ where $a > 0$ is a parameter. Draw some of the level sets of f (for different levels) for each a in the set $\{1, 4, 100\}$. Also draw the gradient in a few points on these level sets.

Ex. 2 — State and prove a theorem similar to Theorem 12.1 for maximization problems.

Ex. 3 — Let $f(\boldsymbol{x}) = \boldsymbol{x}^T A\boldsymbol{x}$ where A is a symmetric $n \times n$ matrix. Assume that A is indefinite, so it has both positive and negative eigenvalues. Show that $\boldsymbol{x} = \boldsymbol{0}$ is a saddlepoint of f.

Ex. 4 — Let $f(x_1, x_2) = 4x_1 + 6x_2 + x_1^2 + 2x_2^2$. Find all stationary points and determine if they are minimum, maximum or saddlepoints. Do the same for the function $g(x_1, x_2) = 4x_1 + 6x_2 + x_1^2 - 2x_2^2$.

Ex. 5 — The function $f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$ is called the *Rosenbrock function*. Compute the gradient and the Hessian matrix at every point $\boldsymbol{x}$. Find every local minimum. Also draw some of the level sets (contour lines) of f using Matlab.

Ex. 6 — Let $f(\boldsymbol{x}) = (1/2)\boldsymbol{x}^T A\boldsymbol{x} - \boldsymbol{b}^T\boldsymbol{x}$ where A is a positive definite $n \times n$ matrix. Consider the steepest descent method applied to the minimization of f, where we assume exact line search is used. Assume that the search direction happens to be equal to an eigenvector of A. Show that then the minimum is reached in just one step.

Ex. 7 — Consider the second order Taylor approximation

$$T_f^2(\boldsymbol{x}; \boldsymbol{x} + \boldsymbol{h}) = f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T\boldsymbol{h} + (1/2)\boldsymbol{h}^T\nabla^2 f(\boldsymbol{x})\boldsymbol{h}.$$

a. Show that $\nabla_{\boldsymbol{h}} T_f^2 = \nabla f(\boldsymbol{x})^T + \nabla^2 f(\boldsymbol{x})\boldsymbol{h}$.

b. Minimizing T_f^2 with respect to $\boldsymbol{h}$ implies solving $\nabla_{\boldsymbol{h}} T^2 f = \boldsymbol{0}$, i.e. $\nabla f(\boldsymbol{x})^T + \nabla^2 f(\boldsymbol{x})\boldsymbol{h} = \boldsymbol{0}$ from a.. If $\nabla^2 f(\boldsymbol{x})$ is positive definite, explain that it is invertible, so that this equation has the unique solution $\boldsymbol{h} = -\nabla^2 f(\boldsymbol{x}_k)^{-1}\nabla f(\boldsymbol{x}_k)$, as previously noted for the Newton step.

Ex. 8 — Implement the steepest descent method. Test the algorithm on the functions in exercises 4 and 5. Use different starting points.

Ex. 9 — Implement a function

```
function alpha=armijorule(f,df,x,d)
```

which returns α chosen according to the Armijo rule for a function $\boldsymbol{f}$ with the given gradient, at point $\boldsymbol{x}$, with search direction $\boldsymbol{d}$. The function shuld compute m_k from Equation (12.7) with $\beta = 0.2$, $s = 0.5$, $\sigma = 10^{-3}$, and return $\alpha = \beta^{m_k} s$.

Ex. 10 — Write a function

```
[xopt,numit]=newtonbacktrack(f,df,d2f,x0)
```

which performs Newton's method for unconstrained optimization. The input parameters are the function, its gradient, its Hesse matrix, and the initial point. The function should also return the number of iterations, and at each iteration write the corresponding function value. The function should use backtracking line search with the function `armijorule` from the previous exercise. Test the algorithm on the functions in exercises 4 and 5. Use different starting points.

Chapter 13

Constrained optimization - theory

In this chapter we consider constrained optimization problems. A general optimization problem is

$$\text{minimize } f(\boldsymbol{x}) \text{ subject to } \boldsymbol{x} \in S$$

where $S \subseteq \mathbb{R}^n$ is a given set and $f : S \to \mathbb{R}$. We here focus on a very general optimization problem which often occurs in applications. Consider the *nonlinear optimization problem* with equality/inequality constraints

$$\begin{aligned} &\text{minimize} && f(\boldsymbol{x}) \\ &\text{subject to} && \\ &&& h_i(\boldsymbol{x}) = 0 \quad (i \leq m) \\ &&& g_j(\boldsymbol{x}) \leq 0 \quad (j \leq r) \end{aligned} \tag{13.1}$$

where f, $h_1, h_2, \ldots, h_m$ and $g_1, g_2, \ldots, g_r$ are continuously differentiable functions from $\mathbb{R}^n$ into $\mathbb{R}$. A point $\boldsymbol{x}$ satisfying all the $m + r$ constraints will be called *feasible*. Thus, we look for a feasible point with smallest f-value.

Our goal is to establish optimality conditions for this problem, starting with the special case with only equality constraints. Then we discuss algorithms for solving this problem. Our presentation is strongly influenced by [2] and [1].

13.1 Equality constraints and the Lagrangian

Consider the nonlinear optimization problem with equality constraints

$$\begin{aligned} &\text{minimize} && f(\boldsymbol{x}) \\ &\text{subject to} && \\ & && h_i(\boldsymbol{x}) = 0 \quad (i \leq m) \end{aligned} \tag{13.2}$$

where f and $h_1, h_2, \ldots, h_m$ are continuously differentiable functions from $\mathbb{R}^n$ into $\mathbb{R}$. We introduce the vector field $\boldsymbol{H} = (h_1, h_2, \ldots, h_m)$, so $\boldsymbol{H} : \mathbb{R}^n \to \mathbb{R}^m$ and $\boldsymbol{H}(\boldsymbol{x}) = (h_1(\boldsymbol{x}), h_2(\boldsymbol{x}), \ldots, h_m(\boldsymbol{x}))$.

We first establish necessary optimality conditions for this problem. A point $\boldsymbol{x} \in \mathbb{R}^n$ is called *regular* if the gradient vectors $\nabla h_i(\boldsymbol{x}^*)$ $(i \leq m)$ are linearly independent.

Theorem 13.1. Let $\boldsymbol{x}^*$ be a local minimum in problem (13.1) and assume that $\boldsymbol{x}^*$ is a regular point. Then there is a unique vector $\boldsymbol{\lambda}^* = (\lambda_1^*, \lambda_2^*, \ldots, \lambda_m^*) \in \mathbb{R}^m$ such that

$$\nabla f(\boldsymbol{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla h_i(\boldsymbol{x}^*) = \boldsymbol{0}. \tag{13.3}$$

If f and each h_i are twice continuously differentiable, then the following also holds

$$\boldsymbol{h}^T(\nabla^2 f(\boldsymbol{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla^2 h_i(\boldsymbol{x}^*))\boldsymbol{h} \geq 0 \quad \text{for all } \boldsymbol{h} \in T(\boldsymbol{x}^*) \tag{13.4}$$

where $T(\boldsymbol{x}^*)$ is the subspace $T(\boldsymbol{x}^*) = \{\boldsymbol{h} \in \mathbb{R}^n : \nabla h_i(\boldsymbol{x}^*) \cdot \boldsymbol{h} = 0 \ (i \leq m)\}$.

The numbers λ_i^* in this theorem are called the *Lagrangian multipliers*. Note that the Lagrangian multiplier vector $\boldsymbol{\lambda}^*$ is unique; this follows directly from the linear independence assumption as $\boldsymbol{x}^*$ is assumed regular. The theorem may also be stated in terms of the *Lagrangian* function $L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ given by

$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i h_i(\boldsymbol{x}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T \boldsymbol{H}(\boldsymbol{x}) \quad (\boldsymbol{x} \in \mathbb{R}^n, \lambda \in \mathbb{R}^m).$$

Then

$$\nabla_{\boldsymbol{x}} L(\boldsymbol{x}, \boldsymbol{\lambda}) = \nabla f(\boldsymbol{x}) + \sum_i \lambda_i \nabla h_i$$

$$\nabla_{\lambda} L(\boldsymbol{x}, \boldsymbol{\lambda}) = \boldsymbol{H}(\boldsymbol{x}).$$

Therefore, the first order conditions in Theorem 13.1 may be rewritten as follows

$$\nabla_{\boldsymbol{x}} L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) = \boldsymbol{0}, \ \nabla_{\boldsymbol{\lambda}} L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) = \boldsymbol{0}.$$

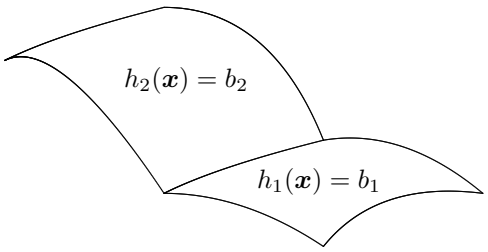


Figure 13.1: The two surfaces $h_1(\boldsymbol{x}) = b_1$ og $h_2(\boldsymbol{x}) = b_2$ intersect each other in a curve. Along this curve the constraints are fulfilled

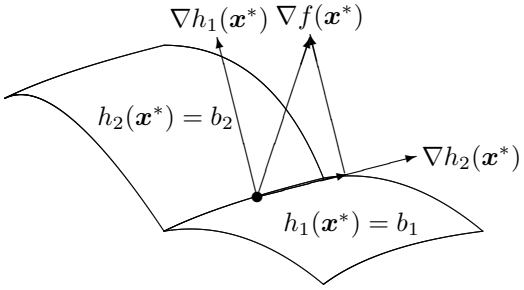


Figure 13.2: $\nabla f(\boldsymbol{x}^*)$ as a linear combination of $\nabla h_1(\boldsymbol{x}^*)$ and $\nabla h_2(\boldsymbol{x}^*)$

Here the second equation simply means that $\boldsymbol{H}(\boldsymbol{x}) = \boldsymbol{0}$. These two equations say that $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)$ is a *stationary point for the Lagrangian*, and it is a system of $n + m$ (possibly nonlinear) equations in $n + m$ variables.

We may interpret the theorem in the following way. At the point $\boldsymbol{x}^*$ the linear subspace $T(\boldsymbol{x}^*)$ consist of the "first order feasible directions". Actually, if each h_i is linear, then $T(\boldsymbol{x}^*)$ consists of those $\boldsymbol{h}$ such that $\boldsymbol{x}^* + \boldsymbol{h}$ is feasible, i.e., $h_i(\boldsymbol{x}^* + \boldsymbol{h}) = 0$ for each $i \leq m$. Thus, (13.3) says that in a local minimum $\boldsymbol{x}^*$ the gradient $\nabla f(\boldsymbol{x}^*)$ is orthogonal to the subspace $T(\boldsymbol{x}^*)$ of the first order feasible variations. This is reasonable since otherwise there would be a feasible direction in which f would decrease. In Figure 13.1 we have plotted a curve where two constraints are fulfilled. In Figure 13.2 we have then shown an interpretation of Theorem 13.1.

Note that this necessary optimality condition corresponds to the condition $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$ in the unconstrained case. The second condition (13.4) is a sim-

ilar generalization of the second order condition in Theorem 12.1 (saying that $\nabla^2 f(\boldsymbol{x}^*)$ is positive semidefinite).

It is possible to prove the theorem by eliminating variables based on the equations and thereby reducing the problem to an unconstrained one. Another proof, which we shall present below is based on the *penalty approach*. This approach is also interesting as it leads to algorithms for actually solving the problem.

Proof. (Theorem 13.1) For $k = 1, 2, \ldots$ consider the modified objective function

$$F^k(\boldsymbol{x}) = f(\boldsymbol{x}) + (k/2)\|\boldsymbol{H}(\boldsymbol{x})\|^2 + (\alpha/2)\|\boldsymbol{x} - \boldsymbol{x}^*\|^2$$

where $\boldsymbol{x}^*$ is the local minimum under consideration, and α is a positive constant. The second term is a penalty term for violating the constraints and the last term is there for proof technical reasons. As $\boldsymbol{x}^*$ is a local minimum there is an $\epsilon > 0$ such that $f(\boldsymbol{x}^*) \leq f(\boldsymbol{x})$ for all $\boldsymbol{x} \in \bar{B}(\boldsymbol{x}^*; \epsilon)$. Choose now an optimal solution $\boldsymbol{x}^k$ of the problem $\min\{F^k(\boldsymbol{x}) : \boldsymbol{x} \in \bar{B}(\boldsymbol{x}^*; \epsilon)\}$; the existence here follows from the extreme value theorem (F^k is continuous and the ball is compact). For every k

$$F^k(\boldsymbol{x}^k) = f(\boldsymbol{x}^k) + (k/2)\|\boldsymbol{H}(\boldsymbol{x}^k)\|^2 + (\alpha/2)\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2 \leq F^k(\boldsymbol{x}^*) = f(\boldsymbol{x}^*).$$

By letting $k \to \infty$ in this inequality we conclude that $\lim_{k\to\infty} \|\boldsymbol{H}(\boldsymbol{x}^k)\| = 0$. So every limit point $\bar{\boldsymbol{x}}$ of the sequence $\{\boldsymbol{x}^k\}$ satisfies $\boldsymbol{H}(\bar{\boldsymbol{x}}) = \boldsymbol{0}$. The inequality above also implies (by dropping a term on the left-hand side) that $f(\boldsymbol{x}^k) + (\alpha/2)\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2 \leq f(\boldsymbol{x}^*)$ for all k, so by passing to the limit we get

$$f(\bar{\boldsymbol{x}}) + (\alpha/2)\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|^2 \leq f(\boldsymbol{x}^*) \leq f(\bar{\boldsymbol{x}})$$

where the last inequality follows from the facts that $\bar{\boldsymbol{x}} \in \bar{B}(\boldsymbol{x}^*; \epsilon)$ and $\boldsymbol{H}(\bar{\boldsymbol{x}}) = \boldsymbol{0}$. Clearly, this gives $\bar{\boldsymbol{x}} = \boldsymbol{x}^*$. We have therefore shown that the sequence $\{\boldsymbol{x}^k\}$ converges to the local minimum $\boldsymbol{x}^*$. Since $\boldsymbol{x}^*$ is the center of the ball $\bar{B}(\boldsymbol{x}^*; \epsilon)$, the points $\boldsymbol{x}^k$ lie in the interior of S for suitably large k. The conclusion is then that $\boldsymbol{x}^k$ is the *unconstrained* minimum of F^k when k is sufficiently large. We may therefore apply Theorem 12.1 so $\nabla F^k(\boldsymbol{x}^k) = \boldsymbol{0}$, so

$$\boldsymbol{0} = \nabla F^k(\boldsymbol{x}^k) = \nabla f(\boldsymbol{x}^k) + k\boldsymbol{H}'(\boldsymbol{x}^k)^T\boldsymbol{H}(\boldsymbol{x}^k) + \alpha(\boldsymbol{x}^k - \boldsymbol{x}^*). \tag{13.5}$$

Here $\boldsymbol{H}'$ denotes the Jacobi matrix of $\boldsymbol{H}$. For suitably large k the matrix $\boldsymbol{H}'(\boldsymbol{x}^k)\boldsymbol{H}'(\boldsymbol{x}^k)^T$ is invertible (as the rows of $\boldsymbol{H}'(\boldsymbol{x}^k)$ are linearly independent due to $\operatorname{rank}(\boldsymbol{H}'(\boldsymbol{x}^*)) = m$ and a continuity argument). Multiply equation (13.5) by $(\boldsymbol{H}'(\boldsymbol{x}^k)\boldsymbol{H}'(\boldsymbol{x}^k)^T)^{-1}\boldsymbol{H}'(\boldsymbol{x}^k)$ to obtain

$$k\boldsymbol{H}(\boldsymbol{x}^k) = -(\boldsymbol{H}'(\boldsymbol{x}^k)\boldsymbol{H}'(\boldsymbol{x}^k)^T)^{-1}\boldsymbol{H}'(\boldsymbol{x}^k)(\nabla f(\boldsymbol{x}^k) + \alpha(\boldsymbol{x}^k - \boldsymbol{x}^*)).$$

Letting $k \to \infty$ we see that the sequence $\{k\boldsymbol{H}(\boldsymbol{x}^k)\}$ is convergent and its limit point $\boldsymbol{\lambda}^*$ is given by

$$\boldsymbol{\lambda}^* = -(\boldsymbol{H}'(\boldsymbol{x}^*)\boldsymbol{H}'(\boldsymbol{x}^*)^T)^{-1}\boldsymbol{H}'(\boldsymbol{x}^*)\nabla f(\boldsymbol{x}^*).$$

Finally, by passing to the limit in (13.5) we get

$$\mathbf{0} = \nabla f(\boldsymbol{x}^*) + \boldsymbol{H}'(\boldsymbol{x}^*)^T \boldsymbol{\lambda}^*$$

This proves the first part of the theorem; we omit proving the second part which may be found in [1]. □

The first order necessary condition (13.3) along with the constraints $\boldsymbol{H}(\boldsymbol{x}) = \mathbf{0}$ is a system of $n + m$ equations in the $n + m$ variables $x_1, x_2, \ldots, x_n$ and $\lambda_1, \lambda_2, \ldots, \lambda_m$. One may use e.g. Newton's method for solving these equations and find a candidate for an optimal solution. But usually there are better numerical methods for solving the optimization (13.1), as we shall see soon.

Necessary optimality conditions are used for finding a candidate solution for being optimal. In order to verify optimality we need *sufficient* optimality conditions.

Theorem 13.2. Assume that f and $\boldsymbol{H}$ are twice continuously differentiable functions. Moreover, let $\boldsymbol{x}^*$ be a point satisfying the first order necessary optimality condition (13.3) and the following condition

$$\boldsymbol{y}^T \nabla^2 L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*) \boldsymbol{y} > 0 \quad \text{for all } \boldsymbol{y} \neq \mathbf{0} \text{ with } \boldsymbol{H}'(\boldsymbol{x}^*)^T \boldsymbol{y} = 0 \tag{13.6}$$

where $\nabla^2 L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)$ is the Hessian of the Lagrangian function with second order partial derivatives with respect to $\boldsymbol{x}$. Then $\boldsymbol{x}^*$ is a (strict) local minimum of f subject to $\boldsymbol{H}(\boldsymbol{x}) = \mathbf{0}$.

This theorem may be proved (see [1] for details) by considering the *augmented Lagrangian function*

$$L_c(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T \boldsymbol{H}(\boldsymbol{x}) + (c/2)\|\boldsymbol{H}(\boldsymbol{x})\|^2 \tag{13.7}$$

where c is a positive scalar. This is in fact the Lagrangian function in the modified problem

$$\text{minimize } \; f(\boldsymbol{x}) + (c/2)\|\boldsymbol{H}(\boldsymbol{x})\|^2 \quad \text{subject to } \quad \boldsymbol{H}(\boldsymbol{x}) = \mathbf{0} \tag{13.8}$$

and this problem must have the same local minima as the problem of minimizing $f(\boldsymbol{x})$ subject to $\boldsymbol{H}(\boldsymbol{x}) = \mathbf{0}$. The objective function in (13.8) contains the *penalty term* $(c/2)\|\boldsymbol{H}(\boldsymbol{x})\|^2$ which may be interpreted as a penalty (increased function value) for violating the constraint $\boldsymbol{H}(\boldsymbol{x}) = \mathbf{0}$. In connection with the proof of Theorem 13.2 based on the augmented Lagrangian one also obtains the following interesting and useful fact: *if $\boldsymbol{x}^*$ and $\boldsymbol{\lambda}^*$ satisfy the sufficient conditions in Theorem 13.2 then there exists a positive $\bar{c}$ such that for all $c \geq \bar{c}$ the point $\boldsymbol{x}^*$ is also a local minimum of the augmented Lagrangian $L_c(\cdot, \boldsymbol{\lambda}^*)$*. Thus, the original constrained problem has been converted to an unconstrained one involving the augmented Lagrangian. And, as we know, unconstrained problems are easier to solve (solve the equations saying that the gradient is equal to zero).

13.2 Inequality constraints and KKT

We now consider the general nonlinear optimization problem where there are both equality and inequality constraints. The problem is then

$$\begin{aligned} &\text{minimize} && f(\boldsymbol{x}) \\ &\text{subject to} && \\ & && h_i(\boldsymbol{x}) = 0 \quad (i \leq m) \\ & && g_j(\boldsymbol{x}) \leq 0 \quad (j \leq r) \end{aligned} \tag{13.9}$$

We assume, as usual, that all these functions are continuously differentiable real-valued functions defined on $\mathbb{R}^n$. In short form we write the constraints as $\boldsymbol{H}(\boldsymbol{x}) = \boldsymbol{0}$ and $\boldsymbol{G}(\boldsymbol{x}) \leq \boldsymbol{0}$ where we let $\boldsymbol{H} = (h_1, h_2, \ldots, h_m)$ and $\boldsymbol{G} = (g_1, g_2, \ldots, g_r)$.

A main difficulty in problems with inequality constraints is to determine which of the inequalities that are active in an optimal solution. If we knew the active inequalities, we would essentially have a problem with only equality constraints, $\boldsymbol{H}(\boldsymbol{x}) = \boldsymbol{0}$ plus the active equalities, i.e., a problem of the form discussed in the previous section. For very small problems (solvable by hand-calculation) a direct method is to consider all possible choices of active inequalities and solve the corresponding equality-constrained problem by looking at the Lagrangian function.

Interestingly, one may also transform the problem (13.9) into the following equality-constrained problem

$$\begin{aligned} &\text{minimize} && f(\boldsymbol{x}) \\ &\text{subject to} && \\ & && h_i(\boldsymbol{x}) = 0 \quad (i \leq m) \\ & && g_j(\boldsymbol{x}) + z_j^2 = 0 \quad (j \leq r). \end{aligned} \tag{13.10}$$

We have introduced extra variables z_j, one for each inequality. The square of these variables represent slack in each of the original inequalities. Note that there is no sign constraint on z_j. Clearly, the problems (13.9) and (13.10) are equivalent. This transformation can also be useful computationally. Moreover, it is useful theoretically as one may apply the optimality conditions from the previous section to problem (13.10) to derive the theorem below (see [1]).

We now present a main result in nonlinear optimization. It gives optimality conditions for this problem, and these conditions are called the *Karush-Kuhn-Tucker conditions*, or simply the *KKT conditions.* In order to present the KKT conditions we introduce the *Lagrangian* function $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^r \to \mathbb{R}$ given by

$$L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i h_i(\boldsymbol{x}) + \sum_{j=1}^{r} \mu_j g_j(\boldsymbol{x}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T \boldsymbol{H}(\boldsymbol{x}) + \boldsymbol{\mu}^T \boldsymbol{G}(\boldsymbol{x}).$$

The gradient of L with respect to $\boldsymbol{x}$ is given by

$$\nabla_x L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \nabla f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i \nabla h_i(\boldsymbol{x}) + \sum_{j=1}^{r} \mu_j \nabla g_j(\boldsymbol{x}).$$

The Hessian matrix of L at $(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ containing second order partial derivatives of L with respect to $\boldsymbol{x}$ will be denoted by $\nabla_{\boldsymbol{xx}} L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$. Finally, the indices of the active inequalities at $\boldsymbol{x}$ is denoted by $A(\boldsymbol{x})$, so $A(\boldsymbol{x}) = \{j \leq r : g_j(\boldsymbol{x}) = 0\}$. A point $\boldsymbol{x}$ is called *regular* if $\{\nabla h_1(\boldsymbol{x}), \dots \nabla h_m(\boldsymbol{x})\} \cup \{\nabla g_i(\boldsymbol{x}) : i \in A(\boldsymbol{x})\}$ is linearly independent.

In the following theorem the first part contains necessary conditions while the second part contains sufficient conditions for optimality.

Theorem 13.3. Consider problem (13.9) with the usual differentiability assumptions.

(*i*) Let $\boldsymbol{x}^*$ be a local minimum of this problem and assume that $\boldsymbol{x}^*$ is a regular point. Then there are unique Lagrange multiplier vectors $\boldsymbol{\lambda}^* = (\lambda_1^*, \lambda_2^*, \dots, \lambda_m^*)$ and $\boldsymbol{\mu}^* = (\mu_1^*, \mu_2^*, \dots, \mu_r^*)$ such that

$$\begin{aligned} &\nabla_x L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \mathbf{0} \\ &\quad \mu_j^* \geq 0 \qquad (j \leq r) \\ &\quad \mu_j^* = 0 \qquad (j \notin A(\boldsymbol{x}^*)). \end{aligned} \tag{13.11}$$

If f, g and h are twice continuously differentiable, then the following also holds

$$\boldsymbol{y}^T \nabla^2_{\boldsymbol{xx}} L(\boldsymbol{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) \boldsymbol{y} \geq 0 \tag{13.12}$$

for all $\boldsymbol{y}$ with $\nabla h_i(\boldsymbol{x}^*)^T \boldsymbol{y} = 0$ $(i \leq m)$ and $\nabla g_j(\boldsymbol{x}^*)^T \boldsymbol{y} = 0$ $(j \in A(\boldsymbol{x}^*))$.

(*ii*) Assume that $\boldsymbol{x}^*$, $\boldsymbol{\lambda}^*$ and $\boldsymbol{\mu}^*$ are such that $\boldsymbol{x}^*$ is a feasible point and (13.11) holds. Assume, moreover, that (13.12) holds with strict inequality for each $\boldsymbol{y}$. Then $\boldsymbol{x}^*$ is a (strict) local minimum in problem (13.9).

Proof. We shall derive this result from Theorem 13.1.

(i) By assumption $\boldsymbol{x}^*$ is a local minimum of problem (13.9), and $\boldsymbol{x}^*$ is a regular point. Consider the *constrained* problem

$$\begin{array}{lll} \text{minimize} & f(\boldsymbol{x}) & \\ \text{subject to} & & \\ & h_i(\boldsymbol{x}) = 0 \quad (i \leq m) & \\ & g_j(\boldsymbol{x}) = 0 \quad (j \in A(\boldsymbol{x}^*)) & \end{array} \tag{13.13}$$

which is obtained by removing all inactive constraints in $\boldsymbol{x}^*$. Then $\boldsymbol{x}^*$ must be a local minimum in (13.13); otherwise there would be a point $\boldsymbol{x}'$ in the neighborhood of $\boldsymbol{x}^*$ which is feasible in (13.13) and satisfying $f(\boldsymbol{x}') < f(\boldsymbol{x}^*)$. By choosing $\boldsymbol{x}'$ sufficiently near $\boldsymbol{x}^*$ we would get $g_j(\boldsymbol{x}') < 0$ for all $j \in A(\boldsymbol{x}^*)$, contradicting that $\boldsymbol{x}^*$ is a local minimum in (13.9). Therefore we may apply Theorem 13.1 to problem (13.13) and by regularity of $\boldsymbol{x}^*$ there must be unique Lagrange multiplier vectors $\boldsymbol{\lambda}^* = (\lambda_1^*, \lambda_2^*, \ldots, \lambda_m^*)$ and μ_j^* $(j \in A(\boldsymbol{x}^*))$ such that

$$\nabla f(\boldsymbol{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla h_i(\boldsymbol{x}^*) + \sum_{j \in A(\boldsymbol{x}^*)} \mu_j^* \nabla g_j(\boldsymbol{x}^*) = \boldsymbol{0}$$

By defining $\mu_j = 0$ for $j \notin A(\boldsymbol{x}^*)$ we get (13.11), except for the nonnegativity of μ.

The remaining part of the theorem may be proved, after some work, by studying the equality-constrained reformulation (13.10) of (13.9) and applying Theorem 13.1 to (13.10). The details may be found in [1]. □

The KKT conditions have an interesting geometrical interpretation. They say that $-\nabla f(\boldsymbol{x}^*)$ may be written as linear combination of the gradients of the h_i's plus a nonnegative linear combination of the gradients of the g_j's that are active at $\boldsymbol{x}^*$.

We remark that the assumption that $\boldsymbol{x}^*$ is a regular point may be too restrictive in some situations, for instance there may be more than n active inequalities in $\boldsymbol{x}^*$. There exist several other weaker assumptions that assure the existence of Lagrangian multipliers (and similar necessary conditions). Let us briefly say a bit more on this matter.

Definition 13.4 (Tangent vector)**.** Let $C \subseteq \mathbb{R}^n$ and let $\boldsymbol{x} \in C$. A vector $\boldsymbol{d} \in \mathbb{R}^n$ is called a *tangent* (vector) to C at $\boldsymbol{x}$ if there is a sequence $\{\boldsymbol{x}^k\}$ in C and a sequence $\{\alpha_k\}$ in $\mathbb{R}_+$ such that

$$\lim_{k \to \infty} (\boldsymbol{x}^k - \boldsymbol{x})/\alpha_k = \boldsymbol{d}.$$

The set of tangent vectors at $\boldsymbol{x}$ is denoted by $T_C(\boldsymbol{x})$.

$T_C(\boldsymbol{x})$ always contains the zero vector and it is a cone, meaning that it contains each positive multiple of its vectors. Consider now problem (13.9) and let C be the set of feasible solutions (those $\boldsymbol{x}$ satisfying all the equality and inequality constraints).

Definition 13.5 (Linearized feasible directions)**.** A *linearized feasible direction* at $\boldsymbol{x} \in C$ is a vector $\boldsymbol{d}$ such that

$$\begin{aligned} \boldsymbol{d} \cdot \nabla h_i(\boldsymbol{x}) &= 0 \quad (i \leq m) \\ \boldsymbol{d} \cdot \nabla g_j(\boldsymbol{x}) &= 0 \quad (j \in A(\boldsymbol{x}^*)). \end{aligned}$$

Let $LF_C(\boldsymbol{x})$ be the set of all linearized feasible directions at $\boldsymbol{x}$.

So, if we move from $\boldsymbol{x}$ along a linearized feasible direction with a suitably small step, then the new point is feasible if we only care about the *linearized* constraints at $\boldsymbol{x}$ (the first order Taylor approximations) of each h_i and each g_j for active constraints at $\boldsymbol{x}$, i.e., those inequality constraints that hold with equality. With this notation we have the following lemma. The proof may be found in [11] and it involves the implicit function theorem from multivariate calculus [8].

Lemma 13.6. Let $\boldsymbol{x}^* \in C$. Then $T_C(\boldsymbol{x}^*) \subseteq LF_C(\boldsymbol{x})$. If $\boldsymbol{x}^*$ is a regular point, then $T_C(\boldsymbol{x}^*) = LF_C(\boldsymbol{x})$.

The purpose of constraint qualifications is to assure that $T_C(\boldsymbol{x}^*) = LF_C(\boldsymbol{x})$. This property is central for obtaining the necessary optimality conditions discussed above. An important example is when C is defined only by *linear constraints*, i.e., each h_i and c_j is a linear function. Then $T_C(\boldsymbol{x}^*) = LF_C(\boldsymbol{x})$ holds for each $\boldsymbol{x} \in C$.

For a more thorough discussion of these matters, see e.g. [11, 1].

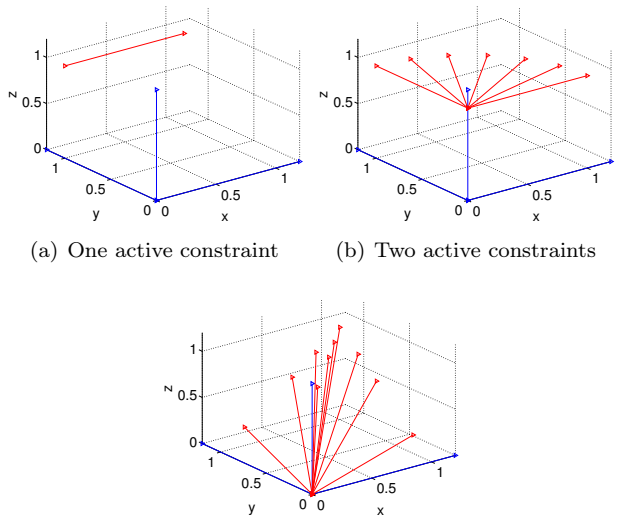
In the remaining part of this section we discuss some examples; the main tool is to establish the KKT conditions.

Example 13.7. Consider the one-variable problem: minimize $f(x)$ subject to $x \geq 0$, where $f : \mathbb{R} \to \mathbb{R}$ is a differentiable convex function. We here let $g_1(x) = -x$ and $m = 0$. The KKT conditions then become: there is a number μ such that $f'(x) - \mu = 0$, $\mu \geq 0$ and $\mu = 0$ if $x > 0$. This is one of the (rare) occasions where we can eliminate the Lagrangian variable μ via the equation $\mu = f'(x)$. So the optimality conditions are: $x \geq 0$ (feasibility), $f'(x) \geq 0$, and $f'(x) = 0$ if $x > 0$ (x is an interior point of the domain so the derivative must be zero), and if $x = 0$ we must have $f'(0) \geq 0$.

Example 13.8. More generally, consider the problem to minimize $f(\boldsymbol{x})$ subject to $\boldsymbol{x} \geq \boldsymbol{0}$, where $f : \mathbb{R}^n \to \mathbb{R}$. So here $C = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{x} \geq \boldsymbol{0}\}$ is the nonnegative orthant. We have that $g_i(x) = -x_i$, so that $\nabla g_i = -\boldsymbol{e}_i$. The KKT conditions say that $-\nabla f(\boldsymbol{x}^*)$ is a nonnegative combination of $-\boldsymbol{e}_i$ for i so that $x_i = 0$. In other words, $\nabla f(\boldsymbol{x}^*)$ is a nonnegative combination of $\boldsymbol{e}_i$ for i so that $x_i = 0$. This means that

$$\begin{aligned} \partial f(\boldsymbol{x}^*)/\partial x_i &= 0 \quad \text{for all } i \leq n \text{ with } \boldsymbol{x}_i^* > 0, \text{ and} \\ \partial f(\boldsymbol{x}^*)/\partial x_i &\geq 0 \quad \text{for all } i \leq n \text{ with } \boldsymbol{x}_i^* = 0. \end{aligned}$$

It we interpret this for $n = 3$ we get the following cases:



(a) One active constraint (b) Two active constraints

(c) Three active constraints

Figure 13.3: The different possibilities for ∇f in a minimum of f, under the constraints $\boldsymbol{x} \geq \mathbf{0}$.

- No active constraints: This means that $x, y, z > 0$. The KKT-conditions say that all partial derivatives are 0, so that $\nabla f(\boldsymbol{x}^*) = 0$. This is reasonable, since these points are internal points.
- One active constraint, such as $x = 0$, $y, z > 0$ The KKT-conditions say that $\partial f(\boldsymbol{x}^*)/\partial y = \partial f(\boldsymbol{x}^*)/\partial z = 0$, so that $\nabla f(\boldsymbol{x}^*)$ points in the positive direction of $\boldsymbol{e}_1$, as shown in Figure 13.3(a).
- Two active constraints, such $x = y = 0$, $z > 0$. The KKT-conditions say that $\partial f(\boldsymbol{x}^*)/\partial z = 0$, so that $\nabla f(\boldsymbol{x}^*)$ lies in the cone spanned by $\boldsymbol{e}_1, \boldsymbol{e}_2$, i.e. $\nabla f(\boldsymbol{x}^*)$ lies in the first quadrant of the xy-plane, as shown in Figure 13.3(b).
- Three active constraints: This means that $x = y = z = 0$. The KKT conditions say that $\nabla f(\boldsymbol{x}^*)$ is in the cone spanned by $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, $\boldsymbol{e}_3$, as shown in Figure 13.3(c).

In all cases $\nabla f(\boldsymbol{x}^*)$ points into a cone spanned by gradients corresponding to the active inequalities (in general, by a cone we mean the set of all linear combinations of a set of vectors, with positive coefficients). Note that for the third case above, we are used to finding minimum values from before: if we restrict f to values where $x = y = 0$, we have a one-dimensional problem where we want to minimize $g(z) = f(x, y, z)$, which is equivalent to finding z so that $g'(z) = \partial f(\boldsymbol{x}^*)/\partial z = 0$, as stated by the KKT-conditions.

Example 13.9. Consider a quadratic optimization problem with linear equality constraints

$$\begin{aligned} &\text{minimize} && (1/2)\, \boldsymbol{x}^T D\boldsymbol{x} - \boldsymbol{q}^T \boldsymbol{x} \\ &\text{subject to} && \\ & && A\boldsymbol{x} = \boldsymbol{b} \end{aligned}$$

where D is positive semidefinite and $A \in \mathbb{R}^{m\times n}, \boldsymbol{b} \in \mathbb{R}^m$. This is a special case of (13.15) where $f(\boldsymbol{x}) = (1/2)\, \boldsymbol{x}^T D\boldsymbol{x} - \boldsymbol{q}^T \boldsymbol{x}$. Then $\nabla f(\boldsymbol{x}) = D\boldsymbol{x} - \boldsymbol{q}$ (see Exercise 9.8). Thus, the KKT conditions are: there is some $\boldsymbol{\lambda} \in \mathbb{R}^m$ such that $D\boldsymbol{x} - \boldsymbol{q} + A^T\boldsymbol{\lambda} = \boldsymbol{0}$. In addition, the vector $\boldsymbol{x}$ is feasible so we have $A\boldsymbol{x} = \boldsymbol{b}$. Thus, solving the quadratic optimization problem amounts to solving the linear system of equations

$$D\boldsymbol{x} + A^T\boldsymbol{\lambda} = \boldsymbol{q},\ A\boldsymbol{x} = \boldsymbol{b}$$

which may be written as

$$\begin{bmatrix} D & A^T \\ A & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{b} \end{bmatrix}. \tag{13.14}$$

Under the additional assumption that D is positive definite and A has full row rank, one can show that the coefficient matrix in (13.14) is invertible so this system has a unique solution $\boldsymbol{x}, \boldsymbol{\lambda}$. Thus, for this problem, we may write down an explicit solution (in terms of the inverse of the block matrix). Numerically, one finds $\boldsymbol{x}$ (and the Lagrangian multiplier $\boldsymbol{\lambda}$) by solving the linear system (13.14) by e.g. Gaussian elimination or some faster (direct or iterative) method.

Example 13.10. Consider an extension of the previous example by allowing linear inequality constraints as well:

$$\begin{aligned} &\text{minimize} && (1/2)\, \boldsymbol{x}^T D\boldsymbol{x} - \boldsymbol{q}^T \boldsymbol{x} \\ &\text{subject to} && \\ & && A\boldsymbol{x} = \boldsymbol{b} \\ & && \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

Here D, A and $\boldsymbol{b}$ are as above. Then $\nabla f(\boldsymbol{x}) = D\boldsymbol{x} - \boldsymbol{q}$ and $\nabla g_k(\boldsymbol{x}) = -\boldsymbol{e}_k$. Thus, the KKT conditions for this problem are: there are $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\boldsymbol{\mu} \in \mathbb{R}^n$ such that $D\boldsymbol{x} - \boldsymbol{q} + A^T\boldsymbol{\lambda} - \boldsymbol{\mu} = \boldsymbol{0}$, $\boldsymbol{\mu} \geq \boldsymbol{0}$ and $\mu_k = 0$ if $x_k > 0$ $(k \leq n)$. We eliminate $\boldsymbol{\mu}$ from the first equation and obtain the equivalent condition: there is a $\boldsymbol{\lambda} \in \mathbb{R}^m$ such that $D\boldsymbol{x} + A^T\boldsymbol{\lambda} \geq \boldsymbol{q}$ and $(D\boldsymbol{x} + A^T\boldsymbol{\lambda} - \boldsymbol{q})_k \cdot x_k = 0$ $(k \leq n)$. In addition, we have $A\boldsymbol{x} = \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$. This problem may be solved numerically, for instance, by a so-called active set method, see [9].

Example 13.11. Linear optimization is a problem of the form

$$\text{minimize } \boldsymbol{c}^T\boldsymbol{x} \quad \text{subject to } A\boldsymbol{x} = \boldsymbol{b}, \quad \boldsymbol{x} \geq \boldsymbol{0}$$

This is a special case of the convex programming problem (13.15) where $g_j(\boldsymbol{x}) = -x_j$ $(j \leq n)$. Here $\nabla f(\boldsymbol{x}) = \boldsymbol{c}$ and $\nabla g_k(\boldsymbol{x}) = -\boldsymbol{e}_k$. Let $\boldsymbol{x}$ be a feasible solution. The KKT conditions state that there are vectors $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\boldsymbol{\mu} \in \mathbb{R}^n$ such that $\boldsymbol{c} + A^T\boldsymbol{\lambda} - \boldsymbol{\mu} = \boldsymbol{0}$, $\boldsymbol{\mu} \geq \boldsymbol{0}$ and $\mu_k = 0$ if $x_k > 0$ $(k \leq n)$. Here we eliminate $\boldsymbol{\mu}$ and obtain the equivalent set of KKT conditions: there is a vector $\boldsymbol{\lambda} \in \mathbb{R}^m$ such that $\boldsymbol{c} + A^T\boldsymbol{\lambda} \geq \boldsymbol{0}$, $(\boldsymbol{c} + A^T\boldsymbol{\lambda})_k \cdot x_k = 0$ $(k \leq n)$. These conditions are the familiar optimality conditions in linear optimization theory. The vector $\boldsymbol{\lambda}$ is feasible in the so-called dual problem and complementary slack holds. We do not go into details on this here, but refer to the course INF-MAT3370 *Linear optimization* where these matters are treated in detail.

13.3 Convex optimization

A *convex optimization* problem is to minimize a convex function f over a convex set C in $\mathbb{R}^n$. These problems are especially attractive, both from a theoretic and algorithmic perspective.

First, let us consider some general results.

Theorem 13.12. Let $f : C \to \mathbb{R}$ be a convex function defined on a convex set $C \subseteq \mathbb{R}^n$.

1. Then every local minimum of f over C is also a global minimum.
2. If f is continuous and C is closed, then the set of local (and therefore global) minimum points of f over C is a closed convex set.
3. Assume, furthermore, that $f : C \to \mathbb{R}$ is differentiable and C is open. Let $\boldsymbol{x}^* \in C$. Then $\boldsymbol{x}^* \in C$ is a local (global) minimum if and only if $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$.

Proof. 1.) The proof of property 1 is exactly as the proof of the first part of Theorem 12.4, except that we work with local and global minimum of f over C.

2.) Assume the set C^* of minimum points is nonempty and let $\alpha = \min_{\boldsymbol{x} \in C} f(\boldsymbol{x})$. Then $C^* = \{\boldsymbol{x} \in C : f(\boldsymbol{x}) \leq \alpha\}$ is a convex set, see Proposition 10.5. Moreover, this set is closed as f is continuous.

3.) This follows directly from Theorem 10.9. □

Next, we consider a quite general convex optimization problem which is of the form (13.9):

$$\begin{array}{lll} \text{minimize} & f(\boldsymbol{x}) & \\ \text{subject to} & & \\ & A\boldsymbol{x} = \boldsymbol{b} & \\ & g_j(\boldsymbol{x}) \leq 0 \quad (j \leq r) & \end{array} \tag{13.15}$$

where all the functions f and g_j are differentiable convex functions, and $A \in \mathbb{R}^{m\times n}$ and $\boldsymbol{b} \in \mathbb{R}^m$. Let C denote the feasible set of problem (13.15). Then C is a convex set, see Proposition 10.5. A special case of (13.15) is linear optimization.

An important concept in convex optimization is *duality*. To briefly explain this introduce again the Lagrangian function $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^r_+ \to \mathbb{R}$ given by

$$L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T(A\boldsymbol{x} - \boldsymbol{b}) + \boldsymbol{\nu}^T\boldsymbol{G}(\boldsymbol{x}) \qquad (\boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{\lambda} \in \mathbb{R}^m, \boldsymbol{\nu} \in \mathbb{R}^r_+)$$

Remark: we use the variable name $\boldsymbol{\nu}$ here in stead of the $\boldsymbol{\mu}$ used before because of another parameter $\boldsymbol{\mu}$ to be used soon. Note that we require $\boldsymbol{\nu} \geq \boldsymbol{0}$.

Define the new function $g : \mathbb{R}^m \times \mathbb{R}^r_+ \to \bar{\mathbb{R}}$ by

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\boldsymbol{x}} L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$$

Note that this infimum may sometimes be equal to $-\infty$ (meaning that the function $\boldsymbol{x} \to L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$ is unbounded below). The function g is the pointwise infimum of a family of affine functions in $(\boldsymbol{\lambda}, \boldsymbol{\mu})$, one function for each $\boldsymbol{x}$, and this implies that g is a concave function. We are interested in g due to the following fact, which is easy to prove. It is usually referred to as *weak duality*.

Lemma 13.13. Let $\boldsymbol{x}$ be feasible in problem (13.15) and let $\boldsymbol{\lambda} \in \mathbb{R}^m, \boldsymbol{\nu} \in \mathbb{R}^r$ where $\boldsymbol{\nu} \geq \boldsymbol{0}$. Then

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \leq f(\boldsymbol{x}).$$

Proof. For $\boldsymbol{\lambda} \in \mathbb{R}^m$, $\boldsymbol{\nu} \in \mathbb{R}^r$ with $\boldsymbol{\nu} \geq \boldsymbol{0}$ and $\boldsymbol{x}$ feasible in problem (13.15) we have

$$\begin{aligned} g(\boldsymbol{\lambda}, \boldsymbol{\nu}) &\leq L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) \\ &= f(\boldsymbol{x}) + \boldsymbol{\lambda}^T(A\boldsymbol{x} - \boldsymbol{b}) + \boldsymbol{\nu}^T\boldsymbol{G}(\boldsymbol{x}) \\ &\leq f(\boldsymbol{x}) \end{aligned}$$

as $A\boldsymbol{x} = \boldsymbol{b}$, $\boldsymbol{\nu} \geq \boldsymbol{0}$ and $\boldsymbol{G}(\boldsymbol{x}) \leq \boldsymbol{0}$. □

Thus, $g(\boldsymbol{\lambda}, \boldsymbol{\nu})$ provides a lower bound on the optimal value in (13.15). It is natural to look for a best possible such lower bound and this is precisely the so-called *dual problem* which is

$$\begin{aligned} &\text{maximize} && g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \\ &\text{subject to} && \\ & && \boldsymbol{\nu} \geq \mathbf{0}. \end{aligned} \tag{13.16}$$

Actually, in this dual problem, we may further restrict the attention to those $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ for which $g(\boldsymbol{\lambda}, \boldsymbol{\nu})$ is finite.

The original problem (13.15) will be called the *primal problem*. It follows from Lemma 13.13 that

$$g^* \leq f^*$$

where f^* denotes the optimal value in the primal problem and g^* the optimal value in the dual problem. If $g^* < f^*$, we say that there is a *duality gap*. Note that the derivation above, and weak duality, holds for *arbitrary* functions f and g_j $(j \leq r)$. The concavity of g also holds generally.

The dual problem is useful when the dual objective function g may be computed efficiently, either analytically or numerically. Duality provides a powerful method for proving that a solution is optimal or, possibly, near-optimal. If we have a feasible $\boldsymbol{x}$ in (13.15) and we have found a dual solution $(\boldsymbol{\lambda}, \boldsymbol{\nu})$ with $\boldsymbol{\nu} \geq \mathbf{0}$ such that

$$f(\boldsymbol{x}) = g(\boldsymbol{\lambda}, \boldsymbol{\nu}) + \epsilon$$

for some ϵ (which then has to be nonnegative), then we can conclude that $\boldsymbol{x}$ is "nearly optimal", it is not possible to improve f by more than ϵ. Such a point $\boldsymbol{x}$ is sometimes called ϵ-*optimal*, where the case $\epsilon = 0$ means optimal.

So, how good is this duality approach? For *convex problems* it is often perfect as the next theorem says. We omit most of the proof, see [5, 1, 14]). For nonconvex problems one should expect a duality gap. Recall that $\boldsymbol{G}'(\boldsymbol{x})$ denotes the Jacobi matrix of $\boldsymbol{G} = (g_1, g_2, \ldots, g_r)$ at $\boldsymbol{x}$.

Theorem 13.14. Consider convex optimization problem (13.15) and assume this problem has a feasible point satisfying

$$g_j(\boldsymbol{x}') < 0 \quad (j \leq r).$$

Then $f^* = g^*$, so there is no duality gap. Moreover, $\boldsymbol{x}$ is a (local and global) minimum in (13.15) if and only if there are $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\boldsymbol{\nu} \in \mathbb{R}^r$ with $\boldsymbol{\nu} \geq \mathbf{0}$ and

$$\nabla f(\boldsymbol{x}) + A^T \boldsymbol{\lambda} + \boldsymbol{G}'(\boldsymbol{x})^T \boldsymbol{\nu} = \mathbf{0}$$

and

$$\nu_j g_j(\boldsymbol{x}) = 0 \quad (j \leq r).$$

Proof. We only prove the second part (see the references above). So assume that $f^* = g^*$ and the infimum and supremum are attained in the primal and dual problems, respectively. Let $\boldsymbol{x}$ be a feasible point in the primal problem.

Then $\boldsymbol{x}$ is a minimum in the primal problem if and only if there are $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\boldsymbol{\nu} \in \mathbb{R}^r$ such that all the inequalities in the proof of Lemma 13.13 hold with equality. This means that $g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$ and $\boldsymbol{\nu}^T \boldsymbol{G}(\boldsymbol{x}) = 0$. But $L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$ is convex in $\boldsymbol{x}$ so it is minimized by $\boldsymbol{x}$ if and only if its gradient is the zero vector, i.e., $\nabla f(\boldsymbol{x}) + \boldsymbol{\lambda}^T A + \boldsymbol{G}'(\boldsymbol{x})^T \boldsymbol{\nu} = \boldsymbol{0}$. This leads to the desired characterization. □

The assumption stated in the theorem, that $g_j(\boldsymbol{x}') < 0$ for each j, is called the *weak Slater condition*.

Finally, we mention a theorem on convex optimization which is used in several applications.

Theorem 13.15. Let $\boldsymbol{x}^* \in C$. Then $\boldsymbol{x}^*$ is a (local and therefore global) minimum of f over C if and only if

$$\nabla f(\boldsymbol{x}^*)^T (\boldsymbol{x} - \boldsymbol{x}^*) \geq 0 \quad \text{for all } \boldsymbol{x} \in C. \tag{13.17}$$

Proof. Assume first that $\nabla f(\boldsymbol{x}^*)^T (\boldsymbol{x} - \boldsymbol{x}^*) < 0$ for some $\boldsymbol{x} \in C$. Consider the function $g(\epsilon) = f(\boldsymbol{x}^* + \epsilon(\boldsymbol{x} - \boldsymbol{x}^*))$ and apply the mean value theorem to this function. Thus, for every $\epsilon > 0$ there exists an $s \in [0, 1]$ with

$$f(\boldsymbol{x}^* + \epsilon(\boldsymbol{x} - \boldsymbol{x}^*)) = f(\boldsymbol{x}^*) + \epsilon \nabla f(\boldsymbol{x}^* + s\epsilon(\boldsymbol{x} - \boldsymbol{x}^*))^T (\boldsymbol{x} - \boldsymbol{x}^*).$$

Since $\nabla f(\boldsymbol{x}^*)^T (\boldsymbol{x} - \boldsymbol{x}^*) < 0$ and the gradient function is continuous (our standard assumption!) we have for sufficiently small $\epsilon > 0$ that $\nabla f(\boldsymbol{x}^* + s\epsilon(\boldsymbol{x} - \boldsymbol{x}^*))^T (\boldsymbol{x} - \boldsymbol{x}^*) < 0$. This implies that $f(\boldsymbol{x}^* + \epsilon(\boldsymbol{x} - \boldsymbol{x}^*)) < f(\boldsymbol{x}^*)$. But, as C is convex, the point $\boldsymbol{x}^* + \epsilon(\boldsymbol{x} - \boldsymbol{x}^*)$ also lies in C and so we conclude that $\boldsymbol{x}^*$ is not a local minimum. This proves that (13.17) is necessary for $\boldsymbol{x}^*$ to be a local minimum of f over C.

Next, assume that (13.17) holds. Using Theorem 10.9 we then get

$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}^*) + \nabla f(\boldsymbol{x}^*)^T (\boldsymbol{x} - \boldsymbol{x}^*) \geq f(\boldsymbol{x}^*) \quad \text{for every } \boldsymbol{x} \in C$$

so $\boldsymbol{x}^*$ is a (global) minimum. □

Exercises for section 13.3

Ex. 1 — In the plane consider a rectangle R with sides of length x and y and with perimeter equal to α (so $2x + 2y = \alpha$). Determine x and y so that the area of R is largest possible.

Ex. 2 — Consider the optimization problem

$$\text{minimize } f(x_1, x_2) \text{ subject to } (x_1, x_2) \in C$$

where $C = \{(x_1, x_2) \in \mathbb{R}^2 : x_1, x_2 \geq 0, 4x_1 + x_2 \geq 8, 2x_1 + 3x_3 \leq 12\}$.
Draw the feasible set C in the plane. Find the set of optimal solutions in each of the cases given below.

a. $f(x_1, x_2) = 1$.
b. $f(x_1, x_2) = x_1$.
c. $f(x_1, x_2) = 3x_1 + x_2$.
d. $f(x_1, x_2) = (x_1 - 1)^2 + (x_2 - 1)^2$.
e. $f(x_1, x_2) = (x_1 - 10)^2 + (x_2 - 8)^2$.

Ex. 3 — Solve

$$\max\{x_1 x_2 \cdots x_n : \sum_{j=1}^{n} x_j = 1\}.$$

Ex. 4 — Let $S = \{\boldsymbol{x} \in \mathbb{R}^2 : \|\boldsymbol{x}\| = 1\}$ be the unit circle in the plane. Let $\boldsymbol{a} \in \mathbb{R}^2$ be a given point. Formulate the problem of finding a nearest point in S to $\boldsymbol{a}$ as a nonlinear optimization problem. How can you solve this problem directly using a geometrical argument?

Ex. 5 — Let S be the unit circle as in the previous exercise. Let $\boldsymbol{a}_1, \boldsymbol{a}_2$ be two given points in the plane. Let $f(\boldsymbol{x}) = \sum_{i=1}^{2} \|\boldsymbol{x} - \boldsymbol{a}_i\|^2$. Formulate this as an optimization problem and find its Lagrangian function L. Find the stationary points of L, and use this to solve the optimization problem.

Ex. 6 — Solve

$$\text{minimize } x_1 + x_2 \text{ subject to } x_1^2 + x_2^2 = 1.$$

using the Lagrangian, see Theorem 13.1. Next, solve the problem by eliminating x_2 (using the constraint).

Ex. 7 — Let $g(x_1, x_2) = 3x_1^2 + 10x_1x_2 + 3x_2^2 - 2$. Solve

$$\min\{\|(x_1, x_2)\| : g(x_1, x_2) = 0\}.$$

Ex. 8 — Same question as in previous exercise, but with $g(x_1, x_2) = 5x_1^2 - 4x_1x_2 + 4x_2^2 - 6$.

Ex. 9 — Let f be a two times differentiable function $f : \mathbb{R}^n \to \mathbb{R}$. Consider the optimization problem

$$\text{minimize } f(\boldsymbol{x}) \text{ subject to } x_1 + x_2 + \cdots + x_n = 1.$$

Characterize the stationary points (find the equation they satisfy).

Ex. 10 — Consider the previous exercise. Explain how to convert this into an unconstrained problem by eliminating x_n. Find an

Ex. 11 — Let A be a real symmetric $n \times n$ matrix. Consider the optimization problem

$$\max\{x^T Ax : \|x\| = 1\}$$

Rewrite the constraint as $-\|x\| = -1$ and show that an optimal solution of this problem must be an eigenvector of A. What can you say about the Lagrangian multiplier?

Ex. 12 — Solve

$$\min\{(1/2)(x_1^2 + x_2^2 + x_3^2 : x_1 + x_2 + x_3 \leq -6\}.$$

Hint: Use KKT and discuss depending on whether the constraint is active or not.

Ex. 13 — Solve

$$\min\{(x_1 - 3)^2 + (x_2 - 5)^2 + x_1 x_2 : 0 \leq x_1, x_2 \leq 1\}.$$

Ex. 14 — Solve

$$\min\{x_1 + x_2 : x_1^2 + x_2^2 \leq 2\}.$$

Ex. 15 — Use Theorem 13.15 to find optimality conditions for the convex optimization problem

$$\min\{f(x_1, x_2, \ldots, x_n) : x_j \geq 0 \ (j \leq n), \ \sum_{j=1}^{n} x_j \leq 1\}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a differentiable convex function.

Chapter 14

Constrained optimization - methods

In this final chapter we present numerical methods for solving nonlinear optimization problems. This is a huge area, so we can here only give a small taste of it! The algorithms we present are known good methods.

14.1 Equality constraints

We here consider the nonlinear optimization problem with linear equality constraints

$$\begin{aligned} &\text{minimize} && f(\boldsymbol{x}) \\ &\text{subject to} && \\ & && A\boldsymbol{x} = \boldsymbol{b} \end{aligned} \tag{14.1}$$

Newton's method may be applied to this problem. The method is very similar to the unconstrained case, but with two modifications. First, the initial point $\boldsymbol{x}_0$ must be chosen so that it is feasible, i.e., $A\boldsymbol{x}_0 = \boldsymbol{b}$. Next, the search direction $\boldsymbol{d}$ must be such that the new iterate is feasible as well. This means that $A\boldsymbol{d} = \boldsymbol{0}$, so the search direction lies in the nullspace of A.

The second order Taylor approximation of f at an iterate $\boldsymbol{x}_k$ is

$$T_f^1(\boldsymbol{x}_k; \boldsymbol{x}_k + \boldsymbol{h}) = f(\boldsymbol{x}_k) + \nabla f(\boldsymbol{x}_k)^T \boldsymbol{h} + (1/2)\boldsymbol{h}^T \nabla^2 f(\boldsymbol{x}_k)\boldsymbol{h}$$

and we want to minimize this w.r.t. $\boldsymbol{h}$ subject to the constraint

$$A(\boldsymbol{x}_k + \boldsymbol{h}) = \boldsymbol{b}$$

This is a quadratic optimization problem in $\boldsymbol{h}$ with a linear equality constraint ($A\boldsymbol{h} = \mathbf{0}$) as in Example 13.9. The KKT conditions for this problem are thus

$$\begin{bmatrix} \nabla^2 f(\boldsymbol{x}_k) & A^T \\ A & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{h} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\nabla f(\boldsymbol{x}_k) \\ \mathbf{0} \end{bmatrix}$$

where $\boldsymbol{\lambda}$ is the Lagrange multiplier. The Newton step is only defined when the coefficient matrix in the KKT problem is invertible. In that case, the problem has a unique solution $(\boldsymbol{h}, \boldsymbol{\lambda})$ and we define $\boldsymbol{d}_{Nt} = \boldsymbol{h}$ and call this the *Newton step*.

Newton's method for solving (14.1) may now be described as follows. Again $\epsilon > 0$ is a small stopping criterion.

Newton's method for linear equality constrained optimization:
1. Choose an initial point $\boldsymbol{x}_0$ satisfying $A\boldsymbol{x}_0 = \boldsymbol{b}$ and let $\boldsymbol{x} = \boldsymbol{x}_0$.
2. repeat
   - (i) Compute the Newton step $\boldsymbol{d}_{Nt}$ and $\eta := \boldsymbol{d}_{Nt}^T \nabla^2 f(\boldsymbol{x}) \boldsymbol{d}_{Nt}$.
   - (ii) If $\eta^2/2 < \epsilon$: stop.
   - (iii) Use backtracking line search to find step size α
   - (iv) Update $\boldsymbol{x} := \boldsymbol{x} + \alpha \boldsymbol{d}_{Nt}$

This leads to an algorithm for Newtons's method for linear equality constrained optimization which is very similar to the function `newtonbacktrack` from Exercise 12.2.10. We do not state a formal convergence theorem for this method, but it behaves very much like Newton's method for unconstrained optimization. Actually, it can be seen that the method just described corresponds to eliminating variables based on the equations $A\boldsymbol{x} = \boldsymbol{b}$ and using the unconstrained Newton method for the resulting (smaller) problem. So as soon as the solution is "sufficiently near" an optimal solution, the convergence rate is quadratic, so extremely few iterations are needed in this final stage.

14.2 Inequality constraints

We here briefly discuss an algorithm for inequality constrained nonlinear optimization problems. The presentation is mainly based on [2, 11]. We restrict the attention to convex optimization problems, but many of the ideas are used for nonconvex problems as well.

The method we present is an *interior-point method*, more precisely, an *interior-point barrier method*. This is an iterative method which produces a sequence of points lying in the relative interior of the feasible set. The barrier idea is to approximate the problem by a simpler one in which constraints are replaced by a penalty term. The purpose of this penalty term is to give large objective

function values to points near the (relative) boundary of the feasible set, which effectively becomes a barrier against leaving the feasible set.

Consider again the convex optimization problem

$$\begin{aligned} &\text{minimize} && f(\boldsymbol{x}) \\ &\text{subject to} && \\ & && A\boldsymbol{x} = \boldsymbol{b} \\ & && g_j(\boldsymbol{x}) \leq 0 \quad (j \leq r) \end{aligned} \tag{14.2}$$

where A is an $m \times n$ matrix and $\boldsymbol{b} \in \mathbb{R}^m$. The feasible set here is $F = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\ g_j(\boldsymbol{x}) \leq 0 \ \ (j \leq r)\}$. We assume that the weak Slater condition holds, and therefore by Theorem 13.14 the KKT conditions for problem (14.2) are

$$\begin{gathered} A\boldsymbol{x} = \boldsymbol{b}, \quad g_j(\boldsymbol{x}) \leq 0 \quad (j \leq r) \\ \boldsymbol{\nu} \geq \boldsymbol{0}, \quad \nabla f(\boldsymbol{x}) + A^T\boldsymbol{\lambda} + \boldsymbol{G}'(\boldsymbol{x})^T\boldsymbol{\nu} = \boldsymbol{0} \\ \nu_j g_j(\boldsymbol{x}) = 0 \quad (j \leq r). \end{gathered} \tag{14.3}$$

So, $\boldsymbol{x}$ is a minimum in (14.2) if and only if there are $\boldsymbol{\lambda} \in \mathbb{R}^m$ and $\boldsymbol{\nu} \in \mathbb{R}^r$ such that (14.3) holds.

Let us state an algorithm for Newton's method for linear equality constrained optimization with inequality constraints. Before we do this there is one final problem we need to address: The α we get from backtracking line search may be so that $\boldsymbol{x} + \alpha\boldsymbol{d}_{Nt}$ do not satisfty the inequality constraints (in the exercises you will be asked to verify that this is the case for a certain function). The problem comes from that the iterates $\boldsymbol{x}_k + \beta^m s\boldsymbol{d}_k$ from Armijo's rule do not necessarily satisfy the inequality constraints. However, we can choose m large enough so that all succeeding iterates satisfy these constraints. We can reimplement the function `armijorule` to address this as follows:

```
function alpha=armijoruleg1g2(f,df,x,d,g1,g2)
  beta=0.2; s=0.5; sigma=10^(-3);
  m=0;
  while (g1(x+beta^m*s*d)>0 || g2(x+beta^m*s*d)>0)
      m=m+1;
  end
  while (f(x)-f(x+beta^m*s*d) < -sigma *beta^m*s *(df(x))'*d)
      m=m+1;
  end
  alpha = beta^m*s;
```

Here `g1` and `g2` are function handles which represent the inequality constraints, and we have added a first loop, which secures that m is so large that the inequality constraints are satisfied. The rest of the code is as in the function `armijorule`. After this we can also modify the function `newtonbacktrack` from Exercise 12.2.10 to a function `newtonbacktrackg1g2` in the obvious way, so that the inequality constraints are passed to `armijoruleg1g2`:

```
function [x,numit]=newtonbacktrackg1g2LEC(f,df,d2f,A,b,x0,g1,g2)
  epsilon=10^(-3);
  x=x0;
  maxit=100;
  for numit=1:maxit
      matr=[d2f(x) A'; A zeros(size(A,1))];
      vect=[-df(x); zeros(size(A,1),1)];
      solvedvals=matr\vect;
      d=solvedvals(1:size(A,2));
      eta=d'*d2f(x)*d;
      if eta^2/2<epsilon
          break;
      end
      alpha=armijoruleg1g2(f,df,x,d,g1,g2);
      x=x+alpha*d;
  end
```

Both these function work in all cases where there are exactly two inequality constraints.

The interior-point barrier method is based on an approximation of problem (14.2) by the *barrier problem*

$$\begin{aligned} &\text{minimize} && f(\boldsymbol{x}) + \mu\phi(\boldsymbol{x}) \\ &\text{subject to} && \\ & && A\boldsymbol{x} = \boldsymbol{b} \end{aligned} \tag{14.4}$$

where

$$\phi(\boldsymbol{x}) = -\sum_{j=1}^{r} \ln(-g_j(\boldsymbol{x}))$$

and $\mu > 0$ is a parameter (in $\mathbb{R}$). The function ϕ is called the (*logarithmic*) *barrier function* and its domain is the relative interior of the feasible set

$$F^\circ = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} = \boldsymbol{b},\ g_j(\boldsymbol{x}) < 0 \ \ (j \le r)\}.$$

The same set F° is the feasible set of the barrier problem. The key properties of the barrier function are:

- ϕ is concave, i.e. $-\phi$ is a convex function. This may be shown from the definition using that g_j is convex and the fact that the logarithm function is concave and increasing.
- If $\{\boldsymbol{x}_k\}$ is a sequence in F° such that $g_j(\boldsymbol{x}_k) \to 0$ for some $j \le r$, then $\phi(\boldsymbol{x}_k) \to \infty$. This is the barrier property.

- ϕ is twice differentiable and

$$\nabla\phi(\boldsymbol{x}) = \sum_{j=1}^{r} \frac{1}{(-g_j(\boldsymbol{x}))} \nabla g_j(\boldsymbol{x}) \tag{14.5}$$

and

$$\nabla^2\phi(\boldsymbol{x}) = \sum_{j=1}^{r} \frac{1}{g_j^2(\boldsymbol{x})} \nabla g_j(\boldsymbol{x}) \nabla g_j(\boldsymbol{x})^T + \sum_{j=1}^{r} \frac{1}{(-g_j(\boldsymbol{x}))} \nabla^2 g_j(\boldsymbol{x}) \tag{14.6}$$

The idea here is that for points $\boldsymbol{x}$ near the boundary of F the value of $\phi(\boldsymbol{x})$ is very large. So, an iterative method which moves around in the interior F° of F will typically avoid points near the boundary as the logarithmic penalty term makes the function value $f(\boldsymbol{x}) + \mu\phi(\boldsymbol{x})$ very large.

The interior point method consists in solving the barrier problem, using Newton's method, for a sequence $\{\mu_k\}$ of (positive) barrier parameters; these are called the *outer iterations*. The solution $\boldsymbol{x}_k$ found for $\mu = \mu_k$ is used as the starting point in Newton's method in the next outer iteration where $\mu = \mu_{k+1}$. The sequence $\{\mu_k\}$ is chosen such that $\mu_k \to 0$. When μ is very small, the barrier function approximates the "ideal" penalty function $\eta(x)$ which is zero in F and $-\infty$ when one of the inequalities $g_j(\boldsymbol{x}) \leq 0$ is violated.

A natural question is why one bothers to solve the barrier problems for more than one single μ, typically a very small value. The reason is that it would be hard to find a good starting point for Newton's method in that case; the Hessian matrix of $\mu\phi$ is typically ill-conditioned for small μ.

Assume now that the barrier problem has a *unique* optimal solution $\boldsymbol{x}(\mu)$; this is true under reasonable assumptions that we shall return to. The point $\boldsymbol{x}(\mu)$ is called a *central point*. Assume also that Newton's method may be applied to solve the barrier problem. The set of points $\boldsymbol{x}(\mu)$ for $\mu > 0$ is called the *central path*; it is a path (or curve) as we know it from multivariate calculus. In order to investigate the central path we prefer to work with the equivalent problem[1] to (14.4) obtained by multiplying the objection function by $1/\mu$, so

$$\begin{aligned} &\text{minimize} && (1/\mu)f(\boldsymbol{x}) + \phi(\boldsymbol{x}) \\ &\text{subject to} && \\ & && A\boldsymbol{x} = \boldsymbol{b}. \end{aligned} \tag{14.7}$$

A central point $\boldsymbol{x}(\mu)$ is characterized by

$$\begin{aligned} &A\boldsymbol{x}(\mu) = \boldsymbol{b} \\ &g_j(\boldsymbol{x}(\mu)) < 0 \quad (j \leq r) \end{aligned}$$

and the existence of $\boldsymbol{\lambda} \in \mathbb{R}^m$ (the Lagrange multiplier vector) such that

$$(1/\mu)\nabla f(\boldsymbol{x}(\mu)) + \nabla\phi(\boldsymbol{x}(\mu)) + A^T\boldsymbol{\lambda} = \boldsymbol{0}$$

[1] Equivalent here means the same minimum points.

i.e.,

$$(1/\mu)\nabla f(\boldsymbol{x}(\mu)) + \sum_{j=1}^{r} \frac{1}{(-g_j(\boldsymbol{x}))} \nabla g_j(\boldsymbol{x}) + A^T \boldsymbol{\lambda} = \mathbf{0}. \tag{14.8}$$

A fundamental question is: how far from being optimal is the central point $\boldsymbol{x}(\mu)$? We now show that duality provides a very elegant way of answering this question.

Theorem 14.1. For each $\mu > 0$ the central point $\boldsymbol{x}(\mu)$ satisfies

$$f^* \leq f(\boldsymbol{x}(\mu)) \leq f^* + r\mu.$$

Proof. Define $\boldsymbol{\nu}(\mu) = (\nu_1(\mu), \ldots, \nu_r(\mu)) \in \mathbb{R}^r$ and $\boldsymbol{\lambda}(\mu) \in \mathbb{R}^m$ by

$$\begin{aligned} \nu_j(\mu) &= -\mu / g_j(\boldsymbol{x}(\mu)), \qquad (j \leq r); \\ \boldsymbol{\lambda}(\mu) &= \mu \boldsymbol{\lambda}. \end{aligned} \tag{14.9}$$

We want to show that the pair $(\boldsymbol{\lambda}(\mu), \boldsymbol{\nu}(\mu))$ is a feasible solution in the dual problem to (14.2), see Section 13.3. So there are two properties to verify, that $\boldsymbol{\nu}(\mu)$ is nonnegative and that $\boldsymbol{x}(\mu)$ minimizes the Lagrangian function for the given $(\boldsymbol{\lambda}(\mu), \boldsymbol{\nu}(\mu))$. The first property is immediate: as $g_j(\boldsymbol{x}(\mu)) < 0$ and $\mu > 0$, we get $\nu_j(\mu) = -\mu / g_j(\boldsymbol{x}(\mu)) > 0$ for each j. Concerning the second property, note first that the Lagrangian function $L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T(A\boldsymbol{x} - \boldsymbol{b}) + \boldsymbol{\nu}^T \boldsymbol{G}(\boldsymbol{x})$ is convex in $\boldsymbol{x}$ for given $\boldsymbol{\lambda}$ and $\mu \geq 0$. Thus, $\boldsymbol{x}$ minimizes this function if and only if $\nabla_{\boldsymbol{x}} L = \mathbf{0}$. Now,

$$\nabla_{\boldsymbol{x}} L(\boldsymbol{x}(\mu), \boldsymbol{\lambda}(\mu), \boldsymbol{\nu}(\mu)) = \nabla f(\boldsymbol{x}(\mu)) + A^T \boldsymbol{\lambda}(\mu) + \sum_j \nu_j(\mu) \nabla g_j(\boldsymbol{x}(\mu)) = \mathbf{0}$$

by (14.8) and the definition of the dual variables (14.9). This shows that $(\boldsymbol{\lambda}(\mu), \boldsymbol{\nu}(\mu))$ is a feasible solution to the dual problem.

By weak duality Lemma 13.13 we therefore obtain

$$\begin{aligned} f^* &\geq g(\boldsymbol{\lambda}(\mu), \boldsymbol{\nu}(\mu)) \\ &= L(\boldsymbol{x}(\mu), \boldsymbol{\lambda}(\mu), \boldsymbol{\nu}(\mu)) \\ &= f(\boldsymbol{x}(\mu)) + \boldsymbol{\lambda}(\mu)^T (A\boldsymbol{x}(\mu) - \boldsymbol{b}) + \sum_{j=1}^{r} \nu_j(\mu) g_j(\boldsymbol{x}(\mu)) \\ &= f(\boldsymbol{x}(\mu)) - r\mu \end{aligned}$$

which proves the result. □

This theorem is very useful and shows why letting $\mu \to 0$ (more accurately $\mu \to 0^+$) is a good idea.

Corollary 14.2. The central path has the following property

$$\lim_{\mu \to 0} f(\boldsymbol{x}(\mu)) = f^*.$$

In particular, if f is continuous and $\lim_{\mu \to 0} \boldsymbol{x}(\mu) = \boldsymbol{x}^*$ for some $\boldsymbol{x}^*$, then $\boldsymbol{x}^*$ is a global minimum in (14.2).

Proof. This follows from Theorem 14.1 by letting $\mu \to 0$. The second part follows from

$$f(\boldsymbol{x}^*) = f(\lim_{\mu \to 0} \boldsymbol{x}(\mu)) = \lim_{\mu \to 0} f(\boldsymbol{x}(\mu)) = f^*$$

by the first part and the continuity of f; moreover $\boldsymbol{x}^*$ must be a feasible point by elementary topology. □

After these considerations we may now present the interior-point barrier method. It uses a tolerance $\epsilon > 0$ in its stopping criterion.

Interior-point barrier method:

1. Choose an initial point $\boldsymbol{x} = \boldsymbol{x}_0$ in F°, $\mu = \mu^0$ and $\alpha < 1$.
2. while $r\mu > \epsilon$ do
   - (i) (Centering step) Using initial point $\boldsymbol{x}$ find the solution $\boldsymbol{x}(\mu)$ of (14.4)
   - (ii) (Update) $\boldsymbol{x} := \boldsymbol{x}(\mu)$
   - (iii) (Decrease μ) $\mu := \alpha\mu$.

This leads to the following algorithm for the internal point barrier method for the case of equality constraints, and 2 inequality constraints:

```
function xopt=IPBopt(f,g1,g2,df,dg1,dg2,d2f,d2g1,d2g2,A,b,x0)
  xopt=x0;
  mu=1;
  alpha=0.1;
  r=2;
  epsilon=10^(-3);
  numitouter=0;
  while (r*mu>epsilon)
    [xopt,numit]=newtonbacktrackg1g2LEC(...
      @(x)(f(x)-mu*log(-g1(x))-mu*log(-g2(x))),...
      @(x)(df(x) - mu*dg1(x)/g1(x) - mu*dg2(x)/g2(x)),...
      @(x)(d2f(x) + mu*dg1(x)*dg1(x)'/(g1(x)^2) ...
           + mu*dg2(x)*dg2(x)'/(g2(x)^2) - mu*d2g1(x)/g1(x)...
           - mu*d2g2(x)/g2(x) ),A,b,xopt,g1,g2);
    mu=alpha*mu;
```

```
    numitouter=numitouter+1;
    fprintf('Iteration %i:',numitouter);
    fprintf('(%f,%f)\n',xopt,f(xopt));
end
```

Note that we here have inserted the expressions from Equation 14.5 and Equation 14.6 for the gradient and the Hesse matrix of the barrier function. The input are f, g_1, g_2, their gradients and their Hesse matrices, the matrix A, the vector $\boldsymbol{b}$, and an initial feasible point $\boldsymbol{x}_0$. The function calls `newtonbacktrackg1g2LEC`, and returns the optimal solution $\boldsymbol{x}^*$. It also gives some information on the values of f during the iterations. The iterations used in Newton's method is called the *inner iterations*. There are different implementation details here that we do not discuss very much. A typical value on α is 0.1. The choice of the initial μ^0 can be difficult, if it is chosen too large, one may experience many outer iterations. Another issue is how accurately one solves (14.4). It may be sufficient to find a near-optimal solution here as this saves inner iterations. For this reason the method is also called a *path-following method*; it follows in the neighborhood of the central path.

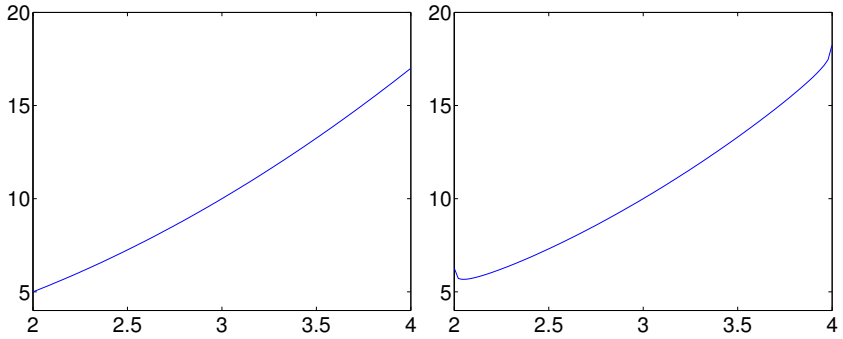
Finally, it should be mentioned that there exists a variant of the interior-point barrier method which permits an infeasible starting point. For more details on this and various implementation issues one may consult [2] or [11].

Example 14.3. Consider the function $f(x) = x^2 + 1$, $2 \leq x \leq 4$. Minimizing f can be considered as the problem of finding a minimum subject to the constraints $g_1(x) = 2 - x \leq 0$, and $g_2(x) = x - 4 \leq 0$. The barrier problem is to minimize the function

$$f(x) + \mu\phi(x) = x^2 + 1 - \mu \ln(x - 2) - \mu \ln(4 - x).$$

Some of these are drawn in Figure 14.1, where we clearly can see the effect of decreasing μ in the barrier function: The function converges to f pointwise, except at the boundaries. It is easy to see that $x = 2$ is the minimum of f under the given constraints, and that $f(2) = 5$ is the minimum value. There are no equality constrains in this case, so that we can use the barrier method with Newton's method for unconstrained optimization, as this was implemented in Exercise 12.2.10. We need, however, to make sure also here that the iterates from Armijo's rule satisfy the inequality constraints. In fact, in the exercises you will be asked to verify that, for the function f considered here, some of the iterates from Armijo's rule do not satisfy the constraints.

It is straightforward to implement a function `newtonbacktrackg1g2` which implements Newtons method for two inequality constraints and no equality constraints (this can follow the implementation of the function `newtonbacktrack` from Exercise 12.2.10, and use the function `armijoruleg1g2`, just as the function `newtonbacktrackg1g2LEC`). This leads to the following algorithm for the internal point barrier method for the case of no equality constraints, but 2 inequality constraints:



(a) $f(x)$ (b) Barrier problem with $\mu = 0.2$

(c) Barrier problem with $\mu = 0.5$ (d) Barrier problem with $\mu = 1$

Figure 14.1: The function from Example 14.3 and some if its barrier functions.

```
function xopt=IPBopt2(f,g1,g2,df,dg1,dg2,d2f,d2g1,d2g2,x0)
  xopt=x0;
  mu=1; alpha=0.1; r=2; epsilon=10^(-3);
  numitouter=0;
  while (r*mu>epsilon)
    [xopt,numit]=newtonbacktrackg1g2(...
      @(x)(f(x)-mu*log(-g1(x))-mu*log(-g2(x))),...
      @(x)(df(x) - mu*dg1(x)/g1(x) - mu*dg2(x)/g2(x)),...
      @(x)(d2f(x) + mu*dg1(x)*dg1(x)'/(g1(x)^2) ...
           + mu*dg2(x)*dg2(x)'/(g2(x)^2) ...
           - mu*d2g1(x)/g1(x) - mu*d2g2(x)/g2(x) ),xopt,g1,g2);
    mu=alpha*mu;
    numitouter=numitouter+1;
    fprintf('Iteration %i:',numitouter);
    fprintf('(%f,%f)\n',xopt,f(xopt));
  end
```

Note that this function also prints a summary for each of the outer iterations, so that we can see the progress in the barrier method. We can now find the minimum of f with the following code, where we have subsituted with Matlab functions for f, g_i, their gradients, and their Hesse matrices.

```
IPBopt2(@(x)(x.^2+1),@(x)(2-x),@(x)(x-4),...
       @(x)(2*x),@(x)(-1),@(x)(1),...
       @(x)(2),@(x)(0),@(x)(0),3)
```

Running this code gives a good approximation to the minumum $x = 2$ after 4 outer iterations.

Exercises for Section 14.2

Ex. 1 — Consider problem (14.1) in Section 14.1. Verify that the KKT conditions for this problem are as stated there.

Ex. 2 — Define the function $f(x, y) = x + y$. We will attempt to minimize f under the constraints $y - x = 1$, and $x, y \geq 0$

a. Find A, $\boldsymbol{b}$, and functions g_1, g_2 so that the problem takes the same form as in Equation (14.2).

b. Draw the contours of the barrier function $f(x, y) + \mu\phi(x, y)$ for $\mu = 0.1, 0.2, 0.5, 1$, where $\phi(x, y) = -\ln(-g_1(x, y)) - \ln(-g_2(x, y))$.

c. Solve the barrier problem analytically using the Lagrange method.

d. It is straightforward to find the minimum of f under the mentioned constraints. State a simple argument for finding this minimum.

e. State the KKT conditions for finding the minimum, and solve these.

f. Show that the central path converges to the same solution which you found in d. and e..

Ex. 3 — Use the function `IPBopt` to verify the solution you found in Exercise 2. Initially you must compute a feasible starting point $\boldsymbol{x}_0$.

Ex. 4 — State the KKT conditions for finding the minimum for the contstrained problem of Example 14.3, and solve these. Verify that you get the same solution as in Example 14.3.

Ex. 5 — In the function `IPBopt2`, replace the call to the function `newtonbacktrackg1g2` with a call to the function `newtonbacktrack`, with the obvious modification to the parameters. Verify that the code does not return the expected minimum in this case.

Ex. 6 — Consider the function $f(x) = (x-3)^2$, with the same constraints $2 \leq x \leq 4$ as in Example 14.3. Verify in this case that the function `IPBopt2` returns the correct minimum regardless of whether you call `newtonbacktrackg1g2` or `newtonbacktrack`. This shows that, at least in some cases where the minimum is an interior point, the iterates from Newtons method satisfy the inequality constraints as well.