Lec9p1, ORF363/COS323

Instructor:
Amir Ali Ahmadi

This lecture:

- Multivariate Newton's method
- Rate of convergence
- Modifications for global convergence
- Nonlinear least squares
    - The Gauss-Newton algorithm

- In the previous lecture, we saw the general framework of descent algorithms, with several choices for the step size and the descent direction. We also discussed convergence issues associated with these methods and provided some formal definitions for studying rates of convergence. Our focus before was on gradient descent methods and variants, which use only first order information (first order derivatives). These algorithms achieved a linear rate of convergence.

- Today, we see a wonderful descent method with superlinear (in fact quadratic) rate of convergence: the Newton algorithm. This is a generalization of what we saw a couple of lectures ago in dimension one for root finding and function minimization.

- The Newton's method is nothing but a descent method with a specific choice of a descent direction; one that iteratively adjusts itself to the local geometry of the function to be minimized.

- In practice, Newton's method can converge with much fewer iterations than gradient methods. For example, for quadratic functions, while we saw that gradient methods can zigzag for a long time (depending on the underlying condition number), Newton's method will always get the optimal solution in a single step.

- The cost that we pay for fast convergence is the need to (i) access second order information (i.e., derivatives of first and second order), and (ii) solve a linear system of equations at every step of the algorithm.

- Our presentation uses references [Bert03],[CZ13], [Bert09], [Tit13]. Let's get right to the derivation of the Newton iterates!

- Recall the general form of our descent methods:

$$x_{k+1} = x_k + \alpha_k d_k$$

where

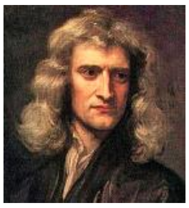
$k \in \mathbb{Z}_+$: index of time (iteration number)

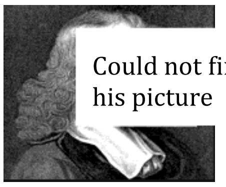
$x_k \in \mathbb{R}^n$: Current point

$x_{k+1} \in \mathbb{R}^n$: Next point

$d_k \in \mathbb{R}^n$: Direction to move along at iteration k

$\alpha_k \in \mathbb{R}_+$: Step size at iteration k


Newton
(1642-1727)



Raphson
(1648-1715?)

Apparently, Raphson published his discovery of the "Newton's method" 50 years prior to Newton [Bert09]. That's what happens when people can't Google!

- Let us have $\alpha_k = 1, \forall k$ for now. So we take full steps at each iteration. (This is sometimes called the *pure* Newton method.)
- Newton's method is simply the following choice for the descent direction:

$$d_k = -(\nabla^2 f(x_k))^{-1} \nabla f(x_k).$$

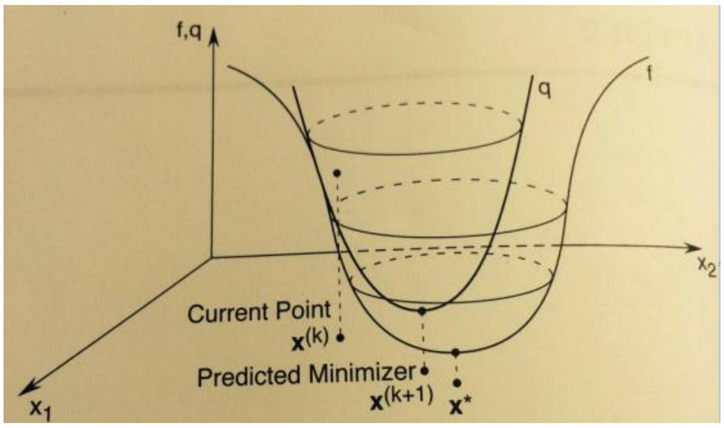
  - Recall our notation $\nabla f(x_k)$ and $\nabla^2 f(x_k)$ respectively denote the gradient and the Hessian at x_k.
  - Iteration only well-defined when the Hessian at x_k is invertible.

Where does this come from?

- One motivation: minimizing a quadratic approximation of the function sequentially.
  - Around the current iterate x_k, let's Taylor expand our function to second order and minimize the resulting quadratic function.

$$f(x) \approx f(x_k) + \nabla f(x_k)^T (x - x_k) + \frac{1}{2}(x - x_k)^T \nabla^2 f(x_k)(x - x_k) := q(x)$$

Image credit: [CZ13]



$$q(x) = f(x_k) + \nabla f(x_k)^T(x - x_k) + \frac{1}{2}(x - x_k)^T \nabla^2 f(x_k)(x - x_k)$$

- What is the minimum of this function?
- q is lower bounded only if $\nabla^2 f(x_k) \succcurlyeq 0$ (why?).
- First order necessary condition tells us that $\nabla q(x_*) = 0$. Hence,

$$\nabla f(x_k) + \nabla^2 f(x_k)x - \nabla^2 f(x_k)x_k = 0$$

$$\Rightarrow x = x_k - \left(\nabla^2 f(x_k)\right)^{-1} \nabla f(x_k)$$

- If $\nabla^2 f(x_k) \succcurlyeq 0$, then q is convex and hence our stationary point is a global optimum.
- Newton's method picks this point as the next iterate

$$x_{k+1} = x_k - \left(\nabla^2 f(x_k)\right)^{-1} \nabla f(x_k).$$

Newton's method for minimizing quadratic functions

- If $f = \frac{1}{2}x^T Q x + b^T x + c$, with $Q \succ 0$, Newton's method finds the global minimum in a single iteration.
    - Our derivation above already shows this.

- When f is not quadratic, we still have the following convergence result.

Convergence and Rate of Convergence

Theorem. Suppose $f \in C^3$ (i.e., three times continuously differentiable) and $x_* \in \mathbb{R}^n$ is such that $\nabla f(x_*) = 0$ and $\nabla^2 f(x_*)$ is invertible. Then, there exists $\epsilon > 0$ such that itearations of Newton's method starting from any point $x_0 \in B(x_*, \epsilon)$ are well-defined and converge to x_*. Moreover, the convergence rate is quadratic.

- **Interpretation:** under the assumptions of the theorem, there is a basin around stationary points such that once you are trapped in it, you converge to the stationary point very fast. Once in this basin, typically no more than 4,5 iterations are needed to obtain the limit point with high accuracy.
- **Caution:**
  - This is a local statement. No guarantee that Newton iterations would converge if we start far away.
  - No guarantee that our limit point will be a local minimum. It can potentially even be a local maximum!

To prove this theorem, we need two lemmas.

Lemma 1. Let A be an $n \times n$ matrix and $x \in \mathbb{R}^n$. Let $||.||$ denote a vector norm and also its associated induced matrix norm (definition right below). Then, $||Ax|| \leq ||A|| \cdot ||x||$.

Proof. By definition, $\overbrace{\|A\| = \max_{\|y\|=1} \|Ay\|}^{①}$. Suppose $\exists x \in \mathbb{R}^n$ s.t. $\|Ax\| > \|A\|\,\|x\|$.

Then, $\|A \underbrace{\frac{x}{\|x\|}}_{\hookrightarrow \text{norm } 1.}\| = \frac{1}{\|x\|}\|Ax\| > \|A\|$. This contradicts ①.

Lemma 2. Let $A: \mathbb{R}^n \to \mathbb{R}^{n \times n}$ be a matrix valued function that is continuous at a point $u \in \mathbb{R}^n$. If $A(u)^{-1}$ exists, then there exists a scalar $\epsilon > 0$ such that $A(v)^{-1}$ also exists for all $v \in B(u, \epsilon)$.

Proof. See, e.g., Lemma 5.3 in Section 5.1 of [CZ13].

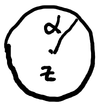
Proof of the theorem. We follow [CZ13], with some details filled in.

Since $f \in C^3$, the Taylor expansion of ∇f around x_0 (our starting point) gives

$$\nabla f(x) = \nabla f(x_0) + \nabla^2 f(x_0)(x - x_0) + O(\|x - x_0\|^2). \qquad ①$$

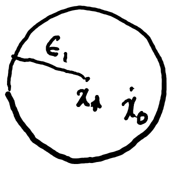
(This error bound on the Taylor expansion is proven in [CZ13], Sect. 5.6.)

[Notation: For $z \in \mathbb{R}^n$, $\alpha \in \mathbb{R}_+$, let $B(z,\alpha) := \{x \in \mathbb{R}^n \mid \|x - z\| \leq \alpha\}$.
This is a ball of radius α centered at z.]



① $\Rightarrow$ $\exists\, \epsilon_1, c_1 > 0$ s.t. if $x_0, x \in B(x_*, \epsilon_1)$, then

$$\|\nabla f(x) - \nabla f(x_0) - \nabla^2 f(x_0)(x - x_0)\| \leq c_1 \|x - x_0\|^2. \quad \text{(why?)}$$



$\nabla^2 f(x_*)$ invertible + Lemma 2 $\Rightarrow$ $\exists\, \epsilon_2, c_2 > 0$ s.t. $\forall x \in B(x_*, \epsilon_2)$

$\nabla^2 f(x)$ exists and $\|\nabla^2 f^{-1}(x)\| \leq c_2$.

Let $\epsilon = \min\{\epsilon_1, \epsilon_2\}$. Then, $x_0, x \in B(x_*, \epsilon) \Rightarrow$

- $\|\nabla f(x) - \nabla f(x_0) - \nabla^2 f(x_0)(x - x_0)\| \leq c_1 \|x - x_0\|^2$,
- $\nabla^2 f(x)$ exists and $\|\nabla^2 f^{-1}(x)\| \leq c_2$. ②

Suppose $x_0 \in B(x_*, \epsilon)$. Substituting $x = x_*$ above and recalling that $\nabla f(x_*) = 0$, we get

$$\|\nabla^2 f(x_0)(x_0 - x_*) - \nabla f(x_0)\| \leq c_1 \|x_* - x_0\|^2. \qquad ③$$

Proof (Cont'd). Recall the Newton iterations:

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1} \nabla f(x_k) .$$

Writing this for $k=0$, subtracting x_* from both sides, and taking norms we get

$$\|x_1 - x_*\| = \| x_0 - x_* - \nabla^2 f^{-1}(x_0) \nabla f(x_0)\|$$

$$= \| \nabla^2 f^{-1}(x_0) \left(\nabla^2 f(x_0) (x_0 - x_*) - \nabla f(x_0)\right)\|$$

$$\xrightarrow{\text{Lemma 1}} \leq \| \nabla^2 f^{-1}(x_0)\| \, \| \nabla^2 f(x_0) (x_0 - x_*) - \nabla f(x_0)\| .$$

In view of ②, ③, we get

$$\| x_1 - x_*\| \leq c_1 c_2 \|x_0 - x_*\|^2 .$$

Let x_0 be such

$$\|x_0 - x_*\| \leq \frac{\alpha}{c_1 c_2} , \quad \text{where } 0 < \alpha < 1 .$$

$$\Rightarrow \|x_1 - x_*\| \leq \alpha \|x_0 - x_*\| .$$

This implies that $x_1 \in B(x_*, \epsilon)$, and $\|x_1 - x_*\| \leq \frac{\alpha}{c_1 c_2}$.

So the same arguments apply to x_1 and we can conclude

$$\|x_2 - x_*\| \leq c_1 c_2 \|x_1 - x_*\|^2 , \qquad \|x_2 - x_*\| \leq \alpha \|x_1 - x_*\| .$$

By induction,

$$\|x_{k+1} - x_*\| \leq c_1 c_2 \|x_k - x_*\|^2 , \qquad \|x_{k+1} - x_*\| \leq \alpha \|x_k - x_*\| .$$

$$\Downarrow \qquad\qquad\qquad\qquad \Downarrow$$

quadratic convergence. $\qquad \|x_k - x_*\| \leq \alpha^k \|x_0 - x_*\|$

$$\Downarrow \alpha < 1$$

$$\{x_k\} \longrightarrow x_* .$$

$\square$

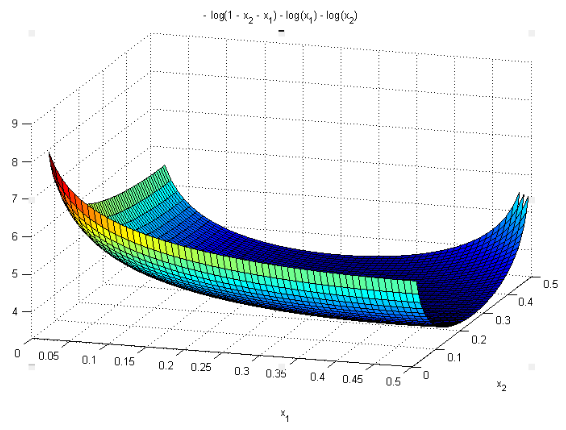
Example 1 ([Ber09]).

$$f(x) = -\log(1 - x_1 - x_2) - \log x_1 - \log x_2$$

$$x_* = \left(\frac{1}{3}, \frac{1}{3}\right),$$
$$f(x_*) = 3.295836867$$

- Is this function convex? Why?

```
>> syms x1 x2;
>> f=-log(1-x1-x2)-log(x1)-log(x2);
>> gradf=gradient(f)

gradf =

 - 1/(x1 + x2 - 1) - 1/x1
 - 1/(x1 + x2 - 1) - 1/x2

>> Hf=hessian(f)

Hf =

[ 1/(x1 + x2 - 1)^2 + 1/x1^2,         1/(x1 + x2 - 1)^2]
[         1/(x1 + x2 - 1)^2, 1/(x1 + x2 - 1)^2 + 1/x2^2]

>> ezsurf(f,[0,.5,0,.5])
```



Simple code for Newton iterations:

```
syms x1 x2;
f=-log(1-x1-x2)-log(x1)-log(x2);
gradf=gradient(f);
Hf=hessian(f);
N=10; %number of Newton iterations
x=zeros(2,N); fval=zeros(N,1); er=zeros(N,1);
x(:,1)=[.8;.1]; %initial point

for k=1:N-1
   fval(k)=subs(f,{x1,x2},{x(1,k),x(2,k)});
   er(k)=norm([1/3;1/3]-x(:,k));
   x(:,k+1)=x(:,k)- subs(Hf,{x1,x2},{x(1,k),x(2,k)})\
subs(gradf,{x1,x2},{x(1,k),x(2,k)});
end
 fval(k+1)=subs(f,{x1,x2},{x(1,k+1),x(2,k+1)});
 er(k+1)=norm([1/3;1/3]-x(:,k+1));
format long
output=[(1:N)' x' er fval]
```

output =

```
 k                     x_k                        ||x_k - x_*||         f(x_k)
 1  0.800000000000000  0.100000000000000  0.521749194749951  4.828313737302302
 2  0.630303030303030  0.184848484848485  0.332022214840878  3.837992155333637
 3  0.407373701516407  0.296313149241797  0.082779648168232  3.330701223771961
 4  0.328873379058184  0.335563310470908  0.004986380467888  3.295971739464466
 5  0.333302700862786  0.333348649568607  0.000034248143232  3.295836872338374
 6  0.333333331925552  0.333333334037224  0.000000001573947  3.295836866004329
 7  0.333333333333333  0.333333333333333          0          3.295836866004329
 8  0.333333333333333  0.333333333333333          0          3.295836866004329
 9  0.333333333333333  0.333333333333333          0          3.295836866004329
10  0.333333333333333  0.333333333333333          0          3.295836866004329
```

Number of correct significant digits doubles in each iteration.

Example 2 ([Ber09]).

$$f(x) = 7x - \log x \qquad x_* = \frac{1}{7} = 0.1428571428$$

$$f'(x) = 7 - \frac{1}{x}, \qquad f''(x) = \frac{1}{x^2}$$

$$x_{k+1} = x_k - x_k^2\left(7 - \frac{1}{x_k}\right) = 2x_k - 7x_k^2$$

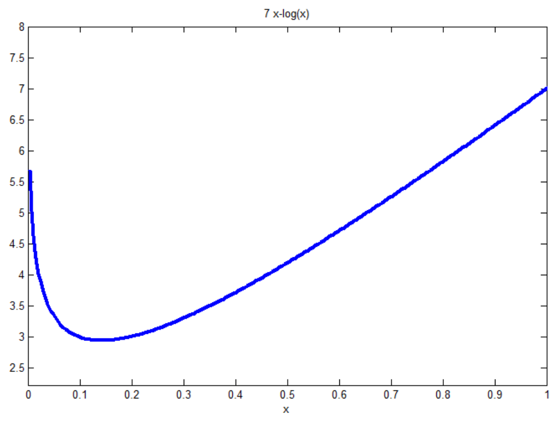
	✗	✗	✓	✓
k	x^k	x^k	x^k	x^k
0	1	0	0.01	0.1
1	-5	0	0.0193	0.13
2	-185	0	0.03599	0.1417
3	-239945	0	0.062917	0.14284777
4	-4E11	0	0.098124	0.142857142
5	-112E22	0	0.128849782	0.142857143
6		0	0.141483700	0.142857143
7		0	0.142843938	0.142857143
8		0	0.142857142	0.142857143

From [Ber09]

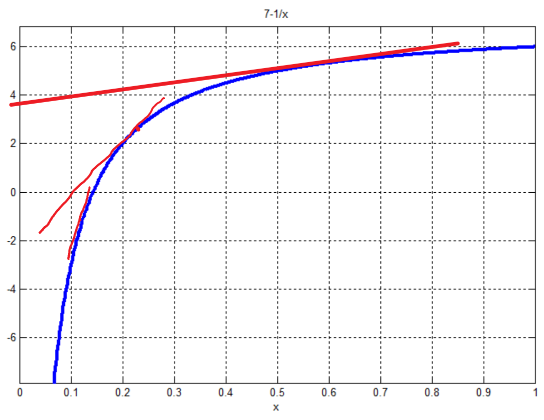
Four different initial conditions

- Lack of global convergence.
- Sensitivity to starting point.

$f(x)$ $f'(x)$



Want to minimize.



Want to find a zero.

- The geometric interpretation for finding a root of $f'(x)$ via Newton's method shows what exactly is going wrong if we start far off.

k	x^k ✗	x^k ✗	x^k ✓	x^k ✓
0	1	0	0.01	0.1
1	-5	0	0.0193	0.13
2	-185	0	0.03599	0.1417
3	-239945	0	0.062917	0.14284777
4	-4E11	0	0.098124	0.142857142
5	-112E22	0	0.128849782	0.142857143
6		0	0.141483700	0.142857143
7		0	0.142843938	0.142857143
8		0	0.142857142	0.142857143

- What is the basin of attraction of the root? $\left(0, \frac{2}{7}\right)$
- Do you see why?

- This motivates the *guarded* Newton method
  - Let's not take full Newton steps
  - Introduce a step size.

- Newton's method does not guarantee descent in every iteration; i.e., we may have $f(x_{k+1}) \geq f(x_k)$.
- But the Newton direction is a descent direction for convex functions.
  - Means that if you make a small enough step, you do get the decrease property.
- Let's recall this notion and prove this statement formally.

Definition. For a given point $x \in \mathbb{R}^n$, a direction $d \in \mathbb{R}^n$ is called a descent direction, if there exists $\bar{\alpha} > 0$ $(\alpha \in \mathbb{R})$ such that

$$f(x + \alpha d) < f(x), \qquad \forall \alpha \in (0, \bar{\alpha}).$$

Theorem. At any point $x \in \mathbb{R}^n$ where $\nabla f(x) \neq 0$ and $\nabla^2 f(x) \succ 0$, the Newton direction $-(\nabla^2 f(x))^{-1} \nabla f(x)$ is a descent direction.

Proof. Define $h(\alpha) = f(x - \alpha [\nabla^2 f(x)]^{-1} \nabla f(x))$.

Then, $h'(\alpha) = (-[\nabla^2 f(x)]^{-1} \nabla f(x))^T \nabla f(x - \alpha [\nabla^2 f(x)]^{-1} \nabla f(x))$.

$h'(0) = -\nabla f^T(x) [\nabla^2 f(x)]^{-1} \nabla f(x)$ ①

①

$$\left.\begin{array}{l} \nabla f(x) \neq 0 \\ \nabla^2 f(x) \succ 0 \overset{(why?)}{\Rightarrow} [\nabla^2 f(x)]^{-1} \succ 0 \end{array}\right\} \Rightarrow h'(0) < 0.$$

$h'(0) < 0 \Rightarrow \exists \bar{\alpha} > 0$, s.t. $h(\alpha) < h(0)$ $\forall \alpha \in (0, \bar{\alpha})$.

$\Rightarrow f(x - \alpha [\nabla^2 f(x)]^{-1} \nabla f(x)) < f(x)$ $\forall \alpha \in (0, \bar{\alpha})$. $\square$

- Note that what we just proved is a corollary of a more general result from last lecture.

Newton method with a step size

$$x_{k+1} = x_k - \alpha_k\left(\nabla^2 f(x_k)\right)^{-1}\nabla f(x_k)$$

- We saw many choices of the step size $\alpha_k \geq 0$ in the previous lecture.
- Let's learn about a new and a very popular one:

The Armijo Rule

- This is an *inexact line search* method. It does not find the exact minimum along the line. But it guarantees sufficient decrease and it's cheap.
- Armijo rule requires two parameters: $\epsilon \in (0,1), \delta > 1$.

Suppose we would like to minimize a univariate function $h(\alpha)$ over $\alpha \geq 0$. (For us, $h(\alpha) = f(x_k + \alpha d_k)$ where x_k and d_k are fixed and d_k is a direction to go along, e.g., the Newton direction.)

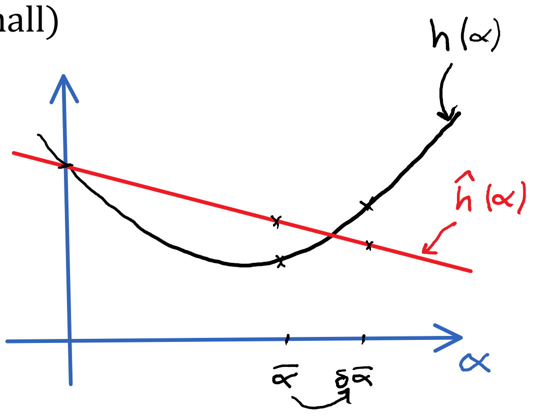
Define an affine univariate function as

$$\hat{h}(\alpha) = h(0) + \epsilon h'(0)\alpha$$

Armijo rule accepts a stepsize $\bar{\alpha}$ if:

- $h(\bar{\alpha}) \leq \hat{h}(\bar{\alpha})$ (ensures sufficient decrease)
- $h(\delta\bar{\alpha}) \geq \hat{h}(\delta\bar{\alpha})$ (ensures stepsize is not too small)

Note that in general there will be a whole range of step sizes that would be accepted.



The Armijo backtracking algorithm:

- Start with some initial step size α_0.
- At iteration j:
  - If $h(\alpha_j) \leq \hat{h}(\alpha_j)$, stop; declare α_j as your step size.
  - If $h(\alpha_j) > \hat{h}(\alpha_j)$, let $\alpha_{j+1} = \frac{1}{\delta}\alpha_j$.

Levenberg-Marquardt Modification

- If $\nabla^2 f(x) \nsucc 0$, Newton's direction may not be a descent direction.
- If $\nabla^2 f(x)$ is singular, Newton's direction is not even well-defined.
- Idea: let's make $\nabla^2 f(x)$ positive definite if it isn't.

$$x_{k+1} = x_k - (\nabla^2 f(x_k) + \mu_k I)^{-1} \nabla f(x_k), \qquad \mu_k \geq 0. \qquad (*)$$

Lemma. Let A an $n \times n$ matrix with eigenvalues $\lambda_1, \dots, \lambda_n$, and let $\mu \in \mathbb{R}$. Then, eigenvalues of $A + \mu I$ are $\lambda_1 + \mu, \dots, \lambda_n + \mu$.

Proof. Let λ_i be an eigenvalue of A with eigenvector v_i;

i.e., $A v_i = \lambda_i v_i$.

Now $(A + \mu I) v_i = A v_i + \mu v_i = \lambda_i v_i + \mu v_i = (\lambda_i + \mu) v_i$

$\Rightarrow \lambda_i + \mu$ is an eigenvalue of $A + \mu I$. $\square$

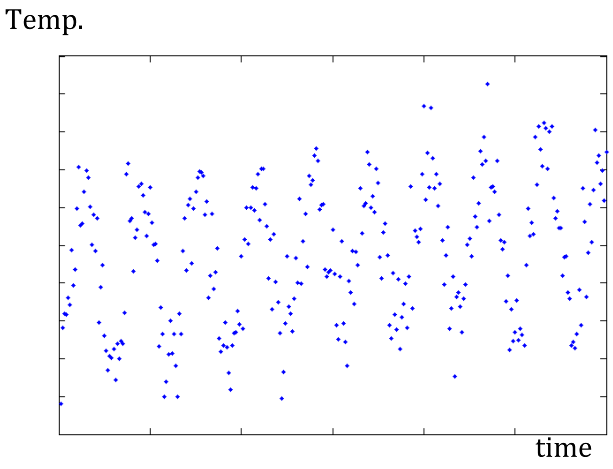
Comments about $(*)$:

- If μ_k is large enough, $-(\nabla^2 f(x_k) + \mu_k I)^{-1} \nabla f(x_k)$ will be a descent direction and by choosing a small enough step size α_k we can ensure descent.
- As $\mu_k \to 0$, we approach the regular Newton method.
- As $\mu_k \to \infty$, we approach a pure gradient method with a small step size (why?).
- In practice, we can start with a small value of μ_k and increase it slowly until we observe descent: $f(x_{k+1}) < f(x_k)$.

The Gauss-Newton method for nonlinear least squares

Suppose you have observed the temperature in a town over the past several years. In view of seasonal effects and global warming, you postulate a model of the following form for the temperature at day t:

$T(t) = \mathrm{a} \cdot \sin(\omega t + \phi) + bt$



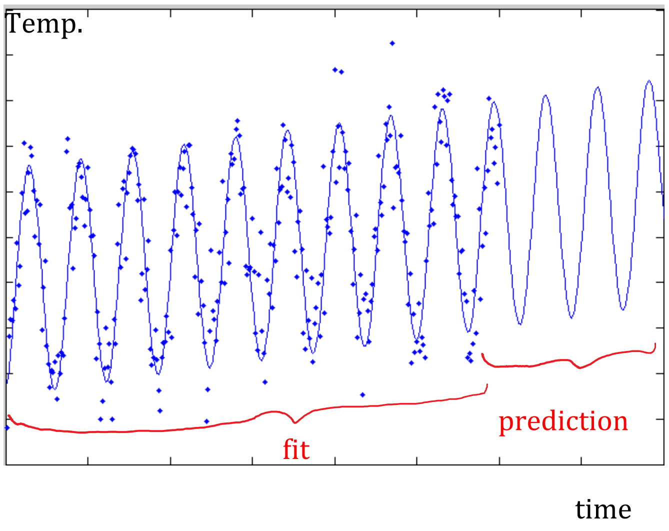
The task is to find the parameters a, ω, ϕ, b that best fit the data. Once this is done, we can use them to predict temperature at a future date.

Denote the given data points by

$(t_i, T_i), \qquad i = 1, \dots, m.$

Goal is to minimize

$$f(a, \omega, \phi, b) = \sum_{i=1}^{m} (T_i - a\sin(\omega t_i + \phi) - bt_i)^2$$



This is a **nonlinear least squares** problem.

More generally, we have a list of (possibly nonlinear) functions $g_i(x): \mathbb{R}^n \to \mathbb{R}, i = 1, \dots, m,$

and would like to minimize

$$f(x) = \sum_{i=1}^{m} g_i^2(x)$$

The Gauss-Newton method is an approximation of Newton's method for minimizing this function.

Given (possibly nonlinear) functions $g_i(x)\colon \mathbb{R}^n \to \mathbb{R}, i = 1, \dots, m$, (tipically $m \geq n$), we would like to minimize

$$f(x) = \frac{1}{2}\sum_{i=1}^{m} g_i^2(x).$$

Let $g\colon \mathbb{R}^n \to \mathbb{R}^m$ be defined as $g = \begin{pmatrix} g_1 \\ \vdots \\ g_m \end{pmatrix}$. Let $J(x)$ be the $m \times n$ Jacobian matrix of g; i.e., $J_{i,j}(x) = \frac{\partial g_i(x)}{\partial x_j}$. Then the Gauss-Newton iteration reads

$$x_{k+1} = x_k - (J(x_k)^T J(x_k))^{-1} J(x_k)^T g(x_k)$$

Comments:

- Unlike Newton, the Gauss-Newton method only uses first order information (you only see first derivatives, no second derivatives).
- The direction $-(J(x_k)^T J(x_k))^{-1} J(x_k)^T g(x_k)$ is a descent direction, because $J(x_k)^T g(x_k)$ is the gradient of f and $(J(x_k)^T J(x_k))^{-1}$ is positive semidefinite (why?)
- If $J(x_k)^T J(x_k)$ is not invertible (i.e., it's not positive definite), then we can apply the Levenberg-Marquardt modification as before (shift eigenvalues to the right by a little bit).
- If you were to write down the Newton iteration for minimizing f, you would get:

$$x_{k+1} = x_k - \left(J(x_k)^T J(x_k) + \sum_{i=1}^{m} \nabla^2 g_i(x_k) g_i(x_k) \right)^{-1} J(x_k)^T g(x_k)$$

- Note that we are ignoring the term $\sum_{i=1}^{m} \nabla^2 g_i(x_k) g_i(x_k)$. This is a good approximation when g_i is close to linear or when g_i is small.
- If g_i is linear for $i = 1, \dots, m$
  - We have a (linear) least squares problem.
  - Gauss-Newton equals Newton (why?).
  - One iteration is enough to solve the problem globally (why?).

Given (possibly nonlinear) functions $g_i(x)\colon \mathbb{R}^n \to \mathbb{R}, i = 1, \ldots, m$, (tipically $m \geq n$), we would like to minimize

$$f(x) = \frac{1}{2}\sum_{i=1}^{m} \left(g_i(x)\right)^2.$$

Let $g\colon \mathbb{R}^n \to \mathbb{R}^m$ be defined as $g = \begin{pmatrix} g_1 \\ \vdots \\ g_m \end{pmatrix}$. Let $J(x)$ be the $m \times n$ Jacobian matrix of g; i.e., $J_{i,j}(x) = \frac{\partial g_i(x)}{\partial x_j}$. Then the Gauss-Newton iteration reads

$$x_{k+1} = x_k - \left(J(x_k)^T J(x_k)\right)^{-1} J(x_k)^T g(x_k)$$

Derivation.

1. Replace $g(x)$ with its first order approximation $\tilde{g}(x, x_k)$ near the current iterate x_k.
2. Instead minimize $\frac{1}{2}||\tilde{g}(x, x_k)||^2$. This is now a convex quadratic function; we know how to do it.

$$\tilde{g}(x, x_k) = g(x_k) + J(x_k)\,(x - x_k)$$

Hint: $f(x) = \frac{1}{2}(x-x_k)^T Q\,(x-x_k) + c^T(x-x_k) + \frac{b}{2}$

$\nabla f(x) = Q\,(x-x_k) + c$

$\nabla f(x) = 0 \Rightarrow Qx - Qx_k = -c$

$\Rightarrow x = Q^{-1}\,(Qx_k - c)$

$= x_k - Q^{-1}c$

$$\min_x \; \frac{1}{2}||\tilde{g}(x, x_k)||^2$$

$$\frac{1}{2}||\tilde{g}(x,x_k)||^2 = \frac{1}{2}\Big[\overbrace{||g(x_k)||^2}^{b} + 2\,(x-x_k)^T \overbrace{J^T(x_k)\,g(x_k)}^{c} + (x-x_k)^T \underbrace{J^T(x_k)\,J(x_k)}_{Q}\,(x-x_k)\Big]$$

$$x_+ = x_k - \underbrace{\left(J^T(x_k)\,J(x_k)\right)^{-1} J(x_k)\,g(x_k)}_{Q^{-1}c}.$$

Notes:

The relevant [CZ13]chapter for this lecture is Chapter 9.

References:

- [Bert09] D. Bertsimas. Lecture notes on optimization methods (6.255). MIT OpenCourseWare, 2009.

- [Bert03] D.P. Bertsekas. Nonlinear Programming. Second edition. Athena Scientific, 2003.

- [CZ13] E.K.P. Chong and S.H. Zak. An Introduction to Optimization. Fourth edition. Wiley, 2013.

- [Tit13] A.L. Tits. Lecture notes on optimal control. University of Maryland, 2013.