10-425/625: Introduction to Convex Optimization (Fall 2023)

Lecture 16: Newton's Method, Log-Barrier Method

Instructor:[1] *Matt Gormley* *October 25, 2023*

Today we will begin by studying our first second-order optimization algorithm: Newton's method. Although this first algorithm doesn't relate to the ideas of duality that we just studied, we will revisit them as we explore interior-point methods. The other algorithm we will briefly look at today is the Log-Barrier Method. The particular formulation we'll discuss will only consider the primal (not the dual), but other primal-dual versions of this algorithm exist as well.

16.1 Newton's Method

16.1.1 The Algorithm

Newton's method Given unconstrained, smooth convex optimization

$$\min_x \ f(x)$$

where f is convex, twice differentable, and $\text{dom}(f) = \mathbb{R}^n$. Recall that gradient descent chooses initial $x^{(0)} \in \mathbb{R}^n$, and repeats

$$x^{(k)} = x^{(k-1)} - t_k \cdot \nabla f(x^{(k-1)}), \quad k = 1, 2, 3, \ldots$$

In comparison, Newton's method repeats

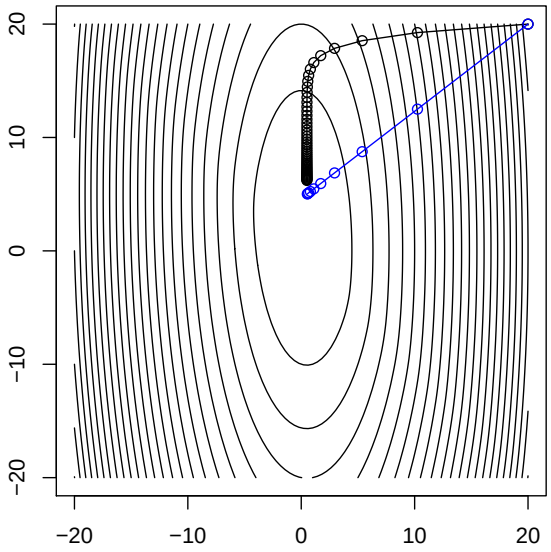
$$x^{(k)} = x^{(k-1)} - \left(\nabla^2 f(x^{(k-1)})\right)^{-1} \nabla f(x^{(k-1)}), \quad k = 1, 2, 3, \ldots$$

Here $\nabla^2 f(x^{(k-1)})$ is the Hessian matrix of f at $x^{(k-1)}$.

Example: Newton's Method vs. Gradient Descent Consider minimizing $f(x) = (10x_1^2 + x_2^2)/2 + 5\log(1 + e^{-x_1 - x_2})$ (this must be a nonquadratic ... why?)

[1] These notes were originally written by Ryan Tibshirani for 10-725 Fall 2019 (original version: here) and were edited and adapted for 10-425/625.

We compare gradient descent (black) to Newton's method (blue), where both take steps of roughly same length



Notice that gradient descent moves in a direction that is orthogonal to the contour lines of the function. By contrast, Newton's method chooses a direction that moves to the minimum much more quickly.

16.1.2 Newton's method interpretation

Minimizing a quadratic local approximation Recall the motivation for gradient descent step at x: we minimize the quadratic approximation

$$f(y) \approx f(x) + \nabla f(x)^T (y - x) + \frac{1}{2t}\|y - x\|_2^2$$

over y, and this yields the update $x^+ = x - t\nabla f(x)$. This comes from approximating the Hessian as $\frac{1}{t}I$ in a second-order Taylor series approximation (i.e. $(y-x)^T \frac{1}{t} I (y-x) = \frac{1}{t}\|y-x\|_2^2$.

Newton's method uses in a sense a better quadratic approximation

$$f(y) \approx f(x) + \nabla f(x)^T (y - x) + \frac{1}{2}(y - x)^T \nabla^2 f(x)(y - x)$$

and minimizes over y to yield $x^+ = x - (\nabla^2 f(x))^{-1}\nabla f(x)$. This is exactly the second-order Taylor series approximation that we've seen before.

Minimizing this second quadratic yields the Newton's method update.

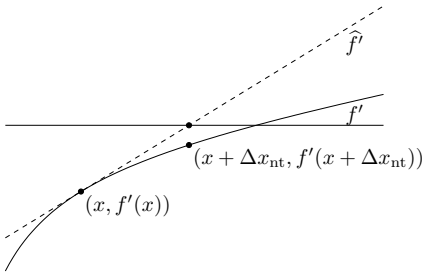
$$\begin{aligned} Q(y) &= f(x) + \nabla_x f(x)^T (y-x) + \frac{1}{2}(y-x)^T \nabla_x^2 f(x)(y-x) \\ \nabla_y Q(y) &= \nabla_x f(x) + \nabla_x^2 f(x)(y-x) \\ \nabla_y Q(y) = 0 &\Rightarrow y = x - (\nabla_x^2 f(x))^{-1} \nabla f(x) \end{aligned}$$

Linearized optimality condition Newton's method was originally invented to solve a system of nonlinear equations $F(x) = 0$ (polynomial equations). The idea was to repeatedly update the variables until convergence.

Aternative interpretation of Newton step at x: we seek a direction v so that $\nabla f(x+v) = 0$. Let $F(x) = \nabla f(x)$. Consider linearizing F around x, via a first-order approximation:

$$0 = F(x+v) \approx F(x) + F'(x)v$$

Solving for v yields $v = -(F'(x))^{-1}F(x) = -(\nabla^2 f(x))^{-1}\nabla f(x)$.



(From B & V page 486)

History: work of Newton (1685) and Raphson (1690) originally focused on finding roots of polynomials. Simpson (1740) applied this idea to general nonlinear equations, and minimization by setting the gradient to zero

Affine invariance of Newton's method Important property Newton's method: affine invariance. Given f, nonsingular $A \in \mathbb{R}^{n \times n}$. Let $x = Ay$, and $g(y) = f(Ay)$. Newton steps on g are

$$\begin{aligned} y^+ &= y - \big(\nabla^2 g(y)\big)^{-1} \nabla g(y) \\ &= y - \big(A^T \nabla^2 f(Ay) A\big)^{-1} A^T \nabla f(Ay) \\ &= y - A^{-1} \big(\nabla^2 f(Ay)\big)^{-1} \nabla f(Ay) \end{aligned}$$

Hence

$$Ay^+ = Ay - \big(\nabla^2 f(Ay)\big)^{-1} \nabla f(Ay)$$

i.e.,

$$x^+ = x - \big(\nabla^2 f(x)\big)^{-1} \nabla f(x)$$

So progress is independent of problem scaling. This is not true of gradient descent!

16.1.3 Damped Newton's method

Backtracking line search So far we've seen pure Newton's method. This need not converge. In practice, we use damped Newton's method (typically just called Newton's method), which repeats

$$x^+ = x - t\big(\nabla^2 f(x)\big)^{-1} \nabla f(x)$$

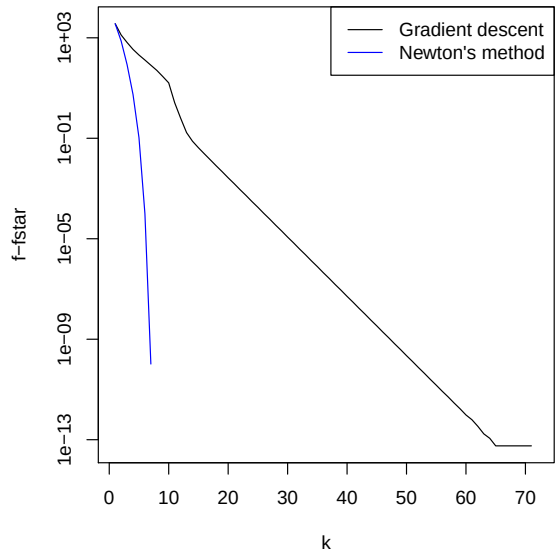
Note that the pure method uses $t = 1$

Step sizes here are chosen by backtracking search, with parameters $0 < \alpha \leq 1/2$, $0 < \beta < 1$. At each iteration, start with $t = 1$, while

$$f(x + tv) > f(x) + \alpha t \nabla f(x)^T v$$

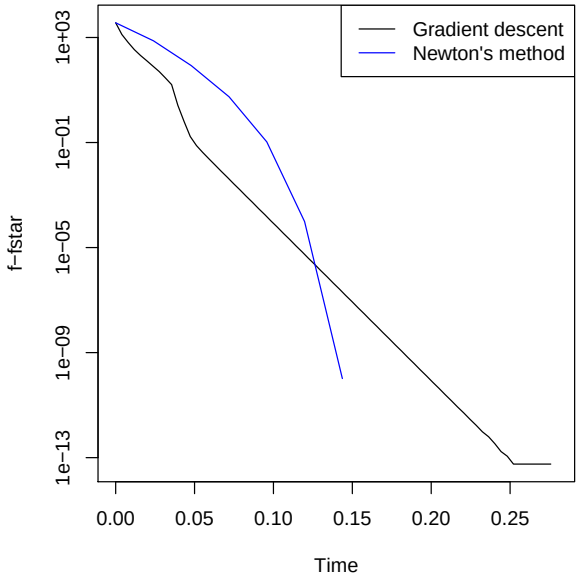
we shrink $t = \beta t$, else we perform the Newton update. Note that here $v = -(\nabla^2 f(x))^{-1} \nabla f(x)$, so $\nabla f(x)^T v = -\lambda^2(x)$

Example: logistic regression Logistic regression example, with $n = 500$, $p = 100$: we compare gradient descent and Newton's method, both with backtracking



Newton's method: in a totally different regime of convergence...!

Back to logistic regression example: now x-axis is parametrized in terms of time taken per iteration



Each gradient descent step is $O(p)$, but each Newton step is $O(p^3)$

16.2 Log-Barrier Method

16.2.1 The log-barrier function

Log barrier function Consider the convex optimization problem

$$\begin{array}{ll} \min\limits_{x} & f(x) \\ \text{subject to} & h_i(x) \leq 0,\ i = 1, \ldots, m \\ & Ax = b \end{array}$$

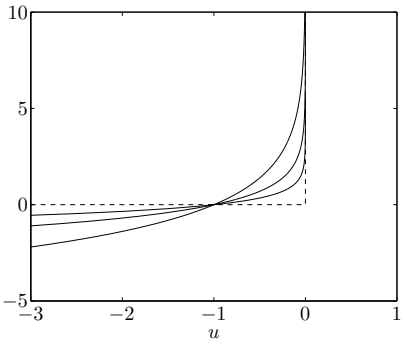
We will assume that f, $h_1, \ldots, h_m$ are convex, twice differentiable, each with domain $\mathbb{R}^n$. The function

$$\phi(x) = -\sum_{i=1}^{m} \log(-h_i(x))$$

is called the log barrier for the above problem. Its domain is the set of strictly feasible points, $\{x : h_i(x) < 0,\ i = 1, \ldots, m\}$, which we assume is nonempty. (Note this implies strong duality holds)

Ignoring equality constraints for now, our problem can be written as

$$\min_{x}\ f(x) + \sum_{i=1}^{m} I_{\{h_i(x) \leq 0\}}(x)$$



We can approximate the sum of indicators by the log barrier:

$$\min_{x}\ f(x) - \frac{1}{t}\sum_{i=1}^{m} \log(-h_i(x))$$

where $t > 0$ is a large number

This approximation is more accurate for larger t. But for any value of t, the log barrier approaches ∞ if any $h_i(x) \to 0$

Log barrier calculus

For the log barrier function

$$\phi(x) = -\sum_{i=1}^{m} \log(-h_i(x))$$

we have for its gradient:

$$\nabla\phi(x) = -\sum_{i=1}^{m} \frac{1}{h_i(x)} \nabla h_i(x)$$

and for its Hessian:

$$\nabla^2\phi(x) = \sum_{i=1}^{m} \frac{1}{h_i(x)^2} \nabla h_i(x) \nabla h_i(x)^T - \sum_{i=1}^{m} \frac{1}{h_i(x)} \nabla^2 h_i(x)$$

16.2.2 The Algorithm

Barrier method The barrier method solves a sequence of problems

$$\begin{array}{ll} \min\limits_{x} & tf(x) + \phi(x) \\ \text{subject to} & Ax = b \end{array}$$

for increasing values of $t > 0$, until duality gap satisfies $m/t \leq \epsilon$. We fix $t^{(0)} > 0$, $\mu > 1$. We use Newton to compute $x^{(0)} = x^\star(t)$, solution to barrier problem at $t = t^{(0)}$. For $k = 1, 2, 3, \ldots$

- Solve the barrier problem at $t = t^{(k)}$, using Newton initialized at $x^{(k-1)}$, to yield $x^{(k)} = x^\star(t)$
- Stop if $m/t \leq \epsilon$, else update $t^{(k+1)} = \mu t$

The first step above is called a centering step (since it brings $x^{(k)}$ onto the central path)

Considerations:

- Choice of μ: if μ is too small, then many outer iterations might be needed; if μ is too big, then Newton's method (each centering step) might take many iterations

- Choice of $t^{(0)}$: if $t^{(0)}$ is too small, then many outer iterations might be needed; if $t^{(0)}$ is too big, then the first Newton solve (first centering step) might require many iterations

Fortunately, the performance of the barrier method is often quite robust to the choice of μ and $t^{(0)}$ in practice

(However, note that the appropriate range for these parameters is scale dependent)

Example of a small LP in $n = 50$ dimensions, $m = 100$ inequality constraints (from B & V page 571):