Chapter 4
Optimization Theory

> We demand rigidly defined areas of doubt and uncertainty!
>
> *The Hitchhiker's Guide to the Galaxy*
> Douglas Adams

This chapter reviews the basic elements of optimization theory and practice, without going into the fine details of numerical implementation. Many UQ problems involve a notion of 'best fit', in the sense of minimizing some error function, and so it is helpful to establish some terminology for optimization problems. In particular, many of the optimization problems in this book will fall into the simple settings of linear programming and least squares (quadratic programming), with and without constraints.

4.1 Optimization Problems and Terminology

In an optimization problem, the objective is to find the extreme values (either the minimal value, the maximal value, or both) $f(x)$ of a given function f among all x in a given subset of the domain of f, along with the point or points x that realize those extreme values. The general form of a constrained optimization problem is

$$\begin{aligned} \text{extremize: } & f(x) \\ \text{with respect to: } & x \in \mathcal{X} \\ \text{subject to: } & g_i(x) \in E_i \quad \text{for } i = 1, 2, \ldots, \end{aligned}$$

where $\mathcal{X}$ is some set; $f \colon \mathcal{X} \to \mathbb{R} \cup \{\pm\infty\}$ is a function called the *objective function*; and, for each i, $g_i \colon \mathcal{X} \to \mathcal{Y}_i$ is a function and $E_i \subseteq \mathcal{Y}_i$ some subset.

T.J. Sullivan, *Introduction to Uncertainty Quantification*, Texts in Applied Mathematics 63, DOI 10.1007/978-3-319-23395-6_4

The conditions $\{g_i(x) \in E_i \mid i = 1, 2, \dots\}$ are called *constraints*, and a point $x \in \mathcal{X}$ for which all the constraints are satisfied is called *feasible*; the set of feasible points,

$$\{x \in \mathcal{X} \mid g_i(x) \in E_i \text{ for } i = 1, 2, \dots\},$$

is called the *feasible set*. If there are no constraints, so that the problem is a search over all of $\mathcal{X}$, then the problem is said to be *unconstrained*. In the case of a minimization problem, the objective function f is also called the *cost function* or *energy*; for maximization problems, the objective function is also called the *utility function*.

From a purely mathematical point of view, the distinction between constrained and unconstrained optimization is artificial: constrained minimization over $\mathcal{X}$ is the same as unconstrained minimization over the feasible set. However, from a practical standpoint, the difference is huge. Typically, $\mathcal{X}$ is $\mathbb{R}^n$ for some n, or perhaps a simple subset specified using inequalities on one coordinate at a time, such as $[a_1, b_1] \times \cdots \times [a_n, b_n]$; a bona fide non-trivial constraint is one that involves a more complicated function of one coordinate, or two or more coordinates, such as

$$g_1(x) := \cos(x) - \sin(x) > 0$$

or

$$g_2(x_1, x_2, x_3) := x_1 x_2 - x_3 = 0.$$

Definition 4.1. Given $f \colon \mathcal{X} \to \mathbb{R} \cup \{\pm\infty\}$, the *arg min* or *set of global minimizers* of f is defined to be

$$\arg\min_{x \in \mathcal{X}} f(x) := \left\{ x \in \mathcal{X} \,\middle|\, f(x) = \inf_{x' \in \mathcal{X}} f(x') \right\},$$

and the *arg max* or *set of global maximizers* of f is defined to be

$$\arg\max_{x \in \mathcal{X}} f(x) := \left\{ x \in \mathcal{X} \,\middle|\, f(x) = \sup_{x' \in \mathcal{X}} f(x') \right\}.$$

Definition 4.2. For a given constrained or unconstrained optimization problem, a constraint is said to be

(a) *redundant* if it does not change the feasible set, and *non-redundant* or *relevant* otherwise;
(b) *non-binding* if it does not change the extreme value, and *binding* otherwise;
(c) *active* if it is an inequality constraint that holds as an equality at the extremizer, and *inactive* otherwise.

Example 4.3. Consider $f \colon \mathbb{R}^2 \to \mathbb{R}$, $f(x, y) := y$. Suppose that we wish to minimize f over the unbounded w-shaped region

$$W := \{(x, y) \in \mathbb{R}^2 \mid y \geq (x^2 - 1)^2\}.$$

Over W, f takes the minimum value 0 at $(x, y) = (\pm 1, 0)$. Note that the inequality constraint $y \geq (x^2 - 1)^2$ is an active constraint. The additional constraint $y \geq 0$ would be redundant with respect to this feasible set W, and hence also non-binding. The additional constraint $x > 0$ would be non-redundant, but also non-binding, since it excludes the previous minimizer at $(x, y) = (-1, 0)$ but not the one at $(x, y) = (1, 0)$. Similarly, the additional equality constraint $y = (x^2 - 1)^2$ would be non-redundant and non-binding.

The importance of these concepts for UQ lies in the fact that many UQ problems are, in part or in whole, optimization problems: a good example is the calibration of parameters in a model in order to best explain some observed data. Each piece of information about the problem (e.g. a hypothesis about the form of the model, such as a physical law) can be seen as a constraint on that optimization problem. It is easy to imagine that each additional constraint may introduce additional difficulties in computing the parameters of best fit. Therefore, it is natural to want to exclude from consideration those constraints (pieces of information) that are merely complicating the solution process, and not actually determining the optimal parameters, and to have some terminology for describing the various ways in which this can occur.

4.2 Unconstrained Global Optimization

In general, finding a global minimizer of an arbitrary function is *very hard*, especially in high-dimensional settings and without nice features like convexity. Except in very simple settings like linear least squares (Section 4.6), it is necessary to construct an approximate solution, and to do so iteratively; that is, one computes a sequence $(x_n)_{n \in \mathbb{N}}$ in $\mathcal{X}$ such that x_n converges as $n \to \infty$ to an extremizer of the objective function within the feasible set. A simple example of a deterministic iterative method for finding the critical points, and hence extrema, of a smooth function is Newton's method:

Definition 4.4. Let $\mathcal{X}$ be a normed vector space. Given a differentiable function $g \colon \mathcal{X} \to \mathcal{X}$ and an initial state x_0, *Newton's method* for finding a zero of g is the sequence generated by the iteration

$$x_{n+1} := x_n - \big(\mathrm{D}g(x_n)\big)^{-1} g(x_n), \tag{4.1}$$

where $\mathrm{D}g(x_n) \colon \mathcal{X} \to \mathcal{X}$ is the Fréchet derivative of g at x_n. Newton's method is often applied to find critical points of $f \colon \mathcal{X} \to \mathbb{R}$, i.e. points where $\mathrm{D}f$ vanishes, in which case the iteration is.

$$x_{n+1} := x_n - \big(\mathrm{D}^2 f(x_n)\big)^{-1} \mathrm{D}f(x_n). \tag{4.2}$$

(In (4.2), the second derivative (Hessian) $\mathrm{D}^2 f(x_n)$ is interpreted as a linear map $\mathcal{X} \to \mathcal{X}$ rather than a bilinear map $\mathcal{X} \times \mathcal{X} \to \mathbb{R}$.)

Remark 4.5. (a) Newton's method for the determination of critical points of f amounts to local quadratic approximation: we model f about x_n using its Taylor expansion up to second order, and then take as x_{n+1} a critical point of this quadratic approximation. In particular, as shown in Exercise 4.3, Newton's method yields the exact minimizer of f in one iteration when f is in fact a quadratic function.

(b) We will not dwell at this point on the important practical issue of numerical (and hence approximate) evaluation of derivatives for methods such as Newton iteration. However, this issue will be revisited in Section 10.2 in the context of sensitivity analysis.

For objective functions $f\colon \mathcal{X} \to \mathbb{R} \cup \{\pm\infty\}$ that have little to no smoothness, or that have many local extremizers, it is often necessary to resort to random searches of the space $\mathcal{X}$. For such algorithms, there can only be a probabilistic guarantee of convergence. The rate of convergence and the degree of approximate optimality naturally depend upon features like randomness of the generation of new elements of $\mathcal{X}$ and whether the extremizers of f are difficult to reach, e.g. because they are located in narrow 'valleys'. We now describe three very simple random iterative algorithms for minimization of a prescribed objective function `f`, in order to illustrate some of the relevant issues. For simplicity, suppose that `f` has a unique global minimizer `x_min` and write `f_min` for `f(x_min)`.

Algorithm 4.6 (Random sampling)**.** For simplicity, the following algorithm runs for `n_max` steps with no convergence checks. The algorithm returns an approximate minimizer `x_best` along with the corresponding value of `f`. Suppose that `random()` generates independent samples of $\mathcal{X}$ from a probability measure μ with support $\mathcal{X}$.

```
f_best = +inf
n = 0
while n < n_max:
  x_new = random()
  f_new = f(x_new)
  if f_new < f_best:
    x_best = x_new
    f_best = f_new
  n = n + 1
return [x_best, f_best]
```

A weakness of Algorithm 4.6 is that it completely neglects local information about `f`. Even if the current state `x_best` is very close to the global minimizer `x_min`, the algorithm may continue to sample points `x_new` that are very far away and have `f(x_new)` $\gg$ `f(x_best)`. It would be preferable to explore a neighbourhood of `x_best` more thoroughly and hence find a better approximation of `[x_min, f_min]`. The next algorithm attempts to rectify this deficiency.

Algorithm 4.7 (Random walk)**.** As before, this algorithm runs for `n_max` steps. The algorithm returns an approximate minimizer `x_best` along with the corresponding value of `f`. Suppose that an initial state `x0` is given, and that `jump()` generates independent samples of $\mathcal{X}$ from a probability measure μ with support equal to the unit ball of $\mathcal{X}$.

```
x_best = x0
f_best = f(x_best)
n = 0
while n < n_max:
  x_new = x_best + jump()
  f_new = f(x_new)
  if f_new < f_best:
    x_best = x_new
    f_best = f_new
  n = n + 1
return [x_best, f_best]
```

Algorithm 4.7 also has a weakness: since the state is only ever updated to states with a strictly lower value of `f`, and only looks for new states within unit distance of the current one, the algorithm is prone to becoming stuck in local minima if they are surrounded by wells that are sufficiently wide, even if they are very shallow. The next algorithm, the *simulated annealing* method of Kirkpatrick et al. (1983), attempts to rectify this problem by allowing the optimizer to make some 'uphill' moves, which can be accepted or rejected according to comparison of a uniformly distributed random variable with a user-prescribed acceptance probability function. Therefore, in the simulated annealing algorithm, a distinction is made between the current state `x` of the algorithm and the best state so far, `x_best`; unlike in the previous two algorithms, proposed states `x_new` may be accepted and become `x` even if `f(x_new)` $>$ `f(x_best)`. The idea is to introduce a parameter `T`, to be thought of as 'temperature': the optimizer starts off 'hot', and 'uphill' moves are likely to be accepted; by the end of the calculation, the optimizer is relatively 'cold', and 'uphill' moves are unlikely to accepted.

Algorithm 4.8 (Simulated annealing)**.** Suppose that an initial state `x0` is given. Suppose also that functions `temperature()`, `neighbour()` and `acceptance_prob()` have been specified. Suppose that `uniform()` generates independent samples from the uniform distribution on $[0, 1]$. Then the simulated annealing algorithm is

```
x = x0
fx = f(x)
x_best = x
f_best = fx
n = 0
while n < n_max:
```

```
    T = temperature(n / n_max)
    x_new = neighbour(x)
    f_new = f(x_new)
    if acceptance_prob(fx, f_new, T) > uniform():
      x = x_new
      fx = f_new
    if f_new < f_best:
      x_best = x_new
      f_best = f_new
    n = n + 1
  return [x_best, f_best]
```

Like Algorithm 4.6, the simulated annealing method can guarantee to find the global minimizer of `f` provided that the `neighbour()` function allows full exploration of the state space and the maximum run time `n_max` is large enough. However, the difficulty lies in coming up with functions `temperature()` and `acceptance_prob()` such that the algorithm finds the global minimizer in reasonable time: simulated annealing calculations can be extremely computationally costly. A commonly used acceptance probability function P is the one from the *Metropolis–Hastings algorithm* (see also Section 9.5):

$$P(e, e', T) = \begin{cases} 1, & \text{if } e' < e, \\ \exp(-(e' - e)/T), & \text{if } e' \geq e. \end{cases}$$

There are, however, many other choices; in particular, it is not necessary to automatically accept downhill moves, and it is permissible to have $P(e, e', T) < 1$ for $e' < e$.

4.3 Constrained Optimization

It is well known that the unconstrained extremizers of smooth enough functions must be critical points, i.e. points where the derivative vanishes. The following theorem, the Lagrange multiplier theorem, states that the constrained minimizers of a smooth enough function, subject to smooth enough equality constraints, are critical points of an appropriately generalized function:

Theorem 4.9 (Lagrange multipliers)**.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be real Banach spaces. Let $U \subseteq \mathcal{X}$ be open and let $f \in \mathcal{C}^1(U; \mathbb{R})$. Let $g \in \mathcal{C}^1(U; \mathcal{Y})$, and suppose that $x \in U$ is a constrained extremizer of f subject to the constraint that $g(x) = 0$. Suppose also that the Fréchet derivative $\mathrm{D}g(x) \colon \mathcal{X} \to \mathcal{Y}$ is surjective. Then there exists a* Lagrange multiplier *$\lambda \in \mathcal{Y}'$ such that (x, λ) is an unconstrained critical point of the* Lagrangian *$\mathcal{L}$ defined by*

$$U \times \mathcal{Y}' \ni (x, \lambda) \mapsto \mathcal{L}(x, \lambda) := f(x) + \langle \lambda \,|\, g(x) \rangle \in \mathbb{R}.$$

i.e. $\mathrm{D}f(x) = -\lambda \circ \mathrm{D}g(x)$ as linear maps from $\mathcal{X}$ to $\mathbb{R}$.

The corresponding result for inequality constraints is the Karush–Kuhn–Tucker theorem, which we state here for a finite system of inequality constraints:

Theorem 4.10 (Karush–Kuhn–Tucker)**.** *Let U be an open subset of a Banach space $\mathcal{X}$, and let $f \in \mathcal{C}^1(U;\mathbb{R})$ and $h \in \mathcal{C}^1(U;\mathbb{R}^m)$. Suppose that $x \in U$ is a local minimizer of f subject to the inequality constraints $h_i(x) \leq 0$ for $i = 1, \ldots, m$, and suppose that $\mathrm{D}h(x) \colon \mathcal{X} \to \mathbb{R}^m$ is surjective. Then there exists $\mu = (\mu_1, \ldots, \mu_m) \in (\mathbb{R}^m)'$ such that*

$$-\mathrm{D}f(x) = \mu \circ \mathrm{D}h(x),$$

where μ satisfies the dual feasibility criteria $\mu_i \geq 0$ and the complementary slackness criteria $\mu_i h_i(x) = 0$ for $i = 1, \ldots, m$.

The Lagrange and Karush–Kuhn–Tucker theorems can be combined to incorporate equality constraints g_i and inequality constraints h_j. Strictly speaking, the validity of the Karush–Kuhn–Tucker theorem also depends upon some regularity conditions on the constraints called *constraint qualification conditions*, of which there are many variations that can easily be found in the literature. A very simple one is that if g_i and h_j are affine functions, then no further regularity is needed; another is that the gradients of the active inequality constraints and the gradients of the equality constraints be linearly independent at the optimal point x.

Numerical Implementation of Constraints. In the numerical treatment of constrained optimization problems, there are many ways to implement constraints, not all of which actually *enforce* the constraints in the sense of ensuring that trial states `x_new`, accepted states `x`, or even the final solution `x_best` are actually members of the feasible set. For definiteness, consider the constrained minimization problem

$$\begin{aligned} \text{minimize: } & f(x) \\ \text{with respect to: } & x \in \mathcal{X} \\ \text{subject to: } & c(x) \leq 0 \end{aligned}$$

for some functions $f, c \colon \mathcal{X} \to \mathbb{R} \cup \{\pm\infty\}$. One way of seeing the constraint '$c(x) \leq 0$' is as a Boolean true/false condition: either the inequality is satisfied, or it is not. Supposing that `neighbour(x)` generates new (possibly infeasible) elements of $\mathcal{X}$ given a current state `x`, one approach to generating feasible trial states `x_new` is the following:

```
x' = neighbour(x)
while c(x') > 0:
  x' = neighbour(x)
x_new = x'
```

However, this accept/reject approach is extremely wasteful: if the feasible set is very small, then `x'` will 'usually' be rejected, thereby wasting a lot of computational time, and this approach takes no account of how 'nearly feasible' an infeasible `x'` might be.

One alternative approach is to use *penalty functions*: instead of considering the constrained problem of minimizing $f(x)$ subject to $c(x) \leq 0$, one can consider the unconstrained problem of minimizing $x \mapsto f(x) + p(x)$, where $p \colon \mathcal{X} \to [0, \infty)$ is some function that equals zero on the feasible set and takes larger values the 'more' the constraint inequality $c(x) \leq 0$ is violated, e.g., for $\mu > 0$.

$$p_\mu(x) = \begin{cases} 0, & \text{if } c(x) \leq 0, \\ \exp(c(x)/\mu) - 1, & \text{if } c(x) > 0. \end{cases}$$

The hope is that (a) the minimization of $f + p_\mu$ over all of $\mathcal{X}$ is easy, and (b) as $\mu \to 0$, minimizers of $f + p_\mu$ converge to minimizers of f on the original feasible set. The penalty function approach is attractive, but the choice of penalty function is rather ad hoc, and issues can easily arise of competition between the penalties corresponding to multiple constraints.

An alternative to the use of penalty functions is to construct *constraining functions* that enforce the constraints exactly. That is, we seek a function `C()` that takes as input a possibly infeasible `x'` and returns some `x_new = C(x')` that is guaranteed to satisfy the constraint `c(x_new) <= 0`. For example, suppose that $\mathcal{X} = \mathbb{R}^n$ and the feasible set is the Euclidean unit ball, so the constraint is

$$c(x) := \|x\|_2^2 - 1 \leq 0.$$

Then a suitable constraining function could be

$$C(x) := \begin{cases} x, & \text{if } \|x\|_2 \leq 1, \\ x/\|x\|_2, & \text{if } \|x\|_2 > 1. \end{cases}$$

Constraining functions are very attractive because the constraints are treated exactly. However, they must often be designed on a case-by-case basis for each constraint function c, and care must be taken to ensure that multiple constraining functions interact well and do not unduly favour parts of the feasible set over others; for example, the above constraining function C maps the entire infeasible set to the unit sphere, which might be considered undesirable in certain settings, and so a function such as

$$\widetilde{C}(x) := \begin{cases} x, & \text{if } \|x\|_2 \leq 1, \\ x/\|x\|_2^2, & \text{if } \|x\|_2 > 1. \end{cases}$$

might be more appropriate. Finally, note that the original accept/reject method of finding feasible states is a constraining function in this sense, albeit a very inefficient one.

4.4 Convex Optimization

The topic of this section is *convex optimization*. As will be seen, convexity is a powerful property that makes optimization problems tractable to a much greater extent than any amount of smoothness (which still permits local minima) or low-dimensionality can do.

In this section, $\mathcal{X}$ will be a normed vector space. (More generally, the properties that are of importance to the discussion hold for any Hausdorff, locally convex topological vector space.) Given two points x_0 and x_1 of $\mathcal{X}$ and $t \in [0,1]$, x_t will denote the *convex combination*

$$x_t := (1-t)x_0 + tx_1.$$

More generally, given points $x_0, \ldots, x_n$ of a vector space, a sum of the form

$$\alpha_0 x_0 + \cdots + \alpha_n x_n$$

is called a *linear combination* if the α_i are any field elements, an *affine combination* if their sum is 1, and a *convex combination* if they are non-negative and sum to 1.

Definition 4.11. (a) A subset $K \subseteq \mathcal{X}$ is a *convex set* if, for all $x_0, x_1 \in K$ and $t \in [0,1]$, $x_t \in K$; it is said to be *strictly convex* if $x_t \in \mathring{K}$ whenever x_0 and x_1 are distinct points of $\bar{K}$ and $t \in (0,1)$.

(b) An *extreme point* of a convex set K is a point of K that cannot be written as a non-trivial convex combination of distinct elements of K; the set of all extreme points of K is denoted $\mathrm{ext}(K)$.

(c) The *convex hull* $\mathrm{co}(S)$ (resp. *closed convex hull* $\overline{\mathrm{co}}(S)$) of $S \subseteq \mathcal{X}$ is defined to be the intersection of all convex (resp. closed and convex) subsets of $\mathcal{X}$ that contain S.

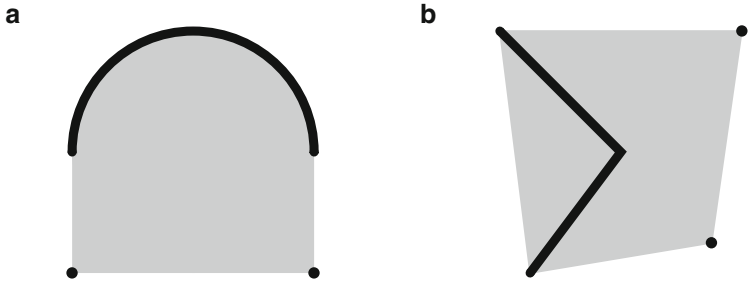
Example 4.12. (a) The square $[-1,1]^2$ is a convex subset of $\mathbb{R}^2$, but is not strictly convex, and its extreme points are the four vertices $(\pm 1, \pm 1)$.

(b) The closed unit disc $\{(x,y) \in \mathbb{R}^2 \mid x^2 + y^2 \leq 1\}$ is a strictly convex subset of $\mathbb{R}^2$, and its extreme points are the points of the unit circle $\{(x,y) \in \mathbb{R}^2 \mid x^2 + y^2 = 1\}$.

(c) If $p_0, \ldots, p_d \in \mathcal{X}$ are distinct points such that $p_1 - p_0, \ldots, p_d - p_0$ are linearly independent, then their (closed) convex hull is called a *d-dimensional simplex*. The points $p_0, \ldots, p_d$ are the extreme points of the simplex.

(d) See Figure 4.1 for further examples.

Example 4.13. $\mathcal{M}_1(\mathcal{X})$ is a convex subset of the space of all (signed) Borel measures on $\mathcal{X}$. The extremal probability measures are the *zero-one measures*, i.e. those for which, for every measurable set $E \subseteq \mathcal{X}$, $\mu(E) \in \{0,1\}$. Furthermore, as will be discussed in Chapter 14, if $\mathcal{X}$ is, say, a Polish space,



A convex set (grey) and its set of extreme points (black).

A non-convex set (black) and its convex hull (grey).

Fig. 4.1: Convex sets, extreme points and convex hulls of some subsets of the plane $\mathbb{R}^2$.

then the zero-one measures (and hence the extremal probability measures) on $\mathcal{X}$ are the Dirac point masses. Indeed, in this situation,

$$\mathcal{M}_1(\mathcal{X}) = \overline{\text{co}}\big(\{\delta_x \mid x \in \mathcal{X}\}\big) \subseteq \mathcal{M}_{\pm}(\mathcal{X}).$$

The principal reason to confine attention to normed spaces[1] $\mathcal{X}$ is that it is highly inconvenient to have to work with spaces for which the following 'common sense' results do not hold:

Theorem 4.14 (Kreĭn–Milman)**.** *Let $K \subseteq \mathcal{X}$ be compact and convex. Then K is the closed convex hull of its extreme points.*

Theorem 4.15 (Choquet–Bishop–de Leeuw)**.** *Let $K \subseteq \mathcal{X}$ be compact and convex, and let $c \in K$. Then there exists a probability measure p supported on* $\text{ext}(K)$ *such that, for all affine functions f on K,*

$$f(c) = \int_{\text{ext}(K)} f(e)\,\mathrm{d}p(e).$$

The point c in Theorem 4.15 is called a *barycentre* of the set K, and the probability measure p is said to *represent* the point c. Informally speaking, the Kreĭn–Milman and Choquet–Bishop–de Leeuw theorems together ensure that a compact, convex subset K of a topologically respectable space is entirely characterized by its set of extreme points in the following sense: every point of K can be obtained as an average of extremal points of K, and, indeed, the value of any affine function at any point of K can be obtained as an average of its values at the extremal points in the same way.

[1] Or, more generally, Hausdorff, locally convex, topological vector spaces.

Definition 4.16. Let $K \subseteq \mathcal{X}$ be convex. A function $f\colon K \to \mathbb{R} \cup \{\pm\infty\}$ is a *convex function* if, for all $x_0, x_1 \in K$ and $t \in [0,1]$,

$$f(x_t) \leq (1-t)f(x_0) + tf(x_1), \tag{4.3}$$

and is called a *strictly convex function* if, for all distinct $x_0, x_1 \in K$ and $t \in (0,1)$,

$$f(x_t) < (1-t)f(x_0) + tf(x_1).$$

The inequality (4.3) defining convexity can be seen as a special case — with $X \sim \mu$ supported on two points x_0 and x_1 — of the following result:

Theorem 4.17 (Jensen)**.** *Let $(\Theta, \mathscr{F}, \mu)$ be a probability space, let $K \subseteq \mathcal{X}$ and $f\colon K \to \mathbb{R} \cup \{\pm\infty\}$ be convex, and let $X \in L^1(\Theta, \mu; \mathcal{X})$ take values in K. Then*

$$f\big(\mathbb{E}_\mu[X]\big) \leq \mathbb{E}_\mu\big[f(X)\big], \tag{4.4}$$

where $\mathbb{E}_\mu[X] \in \mathcal{X}$ is defined by the relation $\langle \ell \,|\, \mathbb{E}_\mu[X]\rangle = \mathbb{E}_\mu[\langle \ell \,|\, X\rangle]$ for every $\ell \in \mathcal{X}'$. Furthermore, if f is strictly convex, then equality holds in (4.4) if and only if X is μ-almost surely constant.

It is straightforward to see that $f\colon K \to \mathbb{R}\cup\{\pm\infty\}$ is convex (resp. strictly convex) if and only if its *epigraph*

$$\operatorname{epi}(f) := \{(x, v) \in K \times \mathbb{R} \mid v \geq f(x)\}$$

is a convex (resp. strictly convex) subset of $K \times \mathbb{R}$. Furthermore, twice-differentiable convex functions are easily characterized in terms of their second derivative (Hessian):

Theorem 4.18. *Let $f\colon K \to \mathbb{R}$ be twice continuously differentiable on an open, convex set K. Then f is convex if and only if $\mathrm{D}^2 f(x)$ is positive semi-definite for all $x \in K$. If $\mathrm{D}^2 f(x)$ is positive definite for all $x \in K$, then f is strictly convex, though the converse is false.*

Convex functions have many convenient properties with respect to minimization and maximization:

Theorem 4.19. *Let $f\colon K \to \mathbb{R}$ be a convex function on a convex set $K \subseteq \mathcal{X}$. Then*
(a) any local minimizer of f in K is also a global minimizer;
(b) the set $\arg\min_K f$ of global minimizers of f in K is convex;
(c) if f is strictly convex, then it has at most one global minimizer in K;
(d) f has the same maximum values on K and $\operatorname{ext}(K)$.

Proof. (a) Suppose that x_0 is a local minimizer of f in K that is not a global minimizer: that is, suppose that x_0 is a minimizer of f in some open neighbourhood N of x_0, and also that there exists $x_1 \in K \setminus N$ such that $f(x_1) < f(x_0)$. Then, for sufficiently small $t > 0$, $x_t \in N$, but convexity implies that

$$f(x_t) \leq (1-t)f(x_0) + tf(x_1) < (1-t)f(x_0) + tf(x_0) = f(x_0),$$

which contradicts the assumption that x_0 is a minimizer of f in N.

(b) Suppose that $x_0, x_1 \in K$ are global minimizers of f. Then, for all $t \in [0,1]$, $x_t \in K$ and

$$f(x_0) \leq f(x_t) \leq (1-t)f(x_0) + tf(x_1) = f(x_0).$$

Hence, $x_t \in \arg\min_K f$, and so $\arg\min_K f$ is convex.

(c) Suppose that $x_0, x_1 \in K$ are distinct global minimizers of f, and let $t \in (0,1)$. Then $x_t \in K$ and

$$f(x_0) \leq f(x_t) < (1-t)f(x_0) + tf(x_1) = f(x_0),$$

which is a contradiction. Hence, f has at most one minimizer in K.

(d) Suppose that $c \in K \setminus \operatorname{ext}(K)$ has $f(c) > \sup_{\operatorname{ext}(K)} f$. By Theorem 4.15, there exists a probability measure p on $\operatorname{ext}(K)$ such that, for all affine functions ℓ on K,

$$\ell(c) = \int_{\operatorname{ext}(K)} \ell(x)\,\mathrm{d}p(x).$$

i.e. $c = \mathbb{E}_{X\sim p}[X]$. Then Jensen's inequality implies that

$$\mathbb{E}_{X\sim p}\big[f(X)\big] \geq f(c) > \sup_{\operatorname{ext}(K)} f,$$

which is a contradiction. Hence, since $\sup_K f \geq \sup_{\operatorname{ext}(K)} f$, f must have the same maximum value on $\operatorname{ext}(K)$ as it does on K. □

Remark 4.20. Note well that Theorem 4.19 does not assert the existence of minimizers, which requires non-emptiness and compactness of K, and lower semicontinuity of f. For example:

- the exponential function on $\mathbb{R}$ is strictly convex, continuous and bounded below by 0 yet has no minimizer;
- the interval $[-1,1]$ is compact, and the function $f\colon [-1,1] \to \mathbb{R} \cup \{\pm\infty\}$ defined by

$$f(x) := \begin{cases} x, & \text{if } |x| < \frac{1}{2}, \\ +\infty, & \text{if } |x| \geq \frac{1}{2}, \end{cases}$$

is convex, yet f has no minimizer — although $\inf_{x\in[-1,1]} f(x) = -\frac{1}{2}$, there is no x for which $f(x)$ attains this infimal value.

Definition 4.21. A *convex optimization problem* (or *convex program*) is a minimization problem in which the objective function and all constraints are equalities or inequalities with respect to convex functions.

Remark 4.22. (a) Beware of the common pitfall of saying that a convex program is simply the minimization of a convex function over a convex

set. Of course, by Theorem 4.19, such minimization problems are nicer than general minimization problems, but bona fide convex programs are an even nicer special case.

(b) In practice, many problems are not obviously convex programs, but can be transformed into convex programs by, e.g., a cunning change of variables. Being able to spot the right equivalent problem is a major part of the art of optimization.

It is difficult to overstate the importance of convexity in making optimization problems tractable. Indeed, it has been remarked that lack of convexity is a much greater obstacle to tractability than high dimension. There are many powerful methods for the solution of convex programs, with corresponding standard software libraries such as `cvxopt`. For example, *interior point methods* explore the interior of the feasible set in search of the solution to the convex program, while being kept away from the boundary of the feasible set by a *barrier function*. The discussion that follows is only intended as an outline; for details, see Boyd and Vandenberghe (2004, Chapter 11).

Consider the convex program

$$\begin{aligned} \text{minimize: } & f(x) \\ \text{with respect to: } & x \in \mathbb{R}^n \\ \text{subject to: } & c_i(x) \leq 0 \quad \text{for } i = 1, \ldots, m, \end{aligned}$$

where the functions $f, c_1, \ldots, c_m \colon \mathbb{R}^n \to \mathbb{R}$ are all convex and differentiable. Let F denote the feasible set for this program. Let $0 < \mu \ll 1$ be a small scalar, called the *barrier parameter*, and define the *barrier function* associated to the program by

$$B(x; \mu) := f(x) - \mu \sum_{i=1}^{m} \log c_i(x).$$

Note that $B(\,\cdot\,; \mu)$ is strictly convex for $\mu > 0$, that $B(x; \mu) \to +\infty$ as $x \to \partial F$, and that $B(\,\cdot\,; 0) = f$; therefore, the unique minimizer x_μ^* of $B(\,\cdot\,; \mu)$ lies in $\mathring{F}$ and (hopefully) converges to the minimizer of the original problem as $\mu \to 0$. Indeed, using arguments based on convex duality, one can show that

$$f(x_\mu^*) - \inf_{x \in F} f(x) \leq m\mu.$$

The strictly convex problem of minimizing $B(\,\cdot\,; \mu)$ can be solved approximately using Newton's method. In fact, however, one settles for a partial minimization of $B(\,\cdot\,; \mu)$ using only one or two steps of Newton's method, then decreases μ to μ', performs another partial minimization of $B(\,\cdot\,; \mu')$ using Newton's method, and so on in this alternating fashion.

4.5 Linear Programming

Theorem 4.19 has the following immediate corollary for the minimization and maximization of affine functions on convex sets:

Corollary 4.23. *Let $\ell\colon K \to \mathbb{R}$ be a continuous affine function on a non-empty, compact, convex set $K \subseteq \mathcal{X}$. Then*

$$\operatorname{ext}\{\ell(x) \mid x \in K\} = \operatorname{ext}\{\ell(x) \mid x \in \operatorname{ext}(K)\}.$$

That is, ℓ has the same minimum and maximum values over both K and the set of extreme points of K.

Definition 4.24. A *linear program* is an optimization problem of the form

$$\begin{aligned} \text{extremize: } & f(x) \\ \text{with respect to: } & x \in \mathbb{R}^p \\ \text{subject to: } & g_i(x) \leq 0 \quad \text{for } i = 1, \ldots, q, \end{aligned}$$

where the functions $f, g_1, \ldots, g_q\colon \mathbb{R}^p \to \mathbb{R}$ are all affine functions. Linear programs are often written in the *canonical form*

$$\begin{aligned} \text{maximize: } & c \cdot x \\ \text{with respect to: } & x \in \mathbb{R}^n \\ \text{subject to: } & Ax \leq b \\ & x \geq 0, \end{aligned}$$

where $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ are given, and the two inequalities are interpreted componentwise. (Conversion to canonical form, and in particular the introduction of the non-negativity constraint $x \geq 0$, is accomplished by augmenting the original $x \in \mathbb{R}^p$ with additional variables called *slack variables* to form the extended variable $x \in \mathbb{R}^n$.)

Note that the feasible set for a linear program is an intersection of finitely many half-spaces of $\mathbb{R}^n$, i.e. a *polytope*. This polytope may be empty, in which case the constraints are mutually contradictory and the program is said to be *infeasible*. Also, the polytope may be unbounded in the direction of c, in which case the extreme value of the problem is infinite.

Since linear programs are special cases of convex programs, methods such as interior point methods are applicable to linear programs as well. Such methods approach the optimum point x^*, which is necessarily an extremal element of the feasible polytope, from the interior of the feasible polytope. Historically, however, such methods were preceded by methods such as Dantzig's simplex algorithm, which sets out to directly explore the set of extreme points in a (hopefully) efficient way. Although the theoretical worst-case complexity of simplex method as formulated by Dantzig is exponential

in n and m, in practice the simplex method is remarkably efficient (typically having polynomial running time) provided that certain precautions are taken to avoid pathologies such as 'stalling'.

4.6 Least Squares

An elementary example of convex programming is unconstrained quadratic minimization, otherwise known as *least squares*. Least squares minimization plays a central role in elementary statistical estimation, as will be demonstrated by the Gauss–Markov theorem (Theorem 6.2). The next three results show that least squares problems have unique solutions, which are given in terms of an orthogonality criterion, which in turn reduces to a system of linear equations, the *normal equations*.

Lemma 4.25. *Let K be a non-empty, closed, convex subset of a Hilbert space $\mathcal{H}$. Then, for each $y \in \mathcal{H}$, there is a unique element $\hat{x} = \Pi_K y \in K$ such that*

$$\hat{x} \in \operatorname*{arg\,min}_{x \in K} \|y - x\|.$$

Proof. By Exercise 4.1, the function $J\colon \mathcal{X} \to [0, \infty)$ defined by $J(x) := \|y - x\|^2$ is strictly convex, and hence it has at most one minimizer in K. Therefore, it only remains to show that J has at least one minimizer in K. Since J is bounded below (on $\mathcal{X}$, not just on K), J has a sequence of approximate minimizers: let

$$I := \inf_{x \in K} \|y - x\|^2, \quad I^2 \leq \|y - x_n\|^2 \leq I^2 + \tfrac{1}{n}.$$

By the parallelogram identity for the Hilbert norm $\|\cdot\|$,

$$\|(y - x_m) + (y - x_n)\|^2 + \|(y - x_m) - (y - x_n)\|^2 = 2\|y - x_m\|^2 + 2\|y - x_n\|^2,$$

and hence

$$\|2y - (x_m + x_n)\|^2 + \|x_n - x_m\|^2 \leq 4I^2 + \tfrac{2}{n} + \tfrac{2}{m}.$$

Since K is convex, $\frac{1}{2}(x_m + x_n) \in K$, so the first term on the left-hand side above is bounded below as follows:

$$\|2y - (x_m + x_n)\|^2 = 4 \left\| y - \frac{x_m + x_n}{2} \right\|^2 \geq 4I^2.$$

Hence,

$$\|x_n - x_m\|^2 \leq 4I^2 + \tfrac{2}{n} + \tfrac{2}{m} - 4I^2 = \tfrac{2}{n} + \tfrac{2}{m},$$

and so the sequence $(x_n)_{n \in \mathbb{N}}$ is Cauchy; since $\mathcal{H}$ is complete and K is closed, this sequence converges to some $\hat{x} \in K$. Since the norm $\|\cdot\|$ is continuous, $\|y - \hat{x}\| = I$. □

Lemma 4.26 (Orthogonality of the residual)**.** *Let V be a closed subspace of a Hilbert space $\mathcal{H}$ and let $b \in \mathcal{H}$. Then $\hat{x} \in V$ minimizes the distance to b if and only if the* residual *$\hat{x} - b$ is orthogonal to V, i.e.*

$$\hat{x} = \operatorname*{arg\,min}_{x \in V} \|x - b\| \iff (\hat{x} - b) \perp V.$$

Proof. Let $J(x) := \frac{1}{2}\|x - b\|^2$, which has the same minimizers as $x \mapsto \|x - b\|$; by Lemma 4.25, such a minimizer exists and is unique. Suppose that $(x - b) \perp V$ and let $y \in V$. Then $y - x \in V$ and so $(y - x) \perp (x - b)$. Hence, by Pythagoras' theorem,

$$\|y - b\|^2 = \|y - x\|^2 + \|x - b\|^2 \geq \|x - b\|^2,$$

and so x minimizes J.

Conversely, suppose that x minimizes J. Then, for every $y \in V$,

$$0 = \frac{\partial}{\partial \lambda} J(x + \lambda y)\bigg|_{\lambda=0} = \frac{1}{2}\left(\langle y, x - b\rangle + \langle x - b, y\rangle\right) = \operatorname{Re}\langle x - b, y\rangle$$

and, in the complex case,

$$0 = \frac{\partial}{\partial \lambda} J(x + \lambda i y)\bigg|_{\lambda=0} = \frac{1}{2}\left(-i\langle y, x - b\rangle + i\langle x - b, y\rangle\right) = -\operatorname{Im}\langle x - b, y\rangle.$$

Hence, $\langle x - b, y\rangle = 0$, and since y was arbitrary, $(x - b) \perp V$. □

Lemma 4.27 (Normal equations)**.** *Let $A\colon \mathcal{H} \to \mathcal{K}$ be a linear operator between Hilbert spaces such that* $\operatorname{ran} A \subseteq \mathcal{K}$ *is closed. Then, given $b \in \mathcal{K}$,*

$$\hat{x} \in \operatorname*{arg\,min}_{x \in \mathcal{H}} \|Ax - b\|_{\mathcal{K}} \iff A^*A\hat{x} = A^*b, \tag{4.5}$$

the equations on the right-hand side being known as the normal equations*. If, in addition, A is injective, then A^*A is invertible and the least squares problem / normal equations have a unique solution.*

Proof. As a consequence of completeness, the only element of a Hilbert space that is orthogonal to every other element of the space is the zero element. Hence,

$$\begin{aligned}
&\|Ax - b\|_{\mathcal{K}} \text{ is minimal} \\
&\quad\iff (Ax - b) \perp Av \text{ for all } v \in \mathcal{H} && \text{by Lemma 4.26} \\
&\quad\iff \langle Ax - b, Av\rangle_{\mathcal{K}} = 0 \text{ for all } v \in \mathcal{H} \\
&\quad\iff \langle A^*Ax - A^*b, v\rangle_{\mathcal{H}} = 0 \text{ for all } v \in \mathcal{H} \\
&\quad\iff A^*Ax = A^*b && \text{by completeness of } \mathcal{H},
\end{aligned}$$

and this shows the equivalence (4.5).

By Proposition 3.16(d), $\ker A^* = (\operatorname{ran} A)^\perp$. Therefore, the restriction of A^* to the range of A is injective. Hence, if A itself is injective, then it follows that A^*A is injective. Again by Proposition 3.16(d), $(\operatorname{ran} A^*)^\perp = \ker A = \{0\}$, and since $\mathcal{H}$ is complete, this implies that A^* is surjective. Since A is surjective onto its range, it follows that A^*A is surjective, and hence bijective and invertible. □

Weighting and Regularization. It is common in practice that one does not want to minimize the $\mathcal{K}$-norm directly, but perhaps some re-weighted version of the $\mathcal{K}$-norm. This re-weighting is accomplished by a self-adjoint and positive definite[2] operator $Q\colon \mathcal{K} \to \mathcal{K}$: we define a new inner product and norm on $\mathcal{K}$ by

$$\begin{aligned} \langle k, k' \rangle_Q &:= \langle k, Qk' \rangle_{\mathcal{K}}, \\ \|k\|_Q &:= \langle k, k \rangle_Q^{1/2}. \end{aligned}$$

It is a standard fact that the self-adjoint operator Q possesses an operator square root, i.e. a self-adjoint $Q^{1/2}\colon \mathcal{K} \to \mathcal{K}$ such that $Q^{1/2}Q^{1/2} = Q$; for reasons of symmetry, it is common to express the inner product and norm induced by Q using this square root:

$$\begin{aligned} \langle k, k' \rangle_Q &= \left\langle Q^{1/2}k, Q^{1/2}k' \right\rangle_{\mathcal{K}}, \\ \|k\|_Q &= \left\|Q^{1/2}k\right\|_{\mathcal{K}}. \end{aligned}$$

We then consider the problem, given $b \in \mathcal{K}$, of finding $x \in \mathcal{H}$ to minimize

$$\frac{1}{2}\|Ax - b\|_Q^2 \equiv \frac{1}{2}\left\|Q^{1/2}(Ax - b)\right\|_{\mathcal{K}}^2.$$

Another situation that arises frequently in practice is that the normal equations do not have a unique solution (e.g. because A^*A is not invertible) and so it is necessary to select one by some means, or that one has some prior belief that 'the right solution' should be close to some initial guess x_0. A technique that accomplishes both of these aims is *Tikhonov regularization* (known in the statistics literature as *ridge regression*). In this situation, we minimize the following sum of two quadratic functionals:

$$\frac{1}{2}\|Ax - b\|_{\mathcal{K}}^2 + \frac{1}{2}\|x - x_0\|_R^2,$$

where $R\colon \mathcal{H} \to \mathcal{H}$ is self-adjoint and positive definite, and $x_0 \in \mathcal{H}$.

[2] If Q is not positive definite, but merely positive semi-definite and self-adjoint, then existence of solutions to the associated least squares problems still holds, but uniqueness can fail.

These two modifications to ordinary least squares, weighting and regularization, can be combined. The normal equations for weighted and regularized least squares are easily derived from Lemma 4.27:

Theorem 4.28 (Normal equations for weighted and Tikhonov-regularized least squares)**.** *Let $\mathcal{H}$ and $\mathcal{K}$ be Hilbert spaces, let $A\colon \mathcal{H} \to \mathcal{K}$ have closed range, let Q and R be self-adjoint and positive definite on $\mathcal{K}$ and $\mathcal{H}$ respectively, and let $b \in \mathcal{K}$, $x_0 \in \mathcal{H}$. Let*

$$J(x) := \frac{1}{2}\|Ax - b\|_Q^2 + \frac{1}{2}\|x - x_0\|_R^2.$$

Then

$$\hat{x} \in \operatorname*{arg\,min}_{x \in \mathcal{H}} J(x) \iff (A^*QA + R)\hat{x} = A^*Qb + Rx_0.$$

Proof. Exercise 4.4. □

It is also interesting to consider regularizations that do not come from a Hilbert norm, but instead from some other function. As will be elaborated upon in Chapter 6, there is a strong connection between regularized optimization problems and inverse problems, and the choice of regularization in some sense describes the practitioner's 'prior beliefs' about the structure of the solution.

Nonlinear Least Squares and Gauss–Newton Iteration. It often occurs in practice that one wishes to find a vector of parameters $\theta \in \mathbb{R}^p$ such that a function $\mathbb{R}^k \ni x \mapsto f(x;\theta) \in \mathbb{R}^\ell$ best fits a collection of data points $\{(x_i, y_i) \in \mathbb{R}^k \times \mathbb{R}^\ell \mid i = 1, \dots, m\}$. For each candidate parameter vector θ, define the *residual vector*

$$r(\theta) := \begin{bmatrix} r_1(\theta) \\ \vdots \\ r_m(\theta) \end{bmatrix} = \begin{bmatrix} y_1 - f(x_1;\theta) \\ \vdots \\ y_m - f(x_m;\theta) \end{bmatrix} \in \mathbb{R}^m.$$

The aim is to find θ to minimize the objective function $J(\theta) := \|r(\theta)\|_2^2$. Let

$$A := \left.\begin{bmatrix} \frac{\partial r_1(\theta)}{\partial \theta^1} & \cdots & \frac{\partial r_1(\theta)}{\partial \theta^p} \\ \vdots & \ddots & \vdots \\ \frac{\partial r_m(\theta)}{\partial \theta^1} & \cdots & \frac{\partial r_m(\theta)}{\partial \theta^p} \end{bmatrix}\right|_{\theta=\theta_n} \in \mathbb{R}^{m \times p}$$

be the Jacobian matrix of the residual vector, and note that $A = -\mathrm{D}F(\theta_n)$, where

$$F(\theta) := \begin{bmatrix} f(x_1;\theta) \\ \vdots \\ f(x_m;\theta) \end{bmatrix} \in \mathbb{R}^m.$$

Consider the first-order Taylor approximation

$$r(\theta) \approx r(\theta_n) + A(r(\theta) - r(\theta_n)).$$

Thus, to approximately minimize $\|r(\theta)\|_2$, we find $\delta := r(\theta) - r(\theta_n)$ that makes the right-hand side of the approximation equal to zero. This is an ordinary linear least squares problem, the solution of which is given by the normal equations as

$$\delta = (A^*A)^{-1}A^*r(\theta_n).$$

Thus, we obtain the *Gauss–Newton iteration* for a sequence $(\theta_n)_{n\in\mathbb{N}}$ of approximate minimizers of J:

$$\begin{aligned}\theta_{n+1} &:= \theta_n - (A^*A)^{-1}A^*r(\theta_n) \\ &= \theta_n + \big((\mathrm{D}F(\theta_n))^*(\mathrm{D}F(\theta_n))\big)^{-1}(\mathrm{D}F(\theta_n))^*r(\theta_n).\end{aligned}$$

In general, the Gauss–Newton iteration is not guaranteed to converge to the exact solution, particularly if δ is 'too large', in which case it may be appropriate to use a judiciously chosen small positive multiple of δ. The use of Tikhonov regularization in this context is known as the *Levenberg–Marquardt algorithm* or *trust region* method, and the small multiplier applied to δ is essentially the reciprocal of the Tikhonov regularization parameter.

4.7 Bibliography

The book of Boyd and Vandenberghe (2004) is an excellent reference on the theory and practice of convex optimization, as is the associated software library `cvxopt`. The classic reference for convex analysis in general is the monograph of Rockafellar (1997); a more recent text is that of Krantz (2015). A good short reference on Choquet theory is the book of Phelps (2001); in particular, Theorems 4.14 and 4.15 are due to Krein and Milman (1940) and Bishop and de Leeuw (1959) respectively. A standard reference on numerical methods for optimization is the book of Nocedal and Wright (2006). The Banach space version of the Lagrange multiplier theorem, Theorem 4.9, can be found in Zeidler (1995, Section 4.14). Theorem 4.10 originates with Karush (1939) and Kuhn and Tucker (1951); see, e.g., Gould and Tolle (1975) for discussion of the infinite-dimensional version.

For constrained global optimization in the absence of 'nice' features, particularly for the UQ methods in Chapter 14, variations upon the genetic evolution approach, e.g. the differential evolution algorithm (Price et al., 2005; Storn and Price, 1997), have proved up to the task of producing robust results, if not always quick ones. There is no 'one size fits all' approach to constrained global optimization: it is basically impossible to be quick, robust, and general all at the same time.

In practice, it is very useful to work using an optimization framework that provides easy interfaces to many optimization methods, with easy interchange among strategies for population generation, enforcement of constraints, termination criteria, and so on: see, for example, the DAKOTA (Adams et al., 2014) and Mystic (McKerns et al., 2009, 2011) projects.

4.8 Exercises

Exercise 4.1. Let $\|\cdot\|$ be a norm on a vector space $\mathcal{V}$, and fix $\bar{x} \in \mathcal{V}$. Show that the function $J\colon \mathcal{V} \to [0,\infty)$ defined by $J(x) := \|x - \bar{x}\|$ is convex, and that $J(x) := \frac{1}{2}\|x - \bar{x}\|^2$ is strictly convex if the norm is induced by an inner product. Give an example of a norm for which $J(x) := \frac{1}{2}\|x - \bar{x}\|^2$ is not strictly convex.

Exercise 4.2. Let K be a non-empty, closed, convex subset of a Hilbert space $\mathcal{H}$. Lemma 4.25 shows that there is a well-defined function $\Pi_K\colon \mathcal{H} \to K$ that assigns to each $y \in \mathcal{H}$ the unique $\Pi_K y \in K$ that is closest to y with respect to the norm on $\mathcal{H}$.
(a) Prove the variational inequality that $x = \Pi_K y$ if and only if $x \in K$ and

$$\langle x, z - x\rangle \geq \langle y, z - x\rangle \quad \text{for all } z \in K.$$

(b) Prove that Π_K is non-expansive, i.e.

$$\|\Pi_K y_1 - \Pi_K y_2\| \leq \|y_1 - y_2\| \quad \text{for all } y_1, y_2 \in \mathcal{H},$$

and hence a continuous function.

Exercise 4.3. Let $A\colon \mathcal{H} \to \mathcal{K}$ be a linear operator between Hilbert spaces such that $\operatorname{ran} A$ is a closed subspace of $\mathcal{K}$, let $Q\colon \mathcal{K} \to \mathcal{K}$ be self-adjoint and positive-definite, and let $b \in \mathcal{K}$. Let

$$J(x) := \frac{1}{2}\|Ax - b\|_Q^2$$

Calculate the gradient and Hessian (second derivative) of J. Hence show that, regardless of the initial condition $x_0 \in \mathcal{H}$, Newton's method finds the minimum of J in one step.

Exercise 4.4. Prove Theorem 4.28. Hint: Consider the operator from $\mathcal{H}$ into $\mathcal{K} \oplus \mathcal{L}$ given by

$$x \mapsto \begin{bmatrix} Q^{1/2}Ax \\ R^{1/2}x \end{bmatrix}.$$