THE UNIVERSITY OF
CHICAGO

Lecture 4

January 13, 2011

3.3 DEALING WITH INDEFINITE HESSIANS MATRICES

Closest Positive Definite Matrix

- But Hessian is positive definite (maybe) ONLY at solution!! What do we do?
- Answer: Perturb the matrix.
- Frobenius NORM
- Closest Positive Definite Matrix (symmetric A)

$$\|A\|_F = \sqrt{\sum_{i,j=1}^{n} |a_{ij}|^2} = \sqrt{tr(A^* A)} = \sqrt{\sum_{i=1}^{n} \sigma_i^2}$$

$$A = A^T \Rightarrow \|A\|_F = \sqrt{\sum_{i,j=1}^{n} |a_{ij}|^2} = \sqrt{tr(A^2)} = \sqrt{\sum_{i=1}^{n} \lambda_i^2}$$

$$Q_1^T Q_1 = Q_2^T Q_2 \Rightarrow \|Q_1 A Q_2\|_F = \|A\|_F$$

$$A = QDQ^T \longrightarrow A_1 = QBQ^T$$

$$B = \begin{cases} \lambda_i & \lambda_i \geq \delta > 0 \\ \delta & \lambda_i < \delta \end{cases}$$

Modifying Hessian

Given initial point x_0;

for $k = 0, 1, 2, \ldots$

Factorize the matrix $B_k = \nabla^2 f(x_k) + E_k$, where $E_k = 0$ if $\nabla^2 f(x_k)$ is sufficiently positive definite; otherwise, E_k is chosen to ensure that B_k is sufficiently positive definite;

Solve $B_k p_k = -\nabla f(x_k)$;

Set $x_{k+1} \leftarrow x_k + \alpha_k p_k$, where α_k satisfies the Wolfe, Goldstein, or Armijo backtracking conditions;

end

1.Adding Multiple of the Identity

Algorithm 3.3 (Cholesky with Added Multiple of the Identity).

Choose $\beta > 0$;
if $\min_i a_{ii} > 0$
    set $\tau_0 \leftarrow 0$;
else
    $\tau_0 = -\min(a_{ii}) + \beta$;
end (if)
for $k = 0, 1, 2, \ldots$
    Attempt to apply the Cholesky algorithm to obtain $LL^T = A + \tau_k I$;
    if the factorization is completed successfully
        stop and return L;
    else
        $\tau_{k+1} \leftarrow \max(2\tau_k, \beta)$;
    end (if)
end (for)

- Q: what may be the downside of the approach?

2. Modified Cholesky

for $j = 1, 2, \ldots, n$
$c_{jj} \leftarrow a_{jj} - \sum_{s=1}^{j-1} d_s l_{js}^2;$
$d_j \leftarrow c_{jj};$
for $i = j+1, \ldots, n$
$c_{ij} \leftarrow a_{ij} - \sum_{s=1}^{j-1} d_s l_{is} l_{js};$
$l_{ij} \leftarrow c_{ij}/d_j;$
end
end

$$B_k d = -\nabla f(x_k) \quad \Leftrightarrow LDL^T d = \nabla f(x_k)$$
$$B_k = \nabla^2_{xx} f(x_k) + E_k$$

•Ensuring Quality of the Modified Factorization (i.e. entries do not blow up by division to smal l elelments)
•AIM:

$$d_j \geq \delta, \qquad |m_{ij}| \leq \beta, \qquad i = j+1, j+2, \ldots, n,$$

•Solution: Once a “too small d” is encountered Replace its value by :

$$d_j = \max\left(|c_{jj}|, \left(\frac{\theta_j}{\beta}\right)^2, \delta\right), \quad \text{with } \theta_j = \max_{j<i\leq n} |c_{ij}|$$

•Then:

$$|m_{ij}| = |l_{ij}\sqrt{d_j}| = \frac{|c_{ij}|}{\sqrt{d_j}} \leq \frac{|c_{ij}|\beta}{\theta_j} \leq \beta, \quad \text{for all } i > j$$

• Q: Cholesky does not need pivoting. But does it make sense here to NOT pivot?

LDL factorization WITH permutation (why?)

- EXPAND

3. Modified LDLT (maybe most practical to implement ?)

- What seems to be a practical perturbation to PD that makes it have smallest eigenvalue Delta?
- Solution: Keep same L,P, modify only the B!

$$PAP^T = LBL^T$$

I will ask you to code it with Armijo

$$F = Q\,\mathrm{diag}(\tau_i)\,Q^T, \qquad \tau_i = \begin{cases} 0, & \lambda_i \geq \delta, \\ \delta - \lambda_i, & \lambda_i < \delta, \end{cases} \qquad i = 1, 2, \ldots, n,$$

$$P(A+E)P^T = L(B+F)L^T, \quad \text{where } E = P^T L F L^T P.$$

3.4 QUASI-NEWTON METHODS

3.4 QUASI-NEWTON METHODS: ESSENTIALS

Secant Method – Derivation (NLE)

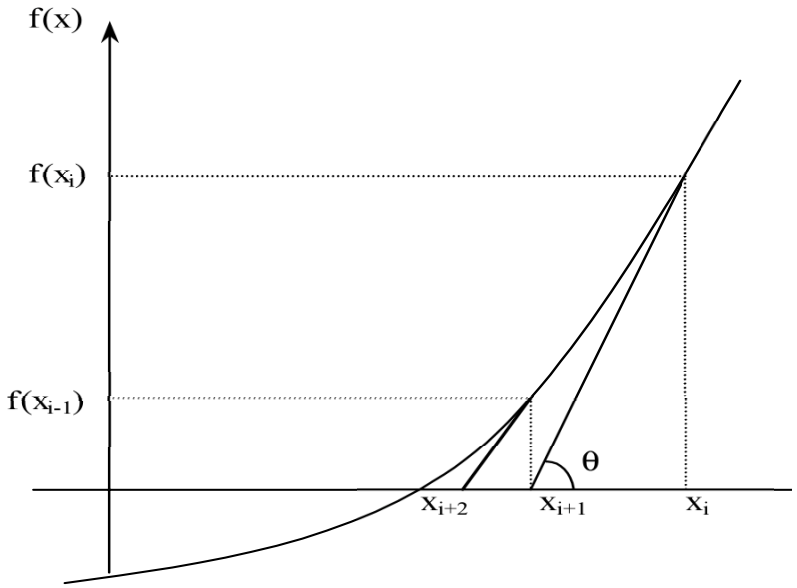


Figure 1 Geometrical illustration of the Newton-Raphson method.

$$f(x) = 0$$

Newton's Method

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \qquad (1)$$

Approximate the derivative

$$f'(x_i) = \frac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}} \qquad (2)$$

Substituting Equation (2) into Equation (1) gives the Secant method

$$x_{i+1} = x_i - \frac{f(x_i)(x_i - x_{i-1})}{f(x_i) - f(x_{i-1})}$$

Secant Method – Derivation

The secant method can also be derived from geometry:

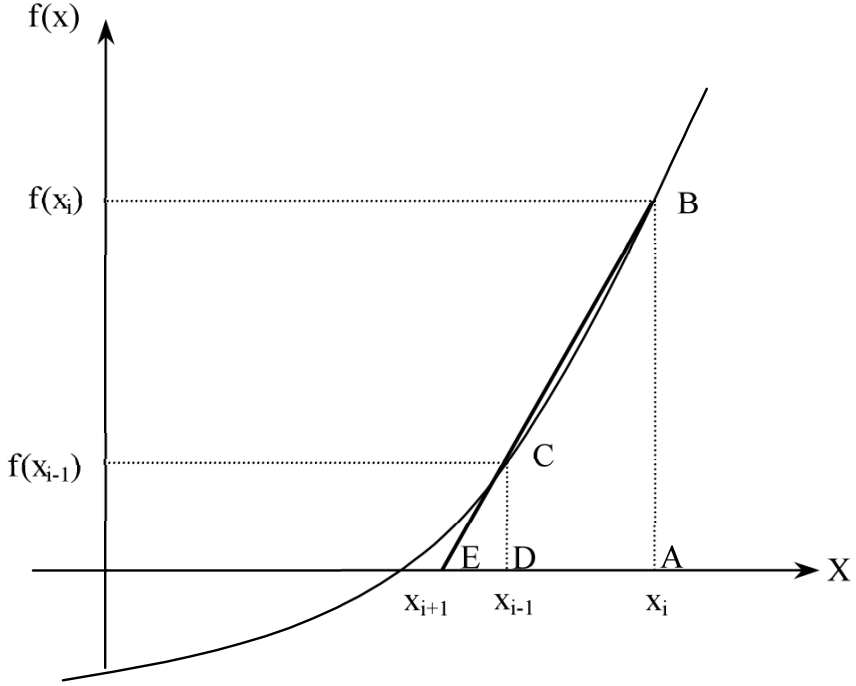


Figure 2 Geometrical representation of the Secant method.

The Geometric Similar Triangles

$$\frac{AB}{AE} = \frac{DC}{DE}$$

can be written as

$$\frac{f(x_i)}{x_i - x_{i+1}} = \frac{f(x_{i-1})}{x_{i-1} - x_{i+1}}$$

On rearranging, the secant method is given as

$$x_{i+1} = x_i - \frac{f(x_i)(x_i - x_{i-1})}{f(x_i) - f(x_{i-1})}$$

Multidimensional Secant Condtions.

Given two points x_k and x_{k+1} , we define (for an optimization problem)

$g_k = \nabla f(x_k)$ and $g_{k+1} = \nabla f(x_{k+1})$

Further, let $p_k = x_{k+1} - x_k$, then

$$g_{k+1} - g_k \approx H(x_k)\, p_k$$ ← The Secant Condition

If the Hessian is constant, then

$$g_{k+1} - g_k = H\, p_k \quad \text{which can be rewritten as} \quad q_k = H\, p_k$$

If the Hessian is constant, then the following condition would hold as well

$$H^{-1}_{k+1}\, q_i = p_i \qquad 0 \le i \le k$$

This is called the quasi-Newton condition.

Broyden–Fletcher–Goldfarb–Shanno

Remember that $\mathbf{q}_i = \mathbf{H}_{k+1}\,\mathbf{p}_i$ and $\mathbf{H}^{-1}{}_{k+1}\,\mathbf{q}_i = \mathbf{p}_i$ (or, $\mathbf{B}_{k+1}\,\mathbf{q}_i = \mathbf{p}_i$) $\quad 0 = i = k$

Both equations have exactly the same form, except that $\mathbf{q}_i$ and $\mathbf{p}_i$ are interchanged and $\mathbf{H}$ is replaced by $\mathbf{B}$ (or vice versa).

This leads to the observation that any update formula for $\mathbf{B}$ can be transformed into a corresponding complimentary formula for $\mathbf{H}$ by interchanging the roles of $\mathbf{B}$ and $\mathbf{H}$ and of $\mathbf{q}$ and $\mathbf{p}$. The reverse is also true.

Broyden–Fletcher–Goldfarb–Shanno formula update of $\mathbf{H}_k$ is obtained by taking the complimentary formula of the DFP formula, thus:

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{\mathbf{q}_k\mathbf{q}_k{}^T}{\mathbf{q}_k{}^T\mathbf{p}_k} - \frac{\mathbf{H}_k\mathbf{p}_k\mathbf{p}_k{}^T\mathbf{H}_k}{\mathbf{p}_k{}^T\mathbf{H}_k\mathbf{p}_k}$$

By taking the inverse, the BFGS update formula for $\mathbf{B}_{k+1}$ (i.e., $\mathbf{H}^{-1}{}_{k+1}$) is obtained:

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \left(\frac{1 + \mathbf{q}_k{}^T\mathbf{B}_k\mathbf{q}_k}{\mathbf{q}_k{}^T\mathbf{p}_k}\right)\frac{\mathbf{p}_k\mathbf{p}_k{}^T}{\mathbf{p}_k{}^T\mathbf{q}_k} - \frac{\mathbf{p}_k\mathbf{q}_k{}^T\mathbf{B}_k + \mathbf{B}_k\mathbf{q}_k\mathbf{p}_k{}^T}{\mathbf{q}_k{}^T\mathbf{p}_k}$$

Advantage of quasi-Newton

- Matrix is ALWAYS positive definite, so line search works fine.
- It needs ONLY gradient information.
- It behaves *almost* like Newton in the limit (convergence is superlinear).
- In its L-BFGS variant it is the workhorse of weather forecast and operational data assimilation in general (a max likelihood procedure, really).

3.4.2 QUASI-NEWTON METHODS: EXTRAS

Background

Assumption: the evaluation of the Hessian is impractical or costly.

- Central idea underlying quasi-Newton methods is to use an approximation of the inverse Hessian based on THE NONLINEAR EQUATION SECANT INTERPRETATION.
- Form of approximation differs among methods.

- The quasi-Newton methods that build up an approximation of the inverse Hessian are often regarded as the most sophisticated for solving unconstrained problems.

Question: What is the simplest approximation?

Modified Newton Method

The Modified Newton method for finding an extreme point is

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \mathbf{S}_k \nabla y(\mathbf{x}_k)$$

Note that:

if $\mathbf{S}_k = \mathbf{I}$, then we have the method of steepest descent

if $\mathbf{S}_k = \mathbf{H}^{-1}(\mathbf{x}_k)$ and $\alpha = 1$, then we have the "pure" Newton method

if $y(\mathbf{x}) = 0.5\, \mathbf{x}^T\mathbf{Q}\mathbf{x} - \mathbf{b}^T\mathbf{x}$, then $\mathbf{S}_k = \mathbf{H}^{-1}(\mathbf{x}_k) = \mathbf{Q}$ (quadratic case)

Classical Modified Newton's Method:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \mathbf{H}^{-1}(\mathbf{x}_0) \nabla y(\mathbf{x}_k)$$

Note that the Hessian is only evaluated at the initial point $\mathbf{x}_0$.

Question: What is a measure of effectiveness for the Classical Modified Newton Method?

Quasi-Newton Methods

In quasi-Newton methods, instead of the true Hessian, an initial matrix H_0 is chosen (usually $H_0 = I$) which is subsequently updated by an update formula:

$$H_{k+1} = H_k + H_k^u$$

where H_k^u is the update matrix.

This updating can also be done with the inverse of the Hessian H^{-1} as follows:

Let $B = H^{-1}$; then the updating formula for the inverse is also of the form

$$B_{k+1} = B_k + B_k^u$$

Big question: What is the update matrix?

Rank One and Rank Two Updates

Let $B = H^{-1}$, then the quasi-Newton condition becomes $B_{k+1}\, q_i = p_i \qquad 0 \le i \le k$
Substitute the updating formula $B_{k+1} = B_k + B^u_k$ and the condition becomes

$$p_i = B_k\, q_i + B^u_k\, q_i \qquad (1)$$

(remember: $p_i = x_{i+1} - x_i$ and $q_i = g_{i+1} - g_i$)

Note: There is no unique solution to funding the update matrix B^u_k

A general form is $B^u_k = a\, uu^T + b\, vv^T$

where a and b are scalars and u and v are vectors satisfying condition (1).

The quantities auu^T and bvv^T are symmetric matrices of (at most) rank one.

Quasi-Newton methods that take $b = 0$ are using rank one updates.
Quasi-Newton methods that take $b \neq 0$ are using rank two updates.

Note that $b \neq 0$ provides more flexibility.

Update Formulas

Rank one updates are simple, but have limitations.

Rank two updates are the most widely used schemes.

The rationale can be quite complicated (see, e.g., Luenberger).

The following two update formulas have received wide acceptance:

- Davidon -Fletcher-Powell (DFP) formula
- Broyden-Fletcher-Goldfarb-Shanno (BFGS) formula.

Davidon-Fletcher-Powel Formula

- Earliest (and one of the most clever) schemes for constructing the inverse Hessian was originally proposed by Davidon (1959) and later developed by Fletcher and Powell (1963).
- It has the interesting property that, for a quadratic objective, it simultaneously generates the directions of the conjugate gradient method while constructing the inverse Hessian.
- The method is also referred to as the variable metric method (originally suggested by Davidon).

Quasi-Newton condition with rank two update substituted is

$$\mathbf{p}_i = \mathbf{B}_k \mathbf{q}_i + a\,\mathbf{u}\mathbf{u}^T\,\mathbf{q}_i + b\,\mathbf{v}\mathbf{v}^T\,\mathbf{q}_i$$

Set $\mathbf{u} = \mathbf{p}_k$, $\mathbf{v} = \mathbf{B}_k\mathbf{q}_k$ and let $a\mathbf{u}^T\mathbf{q}_k = 1$, $b\mathbf{v}^T\mathbf{q}_k = -1$ to determine a and b.

Resulting Davidon-Fletcher-Powell update formula is

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{\mathbf{p}_k\mathbf{p}_k^T}{\mathbf{p}_k^T\mathbf{q}_k} - \frac{\mathbf{B}_k\mathbf{q}_k\mathbf{q}_k^T\mathbf{B}_k}{\mathbf{q}_k^T\mathbf{B}_k\mathbf{q}_k}$$

Broyden–Fletcher–Goldfarb–Shanno

Remember that $\mathbf{q}_i = \mathbf{H}_{k+1}\,\mathbf{p}_i$ and $\mathbf{H}^{-1}{}_{k+1}\,\mathbf{q}_i = \mathbf{p}_i$ (or, $\mathbf{B}_{k+1}\,\mathbf{q}_i = \mathbf{p}_i$) $\quad 0 = i = k$

Both equations have exactly the same form, except that $\mathbf{q}_i$ and $\mathbf{p}_i$ are interchanged and $\mathbf{H}$ is replaced by $\mathbf{B}$ (or vice versa).

This leads to the observation that any update formula for $\mathbf{B}$ can be transformed into a corresponding complimentary formula for $\mathbf{H}$ by interchanging the roles of $\mathbf{B}$ and $\mathbf{H}$ and of $\mathbf{q}$ and $\mathbf{p}$. The reverse is also true.

Broyden–Fletcher–Goldfarb–Shanno formula update of $\mathbf{H}_k$ is obtained by taking the complimentary formula of the DFP formula, thus:

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{\mathbf{q}_k\mathbf{q}_k{}^T}{\mathbf{q}_k{}^T\mathbf{p}_k} - \frac{\mathbf{H}_k\mathbf{p}_k\mathbf{p}_k{}^T\mathbf{H}_k}{\mathbf{p}_k{}^T\mathbf{H}_k\mathbf{p}_k}$$

By taking the inverse, the BFGS update formula for $\mathbf{B}_{k+1}$ (i.e., $\mathbf{H}^{-1}{}_{k+1}$) is obtained:

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \left(\frac{1 + \mathbf{q}_k{}^T\mathbf{B}_k\mathbf{q}_k}{\mathbf{q}_k{}^T\mathbf{p}_k}\right)\frac{\mathbf{p}_k\mathbf{p}_k{}^T}{\mathbf{p}_k{}^T\mathbf{q}_k} - \frac{\mathbf{p}_k\mathbf{q}_k{}^T\mathbf{B}_k + \mathbf{B}_k\mathbf{q}_k\mathbf{p}_k{}^T}{\mathbf{q}_k{}^T\mathbf{p}_k}$$

Some Comments on Broyden Methods

- Broyden–Fletcher–Goldfarb–Shanno formula is more complicated than DFP, but straightforward to apply
- BFGS update formula can be used exactly like DFP formula.
- Numerical experiments have shown that BFGS formula's performance is superior over DFP formula. Hence, BFGS is often preferred over DFP.

Both DFP and BFGS updates have symmetric rank two corrections that are constructed from the vectors pk and Bkqk. Weighted combinations of these formulae will therefore also have the same properties. This observation leads to a whole collection of updates, know as the Broyden family, defined by:

$$B^{f} = (1 - f)B^{DFP} + fB^{BFGS}$$

where f is a parameter that may take any real value.

Quasi-Newton Algorithm

1. Input x_0, B_0, termination criteria.

2. For any k, set $S_k = - B_k g_k$.

3. Compute a step size a (e.g., by line search on $y(x_k + aS_k)$) and set $x_{k+1} = x_k + aS_k$.

4. Compute the update matrix B^u_k according to a given formula (say, DFP or BFGS) using the values $q_k = g_{k+1} - g_k$, $p_k = x_{k+1} - x_k$, and B_k.

5. Set $B_{k+1} = B_k + B^u_k$.

6. Continue with next k until termination criteria are satisfied.

Note: You do have to calculate the vector of first order derivatives g for each iteration.

Some Closing Remarks

- Both DFP and BFGS methods have theoretical properties that guarantee superlinear (fast) convergence rate and global convergence under certain conditions.

- However, both methods could fail for general nonlinear problems. Specifically,

  • DFP is highly sensitive to inaccuracies in line searches.

  • Both methods can get stuck on a saddle-point. In Newton's method, a saddle-point can be detected during modifications of the (true) Hessian. Therefore, search around the final point when using quasi-Newton methods.

  • Update of Hessian becomes "corrupted" by round-off and other inaccuracies.

- All kind of "tricks" such as scaling and preconditioning exist to boost the performance of the methods.

THE UNIVERSITY OF
CHICAGO

SECTION 4 : Trust Region Methods

Mihai Anitescu

4.1 TRUST REGION FUNDAMENTALS

Trust Region Idea

- Notations

$$f^k = f\left(x^k\right) \quad \nabla f^k = \nabla f\left(x^k\right)$$

- Quadratic Model

$$m_k(p) = f^k + p^T g^k + \frac{1}{2} p^T B^k p$$

- Order of Quadratic Model (Taylor)

$$f\left(x^k + p\right) = f^k + p^T g^k + \frac{1}{2} p^T \nabla^2_{xx} f\left(x^k + tp\right) p \quad t \in [0,1]$$

$$m_k(p) - f\left(x^k + p\right) = \begin{cases} O\left(\|p\|^2\right) \\ O\left(\|p\|^2\right) & B^k = \nabla^2_{xx} f\left(x^k\right) \end{cases}$$

Trust Region Subproblem

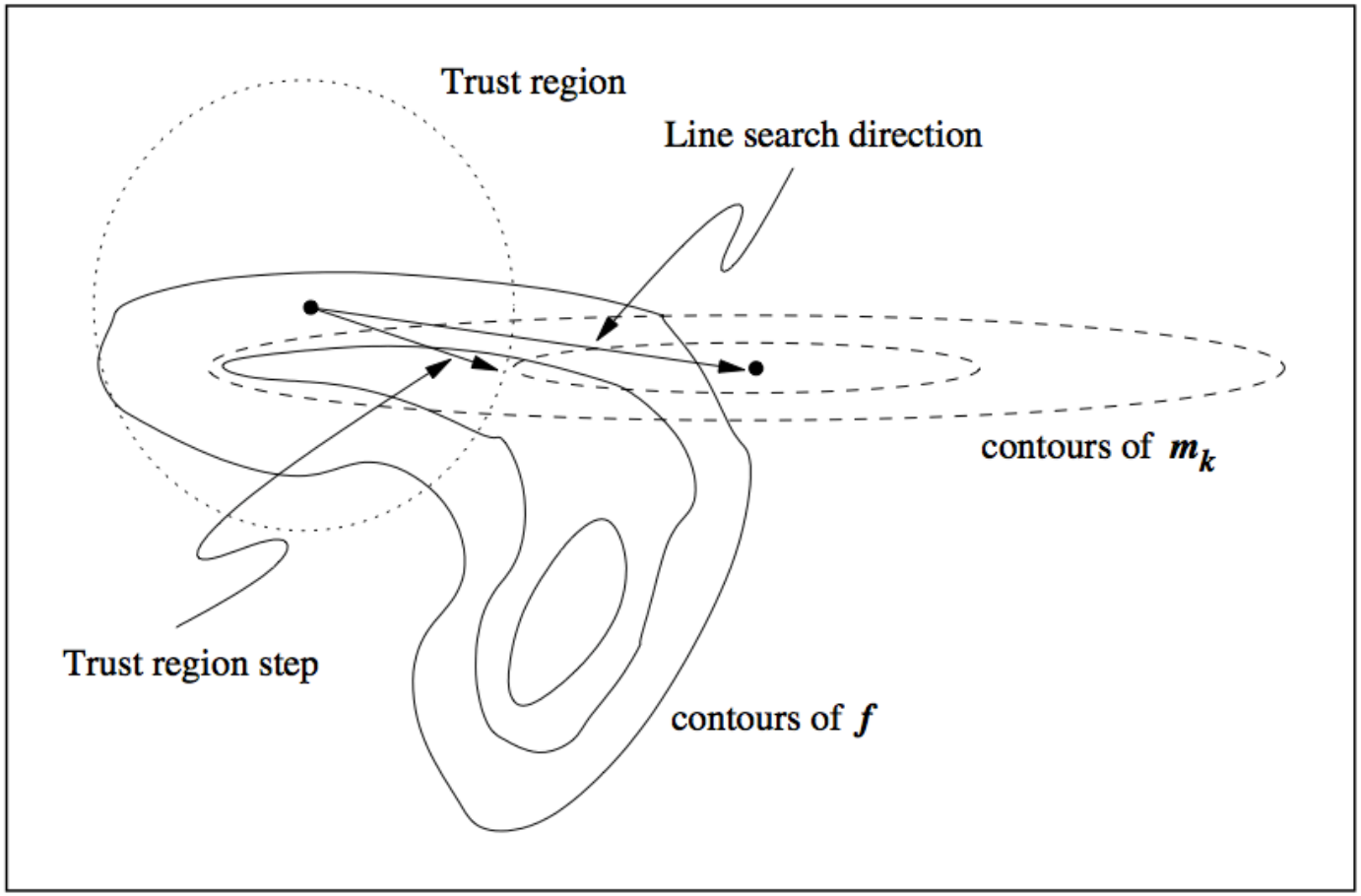
$$\min_{p \in R^n} \quad m_k(p)$$

$$\text{subject to} \quad \|p\| \le \Delta^k$$

Called Trust Region Constraint

- If $B^k \succ 0$ and $p^{*k} = \left(B^k\right)^{-1} g^k$; where $\|p^{*k}\| \le \Delta^k$ then p^k is the solution of the TR subproblem.
- But the interesting case lies in the opposite situation (since not, why would you need the TR in first place)?

Trust Region Geometric Intuition



Example

$$\min_x \left(x^2 - 1\right)^2$$

- Line search started at 0 cannot progress.
- How about the trust-region?

$$\min_d -2d^2; \quad |d| \le \Delta$$

- Either solution will escape the saddle point -- that is the principle of trust-region.

General approach

- How do we solve the TR subproblem?
- If $B^k \succ 0$ (or if we are not obsessed with stopping at saddle points) we use "dogleg" method. (LS, NLE). Most linear algebra is in computing

$$B^k d^{k,U} = -g^k$$

- If fear saddle points, we have to mess around with eigenvalues and eigenvectors – much harder problem.

Trust Region Management: Parameters

- The quality of the reduction.

$$\rho^k = \frac{f\left(x^k\right) - f\left(x^k + p^k\right)}{m_k(0) - m_k\left(p^k\right)}$$

Actual Reduction

Predicted Reduction

- Define the acceptance ratio

$$\eta \in \left[0, \frac{1}{4}\right)$$

- Define the maximum TR size

$$\hat{\Delta}; \quad \Delta \in \left[0, \hat{\Delta}\right)$$

TR management

I will ask you to code It with dogleg

Algorithm 4.1 (Trust Region).

Given $\hat{\Delta} > 0$, $\Delta_0 \in (0, \hat{\Delta})$, and $\eta \in [0, \frac{1}{4})$:

for $k = 0, 1, 2, \ldots$

Obtain p_k by (approximately) solving (4.3);

Evaluate ρ_k from (4.4);

if $\rho_k < \frac{1}{4}$

$\Delta_{k+1} = \frac{1}{4}\Delta_k$

else

if $\rho_k > \frac{3}{4}$ and $\|p_k\| = \Delta_k$

$\Delta_{k+1} = \min(2\Delta_k, \hat{\Delta})$

else

$\Delta_{k+1} = \Delta_k$;

if $\rho_k > \eta$

$x_{k+1} = x_k + p_k$

else

What if I cannot solve the TR exactly ?

- Since it is a hard problem.
- Will this destroy the "Global" convergence behavior?
- Idea: Accept a "sufficient" reduction.
- But, I have no Armijo (or Wolfe, Goldshtein criterion) …
- What do I do?
- Idea? Solve a simple TR problem that creates the yardstick for acceptance – the Cauchy point.

4.2 THE CAUCHY POINT

The Cauchy Point

- What is an easy model to solve? Linear model

$$l_k(p) = f^k + g^{k,T} p$$

- Solve TR linear model

$$p^{k,s} = \arg\min_{p \in R^n \quad \|p\| \le \Delta^k} l_k(p)$$

- The Cauchy point.

$$\tau^k = \arg\min_{\tau \in R \quad \|\tau p^{k,s}\| \le \Delta^k} m_k\left(\tau p^{k,s}\right)$$

$$p^{k,c} = \tau^k p^{k,s}; \quad x^{k,c} = x^k + p^{k,c}$$

- The reduction $m(0) - m\left(p^{k,c}\right)$ becomes my yardstick; if trust region has at least this decrease, I can guarantee "global" convergence (reduction is $o\left(\|g^k\|^2\right)$)

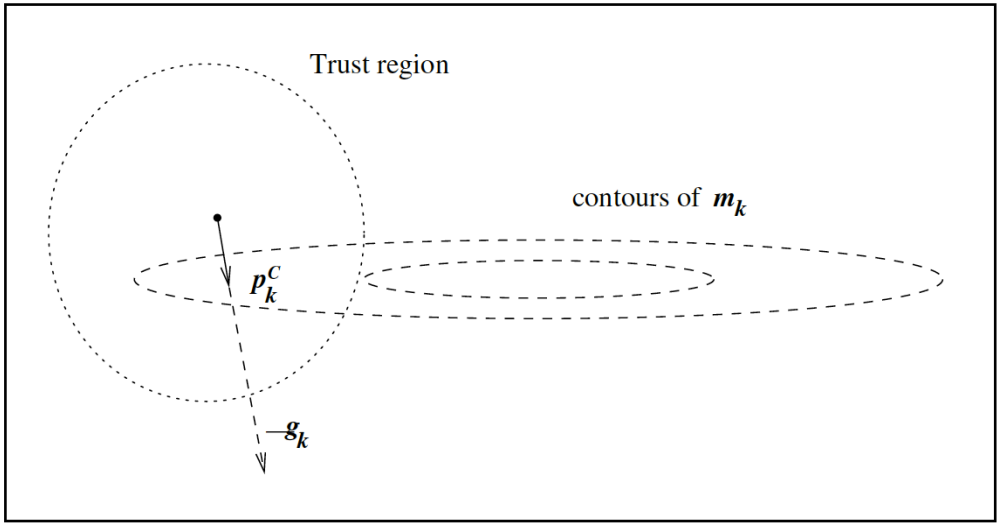
Cauchy Point Solution

- First, solution of the linear problem is

$$p_k^s = -\frac{\Delta^k}{\|g^k\|} g^k$$

- Then, it immediately follows that

$$\tau_k = \begin{cases} 1 & g_k^T B_k g_k \le 0 \\ \min\left(\frac{\|g_k\|^3}{\left(g_k^T B_k g_k\right)\Delta_k}, 1 \right) & \text{otherwise} \end{cases}$$



Dogleg Methods: Improve CP

- If Cauchy point is on the boundary I have a lot of decrease and I accept it (e.g if $g^{k,T} B_k g^k > 0;$)
- If Cauchy point is interior,

$$g^{k,T} B_k g^k > 0; \quad p^{k,c} = -\frac{\|g_k\|^2}{g^{k,T} B_k g^k} g^k$$

- Take now "Newton" step $p^B = -B_k^{-1} g^k$ (note, B need not be pd, all I need is nonsingular).

Dogleg Method Continued

I will ask you to code it with TR

- Define dogleg path

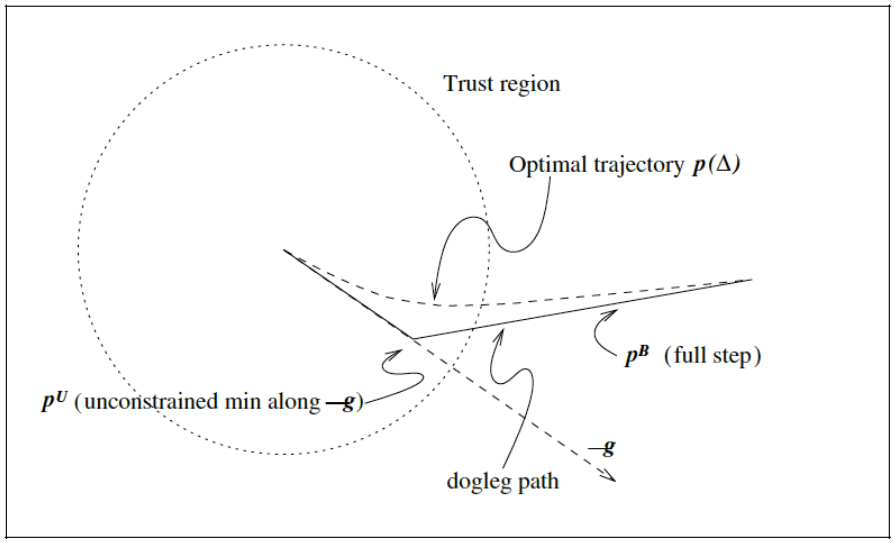
$$\tilde{p}(\tau)=\begin{cases} \tau p^{k,c} & \tau \le 1 \\ p^{k,c}+(\tau-1)\left(p^{B}-p^{k,c}\right) & 1 \le \tau \le 2 \end{cases}$$

- The dogleg point:

$$\tilde{p}(\tau_D);\quad \tau_D = \arg\min_{\tau;\|\tilde{p}(\tau)\|\le\Delta_k} m_k\left(\tilde{p}(\tau)\right)$$

- It is obtained by solving 2 quadratics.
- Sufficiently close to the solution it allows me to choose the Newton step, $\tau = 2$ and thus quadratic convergence.

Dogleg Method: Theory



Lemma 4.2.

Let B be positive definite. Then

(i) $\|\tilde{p}(\tau)\|$ *is an increasing function of* τ, *and*

(ii) $m(\tilde{p}(\tau))$ *is a decreasing function of* τ.

Global Convergence of CP Methods

Lemma 4.3.

The Cauchy point p_k^c satisfies (4.20) with $c_1 = \frac{1}{2}$, that is,

$$m_k(0) - m_k(p_k^c) \geq \tfrac{1}{2}\|g_k\| \min\left(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right).$$

$$\|p_k\| \leq \gamma \Delta_k, \qquad \text{for some constant } \gamma \geq 1. \tag{4.25}$$

$$m_k(0) - m_k(p_k) \geq c_1 \|g_k\| \min\left(\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right), \tag{4.20}$$

Theorem 4.5.

Let $\eta = 0$ in Algorithm 4.1. Suppose that $\|B_k\| \leq \beta$ for some constant β, that f is bounded below on the level set S defined by (4.24) and Lipschitz continuously differentiable in the neighborhood $S(R_0)$ for some $R_0 > 0$, and that all approximate solutions of (4.3) satisfy the inequalities (4.20) and (4.25), for some positive constants c_1 and γ. We then have

$$\liminf_{k \to \infty} \|g_k\| = 0. \tag{4.26}$$

Numerical comparison between methods

- What is a fair comparison between methods?
- Probably : starting from same point 1) number of function evaluations and 2) number of linear systems (the rest depends too much on the hardware and software platform). I will ask you to do this.
- Trust region tends to use fewer function evaluations (the modern preferred metric;) than line search .
- Also dogleg does not force positive definite matrix, so it has fewer chances of stopping at a saddle point, (but it is not guaranteed either).

4.3 GENERAL CASE: SOLVING THE ACTUAL TR PROBLEM (DOGLEG DOES NOT QUITE DO IT)

Trust Region Equation

Theorem 4.1.

The vector p^* is a global solution of the trust-region problem

$$\min_{p \in \mathbb{R}^n} m(p) = f + g^T p + \tfrac{1}{2} p^T B p, \quad \text{s.t. } \|p\| \leq \Delta, \tag{4.7}$$

if and only if p^* is feasible and there is a scalar $\lambda \geq 0$ such that the following conditions are satisfied:

$$(B + \lambda I) p^* = -g, \tag{4.8a}$$

$$\lambda(\Delta - ||p^*||) = 0, \tag{4.8b}$$

$$(B + \lambda I) \quad \text{is positive semidefinite.} \tag{4.8c}$$

Theory of Trust Region Problem

Global convergence away from saddle point

Theorem 4.8.

Suppose that the assumptions of Theorem 4.6 are satisfied and in addition that f is twice continuously differentiable in the level set S. Suppose that $B_k = \nabla^2 f(x_k)$ for all k, and that the approximate solution p_k of (4.3) at each iteration satisfies (4.52) for some fixed $\gamma > 0$. Then $\lim_{k\to\infty} \|g_k\| = 0$.

If, in addition, the level set S of (4.24) is compact, then either the algorithm terminates at a point x_k at which the second-order necessary conditions (Theorem 2.3) for a local solution hold, or else $\{x_k\}$ has a limit point x^* in S at which the second-order necessary conditions hold.

Fast Local Convergence

Theorem 4.9.

Let f be twice Lipschitz continuously differentiable in a neighborhhod of a point x^* at which second-order sufficient conditions (Theorem 2.4) are satisfied. Suppose the sequence $\{x_k\}$ converges to x^* and that for all k sufficiently large, the trust-region algorithm based on (4.3) with $B_k = \nabla^2 f(x_k)$ chooses steps p_k that satisfy the Cauchy-point-based model reduction criterion (4.20) and are asymptotically similar to Newton steps p_k^N whenever $\|p_k^N\| \leq \frac{1}{2}\Delta_k$, that is,

$$\|p_k - p_k^N\| = o(\|p_k^N\|). \tag{4.53}$$

Then the trust-region bound Δ_k becomes inactive for all k sufficiently large and the sequence $\{x_k\}$ converges superlinearly to x^*.

How do we solve the subproblem?

- Very sophisticated approach based on theorem on structure of TR solution, eigenvalue analysis and/or an "inner" Newton iteration.
- Foundation: Find Solution for

$$p(\lambda) = -(B + \lambda I)^{-1} g$$

$$\| p(\lambda) \| = \Delta .$$

How do I find such a solution?

$$B = Q\Lambda Q^T \qquad \Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n),$$

$$p(\lambda) = -Q(\Lambda + \lambda I)^{-1} Q^T g = -\sum_{j=1}^{n} \frac{q_j^T g}{\lambda_j + \lambda} q_j,$$

, by orthonormality of $q_1, q_2, \ldots, q_n$

$$\|p(\lambda)\|^2 = \sum_{j=1}^{n} \frac{\left(q_j^T g\right)^2}{(\lambda_j + \lambda)^2}.$$

TR problem has a solution

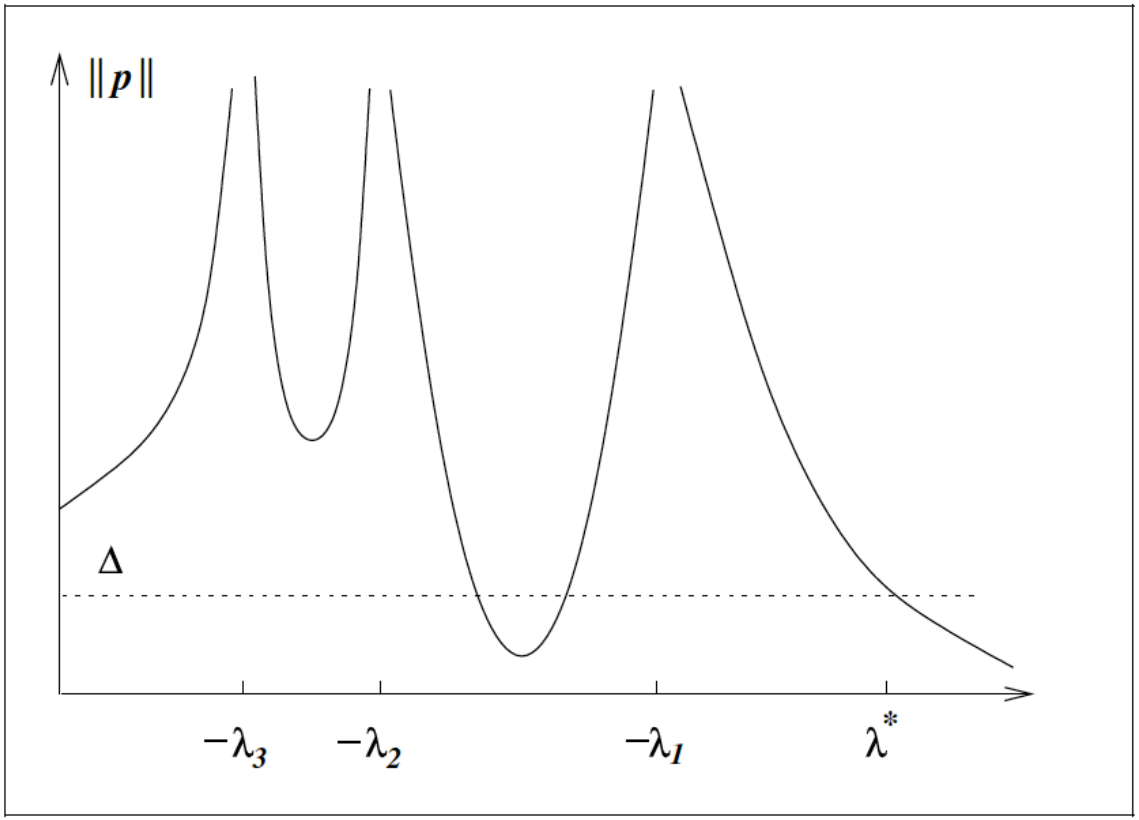


Figure 4.5 $\|p(\lambda)\|$ as a function of λ.

$$\lim_{\lambda \to \infty} \|p(\lambda)\| = 0. \qquad q_j^T g \neq 0 \quad \Rightarrow \quad \lim_{\lambda \to -\lambda_j} \|p(\lambda)\| = \infty.$$

Practical (INCOMPLETE) algorithm

$$\phi_2(\lambda) = \frac{1}{\Delta} - \frac{1}{\|p(\lambda)\|}, \qquad \lambda^{(\ell+1)} = \lambda^{(\ell)} - \frac{\phi_2\left(\lambda^{(\ell)}\right)}{\phi_2'\left(\lambda^{(\ell)}\right)}.$$

Algorithm 4.3 (Trust Region Subproblem).
Given $\lambda^{(0)}$, $\Delta > 0$:
for $\ell = 0, 1, 2, \ldots$
Factor $B + \lambda^{(\ell)} I = R^T R$;
Solve $R^T R p_\ell = -g$, $R^T q_\ell = p_\ell$;

Set

$$\lambda^{(\ell+1)} = \lambda^{(\ell)} + \left(\frac{\|p_\ell\|}{\|q_\ell\|}\right)^2 \left(\frac{\|p_\ell\| - \Delta}{\Delta}\right);$$

end (for).

It generally gives a machine precision solution in 2-3 iterations (Cholesky)

The Hard Case

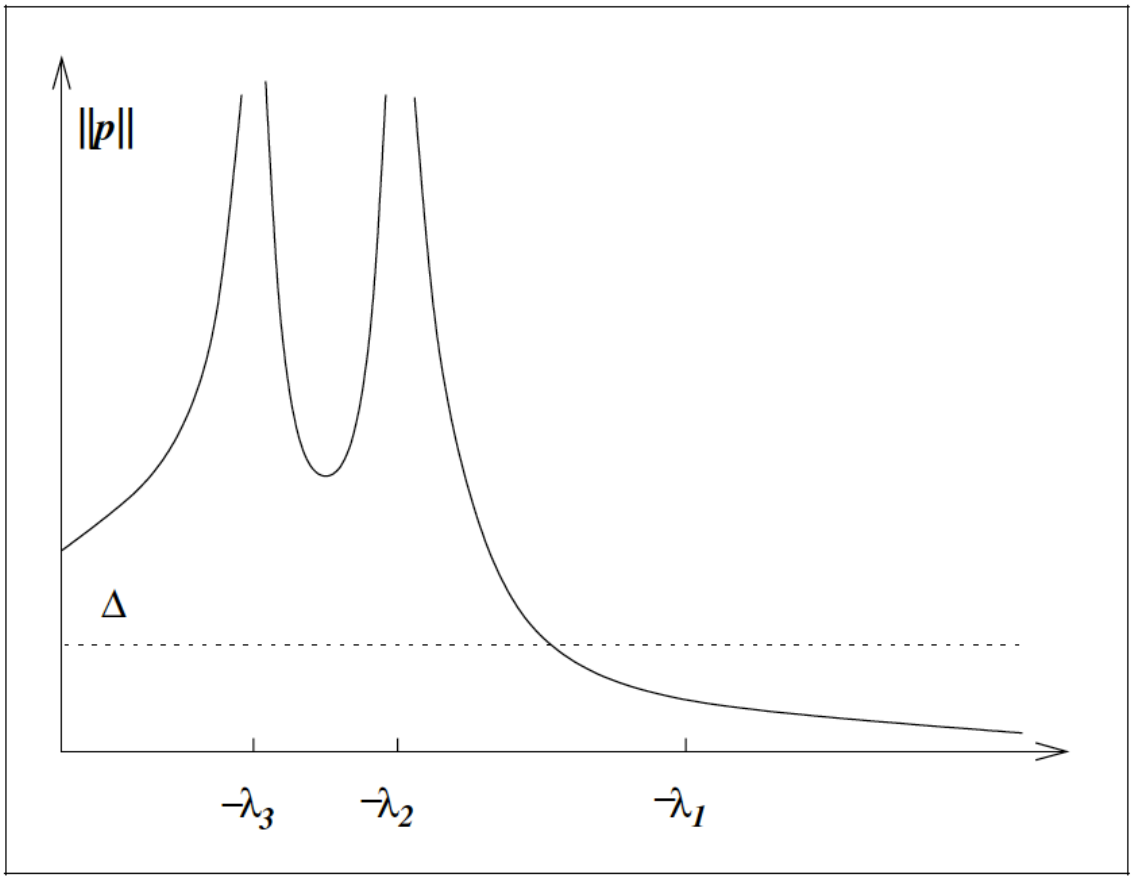


Figure 4.7 The hard case: $\|p(\lambda)\| < \Delta$ for all $\lambda \in (-\lambda_1, \infty)$.

$$q_j^T g = 0$$

$$\lambda = -\lambda_1 \Rightarrow p = \sum_{j:\lambda_j \neq \lambda_1} \frac{q_j^T g}{\lambda_j - \lambda_1} q_j$$

$$p(\tau) = \sum_{j:\lambda_j \neq \lambda_1} \frac{q_j^T g}{\lambda_j - \lambda_1} q_j + \tau q_1$$

$$\exists \tau \quad \|p(\tau)\| = \Delta^k$$

If double root, things continue to be complicated …

Summary and Comparisons

- Line search problems have easier subproblems (if we modify Cholesky).
- But they cannot be guaranteed to converge to a point with positive semidefinite Hessian.
- Trust-region problems can, at the cost of solving a complicated subproblem.
- Dogleg methods leave "between" these two situations.