III. Constrained Convex Optimization

So far we have focused exclusively on *unconstrained* optimization problems. In such a setting, our goal is typically clear: find a point where the gradient (or subgradient) is equal to zero. All of the algorithms we have explored so far were different strategies for finding such a point. Once we add constraints, however, things get a bit more complicated. In particular, there may no longer be any points that satisfy the constraints we are imposing where the gradient vanishes. Showing that we have found an optimal point will now involve a more complicated relationship between the gradient of the function we are minimizing together with the constraints.

In these notes we will look at a specific class of constrained optimization problems of the form

$$\underset{\boldsymbol{x} \in \mathbb{R}^N}{\text{minimize}} \; f(\boldsymbol{x}) \quad \text{subject to} \quad g_m(\boldsymbol{x}) \leq 0, \quad m = 1, \ldots, M. \tag{1}$$

Here, we represent the constraints as functions $g_1, \ldots, g_M$, which by convention we define so that we are always imposing $g_m(\boldsymbol{x}) \leq 0$. Note that if we had a constraint of the form $h(\boldsymbol{x}) = 0$ we could write this as $h(\boldsymbol{x}) \leq 0$ combined with $-h(\boldsymbol{x}) \leq 0$, so equality constraints can also be handled, although we will encounter these less frequently in this course.[1]

While some of what we will say actually applies to the case where the g_m are nonconvex, we will mostly only be interested in the case where the g_m are convex functions.

[1] In practice, you would want to handle equality constraints more explicitly, but focusing only on inequality constraints will make the exposition quite a bit cleaner without really sacrificing any intuition.

An important consideration in constrained optimization problems is the concept of *feasibility*. A vector $\boldsymbol{x}$ is **feasible** if it satisfies the constraints of (1). Specifically, a feasible $\boldsymbol{x}$ must satisfy $g_m(\boldsymbol{x}) \leq 0$ for $m = 1, \ldots, M$. It is not a given that any feasible $\boldsymbol{x}$ exists. For instance, I might demand that $\sum_{n=1}^{N} x_n > 1$ and also $\sum_{n=1}^{N} x_n < -1$. Clearly, no $\boldsymbol{x}$ that simultaneously satisfies both of these constraints can exist. In our discussion below, we will assume that the **feasible set**, i.e., the set

$$\mathcal{C} = \{\boldsymbol{x} \in \mathbb{R}^N \, : \, g_m(\boldsymbol{x}) \leq 0 \; m = 1, \ldots, M\},$$

is non-empty.

The Lagrangian

The **Lagrangian** takes the constraints in the program above and integrates them into the objective function. Specifically, the Lagrangian $\mathcal{L} : \mathbb{R}^N \times \mathbb{R}^M \to \mathbb{R}$ associated with (1) is

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{m=1}^{M} \lambda_m g_m(\boldsymbol{x}).$$

For reasons that will become clearer below, the $\boldsymbol{x}$ above are referred to as **primal variables**, and the $\boldsymbol{\lambda}$ as either **dual variables** or **Lagrange multipliers**.

The Lagrangian allows us to transform the *constrained* optimization problem in (1) into an *unconstrained* one. Specifically, consider the problem given by

$$\underset{\boldsymbol{x} \in \mathbb{R}^N}{\text{minimize}} \;\; f(\boldsymbol{x}) + \sum_{m=1}^{M} \lambda_m g_m(\boldsymbol{x}). \tag{2}$$

To get some intuition, suppose that the $\lambda_1, \ldots, \lambda_M$ are very large (positive) numbers. In this case, violating any constraint (allowing $g_m(\boldsymbol{x}) > 0$) will result in a very large penalty being added to the objective function. Thus, by setting the corresponding λ_m to be sufficiently large we can force the constraints to be satisfied.

The problem here is that large values of λ_m not only avoid the setting where $g_m(\boldsymbol{x}) > 0$, but actually encourages $g_m(\boldsymbol{x}) \ll 0$ (since we can potentially benefit by not just satisfying the constraints but by exceeding them by a large margin).

This raises a natural question: can we set $\boldsymbol{\lambda}$ so that the solution to the unconstrained problem (2) is the same as the constrained problem (1)? Here we will provide an answer in the case where the objective function f and the constraints $g_1, \ldots, g_M$ are both convex and differentiable.

Suppose that $\boldsymbol{x}^\star$ is a solution to the constrained problem (1). Then $\boldsymbol{x}^\star$ will also be a solution to (2) if and only if

$$\nabla \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}) = \nabla f(\boldsymbol{x}^\star) + \sum_{m=1}^{M} \lambda_m \nabla g_m(\boldsymbol{x}^\star) = \mathbf{0}. \tag{3}$$

If we knew $\boldsymbol{x}^\star$ already, finding a $\boldsymbol{\lambda}$ that would make the unconstrained and constrained problems equivalent (meaning that they both have the same solution $\boldsymbol{x}^\star$) would just amount to finding a $\boldsymbol{\lambda}$ such that (3) holds. Unfortunately, this might not seem to be particularly useful since $\boldsymbol{x}^\star$ is what we are trying to find to begin with.

To see how we might compute a $\boldsymbol{\lambda}$ that makes the unconstrained and constrained problems equivalent, we will need to begin our first exploration of one of the deepest and most important ideas of optimization: **duality**.

The Lagrange dual function

We can think of the unconstrained optimization problem (2) as actually representing a family of different optimization problems (depending on $\boldsymbol{\lambda}$). For any fixed $\boldsymbol{\lambda}$, imagine solving (2) and computing the minimal value of the objective function – we can think of this as actually defining a function that maps $\boldsymbol{\lambda} \in \mathbb{R}^M$ to $\mathbb{R}$. We call this the **Lagrange dual function** $d(\boldsymbol{\lambda})$, which is defined as

$$d(\boldsymbol{\lambda}) = \min_{\boldsymbol{x} \in \mathbb{R}^N} \left(f(\boldsymbol{x}) + \sum_{m=1}^{M} \lambda_m g_m(\boldsymbol{x}) \right).$$

Note that since the dual is the pointwise infimum of a family of affine functions in $\boldsymbol{\lambda}$, the Lagrange dual function is **always concave**, regardless of whether or not f and the g_m are convex. While we will not stress this much here, this is a remarkable fact and can be very useful when dealing with nonconvex problems.

A key fact about the dual function is that it can provide a lower bound on the optimal value of the original program. In the discussion below, we assume throughout that $\boldsymbol{\lambda} \geq 0$ is arbitrary. Our main claim is that if $p^\star = f(\boldsymbol{x}^\star)$ is the optimal value for (1),[2] then we have

$$d(\boldsymbol{\lambda}) \leq p^\star.$$

This is very easy to show. Specifically, for any feasible point $\boldsymbol{x}'$, we must have $g_m(\boldsymbol{x}') \leq 0$ for all m and hence

$$\sum_{m=1}^{M} \lambda_m g_m(\boldsymbol{x}') \leq 0.$$

[2] We use $p^\star$ instead of $p^\star$ to indicate the optimal value of the *primal* problem, which we will soon be opposing to the optimal value of the *dual* problem.

From this we have that

$$\mathcal{L}(\boldsymbol{x}', \boldsymbol{\lambda}) \leq f(\boldsymbol{x}'),$$

meaning

$$d(\boldsymbol{\lambda}) = \min_{\boldsymbol{x} \in \mathbb{R}^N} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) \leq \mathcal{L}(\boldsymbol{x}', \boldsymbol{\lambda}) \leq f(\boldsymbol{x}').$$

Since this holds for all feasible $\boldsymbol{x}'$, including the minimizer of (1), we have $d(\boldsymbol{\lambda}) \leq p^\star$.

The (Lagrange) dual problem

Given that $d(\boldsymbol{\lambda})$ provides a lower bound on $p^\star$, if you wanted to get an idea of what $p^\star$ looks like (for example, to see if you are close to convergence), it is natural to see how large you can make this lower bound. This gives rise to what we call the **dual problem** of (1):

$$\underset{\boldsymbol{\lambda} \in \mathbb{R}^M}{\text{maximize}} \; d(\boldsymbol{\lambda}) \quad \text{subject to} \quad \boldsymbol{\lambda} \geq \mathbf{0}. \tag{4}$$

The dual optimal value $d^\star$ is

$$d^\star = \max_{\boldsymbol{\lambda} \geq \mathbf{0}} d(\boldsymbol{\lambda}) = \max_{\boldsymbol{\lambda} \geq \mathbf{0}} \min_{\boldsymbol{x} \in \mathbb{R}^N} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}).$$

Since $d(\boldsymbol{\lambda}) \leq p^\star$, we know that

$$d^\star \leq p^\star.$$

The quantity $p^\star - d^\star$ is called the **duality gap**. If $p^\star = d^\star$, then we say that (1) and (4) exhibit **strong duality**.

We will soon discuss when strong duality holds, but first, why is it important? Suppose that $\boldsymbol{x}^\star$ is a solution to the original constrained problem (1) – which we will call the **primal problem** to distinguish it from the dual problem – and suppose that $\boldsymbol{\lambda}^\star$ is a solution to the dual problem (4). It turns out that if we have strong duality, then $\boldsymbol{\lambda}^\star$ is exactly what we need to make $\boldsymbol{x}^\star$ the solution to the unconstrained problem (2).

To see why, note that if we have strong duality then

$$\begin{aligned} f(\boldsymbol{x}^\star) &= d(\boldsymbol{\lambda}^\star) \\ &= \min_{\boldsymbol{x}\in\mathbb{R}^N} \left(f(\boldsymbol{x}) + \sum_{m=1}^{M} \lambda_m^\star g_m(\boldsymbol{x}) \right) \\ &\le f(\boldsymbol{x}^\star) + \sum_{m=1}^{M} \lambda_m^\star g_m(\boldsymbol{x}^\star) \\ &\le f(\boldsymbol{x}^\star). \end{aligned} \tag{5}$$

where the last inequality follows from the fact that we must have $\lambda_m^\star \ge 0$ and $g_m(\boldsymbol{x}^\star) \le 0$. Looking at this entire chain of inequalities, where the first and last term are both $f(\boldsymbol{x}^\star)$, means that

$$f(\boldsymbol{x}^\star) = \min_{\boldsymbol{x}\in\mathbb{R}^N} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}^\star) = \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\lambda}^\star).$$

In words, the solution to the primal problem $\boldsymbol{x}^\star$ is also a minimizer of $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}^\star)$.

Strong duality and the duality gap

So when does strong duality hold? The answer can be pretty complicated and depends on the structure of the constraints. There are a variety of so-called "constraint qualifications" that serve as sufficient conditions to guarantee strong duality.

Probably the simplest is known as **Slater's condition**, which is essentially that the g_m are affine inequality constraints (i.e., they can be expressed as $\boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}$), and that there is an $\boldsymbol{x}$ that is strictly feasible for the remaining constraints (i.e., an $\boldsymbol{x}$ such that for all the g_m which are not affine we have $g_m(\boldsymbol{x}) < 0$). Actually proving that this condition implies strong duality is somewhat involved. We will not worry too much about this, in all of the problems that we will encounter in this course, strong duality will hold.

One other useful fact regarding the duality gap is that it can serve as a way of measuring how far away we are from finding an optimal solution to our optimization problem. To see this recall that any $\boldsymbol{\lambda} \geq \boldsymbol{0}$ gives us a lower bound on $p^\star$, since $d(\boldsymbol{\lambda}) \leq p^\star$. Thus, we know that for any (feasible) $\boldsymbol{x}$ we have

$$f(\boldsymbol{x}) - p^\star \ \leq \ f(\boldsymbol{x}) - d(\boldsymbol{\lambda}).$$

This tells us that

$$p^\star \in [d(\boldsymbol{\lambda}), f(\boldsymbol{x})], \quad \text{and likewise} \quad d^\star \in [d(\boldsymbol{\lambda}), f(\boldsymbol{x})].$$

If we are ever able to reduce this gap to zero, then we know that $\boldsymbol{x}$ is primal optimal, and $\boldsymbol{\lambda}$ is dual optimal, i.e., we have solved both the primal and dual problems. There are certain kinds of "primal-dual" algorithms that exploit this property that we will encounter later in this course.

Example

Consider the optimization problem

$$\underset{\boldsymbol{x}\in\mathbb{R}^N}{\text{minimize}}\ \langle \boldsymbol{x}, \boldsymbol{c}\rangle \quad \text{subject to} \quad \boldsymbol{A}\boldsymbol{x} \le \boldsymbol{b}.$$

This is called a **linear program** (since the objective function is just a linear function of $\boldsymbol{x}$). The Lagrangian is

$$\begin{aligned}\mathcal{L}(\boldsymbol{x},\boldsymbol{\lambda}) &= \langle \boldsymbol{x}, \boldsymbol{c}\rangle + \sum_{m=1}^{M} \lambda_m \left(\langle \boldsymbol{x}, \boldsymbol{a}_m\rangle - b_m\right) \\ &= \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} - \boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{b} + \boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}.\end{aligned}$$

This is a linear function of $\boldsymbol{x}$. Note that it is unbounded below unless

$$\boldsymbol{c} + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{\lambda} = \boldsymbol{0}.$$

Thus

$$\begin{aligned}d(\boldsymbol{\lambda}) &= \min_{\boldsymbol{x}} \left(\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} - \boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{b} + \boldsymbol{\lambda}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}\right) \\ &= \begin{cases} -\langle \boldsymbol{\lambda}, \boldsymbol{b}\rangle, & \boldsymbol{c} + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{\lambda} = \boldsymbol{0} \\ -\infty, & \text{otherwise.}\end{cases}\end{aligned}$$

So the Lagrange dual program is

$$\begin{aligned}\underset{\boldsymbol{\lambda}\in\mathbb{R}^M}{\text{maximize}}\ -\langle \boldsymbol{\lambda}, \boldsymbol{b}\rangle \quad \text{subject to} \quad & \boldsymbol{A}^{\mathrm{T}}\boldsymbol{\lambda} = -\boldsymbol{c} \\ & \boldsymbol{\lambda} \ge \boldsymbol{0}.\end{aligned}$$

Note that the dual is another linear program.