Chapter 3

Quadratic Programming

In this chapter, we show that the problem of computing the smallest enclosing ball (as well as another interesting problem) can be formulated as a *quadratic program* (QP). The implications are twofold. On the one hand, there are (at least practically) efficient algorithms for computing (approximate) solutions to QP, even in high dimensions; on the other hand, the QP approach shows that the smallest enclosing ball is fully determined by the n^2 pairwise scalar products of the n input points. We will make use of this surprising fact in the next chapter.

3.1 Simple convex programming

A function $f : \mathbb{R}^m \to \mathbb{R}$ is called *convex*, if for all $x, x' \in \mathbb{R}^m$ and all $\lambda \in [0, 1]$,

$$f((1-\lambda)x + \lambda x') \leq (1-\lambda)f(x) + \lambda f(x'). \tag{3.1}$$

Geometrically, this means that any segment connecting two points on the *graph* of f lies above the graph, see Figure 3.1. The graph of f is the set $\{(x, f(x)) \mid x \in \mathbb{R}^m\} \subseteq \mathbb{R}^{m+1}$, and if the function is convex, the graph looks like a 'bowl'.

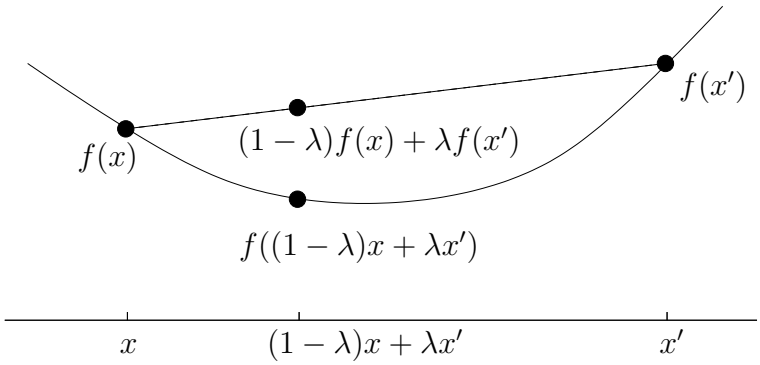

Figure 3.1: A convex function

We do not require strict inequality in (3.1), so the graph of f may be 'flat' in certain parts. In particular, a linear function (whose graph can be considered as a nonvertical hyperplane in $\mathbb{R}^{m+1}$) is convex. Convex functions are continuous.

In the sequel, we will deal with differentiable convex functions that have continuous partial derivatives. In this case, we have

Fact 3.1.1 *f is convex if and only if*

$$f(x) + \nabla f(x)(x' - x) \leq f(x'),$$

for all $x, x' \in \mathbb{R}^m$, where $\nabla f : \mathbb{R}^m \to \mathbb{R}^m$ is the gradient operator (whose values are row vectors by convention).

For a proof (and other interesting material about convex functions), see the very nice book by Peressini, Sullivan and Uhl [9]. Geometrically, this fact says that f is convex if and only if all tangential hyperplanes to the graph of f are below the graph.

Fixing f, we now consider the *simple convex program*

$$\begin{array}{lll} \text{(SCP)} & \text{minimize} & f(x) \\ & \text{subject to} & \sum_{i=1}^m x_i = 1, \\ & & x_i \geq 0, \quad i = 1, \ldots, m. \end{array}$$

This means, we are looking for a point $x \in \mathbb{R}^m$ that satisfies the *constraints*

$$\sum_{i=1}^m x_i = 1, \quad x_i \geq 0, i = 1, \ldots, m$$

and has minimum function value among all such points. Any point satisfying the constraints is a *feasible solution*, and the set of all feasible solutions is the *feasible region*. It is an $m-1$-dimensional *simplex* (see Figure 3.2); in particular, it is a nonempty compact set, and so f as a continuous function does have a minimum over this simplex. Any feasible solution x for which $f(x)$ achieves this minimum value is called a *minimizer* of f over the feasible region, or an *optimal solution* to (SCP).

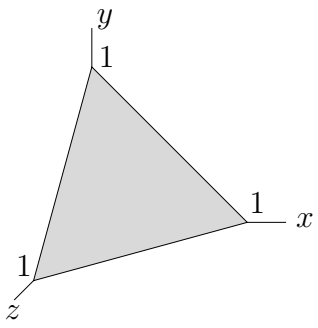

Figure 3.2: The feasible region of (SCP), $m = 3$

The convexity of f lets us characterize the set of optimal solutions to (SCP). Because the feasible region of (SCP) is convex, the following is actually a more general statement.

Lemma 3.1.2 *Let $C \subseteq \mathbb{R}^m$ be a nonempty convex set. $x \in C$ is a minimizer of f over C if and only if for all $x' \in C$,*

$$\nabla f(x)(x' - x) \geq 0. \tag{3.2}$$

Proof. If condition (3.2) holds, Fact 3.1.1 yields for all $x' \in C$

$$f(x') \geq f(x) + \nabla f(x)(x' - x) \geq f(x),$$

so x is a minimizer of f over C. Vice versa, assume that x is a minimizer and choose $x' \in C$. By convexity, we have $x(\lambda) = (1 - \lambda)x + \lambda x' \in C$, $\lambda \in [0, 1]$, and because x is a minimizer of f over C, we obtain

$$\frac{\partial}{\partial \lambda} f(x(\lambda))|_{\lambda=0} := \lim_{\lambda \searrow 0} \frac{f(x(\lambda)) - f(x)}{\lambda} \geq 0. \tag{3.3}$$

On the other hand, the chain rule yields

$$\frac{\partial}{\partial \lambda} f(x(\lambda)) = \frac{\partial}{\partial \lambda} f((1 - \lambda)x + \lambda x') = \nabla f(x(\lambda))(x' - x).$$

With $\lambda = 0$ and (3.3), condition (3.2) follows. □

In our specific case, we can refine the optimality condition (3.2) as follows.

Lemma 3.1.3 *Let $x \in \mathbb{R}^m$ be a feasible solution to (SCP). Then x is an optimal solution to (SCP) if and only if for all $i = 1, \ldots, m$,*

$$\nabla f(x)(\mathbf{e}_i - x) \begin{cases} = 0, & \text{if } x_i > 0, \\ \geq 0 & \text{otherwise}, \end{cases} \tag{3.4}$$

where $\mathbf{e}_i$ is the i-th unit vector.

Proof. If x is optimal, '≥ 0' in (3.4) follows from (3.2), because $x' = \mathbf{e}_i$ is a feasible solution. If $x_i > 0$, there exists $\varepsilon > 0$ such that $x' = (1 - \lambda)x + \lambda \mathbf{e}_i$ is feasible for all $\lambda \in [-\varepsilon, \varepsilon]$. From Lemma 3.1.2, we then get

$$0 \leq \nabla f(x)(x' - x) = \lambda \nabla f(x)(\mathbf{e}_i - x).$$

As this in particular holds for $\lambda = -\varepsilon, \varepsilon$, we must have $\nabla f(x)(\mathbf{e}_i - x) = 0$.

If, on the other hand, (3.4) holds for all i, we get for any other feasible solution x' that

$$\nabla f(x)(x' - x) = \nabla f(x) \left(\sum_{i=1}^{m} x'_i \mathbf{e}_i - x \right) = \sum_{i=1}^{m} x'_i \nabla f(x)(\mathbf{e}_i - x) \geq 0,$$

because all coordinates x'_i of x' are nonnegative. With Lemma 3.1.2, it follows that x is optimal. □

3.2 Smallest enclosing balls

We now show that the problem of finding the smallest enclosing ball of an n-point set $P \subseteq \mathbb{E}^d$ can be formulated in the form of (SCP), with f being a *quadratic* function. In this case, (SCP) is called a *quadratic program*.

Theorem 3.2.1 *Let $P = \{p_1, \ldots, p_n\} \subseteq \mathbb{E}^d$. Let Q be the $(d \times n)$-matrix whose columns are the points of P, i.e.*

$$Q = \begin{pmatrix} p_{11} & \cdots & p_{n1} \\ \vdots & & \vdots \\ p_{1d} & \cdots & p_{nd} \end{pmatrix}. \tag{3.5}$$

Then the following statements hold.

(i) $f : \mathbb{R}^n \to \mathbb{R}$ defined by

$$f(x) = x^T Q^T Q x - \sum_{i=1}^{n} p_i^T p_i x_i$$

is a convex function (Exercise 12). [1]

(ii) Let x^* be any optimal solution to (SCP) with the function f from (i). Then the point

$$c^* = Qx^*$$

is the center c_P of the smallest enclosing ball $B(P)$ of P, and the value $-f(x^*)$ is the squared radius R_P^2 of $B(P)$.

Proof. We only prove (ii) here, assuming (i). We first show that c^* is the center of an enclosing ball with squared radius $-f(x^*)$. Using

$$\nabla f(x) = 2x^T Q^T Q - (p_1^T p_1, \ldots, p_n^T p_n)$$

(you might want to check this), Lemma 3.1.3 yields

$$\begin{aligned} 0 &\leq \nabla f(x^*)(\mathbf{e}_i - x^*) \\ &= \left(2x^{*T} Q^T Q - (p_1^T p_1, \ldots, p_n^T p_n)\right)(\mathbf{e}_i - x^*) \\ &= \left(2c^{*T} Q - (p_1^T p_1, \ldots, p_n^T p_n)\right)(\mathbf{e}_i - x^*) \\ &= \left(2c^{*T} p_i - p_i^T p_i\right) - \left(2c^{*T} c^* - (p_1^T p_1, \ldots, p_n^T p_n)x^*\right) \\ &= \left(2c^{*T} p_i - p_i^T p_i - c^{*T} c^*\right) - \left(c^{*T} c^* - \sum_{i=1}^{n} p_i^T p_i x_i^*\right) \\ &= -(p_i - c^*)^T (p_i - c^*) - f(x^*). \end{aligned} \tag{3.6}$$

[1] For $x, y \in \mathbb{E}^d$, the expression $x^T y$ is just a different way of writing the scalar product $x \cdot y$.

In other words,

$$\|p_i - c^*\|^2 \leq -f(x^*), \quad i = 1, \ldots, n,$$

and this proves that $B_d(c^*, \sqrt{-f(x^*)})$ is a ball containing all points of P. It remains to prove that there is no smaller ball with this property.

Given a potential center $c \neq c^*$, we can uniquely write it in the form

$$c = c^* + \lambda u,$$

where u is some vector of length 1 and $\lambda > 0$. Define $r^2 := -f(x^*)$ and

$$F := \{i \in \{1, \ldots, n\} \mid x_i^* > 0\}.$$

F is nonempty because $\sum_{i=1}^n x_i^* = 1$. With $i \in F$, we get

$$\begin{aligned}(p_i - c)^T(p_i - c) &= (p_i - c^* - \lambda u)^T(p_i - c^* - \lambda u) \\ &= (p_i - c^*)^T(p_i - c^*) + \lambda^2 u^T u - 2\lambda u^T(p_i - c^*) \\ &= r^2 + \lambda^2 - 2\lambda u^T(p_i - c^*),\end{aligned}$$

where the last equality holds because $x_i^* > 0$ implies equality in (3.6) via Lemma 3.1.3.

In order for c to define a ball of squared radius at most r^2 which contains all points—in particular the points $p_i, i \in F$—we must have

$$u^T(p_i - c^*) > 0, \quad i \in F, \tag{3.7}$$

because whenever this fails for some $i \in F$, we get $(p_i - c)^T(p_i - c) > r^2$.

Using $x_i^* > 0, i \in F$, inequality (3.7) then yields

$$\sum_{i \in F} x_i^* u^T(p_i - c^*) > 0.$$

On the other hand, using $\sum_{i \in F} x_i^* = 1$ and $c^* = Qx^* = \sum_{i=1}^n x_i^* p_i = \sum_{i \in F} x_i^* p_i$, we have

$$\sum_{i \in F} x_i^* u^T(p_i - c^*) = u^T \left(\sum_{i \in F} x_i^* p_i - \sum_{i \in F} x_i^* c^* \right) = u^T(c^* - c^*) = 0,$$

a contradiction. This means, c cannot define an enclosing ball of squared radius at most r^2, so c^* is indeed the center of $B(P)$, and $r^2 = -f(x^*)$ is its squared radius. □

The proof actually shows that all points $p_i, i \in F$, are *on the boundary* of $B(P)$, because they satisfy $\|p_i - c^*\|^2 = -f(x^*)$, due to equality in (3.6). Even more is true: the equations

$$c^* = \sum_{i \in F} x_i^* p_i, \quad \sum_{i \in F} x_i^* = 1,$$

along with $x_i^* > 0, i \in F$ show that c^* is a *convex combination* of the $p_i, i \in F$, see Lemma 1.4.1(iii). This means, the solution x^* of the convex program does not only give us $B(P)$, it also provides us with a set of boundary points whose convex hull contains c^*, see Figure 3.3.

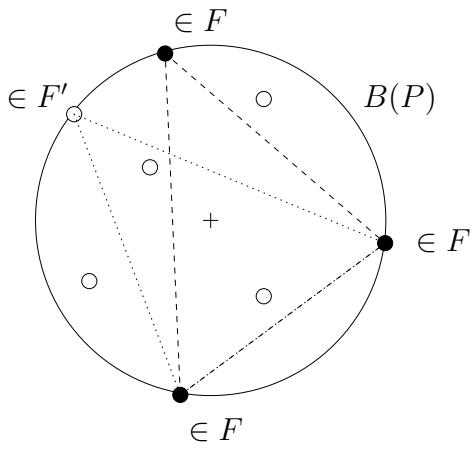

Figure 3.3: P and $B(P)$, with the points $p_i, i \in F$ filled and their convex hull dashed; x^* is not unique in general: there is another optimal solution x'^* defining a set $F' \neq F$ (dotted convex hull)

3.3 Quadratic programming

There are methods for approximately solving (SCP) with a quadratic function efficiently in practice, even for large d and n; these methods are *heuristics*, though, and they do not come with an approximation guarantee. In certain cases, there are fast methods for which some quality of the solution can be guaranteed. This in particular applies to the situation in which the matrix Q from (3.5) is *sparse*, meaning that it has only few nonzero entries. All these methods work for more general problems than we have discussed so far. A general convex quadratic program assumes the form

$$\begin{array}{lll} \text{(QP)} & \text{minimize} & x^T D x + c^T x \\ & \text{subject to} & Ax = b, \\ & & x \geq 0, \end{array}$$

where $D \in \mathbb{R}^{m \times m}$ is a *positive-semidefinite matrix*,[2], $c \in \mathbb{R}^m$, $A \in \mathbb{R}^{k \times m}$ any matrix, and $b \in \mathbb{R}^k$. $x \geq 0$ is a shortcut for $x_i \geq 0, i = 1, \ldots, m$. Optimality conditions similar to the ones in Lemma 3.1.3 exist, and the main reason why quadratic programs are still 'easy' to solve is that the gradient of a quadratic function is linear. General convex programs (CP), where the quadratic function of (QP) is replaced with an arbitrary convex function, still have nice optimality conditions, but these are nonlinear in general, making the problem much harder to solve in practice.

[2]meaning that $x^T D x \geq 0$ for all x

3.4 Polytope Distance

There is another problem which is solvable via the quadratic programming approach. Let $P, P' \subseteq \mathbb{E}^d$ be two point sets. The polytope distance problem is concerned with the computation of the distance between $\mathrm{conv}(P)$ and $\mathrm{conv}(P')$,

$$\delta(P, P') = \min\{\|x - x'\| \mid x \in \mathrm{conv}(P), x' \in \mathrm{conv}(P')\},$$

see Figure 3.4.[3]

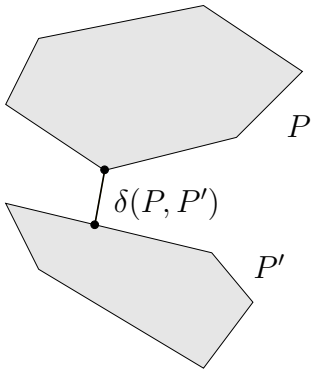

Figure 3.4: The polytope distance problem

If we can compute $\delta(P, P')$, we can in particular decide whether the convex hulls intersect—this is the case if and only if $\delta(P, P') = 0$.

Exercise 14 asks you to write down a formulation of the polytope distance problem in the form of (QP).

3.5 It's all about scalar products

Looking at the problem (SCP) with the function f used for smallest enclosing balls (Theorem 3.2.1), we see that the actual coordinates of the points p_i are never needed. The entries of the matrix $Q^T Q$ are scalar products of the form $p_i \cdot p_j$, and there is an additional linear term in f involving the scalar products $p_i \cdot p_i$. This implies

Corollary 3.5.1 *Let $P \subseteq \mathbb{E}^d$, $|P| = n$. The smallest enclosing ball $B(P)$ can be computed in time*

$$O(dn^2 + g(n)),$$

where $g(n)$ is the time necessary to solve (SCP) with the function f of Theorem 3.2.1.

[3] $\delta(P, P')$ exists, because we are minimizing a continuous function $g : \mathbb{R}^{2d} \to \mathbb{R}$ (which one?) over a compact set $K \subseteq \mathbb{R}^{2d}$ (which one?)

Proof. It takes $O(dn^2)$ time to compute all scalar products, $g(n)$ time to solve (SCP) and another $O(dn)$ time to compute $B(P)$ from the solution, according to Theorem 3.2.1(ii). □

If $n \ll d$, this is a major improvement over other exact methods. Exercise 15 gives a scenario, where the dependence on d can be removed altogether.

3.6 Exercises

Exercise 12 *Prove Theorem 3.2.1 (i).*

Exercise 13 *Use the arguments in the proof of Theorem 3.2.1(ii) to prove Lemma 2.2.2.*

Exercise 14 *Formulate the polytope distance problem in the form of a quadratic program (QP). Can you do it in such a way that the formulation only involves* scalar products *of points from $\mathbb{E}^d$?*

Exercise 15 *The* moment curve *in $\mathbb{E}^d$ is the point set*

$$M_d = \{x \in \mathbb{E}^d \mid x = (t, t^2, \ldots, t^d), t \in \mathbb{R}\}.$$

Let $P \subseteq M_d, |P| = n$. Prove that the radius R_P of $B(P)$ can be computed in time independent from d.