Week 8: Wednesday, Mar 16

Problem du jour

Suppose $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $f(x_k) \neq 0$, and $f'(x_k)$ is invertible. If u is the Newton direction $u = -f'(x_k)^{-1} f(x_k)$, show

$$\left. \frac{\partial \|f(x)\|^2}{\partial u} \right|_{x=x_k} < 0.$$

Answer: First, note that

$$\frac{\partial f(x)^T f(x)}{\partial u} = 2 f(x)^T \frac{\partial f(x)}{\partial u}.$$

By the chain rule and the definition of u, we have

$$\left. \frac{\partial f(x)}{\partial u} \right|_{x=x_k} = f'(x_k) u. = f'(x_k) \left(-f'(x_k)^{-1} f(x_k) \right) = f(x_k).$$

Therefore,

$$\left. \frac{\partial \|f(x)\|^2}{\partial u} \right|_{x=x_k} = -2 \|f(x_k)\|^2 < 0.$$

The multi-dimensional case

As in the one-dimensional case, there are several different options for finding the minimum of a function g depending on several variables, depending on how many derivatives of g we are willing to compute. These include:

1. Guarded versions of Newton if we are willing to compute Hessians.
2. Modified Newton variants if we are willing to get some second derivative information but it is too expensive to compute and solve with an exact Hessian at every step.
3. Steepest descent and coordinate descent methods if we are willing to compute gradients but do not want to approximate Hessians.
4. Direct search methods (such as Nelder-Mead) when we are only willing to compare the magnitudes of function values.

Going downhill

Perhaps the simplest unconstrained optimization algorithm around is *gradient descent* (sometimes also called *steepest descent*):

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k),$$

where α_k is chosen by some line search procedure. Note that

$$\begin{aligned} f(x_{k+1}) &= f(x_k) + f'(x_k)(-\alpha_k \nabla f(x_k)) + O(\alpha_k)^2 \\ &= f(x_k) - \alpha_k \|f'(x_k)\|^2 + O(\alpha_k^2). \end{aligned}$$

Therefore, if α_k is small enough (and x_k is not a stationary point), each step of gradient step will make some progress in decreasing the function value. Unfortunately, gradient descent can be agonizingly slow.

If $f : \mathbb{R}^n \rightarrow \mathbb{R}$ has two continuous derivatives, we know that any local minimizer x_* is a stationary point ($\nabla f(x_*) = 0$). If we have a good guess at a local minimizer, therefore, we can simply use Newton iteration with line search to solve the system of equations $\nabla f(x_*) = 0$:

$$x_{k+1} = x_k - \alpha_k H_f(x_k)^{-1} \nabla f(x_k).$$

Alas, there is a problem with Newton iteration that doesn't occur with gradient descent: maxima and saddle points are also stationary points!

Both Newton iteration and gradient descent have the form

$$x_{k+1} = x_k - \alpha_k u_k$$

for some search direction u_k. When will such an iteration actually decrease the value of f? Using Taylor expansion at x_k, we have

$$f(x_{k+1}) = x_k - \alpha_k f'(x_k) u_k + O(\alpha_k^2),$$

so a reasonable requirement is that $f'(x_k)u_k > 0$ (i.e. u_k forms an acute angle to the gradient vector). That is, u_k should be a *descent* direction.

The picture here is similar to the picture we saw in one dimension. If $H_f(x_k)$ is positive definite, then so is $H_f(x_k)^{-1}$, and

$$f'(x_k)u_k = f'(x_k) H_f(x_k)^{-1} f'(x_k) > 0.$$

If $H_f(x_k)$ is *not* positive definite, then we want to consider something other than the Newton direction. A standard trick is to form a modified Hessian matrix $\hat{H}_k$ that is changed just enough from $H_f(x_k)$ to be positive definite. This can be done by adding a multiple of the identity, for example, or by fixing a Cholesky factorization on the fly.

Step direction and step size

Forming Hessians is a pain. What would happen if we just stuck to old-fashioned gradient descent with a line search strategy? Let's look at a model problem:

$$\phi(x) = \frac{1}{2}x^T Ax$$

Note that $\nabla\phi = Ax$ (Ivo did this in section), and gradient descent with a line search is

$$x_{k+1} = (I - \alpha_k A)x_k.$$

Notice that

$$\|x_{k+1}\| \leq \|I - \alpha_k A\|\|x_k\|.$$

Using the fact that A is a symmetric matrix, we have

$$\|I - \alpha_k A\| = \max\{|1 - \alpha_k\lambda_1|, |1 - \alpha_k, \lambda_n|\},$$

where λ_1 and λ_n are the largest and smallest eigenvalues of A, respectively. The value of α_k that makes this number smallest is

$$\alpha_* = \frac{2}{\lambda_1 + \lambda_n},$$

which yields

$$\frac{\|x_{k+1}\|}{\|x_k\|} \leq \frac{\lambda_1 - \lambda_n}{\lambda_1 + \lambda_n} = \frac{\kappa(A) - 1}{\kappa(A) + 1}.$$

This bound reflects what we actually see in practice with either a fixed step size or a very inexact line search method. For optima with very ill-conditioned Hessians, corresponding to a long, shallow "bowl" in space, gradient descent tends to be slow.

What if we used a line minimization strategy? One can show (though we won't) that even in this case, we have fairly slow descent of the objective function value:

$$\frac{\phi(x_{k+1})}{\phi(x_k)} \leq \left(\frac{\kappa(A) - 1}{\kappa(A) + 1}\right)^2,$$

and there exist starting points x_0 such that this bound is sharp.

What if we have some estimate D of the inverse of the Hessian A? In this case, the (scaled) gradient descent iteration is

$$x_{k+1} = (I - \alpha_k D^{-1}A)x_k,$$

and we essentially replace $\kappa(A)$ with $\kappa(D^{-1}A)$ in all the above bounds. And, of course, we get convergence in a single step if $D^{-1} = A^{-1}$.

It looks, therefore, as though it might be worth getting some estimates on the behavior of the Hessian matrix whenever possible.