

CS 4220: Numerical Analysis

Constraints and multipliers

David Bindel

2026-04-24

Consider constraints

So far, we have considered *unconstrained* optimization problems. The *constrained* problem is

$$\text{minimize } \phi(x) \text{ s.t. } x \in \Omega$$

where $\Omega \subset \mathbb{R}^n$. We usually define x in terms of a collection of constraint equations and inequalities:

$$\Omega = \{x \in \mathbb{R}^n : c_i(x) = 0, i \in \mathcal{E} \text{ and } c_i(x) \leq 0, i \in \mathcal{I}\}.$$

We will suppose throughout our discussions that both ϕ and all the functions c are differentiable.

If x_* is a solution to the constrained minimization problem, we say constraint $i \in \mathcal{I}$ is *active* if $c_i(x) = 0$. Often, the hard part of solving constrained optimization problems is figuring out which constraints are active. From this perspective, the equality constrained problem sits somewhere in difficulty between the unconstrained problem and the general constrained problem.

Our treatment of constrained optimization is necessarily brief; but in the next two lectures, I hope to lay out some of the big ideas. Today we will focus on formulations; next time, algorithms.

Three recipes

Most methods for constrained optimization involve a reduction to an unconstrained problem (or subproblem). There are three ways such a reduction might work:

- We might *remove* variables by eliminating constraints.

- We might keep the *same* number of variables and try to fold the constraints into the objective function.
- We might *add* variables to enforce constraints via the method of Lagrange multipliers.

These approaches are not mutually exclusive, and indeed one often alternates between perspectives in modern optimization algorithms.

Constraint elimination

The idea of constraint elimination is straightforward. Suppose we want to solve an optimization problem with only equality constraints: $c_i(x) = 0$ for $i \in \mathcal{E}$, where $|\mathcal{E}| < n$ and the constraints are independent – that is, the $|\mathcal{E}| \times n$ Jacobian matrix $\partial c / \partial x$ has full row rank. Then we can think (locally) of x satisfying the constraints in terms of an implicitly defined function $x = g(y)$ for $y \in \mathbb{R}^{n-|\mathcal{E}|}$. If this characterization can be made global, then we can solve the unconstrained problem

$$\text{minimize } \phi(g(y))$$

over all $y \in \mathbb{R}^{n-|\mathcal{E}|}$.

The difficulty with constraint elimination is that it requires that we find a global parameterization of the solutions to the constraint equations. This is usually difficult. An exception is when the constraints are *linear*:

$$c(x) = A^T x - b$$

In this case, the feasible set $\Omega = \{x : A^T x - b = 0\}$ can be written as $x \in \{x^p + z : z \in \mathcal{N}(A)\}$, where x^p is a *particular solution* and $\mathcal{N}(A)$ is the null space of A . We can find both a particular solution and the null space by doing a full QR decomposition on A :

$$A = [Q_1 \quad Q_2] \begin{bmatrix} R_1 \\ 0 \end{bmatrix}.$$

Then solutions to the constraint equations have the form

$$x = A^\dagger b + Q_2 y = Q_1 R_1^{-T} b + Q_2 y$$

where the first term is a particular solution and the second term gives a vector in the null space.

For problems with linear equality constraints, constraint elimination has some attractive properties. If there are many constraints, the problem after constraint elimination may be much smaller. And if the original problem was convex, then so is the reduced problem, and with a better-conditioned Hessian matrix. The main drawback is that we may lose sparsity of the original problem. Constraint elimination is also attractive for solving equality-constrained subproblems in optimization algorithms for problems with linear *inequality* constraints, particularly if those constraints are simple (e.g. elementwise non-negativity of the solution vector).

For problems with more complicated equality constraints, constraint elimination is hard. Moreover, it may not be worthwhile; in some cases, eliminating constraints results in problems that are smaller than the original formulation, but are harder to solve.

The idea of constraint elimination is not limited to equality constraints: one can also sometimes use an alternate parameterization to convert simple inequality-constrained problems to unconstrained problems. For example, if we want to solve a non-negative optimization problem (all $x_i \geq 0$), we might write $x_i = y_i^2$, or possibly $x_i = \exp(y_i)$ (though in this case we would need to let $y_i \rightarrow -\infty$ to exactly hit the constraint). But while they eliminate constraints, these re-parameterizations can also destroy nice features of the original problem (e.g. convexity). So while such transformations are a useful part of the computational arsenal, they should be treated as one tool among many, and not always as the best tool available.

Questions

Using constraint elimination, how would you solve the problem of minimizing $\|Ax - b\|^2$ subject to $\sum_j x_j = 1$?

Penalties and barriers

Constraint elimination methods convert a constrained to an unconstrained problem by changing the coordinate system in which the problem is posed. Penalty and barrier methods accomplish the same reduction to the unconstrained case by changing the function.

As an example of a *penalty* method, consider the problem

$$\text{minimize } \phi(x) + \frac{1}{2\mu} \sum_{i \in \mathcal{E}} c_i(x)^2 + \frac{1}{2\mu} \sum_{i \in \mathcal{J}} \max(c_i(x), 0)^2.$$

When the constraints are violated ($c_i > 0$ for inequality constraints and $c_i \neq 0$ for equality constraints), the extra terms (penalty terms) beyond the original objective function are positive; and as $\mu \rightarrow 0$, those penalty terms come to dominate the behavior outside the feasible region. Hence as we let $\mu \rightarrow 0$, the solutions to the penalized problem approach solutions to the original (true) problem. At the same time, as $\mu \rightarrow 0$ we have much wilder derivatives of ϕ , and the optimization problems become more and more problematic from the perspective of conditioning and numerical stability. Penalty methods also have the potentially undesirable property that if any constraints are active at the true solution, the solutions to the penalty problem tend to converge from *outside* the feasible region. This poses a significant problem if, for example, the original objective function ϕ is undefined outside the feasible region.

As an example of a *barrier* method, consider the purely inequality constrained case, and approximate the original constrained problem by the unconstrained problem

$$\text{minimize } \phi(x) - \mu \sum_{i \in \mathcal{J}} \log(-c_i(x)).$$

As $c_i(x)$ approaches zero from below, the barrier term $-\mu \log(-c_i(x))$ grows rapidly; but at any fixed x in the interior of the domain, the barrier goes to zero as μ goes to zero. Hence, as $\mu \rightarrow 0$ through positive values, the solution to the barrier problem approaches the solution to the true problem through a sequence of *feasible* points (i.e. approximate solutions that satisfy the constraints). Though the feasibility of the approximations is an advantage over penalty based formulations, interior formulations share with penalty formulations the disadvantage that the solutions for $\mu > 0$ lie at points with increasingly large derivatives (and bad conditioning) if the true solution has active constraints.

There are *exact penalty* formulations for which the solution to the penalized problem is an exact solution for the original problem. Suppose we have an inequality constrained problem in which the feasible region is closed and bounded, each constraint c_i has continuous derivatives, and $\nabla c_i(x) \neq 0$ at any boundary point x where constraint i is active. Then the solution to the problem

$$\text{minimize } \phi(x) + \frac{1}{\mu} \sum_i \max(c_i(x), 0)$$

is *exactly* the solution to the original constrained optimization problem for some $\mu > 0$. In this case, we used a *nondifferentiable* exact penalty, but there are also exact differentiable penalties.

Questions

How might you approximate the problem of minimizing $\|Ax - b\|^2$ subject to $\sum_j x_j = 1$ via a penalty formulation?

Lagrange multipliers

Picture a function $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$; if you'd like to be concrete, let $n = 2$. Absent a computer, we might optimize ϕ by the physical experiment of dropping a tiny ball onto the surface and watching it roll downhill (in the steepest descent direction) until it reaches the minimum. If we wanted to solve a constrained minimization problem, we could build a great wall between the feasible and the infeasible region. A ball rolling into the wall would still roll freely in directions tangent to the wall (or away from the wall) if those directions were downhill; at a constrained minimizer, the force pulling the ball downhill would be perfectly balanced against an opposing force pushing into the feasible region in the direction of the normal to the wall. If the feasible region is $\{x : c(x) \leq 0\}$, the normal direction pointing inward at a boundary point

x_* s.t. $c(x_*) = 0$ is proportional to $-\nabla c(x_*)$. Hence, if x_* is a constrained minimum, we expect the sum of the “rolling downhill” force ($-\nabla\phi$) and something proportional to $-\nabla c(x_*)$ to be zero:

$$-\nabla\phi(x_*) - \mu\nabla c(x_*) = 0.$$

The *Lagrange multiplier* μ in this picture represents the magnitude of the restoring force from the wall balancing the tendency to roll downhill.

More abstractly, and more generally, suppose that we have a mix of equality and inequality constraints. We define the *Lagrangian*

$$L(x, \lambda, \mu) = \phi(x) + \sum_{i \in \mathcal{E}} \lambda_i c_i(x) + \sum_{i \in \mathcal{J}} \mu_i c_i(x).$$

The *Karush-Kuhn-Tucker (KKT) conditions* for x_* to be a constrained minimizer are

$$\begin{aligned} \nabla_x L(x_*) &= 0 \\ c_i(x_*) &= 0, \quad i \in \mathcal{E} && \text{equality constraints} \\ c_i(x_*) &\leq 0, \quad i \in \mathcal{J} && \text{inequality constraints} \\ \mu_i &\geq 0, \quad i \in \mathcal{J} && \text{non-negativity of multipliers} \\ c_i(x_*)\mu_i &= 0, \quad i \in \mathcal{J} && \text{complementary slackness} \end{aligned}$$

where the (negative of) the “total force” at x_* is

$$\nabla_x L(x_*) = \nabla\phi(x_*) + \sum_{i \in \mathcal{E}} \lambda_i \nabla c_i(x_*) + \sum_{i \in \mathcal{J}} \mu_i \nabla c_i(x_*).$$

The complementary slackness condition corresponds to the idea that a multiplier should be nonzero only if the corresponding constraint is active (a “restoring force” is only present if our test ball is pushed into a wall).

Like the critical point equation in the unconstrained case, the KKT conditions define a set of (necessary but not sufficient) nonlinear algebraic equations that must be satisfied at a minimizer. Because of the multipliers, we have *more* variables than were present in the original problem. However, the Jacobian matrix (KKT matrix)

$$J = \begin{bmatrix} H_\phi(x_*) & \nabla c \\ (\nabla c)^T & 0 \end{bmatrix}$$

has a saddle point structure even when $H_\phi(x_*)$ is positive definite. Also, unlike the penalty and barrier approaches described before, the Lagrange multiplier approach requires that we figure out which multipliers are active or not — an approach that seems to lead to a combinatorial search in the worst case.

Questions

How would you solve the problem of minimizing $\|Ax - b\|^2$ subject to $\sum_j x_j = 1$ via the method of Lagrange multipliers?