Numerical Optimization SS 2023
Lernskript
Version 6. Juli 2023, 13:37:37

Frank Wübbeling

6. Juli 2023

Contents

Chapter -1

Foreword

These are the lecture notes for the lecture Numerical Optimization, held in the summer semester of 2023, by Frank Wübbeling. Note that the notes contain the major definitions, lemmas, etc. of the lecture, but may lack proofs, or rather point at the relevant literature.

I try to be as correct as possible, but errors are, unfortunately, unavoidable. Please send me all the bugs you find.

Chapter 0

Introduction and Examples

0.1 The General Problem Formulation

In the lecture, we will deal with minimization problems of the general form

$$\min f(x),\ x \in \mathbb{R}^n,\ \text{subject to } c_E(x) = 0,\ c_I(x) \leq 0. \qquad (*)$$

The problem involves finding an argument x where that minimum is attained.

Note that here, we restrict ourselves to problems in $\mathbb{R}^n$, we might extend that to problems in infinite–dimensional Hilbert–spaces.

Throughout the lecture, we will use the following notation:

- x is the optimization variable.
- $f : \mathbb{R}^n \mapsto \mathbb{R} \cup \{\infty\}$ is the objective function, we set $f =: f_0$.
- $c_I : \mathbb{R}^n \mapsto (\mathbb{R} \cup \{\infty\})^m$ is the inequality constraint, we set $c_I = (f_1, \dots, f_m)$.
- $c_E : \mathbb{R}^n \mapsto (\mathbb{R} \cup \{\infty\})^p$ is the equality constraint, we set $c_E = (h_1, \dots, h_p)$.

Note that we can replace the equality condition $c_E(x) = 0$ with the inequality conditions $c_I(x) \leq 0$ and $-c_I((x) \leq 0$ and thus arrive at a problem which has no equality constraints.

We allow for the functions to have an infinite value. If $f(x) = \infty$, think of f as undefined in x. Consequently, we define the domains

$$\mathcal{D} = \operatorname{dom} f := \{x : f(x) < \infty\},\ \operatorname{dom}(*) = (\cap \operatorname{dom} f_i) \cap (\cap \operatorname{dom} h_i).$$

Minimization problems are classified based upon the properties of their defining functions. Some important classes we will treat are

- Linear problems/linear programming: iff f, c_E, c_I are affine.
- Nonlinear problems: iff at least one of f, c_E, c_I is nonlinear (not affine).
- Convex problems: iff f and c_I are convex and c_E is affine.

Remember:

- f is affine if $f(x) - f(0)$ is linear.
- f is convex if
$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y)\, \forall x, y \in \mathbb{R}^n, \lambda \in [0,1].$$
- $D \subset \mathbb{R}^n$ is convex iff
$$\lambda x + (1-\lambda)y \in D\, \forall x, y \in \mathbb{R}^n, \lambda \in [0,1].$$

Note that convex problems have very special properties. In this case, $\mathcal{D}$ is convex, and all local minima are also global minima (exercises).

Note that there are many more types of optimization problems which we will not cover in this lecture. Among them are discrete optimization (where $x \in \mathbb{Z}^n$) and optimization on Banach spaces (where $x \in V$ and V is an infinite–dimensional Banach space).

0.2 Some Examples

Note that these examples are a motivation. We will not cover the concrete practical examples in this lecture.

0.2.1 Linear Programming: Maximizing profit with restricted resources

The classical optimization problem which comes up in all MBA textbooks, usually together with the corresponding simplex method for its solution, is the following:

A company can produce n products. For the production, m resources are needed. For the production of one unit of product k one needs a_{lk} units of resource l. Assume that resources are limited, and that there is a maximum of R_l units available for resource l. Assume that P_k units of product k shall be produced, $k = 1 \ldots n$. Then this is within the resource limit iff

$$\sum_k a_{lk} P_k \leq R_l \text{ or } AP \leq R$$

with the obvious definitions of A, P, R.

Assume that product k generates a profit of c_k. Then we wish to select P such that the profit is maximized, or

$$\min -c^t P \text{ where } AP \leq R.$$

This is a linear minimization problem.

0.2.2 Data Fitting/Regression/Deep Learning

We take the example of classification. Assume that the task is to recognize hand–written digits. There is a large library of digital images I_k of digits (in $\mathbb{R}^{n\times n}$) available, and the images have been manually sorted into ten classes, Class 0 has all images with a 0 on them, Class 1 with a 1 and so on.

Further assume that there is a class $\mathcal{P}$ of functions, which are parameterized by a vector $c \in \mathbb{R}^N$. N is typically extremely large. For simplicity, think of $p \in \mathcal{P}$ as polynomials on $\mathbb{R}^{n\times n}$, which are parameterized by their coefficients (and the vector of coefficients is in $\mathbb{R}^N$).

In Regression, we try to find a parameter c such that the corresponding function p_c sorts the images into the correct classes. In our example, we try to find coefficients for a polynomial p such that

$$p(I_k) = l_k \text{ if image } I_k \text{ is in class } l_k.$$

You could think of this as multidimensional, nonlinear interpolation. The idea of course is that if such a p is found, then $p(I) \sim l$ for an image I that contains the digit l but is not part of the training set.

Usually, an exact p cannot be found, so one would instead try to find a solution to the minimization problem

$$\min_c \sum_k |p_c(I_k) - l_k|$$

or: try to find coefficients such that the corresponding polynomial best fits the images to the data.

This is an unrestricted nonlinear problem.

0.2.3 Overdetermined Linear Equations

A specialization of these problems arises particularly in inverse problems. Assume that $A \in \mathbb{R}^{m \times n}$, $m > n$. Then the problem: Find x such that $Ax = b$ has more equations than unknowns and generally has no solution at all. In this situation, we define least squares solutions x^* as the solutions to

$$\min_x ||Ax - b||_2^2.$$

One can show that x^* is a least squares solution of $Ax = b$ iff $A^t Ax = A^t b$.

Since x^* is not necessarily uniquely defined, the minimum norm solution x^+ is defined as the smallest least squares solution, or

$$x^+ \text{ is a least squares solution and } ||x^+|| \leq ||y|| \, \forall \text{ least squares solutions } y.$$

One can show that x^+ is uniquely defined (for all m and n) and satisfies the conditions (exercises)

$$A^t Ax^+ = A^t b, \; x^+ \in \text{range}(A^t).$$

0.2.4 Graphical Solution

When $x \in \mathbb{R}^2$, one can solve the optimization problem graphically. To this end, one first draws the permissible set consisting of all x that satisfy the constraints. Next, one draws the contours, that is the graphs of $f(x) = c$, for various values of c and finds the smallest c such that the contours and the permissible set are not disjoint.

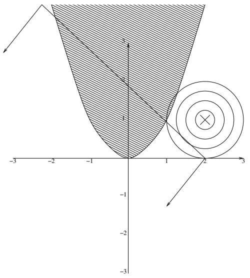

Figure 1: Graphical Solution of Minimization Problems

In the example above, set $A = I$ and $b = (2, 1)$, but apply the conditions $x_1^2 - x_2 \leq 0$ and $x_1 + x_2 \leq 2$. Then the permissible set is the intersection of the parabola $x_2 \geq x_1^2$ and the area under the line $x_2 = 2 - x_1$. See figure 1.

The contours of the objective function

$$f(x_1, x_2) = (x_1 - 2)^2 + (x_2 - 1)^2$$

are the circles of radius c around $(2, 1)$. The smallest c such that one of these circles hits the permissible set gives the optimal point. From the graphic, it is easily seen that the optimal point is the intersection of the ellipse and the line at $(1, 1)$.

0.2.5 Treatment Planning

Obviously, every inverse problem of the form $Rf = g$ can be written as a minimization problem of the form

$$\min_f ||Rf - g||.$$

This is particularly useful if restrictions on f are known (which can be incorporated into the minimization problem, but not into the equations).

These approaches have become more and more popular in recent years. However, there are (at least) two problems in medical imaging which have always been treated this way.

In emission tomography, a radioactive substance is injected into a patient. The image f of the distribution of the substance inside the body then reveals the location of tumors.

To find f, external measurements g are made, which allow to compute f from the equation $Rf = g$ (where R is an operator, the Radon Transform).

However, it is very important to keep in mind that the distribution of the substance is a nonnegative function, and so one solves the corresponding problem

$$\min_f ||Rf - g|| \text{ where } -f \leq 0.$$

We will see that constrained problems of this kind can be solved using the KKT conditions, which in the case of emission tomography immediately lead to the EM (expectation maximization) algorithm.

The other problem is Treatment Planning. Suppose that a tumor cannot be directly removed by surgery. Then, one may resort to radioactive treatment of the tumor. Radioactive rays from various directions are directed onto the tumor, and if the radioactive load is high enough (larger than a threshold c) then the tumor will be destroyed.

Of course, one wishes to select the radioactive rays such that exposure of delicate parts of the body to the rays is minimized. So we arrive at a minimization problem of the kind

$$\min_g \int_{x\notin T} Rg(x)\,dx \text{ where } g \geq 0,\ Rg(x) \geq c \forall x \in T$$

where T is the tumor and $(Rg)(x)$ is the radioactive exposure at point x for treatment plan g (and yes, again it turns out that R is the Radon transform).

Both problems, with solutions, are extensively treated in the classical textbook Parallel Optimization by Censor and Zenios.

Chapter 1

Basics

We start by defining some basic terminology. Assume that a minimization problem $(*)$ is given.

1.1 Terminology

Definition 1.1 *(Feasibility)*
$x \in \mathcal{D}$ is feasible iff all constraints ($g_I(x) \leq 0$, $g_E(x) = 0$) are satisfied.

$$\mathcal{F} = \{x : x \text{ is feasible } \}$$

is the feasible set.
$(*)$ is feasible if there is a feasible x.
Otherwise $(*)$ is infeasible.

Definition 1.2 *(Optimal values)*

$$p^* = \inf\{f_0(x) : x \in \mathcal{D} \text{ is feasible } \}$$

is the optimal value of $(*)$.
Note that this includes the case when $(*)$ is infeasible, then p^* is ∞, the infimum of the empty set.
If $p^* = -\infty$, then $(*)$ is unbounded from below.

Definition 1.3 *(Optimal points)*
x^* is globally optimal point iff x^* is feasible and $f(x^*) = p^*$.
x^* is strictly globally optimal if additionally $f(z) > p^*$ for all feasible z with $z \neq x^*$.

The optimal set X_{opt} is defined as

$$X_{\mathit{opt}} = \{x \in \mathcal{D} : x \text{ is optimal point}\}.$$

A feasible point x is locally optimal iff there is an open set B, $x \in B$, such that

$$f_0(x) \leq f_0(z)\, \forall z \in \mathcal{D} \cap B.$$

It is called strictly locally optimal iff the inequality holds with $<$ for $x \neq z$.

Theorem 1.4 *Let $(*)$ a convex optimization problem. Then x^* is locally optimal iff x^* globally optimal.*

Proof: Exercises.
Question: Is the existence of a globally optimal point guaranteed for convex problems?
Answer: No, consider the unrestricted problem $f_0(x) = x$.

1.2 Transformations

The formulation of an optimization problem is not unique. Using simple transformations, it can be reformulated to a form that is sometimes easier to solve. Here are some examples.

- We already noted: $h_k(x) = 0$ can be reformulated as $h_k(x) \leq 0$ and $-h_k(x) \leq 0$, every equation can be formulated as two inequalities.
- $f_i(x) \leq 0$ can be reformulated as $f_i(x) + s_i = 0$, $-s_i \leq 0$. So complicated inequality conditions can be formulated as simple inequality conditions (with a complicated equality condition). s_i is called slack variable (Schlupfvariable, in German). This is extensively used in linear programming.
- Let $\varphi : \mathbb{R}^n \mapsto \mathbb{R}^n$ invertible, and replace

  $$\widetilde{f_k} = f_k \circ \varphi,\, \widetilde{h_j} = h_j \circ \varphi.$$

  Then x is an optimal point of the old problem iff $\varphi^{-1}(x)$ is an optimal point of the new system.
- Let Ψ monotonically increasing, and let $\widetilde{f_0} = \Psi \circ f_0$. Then the optimal points do not change.
  Simple example: Sometimes it is easier to minimize the sum of two terms rather than their product. In this case, minimize $\log(FG) = \log F + \log G$. This is used in statistics, where the likelihood–function l is introduced, but the objective function always used is the log likelihood function.

- Let $\chi(x) \leq 0$ iff $x \leq 0$. Then $f_j(x) \leq 0$ iff $(\chi \circ f_j)(x) \leq 0$.
- Let $\chi(x) = 0$ iff $x = 0$. Then $h_k(x) = 0$ iff $(\chi \circ h_k)(x) = 0$.
- Assume that $h_k(x_1, x_2) = 0$ can be solved for x_2, thus $x_2 = \widetilde{h_k}(x_1)$. Then x_2 can be eliminated in $(*)$ completely.
- Every restricted problem can be reformulated as an unrestricted problem. Simply use the objective function

  $$\widetilde{f_0}(x) = \begin{cases} f_0(x), & x \text{ feasible} \\ \infty, & \text{else} \end{cases}$$

  Note that generally $\widetilde{f_0}$ will not be differentiable, even when f_0 is, so this seems pretty useless. Nevertheless, this leads to an interesting idea: Maybe we can find a *differentiable* function $\chi(x)$, which is small for x feasible and large otherwise, and consider the unrestricted problem with objective function $f_0(x) + \chi(x)$ as an approximation to the restricted problem. This is called a penalty or augmentation method.
- We can also get rid of (complicated) objective functions and move them to the restrictions. Consider

  $$\min_{t,x} t \text{ where } f_0(x) - t \leq 0,\ x \text{ feasible}\ .$$

Chapter 2

Optimality Conditions

In this section, we derive sufficient and necessary conditions for optimality. The conditions for unconstrained problems should be very familiar from your analysis lectures. In the following, if not explicitly defined, assume that the derivatives used exist.

2.1 Unrestricted Problems

I remind you of the first and second derivative for functions on $\mathbb{R}^n$:

Definition 2.1 *(Gradient, Hessian and Jacobian)*
Let $f : \mathbb{R}^n \mapsto \mathbb{R}$*,* $g : \mathbb{R}^n \mapsto \mathbb{R}^m$*.*

1. *Let* $f \in C^1$*. Then the gradient of* f *in* x *is defined as the vector in* $\mathbb{R}^n$

$$\nabla f(x) := (Df)(x)^t := (\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n})^t(x).$$

2. *Let* $f \in C^2$*. Then the Hessian of* f *in* x *is defined as the* $n \times n$*–matrix*

$$\textit{Hess} f(x) = D^2 f(x) = \nabla^2 f(x) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}(x).$$

3. *Let* $g \in C^1$*,* $g = (g_1, \dots, g_m)^t$*. Then the Jacobian* $Dg(x)$ *of* g *in* x *is defined as the* $m \times n$*–matrix*

$$Dg(x) = (\nabla g_1(x), \dots, \nabla g_m(x))^t.$$

In school, you learned: If x^* is a minimum of the function $f : \mathbb{R} \mapsto \mathbb{R}$, $f \in C^1$, then $f'(x^*) = 0$ (necessary condition), and $f''(x^*) \geq 0$. If $f'(x^*) = 0$ and $f''(x^*) > 0$, then x^* is a minimum (sufficient condition).

For multidimensions, I remind you of the multidimensional Taylor expansion. Please look this up in your analysis textbook, e.g. Forster, Analysis II, Theorem 7.2. We write Corollaries 1 and 2 from this book (or any other source) in a special form:

Theorem 2.2 *(Taylor's Formula for linear and quadratic approximation)*
Let $f : \mathbb{R}^n \mapsto \mathbb{R}$, $x, \xi \in \mathbb{R}^n$ fixed, $\alpha > 0$. Then

1. *Let $f \in C^1$. Then*

$$f(x+\alpha\xi) = f(x)+\sum_{k=1}^{n} \frac{\partial f}{\partial x_k}(x)\, \alpha\xi_k+h(\alpha) = f(x)+\underbrace{\alpha(\nabla f(x), \xi) + h(\alpha)}_{I_1}, \lim_{\alpha \to 0} \frac{h(\alpha)}{\alpha} = 0.$$

2. *Let $f \in C^2$. Then*

$$f(x + \alpha\xi) = f(x) + \underbrace{\alpha(\nabla f(x), \xi) + \frac{1}{2}\alpha^2 \xi^t (D^2 f(x))\xi + \widetilde{h}(\alpha)}_{I_2}, \lim_{\alpha \to 0} \frac{\widetilde{h}(\alpha)}{\alpha^2} = 0.$$

Some thoughts: Obviously, if x is a local minimum of f, then I_1 must be nonnegative for all $x + \alpha\xi$ in a neighborhood of x. However,

$$\frac{I_1(\alpha)}{\alpha} \to_{\alpha \to 0} (\nabla f(x), \xi),$$

so if $(\nabla f(x), \xi) \neq 0$ the sign of I_1 is the sign of $(\nabla f(x), \xi)$ for $\alpha \geq 0$ small enough.

Let us assume that $\nabla f(x) \neq 0$, and choose ξ such that $||\xi|| = ||\nabla f(x)||$. Then

$$(\nabla f(x), \xi) \begin{cases} \text{is maximal and positive } = ||\xi||^2, & \text{if } \xi = \nabla f(x) \\ \text{vanishes}, & \text{if } \xi \text{ and } \nabla f(x) \text{ are orthogonal} \\ \text{is minimal and negative } = -||\xi||^2, & \text{if } \xi = -\nabla f(x). \end{cases}$$

The scalar product is minimal/maximal due to Cauchy–Schwarz.

So $\nabla f(x)$ is the direction of steepest ascent, and $-\nabla f(x)$ is the direction of steepest descent. If we wish to minimize f and have an approximation x, it seems like a good idea to walk in direction $-\nabla f(x)$.

Also, we immediately see that if $\nabla f(x) \neq 0$, then x cannot be a locally minimal point. We sum this up in the following theorems.

Theorem 2.3 *(Necessary (first and second order) conditions for unconstrained problems)*
Let x^* a local minimum of the unconstrained problem 0.1. Then

1. *If f_0 is in C^1, then $\nabla f_0(x^*) = 0$.*
2. *If f_0 is in C^2, then $D^2 f(x)$ is positive semidefinite.*

Theorem 2.4 *(Sufficient (first and second order) conditions for unconstrained problems)*
Let 0.1 an unconstrained problem, $f_0 \in C^2$, and

1. $\nabla f_0(x^*) = 0$.
2. $D^2 f_0(x^*)$ *positive definite.*

Then x^* is a locally optimal point.

Proof:

1. 2.3, part 1: This is already in the remarks.
2. 2.3, part 2: From part 1, we know that the scalar product in I_2 of 2.2 vanishes. So as in the remarks, the sign of I_2 depends on $\xi^t(D^2 f_0(x^*))\xi$. If this is negative for any ξ, then I_2 is negative in a small neighborhood of x^* and x^* is not an optimal point. So $\xi^t(D^2 f_0(x^*))\xi$ must be nonnegative for all ξ, and $D^2 f_0(x^*)$ is positive semidefinite.
3. 2.4: If the Hessian is positive definite, then I_2 is positive in a small neighborhood of x^*, so x^* is an optimal point.

□

Note that the sufficient conditions are not necessary. Consider the 1D example $f(x) = x^4$.

2.2 Problems with Equality Constraints

For constrained problems, things are slightly different. Assume that 0.1 is a constrained problem and x^* is feasible. Then 2.2 holds for directions ξ where $\alpha\xi$ is feasible for small α. We immediately see: If x^* is in the interior of the feasible set $\mathcal{F}$, nothing changes, and necessary and sufficient conditions are as in 2.3 and 2.4. If x^* is on the boundary of the feasible set, then we can only choose ξ if it points into the feasible set (inequality conditions) or along the boundary (equality conditions), where along the boundary means in the direction of the tangent for differentiable curves.

Thus, in the derivation of the conditions, it might be ok for the scalar product of the gradient and a direction ξ not to vanish at an optimal point, as long as ξ points away from the feasible set.

For a curve in 2D, the tangent of $\mathcal{F}$ in x^* is $x'(t)$ for any differentiable parametrization $x(t)$ of the boundary with $x(t) = x^*$. Following this idea, we define in $\mathbb{R}^n$

Definition 2.5 *(Curves and Tangential Plane)*
Let $\mathcal{H}$ a hypersurface in $\mathbb{R}^n$. A curve x on $\mathcal{H}$ is a continuous mapping $[a, b] \mapsto \mathcal{H}$. $x^* \in \mathcal{H}$ lies on the curve if $\exists\, t^* : x(t^*) = x^*$.
We denote (for $x \in C^1$, $x \in C^2$, resp.)

$$\dot{x}(t) = \frac{dx}{dt}(t),\ \ddot{x}(t) = \frac{d^2x}{dt^2}(t).$$

$\theta \in \mathbb{R}^n$ is a tangential vector to $\mathcal{H}$ in x^* iff there is a curve $x(t) \in C^1$ on $\mathcal{H}$ such that

$$x(t^*) = x^*,\ \theta = \dot{x}(t^*).$$

The set of all tangential vectors in x^* is the tangential plane $T_{x^*}\mathcal{H}$ of $\mathcal{H}$ in x^*.

Note that we define the tangential plane for a hypersurface, but in fact it makes sense for any set $\mathcal{H} \subset \mathbb{R}^n$. Some examples:

Example 2.6 *Let $\mathcal{H} \subset \mathbb{R}^n$.*

- *Let x^* in the interior $\overset{\circ}{\mathcal{H}}$ of $\mathcal{H}$. Then $T_{x^*}\mathcal{H} = \mathbb{R}^n$.*
  Proof: Let $y \in \mathbb{R}^n$. Let

  $$c : [-\epsilon, \epsilon] \mapsto \mathbb{R}^n,\ c(t) := x^* + t\,y.$$

  Since x^* is in the interior of $\mathcal{H}$, there is a small neighborhood of x^* that lies in $\mathcal{H}$. So if $\epsilon > 0$ is small enough, the range of c is in $\mathcal{H}$, and c is a curve on $\mathcal{H}$. Since

  $$c(0) = x^*\,, \dot{c}(0) = y$$

  we have $y \in T_{x^*}\mathcal{H}$.

- *Let x^* an isolated point in $\mathcal{H}$. Then $T_{x^*}\mathcal{H} = \{0\}$.*
  Proof: Let c a differentiable curve on $\mathcal{H}$ that contains x^*. Since x^* is isolated, we have $c(t) = x^*$ (otherwise c would not even be continuous) and thus $\nabla c(t) = 0$.

- *Let $f_0 : \mathbb{R}^{n-1} \mapsto \mathbb{R} \in C^1$ and*

  $$\mathcal{H} = \operatorname{graph} f_0 = \{(x, f_0(x)) \in \mathbb{R}^n : x \in \mathbb{R}^{n-1}\}.$$

  Let $x^* \in \mathbb{R}^{n-1}$ a local minimum of f_0. Then

  $$T_{(x^*, f_0(x^*))}\mathcal{H} = \{(x, 0) \in \mathbb{R}^n : x \in \mathbb{R}^{n-1}\}.$$

  Proof: Since x^* is a local minimum of f_0, we have $\nabla f_0(x^*) = 0$. A curve c on $\mathcal{H}$ has the form $c(t) = (x(t), f_0(x(t))$, thus

  $$\dot{c}(t) = (\dot{x}(t), \dot{x}(t) \cdot \nabla f_0(x(t)).$$

  Thus for $x(t^*) = x^*$

  $$\dot{c}(t^*) = (\dot{x}(t^*), 0).$$

  For the opposite direction set $x(t) = x^* + t\,y$ for any $y \in \mathbb{R}^{n-1}$ as in example 1. Note that this gives a necessary condition for the constrained *minimization problem*

  $$\min_{x \in \mathbb{R}^n} x_n \text{ where } f_0(x_1, \ldots, x_{n-1}) - x_n = 0.$$

We now return to our minimization problem 0.1, with equality, but no inequality constraints ($p > 0$, $m = 0$). If not defined otherwise, we assume $D = \mathbb{R}^n$.

We define $\mathcal{S}$ as the hypersurface that satisfies the equality conditions, thus

$$\mathcal{S} = \{x \in \mathbb{R}^n : \; c_E(x) = 0\}.$$

We will characterize the tangential field $T_{x^*}\mathcal{S}$ for $x^* \in \mathcal{S}$ using the Jacobian $Dc_E(x^*) \in \mathbb{R}^{p \times n}$.

Definition 2.7 *(regular points)*
$x^* \in \mathcal{S}$ is a regular point iff $\nabla h_1(x^*), \ldots, \nabla h_p(x^*)$ are linear independent.

Note that this is equivalent to $\operatorname{rank}(Dc_E(x^*)) = p$, and recall from linear algebra that this implies

$$\operatorname{rank}(Dc_E(x^*))(Dc_E(x^*)^t) = p \Rightarrow (Dc_E(x^*))(Dc_E(x^*))^t \in \mathbb{R}^{p \times p} \text{ invertible.}$$

Next, remember the implicit function theorem (Forster Analysis 2, 8.2). We use it in the following form:

Theorem 2.8 *(Implicit Function Theorem)*
Let

$$F : \mathbb{R} \times \mathbb{R}^p \mapsto \mathbb{R}^p, \; F \in C^1, \; F(0,0) = 0, \; D_u F(0,0) \text{ invertible}$$

where $D_u F$ is the Jacobian of F wrt the second variable. Then

$$\exists \epsilon > 0, u : [-\epsilon, \epsilon] \mapsto \mathbb{R}^p : \; u \in C^1, \; u(0) = 0, \; F(t, u(t)) = 0 \, \forall \, t \in [-\epsilon, \epsilon].$$

Now we can prove

Theorem 2.9 *(Characterization of Tangential Plane)*
Let $x^* \in \mathcal{S}$.

1. $T_{x^*}\mathcal{S} \subset \ker Dc_E(x^*)$.
2. *If x^* is regular, then $T_{x^*}\mathcal{S} = \ker Dc_E(x^*)$.*

Proof: Let $x^* \in \mathcal{S}$.

1. Let $y \in T_{x^*}\mathcal{S}$. By definition of the tangential plane, there exists a curve $x \in C^1$ on $\mathcal{S}$ with $x(t^*) = x^*$, $\dot{x}(t^*) = y$.
   By definition of $\mathcal{S}$ we have $c_E(x(t)) = 0$ which implies
   $$0 = \frac{d}{dt}[c_E(x(t))]\bigg|_{t=t^*} = Dc_E(x(t^*))\, \underbrace{\dot{x}(t^*)}_{=y} \Rightarrow y \in \ker Dc_E(x^*).$$

2. Let $y \in \ker Dc_E(x^*)$ and
   $$F : \mathbb{R} \times \mathbb{R}^p \mapsto \mathbb{R}^p,\ F(t,u) := c_E(x^* + ty + Dc_E(x^*)^t u).$$
   Then
   $$F(0,0) = 0,\ D_uF(0,0) = Dc_E(x^*)Dc_E(x^*)^t \text{ invertible.}$$
   According to 2.8, there is a differentiable curve $u(t)$ such that
   $$u(0) = 0,\ F(t,u(t)) = 0 \Rightarrow c_E(x(t)) = 0,\ x(t) = x^* + ty + Dc_E(x^*)^t u(t)$$
   which implies that $x(t)$ is a curve on $\mathcal{S}$. As in (1), we differentiate the equality constraints wrt t
   $$0 = \frac{d}{dt}[c_E(x(t))]\bigg|_{t=0} = \underbrace{Dc_E(x^*)y}_{=0} + \underbrace{Dc_E(x^*)Dc_E(x^*)^t}_{\text{invertible}}\,\dot{u}(0)$$
   which implies $\dot{u}(0) = 0$, and thus
   $$\dot{x}(0) = y + Dc_E(x^*)^t\dot{u}(0) = y.$$

□

Lemma 2.10 *(1st order necessary condition for equality constraints)*
Let x^* a local minimum of 0.1 and regular wrt c_E. Then $\nabla f_0(x^*)$ is orthogonal to the tangent plane of $\mathcal{S}$ in x^*, that is
$$\nabla f_0(x^*) \cdot y = 0 \,\forall\, y \in \mathbb{R}^n : Dc_E(x^*)\, y = 0.$$

Proof: Let x^* a local optimum and regular point, $y \in \ker(Dc_E(x^*))$. By 2.9 there is a differentiable curve $x(t)$ on $\mathcal{S}$ with $x(0) = x^*$, $\dot{x}(0) = y$. Since x^* is a local minimum of f_0 on $\mathcal{S}$, $t = 0$ is a local optimum of $f_0(x(t))$, and thus

$$0 = \frac{d}{dt}[f_0(x(t))]\bigg|_{t=0} = \nabla f_0(x^*) \cdot y.$$

□

Note that this result is reminiscent of the remarks after 2.3. If the problem is unrestricted or x^* is in the interior of the feasible set, then the scalar product of $\nabla f_0(x^*)$ with any y must vanish. If we have restrictions, only the scalar products with vectors y that do not lead away from $\mathcal{S}$ (are tangential to $\mathcal{S}$) must vanish.

Theorem 2.11 *(Lagrange Multiplier)*
Let x^* a local optimum of 0.1 and regular wrt c_E. Then $\nabla f_0(x^*)$ is a linear combination of $\nabla h_1(x^*), \ldots, \nabla h_p(x^*)$ or

$$\exists \lambda \in \mathbb{R}^p : \ \nabla f_0(x^*) + Dc_E(x^*)^t \lambda = 0 \Longleftrightarrow Df(x^*) + \lambda^t Dc_E(x^*) = 0.$$

λ is called Lagrange Multiplier.

Proof: Recall from linear algebra that for any matrix $A \in \mathbb{R}^{p \times n}$

$$\mathbb{R}^n = \ker(A) \oplus \mathrm{range}(A^t) \Rightarrow \ker(A)^\perp = \mathrm{range}(A^t).$$

According to 2.10 we have

$$\nabla f(x^*) \in \ker(Dc_E(x^*))^\perp = \mathrm{range}(Dc_E(x^*)^t)$$

and thus for some $\lambda \in \mathbb{R}^p$

$$\nabla f(x^*) + Dc_E(x^*)^t \lambda = 0 \Longleftrightarrow Df(x^*) + \lambda^t Dc_E(x^*) = 0.$$

□

Definition 2.12 *(+Corollary, Lagrange Function)*
Let

$$L : \mathbb{R}^n \times \mathbb{R}^p \mapsto \mathbb{R}, \ L(x, \lambda) := f_0(x) + \lambda^t c_E(x).$$

Then

$$\nabla_x L(x, \lambda) = \nabla f_0(x) + Dc_E(x)^t \lambda, \ \nabla_\lambda L(x, \lambda) = c_E(x).$$

If $x^* \in \mathcal{S}$ is a local optimum and regular point, λ the corresponding Lagrange multiplier, then

$$\nabla L(x^*, \lambda) = \begin{pmatrix} \nabla_x L(x^*, \lambda) \\ \nabla_\lambda L(x^*, \lambda) \end{pmatrix} = 0.$$

Further

$$D_{x,x} L(x, \lambda) = D^2 f_0(x) + \sum_{k=1}^{p} \lambda_k D^2 h_k(x^*).$$

Theorem 2.13 *(2nd order necessary conditions for equality constraints)*
Let $f_0,\, h_k \in C^2$. Let x^* a local optimum of 0.1 and regular wrt c_E. Let λ the corresponding Lagrange multiplier. Then

$$D^2 f_0(x^*) + \sum_{k=1}^{p} \lambda_k D^2 h_k(x^*) = D_{x,x} L(x^*, \lambda)$$

is positive semidefinite on $T_{x^*}\mathcal{S}$.

Proof: Since everything is twice differentiable, the curve x constructed in 2.9 can safely be assumed to be twice differentiable (the implicit function theorem guarantees the existence of k times differentiable functions if F is k times differentiable). So let $y \in T_{x^*}\mathcal{S}$, x a curve on $\mathcal{S}$ with

$$x \in C^2,\; x(0) = x^*,\; \dot{x}(0) = y.$$

As in the proof of 2.10, we use that since x^* is an optimal point, $f_0(x(t))$ has a minimum at $t = 0$ and thus (2nd order necessary condition)

$$0 \leq \frac{d^2}{dt^2} f_0(x(t))|_{t=0} = \dot{x}(0)^t D^2 f_0(x^*) \dot{x}(0) + D f_0(x^*) \ddot{x}(0).$$

Since $c_E(x(t)) = 0$, we have

$$0 = \frac{d^2}{dt^2} \lambda^t c_E(x(t)|_{t=0} = \sum_{k=1}^{p} \lambda_k \dot{x}(0)^t D^2 h_k(x^*) \dot{x}(0) + \lambda^t D c_E(x^*) \ddot{x}(0).$$

Since $\lambda^t D c_E(x^*) = -D f_0(x^*)$, we get

$$0 \leq y^t \left[D^2 f_0(x^*) + \sum_{k=1}^{p} \lambda_k D^2 h_k(x^*) \right] y.$$

□

We suspect that an equivalent sufficient condition exists.

Theorem 2.14 *(2nd order sufficient condition for equality constraints)*
Let

$$x^* \in \mathbb{R}^n,\ \lambda \in \mathbb{R}^p,\ c_E(x^*) = 0,\ Df_0(x^*) + \lambda^t Dc_E(x^*) = 0,\ x^* \text{ regular point}\,.$$

Further, let

$$D^2 f_0(x^*) + \sum_{k=1}^{p} \lambda_k D^2 h_k(x^*) = D_{x,x} L(x^*, \lambda)$$

positive definite on $T_{x^*}\mathcal{S}$. Then x^* is a strict local optimum of 0.1.

One might suspect that the proof runs along the lines of the proof of the necessary conditions for the unrestricted problem. However, that technique will only work if $\mathcal{S}$ is convex, which for surfaces it will not be unless the surface is a plane. So we need a slightly more complicated approach.

Proof: Assume that x^* is not a strict local optimum of 0.1. Then for every $\epsilon_k > 0$, $\epsilon_k \to 0$, we find a $y_k \in \mathcal{S}$ with $||x^* - y_k|| \leq \epsilon_k$, $x^* \neq y_k$, $f_0(y_k) \leq f_0(x^*)$. We write y_k as

$$c_E(y_k) = 0,\ f_0(y_k) \leq f_0(x^*),\ y_k = x^* + \delta_k s_k,\ \delta_k > 0,\ ||s_k|| = 1,\ \delta_k \to 0.$$

Due to Bolzano–Weierstraß, s_k has a convergent subsequence. Wlog assume $s_k \to s^*$. Then we have

$$\begin{aligned} 0 &= \frac{c_E(x^* + \delta_k s_k) - c_E(x^*)}{\delta_k} \\ &= \frac{c_E(x^* + \delta_k s_k) - c_E(x^* + \delta_k s^*)}{\delta_k} + \frac{c_E(x^* + \delta_k s^*) - c_E(x^*)}{\delta_k} \\ &\to Dc_E(x^*) s^* \end{aligned}$$

which implies that s^* is in $T_{x^*}\mathcal{S}$.
Now fix k. Using Taylor, we have

$$\begin{aligned} 0 \geq f_0(y_k) - f_0(x^*) \quad &= \delta_k Df_0(x^*) s_k + \frac{\delta_k^2}{2} s_k^t D^2 f_0(\eta_{0,k}) s_k \qquad (E_0) \\ 0 = h_i(y_k) - h_i(x^*) \quad &= \delta_k Dh_i(x^*) s_k + \frac{\delta_k^2}{2} s_k^t D^2 h_i(\eta_{i,k}) s_k \qquad (E_i) \end{aligned}$$

Using the defining property of the Lagrange Multiplicator λ, we get for

$$E_0 + \sum_{i=1}^{p} \lambda_i E_i \Rightarrow 0 \geq \underbrace{\frac{\delta_k^2}{2}}_{>0} s_k^t \left[D^2 f_0(\eta_{0,k}) + \sum_{i=1}^{p} \lambda_i D^2 h_i(\eta_{i,k}) \right] s_k.$$

Now let $k \mapsto \infty$, then in the limit

$$0 \geq (s^*)^t \left[D^2 f_0(x^*) + \sum_{i=1}^{p} \lambda_i D^2 h_i(x^*) \right] s^*$$

which is a $\lightning$ since that matrix was assumed to be positive definite on $T_{x^*}\mathcal{S}$. □

2.3 Equality and Inequality Constraints

Finally, we permit that in 0.1 $p \geq 0$ and $m \geq 0$, so we might have both equality and inequality constraints. Note that in this section, we extend some definitions (Lagrange function, active set) to include inequality constraints.

Again, we wish to derive necessary and sufficient conditions. We begin by noting that if x^* is feasible and $f_j(x^*) < 0$, $j > 0$, there is a small neighborhood $\mathcal{U}$ of x^* such that $f_j(x) < 0 \, \forall x \in \mathcal{U}$.

So for a local minimum, the restriction $f_j(x) < 0$ can be completely ignored since it does not affect the set of feasible points in $\mathcal{U}$. We define

Definition 2.15 *(Active restrictions and regular points for inequality restrictions) Let x^* feasible. Let*

$$\mathcal{A}(x^*) = \{j > 0 : \; f_j(x^*) = 0\}.$$

$j \in \mathcal{A}(x^*)$ are called active indices, the corresponding f_j are called active restrictions.
If $0 < j \leq m$, $j \notin \mathcal{A}(x^*)$, f_j is called inactive restriction.
x^* is a regular point iff

$$\nabla h_1(x^*), \ldots, \nabla h_p(x^*), \nabla f_j(x^*) : j \in \mathcal{A}(x^*) \text{ are linearly independent.}$$

Inactive restrictions can be ignored for local minima.

Theorem 2.16 *(Karush–Kuhn–Tucker (KKT) conditions) Let x^* a locally optimal point of 0.1, x^* regular. Then*

$$\exists \lambda \in \mathbb{R}^p, \; \mu \in \mathbb{R}^m, \mu \geq 0 : \underbrace{\mu \, c_I(x^*) = 0}_{\text{cp–wise}}, \; Df_0(x^*) + \lambda^t Dc_E(x^*) + \mu^t Dc_I(x^*) = 0.$$

λ, μ are the Lagrange multipliers wrt 0.1.

Note the very common shortcut $\mu\, c_I(x^*) = 0$: This simply states that $\mu_i = 0$ if f_i is not active ($i \notin \mathcal{A}(x^*)$) (complementarity condition).

Proof: Let x^* locally optimal. Then it is also an optimal point for the problem

$$\min f_0(x),\; x \in \mathbb{R}^n,\; c_E(x) = 0,\; f_i(x) = 0,\; i \in \mathcal{A}(x^*)$$

which has only equality constraints. So 2.11 states

$$\exists \lambda \in \mathbb{R}^p,\; \mu \in \mathbb{R}^m :\; 0 = Df_0(x^*) + \lambda^t Dc_E(x^*) + \mu^t Dc_I(x^*)$$

where $\mu_i = 0$ for $i \notin \mathcal{A}(x^*)$.
Assume now that $\mu_k < 0$. Remove restriction f_k and let

$$S = \{x \in \mathbb{R}^n :\; c_E(x) = 0,\; f_j(x) = 0 \,\forall j \in \mathcal{A}(x^*) \setminus \{k\}\}.$$

Since x^* was a regular point wrt the original problem, x^* is also a regular point wrt $\mathcal{S}$. Assume that $y^t\, \nabla f_k(x^*) = 0$ for all $y \in T_{x^*}\mathcal{S}$. Then, observing 2.9,

$$\nabla f_k \in \ker(D\widetilde{c_E}(x^*))^\perp = \operatorname{range}(D\widetilde{c_E}(x^*)^t)$$

where $\widetilde{c}_E$ is the function that contains the restrictions for $\mathcal{S}$. Thus ∇f_k is a linear combination of the gradients of c_E and $f_j,\, j \in \mathcal{A}(x^*) \setminus \{k\}$. This is a contradiction, since we assumed that x^* is a regular point. So wlog

$$\exists y \in T_{x^*}\mathcal{S} :\; Df_k(x^*)\, y < 0.$$

Now let x a curve on $\mathcal{S}$ and $x(0) = x^*$, $\dot{x}(0) = y$. Since $x(t)$ is in $\mathcal{S}$, it satisfies all conditions except $f_k(x(t)) \leq 0$.
Since $Df_k(x^*)\,\dot{x}(0) < 0$, we have $f_k(x(t)) < 0$ for $t > 0$ sufficiently small, and $x(t)$ is feasible.
Inserting the Lagrange Multiplier gives, since $y \in \ker(D\widetilde{c_E}(x^*))$,

$$\frac{d}{dt} f_0(x(t))|_{t=0} = Df_0(x^*)y = -(\lambda^t Dc_E(x^*) + \mu^t Dc_I(x^*))y = -\mu_k Df_k(x^*)\, y < 0 \;\lightning\,.$$

□

For the second order conditions, we can simply refer to the case with equality constraints.

Theorem 2.17 *(2nd order necessary conditions)*
Let x^* a locally optimal point of 0.1 and regular. Let $\lambda \in \mathbb{R}^p$, $\mu \in \mathbb{R}^m$ the corresponding Lagrange multipliers from 2.16. Then

$$D^2 f_0(x^*) + \sum_{k=1}^{p} \lambda_k D^2 h_k(x^*) + \sum_{k=1}^{m} \mu_k D^2 f_k(x^*)$$

is positive semi–definite on the tangent space of the equality and active inequality constraints.

Proof: Again, since x^* is locally optimal, it is also optimal for the problem

$$\min f_0(x),\, x \in \mathbb{R}^n,\, c_E(x) = 0,\; f_i(x) = 0,\, i \in \mathcal{A}(x^*).$$

2.13 finishes the proof. □

Theorem 2.18 *(2nd order sufficient conditions)*
Let x^* a feasible point of 0.1 and regular. Let $\lambda \in \mathbb{R}^p$, $\mu \in \mathbb{R}^m$, $\mu \geq 0$, such that

$$\underbrace{\mu\, c_I(x^*) = 0}_{\text{cp–wise}},\; Df_0(x^*) + \lambda^t Dc_E(x^*) + \mu^t Dc_I(x^*) = 0$$

and

$$D^2 f_0(x^*) + \sum_{k=1}^{p} \lambda_k D^2 h_k(x^*) + \sum_{k=1}^{m} \mu_k D^2 f_k(x^*)$$

is positive semi–definite on the tangent space of the equality and active inequality constraints. Then x^* is a strictly locally optimal point.

Proof: We follow the lines of the proof of 2.14. Again, let

$$y_k = x^* + \delta_k s_k,\, \delta_k > 0,\, ||s_k|| = 1,\, y_k \to x^* :\; f_0(x_k) \leq f_0(x^*)$$

and wlog let $s_k \to s^*$. As in 2.14, we want to show that x^* is orthogonal to the gradients of h_k and the active f_k. We have $Dc_E(x^*)s^* = 0$ from the proof of 2.14. Let f_j active constraint. Then, again as in 2.14

$$f_j(y_k) - f_j(x^*) = f_j(y_k) \leq 0 \Rightarrow Df_j(x^*)s^* \leq 0.$$

Use the Lagrange multiplier to obtain

$$0 \geq Df_0(x^*)s^* = -\lambda^t Dc_E(x^*)s^* - \mu^t Dc_I(x^*)s^* = -\sum_{j=1}^{m} \underbrace{\mu_j}_{\geq 0} \underbrace{Df_j(x^*)\, s^*}_{\leq 0} \geq 0.$$

Thus, either $\mu_j = 0$ or $Df_j(x^*)s^* = 0$, in both cases we can continue as in 2.14. □

Chapter 3

Duality

To motivate this chapter, let us go back to the Kuhn–Tucker–conditions (2.16) and assume that we wish to establish a numerical algorithm for 0.1. This will typically be iterative, we will try to find a sequence x^k that converges to a minimizer. To check if we already have a minimum, we would have to check the existence of some pair λ, μ as in 2.16, which would require the solution of a possibly large linear system in every step.

It might be more feasible to state an augmented problem and include the Lagrangians into the problem definition, like: Find a triple (x^*, μ, λ) that satisfies 2.16, resulting in an iterative algorithm for $(x^{(k)}, \mu^{(k)}, \lambda^{(k)})$. In every iteration, we would have to update these values. It is clear how to do that for $x^{(k)}$, e.g. using steepest descent (page 15).

In this section, we will reformulate the original (primal) problem as a minimization (dual) problem in the parameters (μ, λ) that exchanges the roles of x and (μ, λ). First, we might be lucky that the dual problem is easier to solve than the primal problem.

Second, we might envision a two–step algorithm for each iteration of the very loose form

- Fix $\mu^{(k)}$, $\lambda^{(k)}$ and update $x^{(k)}$ to $x^{(k+1)}$ using the primal problem.
- Fix $x^{(k+1)}$ and update $\mu^{(k)}$, $\lambda^{(k)}$ to $\mu^{(k+1)}$, $\lambda^{(k+1)}$ using the dual problem.

Algorithms which follow this structure are called primal–dual (and are very popular in many fields).

We begin by extending the Lagrange function from 2.12 to the full problem 0.1 with inequalities in a straightforward way.

Definition 3.1 *(Lagrange function for equality and inequality constraints)*
In the terminology of 0.1, let

$$L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \mapsto \mathbb{R},\ L(x, \mu, \lambda) = f_0(x) + \mu^t c_I(x) + \lambda^t c_E(x).$$

Note that it is common to define L on all of $\mathbb{R}^n$ in the first variable by setting its value to ∞.

If x is feasible and $\mu \geq 0$, we have

$$L(x, \mu, \lambda) = f_0(x) + \underbrace{\mu^t}_{\geq 0} \underbrace{c_I(x)}_{\leq 0} + \lambda^t \underbrace{c_E(x)}_{=0} \leq f_0(x)$$

which implies that for

$$p(x) := \sup_{\mu \in \mathbb{R}^m_+, \lambda \in \mathbb{R}^p} L(x, \mu, \lambda)$$

we have

$$p(x) = \begin{cases} f_0(x) & x \in \mathcal{F} \\ \infty & \text{else.} \end{cases}$$

Proof: Assume that e.g. $f_1(x) > 0$. Then choose μ_1 arbitrarily large.

Thus, 0.1 is equivalent to the unconstrained problem with minimization function $p(x)$. This gives rise to

Definition 3.2 *(primal–dual problem)*

$$p(x) := \sup_{\mu \in \mathbb{R}^m_+,\, \lambda \in \mathbb{R}^p} L(x, \mu, \lambda)$$

is the primal objective function.

$$d(\mu, \lambda) := \inf_{x \in D} L(x, \mu, \lambda)$$

is the dual objective function. The primal problem then is

$$\min_{x \in D} p(x),$$

its dual problem (DP) is given by

$$\max_{\mu \in \mathbb{R}^m_+,\, \lambda \in \mathbb{R}^p} d(\mu, \lambda).$$

The primal problem is equivalent to 0.1.

Corollary 3.3 *(Weak Duality, Duality Gap, Strong Duality)*
Let p^* the optimal value of 0.1, d^* the optimal value of the dual problem. Then

$$d(\mu, \lambda) \leq p(x) \forall\, \mu \in \mathbb{R}^m_+, \lambda \in \mathbb{R}^p,\, x \in \mathcal{F}$$

and

$$d^* \leq p^*.$$

This is called weak duality. $p^* - d^*$ is the duality gap. If $p^* = d^*$, we say that strong duality holds.

Proof: See preliminary remarks. □
Note that for strong duality, the primal and dual problems are equivalent.

Corollary 3.4 *(concavity of the dual objective function)*
In 3.2, d is concave (and thus $-d$ is convex).

Proof: For $x \in D$, $\mu_k \in \mathbb{R}^m$, $\lambda \in \mathbb{R}^p$, we have

$$\begin{aligned} &L(x, \alpha\mu_1 + (1-\alpha)\mu_2, \alpha\lambda_1 + (1-\alpha)\lambda_2) \\ &= \alpha(f_0(x) + \mu_1^t c_I(x) + \lambda_1^t c_E(x)) + (1-\alpha)(f_0(x) + \mu_2^t c_I(x) + \lambda_2^t c_E(x)) \\ &= \alpha L(x, \mu_1, \lambda_1) + (1-\alpha) L(x, \mu_2, \lambda_2). \end{aligned}$$

Take the $\inf$ over all $x \in D$, then

$$d(\alpha\mu_1 + (1-\alpha)\mu_2, \alpha\lambda_1 + (1-\alpha)\lambda_2) \geq \alpha d(\mu_1, \lambda_1) + (1-\alpha) d(\mu_2, \lambda_2).$$

□

So even if the objective function for 0.1 is not convex, the objective function for the dual problem (minimization of $-d$) is.

Definition 3.5 *(saddle point)*
Let $f : A \times B \mapsto \mathbb{R}$. $(\overline{a}, \overline{b})$ is a saddle point of f iff

$$f(\overline{a}, b) \leq f(\overline{a}, \overline{b}) \leq f(a, \overline{b}) \forall a \in A,\, b \in B.$$

So $(\overline{a}, \overline{b})$ is a maximal point wrt b and minimal wrt a (looks like a saddle), or equivalently

$$f(\overline{a}, \overline{b}) = \inf_{a \in A} f(a, \overline{b}) = \sup_{b \in B} f(\overline{a}, b).$$

Lemma 3.6 *(and proof, InfSup–Principle)*
Let f as in 3.5. Then

$$\begin{aligned} \sup_{b \in B} \inf_{a \in A} f(a, b) &\leq \sup_{b \in B} \inf_{a \in A} \inf_{\tilde{b} \in B} f(a, \tilde{b}) \\ &= \inf_{a \in A} \sup_{\tilde{b} \in B} f(a, \tilde{b}) \end{aligned}$$

Theorem 3.7 *(Strong Duality and the Saddle Point)*
0.1 satisfies strong duality iff L has a saddle point $(x^*, (\mu^*, \lambda^*))$. x^* is a solution to the primal problem, (μ^*, λ^*) are a solution to the dual problem.

Proof: Let $(x^*, (\mu^*, \lambda^*))$ a saddle point of L. We have

$$\begin{aligned} L(x^*, \mu^*, \lambda^*) &= \inf_x L(x, \mu^*, \lambda^*) \\ &\leq \sup_{\mu,\lambda} \inf_x L(x, \mu, \lambda) \\ &\leq \inf_x \sup_{\mu,\lambda} L(x, \mu, \lambda) \\ &\leq \sup_{\lambda,\mu} L(x^*, \mu, \lambda) \\ &= L(x^*, \mu^*, \lambda^*). \end{aligned}$$

So equality must hold, and we have

$$L(x^*, \mu^*, \lambda^*) = \inf_x L(x, \mu^*, \lambda^*) = d(\mu^*, \lambda^*)$$

and

$$L(x^*, \mu^*, \lambda^*) = \sup_{\mu,\lambda} L(x^*, \mu, \lambda) = p(x^*)$$

and strong duality holds. Since $\infty > L(x^*, \mu^*, \lambda^*) = p(x^*)$, we have $x^* \in \mathcal{F}$. Since $d(\lambda, \mu) \leq p(x)$, optimality follows.
On the other hand, let $x^* \in \mathcal{F}$ a minimizer of 0.1, μ^*, λ^* solutions to the dual problem.

$$\begin{aligned} f_0(x^*) &= p^* = d^* = d(\mu^*, \lambda^*) \\ &= \inf_x L(x, \mu^*, \lambda^*) \\ &\leq L(x^*, \mu^*, \lambda^*) \\ &\leq \sup_{\mu,\lambda} L(x^*, \mu, \lambda) \\ &= p(x^*) = f_0(x^*) \end{aligned}$$

Again we have equality which implies

$$\inf_x L(x, \mu^*, \lambda^*) = L(x^*, \mu^*, \lambda^*)$$

and

$$\sup_{\mu,\lambda} L(x^*, \mu, \lambda) = L(x^*, \mu^*, \lambda^*).$$

That is the saddle point property. □

From this proof, we can draw another interesting conclusion: We have that

$$f_0(x^*) + (\mu^*)^t c_I(x^*) + (\lambda^*)^t c_E(x^*) = L(x^*, \mu^*, \lambda^*) = f_0(x^*).$$

This implies that $(\mu^*) c_I(x^*) = 0$ componentwise, which is very much reminiscent of the Kuhn Tucker–conditions. In fact, we have

Theorem 3.8 *(KKT from strong duality)*
Assume that 0.1 has the strong duality property. Let $f_k,\ h_k \in C^1$. Let x^* a solution to the primal problem. Let (μ^*, λ^*) a solution to the dual problem. Then μ^*, λ^* are the corresponding Lagrange multipliers of x^* wrt 0.1, and $(\mu^*)^t c_I(x^*) = 0$.

Proof: x^* is a local mimimizer of L (saddle–point property). Thus $D_x L(x^*, \mu^*, \lambda^*) = 0$, that gives the Lagrange property. Add preliminary remarks. □

Theorem 3.9 *(Duality from KKT for convex problems)*
Let 0.1 convex, and KKT satisfied for $(x^*,\ \mu^*,\ \lambda^*)$. Then x^* is primal optimal, (μ^*, λ^*) is dual optimal.

Proof: Since KKT is satisfied, we have $D_x L(x^*, \mu^*, \lambda^*) = 0$. Since $L(x, \mu^*, \lambda^*)$ is convex, x^* is a local minimizer wrt x (see next chapter). Since $c_E(x^*) = 0$ and $(\mu^*)^t c_I(x^*) = 0$ we have

$$d(\mu^*, \lambda^*) = L(x^*, \mu^*, \lambda^*) = f_0(x^*) \geq f_0(x^*) + \mu^t c_I(x^*) + \lambda^t c_E(x^*) \geq L(x^*, \mu, \lambda)$$

and μ^*, λ^* is a local maximizer. □

Note that strong duality is very useful in computation. Assume that we have an approximation (x, ν, λ) to the optimal primal/dual solutions. Then for strong duality

$$f_0(x) - p^* = f_0(x) - d^* \leq f_0(x) - d(\mu, \lambda).$$

The right hand side can be computed and gives us an upper bound on the minimization error of x (stopping criterion!).

Chapter 4

Convexity

We begin by remembering some definitions and theorems from analysis I+II.

Definition 4.1 *(convex sets and functions)*
Let $C \subset \mathbb{R}^n$, $f : C \mapsto \mathbb{R}$.

1. *Let*
$$x_k \in C,\ \lambda_k \in [0,1],\ k = 1 \dots m, \sum_{k=1}^{m} \lambda_k = 1,\ \overline{x} = \sum_{k=1}^{m} \lambda_k x_k.$$
$\overline{x}$ is called convex combination of x_k.
2. *C is convex iff*
$$\lambda x + (1-\lambda)y \in C \,\forall\, x, y \in C, \lambda \in [0,1].$$
3. *Let C convex. f is convex iff*
$$f(\lambda x + (1 - \lambda y)) \leq \lambda f(x) + (1-\lambda)f(y) \,\forall\, x, y \in C,\ \lambda \in [0,1].$$
4. *Let C convex. f is strictly convex iff*
$$f(\lambda x + (1-\lambda)y) < \lambda f(x) + (1-\lambda)f(y) \,\forall\, x, y \in C, x \neq y,\ \lambda \in (0,1).$$

Definition 4.2 *(convex hull)*
Let $C \subset \mathbb{R}^n$. The convex hull $\operatorname{conv} C$ contains all convex combinations of the elements of C. $\operatorname{conv} C$ is the smallest subset of $\mathbb{R}^n$ that contains C.

Theorem 4.3 *(Caratheodory)*
Let $C \subset \mathbb{R}^n$, $x \in \operatorname{conv} C$. Then x is a convex combination of at most $n+1$ elements of C.

Proof: Exercises. □

Lemma 4.4 *(convexity for differentiable functions)*
Let $C \subset \mathbb{R}^n$ convex, $f : C \mapsto \mathbb{R}$ differentiable.

1. *f is convex iff the graph of f lies above each tangential plane, that is*

$$f(y) \geq f(x) + \nabla f(x)^t (y - x) \,\forall\, x, y \in C.$$

2. *f is strictly convex iff*

$$f(y) > f(x) + \nabla f(x)^t (y - x) \,\forall\, x, y \in C,\ x \neq y.$$

Proof: Let

$$g(\lambda) = f(x + \lambda(y - x)) \Rightarrow g'(\lambda) = \nabla f(x + \lambda(y - x))^t (y - x).$$

Let f convex, then

$$g(\lambda) \leq \lambda g(1) + (1 - \lambda) g(0) \Rightarrow \frac{g(\lambda) - g(0)}{\lambda} \leq g(1) - g(0)$$

which implies the inequality for $\lambda \to 0$.
Now assume that the inequality holds for all points between x and y, then

$$\begin{aligned} g(0) &\geq g(\lambda) - g'(\lambda)\,\lambda && (I) \\ g(1) &\geq g(\lambda) + g'(\lambda)\,(1 - \lambda) && (II) \end{aligned}$$

and thus

$$(1 - \lambda)(I) + \lambda(II) \Rightarrow (1 - \lambda) g(0) + \lambda g(1) \geq g(\lambda)$$

which implies convexity of f. □

Lemma 4.5 *(convexity for twice differentiable functions)*
Let $C \subset \mathbb{R}^n$ convex, $f : C \mapsto \mathbb{R}$ twice differentiable.

1. *f is convex iff*

$$D^2 f(x) \textit{ positive semidefinite } \forall\, x \in C.$$

2. *f is strictly convex if*

$$D^2 f(x) \textit{ positive definite } \forall\, x \in C.$$

Proof: Exercises, reduce to the one–dimensional problem as above and then use the proof from e.g. Forster, Analysis I, Paragraph 16, Theorem 5. □

Theorem 4.6 *(Existence and Uniqueness of Global Solutions for convex problems) Let $C \subset \mathbb{R}^n$ convex, $f : C \mapsto \mathbb{R}$ convex. We examine the problem*

$$\min_{x \in C} f(x).$$

1. *All local minima are global.*
2. *The set of global minima is convex.*
3. *If f is strictly convex, then there is at most one global solution.*
4. *Let f differentiable. If $\nabla f(x) = 0$, then x is a local minimum (sufficient and necessary condition).*

Proof:

1. Let $f(x) > f(y)$. Then for $\epsilon \in (0, 1]$

$$\begin{aligned} f(x + \epsilon(y - x)) &= f((1 - \epsilon)x + \epsilon y) \\ &\leq (1 - \epsilon) f(x) + \epsilon f(y) \\ &= f(x) + \epsilon \underbrace{f(y) - f(x)}_{<0} \\ &< f(x) \end{aligned}$$

   so x is not a local minimum.

2. Let x_1, x_2 two global minima, so $f(x_1) = f(x_2)$ and for $x = \lambda x_1 + (1 - \lambda)x_2$

$$f(x_1) \leq f(x) \leq \lambda f(x_1) + (1 - \lambda) f(x_2) = f(x_1) \, \forall \lambda \in [0, 1]$$

   and x is also a minimum.

3. Let x,y, $x \neq y$, global minima. From 2., we have that $\frac{1}{2}(x + y)$ is also a global minimum, but since f is strictly convex

$$f(\frac{1}{2}(x + y)) < \frac{1}{2}(f(x) + f(y)) = f(x)$$

   and x is not a global minimum, so we must have $x = y$.

4. (4.4).

□

Finally, we will show that convex minimization problems (almost) always possess the strong duality property. This justifies our intense treatment of dual problems in the last chapter.

Definition 4.7 *(projection onto convex set)*
Let $C \in \mathbb{R}^n$ closed and convex. Then

$$P_C : \mathbb{R}^n \mapsto C, \; P_C x := \arg\min_{y \in C} ||y - x||_2^2$$

is properly defined and the orthogonal projection onto C.

Theorem 4.8 *(Characterization of orthogonal projection)*
Let C closed and convex, $x \in C$. $x^+ = P_C x$ iff

$$(x - x^+, y - x^+) \le 0 \, \forall y \in C.$$

So P_C has exactly the same property as the orthogonal projection onto a halfspace.

Proof: Let $x^+ = P_C x$, $y \in C$, $\lambda \in (0, 1]$. Then

$$||x - x^+||^2 \le ||x - (\underbrace{x^+ + \lambda(y - x^+)}_{\in C})||^2 = ||x - x^+||^2 - 2\lambda(x - x^+, y - x^+) + \lambda^2 ||y - x^+||^2$$

which can only hold for $(x - x^+, y - x^+) \le 0$ (divide by λ, let $\lambda \to 0$).
Now assume that the inequality holds for an $x^+ \in C$ and for all $y \in C$. Then with Cauchy–Schwarz

$$0 \ge (x - x^+) \cdot (y - x^+) = (x - x^+) \cdot (x - x^+ + y - x) \ge ||x - x^+||^2 - ||x - x^+|| \, ||y - x||$$

and thus

$$||x - x^+||(||y - x|| - ||x - x^+||) \ge 0.$$

If $x = x^+$, we have $x \in C$ and thus $P_C x = x = x^+$. Otherwise we have $||x - x^+|| \le ||x - y||$ for all $y \in C$, and $P_C x = x^+$. □

Theorem 4.9 *(Separation of convex sets by hyperplanes)*

1. *Let $C \subset \mathbb{R}^n$ convex and closed, $x \notin C$. Then*

$$\exists \, s \in \mathbb{R}^n : \; s \cdot x > \sup_{y \in C} s \cdot y.$$

2. *Let C_1, C_2 convex, closed, disjoint and nonempty. If C_2 is bounded (i.e. compact), then*
$$\exists s \in \mathbb{R}^n : \sup_{y_1 \in C_1} s \cdot y_1 < \min_{y_2 \in C_2} s \cdot y_2.$$
The hyperplane
$$s \cdot z = \alpha = \frac{1}{2}(\min_{y \in C_2} s \cdot y + \sup_{y_1 \in C_1} s \cdot y)$$
separates C_1 and C_2, meaning
$$s \cdot y_1 < \alpha < s \cdot y_2 \forall y_1 \in C_1,\, y_2 \in C_2.$$

3. *Let C_1, C_2 convex, nonempty and disjoint, but not necessarily closed or bounded. Then $<$ becomes $\leq$ in 2., i.e.*
$$\exists s \in \mathbb{R}^n,\, s \neq 0 : \sup_{y_1 \in C_1} s \cdot y_1 \leq \inf_{y_2 \in C_2} s \cdot y_2.$$

Proof: (for 1. and 2., see exercises for 3)

1. Let $s = x - P_C x$, $y \in C$. Then by the previous theorem
$$0 \geq (x - P_C x) \cdot (y - P_C x) = s \cdot (y - x + s) = (s \cdot y - s \cdot x) + ||s||^2.$$

2. $C = C_1 - C_2$ is convex and closed (C_2 is compact). Then in 1. choose $x = 0$ to wit
$$0 = s \cdot 0 > \sup_{y \in C} s \cdot y = \sup_{y_1 \in C_1} s \cdot y_1 - \min_{y_2 \in C_2} s \cdot y_2.$$

□

Theorem 4.10 *(Slater's condition)*
Let 0.1 a convex problem, and $c_E(x) = Ax - b$, $A \in \mathbb{R}^{p \times n}$ has rank p. Assume that a feasible point $\widetilde{x}$ in the interior of $\mathcal{D}$ exists with $c_I(\widetilde{x}) < 0$. Then strong duality holds.

Note that the condition must hold for *one arbitrary feasible point*, not necessarily for x^*.

Proof: Since there is a feasible point, we have $p^* < \infty$. Wlog $p^* > -\infty$. We also assume that $\widetilde{x}$ is in the interior of $\mathcal{D}$.
Let

$$M \subset \mathbb{R}^m \times \mathbb{R}^p \times \mathbb{R},\, M := \{(u, v, t) | \exists x \in \mathcal{D} : c_I(x) \leq u,\, c_E(x) = v, f_0(x) \leq t.\}$$

and

$$N \subset \mathbb{R}^m \times \mathbb{R}^p \times \mathbb{R},\, N := \{(0,0,s)|s < p^*\}.$$

For all $x \in \mathcal{D}$ $(c_I(x), c_E(x), f_0(x)) \in M$.
x is feasible iff $(0,0,f_0(x)) \in M$.
Let $(0,0,s) \in N$. There is no feasible x such that $f_0(x) \leq s < p^*$, so N and M are disjoint.
From 4.9, we have the existence of (μ, λ, ν), $\alpha \in \mathbb{R}$, such that

$$\mu^t u + \lambda^t v + \nu t \geq \alpha \,\forall\, (u,v,t) \in M$$

and

$$\alpha \geq \nu t \,\forall\, (0,0,t) \in N \Rightarrow \alpha \geq \nu p^*.$$

Since $(c_I(x), c_E(x), f_0(x)) \in M$

$$\mu^t c_I(x) + \lambda^t c_E(x) + \nu f_0(x) \geq \nu p^* \,\forall x \in \mathcal{D}.$$

If $\nu > 0$, we have

$$p^* \leq f_0(x) + (\mu/\nu)^t c_I(x) + (\lambda/\nu)^t c_E(x) = L(x, \mu/\nu, \lambda/\nu) \,\forall\, x \in \mathcal{D}$$

which implies

$$p^* \leq \inf_{x \in \mathcal{D}} L(x, \mu/\nu, \lambda) = d(\mu/\nu, \lambda/\nu) \leq \max_{\mu' \geq 0, \lambda'} d(\mu', \lambda') = d^*$$

and thus $p^* = d^*$ and we have strong duality.
Assume $\nu < 0$. Since $(c_I(\widetilde{x}), c_E(\widetilde{x}), t) \in M \,\forall t > f_0(\widetilde{x})$, in the first inequality the left hand side would be unbounded to below. So $\nu \geq 0$ and $\mu \geq 0$ with the same argument.
Assume $\nu = 0$. Then since $\widetilde{x} \in D$, $\widetilde{x}$ feasible and $c_I(\widetilde{x}) < 0$

$$\underbrace{\mu^t}_{\geq 0} \underbrace{c_I(\widetilde{x})}_{<0} + \lambda^t \underbrace{c_E(\widetilde{x})}_{=0} \geq 0 \Rightarrow \mu = 0.$$

Thus

$$\lambda^t c_E(x) \geq 0 \,\forall\, x \in \mathcal{D}.$$

Since $\widetilde{x}$ is in the interior of $\mathcal{D}$,

$$\widetilde{x} - \epsilon A^t \lambda \in D \text{ for } \epsilon \text{ sufficiently small.}$$

Thus

$$0 \leq \lambda^t (A(\widetilde{x} - \epsilon A^t \lambda) - b) = -\epsilon \lambda^t (A A^t \lambda) = -\epsilon ||A^t \lambda||^2 \Longrightarrow A^t \lambda = 0.$$

Since A has full rank, $\lambda = 0$, and thus $\mu = 0$, $\lambda = 0$, $\nu = 0$ ↯. □

4.9 has more useful consequence: For all points x on the boundary of a convex set C, there exists a variant of a tangent, a hyperplane H with $x \in H$ and C is on one side of the hyperplane. Note that this is even true for non–differentiable boundaries.

Theorem 4.11 *(+ Definition, supporting hyperplane)*
Let $C \subset \mathbb{R}^n$ convex, $x \in \partial C$. Then

$$\exists s \in \mathbb{R}^n,\ s \neq 0 :\ s \cdot (x - y) \geq 0\, \forall y \in C.$$

The hyperplane

$$H = \{z : s \cdot z = s \cdot x\}$$

is called supporting hyperplane.

Proof:

$$x \in \partial C \Rightarrow \exists (x_k) :\ x_k \to x,\ x_k \notin \overline{C}.$$

Since $x_k \notin \overline{C}$, due to 4.9

$$\exists s_k \in \mathbb{R}^n :\ s_k \cdot (x_k - y) > 0\, \forall y \in C.$$

Wlog assume $||s_k|| = 1$ (else let $s_k := s_k/||s_k||$). s_k lies in the (compact) unit ball, so it has a convergent subsequence. Wlog $s_k \to s \in \mathbb{R}^n$. Then

$$s_k \cdot (x_k - y) > 0 \Rightarrow s \cdot (x - y) \geq 0.$$

□

H is not necessarily unique (e.g. vertex of a polyhedron), nor does $>$ necessarily hold for $x \neq y$ in the inequality (boundary might be a line segment).

Definition 4.12 *(Extreme points)*
Let $C \subset \mathbb{R}^n$ convex. $x \in C$ is an extreme point of C iff x is not the strict convex combination of two points in C, that is

$$x \neq \lambda x_1 + (1 - \lambda) x_2\, \forall x_1,\ x_2 \in C,\ x_1 \neq x_2,\ \lambda \in (0, 1).$$

Note that if x is not an extreme point, then there are $x_1, x_2 \in C$ such that $x = \frac{1}{2}(x_1 + x_2)$. For C a convex polyhedron, the extreme points are its vertices. Clearly, all extreme points are on the boundary of C.

Theorem 4.13 *(Minkowski)*
Let C compact and convex. Then C is the convex hull of its extreme points.

Proof: Since C is convex, the convex hull of its extreme points is in C.
For the other direction, we use geometric intuition in $\mathbb{R}^2$.
Assume first that $x \in \partial C$. Then there is a supporting hyperplane H in x, which is a line in R^2. If x is not an extreme point, then there are $x_1,\, x_2 \in C$ with $x = \frac{1}{2}(x_1 + x_2)$, and $x_1,\, x_2 \in H$. So a segment of the line through (x_1, x, x_2) is on the boundary. Since C is bounded, that segment must have a starting point x_S and an endpoint x_E. Since C is closed, these must be in C. Also, they are not the strict linear combination of two points on H. So either x is an extreme point, or it is the linear combination of the extreme points x_S and x_E.
Now let x in the interior of C. Take any line L through x. It will hit the boundary in exactly two points, which are a convex combination of extreme points, so x is a convex combination of extreme points.
For $\mathbb{R}^n$, use induction over n. □

Chapter 5

Linear Programming

5.1 Problem formulation and characterization of minimal points

We already introduced linear optimization as a profit maximization problem in economy in 0.2.1. The general in 0.1 had f_0, c_I and c_E affine linear.

We reformulate this using the slack variables from 1.2 to turn inequalities into equalities, and setting $x_k = x_k^+ - x_k^-$, x_k^+, $x_k^- \geq 0$, for unbounded variables.

Definition 5.1 *(linear program)*
A linear program in standard form is given by

$$\min_{x \in \mathbb{R}^n} cx \text{ where } Ax = b,\ x \geq 0 \qquad (LP)$$

and $c \in \mathbb{R}^{1\times n}$, $c \neq 0$, $A \in \mathbb{R}^{p\times n}$, $b \in \mathbb{R}^p$, $p \leq n$, $\operatorname{rank} A = p$. Thus, we minimize cx over the feasible set given by the convex polyhedron

$$K = \{x \geq 0 :\ Ax = b\}.$$

Theorem 5.2 *(extreme points and optimal point)*
Assume that the set X of solutions to 5.1 is nonempty. Then X contains at least one extreme point.

Proof: Assume that K is compact. Since K is a polyhedron, the set $\operatorname{ext} K$ of extreme points is finite (also see next theorem), and

$$\exists z \in \operatorname{ext} K :\ cz \leq cy\, \forall\, y \in \operatorname{ext} K.$$

Now let $x \in K$. Due to 4.13

$$\exists \lambda_i \in [0,1],\, x_i \in \operatorname{ext} K,\, i = 1 \dots r : \sum_{i=1}^{r} \lambda_i = 1,\, x = \sum_{i=1}^{r} \lambda_i x_i.$$

Then

$$cx = c(\sum_{i=1}^{r} \lambda_i x_i) \geq cz$$

and $z \in X$. □

Alternate proof: Let x a solution of 5.1. Then

$$H = \{z : cz = cx\}$$

is a supporting hyperplane, and $H \not\subset K$ since K is a subset of the upper right quadrant. From the geometric intuition in 4.13: H contains at least one extreme point.

The theorem says that in order to solve 5.1, we must only consider the extreme points. So let us characterize them.

Definition 5.3 *(basic feasible point)*
In 5.1, let $\mathcal{I} \subset \{1, \dots, n\}$ an index set. Let a_i the columns of A. $x \in \mathbb{R}^n$ is a basic point wrt to the basis $\mathcal{I}$ iff

$$\{a_i : i \in I\} \textit{ linearly independent},\, x_i = 0 \,\forall\, i \notin \mathcal{I},\, Ax = b.$$

Note that this means $Bx_{\mathcal{I}} = b$ for $B = (a_i)_{i \in \mathcal{I}}$
Since B has full rank, x is unique for fixed $\mathcal{I}$.
x is a basic feasible point iff x is a basic point and $x \geq 0$.

Theorem 5.4 *(extreme and basic feasible points)*
$x \in \mathbb{R}^n$ is an extreme point of 5.1 iff it is a basic feasible point.

Note that this gives us a very simple algorithm for computing a minimal point: Just take all index sets of $\{1 \dots n\}$, compute the corresponding basic feasible points $x_{\mathcal{I}}$ (if they exist), and take their minimal value wrt cx. The number of feasible points is smaller or equal to the number of subsets of $\{1 \dots n\}$ with at most p elements.

Of course, this will only work as long as there is a solution to the problem. In the real algorithm, we will have to deal with that.

Proof: First, let x an extreme point of K, and wlog $x_1, \dots, x_r > 0$, $x_{r+1}, \dots x_n = 0$. Then

$$x_1 a_1 + \dots x_r a_r = Ax = b.$$

Assume that the a_k are linearly dependent, so with at least one $y_k \neq 0$

$$y_1 a_1 + \ldots + y_r a_r = 0 \Rightarrow Ay = 0,\; y = (y_1, \ldots, y_r, 0, \ldots, 0).$$

Then for $\epsilon > 0$ sufficiently small,

$$x \pm \epsilon y \geq 0 \text{ and } A(x \pm \epsilon y) = Ax = b \Rightarrow x \pm \epsilon y \in K$$

and $y \neq 0$. Since

$$x = \frac{1}{2}((x + \epsilon y) + (x - \epsilon y)),$$

x is not an extreme point of K $\lightning$, so the a_i must be linearly independent.
Assume now that x is a basic feasible point. Wlog assume $\mathcal{I} = \{1, \ldots, r\}$, so $x = (x_1, \ldots, x_r, 0, \ldots, 0)$.
Assume $x = \frac{1}{2}(y + z)$ with $y, z \in K$. Since $y, z \geq 0$, this implies

$$y_{r+1} = \ldots = y_n = z_{r+1} = \ldots = z_n = 0.$$

and since $Ay = b = Az$

$$y_1 a_1 + \ldots y_r a_r = b = z_1 a_1 + \ldots z_r a_r.$$

Thus $y = z$ since $a_1, \ldots, a_r$ are linearly independent, which implies that x is an extreme point. □

Examples:

- $$\min / \max x_1 + 2x_2 : \; x_1 + x_2 = 1,\; x_k \geq 0.$$
- $$\min / \max x_1 + 2x_2 + 3x_3 : \; x_1 + x_2 + x_3 = 1,\; x_k \geq 0.$$
- $$\min / \max x_1 + x_2,\; x_k \geq 0.$$
- $$\min / \max 10x_1 + x_2 : \; x_1 \leq 3/4,\; x_2 \leq 3/4, x_1 + x_2 \leq 1,\; x_k \geq 0.$$

  Note that this one is not in normal form! Follow the graphical solution in $\mathbb{R}^2$ and notice that this corresponds to updating the active set in the standard formulation with slack variables in $\mathbb{R}^5$.

5.2 The simplex algorithm: Derivation of phase II

We introduce the *idea* of the simplex algorithm. We solve the model problem where we assume that the equalities are given in the form of

$$(I\,A)\begin{pmatrix} z_B \\ z_N \end{pmatrix} = b \tag{5.1}$$

with $b > 0$, where $I \in \mathbb{R}^{p\times p}$ is the unit matrix and $A \in \mathbb{R}^{p\times n-p}$. $B = \{1 \ldots p\}$ are the basis indices, $N = \{p+1 \ldots n\}$ are the non–basis indices.

The form of the matrix can always be achieved by row transforms as in Gaussian elimination, since the rank of the $p \times n$ matrix in 5.1 was p, $n \geq p$, possibly by exchanging the order of variables.

Note that this form is optimized for the Simplex tableau (to be introduced below). In the literature, often, the more general form

$$(CA)\begin{pmatrix} z_B \\ z_N \end{pmatrix} = b$$

for an invertible $p \times p$–matrix C is assumed. Multiplying the equation from the left with C^{-1}, however, gives 5.1:

$$\left(I\ (C^{-1}A)\right)\begin{pmatrix} z_B \\ z_N \end{pmatrix} = C^{-1}b.$$

Since we assumed $b > 0$, choosing $x_B = b$, $x_N = 0$ is a feasible solution. It is a basic feasible point of 5.1 choosing the index set $I = \{1 \ldots p\}$, since the columns of $(IA)_i$ for $i \in I$ are the (linearly independent) unit vectors.

Our idea is to take $k \in N$ and $l \in B$, and find a new basic feasible point y that exchanges the roles of k and l, i.e.

1. $(IA)y = b$.
2. $y \geq 0$.
3. $y_l = 0$.
4. $y_k \neq 0$ (possibly).
5. $y_i = 0, i \in N, i \neq k$.
6. $cy < cx$.

In essence, this says: Move to a neighbouring basic feasible point, that improves the value of f_0 by exchanging the indices k and l, moving l out of the basis and k into the basis.

First we have

$$(IA)\begin{pmatrix} z_B \\ z_N \end{pmatrix} = b \Leftrightarrow z_m + \sum_{i\in N} A_{m,i} z_i = b_m \, \forall\, m \in B \tag{5.2}$$

where

$$A = (A_{m,i}),\ m = 1 \dots p, i = p + 1 \dots n.$$

Note the (nonstandard) indexing on A which starts at $p+1$ for the columns.

Since $l \in B$ and y is feasible, we can set $z = y$, $m = l$ in 5.2 and using the properties of y

$$b_l = \underbrace{y_l}_{=0} + \sum_{i\in N} A_{l,i} \underbrace{y_i}_{=0, i\neq k} = A_{l,k} y_k \Rightarrow y_k = \frac{b_l}{A_{l,k}}.$$

Note that this assumes that A_{lk} is not zero. We will later formulate the algorithm such that $A_{l,k} > 0$.

For $Ay = b$ to hold, 5.2 must be satisfied also for all $m \in B$, $m \neq l$, so

$$b_m = y_m + \sum_{i\in N} A_{m,i} \underbrace{y_i}_{=0, i\neq k} = y_m + A_{m,k} y_k \Rightarrow y_m = b_m - A_{m,k} y_k.$$

Since y_k has already been determined, this determines y. Note that y satisfies $Ay = b$, but is not necessarily feasible (not necessarily nonnegative).

Remember that what we are essentially doing is move one index from the basis to the non–basis and vice versa. If we want to keep up our initial ordering of variables, we must formally exchange the elements in our vector y and the corresponding columns in the matrix (IA). This will destroy the nice form of the matrix, in fact it then becomes

$$(e_1 \dots e_{l-1} a_k e_{l+1} \dots e_p, a_{p+1} \dots a_{k-1} e_l a_{k+1} \dots a_n) \tag{5.3}$$

where the a_k are the *columns* of A.

However, the original form is easily restored using Gaussian Elimination–style row transforms.

1. Divide the lth equation of $Az = b$ by the diagonal element $(a_l)_k = A_{k,l}$ of the new matrix (yields a 1 on the diagonal).

2. Subtract a multiple of the lth equation from the other equations (yields zeros above and below the diagonal).

This boils down to simple row operations on A and b.

The only question left is: How do we choose the indices such that the new point satisfies $y \geq 0$, and the value of f_0 decreases.

First, we note that for any feasible z we have

$$z_B + A\,z_N = b \Rightarrow c\,z = \underbrace{c_B\,b}_{=f_0(x)} + \underbrace{(c_N - c_B A)}_{=:r}\,z_N. \tag{5.4}$$

Note that $r \in \mathbb{R}^{1\times n-p}$ is a row vector.

Clearly, this means that the value of f_0 will potentially decrease for a basis change if $r_k < 0$. If there is no such k, then there is no direction in which the value could decrease, so x is an optimal point. Note that this argument is also true when the problem is unbounded to below.

In view of 5.2, if $b_k = 0$, the objective value will not decrease even if $r_k < 0$. This is called the degenerate case, see next section.

Assume now that $r_k < 0$. For all $l \in B$, compute the corresponding y and check whether they lead to a feasible solution. If no feasible solution exists, then the problem is unbounded to below (exercises, this is a direct consequence of 5.2).

If the exchange of variables leads to a feasible solution, do the exchange, update all variables and iterate.

5.3 The simplex algorithm: Implementation and the simplex tableau

For the simplex algorithm, we compute a series of basic feasible points, characterized by equivalent representations of our minimization problem.

In each step, we have a matrix A, a right hand side b (which also holds the nonzero elements of our current vertex x), a cost vector r and the value $f_0(x)$. Also, since we are exchanging elements in x, we must keep track of their order in the sequence $I \in \{1..n\}^n$, where the first p elements are the indices in B and $p+1\ldots n$ are the indices in N with respect to the original order.

So we get sequences for all these variables, we denote them $A^{(m)}$, $b^{(m)}$, $r^{(m)}$, $x^{(m)}$, $c^{(m)} = f_0(x^{(m)})$, $I^{(m)}$. Note that we do not have to track $x^{(m)}$, it is constructed from $b^{(m)}$ and $I^{(m)}$.

We start from the model problem 5.1, use 5.4 and set

$$A^{(0)} := A,\ b^{(0)} := b,\ I^{(0)} = \{1 \dots n\}, r^{(0)} = c_N - c_B\,A,\ c^{(0)} = f_0(x^{(0)}).$$

In iteration m, we first decide which variable to move from N to B (k in the outline). This can be any index that satisfies $r_k^{(m)} < 0$. If no such k exists, $x^{(k)}$ is optimal and we have reached a solution.

Otherwise, k is chosen as the index where $r_k^{(m)}$ is minimal, although this choice is not necessarily optimal (exercises). $l \in B$ is then chosen such that y is feasible. If no such l exists, the problem is unbounded (exercises).

After k and l are chosen, we need to update our sequences. Since we are exchanging indices k and l, $I^{(m+1)}$ is generated by exchanging the elements k and l in $I^{(m)}$.

$A^{(m+1)}$ and $b^{(m+1)}$ are generated using 5.3.

Now for the update on the representation of f_0: From 5.4 we have

$$f_0(z) = c^{(m)} + \sum_{i \in N} r_i^{(m)} z_i$$

where, as for A, we set $r = (r_i)$, $i = p+1 \dots n$. Since we are exchanging the roles of indices k and l, this representation must be updated by letting i go through $(N \setminus \{k\}) \cup \{l\}$.

Once more using 5.2, we have for any z that satisfies the equation constraint

$$z_l + \sum_{i \in N} A_{l,i}^{(m)} z_i = b_l^{(m)} \Rightarrow A_{l,k}^{(m)} z_k = b_l^{(m)} - z_l - \sum_{i \in N \setminus \{k\}} A_{l,i}^{(m)} z_i.$$

We already assumed that the pivot element $A_{l,k}^{(m)}$ does not vanish, so

$$r_k^{(m)} z_k = \frac{r_k^{(m)}}{A_{l,k}^{(m)}} (b_l^{(m)} - \sum_{i \in N \setminus \{k\}} A_{l,i}^{(m)} z_i - z_l).$$

Inserting this representation in 5.4 and letting

$$c^{(m+1)} = c^{(m)} + \frac{r_k^{(m)} b_l^{(m)}}{A_{l,k}^{(m)}},\ r_i^{(m+1)} = \begin{cases} -\frac{r_k^{(m)}}{A_{l,k}^{(m)}} & i = l \\ r_i^{(m)} - \frac{r_k^{(m)} A_{l,i}^{(m)}}{A_{l,k}^{(m)}} & i \in N \setminus \{k\} \end{cases}$$

we have

$$f_0(z) = c^{(m+1)} + \sum_{(N\setminus\{k\})\cup\{l\}} r_i^{(m+1)} z_i.$$

Note that formally the indices k and l must still be exchanged.

Typically, the single iterates (except for $I^{(k)}$) are kept in the matrix (simplex tableau)

$$T^{(m)} = \begin{pmatrix} r^{(m)} & -c^{(m)} \\ A^{(m)} & b^{(m)} \end{pmatrix} \in \mathbb{R}^{p+1\times n-p+1}.$$

The point of the tableau is that the updates on the variables can be formulated very easily using the tableau (exercises).

Last question: Is this algorithm guaranteed to always return the optimal result if one exists? Unfortunately, the answer is no. Remember that for degenerate solutions, we get no improvement of the objective function, so we might be exchanging the same pair of variables over and over again. This can be overcome, but we don't treat it here.

In the above derivation, we generally assumed that $b \geq 0$, which guarantees that $(b, 0)$ is a basic feasible solution (which in turn guarantees that at least one feasible solution exists). This is generally called phase II of the simplex algorithm. In phase I, we drop that requirement by solving a different minimization problem that produces a basic feasible solution.

Consider

$$\min_{y\in\mathbb{R}^p,\, x\in\mathbb{R}^n} (1, \ldots, 1)y \text{ where } Ax + y = b,\ y, x \geq 0.$$

We can safely assume here that $b_k \geq 0$. Obviously, this problem satisfies the requirements for phase II. So we can use the phase II program to find the minimum, which is a basic feasible point of problem I if the minimum value is 0.

We summarize the results.

Theorem 5.5 *(The Simplex algorithm)*
We consider the simplex algorithm for 5.1.

1. *The phase I problem is neither unbounded to below nor infeasible.*
2. *If phase I stops with a minimum value $f_0(x) > 0$, then 5.1 is infeasible.*
3. *If phase I returns with $f_0(x) = 0$, then x is a basic feasible point.*
4. *If phase II stops in step m because $r^{(m)} \geq 0$, then $x^{(m)}$ is a solution to 5.1.*

5. *If phase II stops in step m because $A_{i,k}^{(m)} < 0$ for all $i \in B$, then 5.1 is unbounded to below.*
6. *If the $b^{(m)}$ are non–degenerate for all m, then the simplex algorithm will terminate.*

Chapter 6

Smooth optimization for problems without constraints

In the following, we consider the general optimization problem

$$\min_{x \in \mathbb{R}^n} f(x),\ f : \mathbb{R}^n \mapsto \mathbb{R} \text{ smooth} \tag{6.1}$$

where smooth means $f \in C^1$ or even C^2.

6.1 Line search (descent) methods

Definition 6.1 *(descent direction)*
Let f as in 6.1, $x,\, d \in \mathbb{R}^n$. d is a descent direction of f in x iff

$$\exists \alpha_0 : f(x + \alpha d) < f(x)\, \forall 0 < \alpha < \alpha_0.$$

Lemma 6.2 *(gradient descent)*
Let f as in 6.1, $x,\, d \in \mathbb{R}^n$. If $\nabla f(x) \cdot d < 0$ (scalar product) then d is a descent direction of f in x. This condition is not necessary.

Proof: Let

$$g(t) := g(x + t\, d),\, g : \mathbb{R} \mapsto \mathbb{R} \Rightarrow g'(t) = \nabla g(x + td) \cdot d.$$

Since ∇f is continuous,

$$\nabla f(x) \cdot d < 0 \Rightarrow \exists \alpha_0 :\ \nabla f(x + t\, d) \cdot d < 0\, \forall\, 0 < t < \alpha_0.$$

Then for $0 < \alpha < \alpha_0$

$$f(x + \alpha d) - f(x) = g(\alpha) - g(0) = \int_0^\alpha g'(t)\,dt = \int_0^\alpha \underbrace{\nabla f(t) \cdot d}_{<0}\,dt < 0.$$

For the remark, let $f(x) = -x^2$ and $x = 0$. □

Definition 6.3 *(line search algorithm)*
The general structure of a line search algorithm is:

- *Choose an initial starting point $x^{(0)}$ and let $k = 0$.*
  - *Choose a descent direction $d^{(k)}$ of f in $x^{(k)}$.*
  - *Choose a step length $\alpha^{(k)} > 0$ such that $f(x^{(k)} + \alpha^{(k)} d^{(k)}) < f(x^{(k)})$.*
  - *Let $x^{(k+1)} := x^{(k)} + \alpha^{(k)} d^{(k)}$.*
  - *Let $k := k + 1$ and repeat until some convergence criterion is satisfied.*

Example 6.4 *(Gradient descent for linear equations)*
Let $A \in \mathbb{R}^{n\times n}$ positive definite, $b \in \mathbb{R}^n$. Then the solution $\overline{x}$ of $Ax = b$ is the solution of the unconstrained problem

$$\min_{x\in\mathbb{R}^n} f(x),\ f : \mathbb{R}^n \mapsto \mathbb{R}^n,\ f(x) := \frac{1}{2}(x, Ax) - (b, x).$$

Proof: We have

$$\nabla f(x) = Ax - b,\ \nabla^2 f(x) = A$$

which implies that $\overline{x}$ is the unique minimum of f using 2.4 and 2.3.

We define the gradient descent algorithm for linear equations by setting

$$x^{(0)} := 0,\ d^{(k)} := r^{(k)} := b - Ax^{(k)}$$

which implies that

$$d^{(k)} \nabla f(x^{(k)}) = -||r^{(k)}||_2^2 < 0$$

unless $x^{(k)}$ is already a solution.

We select $\alpha^{(k)}$ such that $x^{(k+1)}$ minimizes

$$g : \mathbb{R} \mapsto \mathbb{R},\ g(\alpha) := f(x^{(k)}+\alpha d^{(k)}) = \frac{1}{2}(A(x^{(k)}+\alpha d^{(k)}), x^{(k)}+\alpha d^{(k)}) - (b, x^{(k)}+\alpha d^{(k)}).$$

We then have

$$0 = g'(\alpha^{(k)}) = (Ax^{(k)}, d^{(k)}) + \alpha^{(k)}(d^{(k)}, Ad^{(k)}) - (b, d^{(k)})$$

or

$$\alpha^{(k)} = \frac{(r^{(k)}, d^{(k)})}{(d^{(k)}, Ad^{(k)})}.$$

Of course, not every descent method converges to a solution of 6.1. Simply take

$$f(x) := x^2,\, x^{(0)} = 3,\, d^{(k)} = -1,\, \alpha^{(k)} = \frac{1}{2^k}$$

which satisfies all conditions, but $x^{(k)}$ converges to 1.

We derive general convergence theorems for descent methods and start with some definitions.

Definition 6.5 *(stationary point, sublevel set)*
Let f as in 6.1.

1. *$x \in \mathbb{R}^n$ is a stationary point of f iff $\nabla f(x) = 0$.*
2. *Let*

$$N(f, y) = \{x :\; f(x) \leq y\}.$$

   N is called (sub) level set of f at level y.

Note that for 6.3, $f^{(k)} \in N(f, f(x^{(0)}))$, so we can always restrict our considerations to this subset of $\mathbb{R}^n$.

Corollary 6.6 *(existence of minimal solutions)*
Let f as in 6.1 and $N(f, y)$ compact for $y = x^{(0)} \in \mathbb{R}^n$. Then 6.1 has a solution. Every solution is a stationary point of f.

Proof: Since f is continuous, f attains its minimum on $N(f, y)$. □

In the following, we will always assume that this corollary holds, i.e. that a solution to 6.1 exists.

Lemma 6.7 *(convergence of values of linesearch methods)*
Let f as in 6.6 and $x^{(k)}$ defined by 6.3. Then $f(x^{(k)})$ converges towards its infimum. In particular, $f(x^{(k+1)}) - f(x^{(k)}) \to 0$.

Proof: $f(x^{(k)})$ is monotonous and bounded. □

We want to derive conditions such that $\nabla f(x^{(k)}) \to 0$.

First, note that our derivation of the descent algorithm was based on

$$f(x^{(k)} + \alpha^{(k)} d^{(k)}) - f(x^{(k)}) \sim \alpha^{(k)} \nabla f(x^{(k)}) \cdot d^{(k)}.$$

We want this to hold up to a constant $c_1 > 0$, meaning

$$f(x^{(k)} + \alpha^{(k)} d^{(k)}) - f(x^{(k)}) \leq c_1 \alpha^{(k)} \nabla f(x^{(k)}) \cdot d^{(k)} < 0. \tag{6.2}$$

Since the left hand side converges to zero, we have

$$\alpha^{(k)} \nabla f(x^{(k)}) \cdot d^{(k)} \to 0.$$

If $\nabla f(x^{(k)}) \to 0$, then

$$\frac{(\nabla f(x^{(k)}) \cdot d^{(k)})^2}{||d^{(k)}||^2} \to 0. \tag{6.3}$$

Therefore, we require for some constant $c_2 > 0$

$$\alpha^{(k)} \geq -c_2 \frac{\nabla f(x^{(k)}) \cdot d^{(k)}}{||d^{(k)}||^2}. \tag{6.4}$$

Altogether, with $c = c_1 c_2$ we have

$$f(x^{(k+1)}) \leq f(x^{(k)}) - c \frac{(\nabla f(x^{(k)}) \cdot d^{(k)})^2}{||d^{(k)}||^2}. \tag{6.5}$$

If 6.3 satisfies this condition, the choice of step length is called efficient.

Theorem 6.8 *(existence of efficient step lengths)*
Let f' Lipschitz–continuous on $N(f, x^{(0)})$(or f twice differentiable), $N(f, x^{(0)})$ compact . Then an efficient step length choice exists in 6.3.

Proof: Exercises. □

Now for the selection of $d^{(k)}$. In order to derive convergence from 6.3, the gradient and the descent direction should not become orthogonal, so for some constant $c_3 > 0$

$$-\nabla f(x^{(k)}) \cdot d^{(k)} \geq c_3 ||\nabla f(x^{(k)})|| \, ||d^{(k)}||.$$

This choice is called gradient–based.

Theorem 6.9 *(convergence of gradient–based, efficient line search algorithms)*
Let f as in 6.6 and $x^{(k)}$ from a gradient–based, efficient line search algorithm. Then every accumulation point of $x^{(k)}$ is a stationary point. If $N(f, f(x^{(0)}))$ is compact, the sequence has at least one accumulation point.

Proof: Let $\overline{x}$ an accumulation point of $x^{(k)}$. Wlog assume $x^{(k)} \to \overline{x}$. From the preliminaries, we have that

$$0 \geq -c_3^2 c ||\nabla f(x^{(k)})||^2 \geq -c \left(\frac{\nabla f(x^{(k)}) \cdot d^{(k)}}{||d^{(k)}||} \right)^2 \geq f(x^{(k+1)}) - f(x^{(k)}) \to 0.$$

□

6.2 Step Size Control for line search

In example 6.4, we could choose $\alpha^{(k)}$ such that it minimizes the function value in the descent direction. However, this is more or less the only case where this explicit choice is possible. For most cases, we resort to the idea in 6.2.

Definition 6.10 *(Armijo Rule)*
Let $0 < c_1 \leq 1$ fixed. A parameter choice $\alpha^{(k)}$ for 6.3 satisfies the Armijo rule iff

$$f(x^{(k+1)}) \leq f(x^{(k)}) + c_1 \alpha^{(k)} \nabla f(x^{(k)}) \cdot d^{(k)}.$$

In view of 6.4, one would like to choose $\alpha^{(k)}$ as large as possible. A possible strategy is to try β^k for $k = 0, \ldots$ and choose the first value that satisfies the condition.

Generally, the following update procedure is used.

Definition 6.11 *(Armijo Goldstein Algorithm)*
Let $0 < c_1 < 1$, $\gamma > 0$, $0 < \beta_1 \leq \beta_2 < 1$. Then choose $\alpha_0 = 1$ and let $j = 0$.

1. *If*
$$f(x^{(k)} + \alpha_j d^{(k)}) \leq f(x^{(k)}) + c_1 \alpha_j \nabla f(x^{(k)}) \cdot d$$
let $\alpha^{(k)} = \alpha_j$ and stop.
2. *Choose $\alpha_{j+1} \in [\beta_1 \alpha_j, \beta_2 \alpha_j]$.*
3. *Let $j := j + 1$ and repeat.*

We now examine the case where the descent direction is given by a linear transform of the gradient.

Lemma 6.12 *(linear transforms for Armijo)*
Let f as in 6.6 and ∇f Lipschitz–continuous (or in C^2) with Lipschitz constant L. In 6.3, let $x = x^{(k)}$ and $d = d^{(k)} = -M \nabla f(x)$ for $M \in \mathbb{R}^{n \times n}$ symmetric positive definite. Further assume $\nabla f(x) \neq 0$.

Let $\lambda_{\max} = ||M||_2$ the largest eigenvalue of M and $\lambda_{\min} = ||M^{-1}||_2$ the smallest eigenvalue. Then for the condition number κ_2 we have

$$\kappa_2(M) = \kappa_2(M^{-1}) = \frac{\lambda_{\max}}{\lambda_{\min}}.$$

Then the Armijo condition is satisfied for $\alpha^{(k)} = \alpha$ if

$$0 < \alpha \leq \frac{2(1 - c_1)}{L \lambda_{\max} \kappa_2(M)}.$$

Proof: The eigenvalue relations are easily proved using an orthonormal basis of eigenvectors of M, which exists since M is s.p.d.
We have

$$f(x+\alpha d) - f(x) = \int_0^1 \nabla f(x+\tau\alpha d)\cdot(\alpha d)\, d\tau.$$

Since ∇f is Lipschitz continuous

$$f(x+\alpha d) = f(x) + \alpha\nabla f(x)\cdot d + \alpha\int_0^1 \underbrace{(\nabla f(x+\tau\alpha d) - \nabla f(x))}_{||\cdot||\leq L\tau\,\alpha\,||d||}\cdot d\, d\tau$$

and thus with Cauchy–Schwarz

$$f(x+\alpha d) \leq f(x) + \alpha\nabla f(x)\cdot d + L\alpha^2||d||^2\frac{1}{2}.$$

We have

$$||Mz||^2 \leq ||M||^2||z||^2 = \lambda_{\max}^2||z||^2$$

and

$$||z||^2 = ||M^{-1/2}M^{1/2}z||^2 \leq ||M^{-1/2}||^2 z^t M z = \frac{1}{\lambda_{\min}} z^t M z.$$

Plugging this into the definition of d, we have

$$\begin{aligned}||d||^2 &= ||M\nabla f(x)||^2\\ &\leq \lambda_{\max}^2||\nabla f(x)||_2^2\\ &\leq \frac{\lambda_{\max}^2}{\lambda_{\min}}\nabla f(x)\cdot\underbrace{M\nabla f(x)}_{=-d}.\end{aligned}$$

So all in all,

$$f(x+\alpha d) \leq f(x) + \alpha(1 - \frac{1}{2}L\alpha\kappa_2(M)\lambda_{\max})\nabla f(x)\cdot d.$$

So the Armijo condition is satisfied if

$$(1 - \frac{1}{2}L\alpha\kappa_2(M)\lambda_{\max}) \leq c_1$$

or

$$\alpha \leq \frac{2(1-c_1)}{L\kappa_2(M)\lambda_{\max}}.$$

□

Corollary 6.13 *Let everything as in 6.12, except*

$$d^{(k)} = -M^{(k)}\nabla f(x^{(k)})$$

for $M^{(k)} \in \mathbb{R}^{n\times n}$ positive definite, and assume that the common bounds

$$\lambda_{\max} \geq \lambda_{\max}^{(k)} \geq \lambda_{\min}^{(k)} \geq \lambda_{\min} > 0$$

(with naming as in 6.12) hold. Then the computed step sizes in 6.11 satisfy

$$\alpha^{(k)} \geq \overline{\alpha} := \beta_1 \frac{2(1-c_1)}{L\kappa\lambda_{\max}}$$

where $\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}$. The maximum number of steps taken in 6.11 is

$$\log_{\beta_2}(\frac{2(1-c_1)}{L\kappa\lambda_{\max}}).$$

Proof: Insert everything into 6.12 and use that, if $\alpha^{(k)} = \alpha_j$,

$$\alpha_j \leq \beta_2^j,\ \alpha_j \geq \beta_1\alpha_{j-1} > \overline{\alpha}.$$

The last inequality holds since the algorithm takes the first α_j that satisfies 6.10. □

Theorem 6.14 *(convergence of Armijo–Goldstein)*
Let everything as in 6.13, in particular let f bounded to below. Then any accumulation point of $x^{(k)}$ is a stationary point of f.

Proof: Basically, this is 6.9. The choice of d is gradient–based since

$$(d^{(k)}, \nabla f(x^{(k)}) = (-M\nabla f(x^{(k)}), \nabla f(x^{(k)})) \leq -\lambda_{\min}||\nabla f(x^{(k)})||^2 \leq -\frac{\lambda_{\min}}{\lambda_{\max}}||\nabla f(x^{(k)})||\,||d||.$$

Explicitly, we have

$$\begin{aligned} f(x^{k+1}) - f(x^{(k)}) &\leq -c_1\alpha^{(k)}\nabla f(x^{(k)}) \cdot M^{(k)}\nabla f(x^{(k)}) \\ &\leq -c_1\overline{\alpha}\lambda_{\min}||\nabla f(x^{(k)}||^2 \\ &\leq -c_1 2\frac{\beta_1(1-c_1)}{L\kappa^2}||\nabla f(x^{(k)})||^2 \leq 0. \end{aligned}$$

Now apply 6.7. □

Of course, the Armijo–rule by itself is not sufficient, since it does not exclude the choice of very small steps (very small α). In fact, the Armijo–rule is always satisfied for a step in an interval $(0, \widetilde{\alpha}]$ with no lower bound. The reason that 6.11 works is that we stop as soon as 6.10 is satisfied, and therefore we got a lower bound.

One might feel tempted to choose β_1 and β_2 very small to minimize the number of iterations taken in 6.11. However, that would result in $\alpha^{(k)}$ underestimating the optimal value that would minimize f along direction $d^{(k)}$. The obvious idea is: Take them small, but then raise the value until some condition is satisfied. This is the idea of the Powell–Wolfe type algorithms.

We start by formulating the Powell–Wolfe condition. The optimal value for α would satisfy $\nabla f(x + \alpha d) \cdot d = 0$, so we require that scalar product to be small.

Definition 6.15 *(Powell–Wolfe condition)*
Let everything as in 6.6. Let $0 < c_1 < c_2 < 1$ and $x, d \in \mathbb{R}^n$, $\nabla f(x) \cdot d < 0$. $\alpha > 0$ satisfies the Powell–Wolfe condition iff

$$f(x + \alpha d) \leq f(x) + c_1\, \alpha \nabla f(x) \cdot d \;\; \textit{(Armijo rule)}$$

and

$$\nabla f(x + \alpha d) \cdot d \geq c_2 \nabla f(x) \cdot d.$$

Again, remember that in the second inequality the right hand side is negative. Also note that defining

$$g(t) := f(x + td)$$

this amounts to requiring

$$g(\alpha) \leq g(0) + c_1 \alpha g'(0),\; g'(\alpha) \geq c_2 g'(0).$$

Typically, the Powell–Wolfe condition will be satisfied for α in an interval $[\widetilde{\beta}, \widetilde{\alpha}]$.

Lemma 6.16 *(existence of Powell–Wolfe steplength)*
In 6.15, let $c_1 < 1/2$. Then $\exists \alpha > 0$ that satisfies the Powell–Wolfe condition.

Proof: Define

$$\Psi(\alpha) := f(x + \alpha d) - f(x) - c_1 \alpha \nabla f(x) \cdot d \Rightarrow \Psi'(\alpha) = (\nabla f(x + \alpha d) - c_1 \nabla f(x)) \cdot d$$

so that Armijo is satisfied if $\Psi(\alpha) < 0$. Since

$$\Psi'(0) = (1 - c_1) \nabla f(x) \cdot d < 0,\; \Psi(0) = 0$$

Ψ is negative in an interval around zero. On the other hand, since f is bounded to below, we have

$$\lim_{\alpha \to \infty} \Psi(\alpha) = +\infty.$$

Since Ψ is continuous

$$\exists \alpha^* : \Psi(t) < 0 \, \forall t \in (0, \alpha^*), \; \Psi(\alpha^*) = 0.$$

Since

$$\Psi'(\alpha^*) = \lim_{t<0} \frac{1}{t}(\Psi(\alpha^* + t) - \Psi(\alpha^*)) \geq 0$$

we finally get

$$\nabla f(x + \alpha^* d) \cdot d \geq c_1 \nabla f(x) \cdot d > c_2 \nabla f(x) \cdot d.$$

□

The corresponding algorithm now is obvious.

Definition 6.17 *(Powell–Wolfe algorithm))*
Let everything as in 6.16. Use 6.11 with $\beta_1 = \beta_2 = \frac{1}{2}$ to find an α_2 that does not satisfy 6.10 and an α_1 that does.

1. *If α_1 satisfies 6.15, set $\alpha^{(k)} := \alpha_1$ and stop.*
2. *Let $\alpha = \frac{1}{2}(\alpha_1 + \alpha_2)$.*
3. $$\alpha =: \begin{cases} \alpha_1, & \text{if } \alpha \text{ satisfies 6.10} \\ \alpha_2, & \text{otherwise}. \end{cases}$$

Lemma 6.18 *(Termination of Powell–Wolfe)*
6.17 terminates in finitely many steps.

Proof: In each step of the algorithm, 6.10 is satisfied for α_1, but not satisfied for α_2, implying

$$\Psi(\alpha_1) < 0, \Psi(\alpha_2) > 0$$

with Ψ from 6.16. Since the distance between the two is halved in each step, both converge towards the same value α^*. Using exactly the same argument as in 6.16, α^* satisfies 6.15. But then it also satisfies it in a small neighborhood, which means that the algorithm terminates for $\alpha^* - \alpha_1$ small enough $\lightning$. □

Theorem 6.19 *(semi–convergence of Powell–Wolfe)*
Let everything as in 6.6, particularly, assume that $N(f, f(x^{(0)}))$ is compact. Assume that $d^{(k)}$ is chosen according to 6.13. Then the same convergence result holds.

Proof: Powell–Wolfe is a variant of 6.11. □

Theorem 6.20 *Let ∇f Lipschitz–continuous with Lipschitz–constant L on $N(f, f(x^{(0)}))$. Then $\exists\, \theta > 0$, independent of d and x, such that*

$$f(x + \alpha d) \geq f(x) - \theta \left(\frac{\nabla f(x) \cdot d}{||d||} \right)^2$$

for all α that satisfy 6.15.

Proof: Since α satisfies 6.15, we have

$$(c_2 - 1)\nabla f(x) \cdot d \leq \underbrace{(\nabla f(x + \alpha d) - \nabla f(x))}_{||\cdot|| \leq L\alpha ||d||} \cdot d \leq L\alpha ||d||^2.$$

Inserting into 6.10

$$\begin{aligned} f(x + \alpha d) &\leq f(x + \alpha c_1 \nabla f(x) \cdot d) \\ &\leq f(x) + \frac{(c_2 - 1)\nabla f(x) \cdot d}{L||d||^2} c_1 \nabla f(x) \cdot d \quad = f(x) - \theta \left(\frac{\nabla f(x) \cdot d}{||d||} \right)^2. \end{aligned}$$

□

6.3 Gradient type methods for linear equations

We prove convergence and error estimates for 6.4 and the conjugate gradient method. Note that for simplicity, we always set $x^{(0)} := 0$.

Satz 6.21 *(Convergence of 6.4)*
Assume everything as in 6.4 and assume that $x^{(k)}$ is chosen accordingly. Particularly,

$$x^{(0)} := 0,\ d^{(k)} := r^{(k)} := b - Ax^{(k)}$$

and

$$\alpha^{(k)} := \frac{(r^{(k)}, d^{(k)})}{(d^{(k)}, Ad^{(k)})}.$$

Let

$$(x, y)_A = (Ax, y),\ ||x||_A^2 = (x, Ax),\ e^{(k)} = x - x^{(k)} \Rightarrow Ae^{(k)} = r^{(k)}.$$

Then we have:

1. *$x^{(k)}$ converges to the solution of $Ax = b$.*
2. *Let $\kappa = k(A) = ||A||_2 \cdot ||A^{-1}||_2$. Then*

$$||e_k||_A \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^k ||e_0||_A \Rightarrow ||e^{(k)}|| \leq \sqrt{\kappa}\left(\frac{\kappa - 1}{\kappa + 1}\right)^k ||e_0||.$$

Proof: Let

$$G(\alpha) := ||x^{(k)}+\alpha r^{(k)}-x||_A^2 = ||-e^{(k)}+\alpha r^{(k)}||_A^2 = ||e^{(k)}||_A^2-2\alpha(\underbrace{Ae^{(k)}}_{=r^{(k)}}, r^k)+\alpha^2 \underbrace{||r^{(k)}||_A^2}_{=(r^{(k)},Ar^{(k)})} .$$

Clearly, $\alpha^{(k)}$ is a minimizer of G. For all $\alpha > 0$, we have

$$G(\alpha) = || - e^{(k)} + \alpha r^{(k)}||_A^2 = ||(I - \alpha A)e^{(k)}||_A^2 \leq ||(I - \alpha A)||_A^2\, ||e^{(k)}||_A^2$$

A is positive definite, so there is a unitary matrix $U \in \mathbb{R}^{n\times n}$ and a diagonal matrix Σ with the eigenvalues on the diagonal such that $A = U^t\Sigma^2 U$. Plugging this in we get

$$\begin{aligned}
||(I - \alpha A)||_A^2 &= \sup \frac{((I - \alpha A)x, Ax)}{(x, Ax)} \\
&= \sup \frac{(\Sigma Ux - \alpha\Sigma^3 Ux, \Sigma Ux)}{(\Sigma Ux, \Sigma Ux)} \\
&= \sup \frac{((I - \alpha\Sigma^2)y, y)}{(y, y)},\ y = \Sigma Ux \\
&= ||I - \alpha\Sigma^2||_2^2 \\
&= \max_j |1 - \alpha\lambda_j|^2,\ \lambda_j \text{ Eigenvalue of } A.
\end{aligned}$$

Therefore, $\forall\, \alpha > 0$

$$\begin{aligned}
||e^{(k+1)}||_A^2 &\leq G(\alpha) \\
&= ||(I - \alpha A)e^{(k)}||_A^2 \\
&\leq ||(I - \alpha A)||_A^2 ||e^{(k)}||_A^2 \\
&\leq \max_j |1 - \alpha\lambda_j|^2 \cdot ||e^{(k)}||_A^2.
\end{aligned}$$

Let $\lambda_{\max}$, $\lambda_{\min}$ the largest and smallest eigenvalue of A, resp. Then

$$\kappa = ||A||_2 \cdot ||A^{-1}||_2 = \frac{\lambda_{\max}}{\lambda_{\min}}.$$

Let

$$\alpha = \frac{2}{\lambda_{\max} + \lambda_{\min}}$$

to wit

$$1 - \alpha\lambda_{\max} = \frac{\lambda_{\max} + \lambda_{\min} - 2\lambda_{\max}}{\lambda_{\max} + \lambda_{\min}} = \frac{\lambda_{\min} - \lambda_{\max}}{\lambda_{\max} + \lambda_{\min}} = -\frac{\kappa - 1}{\kappa + 1}$$

and

$$1 - \alpha\lambda_{\min} = \frac{\lambda_{\max} + \lambda_{\min} - 2\lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} = \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} = \frac{\kappa - 1}{\kappa + 1}.$$

All in all, $|1 - \alpha\lambda_j|$ is always in the interval $[-\frac{\kappa-1}{\kappa+1}, \frac{\kappa-1}{\kappa+1}]$ and we have

$$||e^{(k+1)}||_A \leq \frac{\kappa - 1}{\kappa + 1}||e^{(k)}||_A.$$

□

We realize that the condition number is crucial to the convergence speed, over which we have no direct control. However, we are free to rewrite our original equation and try to modify it in such a way that the new matrix has a lower condition number.

Specifically: Assume that H^{-1} is a symmetric positive definite matrix with the property that

$$H^{-1} A \sim I.$$

Then H is called a preconditioner for A. Many techniques for constructing it are available, the simplest one setting $H^{-1} := D^{-1}$ where A is the main diagonal of A.

Then setting $y = H^{1/2}x$, we have

$$Ax = b \Longleftrightarrow H^{-1/2}AH^{-1/2}y = H^{-1/2}b. \tag{6.6}$$

Since $H^{-1/2}AH^{-1/2}$ is positive definite and

$$H^{-1/2}AH^{-1/2} = H^{1/2}H^{-1}AH^{-1/2} \sim I \Rightarrow \kappa_2(H^{-1/2}AH^{-1/2}) \sim 1$$

we can use 6.4 to solve 6.6, and expect a favorable convergence speed. For an extensive overview over preconditioning algorithms see (Saad).

It might be not completely obvious why one would opt to chose a descent direction $d^{(k)}$ other than $-\nabla f(x^{(k)})$, since this choice guarantees maximal decay of the minimization function at $x^{(k)}$. However, this view is based on one single evaluation of ∇f. Assume that $k \geq 1$, then $f(x^{(l)})$ and $\nabla f(x^{(l)})$ are available for $l \leq k$. Then the question arises: Can we use these evaluations to come up with a better choice of the descent direction?

The most popular of these algorithms is the conjugate gradient (CG) algorithm. We give an outline of the algorithm.

Definition 6.22 *(conjugate vectors)*
Let $A \in \mathbb{R}^{n\times n}$ positive definite. For $x, y \in \mathbb{R}^n$, let

$$(x, y)_A := (x, Ay).$$

x, y are (A–) conjugate if they are orthogonal in the A–scalar product,

$$0 = (x, y)_A = (x, Ay).$$

Theorem 6.23 *(gradient descent for conjugate vectors)*
Assume that in 6.4 the descent directions are A–conjugate:

$$(d^{(k)}, d^{(l)})_A = 0 \,\forall k \neq l,\, d^{(k)} \neq 0.$$

Everything else is chosen as in 6.4,

$$x^{(0)} := 0,\, r^{(k)} := b - Ax^{(k)}, \alpha^{(k)} := \frac{(r^{(k)}, d^{(k)})}{(d^{(k)}, Ad^{(k)})}.$$

Define

$$\mathcal{K}_k := \textit{span}\{d^{(0)}, \dots, d^{(k-1)}\}.$$

Let x the solution of $Ax = b$. Then

1. *$x^{(k)}$ is the solution of*
$$\min_{z\in\mathcal{K}_k} ||x - z||_A.$$
2. *$x^{(n)} = x$.*
3. *$x^{(k)}$ is the solution of*
$$\min_{z\in\mathcal{K}_k} f(z).$$

Proof: Note that $u^{(k)} = d^{(k)}/||d^{(k)}||$ is an ONB wrt $(,)_A$, and $x^{(j)} \in \mathcal{K}_j$.

1. The solution is the orthogonal projection of x on $\mathcal{K}_k$, given by
$$\begin{aligned}\sum_{j=0}^{k-1}(x, u^{(j)})_A u^{(j)} &= \sum_{j=0}^{k-1}(\underbrace{Ax}_{=b} - Ax^{(j)}, u^{(j)})u^{(j)} \qquad\qquad x^{(j)} \perp_A u_j\\ &= \sum_{j=0}^{k-1}\frac{(r^{(j)}, d^{(j)})}{||d^{(j)}||_A^2}d^{(j)}\\ &= x^{(k)}.\end{aligned}$$

2. From 1. since $d^{(0)} \dots d^{(n-1)}$ is a basis of $\mathbb{R}^n$.
3. Since $x^{(k)}$ is the orthogonal projection of x onto $\mathcal{K}_k$, we have
$$x = x^{(k)} + u,\ u \perp_A \mathcal{K}_k.$$
Let $v \in \mathcal{K}_k$. Then
$$\begin{aligned} f(x^{(k)} + v) &= \frac{1}{2}(x^{(k)} + v, A(x^{(k)} + v)) - (b, x^{(k)} + v) \\ &= f(x^{(k)}) + \frac{1}{2}(v, Av) + (v, Ax^{(k)} - \underbrace{b}_{=Ax=A(x^{(k)}+u)}) \\ &= f(x^{(k)}) + \frac{1}{2}(v, Av) \geq f(x^{(k)}). \end{aligned}$$

□

Note that this implies that any gradient descent method for 6.1 with A–conjugate directions is guaranteed to terminate after n iterations with the correct solution, implying that it is not really an iterative, but a deterministic algorithm.

However, for a typical large system of equations, it is unrealistic to perform n iterations, the actual number is much smaller.

In the conjugate gradient algorithm, the $d^{(k)}$ are chosen as $r^{(k)}$ as in 6.4, but orthogonalized using Gram–Schmidt.

Definition 6.24 *(Conjugate Gradient algorithm, first form)*
Let everything as in 6.4. Apply Gram–Schmidt orthogonalization in the A–scalar product to
$$\{r^{(0)}, r^{(1)}, \dots, \} \to \{d^{(0)}, d^{(1)}, \dots, \}$$
and perform 6.4.

This procedure will guarantee that the descent directions are conjugate, as long as $r^{(k)} \neq 0$, but then $Ax = b$ anyway. Also note that
$$d^{(k)} \in \text{span}\{r^{(0)}, \dots, r^{(k)}\} = \mathcal{K}_{k+1}.$$

However, applying Gram–Schmidt becomes computationally expensive, when k is large. We simplify this procedure. We start with

Lemma 6.25 *Let everything as in 6.24.*

1. *$\mathcal{K}_k$ is the Krylov–subspace*
$$\mathcal{K}_k = \{p(A)\, r^{(0)} : p \in \mathcal{P}_{k-1}\}.$$

2. $r^{(k)} \perp \mathcal{K}_k$.
3. $r^{(k)} \perp_A \mathcal{K}_{k-1}$.

Proof:

1. By induction. We have
   $r^{(k+1)} = b - Ax^{(k+1)} = b - Ax^{(k)} - \alpha^{(k)} Ad^{(k)} = r^{(k)} - \alpha^{(k)} Ad^{(k)} = p(A)\, r^{(0)} - q(A) r^{(0)}$
   where $p \in \mathcal{P}_{k-1}$ and $q \in \mathcal{P}_k$.
2. Since $x^{(k)}$ minimizes $||z - x||_A$ for $z \in \mathcal{K}_k$, we have
   $$(r^{(k)}, z) = (Ax - Ax^{(k)}, z) = (x - x^{(k)}, z)_A = 0\, \forall z \in \mathcal{K}_k.$$
3. Let $y \in \mathcal{K}_{k-1}$. Then
   $$(r^{(k)}, y)_A = (r^{(k)}, Ay) = 0.$$

□

For Gram–Schmidt, since $d^{(i)} \in \mathcal{K}_{i+1}$, this gives

$$\begin{aligned} d^{(k+1)} &= r^{(k+1)} - \sum_{i=0}^{k} \frac{(d^{(i)}, r^{(k+1)})_A}{||d^{(i)}||_A^2} d^{(i)} \\ &= r^{(k+1)} - \frac{(d^{(k)}, r^{(k+1)})_A}{||d^{(k)}||_A^2} d^{(k)} \\ &= r^{(k+1)} - \frac{(Ad^{(k)}, r^{(k+1)})}{(d^{(k)}, Ad^{(k)})} d^{(k)}. \end{aligned}$$

We end up with the second form of the CG algorithm. Note that this is the full definition, in the context of the lecture we set $x^{(0)} = 0$ for simplicity.

Definition 6.26 *(Conjugate Gradient algorithm, second form)*
Let $A \in \mathbb{R}^{n \times n}$ positiv definite, $b,\ x^{(0)} \in \mathbb{R}^n$. Then the conjugate gradient algorithm is defined by

$$\begin{aligned} r^{(0)} &= b - Ax^{(0)},\ d^{(0)} = r^{(0)} \\ \alpha^{(k)} &= \frac{(d^{(k)}, r^{(k)})}{(d^{(k)}, Ad^{(k)})} \\ x^{(k+1)} &= x^{(k)} + \alpha^{(k)} d^{(k)} \\ r^{(k+1)} &= b - Ax^{(k+1)} = r^{(k)} - \alpha^{(k)} Ad^{(k)} \\ \beta^{(k+1)} &= \frac{(d^{(k)}, r^{(k+1)})_A}{(d^{(k)}, d^{(k)})_A} = \frac{(d^{(k)}, Ar^{(k+1)})}{(d^{(k)}, Ad^{(k)})} \\ d^{(k+1)} &= r^{(k+1)} - \beta^{(k+1)} d^{(k)}. \end{aligned}$$

The algorithm terminates when $r^{(k)} = d^{(k)} = 0$ with $x = x^{(k)}$ the solution of $Ax = b$.

While it is nice that the algorithm is terminating with the correct solution (provided no rounding errors occur), this feature is never used. Rather, the iteration is stopped with $k << n$. In this case, we again need an error estimate.

Lemma 6.27 *Let everything as in 6.26. Let $\lambda_1 \geq \ldots \geq \lambda_n > 0$ the Eigenvalues of A with the corresponding ONB of eigenvectors $\{v_i\}$. Further, let $p \in \mathcal{P}_k$ with*

$$p(0) = 1 \textit{ and } |p(\lambda_j)| \leq r \, \forall \, j.$$

Then the iterates $x^{(k)}$ of the cg algorithm satisfy

$$||e_A^{(k)}|| \leq r||e^{(0)}||_A.$$

Proof: We have

$$p(z) - 1 = zq(z), \; q \in \mathcal{P}_{k-1}.$$

Let $z = q(A)r^{(0)} \in \mathcal{K}_k$. We have

$$\begin{aligned} x - z &= A^{-1}b - q(A)AA^{-1}r^{(0)} \\ &= (I + Aq(A))e^{(0)} \\ &= p(A)e^{(0)}. \end{aligned}$$

Then

$$\begin{aligned} ||x - x^{(k)}||_A^2 &\leq ||x - z||_A^2 \\ &= ||p(A)e^{(0)}||_A^2 \\ &= ||\sum_{i=1}^n \alpha_i p(\lambda_i) v_i||_A^2 \\ &= \left(\sum_{i=1}^n \alpha_i p(\lambda_i) v_i, \sum_{i=1}^n \alpha_i p(\lambda_i) \lambda_i v_i \right) \\ &= \sum_{i=1}^n \lambda_i (p(\lambda_i)\alpha_i)^2 \\ &\leq \sum_{i=1}^n \lambda_i r^2 \alpha_i^2 \\ &= r^2 \left(\sum_{i=1}^n \alpha_i v_i, A \sum_{i=1}^n \alpha_i v_i \right) \\ &= r^2 ||\overline{x} - x_0||_A^2 = r^2 ||e_0||_A^2. \end{aligned}$$

□

This Lemma implies that bounds for the approximation error of x^k in $\mathcal{P}_{k-1}$ yield error bounds for the cg algorithm.

Theorem 6.28 *(error estimate for the cg algorithm)*
For the conjugate gradient algorithm 6.26 we have

$$||e^{(k)}||_A \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k ||e^{(0)}||_A,\ ||e^{(k)}||_2 \leq 2\sqrt{\kappa}\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k ||e^{(0)}||_2.$$

Proof: In 6.27, use the Tschebyscheff polynomials $T_k(x) = \cos(k \arccos x)$ (exercises). □

6.4 Newton's method

In this chapter, we will use Newton's method, a well–known algorithm for the solution of nonlinear equations, for minimization. We start by recalling the definition of convergence speed (order of convergence).

Definition 6.29 *(order of convergence)*
Let $x^{(k)}$ a sequence in $\mathbb{R}^n$, $x^{(k)} \to \overline{x}$, and let the convergence be monotonous in the sense that

$$||x^{(k+1)} - \overline{x}|| \leq ||x^{(k+1)-\overline{x}}||,\ x^{(k)} \neq \overline{x}.$$

1. *The convergence speed is linear iff $\exists \gamma \in (0,1),\ K > 0$:*

$$||x^{(k+1)} - \overline{x}|| \leq \gamma ||x^{(k)} - \overline{x}|| \forall k > K.$$

2. *The convergence speed is superlinear iff*

$$\frac{||x^{(k+1)} - \overline{x}||}{||x^{(k)} - \overline{x}||} \to 0.$$

3. *The convergence speed is of the order q (quadratic for $q = 2$), iff $\exists C > 0$:*

$$||x^{(k+1)} - \overline{x}|| \leq C ||x^{(k)} - \overline{x}||.$$

4. *Let $y^{(k)}$ a sequence that also converges to $\overline{x}$, and*

$$||y^{(k)} - \overline{x}|| \leq ||x^{(k)} - \overline{x}||.$$

If the convergence of $x^{(k)}$ is linear/superlinear/of the order q, then $y^{(k)}$ converges r-linear/r-superlinear/r-of the order q.

Remark 6.30 *(order of convergence)*

1. *Part 4 looks strange in the definition. However, it is needed. Think of the geometric sequence*
$$x = (1, q, q^2, q^3, q^4 \ldots)$$
The convergence to 0 *is obviously linear. Now take the sequence*
$$y = (1, 0, q^2, 0, q^4, 0 \ldots).$$
According to the definition, this is not linear, but intuitively, it should have the same convergence speed as x*. In the literature, very often, there is no exact distinction between the definitions of linear and r-linear etc.*

2. *The gradient descent and the conjugate gradient descent method converge linearly.*

6.4.1 Newton's optimization method

For Newton's optimization method, we look for the solution of the unconstrained problem
$$\min_{x\in\mathbb{R}^n} f(x),\ f \in C^2.$$

Definition 6.31 *(Newton model problem) In the following, we assume that* $\overline{x}$ *is a minimizer that satisfies the sufficient conditions*
$$\nabla f(\overline{x}) = 0,\ \nabla^2 f(\overline{x}) \textit{ positive definite}$$
and that $\nabla^2 f(x)$ *is Lipschitz continuous in a neighborhood* U *of* $\overline{x}$*, i.e.*
$$\exists\, \gamma > 0 : ||\nabla^2 f(x) - \nabla^2 f(y)|| \leq \gamma ||x - y||\, \forall x, y \in U.$$

The classical Newton method for the solution of nonlinear equations is defined as follows.

Theorem 6.32 *(Newton's method for finding the zero of nonlinear equations)*
Let $F : \mathbb{R}^n \mapsto \mathbb{R}^n$ *differentiable. Let* $F(\overline{x}) = 0$*,* $F'(\overline{x})$ *invertible. Let* $x^{(k)}$ *a sequence where* $x^{(0)} \in \mathbb{R}^n$*,* $L^{(k)}(x^{(k+1)}) = 0$ *where* $L^{(k)}$ *is the linearization of* F *at* $x^{(k)}$ *i.e.*
$$L^{(k)}(x) = F(x^{(k)}) + F'(x^{(k)})(x - x^k).$$

If F' *is Lipschitz continuous in a neighborhood of* $\overline{x}$*, then there exists an* $\epsilon > 0$ *such that Newton's method for nonlinear equations is well (and uniquely) defined and converges to* $\overline{x}$ *of the order 2 if*
$$||x^{(0)} - \overline{x}|| \leq \epsilon.$$

In particular, we have

$$x^{(k+1)} = x^{(k)} - F'(x^{(k)})^{-1}F(x^{(k)}).$$

Proof: Numerical Analysis. □

This can be interpreted in the following way: In Newton's method for nonlinear equations, the function F is replaced by a linear model L, and we take each next iteration as the zero of that linear model.

Note that Newton's method converges fast, but requires $x^{(0)}$ to be (very) close to $\overline{x}$, which is unknown. Also note that formally, we are writing down the inverse here, but of course the update is determined by solving an appropriate linear system of equations.

Starting from here, we can easily derive Newton's method for unconstrained nonlinear minimization.

Theorem 6.33 *(Newton's method for unconstrained minimization)*
Let everything as in 6.31.
Let $F(x) = \nabla f(x)$. Then

$$F : \mathbb{R}^n \mapsto \mathbb{R}^n,\ F(\overline{x}) = 0,\ F'(\overline{x}) \textit{ positive definite, in particular invertible}.$$

From 6.32 we have: $\exists\, \epsilon > 0$, such that Newton's method for finding a zero of F is well defined and converges quadratically to the minimizer $\overline{x}$ of f provided $||x^{(0)} - \overline{x}|| \leq \epsilon$.
In particular, we have

$$x^{(k+1)} = x^{(k)} - \nabla^2 f(x^{(k)})^{-1}\nabla f(x^{(k)}).$$

There is a second way of motivating this equation. Let $m^{(k)}(x)$ the local truncated (quadratic) Taylor series of f at $x^{(k)}$, then

$$m^{(k)}(x) = f(x^{(k)}) + \nabla f(x^{(k)}) \cdot (x - x^{(k)}) + \frac{1}{2}(x - x^{(k)})^t \nabla^2 f(x^{(k)})(x - x^{(k)}).$$

In each step, rather than minimizing f, we minimize the quadratic model, and use the minimizer as the next step. Using 6.4, the minimum is given by

$$x^{(k+1)} = x^{(k)} - \nabla^2 f(x^{(k)})^{-1}\nabla f(x^{(k)})$$

and we again get Newton's method.

Note that setting $M^{(k)} = \nabla^2 f(x^{(k)})^{-1}$ this is a descent method in the sense of 6.13 with steplength 1, the only thing we still need to prove are common bounds on the Hessian.

However, again, this is only true in a small neighborhood of $\overline{x}$ where the Hessian is guaranteed not to deviate too much from the Hessian at $\overline{x}$ which is positive definite. So this statement is only locally true.

However, we have 6.9. If we ensure that in each step for the update direction $d^{(k)}$

$$-\nabla f(x^{(k)}) \cdot d^{(k)} \geq \lambda ||\nabla f(x^{(k)})|| \, ||d^{(k)}|| \qquad (*)$$

is satisfied, and the steplength is chosen by the Armijo rule, then the algorithm will converge.

Theorem 6.34 *(Globally convergent Newton algorithm)*
In 6.33 let

$$y^{(k)} = -\nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)}).$$

If $y^{(k)}$ satisfies $(*)$, then use $d^{(k)} = y^{(k)}$ as the descent direction, otherwise use $d^{(k)} = -\nabla f(x^{(k)})$. Determine the steplength $t^{(k)}$ using the Armijo algorithm, and let

$$x^{(k+1)} = x^{(k)} + t^{(k)} d^{(k)}.$$

Then every accumulation point of $x^{(k)}$ is a minimizer (independent of $x^{(0)}$).

Proof: 6.9. □

Note that in this form, the proof is pretty useless. After all, it does not really improve on 6.9, because it could be that the algorithm never chooses Newton directions or steplengths and converges only linearly. However, it can be shown that if $x^{(k)}$ is close enough to $\overline{x}$ for some k, the algorithm will converge quadratically.

6.4.2 Quasi–Newton methods

The major disadvantage of Newton's method is that it is computationally very expensive. To just compute the Hessian $H^{(k)} \sim \nabla^2 f(x^{(k)})$ from finite differences, f has to be evaluated n^2 times compared to n times for the gradient. Additionally, even if the Hessian is available, an $n \times n$ linear system of equations has to be solved. So there is much interest in simplifying the Newton step.

To overcome the first problem, only a very coarse approximation of the Hessian is used. It turns out that Newton's method is convergent as long as $||H^{(k)}|| \leq C$ for some $C > 0$. Even choosing $H^{(k)}$ fixed will do! (Exercises) However, typically one loses the quadratic convergence speed for these simple ideas.

For the second problem, we could take into account that the update direction $d^{(k)}$ is not exactly pointing to $\overline{x}$ anyway. So we might have the idea that the linear equation does not have to be solved exactly, but an iterative algorithm with few (fixed number of) steps will do.

Note that the second case can be rephrased as a special case of the first case.

To formalize the idea, assume that using some variant of Newton's method we have already computed $x^{(k+1)}$. To continue, we define the approximation $H^{(k+1)}$ of the Hessian at $x^{(k+1)}$ using what we have to compute anyway, that is the approximation $H^{(k)}$ for the Hessian at $x^{(k)}$, $\nabla f(x^{(k+1)})$, and $f(x^{(k)}$. Formally, for some function φ, we have

$$H^{(k+1)} = \varphi(H^{(k)}, x^{(k+1)}, x^{(k)}, \nabla f(x^{(k+1)}), \nabla f(x^{(k)})).$$

1. $H^{(k)}$ should be symmetric, possibly s.p.d.
2. φ should be computationally cheap.
3. Again, from the fundamental theorem of calculus, we have

$$\nabla f(x^{(k+1)}) - \nabla f(x^{(k)}) = \underbrace{\int_0^1 \nabla^2 f(x^{(k)} + t(x^{(k+1)} - x^{(k)}))\, dt}_{\sim H^{(k+1)} \text{ if } x^{(k)} \sim x^{(k+1)}} (x^{(k+1)} - x^{(k)}).$$

   This motivates the quasi–Newton condition

$$H^{(k+1)}(x^{(k+1)} - x^{(k)}) = \nabla f(x^{(k+1)}) - \nabla f(x^{(k)}).$$

   Obviously, this is an equation in $\mathbb{R}^n$, but the Hessian is in $\mathbb{R}^{n\times n}$, so φ is far from uniquely defined.

4. Choose $||H^{(k+1)} - H^{(k)}|| \to 0$. This is motivated as follows:

$$||(H^{(k)} - \nabla^2 f(x^{(k)}))\, d^{(k)}|| \le ||(H^{(k)} - H^{(k+1)})d^{(k)}|| + ||H^{(k+1)}d^{(k)} - \nabla^2 f(x^{(k)})||.$$

   Using the quasi–Newton condition, the second term reads

$$||\nabla f(x^{(k+1)} - \nabla f(x^{(k)}) - \nabla^2 f(x^{(k)})d^{(k)}|| = o(||d^{(k)}||)$$

   with Taylor expansion on ∇f. Thus, if $||H^{(k)} - H^{(k+1)}|| \to 0$, both terms are $o(||d^{(k)}||)$.

Since φ is not uniquely defined, there is a plethora of options for the update. The simplest one is the rank 1–update. Here, we choose φ such that $H^{(k+1)} - H^{(k)}$ has rank 1 or

$$H^{(k+1)} = H^{(k)} + v^{(k)}{v^{(k)}}^t.$$

For the quasi–Newton condition, we get

$$H^{(k+1)}d^{(k)} = \underbrace{\nabla f(x^{(k+1)}) - \nabla f(x^{(k)})}_{=:y^{(k)}}.$$

Plugging in the definition of $H^{(k+1)}$ we get

$$y^{(k)} = H^{(k)}d^{(k)} + v^{(k)}{v^{(k)}}^t d^{(k)} = H^{(k)}d^{(k)} + ({v^{(k)}}^t d^{(k)})v^{(k)} \qquad (*)$$

or

$$v^{(k)} = \frac{1}{\underbrace{{v^{(k)}}^t d^{(k)}}_{=:\lambda}}(y^{(k)} - H^{(k)}d^{(k)})$$

where λ is a scalar unknown. Inserting into $(*)$ determines λ up to its sign, and this defines φ.

More sophisticated rules consider a rank 2 update of the form

$$H^{(k+1)} = H^{(k)} + u^{(k)}{u^{(k)}}^t + v^{(k)}{v^{(k)}}^t.$$

Since now we have two degrees of freedom, the update is not uniquely defined, and there are many possible choices. One of the more popular is BFGS (Broyden–Fletcher–Goldfarb–Shanno), which chooses

$$H^{(k+1)} = H^{(k)} + \frac{y^{(k)}{y^{(k)}}^t}{{y^{(k)}}^t d^{(k)}} - \frac{H^{(k)}d^{(k)}(H^{(k)}d^{(k)})^t}{({y^{(k)}}^t d^{(k)})^2} y^{(k)}{y^{(k)}}^t.$$

It can be shown that BFGS is quadratically convergent in a small neighborhood of $\overline{x}$.

6.4.3 Trust Region Methods

Another problem of Newton is its local convergence. To overcome this, trust region methods have been developed, which are covered in the exercises. We give a very short outline of the idea.

1. In our methods, we replace $f(x)$ by a model function (approximation). For Newton, we choose
$$f(x^{(k)}+s) = m^{(k)}(s) = f(x^{(k)}) + g^{(k)^t} \cdot s + \frac{1}{2} s^t H^{(k)} s$$
where $g^{(k)}$ is the gradient and $H^{(k)}$ is the Hessian at $x^{(k)}$.

2. Typically, the model / the approximation will only be reliable in a neighborhood of $x^{(k)}$, so if
$$||s|| \leq \delta^{(k)}.$$

3. When we choose the normal Newton update $d^{(k)}$, and $||d^{(k)}|| > \delta^{(k)}$, this makes no sense (since we are using the model where it is not reliable).

4. Rather than taking the global minimum of $m^{(k)}$, we compute
$$d^{(k)} = \min_{||s|| \leq \delta^{(k)}} m^{(k)}(s).$$
For the quadratic model, this can be computed explicitly (exercises).

5. In each step, $\delta^{(k)}$ is updated. To this end, we compare the predicted reduction in the model
$$pred(s) := m^{(k)}(0) - m^{(k)}(s)$$
and the actual reduction
$$ared(s) := f(x^{(k)}) - f(x^{(k+1)}).$$
If
$$\rho := \frac{ared(s)}{pred(s)} \sim 1$$
then the model is reliable and the current step is accepted. Also, we figure that the reliability radius might be enlarged and choose $\delta^{(k+1)} = 2\delta^{(k)}$, $x^{(k+1)} = x^{(k)} + d^{(k)}$.
If this is not the case, then we throw away the current step, setting $x^{(k+1)} = x^{(k)}$, and reducing the reliability radius by setting $\delta^{(k+1)} = \delta^{(k)}/2$.

Chapter 7

Numerical optimization with constraints

We wish to derive numerical methods for the general problem 0.1 including constraints, i.e.

$$\min f(x) : c_I(x) \leq 0,\, c_E(x) = 0$$

with definitions as in 0.1.

7.1 Penalty Methods

The idea here is to reuse the methods for unconstrained optimization, but add a penalty term $\Psi(x)$ to the minimization function f that is zero for x feasible and large otherwise, i.e.

$$F(x,\gamma) = f(x) + \gamma\Psi(x),\; \Psi(x) = \begin{cases} 0 & x \text{ feasible} \\ > 0 & x \text{ infeasible} \end{cases},\; \gamma > 0.$$

The penalty method then solves for fixed γ the unconstrained minimization problem

$$\min F(x,\gamma)$$

and accepts the solution as an approximation for a minimal point $\overline{x}$.

A typical choice is the quadratic penalty function

$$\Psi_2(x) = \frac{1}{2}(||c_E(x)||^2 + ||c_I(x)^+||^2) \text{ where } z^+ = \max(z,0).$$

Note that if c_I and c_E are differentiable, then also Ψ_2 is and

$$\nabla\Psi_2 = c_E'(x)^t\, c_E(x) + c_I'(x)^t c_I(x)^+$$

where $'$ denotes the Jacobian (remember the definition of the Jacobian in 2.1). Obviously, for x feasible we have

$$F(x,\gamma) = f(x),\ \nabla F(x,\gamma) = \nabla f(x).$$

Formally, for $\gamma \to \infty$, we regain our original problem. However, when γ is large, also the derivative of F is large, leading to numerical instabilities. We ignore this for the moment and prove that penalty methods have some expected properties.

Theorem 7.1 *(properties of the penalty method)*
Let f, c_I, c_E continuous. Let $\gamma^{(k)}$ a strictly increasing sequence with limit ∞, $\gamma^{(0)} > 0$. Assume that the unconstrained minimization problem for F has a solution $x^{(k)}$ for all $\gamma^{(k)}$, and that the constrained problem has a global solution. Then

1. $F(x^{(k)}, \gamma^{(k)})$ *is increasing.*
2. $\Psi(x^{(k)})$ *is decreasing.*
3. $f(x^{(k)})$ *is increasing.*
4. $\Psi(x^{(k)}) \to 0.$
5. *Each accumulation point of $x^{(k)}$ is a global solution of the unconstrained problem.*

Note that we need the additional requirement that the global solution for the unconstrained problems exists, even if the constrained problem has a global solution. Simply assume that the feasible set is compact, and that $f(x)$ is exponentially decreasing, then a solution for the constrained problem exists, but none of the penalty problems is solvable (using quadratic penalty terms). To ensure solvability, one might cut off the function $f(x)$ for $||x|| > x_0$ or apply a damping strictly increasing function to f, e.g. $\min \log f(x)$ instead of $\min f(x)$.

For the proofs, we simply insert the defining minimization property of $x^{(k)}$.

Proof:

1. $$F(x^{(k)}, \gamma^{(k)}) \le F(x^{(k+1)}, \gamma^{(k)}) \le F(x^{(k+1)}, \gamma^{(k+1)}).$$

2. From the optimality condition for $x^{(k)}$, we have

$$\begin{aligned} F(x^{(k)}, \gamma^{(k)}) &\le F(x^{(k+1)}, \gamma^{(k)}) \\ F(x^{(k+1)}, \gamma^{(k+1)}) &\le F(x^{(k)}, \gamma^{(k+1)}). \end{aligned}$$

Adding these we get

$$\underbrace{(\gamma^{(k)} - \gamma^{(k+1)})}_{<0} \Psi(x^{(k)}) \leq (\gamma^{(k)} - \gamma^{(k+1)})\Psi(x^{(k+1)})$$

which implies

$$\Psi(x^{(k+1)}) \leq \Psi(x^{(k)}).$$

3. We have

$$\begin{aligned} 0 &\leq F(x^{(k+1)}, \gamma^{(k)}) - F(x^{(k)}, \gamma^{(k)}) \\ &= f(x^{(k+1)}) - f(x^{(k)}) + \gamma^{(k)} \underbrace{(\Psi(x^{(k+1)}) - \Psi(x^{(k)}))}_{\leq 0\,(2.)} \\ &\leq f(x^{(k+1)}) - f(x^{(k)}). \end{aligned}$$

4. Let $\widehat{x}$ a (feasible) solution to the constrained problem. Then

$$F(x^{(k)}, \gamma^{(k)}) \leq F(\widehat{x}, \gamma^{(k)}) = f(\widehat{x})$$

implying

$$f(\widehat{x}) \geq f(x^{(k)}) + \gamma^{(k)}\Psi(x^{(k)}) \geq f(x^{(0)}) + \gamma^{(k)}\Psi(x^{(k)}).$$

That means that $\gamma^{(k)}\Psi(x^{(k)})$ is bounded, so $\Psi(x^{(k)}) \to 0$.

5. WLOG Let $x^{(k)} \to \overline{x}$. From $(4.)$ and continuity of c_I, c_E we have that $\overline{x}$ is feasible.
Let $x \in \mathbb{R}^n$ feasible. Then

$$\begin{aligned} f(x^{(k)}) &\leq F(x^{(k)}, \gamma^{(k)}) \\ &\leq F(x, \gamma^{(k)}) \\ &= f(x). \end{aligned}$$

Take the limit of this inequality, then since f is continuous

$$f(\overline{x}) \leq f(x).$$

□

We already noted that for Ψ_2 the minimization function F is differentiable, provided f, c_I, c_E are differentiable. For the following remark, we recall the definitions of the Lagrange function. For the full problem 0.1 the Lagrange function is defined as

$$L(x, \mu, \lambda) := f(x) + \mu^t c_I(x) + \lambda^t c_E(x),\ \mu \in \mathbb{R}^m,\ \lambda \in \mathbb{R}^p,\ \mu \geq 0.$$

Then for Ψ_2

$$\begin{aligned} 0 &= \nabla F(x^{(k)}, \gamma^{(k)}) \\ &= \nabla f(x^{(k)}) + \gamma^{(k)} c_E'(x^{(k)})^t \, c_E(x^{(k)}) + \gamma^{(k)} c_I'(x^{(k)})^t c_I(x^{(k)})^+ \\ &= \nabla f(x^{(k)}) + c_E'(x^{(k)})^t \, \underbrace{(\gamma^{(k)} \, c_E(x^{(k)}))}_{=:\lambda^{(k)}} + c_I'(x^{(k)})^t \, \underbrace{(\gamma^{(k)} c_I(x^{(k)})^+)}_{=:\mu^{(k)}} \\ &= \nabla_x L(x^{(k)}, \mu^{(k)}, \lambda^{(k)}) \end{aligned}$$

which is one part of the saddle point definition (and which is reminiscent of Lagrange multipliers).

In fact, we have

Theorem 7.2 *(Penalty Methods and the KKT conditions)*
Let everything as in 7.1 and $\Psi = \Psi_2$.

1. *If $(\overline{x}, \overline{\lambda}, \overline{\mu})$ is an accumulation point of $(x^{(k)}, \lambda^{(k)}, \mu^{(k)})$, then $(\overline{x}, \overline{\mu}, \overline{\lambda})$ satisfies the Kuhn–Tucker–conditions (2.16)*

$$\nabla_x L(\overline{x}, \overline{\mu}, \overline{\lambda}) = 0, \; \overline{x} \textit{ feasible}, \; \overline{\mu} \, c_I(\overline{x}) = 0.$$

2. *If $\overline{x}$ is an accumulation point of $x^{(k)}$ and $\overline{x}$ is regular (i.e. the gradients of the active restrictions are linear independent), then there are $\overline{\mu}, \overline{\lambda}$ such that $(\overline{x}, \overline{\mu}, \overline{\lambda})$ is an accumulation point of $(x^{(k)}, \mu^{(k)}, \lambda^{(k)})$.*

Proof:

1. WLOG let $x^{(k)} \to \overline{x}, \; \mu^{(k)} \to \overline{\mu}, \; \lambda^{(k)} \to \overline{\lambda}$. From 7.1 we have that $\overline{x}$ is a solution, and from the remark we have

$$0 = \nabla_x L(x^{(k)}, \mu^{(k)}, \lambda^{(k)}) \to \nabla_x L(\overline{x}, \overline{\mu}, \overline{\lambda}).$$

   Also

$$0 \leq \gamma^{(k)} c_I(x^{(k)})^+ = \mu^{(k)} \to \overline{\mu}.$$

   Since $\overline{x}$ is feasible, $c_I(x) \leq 0$. Assume $c_I(\overline{x})_j < 0$.
   Then for k larger than some index K, $c_I(x^{(k)})_j < 0$, so $c_I(x^{(k)})_j^+ = 0$ for $k > K$, so $\overline{\mu}_j = 0$. So either $c_I(\overline{x})_j = 0$, or $\overline{\mu}_j = 0$.

2. The only difference to part 1 is that here we do not have the convergence of the corresponding subsequence.
   WLOG assume that all inequality restrictions are inactive (otherwise move them to equality restrictions). Then for $k > K$ as before, $c_I(x^{(k)}) < 0$ and $\mu^{(k)} \to 0 =: \mu$. Let

$$\overline{A} = c_E'(\overline{x})^t = (\nabla c_E(\overline{x})), \; A^{(k)} = c_E'(x^{(k)})^t.$$

Since $\overline{x}$ is regular, $\overline{A}$ has full rank, thus $\overline{A}^t\overline{A}$ is invertible. Also, starting at some index, the determinant of $A^{k^t}A^{(k)}$ does not vanish ($A^{(k)} \to \overline{A}$ and the determinant is a continuous function), so for $k > K$ $A^{k^t}A^{(k)}$ is invertible. Therefore

$$0 = A^{(k)^t}\nabla_x\widetilde{L}(x^{(k)}, \lambda^{(k)}) = A^{(k)^t}\nabla f(x^{(k)}) + A^{(k)^t}A^{(k)}\lambda^{(k)}$$

where $\widetilde{L}(x, \lambda) = L(x, 0, \lambda)$. Since $A^{(k)^t}A^{(k)}$ is invertible, this can be solved for $\lambda^{(k)}$, and

$$\lambda^{(k)} \to -(\overline{A}^t\overline{A})^{-1}\overline{A}^t\nabla f(\overline{x}).$$

□

In a way, these last two theorems are a great result. Although the idea of penalty methods is so simple, yet they have more or less all properties you could wish for: Convergence to the minimum, and even convergence of the dual parameters. Nevertheless, the initial statement holds: For $\gamma^{(k)} \to \infty$ the problems are badly scaled and numerically unstable for $\gamma^{(k)}$ large.

So we would like to circumvent this. It comes as a big surprise that large $\gamma^{(k)}$ is not always needed. In fact, some penalty functions Ψ are exact, meaning they produce the correct result $\overline{x}$ for finite $\gamma^{(k)}$.

Definition 7.3 *(exact penalty methods)*
A penalty function Ψ is called exact for a minimization problem, if a (local) minimizer of the restricted problem is a (local) minimizer of the penalized function F for some $\gamma > 0$.

An example for an exact penalty function for problems with strong duality is the L_1−penalty term

$$\Psi_1(x) = ||c_I(x)^+||_1 + ||c_E(x)||_1.$$

We prove the exactness for convex problems only.

Theorem 7.4 *(Ψ_1 is exact)*
Let f, c_I convex, c_E linear, and $(\overline{x}, \overline{\mu}, \overline{\lambda})$ satisfy the KKT conditions. Then $\overline{x}$ is a solution of (the restricted problem) 0.1, and it is a global minimum of

$$F(x, \gamma) := f(x) + \gamma\Psi_1(x)$$

for $\gamma \geq \max(|\lambda_k|, \mu_i)$.

Proof: $\overline{x}$ is a (feasible) solution to the global problem due to 3.9.

Further, we have for any $x \in \mathbb{R}^n$

$$\begin{aligned}
F(\overline{x}, \gamma) &= f(\overline{x}) \\
&= f(\overline{x}) + \overline{\mu}^t c_I(\overline{x}) + \overline{\lambda}^t c_E(\overline{x}) && \text{complementarity} \\
&= L(\overline{x}, \overline{\mu}, \overline{\lambda}) \\
&\leq L(x, \overline{\mu}, \overline{\lambda}) && \text{saddle point property} \\
&\leq f(x) + \overline{\mu}^t\, c_I(x)^+ + |\overline{\lambda}|^t\, |c_E(x)| && |\cdot| \text{ componentwise} \\
&\leq f(x) + \gamma \Psi_1(x) \\
&= F(x, \gamma).
\end{aligned}$$

□

Now this is really great. However, it comes with a caveat: F will not be differentiable, since $||\cdot||_1$ is not differentiable.

Remark 7.5
Let Ψ exact für γ_0. Then Ψ is exact for $\gamma \geq \gamma_0$.

Proof: Let $\overline{x}$ a minimizer of the restricted problem, thus $\overline{x}$ feasible. From this and the exactness for γ_0, we have for all $x \in \mathbb{R}^n$

$$f(\overline{x}) = F(\overline{x}, \gamma_0) \leq F(x, \gamma_0)$$

and thus

$$F(\overline{x}, \gamma) = f(\overline{x}) \leq F(x, \gamma_0) \leq F(x, \gamma).$$

□

Corollary 7.6 *(exact penalty functions are not differentiable)*
Let Ψ differentiable and exact for γ_0 and a (local) minimum $\overline{x}$. Then $\nabla f(\overline{x}) = 0$.

This corollary says: If a penalty function is exact, then it is either not differentiable, or the constraints do not play a role, $\overline{x}$ is a local minimizer of f even without the constraints.

Proof: From the remark, we have that Ψ is exact for $\gamma > \gamma_0$. Thus

$$\nabla_x F(\overline{x}, \gamma) = 0 = \nabla_x F(\overline{x}, \gamma_0) \Rightarrow (\gamma - \gamma_0)\nabla\Psi(\overline{x}) = 0 \Rightarrow \nabla\Psi(\overline{x}) = 0$$

and again since

$$0 = \nabla_x F(\overline{x}, \gamma)$$

we have $\nabla f(\overline{x}) = 0$. □

However, differentiability is definitely a desirable property. In the following, we will slightly change the definition of the penalty functions. *We first consider only equality constraints, that is $m = 0$ in 0.1.*

Remark 7.7 *(non-exactness of Ψ_2)*
Let $\Psi = \Psi_2$, $\overline{x}$ a solution for 0.1 with equality constraints only, $\overline{\lambda}$ the corresponding Lagrange multiplier, and $\overline{\lambda}_k \neq 0$. Then feasibility of $\overline{x}$ can only be achieved by letting $\gamma^{(k)} \to \infty$.

Proof: From the proof of 7.2, we have that in this case

$$\lambda^{(k)} = \gamma^{(k)} c_E(x^{(k)}) \to \overline{\lambda}.$$

□

So we need a new idea. We start by defining a slightly modified Lagrange function. Remember that for equality–only constraints, we have

$$L(x, \lambda) = f(x) + \lambda^t c_E(x).$$

We define L_A, the augmented Lagrangian, as the Lagrange function for the penalized objective function $F(x, \gamma)$. Remember that since $\Psi(x) = 0$ for x feasible, the penalized problem with restrictions is equivalent to the original problem with restrictions.

Definition 7.8 *(Augmented Lagrangian)*
Consider 0.1 with equality–only constraints. Then the augmented Lagrangian is defined as

$$L_A(x, \lambda, \gamma) = f(x) + \gamma \frac{1}{2} ||c_E(x)||^2 + \lambda^t c_E(x) = L(x, \lambda) + \gamma \Psi_2(x).$$

Remember $c_E(x) = (h_1(x), \ldots, h_p(x))^t$, and set $\Psi = \Psi_2$.

Lemma 7.9 *(Gradient of Augmented Lagrangian)*
We have

$$\nabla_x L_A(x, \lambda, \gamma) = \nabla f(x) + Dc_E(x)^t(\lambda + \gamma c_E(x)) = \nabla f(x) + \sum_{i=1}^{m} (\lambda_i + \gamma h_i(x)) \nabla h_i(x).$$

Idea of the method: In the algorithm, we determine the primal solution x^* and its Lagrange multiplier λ^*. So

1. Input: Approximation $x^{(0)}$ for the minimal point, approximation $\lambda^{(0)}$ for the Lagrange multiplier, $\gamma^{(0)}$ as starting point for the penalty term, reliability parameter τ_0.
2. Compute an approximation $x^{(k+1)}$ to the unrestricted minimization problem for $L_A(\cdot, \lambda^{(k)}, \gamma^{(k)})$ such that
$$||\nabla_x L_A(x^{(k+1)}, \lambda^{(k)}, \gamma^{(k)})|| \leq \tau_k.$$
See 2.12.
3. Update $\lambda^{(k+1)}$ based on $\lambda^{(k)}$.
4. Choose
$$\gamma^{(k+1)} \geq \gamma^{(k)},\ \tau^{(k+1)} \leq \tau^{(k)}.$$
5. Repeat from 2.

Idea for the update of $\lambda^{(k)}$:

From the minimization property, we have

$$0 = \nabla_x L_A(x^{(k+1)}, \lambda^{(k)}, \gamma) = \nabla f(x^{(k+1)}) + \sum_{i=1}^{m} (\mu_i + \gamma h_i(x)) \nabla h_i(x^{(k+1)}).$$

For the true minimizer and its Lagrange multiplicator, we have

$$0 = \nabla_x L(x^*, \mu^*) = \nabla f(x^*) + \sum_j \mu_i^* \nabla h_i(x^*).$$

Assume that $x^{(k+1)} \sim x^*$, $\mu^{(k+1)} \sim \mu^*$, then

$$\lambda^{(k+1)} = \lambda^{(k)} + \gamma c_E(x^{(k+1)})$$

is a decent choice.

Theorem 7.10 *(almost exact property of the augmented Lagrangian)*
Let x^* a (local) solution of 0.1 with equality constraints and regular. Assume that the second order sufficient conditions of 2.14 are satisfied, and that λ^* is the corresponding Lagrange multiplier. Then there is a $\gamma^* > 0$ such that for $\gamma \geq \gamma^*$ $\overline{x}$ is a strict (local) minimizer of $L_A(x, \lambda^*, \gamma)$.

Note that this is close to exactness, but not quite, since we require knowledge of the true Lagrange multiplier.

Reminder: The second order sufficient conditions require that

$$c_E(x^*) = 0,\ \nabla f(x^*) + \lambda^t \nabla c_E(x^*) = 0$$

and that

$$\operatorname{Hess}_x L(x^*, \lambda) = \operatorname{Hess} f(x) + \sum_k \lambda_k \operatorname{Hess} h_k(x^*)$$

is positive definite on the Kernel of $Dc_E(x^*)^t$.

Proof: We show that the second order sufficient conditions 2.14 are satisfied. We have

$$\begin{aligned} \nabla_x L_A(x^*, \lambda^*, \gamma) &= \nabla f(x^*) + \sum_j (\lambda_j^* + \gamma h_j(x^*)) \nabla h_j(x^*) \\ &= \nabla f(x^*) + \lambda^t c_E(x^*) && x^* \text{ is feasible} \\ &= L(x^*, \lambda) = 0 && 2.11. \end{aligned}$$

Now we prove positive definiteness of the Hessian. From the formula for the gradient, taking one more derivative, we see that

$$\operatorname{Hess} L_A(x, \lambda, \gamma) = \operatorname{Hess} L(x, \lambda) + \gamma \sum_j (h_j(x) \operatorname{Hess} h_j(x) + \nabla h_j(x) \nabla h_j(x)^t).$$

Let $A := Dc_E(x^*) = (\nabla h_1(x^*), \dots, \nabla h_p(x^*))^t$. Then A has full rank $p \leq n$ since x^* is regular, and since again x^* is feasible

$$\operatorname{Hess}_x L_A(x^*, \lambda^*, \gamma) = \operatorname{Hess} L(x^*, \lambda^*) + \gamma A^t A.$$

Assume that $\operatorname{Hess} L_A(x^*, \lambda^*, \gamma)$ is not positive definite for all $\gamma > \gamma_0$. Then for all k there exists a $w^{(k)} \in \mathbb{R}^n$, $||w^{(k)}|| = 1$, such that

$$\begin{aligned} 0 &\geq {w^{(k)}}^t \operatorname{Hess}_x L_A(x^*, \lambda^*, \gamma) w^{(k)} \\ &= {w^{(k)}}^t \operatorname{Hess}_x L(x^*, \lambda^*) w^{(k)} + k||Aw^{(k)}||^2. \end{aligned}$$

The unit ball is compact, so $w^{(k)}$ has a convergent subseries. WLOG let $w^{(k)} \to w$, $||w|| = 1$. Further, we have

$$||Aw^{(k)}||^2 \leq -\frac{1}{k} {w^{(k)}}^t \operatorname{Hess}_x L(x^*, \lambda^*) w^{(k)} \to 0$$

so $||Aw|| = 0$ or w is in the kernel of $Dc_E(x^*)^t$.
Finally,

$$0 \geq -k||Aw^{(k)}|| \geq {w^{(k)}}^t \operatorname{Hess}_x L(x^*, \lambda^*) w^{(k)} \to w^t \operatorname{Hess}_x L(x^*, \lambda^*) w$$

which is a contradiction to the assumption that $\operatorname{Hess}_x L(x^*, \lambda^*)$ is positive definite on the kernel of $Dc_E(x^*)^t$ (and $w \neq 0$). □

Idea for inequality constraints: Convert the full problem 0.1 to equality–only by introducing the slack variables s_i, writing $g_i(x) \leq 0$ as $g_i(x) + s_i^2 = 0$. This is a reformulation of the original problem in the new variables (x, s) with equality constraints only.

The corresponding augmented Lagrangian is

$$L_A(x, s, \mu, \lambda, \gamma) = f(x) + \lambda^t h(x) + \frac{1}{2}\gamma ||h(x)||^2 + \sum_i \mu_i (g_i(x) + s_i^2) + \gamma \frac{1}{2}(g_i(x) + s_i^2)^2.$$

Our algorithm is exactly the same, except now we must update not only x and λ, but also s and μ. With respect to the update on s: L_A is a quadratic function in s_i^2, and the minimum can be computed explicitly. It turns out that the minimizer is given by

$$s_i^* = (\max(0, -\frac{\mu_i}{\gamma} + g_i(x)))^{\frac{1}{2}}.$$

7.2 Quadratic Programming

In this section, let f quadratic and c_E, c_I linear, i.e. 0.1 reads

$$\min_x f(x) := \frac{1}{2} x^t Q x + c^t x \text{ with } c_E(x) := Bx = \beta,\ c_I(x) := Ax \leq \alpha$$

where $Q \in \mathbb{R}^{n \times n}$, $c \in \mathbb{R}^n$, $B \in \mathbb{R}^{p \times n}$, $\beta \in \mathbb{R}^p$, $B \in \mathbb{R}^{m \times n}$, $\alpha \in \mathbb{R}^m$.

Let x^* a (feasible) solution. The Karush–Kuhn–Tucker conditions 2.16 guarantee the existence of $\lambda^* \in \mathbb{R}^p$, $\mu^* \in \mathbb{R}^m$ such that

$$Qx^* + c + B^t \lambda^* + A^t \mu^* = 0,\ \mu^* \geq 0,\ \mu^* (Ax^* - \alpha) = 0,\ Ax^* \leq \alpha,\ Ax^* = \beta.$$

7.2.1 Equality Constraints

We start by considering equality restrictions only ($m = 0$). Then the KKT condtions reduce to

$$\begin{pmatrix} Q & B^t \\ B & 0 \end{pmatrix} \begin{pmatrix} x^* \\ \lambda^* \end{pmatrix} = \begin{pmatrix} -c \\ \beta \end{pmatrix}.$$

Assuming that Q is positive definite and $\overline{x}$ is regular (which implies that the rows of B are linearly independent) this system of equations has a unique solution (the minimal point $\overline{x}$ and its corresponding Lagrange multiplier).

7.2.2 Active Set Strategy for inequality constraints

Now we allow both equality and inequality constraints. To motivate the algorithm, assume that x^* is known. Let I the index set

$$I := \{i : (Ax^*)_i = \alpha_i\}.$$

Let

$$A_I := (A_{i,k} : i \in I),\, \alpha_I := (\alpha_i),\, \mu_I^* = (\mu_i^*) : i \in I.$$

Then

$$Bx^* = \beta,\, A_I x^* = \alpha_I.$$

Since $\mu_k^* = 0$ for $k \notin I$, we have

$$\begin{pmatrix} Q & A_I^t & B^t \\ A_I & & \\ B & & \end{pmatrix} \begin{pmatrix} x^* \\ \mu_I^* \\ \lambda^* \end{pmatrix} = \begin{pmatrix} -c \\ \alpha_I \\ \beta \end{pmatrix} \tag{7.1}$$

Again, assuming that Q is positive definite and that x^* is regular, we have that this system of equations has a unique solution.

Note that given any solution of 7.1 and defining $\mu_k^* = 0$ for $k \notin I$, 2.16 is satisfied iff $\mu_i^* \geq 0$ for all $i \in I$.

This motivates the following strategy:

1. Cycle through all index subsets I of $(1, \dots, m)$.
2. Compute the corresponding solution of 7.1.
3. Check if $\mu_I^* \geq 0$. If yes, x^* is a potential (local) minimum.

This is inefficient, so we follow a strategy in the spirit of the simplex algorithm.

1. Input: a feasible (!) $x^{(0)}$, an index set $I = I^{(0)}$ s.t. $A_I x^* = \alpha_I$.
2. Compute x^*, μ^* via 7.1 for the index set $I^{(k)}$.
3. Check if x^* is feasible. Since $A_I x^* = \alpha_I$, $Bx^* = \beta$, this is only possible if $(Ax^*)_k > \alpha_k$ for some $l \notin I^{(k)}$.
4. If x^* is feasible, continue at 6.

5. Add one of the indices that blocks x from feasibility to $I^{(k)}$, i.e. set

$$t^* := \max\{t : x^{(k)}+t(x^*-x^{(k)}) \text{ is feasible}\} = \min\{\frac{\alpha_l - (Ax^{(k)})_l}{(A(x^* - x^{(k)}))_l} : l \notin I^{(k)}\},\ x^{(k+1)} := x^{(k)}+t^*($$

   Then $\exists l \notin I^{(k)} :\ x_l^{(k+1)} = 0$. Set

$$I^{(k+1)} := I^{(k)} \cup \{l\}$$

   and continue at 2.

6. Check if $\mu \geq 0$. If yes, then a KKT point has been found.
7. If not, remove one of the blocking indices from $I^{(k)}$, i.e.

$$l = \arg\min_l \mu_l^*,\ I^{(k+1)} := I^{(k)} \setminus \{l\},\ x^{(k+1)} := x^{(k)}.$$

This algorithm is not guaranteed to terminate or converge. However, at least it is guaranteed that $x^* - x^{(k)}$ is a descent direction if Q is s.p.d.

7.3 Gradient Projection Methods

Assume that $x^{(k)}$ with a descent direction $d^{(k)}$ is given for a restricted minimization problem. Since

$$x^{(k+1)} := x^{(k)} + td^{(k)}$$

might lead away from the feasible set, the simple idea is to project this update onto the feasible set.

In the following, we always assume that the feasible set $\mathcal{F}$ is convex. Then the projection onto the feasible set is given by 4.7:

$$P_{\mathcal{F}} :\ \mathbb{R}^n \mapsto \mathcal{F},\ P_{\mathcal{F}}x := \arg\min_{y\in\mathcal{F}} ||y - x||_2^2.$$

Lemma 7.11 *Assume that $\mathcal{F}$ is convex.*

1. *Let x^* a local minimum of 0.1. Then*

$$\nabla f(x^*) \cdot (x - x^*) \geq 0 \forall\, x \in \mathcal{F}.$$

2. *Let f convex and*

$$\nabla f(x^*) \cdot (x - x^*) \geq 0\, \forall\, x \in \mathcal{F}.$$

   Then x^* is a global minimum.

Proof:

1. Assume
$$\exists x \in \mathcal{F} :\ \nabla f(x^*) \cdot (x - x^*) < 0.$$
Then $d := x - x^*$ is a descent direction, thus $f(x^* + td) < f(x^*)$ for t small enough. Since $\mathcal{F}$ is convex, we have $x^* + td \in \mathcal{F}$ for all $t \in [0,1]$, so x^* is not a local minimum $\lightning$.

2. From 4.4, we have for all $x \in \mathcal{F}$
$$f(x) \geq f(x^*) + \nabla f(x^*) \cdot (x - x^*) \geq f(x^*).$$

□

According to our idea, given an approximation $x^{(k)} \in \mathcal{F}$, we set
$$x^{(k+1)} := P_{\mathcal{F}}(x^{(k)} - t\nabla f(x^{(k)})) = x^{(k)} + d(t),\ d(t) := P_{\mathcal{F}}(x^{(k)} - t\nabla f(x^{(k)})) - x^{(k)}.$$

We show that $d(t)$ is a descent direction.

Theorem 7.12 *Let $x \in \mathcal{F}$, $t > 0$, $\mathcal{F}$ convex. Then*
$$t\, d(t) \cdot \nabla f(x^{(k)}) \leq -||d(t)||^2.$$

Proof: Let $z \in \mathbb{R}^n$. Then $P_{\mathcal{F}}(z)$ is the solution of the constrained problem
$$\min_{x \in \mathcal{F}} g(z) := \frac{1}{2}||x - z||^2.$$

From the lemma, we have for all $y \in \mathcal{F}$
$$(P_{\mathcal{F}}(z) - z) \cdot (y - P_{\mathcal{F}}(z)) = (\nabla g)(P_{\mathcal{F}}(z)) \cdot (y - P_{\mathcal{F}}(z)) \geq 0.$$

Now let $y = x^{(k)}$, $z = x^{(k)} - t\nabla f(x^{(k)})$. Thus
$$(P_{\mathcal{F}}(x^{(k)} - t\nabla f(x^{(k)})) - (x^{(k)} - t\nabla f(x^{(k)})) \cdot (x^{(k)} - P_{\mathcal{F}}(x^{(k)} - t\nabla f(x))) \geq 0.$$

which implies
$$-||d(t)||^2 - t\nabla f(x^{(k)}) \cdot d(t) \geq 0.$$

□

Note that projection methods are typically used in situations where the projection onto the feasible set can be computed very easily, like nonnegativity constraints or box constraints
$$\mathcal{F} = \{x : a \leq x \leq b\}.$$

A very common example is emission tomography, where the resulting distributions of radioactive tracers are nonnegative. The resulting algorithm is POCS (Projection On Convex Sets).

7.4 Sequential Quadratic Programming (SQP)

Again, we motivate the algorithm by considering only equality–constrained problems ($m = 0$). Then the Lagrange function is given by

$$L(x,\lambda) = f(x) + c_E(x)^t\,\lambda$$

and the Karush–Kuhn–Tucker conditons read

$$\begin{aligned} \nabla_x L(x,\lambda) = \nabla f(x) + Dc_E(x)^t\lambda = 0 \\ c_E(x) = 0. \end{aligned}$$

In the quadratic problem, we could solve this equation analytically. If this is not the case, we can view it as a nonlinear problem and solve it using the classical Newton update

$$z^{(k+1)} = z^{(k)} \underbrace{-F'(z^{(k)})^{-1}F(z^{(k)})}_{=:d^{(k)}}$$

where F' denotes the Jacobian of F. Then $d^{(k)}$ satisfies

$$F'(z^{(k)})\,d^{(k)} = -F(z^{(k)}).$$

Let

$$F(x,\lambda) = \begin{pmatrix} \nabla_x L(x,\lambda) \\ c_E(x) \end{pmatrix},\; F'(x,\lambda) = \begin{pmatrix} \mathsf{Hess}_x L(x,\lambda) & Dc_E(x)^t \\ Dc_E(x) & 0 \end{pmatrix}$$

where Dc_E is as in 2.1. Letting $d^{(k)} = (dx^{(k)}, d\lambda^{(k)})^t$ and $H^{(k)} = \mathsf{Hess}_x L(x^{(k)}, \lambda^{(k)})$ we arrive at the update formula

$$\begin{aligned} H^{(k)}dx^{(k)} + Dc_E(x^{(k)})^t d\lambda &= -\nabla_x L(x^{(k)}, \lambda^{(k)}) \\ Dc_E(x^{(k)})dx^{(k)} &= -c_E(x^{(k)}). \end{aligned}$$

Now let $\lambda^* := \lambda^{(k)} + d\lambda$. Then

$$\begin{aligned} H^{(k)}dx^{(k)} + Dc_E(x^{(k)})^t\lambda^* &= -\nabla f(x^{(k)}) \\ Dc_E(x^{(k)})dx^{(k)} &= -c_E(x^{(k)}). \end{aligned}$$

But these are simply the KKT–conditions for the minimization problem

$$\min_{dx^{(k)}} \nabla f(x^{(k)}) \cdot dx^{(k)} + \frac{1}{2} {dx^{(k)}}^t H^{(k)} dx^{(k)}$$

with the equality constraint

$$c_E(x^{(k)}) + Dc_E(x^{(k)})\,dx^{(k)} = 0.$$

This gives rise to the idea of approximating 0.1 using a sequence of quadratic problems

$$\min_{dx^{(k)}} \nabla f(x^{(k)}) \cdot dx^{(k)} + \frac{1}{2} dx^{(k)^t} H^{(k)} dx^{(k)}$$

such that

$$c_E(x^{(k)}) + Dc_E(x)\, dx^{(k)} = 0,\; c_I(x^{(k)}) + Dc_I(x^{(k)})\, dx^{(k)} \leq 0$$

and setting $x^{(k+1)} := x^{(k)} + dx^{(k)}$. This is the basic SQP method.

7.5 Interior point algorithms: The Barrier Method

When solving a constrained minimization problem with active equalities using a penalty method, the solutions of the intermediate minimization problems are typically infeasible. Algorithms that rewrite the original minimization problem as a sequence of minimization problem in such a way that the solution of each intermediate problem is feasible, are called interior point methods. In this section, we consider a very restrictive class of problems only. Our minimization problem 0.1 is convex, and we have strong duality. We remind of some properties.

Throughout this section, we investigate the following problem:

$$\min_x f(x) :\; f_1(x) \ldots f_m(x) \leq 0$$

where

$$Ax = b,\; f, f_k \in C^2(D) \text{ convex },\; A \in \mathbb{R}^{p \times n},\; \text{rank}\, A = p < n.$$

Then this is a convex problem in the sense of 0.1. We assume that Slater's condition 4.10 is satisfied, i.e. there is a feasible point x such that $c_I(x) < 0$ (strictly feasible). Then strong duality holds. Also, we assume that an optimal point x^* exists for the minimization problem and all the problems in the intermediate problems.

So let (x^*, μ^*, λ^*) a primal–dual triple. Then

$$Ax^* = b,\; f_i(x^*) \leq 0,\; \mu_i^* \geq 0,\; \mu_i^* f_i(x^*) = 0$$

and the Lagrange multiplier property

$$0 = \nabla f(x^*) + \sum_{i=1}^m \mu_i^* \nabla f_i(x^*) + A^t \lambda^* = \nabla_x L(x^*, \mu^*, \lambda^*)$$

where

$$L(x, \mu, \lambda) = f(x) + \sum_{i=1}^{m} \mu_i f_i(x) + (Ax - b)^t \lambda$$

holds. Remember our definitions of the primal and dual functions

$$p(x) := \sup_{\mu \geq 0, \lambda} L(x, \mu, \lambda),\ d(\mu, \lambda) := \inf_{x \in D} L(x, \mu, \lambda).$$

The primal/dual problems are given by

$$\min_{x \in D} p(x),\ \max_{\mu \geq 0, \lambda} d(\mu, \lambda)$$

respectively, with optimal values p^* and d^*. Generally, we have $p^* \geq d^*$, and $p^* = d^*$ here due to strong duality.

Our idea ist to again rewrite our minimization problem without the inequalities, so that only equality constraints are left. For barrier methods, we apply penalty functions “the other way round”. Instead of penalizing that an inequality constraint is violated, we penalize that an element gets too close to the border, with the value on the border and the infeasible set set to infinity.

Formally, let

$$\chi^-(u) = \begin{cases} 0 & u \leq 0 \\ \infty & \text{else}. \end{cases}$$

Then the minimiziation problem can be rewritten as

$$\min_x f(x) + \sum_{i=1}^{m} \chi^-(f_i(x)) : Ax = b.$$

For the implementation, we use approximations of $\chi^-(u)$ which are finite for $u < 0$ and infinite for $u \geq 0$. Specifically, we use the log function

$$I(u) := -\log(-u),\ \operatorname{dom} I = \{u : u < 0\}$$

and formally set $I(u) = \infty$ for $u \geq 0$.

Lemma 7.13
I is continuous, differentiable, strictly increasing, and convex on D.

Proof:

$$I'(u) = -\frac{1}{u},\ I''(u) = \frac{1}{u^2} > 0.$$

□

Let

$$\Psi(x) = \sum_{i=1}^{m} I(f_i(x)) = -\sum_{i=1}^{m} \log(-f_i(x)).$$

Ψ is the log barrier function.

Lemma 7.14 Ψ *is convex.*

Proof: We have

$$\nabla\Psi(x) = -\sum_{i=1}^{m} \frac{1}{f_i(x)} \nabla f_i(x)$$

and

$$\text{Hess}\, \psi(x) = \sum_{i=1}^{m} \frac{1}{f_i(x)^2} \nabla f_i(x) \nabla f_i(x)^t - \frac{1}{f_i(x)} \text{Hess}\, f_i(x)$$

which is positive semidefinite for $x \in \mathcal{S}$. □

Then the log barrier problem is given by

$$\min_x f(x) + \frac{1}{t}\Psi(x) \text{ where } Ax = b \ (P_t).$$

Note that this problem is convex. Also note that this reformulation can never be exact if there are active inequalities (since the boundary is not feasible for (P_t)).

In the following, let $x^*(t)$ an optimal point of P_t. Then for every t there is a Lagrange multiplier $\lambda^*(t) \in \mathbb{R}^p$ such that

$$\nabla f(x^*(t)) + \frac{1}{t}\nabla\Psi(x^*(t)) + A^t\lambda^*(t) = 0.$$

Theorem 7.15 *(error estimate for the log barrier problem)*
For a fixed $t > 0$*, let* (x^*, λ^*) *an optimal primal/dual pair for* P_t*. Let*

$$\mu^* \in \mathbb{R}^m, \ \mu_i^* := -\frac{1}{t f_i(x^*)} > 0.$$

Then

$$f(x^*) - d(\mu^*, \lambda^*) = \frac{m}{t}$$

and

$$f(x^*) - p^* \leq \frac{m}{t}.$$

Proof: We have

$$\begin{aligned} 0 &= \nabla f(x^*) + \frac{1}{t}\nabla\Psi(x^*) + A^t\lambda^* \\ &= \nabla f(x^*) + \sum_{i=1}^m \mu_i^* \nabla f_i(x^*) + A^t\lambda^* \\ &= \nabla_x L(x^*, \mu^*, \lambda^*). \end{aligned}$$

Therefore, since L is convex,

$$L(x^*, \mu^*, \lambda^*) = \min_x L(x, \mu^*, \lambda^*).$$

Now

$$\begin{aligned} p^* \geq d(\mu^*, \lambda^*) &= \inf_{x \in D} L(x, \mu^*, \lambda^*) \\ &= L(x^*, \mu^*, \lambda^*) \\ &= f(x^*) - \frac{1}{t}\sum_{i=1}^m \frac{f_i(x^*)}{f_i(x^*)} \\ &= f(x^*) - \frac{m}{t} \end{aligned}$$

□

This inspires the following algorithm.

1. Input: x^* strictly feasible point, $\beta > 1$, $\epsilon > 0$.
2. Solve (P_t) via Newton, using x^* as initial guess.
3. Let $x^* = x^*(t)$, let $t = \beta t$
4. Until $\frac{m}{t} < \epsilon$.

Note that if we use Armijo for stepsize control, since x^* is strictly feasible all iterates of the newton method will be strictly feasible.

To start, we need an initial strictly feasible point x^*. We compute it using a phase I–problem reminiscent of the simplex algorithm. Consider the problem

$$\min_{x,s} s : f_1(x) \dots f_m(x) \leq s, \; Ax = b.$$

For this problem, any (x, s) with $Ax = b$, $s = \max f_i(x) + 1$ is a strictly feasible point. We can use the barrier method to compute a solution to this problem with minimal value s^*.

If $s^* < 0$, then the corresponding x^* is a strictly feasible point.

If $s^* \geq 0$, then no strictly feasible point exists.

Chapter 8

Advanced Examples

In this section, we present two advanced examples without going into details. We give references for all unproven remarks.

8.1 Image and Signal Denoising

Let us assume that a (2D) image or (1D) signal F is measured, the measurement is $g = F + n$ with some noise n. Also assume that F is smooth in some sense (e.g. the function is almost monotonous). n will typically not be smooth, so g is not smooth, and the image/signal will look ugly. How do we remove the non–smooth part?

The solution could be a solution to the problem: Find a function f that is not too far away from g, but is sufficiently smooth. So if Ψ is a penalty functional that is large for functions which are not smooth, we need to solve the problem

$$\min_f ||f - g|| + \alpha\Psi(f)$$

where α is a regularization parameter.

As a measure for smoothness, according to our idea of smoothness, we take the p–norm of the gradient, for the data fidelity the 2–norm. The value of p has a big impact on the result. This can be seen in the following way. Let

$$f_a(x) = \begin{cases} \frac{x}{a} & x \in [0, a] \\ 1 & x \in [a, 1]. \end{cases}.$$

Then (with piecewise differentiation)

$$||f_a'||_2^2 = \int_0^a \frac{1}{a^2}\,dx = \frac{1}{a},\ ||f_a'||_1 = \int_0^a f_a'(x)\,dx = 1$$

which implies that the 1–norm does not care how a function gets from 0 to 1 as long as it is monotonous, while the 2–norm favors straight lines. This implies that edges, sharp variations of the brightness between neighboring pixels, will be washed out and the image or signal is blurred.

So it seems like a good idea to take the 1–norm of the gradient for Ψ. However, since this is not differentiable, we expect to end up with similar problems as for constrained minimization with penalty term in the 1–norm.

Since we can only deal with finite dimensional minimization, all of this will have to be discretized, resulting for signal recovery in the problem

$$\min_{x\in\mathbb{R}^n} \underbrace{\frac{1}{2}||x-g||_2^2}_{=:f(x)} + \underbrace{\alpha||Ax||_1}_{=:g(Ax)}. \tag{8.1}$$

where the matrix A is a discretization of the first derivative (gradient). This was introduced by Rudin, Osher and Fatemi in 1992.

To come up with an algorithm, we need the Fenchel conjugate (and leave out most of the technical details).

Definition 8.1 *(Fenchel conjugate function)*
Let $f : \mathbb{R}^n \mapsto \mathbb{R} \cup \{\infty\}$. Then

$$f^*(s) : \mathbb{R}^n \mapsto \mathbb{R} \cup \{\infty\},\ f^*(s) := \sup_{x\in\mathbb{R}^n} s\cdot x - f(x).$$

This definition has a simple interpretation. In 1D: This is the difference in $\mathbb{R}^2$ of the line through the origin with gradient s. Then $f(x)$ is the max of the difference between the graph of the line and the function.

Example: Let

$$f(x) := \begin{cases} x\log x - x & x \geq 0 \\ \infty & \text{else}. \end{cases}$$

The $\sup$ is assumed where the derivative of the function is zero, so

$$0 = s - f'(x) = s - \log x \Rightarrow x = e^s.$$

Evaluation of the minimization function gives

$$f^*(s) = e^s.$$

Let $f(x) := e^x$. Then

$$f^*(x) = \sup_s s \cdot x - e^x.$$

For $s < 0$, this is ∞ (let $x \to -\infty$). For $s > 0$, the minimum exists, the minimal point is $s = e^x$ or $x = \log s$, giving $f^*(s) = s \log s - s$.

So we see that in this case, $f^{**}(s) = f(s)$, which is in fact true for all differentiable convex functions.

Let $f(s) = \alpha|s|$. Then

$$f^*(s) = \sup_x sx - \alpha|x|.$$

Let $x \geq 0$ and $s \leq \alpha$. Then

$$sx - \alpha|x| = x(s - \alpha) \leq 0.$$

Let $x \leq 0$ and $s \geq -\alpha$. Then

$$sx - \alpha|x| = x(s + \alpha) \leq 0.$$

So if $|s| \leq \alpha$, the term in the $\sup$ is bounded to above by zero. If this is not the case, then the term is unbounded, and we have

$$f^*(s) = \begin{cases} 0 & |s| \leq \alpha \\ \infty & \text{else} \end{cases}.$$

Let $f : \mathbb{R}^n \to \mathbb{R}$, $f(x) := \alpha||x||_1$. Then

$$f^*(s) = \begin{cases} 0 & ||s||_\infty \leq \alpha \\ \infty & \text{else} \end{cases}.$$

Theorem 8.2 *(convexity of the conjugate)*
f^* is convex.

Proof: $f_x(y)$ defined as

$$f_x(y) := y \cdot x - f(x)$$

is affine. So

$$\begin{aligned} f^*(\lambda y + (1-\lambda z)) &= \sup_x f_x(\lambda y + (1-\lambda)z) \\ &= \sup_x(\lambda f_x(y) + (1-\lambda f_x(z))) \\ &\leq \lambda \sup_x f_x(y) + (1-\lambda)\sup_x f(x) \\ &= \lambda f^*(y) + (1-\lambda) f^*(z). \end{aligned}$$

□

The main theorem is given by

Theorem 8.3 *(reformulation of minimization problems using the conjugate function)*
Let σ, $\tau > 0$. $x^* \in \mathbb{R}^n$ is a minimizer for 8.1 iff there is a $p^* \in \mathbb{R}^m$ such that x^* is a solution of

$$\min_y \frac{1}{2}||y - (x^* - \tau A^t p^*)||_2^2 + \tau f(y)$$

and p^* is a solution of

$$\min_q \frac{1}{2}||q - (p^* + \sigma A x^*)||_2^2 + \sigma g^*(q).$$

(see e.g. Clason 2017, lecture notes, Rockafellar, pp 349–).

This inspires the following algorithm:

1. Input: $x^{(0)}$, $p^{(0)}$, τ, σ.
2. Let $x^{(k+1)}$ the solution of the problem

$$\min_y \frac{1}{2}||y - \underbrace{(x^{(k)} - \tau A^t p^{(k)})}_{=:x}||_2^2 + \tau f(y).$$

3. Let $\overline{p}$ the solution of the problem

$$\min_q \frac{1}{2}||q - \underbrace{(p^{(k)} + \sigma A x^{(k)})}_{=:z}||_2^2 + \sigma g^*(q).$$

4. Let

$$p^{(k+1)} = \overline{p} + \theta(\overline{p} - p^{(k)}).$$

At first glance, it looks like we did not win anything – we now even have two minimization problems rather than one.

However, it turns out that the intermediate problems can be solved analytically. In fact, inserting the definitions from 8.1, we have that the minimization function in part 1 is given by

$$\frac{1}{2}||y - x||_2^2 + \frac{1}{2}\tau||y - g||_2^2.$$

The minimum is taken when the gradient is zero, so

$$(y_i - x_i) + \tau(y_i - g_i) = 0$$

or

$$y = \frac{x + \tau g}{1 + \tau} =: \text{prox}_{\tau f}(x)$$

with the prox operator.

The minimization function in the second problem is given by

$$\frac{1}{2}||q - z||_2^2 + \sigma g^*(q).$$

From the introductory remarks we have

$$g^*(q) = \begin{cases} 0 & ||q||_\infty \leq \alpha \\ \infty & \text{else} \end{cases}.$$

Therefore, the second problem reduces to

$$\min_q \frac{1}{2}||q - z||_2^2 \text{ where } ||q||_\infty \leq \alpha$$

and the solution satisfies

$$p_i = \begin{cases} z_i & |z_i| \leq \alpha \\ \operatorname{sgn}(z_i)\,\alpha & \text{else} \end{cases} = Proj_{[-\alpha,\alpha]}(z_i).$$

Both minimization problems can be solved very easily.

Note that this is motivational only – for the convergence proof, see Clason.

```
# # Denoising a 1D signal
import numpy as np
import matplotlib.pyplot as plt
# Generate original and noisy signal
N=256
y=np.zeros(N)
y[N//4:3*N//4]=1
X=np.linspace(-1,1,N)
noise=0.2*np.random.normal(0,1,N)
yn=y+noise
plt.plot(X,y,X,yn)
plt.title('Noisy␣Signal')
plt.legend(['original','noisy']);
# L2 denoising
alpha=0.2
Fyn=np.fft.fft(yn)
Ffilt=np.zeros(N,complex)
for i in range(1,N//2):
    Ffilt[i]=Fyn[i]/(1+alpha*i*i)
    Ffilt[N-i]=Fyn[N-i]/(1+alpha*i*i)
Ffilt[0]=Fyn[0]
filt=np.fft.ifft(Ffilt)
plt.plot(X,filt.real,X,yn,X,y)
plt.title('L2␣denoising')
plt.legend(['filtered','noisy','original'])
# Chambolle-Pock
# This is a shame.
N=256
A=np.zeros([N-1,N])
np.fill_diagonal(A,1)
np.fill_diagonal(A[:,1:],-1)
alpha=10
tau=0.1
sigma=0.01
theta=1
# Poor man's iteration.
x=np.zeros(N)
p=np.zeros(N-1)
for i in range(0,100000):
    z1=x-tau*A.T.dot(p)
    z2=p+sigma*A.dot(x)
    x=(z1+tau*yn)/(1+tau)
    for i in range(0,N-1):
        if (z2[i]>alpha):
            z2[i]=alpha
        if (z2[i]<-alpha):
            z2[i]=-alpha
    p=z2+theta*(z2-p)
plt.plot(X,x,X,yn,X,y)
plt.legend(['filtered','noisy','original'])
```

Listing 8.1: L1 and L2 denoising (denoising.py)

Klicken für den Quellcode von denoising.py

Appendix A

Errata

This is a list of vital changes that were made after the lecture notes were first published in the Learnweb.

A.1 Chapter 1

None yet.

A.2 Chapter 2

- Definition 2.1: There was a transpose missing in the definition of (Df) in part 1.
- Definition 2.1: The definition of the Jacobian was added.
- All of section 2.2: I had written hyperplane instead of hypersurface.
- In theorem 2.18, the second order condition was missing.

A.3 Chapter 3

- Duality moved to new chapter.
- In theorem 3.8, strong duality was used and in the name of the theorem, but it was not in the condition.

A.4 Chapter 4

- In 4.8, there was a catastrophic error in the formulation of the theorem (the proof was correct).
- I unfortunately tend to write wolg instead of wlog. Always read wlog.
- In 4.9, part 3, there was $\min$ instead of $\inf$. $\min$ does no longer make sense because C_2 is no longer compact.
- In 4.6, the proofs would only work for C open. I have rewritten the proofs and dropped the condition of differentiability for 1-3.

A.5 Chapter 5

- In Definition 5.3, linearly independent refers only to the columns corresponding to index elements of $\mathcal{I}$.

A.6 Chapter 6

- Corrected algorithm 6.11, $\sigma \mapsto \alpha$, and clarified return value.
- in example 6.4, the definition of the minimization function was wrong (but the correct definition was used).

A.7 Chapter 7

- Theorem 7.2, part 2: In the proof, inserted missing ${A^{(k)}}^t$.
- Theorem 7.4, statement: Absolute value of λ_k, not μ_k. Was correct in the proof.