Nonlinear systems of equations

We saw earlier how to solve large systems of linear equations: collect them into a single matrix equation, and use an algorithm like Gaussian elimination to construct and solve a factorization.

We also saw how to make a linear approximation to a nonlinear function: if $f \in \mathbb{R}^n \to \mathbb{R}^n$, then we can get a first-order Taylor approximation by calculating the differential,

$$df(x) = f'(x)dx$$

These two tools can work together: suppose that we want to solve a nonlinear system of equations

$$f(x) = 0$$

If we start from a guess x_1 at a solution, we can construct a first-order Taylor expansion

$$df = f'(x_1)dx$$

Holding x_1 fixed, this is a *linear* equation for df in terms of dx. So we can ask to find dx that makes $f(x_1) + df = 0$ — that is, we can solve a linear approximation to the original nonlinear equations. As before, Gaussian elimination or other factorizations can solve this linear system quickly and reliably.

> *If there are multiple solutions — i.e., if f' is singular — we need to pick one; methods for doing so are beyond the scope of these notes.*

With the solution dx in hand, we can construct a new guess

$$x_2 = x_1 + dx$$

We can then make a new Taylor expansion around x_2, leading to a new linear approximation

$$f(x_2) + df = 0 \qquad df = f'(x_2)dx$$

Repeating the process lets us construct x_3, x_4, and so forth. Hopefully each successive x_t comes closer to satisfying $f(x_t) = 0$.

This process is called *Newton's method*, and it often converges rapidly to a solution of the nonlinear system $f(x) = 0$. In fact, the stationary points of Newton's method are strongly related to the solutions: if $dx = 0$ then the second equation implies $df = 0$, so the first equation implies $f(x) = 0$. In the other direction, if $f(x) = 0$ then the first equation constrains $df = 0$. Then if $f'(x)$ is not singular, we have to have $dx = 0$. (If $f'(x)$ is singular, we might be at a stationary point without solving $f(x) = 0$.)

If Newton's method fails, sometimes we can rescue it by *damping*, i.e., decreasing our step size: that is, we set $x_{t+1} = x_t + \alpha_t\, dx$ for some $\alpha_t \in (0, 1)$. But tuning the step size (and other methods beyond damped Newton) are beyond the scope of this set of notes.

Example

Let $f(x) = e^x - 1$, so that $df = e^x dx$. The solution to $f(x) = 0$ is $x = 0$, but let's see if we can find this by Newton's method, starting from somewhere else.

x	f	df	Equation	dx
1	$e - 1$	e	$e\, dx = 1 - e$	$\frac{1-e}{e}$
-0.632	-0.468	0.532	$0.532\, dx = 0.468$	0.880
0.248	0.281	1.281	$1.281\, dx = -0.281$	-0.219

Quite rapidly we have reached $x = 0.029$, very close to the true solution.

Unconstrained optimization

Solving optimization problems is strongly related to solving systems of equations. In an unconstrained optimization problem

$$\min_\theta L(\theta)$$

we can try to find the solution by looking for a critical point: a place where, locally, changes to θ do not change $L(\theta)$.

> *Critical points can be minima or maxima, and they can be either local or global. In addition, they can be neither: they can be places where the function flattens out temporarily, or places where it looks like a saddle. For now, we won't be concerned with checking which is which.*

To find a critical point, we can look at the first order Taylor expansion of L:

$$dL = L'(\theta)\, d\theta$$

At a critical point, all possible changes $d\theta$ should leave $dL = 0$. That means we must have

$$L'(\theta) = 0$$

These equations are the *first-order optimality conditions* for $L(\theta)$. Geometrically, the Taylor expansion is flat (constant):

Of course, the system of equations $L'(\theta) = 0$ could be nonlinear. So, we can apply Newton's method — that is, we can set a first-order Taylor approximation of L' to zero and solve for $d\theta$:

$$L'(\theta) + dL' = 0 \qquad dL' = L''(\theta)d\theta$$

We can find L' and L'' by differentiating L twice.

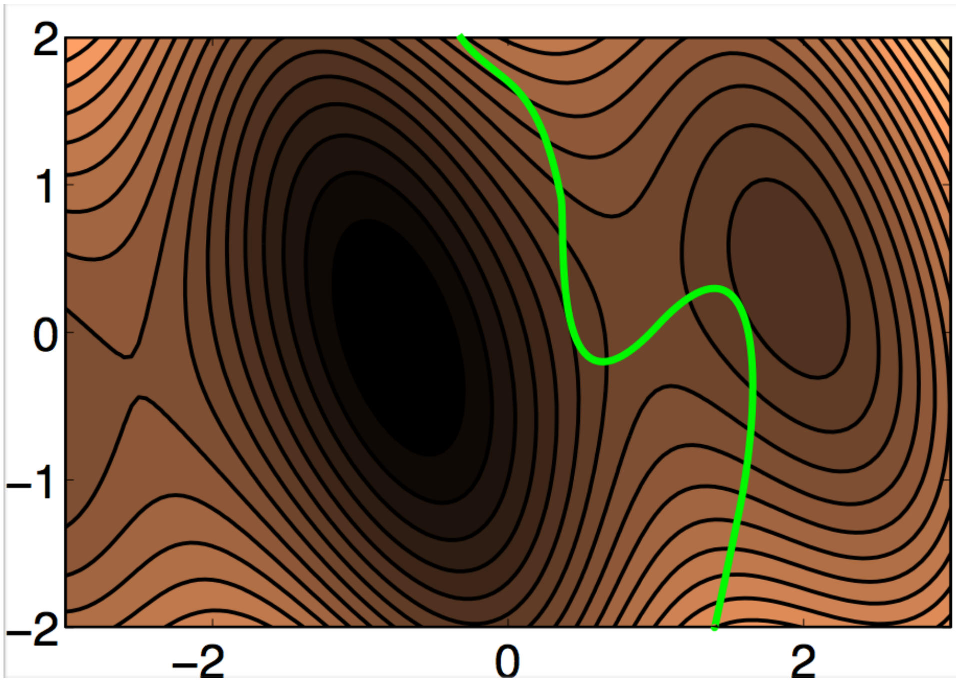
This is such a common application of Newton's method that it shares the same name. If necessary, we can distinguish by calling the two algorithms "Newton's method for solving a system of equations" and "Newton's method for optimizing a function".

Constrained optimization

In a constrained problem

$$\min_{\theta} L(\theta) \quad \text{s.t.} \quad g(\theta) = 0$$

we don't need $L'(\theta)$ to be zero: it's OK if there's a direction of decrease in $L(\theta)$ as long the constraint prevents us from moving in this direction.



To encode this condition, we need to be a bit clever. First note that the solutions to the following problem

$$\min_\theta \left[L(\theta) + \alpha g(\theta)\right] \quad \text{s.t.} \quad g(\theta) = 0$$

are the same as the solutions to our original problem, no matter what the value of α is, since $\alpha g(\theta) = 0$ for any feasible θ.

Then note that, by choosing α appropriately, we can rule out any direction of decrease in L that doesn't satisfy the constraint: if L would decrease on the side of the constraint where $g(\theta) > 0$, then we choose α to be very positive, so that any motion in this direction would cause $L(\theta) + \alpha g(\theta)$ to increase instead of decreasing. Similarly, if L would decrease on the other side if the constraint, where $g(\theta) < 0$, we choose α to be very negative.

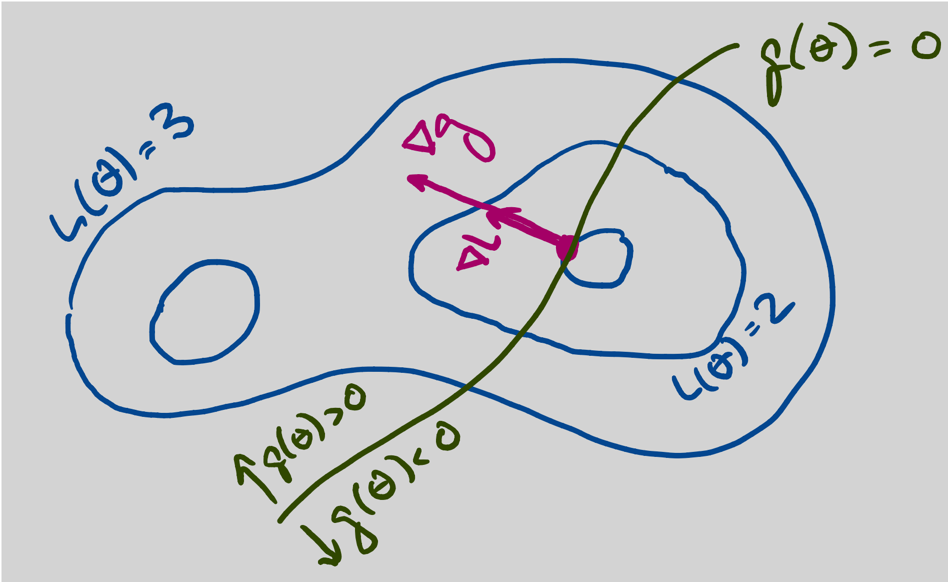
Given this new objective, we can use a Taylor expansion the same way as before. We ask for a critical point: a θ where, to first order, the objective doesn't change as we change $d\theta$. That is,

$$0 = d(L(\theta) + \alpha g(\theta)) = L'(\theta)d\theta + \alpha g'(\theta)d\theta$$

which implies

$$L'(\theta) + \alpha g'(\theta) = 0$$

Geometrically, this equation tells us that at a critical point we can only change $L(\theta)$ by changing θ in a direction *orthogonal* to the constraint (parallel to $g'(\theta)$): sliding in any direction along the constraint doesn't change L, at least to first order.



Interestingly, that means that we didn't have to choose α a priori: any θ and α that satisfy

$$g(\theta) = 0 \qquad L'(\theta) + \alpha g'(\theta) = 0$$

will represent a critical point. So, as before, we've turned our optimization problem into a possibly-nonlinear system of equations. We can solve this system with Newton's method or any other appropriate tool.

The new variable α is called a *Lagrange multiplier* or *dual variable*. We can interpret $-L'(\theta)$ as a force that wants to push our current point θ downhill, toward a minimum of L. We can then think of $-\alpha g'(\theta)$ as a force that pushes back, keeping θ from violating the constraint. At the solution, the two forces balance exactly.

By introducing the dual variable, we've transformed our optimization problem into a system of simultaneous equations, where the objective and the constraints are treated the same way. This transformation was what let us apply Newton's method.

Practice: solve the following problem by introducing a Lagrange multiplier.

$$\min_{x,y} \frac{1}{2}(x^2 + y^2) \quad \text{s.t.} \quad x + 2y = 1$$

Multiple constraints

Suppose we have more than one constraint:

$$\min_{\theta} L(\theta) \quad \text{s.t.} \quad g(\theta) = 0$$

where the output of $g(\theta)$ is in $\mathbb{R}^d$ instead of $\mathbb{R}$. The solution in this case is almost identical: we can still solve

$$g(\theta) = 0 \qquad L'(\theta) + \alpha g'(\theta) = 0$$

But now, instead of $\alpha \in \mathbb{R}$, we need $\alpha \in \mathbb{R}^{1 \times d}$, so that $\alpha g'(\theta)$ is the same shape as $L'(\theta)$.

Each coordinate α_i is still called a Lagrange multiplier. The geometric interpretation is only slightly different from before: we think of each α_i as controlling a separate force, in a direction that's normal to the corresponding constraint $g_i(\theta) = 0$. At equilibrium, all of the forces $\alpha_i g_i'(\theta)$ combine to cancel out $L'(\theta)$.