MATH3027: Optimization 2022

Week 5: The Gradient Method (II)

Prof. Richard Wilkinson
Please send any comments or mistakes to r.d.wilkinson@nottingham.ac.uk

This week we conclude our study of gradient descent type methods. We review the Gauss-Newton method, a variant that is specifically meant for nonlinear least squares problems, and Weiszfeld's method for solving optimal location problems.

We also look at a different class of techniques for unconstrained optimization stemming from Newton's method, and study its convergence and differences with gradient descent. There is less material this week to give you time to complete the coursework.

The two methods we consider in this chapter come from considering different approximations of $f(\mathbf{x})$. In the first case, we consider the linear approximation

$$f(\mathbf{x}') \approx f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{x} - \mathbf{x}'),$$

which leads to the Gauss-Newton method. In the second case, we consider the quadratic approximation

$$f(\mathbf{x}') \approx f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{x} - \mathbf{x}') + \frac{1}{2}(\mathbf{x} - \mathbf{x}')^\top \nabla f(\mathbf{x})^\top (\mathbf{x} - \mathbf{x}')$$

which leads us to Newton's method.

The Gauss-Newton Method

We use the gradient method from last week to build an algorithm for solving nonlinear least squares problem of the type:

$$\text{(NLS):} \quad \min_{\mathbf{x}\in\mathbb{R}^n} \left\{ g(\mathbf{x}) \equiv \sum_{i=1}^{m} (f_i(\mathbf{x}) - c_i)^2 \right\}$$

$f_1, \ldots, f_m$ are continuously differentiable over $\mathbb{R}^n$ and $c_1, \ldots, c_m \in \mathbb{R}$.

Denote:

$$F(\mathbf{x}) = \begin{pmatrix} f_1(\mathbf{x}) - c_1 \\ f_2(\mathbf{x}) - c_2 \\ \vdots \\ f_m(\mathbf{x}) - c_m \end{pmatrix}$$

Then the problem becomes:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|F(\mathbf{x})\|^2$$

The Gauss-Newton method is designed to solve this nonlinear least squares problem. It is an iterative approach like gradient descent, and generates a sequence $\mathbf{x}^k$ for $k = 1, \ldots$ that will hopefully converge to a stationary point.

The method works by forming a linear approximation to $f(\mathbf{x})$, so that given $\mathbf{x}^k$, the next iterate is chosen to minimize the sum of squares of the linearized terms, that is,

$$\mathbf{x}^{k+1} = \underset{\mathbf{x} \in \mathbb{R}^n}{\operatorname{argmin}} \left\{ \sum_{i=1}^{m} \left[f_i(\mathbf{x}^k) + \nabla f_i(\mathbf{x}^k)^\top (\mathbf{x} - \mathbf{x}^k) - c_i \right]^2 \right\}$$

We can see that this has reduced the problem to a linear least squares problem, which we know how to solve. We can write this optimization problem in the form

$$\mathbf{x}^{k+1} = \underset{\mathbf{x} \in \mathbb{R}^n}{\operatorname{argmin}} \left\| \mathbf{J}(\mathbf{x}^k)\mathbf{x} - \mathbf{b}^k \right\|^2$$

where

$$J(\mathbf{x}^k) = \nabla F(\mathbf{x}) = \begin{pmatrix} \frac{\partial f_1}{dx_1}(\mathbf{x}^k) & \ldots & \frac{\partial f_1}{dx_n}(\mathbf{x}^k) \\ \vdots & \vdots & \vdots \\ \frac{\partial f_m}{dx_1}(\mathbf{x}^k) & \ldots & \frac{\partial f_m}{dx_n}(\mathbf{x}^k) \end{pmatrix} = \begin{pmatrix} \nabla f_1 \left(\mathbf{x}^k\right)^\top \\ \nabla f_2 \left(\mathbf{x}^k\right)^\top \\ \vdots \\ \nabla f_m \left(\mathbf{x}^k\right)^\top \end{pmatrix}$$

is the $m \times n$ Jacobian matrix of F evaluated at $\mathbf{x}^k$, and

$$\mathbf{b}^k = \begin{pmatrix} \nabla f_1 \left(\mathbf{x}^k\right)^\top \mathbf{x}^k - f_1 \left(\mathbf{x}^k\right) + c_1 \\ \nabla f_2 \left(\mathbf{x}^k\right)^\top \mathbf{x}^k - f_2 \left(\mathbf{x}^k\right) + c_2 \\ \vdots \\ \nabla f_m \left(\mathbf{x}^k\right)^\top \mathbf{x}^k - f_m \left(\mathbf{x}^k\right) + c_m \end{pmatrix} = J(\mathbf{x}^k)\mathbf{x}^k - F(\mathbf{x}^k) \in \mathbb{R}^m .$$

If we assume the Jacobian matrix is of full rank, then the Gauss-Newton method can thus be written as:

$$\mathbf{x}^{k+1} = \left(J(\mathbf{x}^k)^\top J(\mathbf{x}^k)\right)^{-1} J(\mathbf{x}^k)^\top \mathbf{b}^k .$$

Note that the gradient of the objective function $g(\mathbf{x}) = \|F(\mathbf{x})\|^2$ is

$$\nabla g(\mathbf{x}) = 2J(\mathbf{x})^\top F(\mathbf{x}) .$$

and the Hessian is

$$\nabla^2 g(\mathbf{x}) = 2J(\mathbf{x})^\top J(\mathbf{x}) + 2\sum_{i=1}^{m}(f_i(\mathbf{x}) - c_i)\nabla^2 f_i(\mathbf{x}).$$

The Gauss-Newton method can be rewritten as follows (check):

$$\begin{aligned}
\mathbf{x}^{k+1} &= \left(J(\mathbf{x}^k)^\top J(\mathbf{x}^k)\right)^{-1} J(\mathbf{x}^k)^\top \left(J(\mathbf{x}^k)\mathbf{x}^k - F(\mathbf{x}^k)\right) \\
&= \mathbf{x}^k - \left(J(\mathbf{x}^k)^\top J(\mathbf{x}^k)\right)^{-1} J(\mathbf{x}^k)^\top F(\mathbf{x}^k) \\
&= \mathbf{x}^k - \frac{1}{2}\left(J(\mathbf{x}^k)^\top J(\mathbf{x}^k)\right)^{-1} \nabla g(\mathbf{x}^k).
\end{aligned}$$

So we can see that the Gauss-Newton method is just scaled gradient descent with a special choice of scaling matrix:

$$\mathbf{D}^k = \frac{1}{2}\left(J(\mathbf{x}^k)^\top J(\mathbf{x}^k)\right)^{-1} .$$

The Damped Gauss-Newton Method

The method above does not incorporate a stepsize, and is known as *pure Gauss-Newton*. The lack of a stepsize sometimes causes the algorithm to diverge. A variation of the algorithm that does incorporate a stepsize is the **damped Gauss-Newton method**:

Algorithm 1: The Damped Gauss-Newton Method

Initialization: A tolerance parameter $\varepsilon > 0$ and $\mathbf{x}^0 \in \mathbb{R}^n$.
General Step: for any $k = 0, 1, 2, \ldots$ execute the following steps:

1 Set $\mathbf{d}^k = -\left(J\left(\mathbf{x}^k\right)^\top J\left(\mathbf{x}^k\right)\right)^{-1} J\left(\mathbf{x}^k\right)^\top \mathrm{F}\left(\mathbf{x}^k\right)$.

2 Set t^k by a line search procedure on the function

$$h(t) = g(\mathbf{x}^k + t\mathbf{d}^k) .$$

3 Set $\mathbf{x}^{k+1} = \mathbf{x}^k + t^k \mathrm{d}_k$.

4 If $\left\|\nabla g\left(\mathbf{x}^{k+1}\right)\right\| \leq \varepsilon$, then STOP and $\mathbf{x}^{k+1}$ is the output.

Example: The Fermat-Weber Problem and Weiszfeld's Method

It turns out that there a lot of other methods that have been proposed that are related to the Gauss-Newton method. One particular problem, is the Fermat-Weber problem, which is a form of facility location problem. Suppose we are given the location of m cities, and want to build a new airport. Where should we place the airport so that we minimize the sum of distances from each city to the airport?

Stating the problem mathematically, suppose we are given m points in $\mathbb{R}^n : \mathbf{a}_1, \ldots, \mathbf{a}_m$, called *anchor points*, and m weights $\omega_1, \omega_2, \ldots, \omega_m > 0$, we want to find a point $\mathbf{x} \in \mathbb{R}^n$ that minimizes the weighted distance of $\mathbf{x}$ to each of the points $\omega_1, \omega_2, \ldots, \omega_m > 0$, that is

$$\min_{\mathbf{x}\in\mathbb{R}^n} \left\{ f(\mathbf{x}) \equiv \sum_{i=1}^{m} \omega_i \left\| \mathbf{x} - \mathbf{a}_i \right\|_2 \right\}$$

Note that the objective function is not differentiable at the anchor points $\mathbf{a}_1, \ldots, \mathbf{a}_m$ (why?).

In 1937, with only 16 years old, the Hungarian mathematician Endre Weiszfeld proposed a method for solving this problem, unsurprisingly[1] known as *Weiszfeld's Method*. Under the assumption that the optimum $\mathbf{x}$ is not an anchor point, the first order optimality conditions are

$$\nabla f(\mathbf{x}) = 0.$$

Using the fact that $\nabla ||\mathbf{x}||_2 = \frac{\mathbf{x}}{||\mathbf{x}||_2}$ we can see that this implies

$$\sum_{i=1}^{m} \omega_i \frac{\mathbf{x} - \mathbf{a}_i}{\|\mathbf{x} - \mathbf{a}_i\|} = 0\,,$$
$$\left(\sum_{i=1}^{m} \frac{\omega_i}{\|\mathbf{x} - \mathbf{a}_i\|} \right) \mathbf{x} = \sum_{i=1}^{m} \frac{\omega_i \mathbf{a}_i}{\|\mathbf{x} - \mathbf{a}_i\|}\,.$$

Thus the stationarity condition can be written as a fixed point $\mathbf{x} = T(\mathbf{x})$, where T is the function

$$T(\mathbf{x}) \equiv \frac{1}{\sum_{i=1}^{m} \frac{\omega_i}{\|\mathbf{x}-\mathbf{a}_i\|}} \sum_{i=1}^{m} \frac{\omega_i \mathbf{a}_i}{\|\mathbf{x} - \mathbf{a}_i\|}\,.$$

So finding the optimum $\mathbf{x}$ is equivalent to finding the fixed point of $\mathbf{x} = T(\mathbf{x})$. Weiszfeld proposed a fixed point iteration method:

$$\mathbf{x}^{k+1} = T(\mathbf{x}^k)\,,$$

[1] Or perhaps surprisingly, given Stigler's law of eponymy.

which can be interpreted as a gradient method since

$$\begin{aligned}\mathbf{x}^{k+1} &= \frac{1}{\sum_{i=1}^m \frac{\omega_i}{\|\mathbf{x}^k-\mathbf{a}_i\|}} \sum_{i=1}^m \frac{\omega_i \mathbf{a}_i}{\|\mathbf{x}^k-\mathbf{a}_i\|} \\ &= \mathbf{x}^k - \frac{1}{\sum_{i=1}^m \frac{1}{\|\mathbf{x}^k-\mathbf{a}_i\|}} \sum_{i=1}^m \omega_i \frac{\mathbf{x}^k-\mathbf{a}_i}{\|\mathbf{x}^k-\mathbf{a}_i\|} \\ &= \mathbf{x}^k - \frac{1}{\sum_{i=1}^m \frac{\omega_i}{\|\mathbf{x}^k-\mathbf{a}_i\|}} \nabla f(\mathbf{x}^k)\,.\end{aligned}$$

Therefore, it corresponds to a gradient method with a special choice of stepsize:

$$t^k = \frac{1}{\sum_{i=1}^m \frac{\omega_i}{\|\mathbf{x}^k-\mathbf{a}_i\|}}$$

Newton's Method

We now discuss a different class of algorithms for solving the minimization problem

$$\min\{f(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^n\}\,.$$

Recall that looking for the optimum begins with the search for stationary points $\mathbf{x}^*$ satisfying $\nabla f(\mathbf{x}^*) = 0$. This can be framed as finding roots (zeros) of $g(\mathbf{x}) \equiv \nabla f(\mathbf{x})$. A classical algorithm for finding the zeros of a function is Newton's method, which you have previously studied for functions of one dimension. Recall that for $g : \mathbb{R} \to \mathbb{R}$, Newton's method uses the iteration

$$x^{k+1} = x^k - \frac{g(x^k)}{g'(x^k)}\,.$$

We will now develop this idea in order to minimize a function of several variables. In the Gauss-Newton method we used a linear approximation to $f(\mathbf{x})$ which used the first order derivatives of f. The Newton method is a higher order method, and requires that $f(\mathbf{x})$ is twice continuously differentiable over $\mathbb{R}^n$ and that we can compute the Hessian $\nabla^2 f$. It then uses a quadratic approximation to $f(\mathbf{x})$, generating a sequence $\{\mathbf{x}^k\}_{k=1}^\infty$ using the iteration

$$\mathbf{x}^{k+1} = \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^n} \left\{ f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)^\top(\mathbf{x}-\mathbf{x}^k) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^k)^\top \nabla^2 f(\mathbf{x}^k)(\mathbf{x}-\mathbf{x}^k) \right\}\,.$$

This expression is interpreted as follows. Given a current value $\mathbf{x}^k$, we build a quadratic approximation of $f(\mathbf{x})$ around $\mathbf{x}^k$ (recall the quadratic approximation theorem), and find the minimizer of this quadratic approximation. In the case the minimizer exists, this is the next point of our sequence $\mathbf{x}^{k+1}$. We then repeat the process starting from $\mathbf{x}^{k+1}$ etc.

To solve this minimization problem we find the gradient of the quadratic approximation

$$\nabla\left(f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)^\top(\mathbf{x}-\mathbf{x}^k) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^k)^\top \nabla^2 f(\mathbf{x}^k)(\mathbf{x}-\mathbf{x}^k) \right) = \nabla f(\mathbf{x}^k) + \nabla^2 f(\mathbf{x}^k)(\mathbf{x}-\mathbf{x}^k)$$

and then set the derivative equal to zero (the first order optimization conditions) and solve for $\mathbf{x}$ giving

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

Note that this minimization problem is not well-defined in general, and that the solution above is only defined when the Hessian is positive definite at $\mathbf{x}^k$, i.e., when $\nabla^2 f\left(\mathbf{x}^k\right) \succ 0$. The vector

$$-(\nabla^2 f(\mathbf{x}^k))^{-1} \nabla f(\mathbf{x}^k)$$

is called **Newton's direction**, and the algorithm is called **Pure Newton's Method**:

Algorithm 2: Pure Newton's Method

Initialization: A tolerance parameter $\varepsilon > 0$ and $\mathbf{x}^0 \in \mathbb{R}^n$.

General Step: for any $k = 0, 1, 2, \ldots$ execute the following steps:

1 Compute the Newton direction $\mathbf{d}^k$, which is the solution to the linear system

$$\nabla^2 f(\mathbf{x}^k)\mathbf{d}^k = -\nabla f(\mathbf{x}^k)\,.$$

2 Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{d}^k$.

3 if $\left\|\nabla f\left(\mathbf{x}^{k+1}\right)\right\| \leq \varepsilon$, then STOP and $\mathbf{x}^{k+1}$ is the output.

In the nonlinear least squares problem considered earlier with objective function $g(\mathbf{x}) = \sum_{i=1}^m (f_i(\mathbf{x}) - c_i)^2$, recall that the Gauss-Newton method used the iteration

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \frac{1}{2}\left(J(\mathbf{x}^k)^\top J(\mathbf{x}^k)\right)^{-1} \nabla g(\mathbf{x}^k).$$

where as the Newton method would use the iteration

$$\mathbf{x}^{k+1} = \mathbf{x}^k - (\nabla^2 g(\mathbf{x}^k))^{-1} \nabla g(\mathbf{x}^k)$$

where

$$\nabla^2 g(\mathbf{x}) = 2J(\mathbf{x})^\top J(\mathbf{x}) + 2\sum_{i=1}^m (f_i(\mathbf{x}) - c_i)\nabla^2 f_i(\mathbf{x}).$$

Thus we can see that the Gauss-Newton method is similar to Newton's method, but ignores the second term in the Hessian above. It essentially uses $2J(\mathbf{x})^\top J(\mathbf{x})$ as an approximation to the Hessian matrix $\nabla^2 g$. The advantage of this is that we avoid the (potentially expensive) computation of $\nabla^2 g$, but the disadvantage is that Gauss-Newton may then take longer to converge. In practice, we often find that the first term $(J(\mathbf{x})^\top J(\mathbf{x}))$ is the dominant term in the Hessian, and so this helps explain why the Gauss-Newton approximation often works well in practice.

Convergence of Newton's Method

When will Newton's method converge?

At the very least, Newton's method requires that $\nabla^2 f(\mathbf{x}) > 0$ for every $\mathbf{x} \in \mathbb{R}^n$, which in particular implies that there exists a unique optimal solution $\mathbf{x}^*$. However, this is not sufficient to guarantee convergence.

Strong assumptions are required about f in order to guarantee convergence of Newton's method. In particular, we require that

A1: there exists $m > 0$ for which $\nabla^2 f(\mathbf{x}) \geq m\mathbf{I}$, for any $\mathbf{x} \in \mathbb{R}^n$,

A2: there exists $L > 0$ for which $\left\|\nabla^2 f(\mathbf{x}) - \nabla^2 f(\mathbf{y})\right\| \leq L\|\mathbf{x} - \mathbf{y}\|$ for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$.

A1 is

$$\nabla^2 f(\mathbf{x}) - m\mathbf{I} \geq 0 \quad \Rightarrow \quad \mathbf{x}^\top \nabla^2 f(\mathbf{x})\mathbf{x} \geq m\mathbf{x}^\top \mathbf{x} \text{ for all } \mathbf{x} \in \mathbb{R}^n.$$

This assumption essentially states[2] that the eigenvalues of the Hessian must be uniformly bounded away from 0 for all $\mathbf{x}$.

A2 states that $\nabla^2 f(\mathbf{x})$ is Lipschitz continuous, which is a strong form of uniform continuity (i.e. that $f \in C^{2,2}$).

Theorem (Quadratic Local Convergence of Newton's Method)**.** *Let f be a twice continuously differentiable function defined over $\mathbb{R}^n$, and assume that A1 and A2 hold for f. Let $\{\mathbf{x}^k\}_{k\geq 0}$ be the sequence generated by Newton's method and let $\mathbf{x}^*$ be the unique[3] minimizer of f over $\mathbb{R}^n$. Then for $k = 0, 1, \ldots$*

$$\left\|\mathbf{x}^{k+1} - \mathbf{x}^*\right\| \leq \frac{L}{2m}\left\|\mathbf{x}^k - \mathbf{x}^*\right\|^2.$$

In addition, if $\left\|\mathbf{x}^0 - \mathbf{x}^*\right\| \leq \frac{m}{L}$, then:

$$\left\|\mathbf{x}^k - \mathbf{x}^*\right\| \leq \frac{2m}{L}\left(\frac{1}{4}\right)^{2^k}, \quad k = 0, 1, 2, \ldots$$

Despite all of the assumptions required to guarantee convergence, Newton's method is attractive because of its local quadratic rate of convergence, i.e., if the error at stage k is $e_k := \left\|\mathbf{x}^k - \mathbf{x}^*\right\|$, then

$$e_{k+1} \leq Me_k^2 \text{ for some } M > 0.$$

This essentially means that the number of accuracy digits is doubled at each iteration. This is in contrast to the gradient method in which the convergence theorems are rather independent in the starting point, but only "relatively" slow linear convergence is assured.

[2] Suppose λ is an eigenvalue of $\nabla^2 f(\mathbf{x})$ with corresponding unit eigenvector $\mathbf{v}$. Then

$$\lambda = \mathbf{v}^\top \nabla^2 f(\mathbf{x})\mathbf{v} \geq m\mathbf{v}^\top \mathbf{v} = m.$$

[3] Any minimum is the unique global minimum as $\nabla^2 f(\mathbf{x}) > 0$ for all $\mathbf{x}$.

Proof. We prove the first part of the result. Let k be a nonnegative integer. Then

$$\begin{aligned}
\mathbf{x}^{k+1} - \mathbf{x}^* =& \mathbf{x}^k - (\nabla^2 f(\mathbf{x}^k))^{-1}\nabla f(\mathbf{x}^k) - \mathbf{x}^* \\
=& \mathbf{x}^k - \mathbf{x}^* + (\nabla^2 f(\mathbf{x}^k))^{-1}(\nabla f(\mathbf{x}^*) - \nabla f(\mathbf{x}^k)) \quad \text{as} \quad \nabla f(\mathbf{x}^*) = 0 \\
=& \mathbf{x}^k - \mathbf{x}^* + (\nabla^2 f(\mathbf{x}^k))^{-1} \int_0^1 \left[\nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^* - \mathbf{x}^k))\right](\mathbf{x}^* - \mathbf{x}^k)dt \\
& \qquad \text{by the fundamental theorem of calculus} \\
=& (\nabla^2 f(\mathbf{x}^k))^{-1} \int_0^1 \left[\nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^* - \mathbf{x}^k)) - \nabla^2 f(\mathbf{x}^k)\right](\mathbf{x}^* - \mathbf{x}^k)dt
\end{aligned}$$

Assumption A1, that $\nabla^2 f(\mathbf{x}^k) \geq m\mathbf{I}$, implies that the eigenvalues of the Hessian are all bigger than m, and so (see exercises)

$$\left\|(\nabla^2 f(\mathbf{x}^k))^{-1}\right\| \leq \frac{1}{m}.$$

Thus,

$$\begin{aligned}
\left\|\mathbf{x}^{k+1} - \mathbf{x}^*\right\| &\leq \left\|(\nabla^2 f(\mathbf{x}^k))^{-1}\right\| \left\|\int_0^1 \left[\nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^* - \mathbf{x}^k)) - \nabla^2 f(\mathbf{x}^k)\right](\mathbf{x}^* - \mathbf{x}^k)dt\right\| \\
&\leq \left\|(\nabla^2 f(\mathbf{x}^k))^{-1}\right\| \int_0^1 \left\|\left[\nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^* - \mathbf{x}^k)) - \nabla^2 f(\mathbf{x}^k)\right](\mathbf{x}^* - \mathbf{x}^k)\right\| dt \\
&\leq \left\|(\nabla^2 f(\mathbf{x}^k))^{-1}\right\| \int_0^1 \left\|\nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^* - \mathbf{x}^k)) - \nabla^2 f(\mathbf{x}^k)\right\| \cdot \left\|\mathbf{x}^* - \mathbf{x}^k\right\| dt \\
&\leq \frac{L}{m} \int_0^1 t \left\|\mathbf{x}^k - \mathbf{x}^*\right\|^2 dt = \frac{L}{2m} \left\|\mathbf{x}^k - \mathbf{x}^*\right\|^2.
\end{aligned}$$

The first and third inequalities follows as $||\mathbf{Ax}|| \leq ||\mathbf{A}|| \cdot ||\mathbf{x}||$ for general $\mathbf{A}$ and $\mathbf{x}$. The final inequality uses assumption A2.

The second part of the theorem can be shown by induction (try it!). □

An alternative is to use a damped version of Newton's method, using backtracking:

Algorithm 3: Damped Newton's Method

Initialization: A tolerance parameter $\varepsilon > 0$ and $\mathbf{x}^0 \in \mathbb{R}^n$. (α, β) parameters for the backtracking procedure ($\alpha \in (0,1), \beta \in (0,1)$).

General Step: for any $k = 0, 1, 2, \ldots$ execute the following steps:

1 Compute the Newton direction $\mathbf{d}^k$, which is the solution to the linear system

$$\nabla^2 f(\mathbf{x}^k)\mathbf{d}^k = -\nabla f(\mathbf{x}^k).$$

2 Set $t_k = 1$. **while** $\left(f\left(\mathbf{x}^k\right) - f\left(\mathbf{x}^k + t^k\mathbf{d}^k\right) < -\alpha t^k \nabla f\left(\mathbf{x}^k\right)^\top \mathbf{d}^k\right)$ **do**

3 set $t^k := \beta t^k$.

4 Set $\mathbf{x}^{k+1} = \mathbf{x}^k + t^k\mathbf{d}^k$.

5 If $\left\|\nabla f\left(\mathbf{x}^{k+1}\right)\right\| \leq \varepsilon$, then STOP and $\mathbf{x}^{k+1}$ is the output.

Step 2. is a form of backtracking, and seeks a good stepsize as we did for gradient descent.

Checklist

The idea of this checklist is to help you to self-evaluate your progress and understanding of the subject, and to give you some guidance on where to focus. If you can tick all the boxes it means you're doing alright, otherwise you need to study a bit more, grab a book, watch the videos, or seek help from classmates, the lecturers, or the demonstrators. Try to fill as many gaps as quickly as possible.

And remember to do the 

's!

Learning Outcome	Check
I understand the formulation of nonlinear regression as an optimization problem and its difference with linear least squares.	
I can recognize the Gauss-Newton method as an iterative linearization of the NLS problem.	
I can establish the link between the Gauss-Newton method and gradient descent.	
I have read the formulation of the Fermat-Weber problem and its solution.	

Exercises

1. Analyse what happens to Newton's method when $f(x) = \sqrt{1+x^2}$.
2. Show that $\nabla^2 f(\mathbf{x}^k) \succeq m\mathbf{I}$ implies that the eigenvalues of the Hessian are all bigger than m
$$\left\|(\nabla^2 f(\mathbf{x}^k))^{-1}\right\| \leq \frac{1}{m}\,.$$
3. Using the conditions in the Theorem on the Quadratic Local Convergence of Newton's Method, show that if $\left\|\mathbf{x}^0 - \mathbf{x}^*\right\| \leq \frac{m}{L}$, then:
$$\left\|\mathbf{x}^k - \mathbf{x}^*\right\| \leq \frac{2m}{L}\left(\frac{1}{4}\right)^{2^k}, \quad k = 0, 1, 2, \ldots$$
using proof by induction.
4. Consider the minimization problem
$$\min 100x^4 + 0.01y^4\,.$$
Compare Newton's method against gradient descent with backtracking. Repeat for
$$\min \sqrt{x_1^2+1} + \sqrt{x_2^2+1}\,.$$
starting from $\mathbf{x}_0 = (0.9, 0.9)^\top$ and from $\mathbf{x}_0 = (10,\ 10)^\top$.

Repeat the last numerical example using damped Newton's method starting from $(10, 10)$.

You are given data $\{t_i, y_i\}_{i=1}^n$ for $t_i, y_i \in \mathbb{R}$, and want to fit the model
$$y = x_1 \exp(x_2 t)$$
using least squares estimation, i.e., you want to solve
$$\min_{\mathbf{x}} \sum_{i=1}^{n} (y_i - x_1 \exp(x_2 t_i))^2.$$
Write down a Gauss-Newton method and a pure Newton's method for solving this problem.