Advanced computational statistics, lecture 1

Frank Miller, Department of Computer and Information Science,
Linköping University

March 11, 2025

Course schedule

Optimisation

- Topic 1: **Gradient-based optimisation**
- Topic 2: **Stochastic gradient-based optimisation**
- Topic 3: **Gradient free optimisation**
- Topic 4: **Optimisation with constraints**
- Topic 5: **EM algorithm and bootstrap**
- Topic 6: **Simulation of random variables**
- Topic 7: **Numerical and Monte Carlo integration; importance sampling**

Simulation and Integration

Course homepage: http://www.adoptdesign.de/frankmillereu/adcompstat2025.html

Includes schedule, reading material, lecture notes, assignments

Optimisation in statistics

- Maximum Likelihood
- Minimising risk in (Bayesian) decision theory
- Minimising sum of squares (Least Squares Estimate)
- Maximising information in experimental design
- Machine learning
- Common problem in these examples:
  - $\boldsymbol{x}$ p-dimensional vector, $g\colon\ \mathbb{R}^p \to \mathbb{R}$ function
  - We search $\boldsymbol{x}^*$ with $g(\boldsymbol{x}^*) = \max g(\boldsymbol{x})$
  - Typical: $g = \sum_{i=1}^{n} g_i$ with a (large) sample size n with $g_i\colon\ \mathbb{R}^p \to \mathbb{R}$
  - Minimisation problem turns into maximisation by considering $-g$

Least squares estimation (LSE)

- We search a Least Squares estimate $\widehat{\boldsymbol{\beta}}$ for $\boldsymbol{\beta}$ minimising the distance $g(\widehat{\boldsymbol{\beta}}) = \|\widehat{\boldsymbol{y}} - \boldsymbol{y}\|^2$ from $\widehat{\boldsymbol{y}} = \boldsymbol{X}\widehat{\boldsymbol{\beta}}$ to $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$
- $g(\widehat{\boldsymbol{\beta}}) = \left\|\boldsymbol{X}\widehat{\boldsymbol{\beta}} - \boldsymbol{y}\right\|^2 = \left(\boldsymbol{X}\widehat{\boldsymbol{\beta}} - \boldsymbol{y}\right)^T\left(\boldsymbol{X}\widehat{\boldsymbol{\beta}} - \boldsymbol{y}\right) = \widehat{\boldsymbol{\beta}}^T\boldsymbol{X}^T\boldsymbol{X}\widehat{\boldsymbol{\beta}} - \boldsymbol{2}\widehat{\boldsymbol{\beta}}^T\boldsymbol{X}^T\boldsymbol{y} + \boldsymbol{y}^{\boldsymbol{T}}\boldsymbol{y}$
- Setting the derivative to 0 ($\frac{\partial g}{\partial \widehat{\beta}} = 2\boldsymbol{X}^T\boldsymbol{X}\widehat{\boldsymbol{\beta}} - \boldsymbol{2}\boldsymbol{X}^T\boldsymbol{y} = 0$), we get $\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$
- Note that $g(\widehat{\boldsymbol{\beta}}) = \left\|\boldsymbol{X}\widehat{\boldsymbol{\beta}} - \boldsymbol{y}\right\|^2 = \sum_{i=1}^{n}\left(\boldsymbol{x}_i^T\widehat{\boldsymbol{\beta}} - y_i\right)^2 = \sum_{i=1}^{n} g_i(\widehat{\boldsymbol{\beta}})$
- Optimisation problem:
  - $\widehat{\boldsymbol{\beta}}$ p-dimensional vector, $g\colon \mathbb{R}^p \to \mathbb{R}$ function
  - We search $\widehat{\boldsymbol{\beta}}$ with $g(\widehat{\boldsymbol{\beta}}) = \min g(\boldsymbol{b}) = \min \sum_{i=1}^{n} g_i(\boldsymbol{b})$
- Here, we do not need to iteratively compute this minimum since we have an algebraic solution $\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$

Variations of least squares estimation

- Algebraic solution exists for the LSE, but not if we vary the problem
- Lasso estimate: $g(\widehat{\boldsymbol{\beta}}) = \left\|\boldsymbol{X}\widehat{\boldsymbol{\beta}} - \boldsymbol{y}\right\|^2 + \lambda\left\|\widehat{\boldsymbol{\beta}}\right\|_1 = \sum_{i=1}^{n}\left(\boldsymbol{x}_i\widehat{\boldsymbol{\beta}} - y_i\right)^2 + \lambda\left\|\widehat{\boldsymbol{\beta}}\right\|_1 = \sum_{i=1}^{n} g_i(\widehat{\boldsymbol{\beta}})$
- L_1-estimation: $g(\widehat{\boldsymbol{\beta}}) = \left\|\boldsymbol{X}\widehat{\boldsymbol{\beta}} - \boldsymbol{y}\right\|_1 = \sum_{i=1}^{n}\left|\boldsymbol{x}_i\widehat{\boldsymbol{\beta}} - y_i\right| = \sum_{i=1}^{n} g_i(\widehat{\boldsymbol{\beta}})$
- Many further variations of estimates have been considered
- In all cases, we search $\widehat{\boldsymbol{\beta}}$ with $g(\widehat{\boldsymbol{\beta}}) = \min g(\boldsymbol{b}) = \min \sum_{i=1}^{n} g_i(\boldsymbol{b})$

- Recall: Norms for $\boldsymbol{x} = (x_1, \dots, x_p)^T$: $\|\boldsymbol{x}\| = \|\boldsymbol{x}\|_2 = \sqrt{x_1^2 + \cdots + x_p^2}$ (Euclid), $\|\boldsymbol{x}\|_1 = |x_1| + \cdots + |x_p|$, $\|\boldsymbol{x}\|_\infty = \max\{|x_1|, \dots, |x_p|\}$ (max-norm)

Maximum likelihood estimator (MLE)

- The MLE is solution of $g(\widehat{\boldsymbol{\beta}}) = \max g(\boldsymbol{b})$ with
  $g(\widehat{\boldsymbol{\beta}}) = \text{log–likelihood}(\widehat{\boldsymbol{\beta}}, \boldsymbol{X}, \boldsymbol{y}) = \sum_{i=1}^{n} \text{log–likelihood}(\widehat{\boldsymbol{\beta}}, \boldsymbol{x_i}, y_i)$
  (the latter equation requires independence of observations)
- In the simple case of normally distributed observations, MLE=LSE and we have an algebraic solution
- Otherwise, we need usually iterative methods to compute the MLE

- If the data is from an exponential family, the function g is concave ($-g$ is convex)
- Log likelihoods can be non-concave (e.g., Cauchy-distribution)

Maximising information of experimental designs

- Regression model $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$ (where $\boldsymbol{\varepsilon}$ has iid components)
- $\boldsymbol{X}$ design matrix (depends on choice of observational points)
- Covariance matrix of Least Squares estimate $\widehat{\boldsymbol{\beta}}$ is
$$\mathrm{Cov}(\widehat{\boldsymbol{\beta}}) = (\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})^{-\mathbf{1}} \cdot const$$
- Choose design of an experiment such that $\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X}$ “large”
- D-optimality: $g(\text{"}\mathbf{design}\text{"}) = \det(\boldsymbol{X}^{\boldsymbol{T}}\boldsymbol{X})$
- We search $\mathbf{design}^*$ with $g(\mathbf{design}^*) = \max g(\mathbf{design})$

Maximising information of experimental designs

- Regression model $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$, $\text{Cov}(\widehat{\boldsymbol{\beta}}) = (\boldsymbol{X^T X})^{-1} \cdot const$
- We search **design*** with $g(\textbf{design}^*) = \max g(\textbf{design})$
- Example: cubic regression, $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \varepsilon$, n observations in each of following 4 points: $-1, -a, a, 1$. How should $a \in (0,1)$ be chosen?

$$\boldsymbol{X} = \begin{pmatrix} 1 & -1 & 1 & -1 \\ 1 & -a & a^2 & -a^3 \\ 1 & a & a^2 & a^3 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

$$g(a) = \det(\boldsymbol{X^T X}) = \det(\boldsymbol{X}(a)^{\boldsymbol{T}} \boldsymbol{X}(a))$$

- We search a^* with $g(a^*) = \max g(a)$

Today's schedule

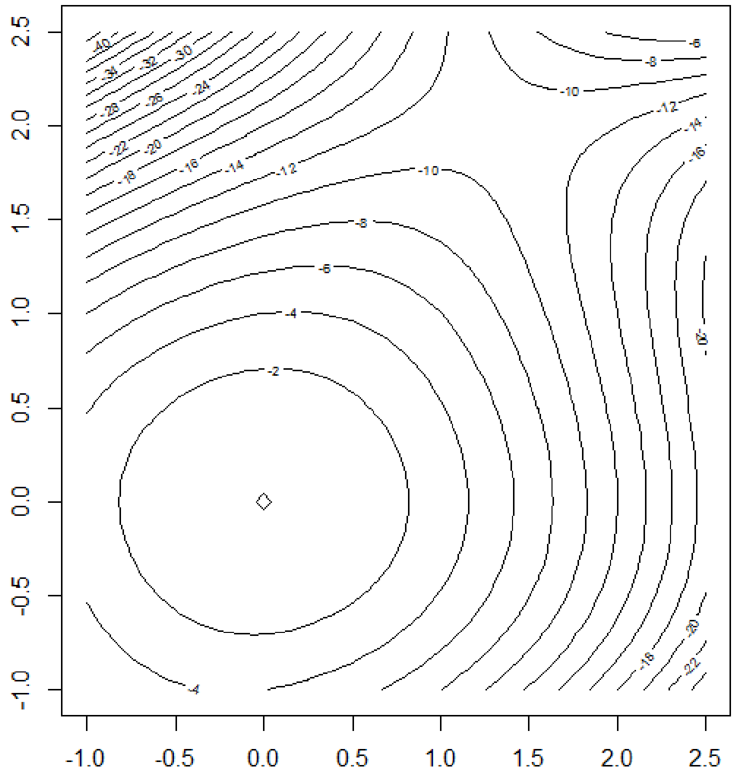
- Analytical optimisation
- Iterative optimisation
  - Bi-section method (univariate optimisation)
  - Convergence speed and stopping criteria
  - Newton
  - Steepest ascent
  - Accelerated steepest ascent
  - Quasi-Newton

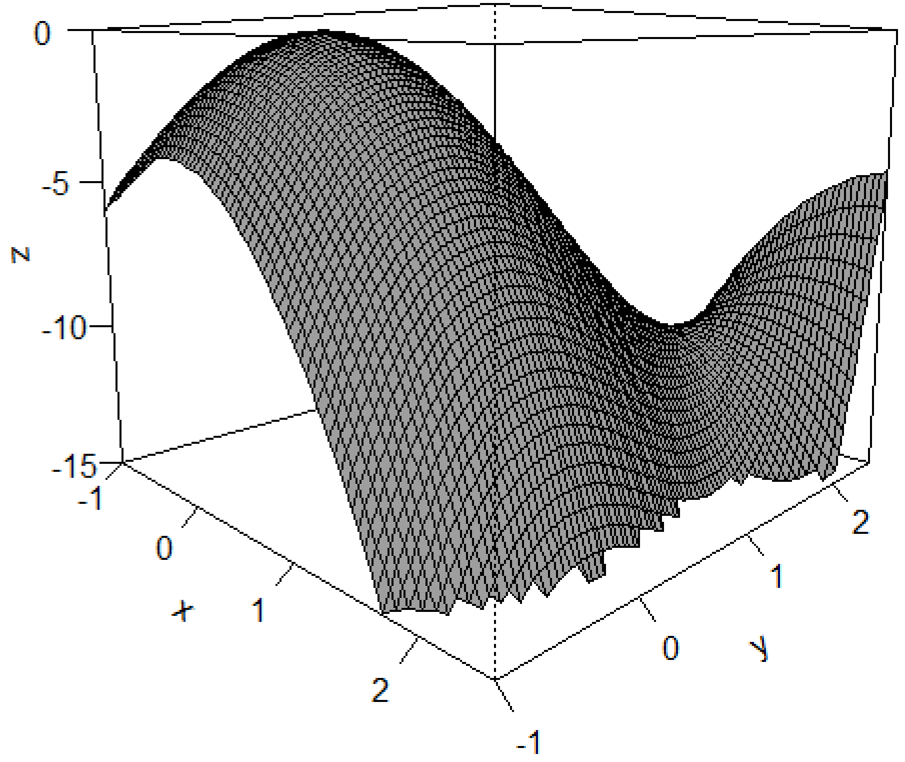
Analytical optimisation – gradient and Hessian

- $\boldsymbol{g}\begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix}$ is a real-valued function
- $\boldsymbol{g}'\begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix} = \begin{pmatrix} \frac{\partial g}{\partial x_1}(\boldsymbol{x}) \\ \vdots \\ \frac{\partial g}{\partial x_p}(\boldsymbol{x}) \end{pmatrix}$ is the gradient, $\boldsymbol{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix}$
- $\boldsymbol{g}''\begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix} = \begin{pmatrix} \frac{\partial g}{\partial x_1 \partial x_1}(\boldsymbol{x}) & \cdots & \frac{\partial g}{\partial x_1 \partial x_p}(\boldsymbol{x}) \\ \vdots & & \vdots \\ \frac{\partial g}{\partial x_1 \partial x_p}(\boldsymbol{x}) & \cdots & \frac{\partial g}{\partial x_p \partial x_p}(\boldsymbol{x}) \end{pmatrix}$ is the Hessian matrix

Bivariate optimisation – visualisation

- $g\begin{pmatrix} x \\ y \end{pmatrix} = -3x^2 - 4y^2 + xy^3$



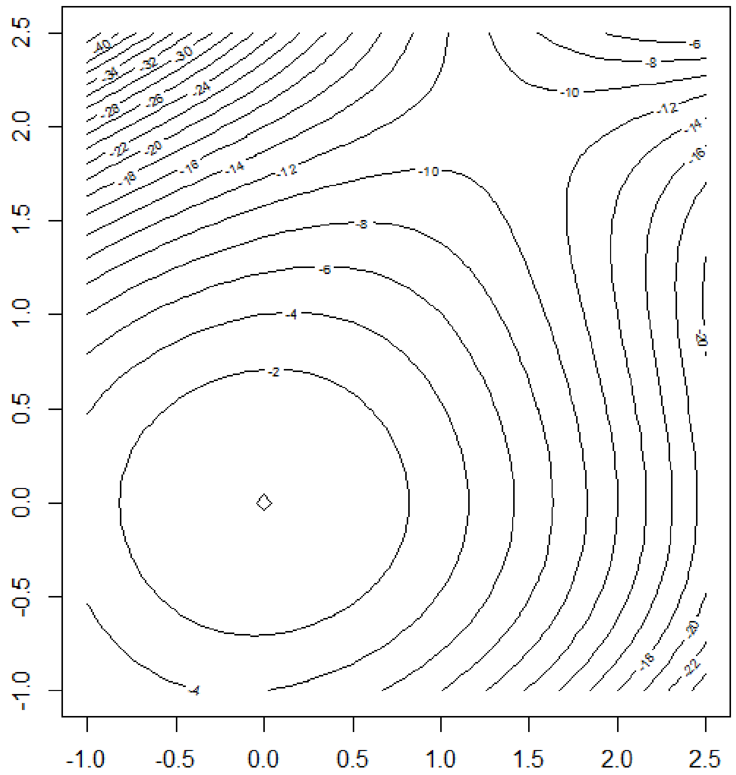


Figures can be drawn using R-core-functions **`contour`** and **`persp`**

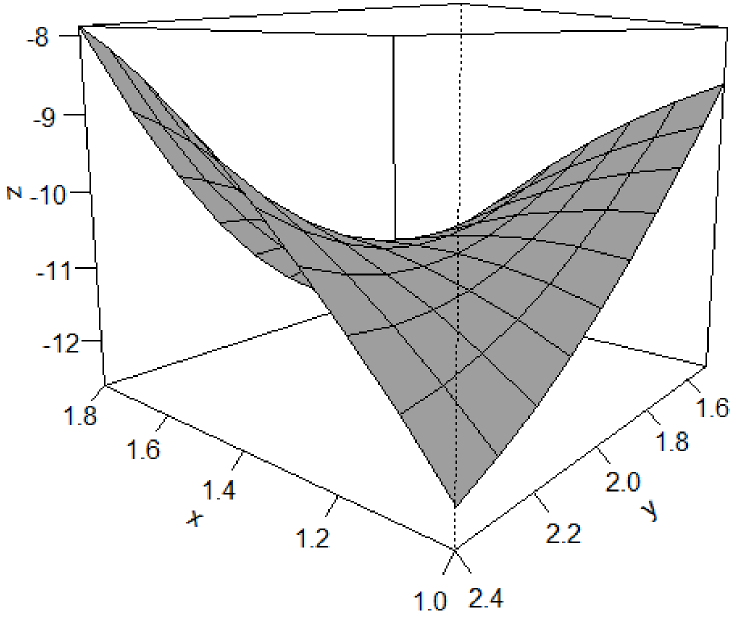
Analytical optimisation

- $\boldsymbol{g}\begin{pmatrix} x \\ y \end{pmatrix} = -3x^2 - 4y^2 + xy^3$
- $\boldsymbol{g}'\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} -6x + y^3 \\ -8y + 3xy^2 \end{pmatrix}$
- $\boldsymbol{g}''\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} -6 & 3y^2 \\ 3y^2 & -8 + 6xy \end{pmatrix}$

- See calculation in following document: `AdvCompStat_AnalytOpt.pdf`
- Maximum at (0,0), saddle point at $(\frac{4}{3}, 2)$



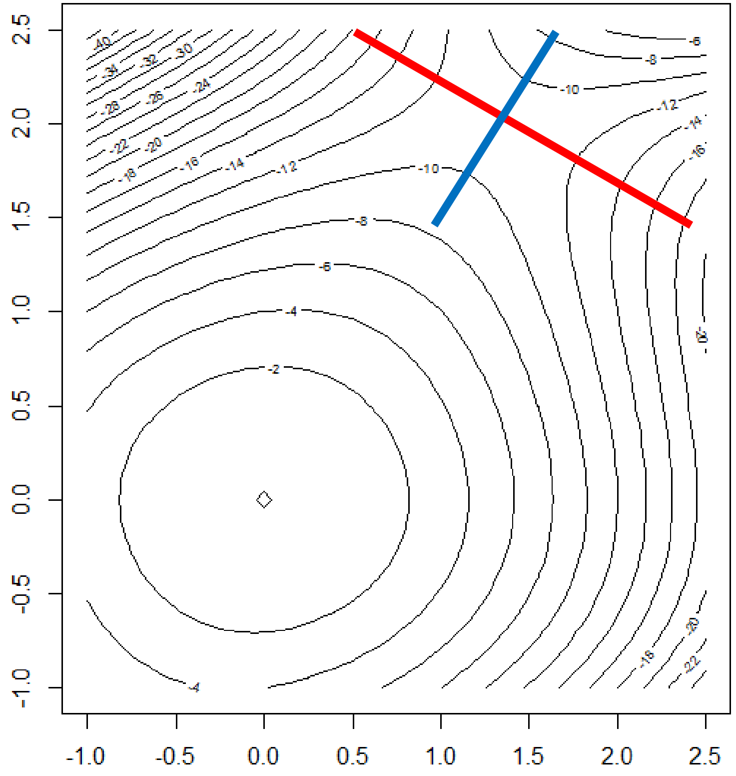
Analytical optimisation – saddle points



Saddle point and eigenvectors of the Hessian

- $\boldsymbol{g}\begin{pmatrix}x\\y\end{pmatrix} = -3x^2 - 4y^2 + xy^3$
- Saddle point at $(\frac{4}{3}, 2)$

- $\boldsymbol{g}'\begin{pmatrix}4/3\\2\end{pmatrix} = \begin{pmatrix}0\\0\end{pmatrix}$
- $\boldsymbol{g}''\begin{pmatrix}4/3\\2\end{pmatrix} = \begin{pmatrix}-6 & 12\\12 & 8\end{pmatrix}$



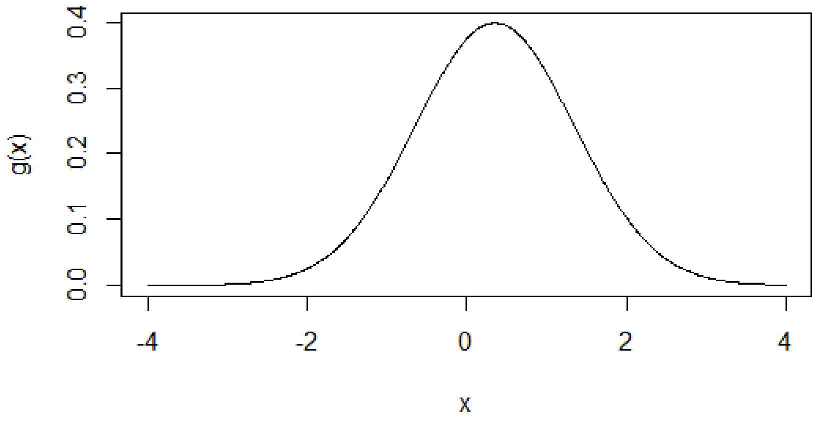
- Eigenvalues 14.89, -12.89; eigenvectors $\begin{pmatrix}0.498\\0.867\end{pmatrix}$, $\begin{pmatrix}-0.867\\0.498\end{pmatrix}$

Today’s schedule

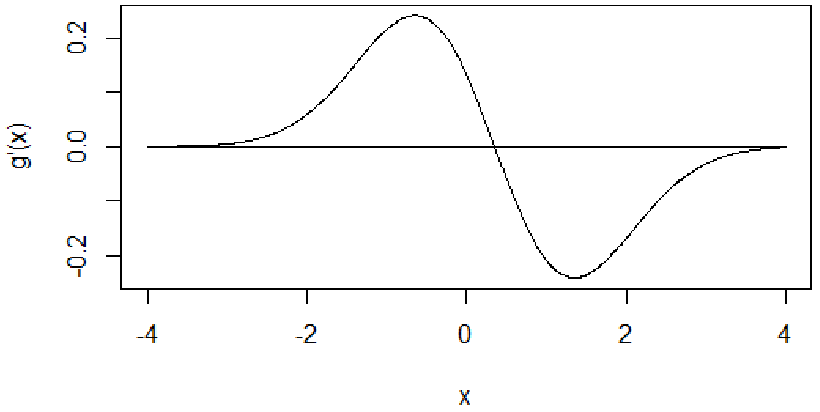
- Analytical optimisation
- Iterative optimisation
  - Bi-section method (univariate optimisation)
  - Convergence speed and stopping criteria
  - Newton
  - Steepest ascent
  - Accelerated steepest ascent
  - Quasi-Newton

Bisection method (univariate optimisation)

- $g: \mathbb{R} \to \mathbb{R}$ continuously differentiable function; search x^* with $g(x^*) = \max g(x)$
- Compute $g'(x)$ and search x^* with $g'(x^*) = 0$
- Improve iteratively approximations for x^*: $x^{(0)}$ -> $x^{(1)}$ -> $x^{(2)}$ -> …



- Choose a and b with $a < b$ such that g' has different signs, $g'(a) \cdot g'(b) < 0$, $t = 0$
- While $b - a > \epsilon$
  - Set $t = t + 1$, set $x^{(t)} = \frac{a+b}{2}$, compute $g'\left(x^{(t)}\right)$
    - If $g'(a) \cdot g'\left(x^{(t)}\right) < 0$, set $b = x^{(t)}$,
    - Otherwise, set $a = x^{(t)}$



Convergence criterion for iterative methods

- Compare $\boldsymbol{x}^{(t)}$ and $\boldsymbol{x}^{(t+1)}$ and stop if they are "close enough"
  - Absolut stopping criterion, $\left\|\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)}\right\| < \epsilon$,
  - Relative stopping criterion, $\left\|\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)}\right\| / \left\|\boldsymbol{x}^{(t+1)}\right\| < \epsilon$,
  - Modified rel. stopping crit., $\frac{\left\|\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)}\right\|}{\left\|\boldsymbol{x}^{(t+1)}\right\| + \varepsilon} < \varepsilon$
  - Different norms $\|\cdot\|$ can be used
- Instead of $\boldsymbol{x}^{(t)}$ and $\boldsymbol{x}^{(t+1)}$, one can compare $g(\boldsymbol{x}^{(t)})$ and $g(\boldsymbol{x}^{(t+1)})$ (but note: not all iterative methods require the calculation of $g(\boldsymbol{x}^{(t)})$ and then, it would add computational time)

Convergence speed of iterative algorithms

- Convergence speed can be quantified by q and c as follows:
  - Let $\varepsilon^{(t)} = \|\boldsymbol{x}^{(t)} - \boldsymbol{x}^*\|$,
  - Find q and c such that $\lim_{t\to\infty} \varepsilon^{(t+1)}/(\varepsilon^{(t)})^q = c$

Convergence order (→ q)

Convergence rate (→ c)

$c \in [0,1)$ for $q = 1$, $c \geq 0$ for $q > 1$

Intuitively, $\varepsilon^{(t+1)} \approx c \cdot (\varepsilon^{(t)})^q$

$$\frac{\|\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(t)} - \boldsymbol{x}^*\|^q} \to c \text{ (for } t \to \infty)$$

- $\varepsilon = 1, 0.5, 0.25, 0.125, 0.063, 0.031, \ldots$ ➡ $q = 1, c = 0.5$,
- $\varepsilon = 1, 0.1, 0.01, 0.001, 0.0001, \ldots$ ➡ $q = 1, c = 0.1$,
- If $q = 1$, we say that convergence is ”linear”

- $\varepsilon = 1, 0.5, 0.125, 0.008, 0.00003, \ldots$ ➡ $q = 2, c = 0.5$.
- If $q = 2$, we say that convergence is ”quadratic”

Today’s schedule

- Analytical optimisation
- Iterative optimisation
  - Bi-section method (univariate optimisation)
  - Convergence speed and stopping criteria
  - Newton
  - Steepest ascent
  - Accelerated steepest ascent
  - Quasi-Newton

Multivariate Taylor and Newton

- Let $f\colon \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function and $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$. Then, the multivariate Taylor expansion for $\boldsymbol{y} \to \boldsymbol{x}$:
$$f(\boldsymbol{y}) = f(\boldsymbol{x}) + \boldsymbol{f}'(\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{x}) + o(\|\boldsymbol{y} - \boldsymbol{x}\|)$$
- Applied to the gradient $\boldsymbol{g}'\colon \mathbb{R}^p \to \mathbb{R}^p$ of a twice cont. diff. function $g\colon \mathbb{R}^p \to \mathbb{R}$,
$$\boldsymbol{g}'(\boldsymbol{y}) = \boldsymbol{g}'(\boldsymbol{x}) + \boldsymbol{g}''(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x}) + o(\|\boldsymbol{y} - \boldsymbol{x}\|)$$
- The multivariate Newton method is motivated by the multivariate Taylor expansion (with $\boldsymbol{x} = \boldsymbol{x}^{(t)}$ and $\boldsymbol{y} = \boldsymbol{x}^*$)
$$0 = \boldsymbol{g}'(\boldsymbol{x}^*) \approx \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right) + \boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)\left(\boldsymbol{x}^* - \boldsymbol{x}^{(t)}\right)$$
- The Newton-iteration works as:
$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \left(\boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)\right)^{-1} \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$$

Univariate Newton(-Raphson)

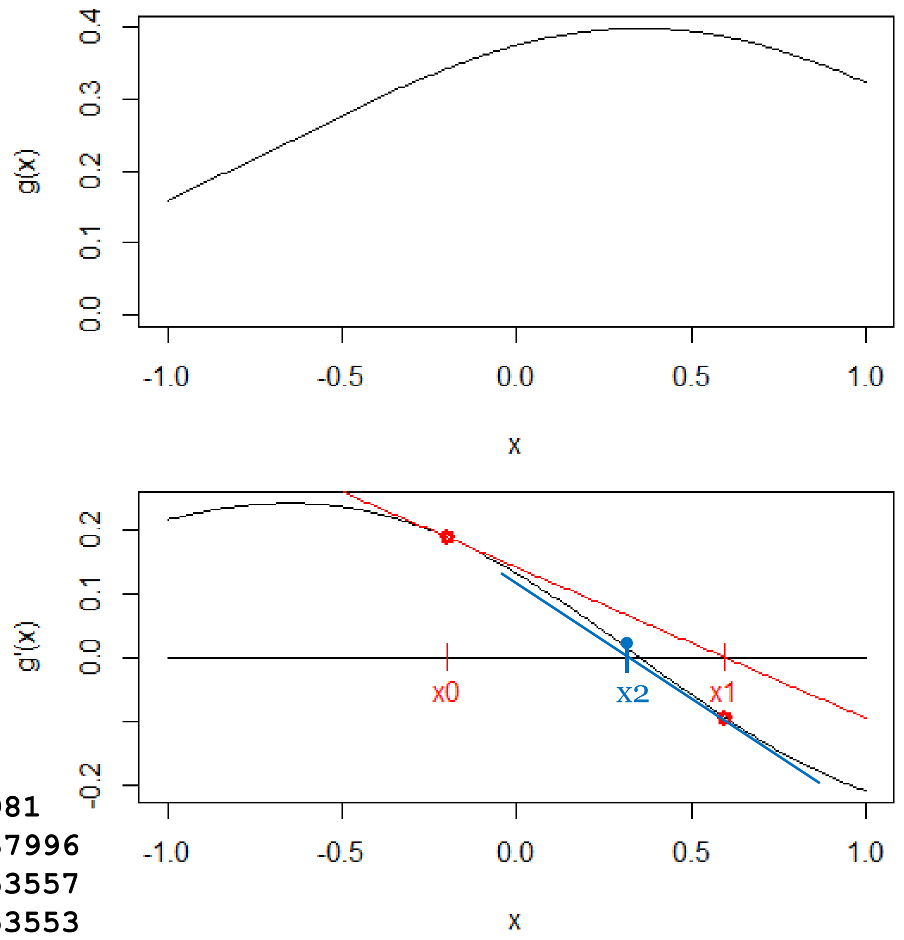
- The Newton-iteration works as:

  $$\boldsymbol{x^{(t+1)} = x^{(t)} - \left(g''\left(x^{(t)}\right)\right)^{-1} g'\left(x^{(t)}\right)}$$

- $x^{(t+1)} = x^{(t)} - g'\left(x^{(t)}\right)/g''\left(x^{(t)}\right)$
- Start with a $x^{(0)}$
- Tangent in $(x^{(0)}, g'(x^{(0)}))$ determines $x^{(1)}$
- Tangent in $(x^{(1)}, g'(x^{(1)}))$ determines $x^{(2)}$
- …
- until convergence criterion met

+Newton method is fast

- Requires existence and computation of $\boldsymbol{g''}$




```
x0   -0.2
x1    0.5981
x2    0.337996
x3    0.353557
x4    0.353553
x5    0.353553
STOP
```

Multivariate Newton

- $\boldsymbol{x}^{(\boldsymbol{t}+\mathbf{1})} = \boldsymbol{x}^{(\boldsymbol{t})} - \left(\boldsymbol{g}''\left(\boldsymbol{x}^{(\boldsymbol{t})}\right)\right)^{-1} \boldsymbol{g}'\left(\boldsymbol{x}^{(\boldsymbol{t})}\right)$
- Example:

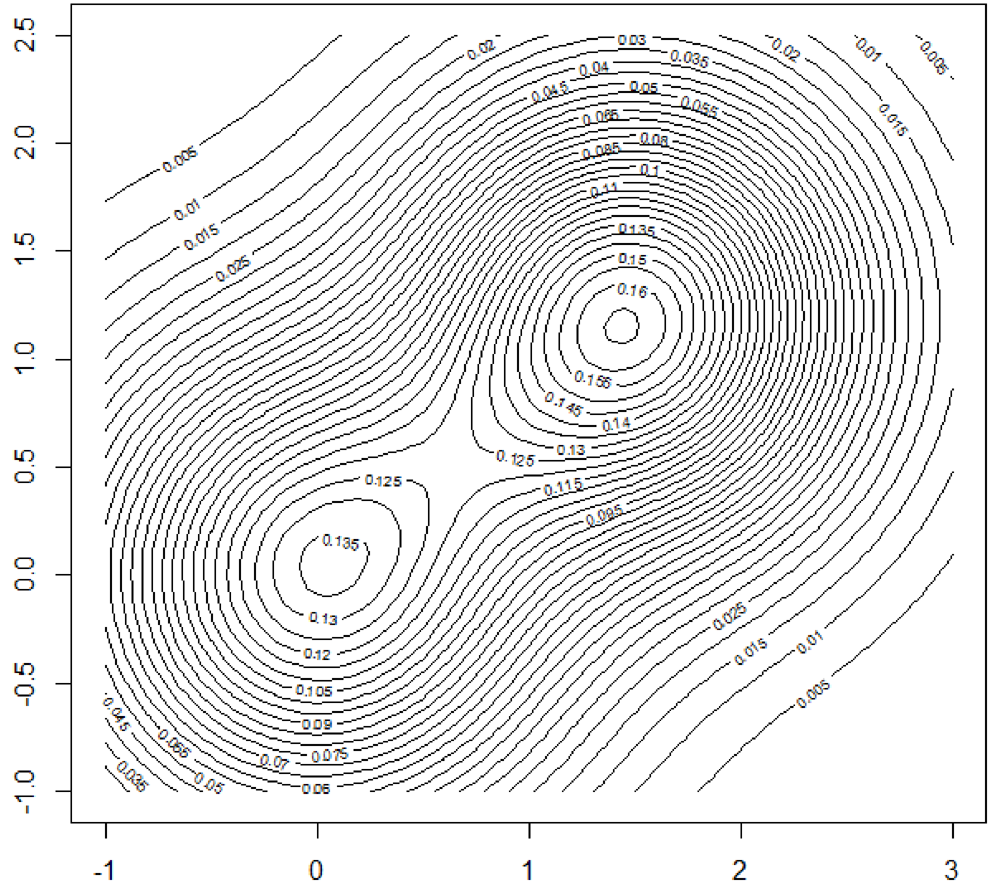
  Let g_1 and g_2 be densities of $N\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}0.6 & 0\\0 & 0.6\end{pmatrix}\right)$ and $N\left(\begin{pmatrix}1.5\\1.2\end{pmatrix}, \begin{pmatrix}0.5 & 0\\0 & 0.5\end{pmatrix}\right)$, respectively, and $g = \frac{g_1+g_2}{2}$, i.e.

  $$g(x_1, x_2) = \frac{1}{4\pi}\left(\frac{1}{0.6} e^{-\left({x_1}^2+{x_2}^2\right)/1.2} + \frac{1}{0.5} e^{-\left((x_1-1.5)^2+(x_2-1.2)^2\right)}\right)$$

  (g is density of a normal mixture distribution).
- Compute point $\boldsymbol{x} = (x_1, x_2)$ where density $g(\boldsymbol{x})$ maximal.
- Do you have a guess?

Multivariate Newton

- $g(x_1, x_2) = \frac{1}{4\pi}\left(\frac{1}{0.6} e^{-\left({x_1}^2 + {x_2}^2\right)/1.2} + \frac{1}{0.5} e^{-\left((x_1 - 1.5)^2 + (x_2 - 1.2)^2\right)}\right)$



Multivariate Newton

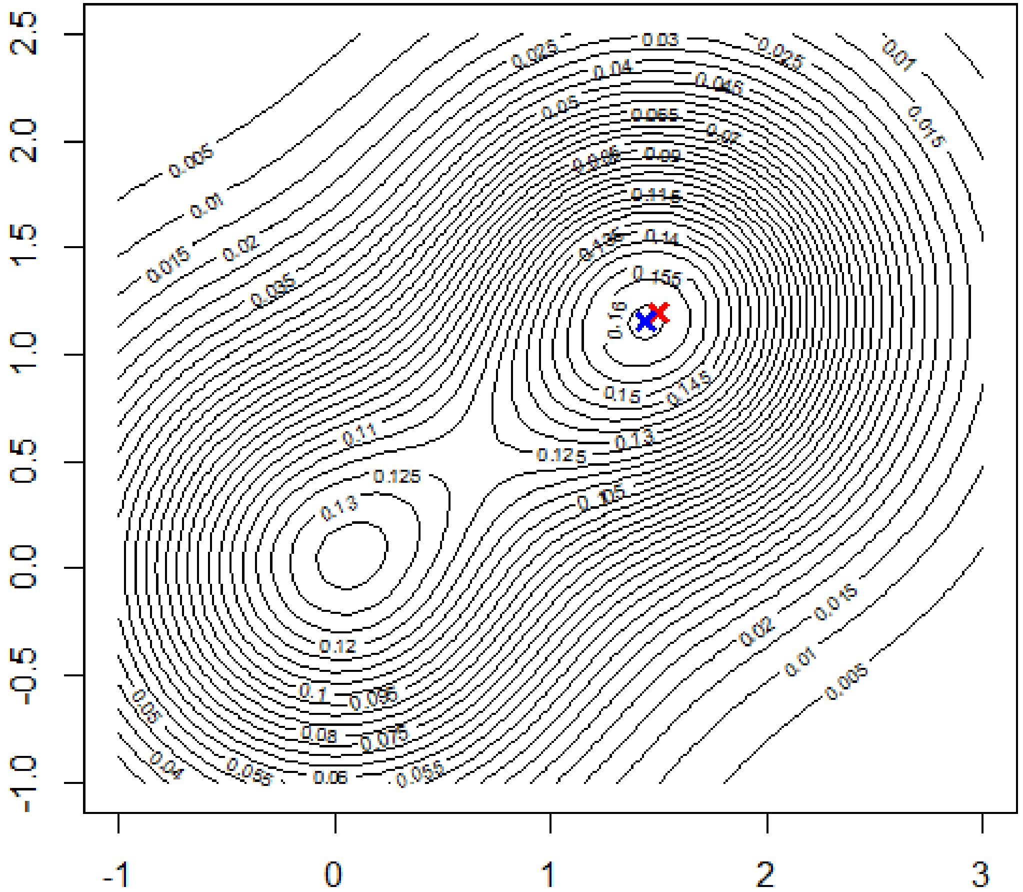
- $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \left(\boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)\right)^{-1} \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$
- We need $\boldsymbol{g}'$ and $\boldsymbol{g}''$ of

$$g(x_1, x_2) = \frac{1}{4\pi}\left(\frac{1}{0.6} e^{-\left({x_1}^2+{x_2}^2\right)/(2\cdot 0.6)} + \frac{1}{0.5} e^{-\left((x_1-1.5)^2+(x_2-1.2)^2\right)}\right)$$

- $\frac{\partial g}{\partial x_1}(x_1, x_2) = \frac{1}{4\pi}\left(\frac{-2x_1}{1.2\cdot 0.6} e^{-\left({x_1}^2+{x_2}^2\right)/1.2} + \frac{-2(x_1-1.5)}{0.5} e^{-\left((x_1-1.5)^2+(x_2-1.2)^2\right)}\right)$
- $\frac{\partial g}{\partial x_2}(x_1, x_2) = \frac{1}{4\pi}\left(\frac{-2x_2}{1.2\cdot 0.6} e^{-\left({x_1}^2+{x_2}^2\right)/1.2} + \frac{-2(x_2-1.2)}{0.5} e^{-\left((x_1-1.5)^2+(x_2-1.2)^2\right)}\right)$
- $\boldsymbol{g}'(x_1, x_2) = \begin{pmatrix} \frac{\partial g}{\partial x_1}(x_1, x_2) \\ \frac{\partial g}{\partial x_2}(x_1, x_2) \end{pmatrix}$
- $\frac{\partial^2 g}{\partial^2 x_1}(x_1, x_2)=\ldots;\ \frac{\partial^2 g}{\partial x_1 \partial x_2}(x_1, x_2)=\ldots;\ \frac{\partial^2 g}{\partial^2 x_2}(x_1, x_2)=\ldots$ lead to $\boldsymbol{g}''$

Multivariate Newton

- $x^{(t+1)} = x^{(t)} - \left(g''\left(x^{(t)}\right)\right)^{-1} g'\left(x^{(t)}\right)$

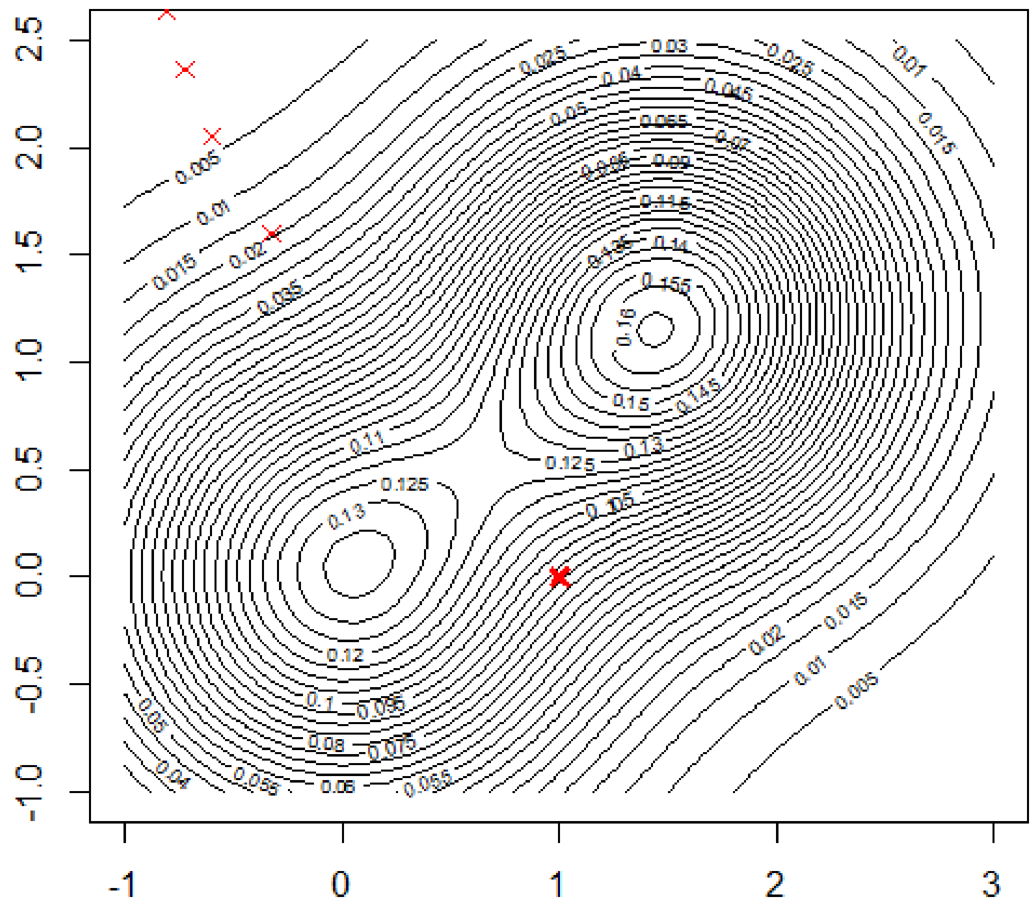


- Start with $x^{(0)} = \begin{pmatrix} 1.5 \\ 1.2 \end{pmatrix}$
- $g'\left(x^{(0)}\right) = \begin{pmatrix} -0.0153 \\ -0.0123 \end{pmatrix}$
- $g''\left(x^{(0)}\right) = \begin{pmatrix} -0.2902 & 0.0306 \\ 0.0306 & -0.3040 \end{pmatrix}$
- $\left(g''\left(x^{(0)}\right)\right)^{-1} g'\left(x^{(0)}\right) = \begin{pmatrix} \mathbf{0.058} \\ \mathbf{0.046} \end{pmatrix}$
- $x^{(1)} = \begin{pmatrix} 1.5 \\ 1.2 \end{pmatrix} - \begin{pmatrix} 0.058 \\ 0.046 \end{pmatrix} = \begin{pmatrix} \mathbf{1.442} \\ \mathbf{1.154} \end{pmatrix}$

  $x^{(2)} = x^{*} = \begin{pmatrix} \mathbf{1.441} \\ \mathbf{1.153} \end{pmatrix}$

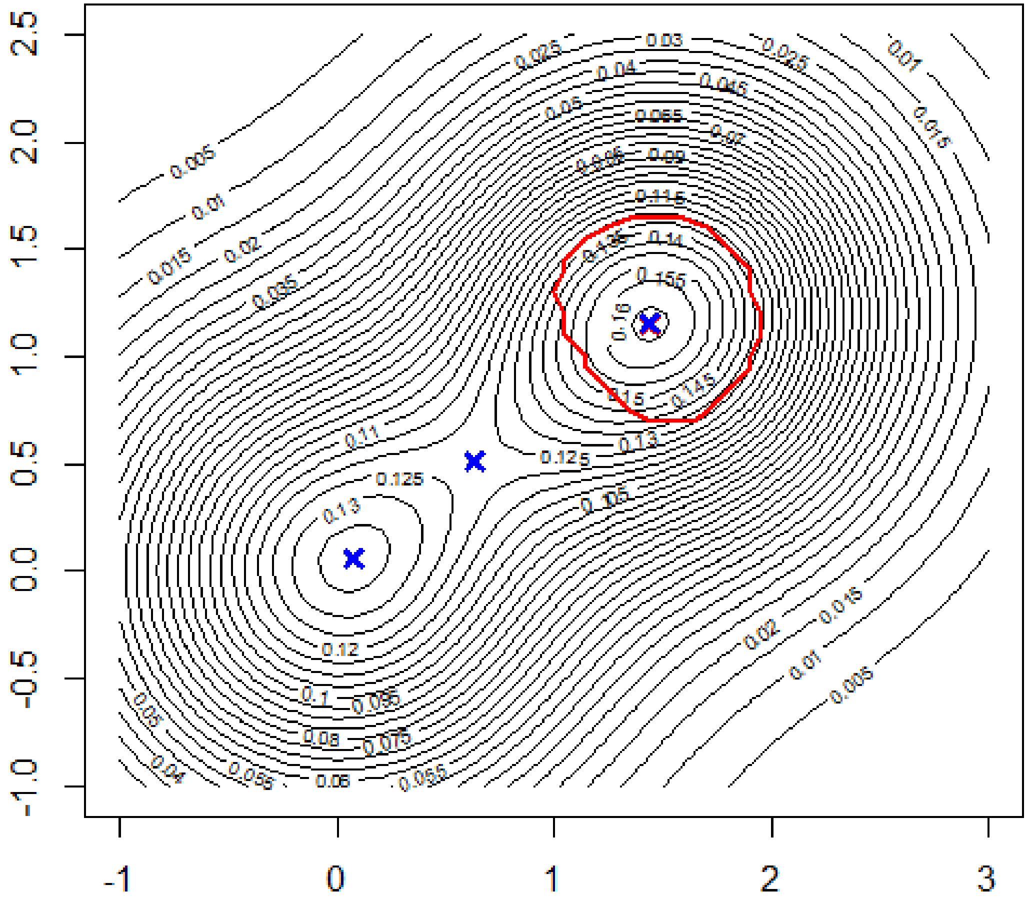
Multivariate Newton

- $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \left(\boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)\right)^{-1} \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$



- Start with $\boldsymbol{x}^{(0)} = \begin{pmatrix} \mathbf{1} \\ \mathbf{0} \end{pmatrix}$
- $\boldsymbol{g}'\left(\boldsymbol{x}^{(0)}\right) = \begin{pmatrix} -0.0667 \\ +0.0705 \end{pmatrix}$
- $\boldsymbol{g}''\left(\boldsymbol{x}^{(0)}\right) = \begin{pmatrix} 0.0347 & 0.0705 \\ 0.0705 & 0.0144 \end{pmatrix}$
- $\left(\boldsymbol{g}''\left(\boldsymbol{x}^{(0)}\right)\right)^{-1} \boldsymbol{g}'\left(\boldsymbol{x}^{(0)}\right) = \begin{pmatrix} \mathbf{1.33} \\ \mathbf{-1.60} \end{pmatrix}$
- $\boldsymbol{x}^{(1)} = \begin{pmatrix} \mathbf{1} \\ \mathbf{0} \end{pmatrix} - \begin{pmatrix} \mathbf{1.33} \\ \mathbf{-1.6} \end{pmatrix} = \begin{pmatrix} \mathbf{-0.33} \\ \mathbf{1.6} \end{pmatrix}$

Multivariate Newton



- Only starting values within the red-marked area converge to the right global maximum
- Convergence very quick
- Other starting values converge to the local maximum or saddle point (both blue-marked) or diverge while searching for a minimum

Newton: convergence

- The **Newton method converges quadratically** to the optimum x^* in a neighborhood of x^* if some assumptions are fulfilled
- E.g., in the univariate case, possible assumptions are: g is three times continuously differentiable and x^* is a simple root of g'
- In this case, the **convergence rate is** $c = \left|\frac{g'''(x^*)}{2\, g''(x^*)}\right|$
- See Givens and Hoeting (2013), section 2.1.1, for a more detailed proof

  Idea: Taylor $0 = g'(x^*) = g'\left(x^{(t)}\right) + g''\left(x^{(t)}\right)\left(x^* - x^{(t)}\right) + \frac{g'''(\tilde{x})}{2}\left(x^* - x^{(t)}\right)^2$

  $\tilde{x}$ between x^* and $x^{(t)}$
- Assumptions can be weakened
- If g is convex/concave, convergence is not only restricted to a neighborhood

Today's schedule

- Analytical optimisation
- Iterative optimisation
    - Bi-section method (univariate optimisation)
    - Convergence speed and stopping criteria
    - Newton
    - Steepest ascent
    - Accelerated steepest ascent
    - Quasi-Newton

Steepest ascent method

- The Newton method does not guarantee that $g(\boldsymbol{x})$ increases in each step
- To compute the Hessian $\boldsymbol{g}''$ can be difficult
- A method forcing improvements in each step is the steepest ascent method

$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \left(\boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)\right)^{-1} \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$$

$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha^{(t)} \boldsymbol{I}\, \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$$

- Other choices instead $\boldsymbol{I}$ in formula above possible
- We know that g will increase for small α

Backtracking line search (for steepest ascent)

$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha^{(t)} \boldsymbol{I}\, \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$$

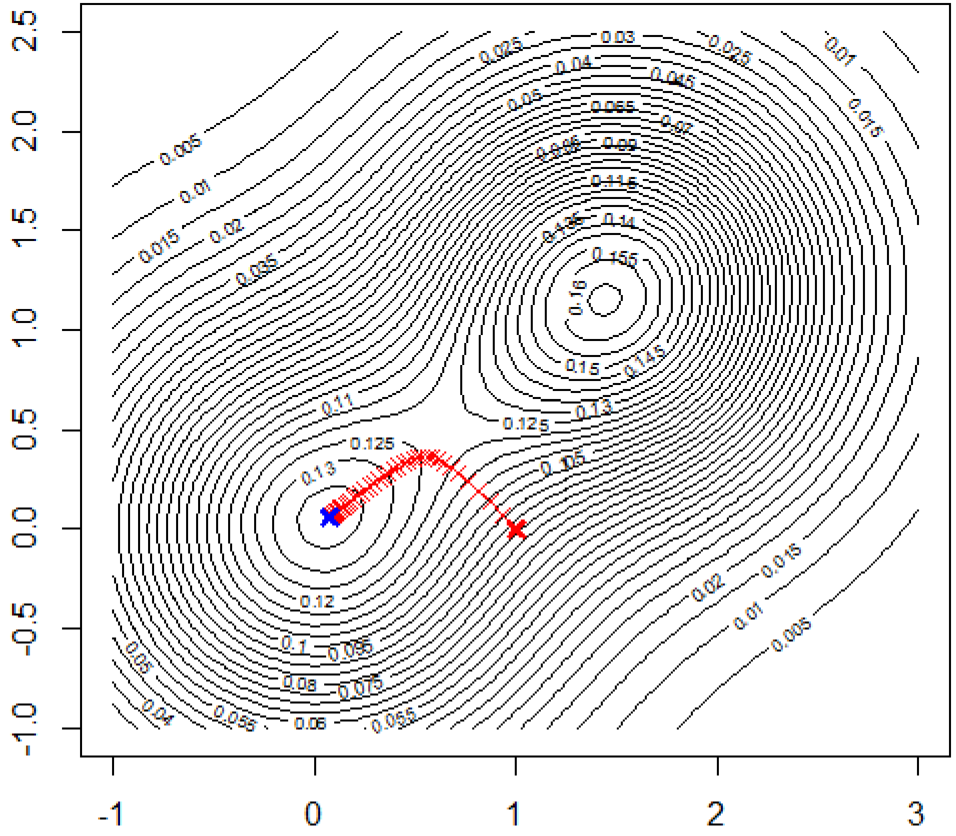
- We know that g will increase for small α

- Try $\alpha^{(t)} = 1$ first
- If g decreases, half $\alpha^{(t)}$ until $g(\boldsymbol{x}^{(t+1)})$ increases

- More sophisticated is to search α such that g becomes maximal, e.g., α can be approximately maximized with an extrapolation-bisection line search (see Section 3.5 of Wright and Recht, 2022)

Steepest ascent

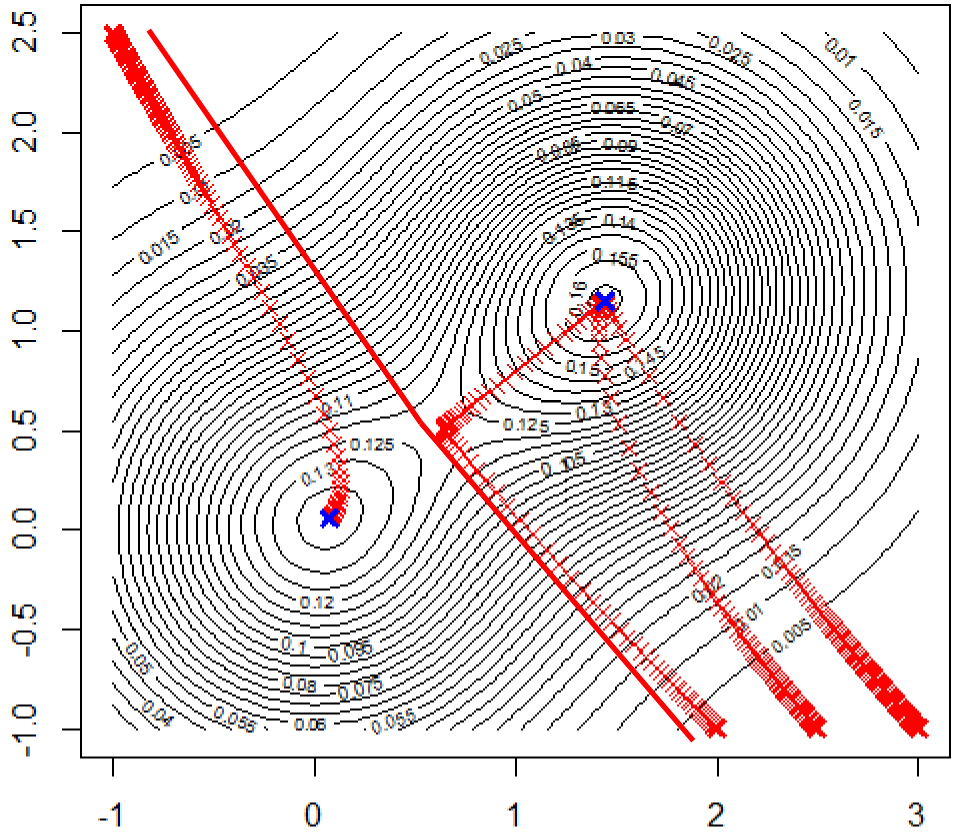
- $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha^{(t)} \boldsymbol{I}\, \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$



- Start with $\boldsymbol{x}^{(0)} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
- $\boldsymbol{g}'\left(\boldsymbol{x}^{(0)}\right) = \begin{pmatrix} -0.0667 \\ +0.0705 \end{pmatrix}$
- $\boldsymbol{x}^{(1)} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \alpha^{(0)} \begin{pmatrix} -0.0667 \\ +0.0705 \end{pmatrix} = \begin{pmatrix} \mathbf{0.9333} \\ \mathbf{0.0705} \end{pmatrix}$

Steepest ascent

- $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha^{(t)} \boldsymbol{I}\, \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$



- Start with $\boldsymbol{x}^{(0)} = \binom{-1}{2.5}, \binom{2}{-1}, \binom{2.5}{-1}, \binom{3}{-1}$
- All these paths converge either to the global or local maximum
- Convergence is much slower than for Newton
- Depending on convergence criterion and alpha-rule, convergence not always guaranteed

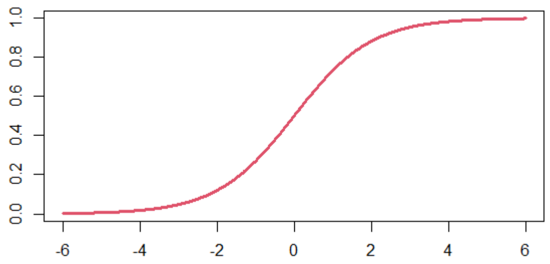
Convergence results for steepest descent

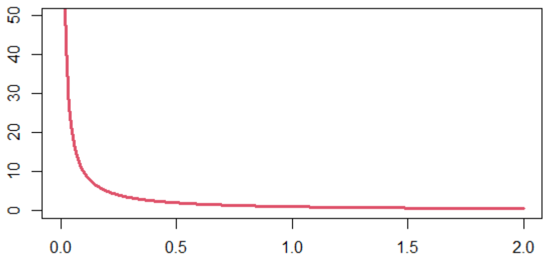
- We want to investigate convergence properties of the steepest descent/ascent
- Convergence depends also on the type of the function which is optimised
- Therefore, we introduce some mathematical concepts:
  - Lipschitz-continuous functions, Lipschitz constant L
  - L-smooth functions
  - Convex (concave) functions
  - Strongly convex functions, m-strongly convex

- Inequalities for these classes of functions help us to show convergence
- Usually, the stronger the assumptions, the stronger results can be shown

Lipschitz continuous functions

- A function $\boldsymbol{f}$ is called *Lipschitz continuous* with Lipschitz constant $L > 0$, if for all $\boldsymbol{x}, \boldsymbol{y}$,
$$\|\boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{y})\|_2 \leq L \cdot \|\boldsymbol{x} - \boldsymbol{y}\|_2.$$
- If $f: (a, b) \to \mathbb{R}$ is *differentiable*, the following is true:
f Lipschitz continuous with constant L if and only if $|f'(x)| \leq L$ for all x
- Examples:
    - $1/(1 + \exp(-x))$ is Lipschitz continuous with $L = 0.25$
    - $1/x$ is not Lipschitz continuous on $(0, \infty)$

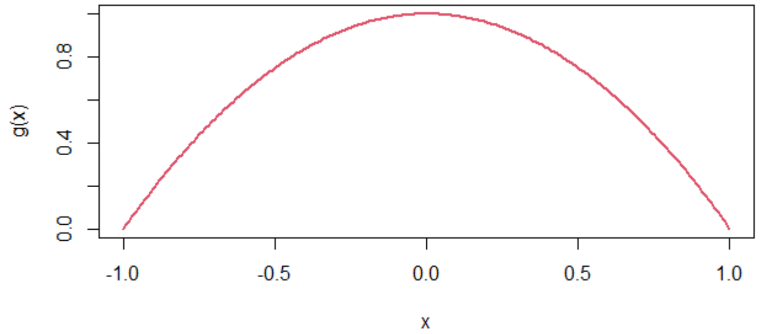




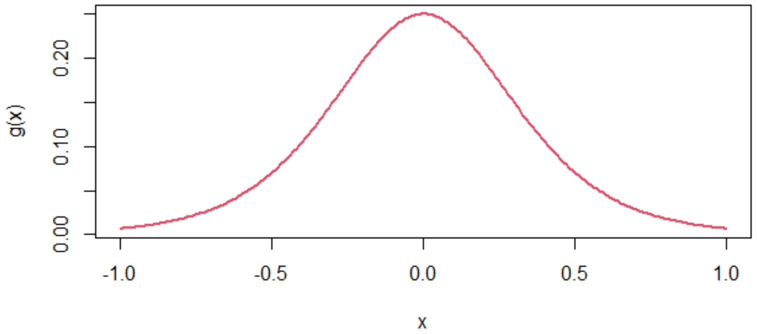
- If f has gradient $\boldsymbol{f}'$ which is Lipschitz continuous with $L > 0$, then f itself is called *L-smooth*. Further, $f(\boldsymbol{x}) \leq f(\boldsymbol{y}) + \boldsymbol{f}'(\boldsymbol{y})^T(\boldsymbol{x} - \boldsymbol{y}) + \frac{L}{2} \cdot \|\boldsymbol{x} - \boldsymbol{y}\|_2^2$.

Convexity / Concavity and global optimum

- f convex, if $f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \leq \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y})$ for all $\boldsymbol{x}, \boldsymbol{y}, \lambda \in (0,1)$
- f concave, if $f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \geq \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y})$ for all $\boldsymbol{x}, \boldsymbol{y}, \lambda \in (0,1)$



concave



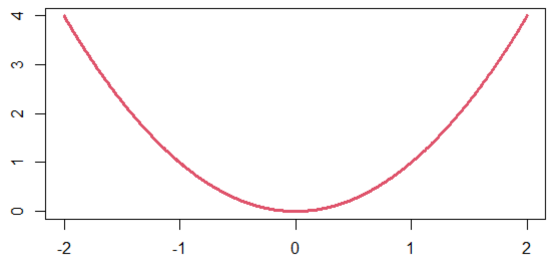
non-concave

- If f is convex (concave), a local minimum (maximum) is global

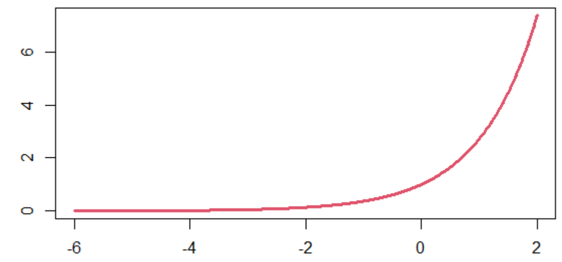
- A *differentiable* function f is convex, if for all $\boldsymbol{x}, \boldsymbol{y}$, $(\boldsymbol{f}'(\boldsymbol{x}) - \boldsymbol{f}'(\boldsymbol{y}))^T(\boldsymbol{x} - \boldsymbol{y}) \geq 0$

Strongly convex functions

- A differentiable function f is called *m-strongly convex* with $m > 0$, if for all $\boldsymbol{x}, \boldsymbol{y}$,
$$(\boldsymbol{f}'(\boldsymbol{x}) - \boldsymbol{f}'(\boldsymbol{y}))^T(\boldsymbol{x} - \boldsymbol{y}) \geq m \cdot \|\boldsymbol{x} - \boldsymbol{y}\|_2^2.$$
- For one-dimensional functions: $(f'(x) - f'(y))/(x - y) \geq m$ for all x, y.
- The function $f(x) = x^2$ is m-strongly convex with $m = 2$



- The function $f(x) = \exp(x)$ is convex but not m-strongly convex since for $x \to -\infty$, smaller and smaller m would be necessary; no $m > 0$ can be found to fulfil condition above



Strongly convex functions

- A differentiable function f is called *m-strongly convex* with $m > 0$, if for all $\boldsymbol{x}, \boldsymbol{y}$,
$$(\boldsymbol{f}'(\boldsymbol{x}) - \boldsymbol{f}'(\boldsymbol{y}))^T(\boldsymbol{x} - \boldsymbol{y}) \geq m \cdot \|\boldsymbol{x} - \boldsymbol{y}\|_2^2.$$
- An equivalent condition is
$$f(\boldsymbol{x}) - f(\boldsymbol{y}) \geq \boldsymbol{f}'(\boldsymbol{y})^T(\boldsymbol{x} - \boldsymbol{y}) + \frac{m}{2} \cdot \|\boldsymbol{x} - \boldsymbol{y}\|_2^2.$$
- A *twice differentiable* f is m-strongly convex ⇔ for all $\boldsymbol{x}$, $\boldsymbol{f}''(\boldsymbol{x}) \succcurlyeq m\boldsymbol{I}$ (⇔ $\boldsymbol{f}''(\boldsymbol{x}) - m\boldsymbol{I}$ is positive semidefinite ⇔ all eigenvalues of $\boldsymbol{f}''(\boldsymbol{x})$ are $\geq m$)

- Note:
A *twice differentiable L-smooth* f fulfils: for all $\boldsymbol{x}$, $\boldsymbol{f}''(\boldsymbol{x}) \preccurlyeq L\boldsymbol{I}$
(⇔ $L\boldsymbol{I} - \boldsymbol{f}''(\boldsymbol{x})$ is positive semidefinite ⇔ all eigenvalues of $\boldsymbol{f}''(\boldsymbol{x})$ are $\leq L$)

Optimal step length of steepest descent

- L-smooth: $f(\boldsymbol{x}) \leq f(\boldsymbol{y}) + \boldsymbol{f}'(\boldsymbol{y})^T(\boldsymbol{x}-\boldsymbol{y}) + \frac{L}{2} \cdot \|\boldsymbol{x}-\boldsymbol{y}\|_2^2$.
- Apply when $g(= f)$ is L-smooth for $\boldsymbol{y} = \boldsymbol{x}^{(t)}, \boldsymbol{x} = \boldsymbol{x}^{(t+1)}$. Then,

$$
\begin{aligned}
g\left(\boldsymbol{x}^{(t+1)}\right) &= g\left(\boldsymbol{x}^{(t)} - \alpha^{(t)} \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)\right) \\
&\leq g\left(\boldsymbol{x}^{(t)}\right) - \alpha^{(t)} \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)^T \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right) + \frac{L}{2} {\alpha^{(t)}}^2 \left\|\boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)\right\|_2^2 \\
&= \left\|\boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)\right\|_2^2 \left(\frac{g\left(\boldsymbol{x}^{(t)}\right)}{\left\|\boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)\right\|_2^2} - \alpha^{(t)} + \frac{L}{2} {\alpha^{(t)}}^2\right)
\end{aligned}
$$

- We minimize the right-hand expression using $\alpha^{(t)} = \frac{1}{L}$, and we have then
- $g\left(\boldsymbol{x}^{(t+1)}\right) \leq g\left(\boldsymbol{x}^{(t)}\right) - \frac{1}{2L} \left\|\boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)\right\|_2^2$

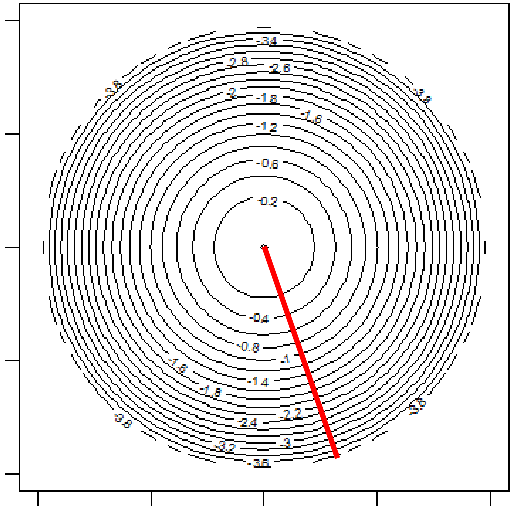
Convergence results for steepest descent

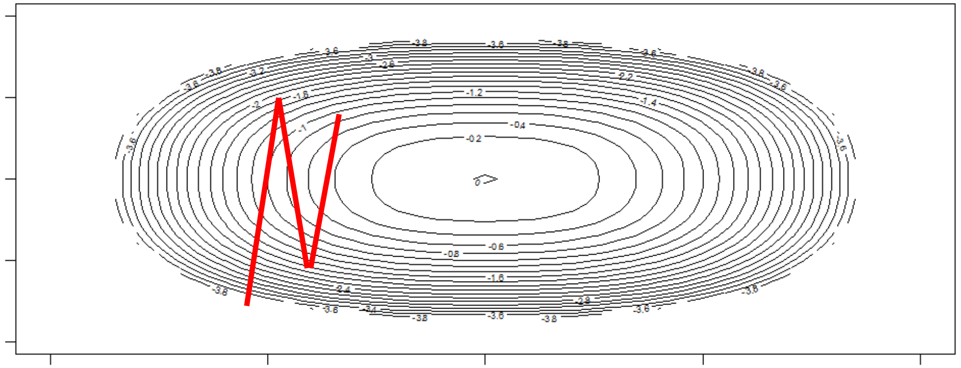
- Let g be a twice differentiable convex function which is L-smooth with global minimum at $\boldsymbol{x}^*$
- We consider the steepest descent algorithm with fixed step-size $\alpha = \frac{1}{L}$, starting vector $\boldsymbol{x}^{(0)}$ and iterations $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots$
- Then, $g\left(\boldsymbol{x}^{(t)}\right) - g(\boldsymbol{x}^*) \leq \frac{L}{2t} \left\|\boldsymbol{x}^{(0)} - \boldsymbol{x}^*\right\|_2^2$
- If g is m-strongly convex, $g\left(\boldsymbol{x}^{(t)}\right) - g(\boldsymbol{x}^*) \leq \left(1 - \frac{m}{L}\right)^t (g(\boldsymbol{x}^{(0)}) - g(\boldsymbol{x}^*))$

Steepest ascent: idea for acceleration

Globen, Stockholm – by Arild Vågen, CC BY-SA 4.0, https://commons.wikimedia.org/wiki/File:Globen_September_2014_02.jpg

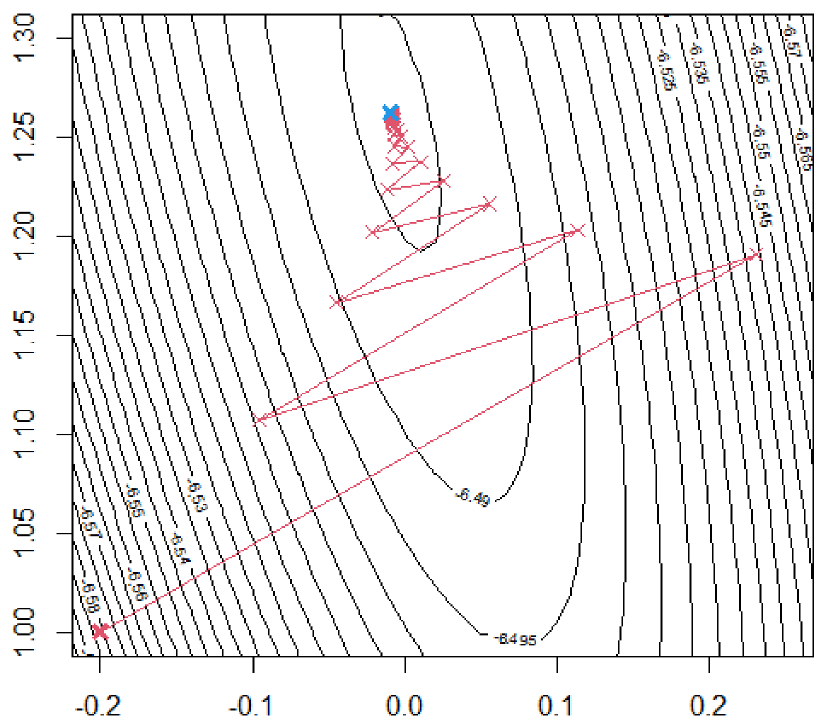
Uluru, Australia – by Stuart Edwards, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=1650537





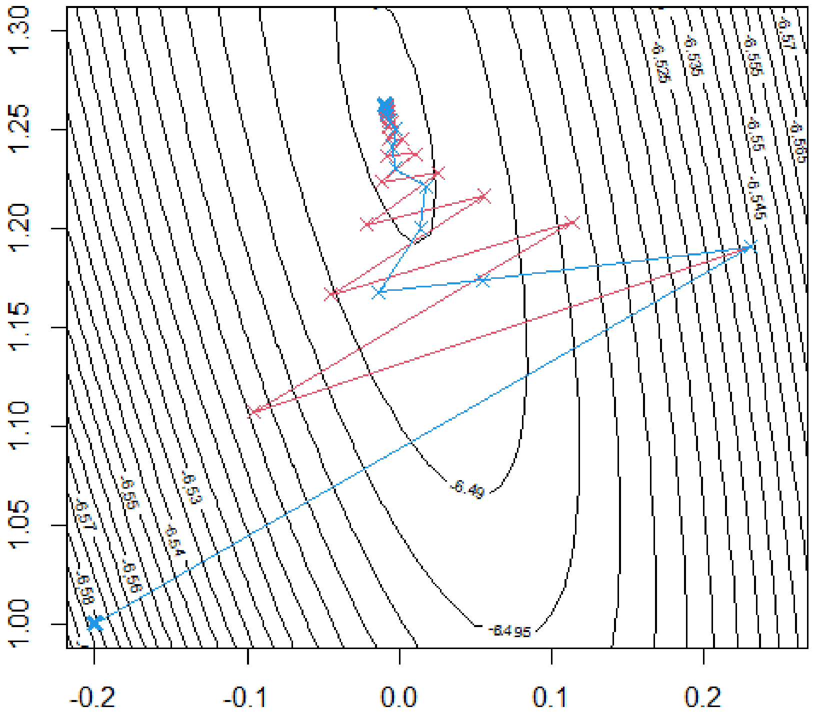
Steepest ascent: idea for acceleration

- Example: ML computation for a two-parameter model with steepest ascent, with fixed $\alpha^{(t)} = 0.667$ (no backtracking)
- Zick-zack path is common and slows down convergence
- Idea to reduce/avoid this issue: use information from last iteration about "momentum" of search path
- Called: **Accelerated steepest ascent** (or steepest ascent with momentum)



Accelerated steepest ascent: Polyak's momentum

- $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha^{(t)}\, \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right) + \beta\left(\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)}\right)$
- Polyak="gradient+momentum"
- Steepest ascent ($\alpha^{(t)} = 0.667$)
- with momentum ($\beta = 0.35$)
- Called also *heavy-ball method*
- Adding momentum reduces number of iterations from 31 to 21 in this example
- Works well in many situations
- Examples exist where Polyak's method fails to converge



Accelerated steepest ascent: Nesterov's momentum

- $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha^{(t)}\, \boldsymbol{g}'\left(\boldsymbol{x}^{(t)} + \beta(\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)})\right) + \beta(\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)})$
- Nesterov = "lookahead gradient + momentum"
- Ideally, this method has the capacity
  - to dampen oscillations and
  - to accelerate if the search path is in right direction
- Nesterov's accelerated ascent has better convergence rate as steepest ascent

Parametrisation of accelerated methods

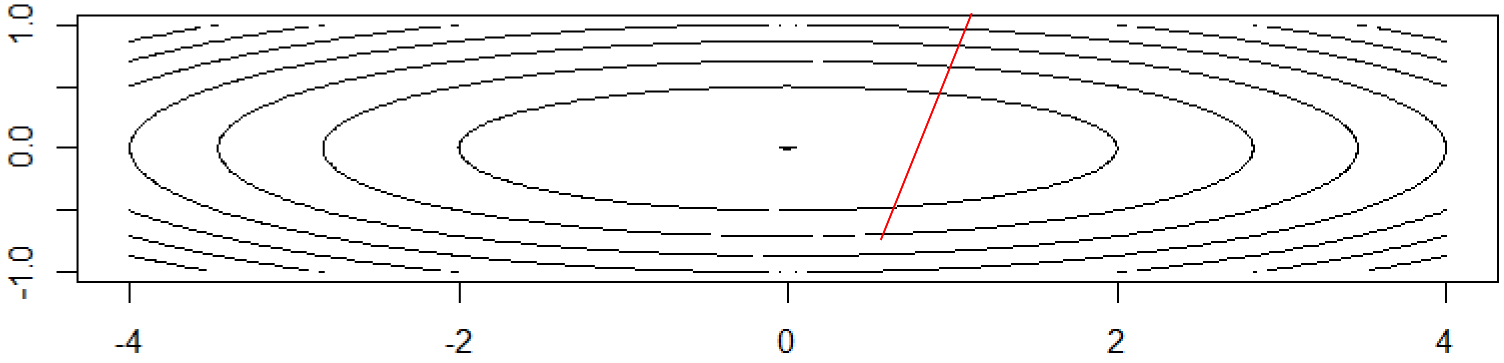
- Polyak's accelerated steepest ascent
  $$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right) + \beta\left(\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)}\right)$$
  can be written also as
  $$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha \boldsymbol{v}^{(t+1)}$$
  $$\boldsymbol{v}^{(t+1)} = \beta \boldsymbol{v}^{(t)} + \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$$
- Nesterov's accelerated steepest ascent
  $$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha \boldsymbol{g}'\left(\boldsymbol{x}^{(t)} + \beta\left(\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)}\right)\right) + \beta\left(\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)}\right)$$
  can be written also as
  $$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha \boldsymbol{v}^{(t+1)}$$
  $$\boldsymbol{v}^{(t+1)} = \beta \boldsymbol{v}^{(t)} + \boldsymbol{g}'\left(\boldsymbol{x}^{(t)} + \alpha\beta \boldsymbol{v}^{(t)}\right)$$

Steepest ascent: optimal choice of step size

- $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$
- Example:
  $g(\boldsymbol{x}) = -\frac{1}{2}\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}^T\boldsymbol{x}$, $\boldsymbol{A}$ symmetric $p \times p$ and of full rank
- $\boldsymbol{g}'(\boldsymbol{x}) = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}$
- To keep things simple (and to avoid a change of basis and some more linear algebra…), we use $\boldsymbol{b} = \boldsymbol{0}$, $\boldsymbol{A}$ =diagonal (i.e. eigenvalues in diagonal), $p = 2$
- $\boldsymbol{A} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}, \boldsymbol{g}'(\boldsymbol{x}) = \begin{pmatrix} -\lambda_1 x_1 \\ -\lambda_2 x_2 \end{pmatrix}, \lambda_1, \lambda_2 > 0$
- Then, steepest ascent is:
- $x_i^{(t+1)} = (1 - \alpha\lambda_i)x_i^{(t)} = (1 - \alpha\lambda_i)^{t+1}x_i^{(0)}$

Steepest ascent: optimal choice of step size

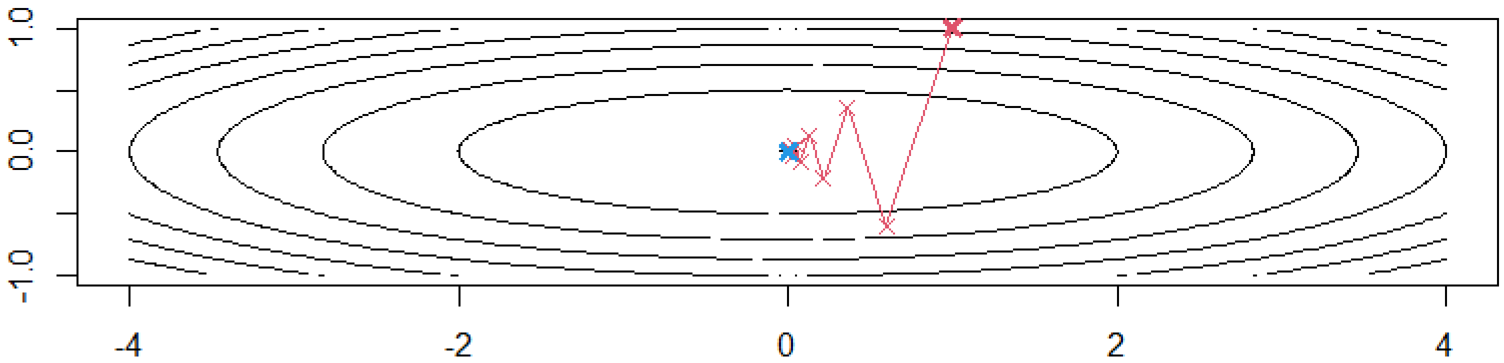
- $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \alpha \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$
- Example: $g(\boldsymbol{x}) = -\frac{1}{2}\boldsymbol{x}^T \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \boldsymbol{x}$, $\boldsymbol{g}'(\boldsymbol{x}) = \begin{pmatrix} -\lambda_1 x_1 \\ -\lambda_2 x_2 \end{pmatrix}$, $\boldsymbol{x}^{(\mathbf{0})} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$
- Steepest ascent: $x_1^{(t+1)} = (1 - \alpha\lambda_1)^{t+1}, x_2^{(t+1)} = (1 - \alpha\lambda_2)^{t+1}$
- For $\lambda_1 = \frac{1}{2}, \lambda_2 = 2$:



- Fastest convergence attained if α such that $\rho = \max\{|1 - \alpha\lambda_1|, |1 - \alpha\lambda_2|\}$ is as small as possible

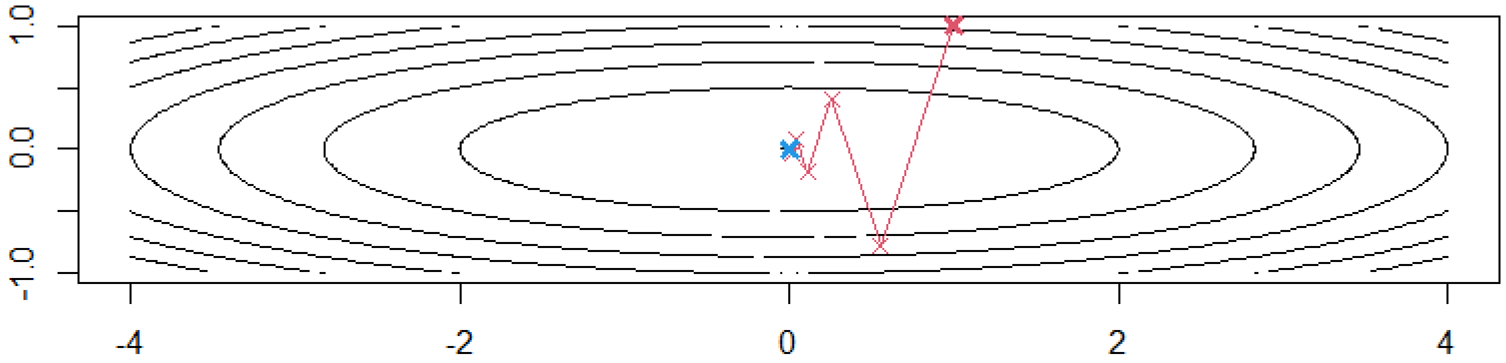
Steepest ascent: optimal choice of step size

- Steepest ascent: $x_1^{(t+1)} = (1 - \alpha\lambda_1)^{t+1}, x_2^{(t+1)} = (1 - \alpha\lambda_2)^{t+1}$
- Fastest convergence attained if α such that $\rho = \max\{|1 - \alpha\lambda_1|, |1 - \alpha\lambda_2|\}$ is as small as possible
- Fulfilled for $\alpha = \frac{2}{\lambda_1+\lambda_2}$ and then $\rho = \frac{\kappa-1}{\kappa+1}$ with $\kappa = \lambda_2/\lambda_1$
- ρ is convergence rate; κ is condition number
- For example, with $\lambda_1 = \frac{1}{2}, \lambda_2 = 2$: $\rho = \frac{3}{5}, \alpha = \frac{4}{5}$.



Accelerated steepest ascent: choice of hyperparameters

- Steepest ascent: convergence rate $\rho = \frac{\kappa-1}{\kappa+1}$ with $\kappa = \frac{\lambda_{max}}{\lambda_{min}}$
- Accelerated steepest ascent:
  - Best convergence rate: $\rho = \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)$
  - Optimal step size: $\alpha = \frac{(1+\rho)^2}{\lambda_{max}} = \frac{(1-\rho)^2}{\lambda_{min}}$
  - Optimal momentum: $\beta = \rho^2$
- For example, with $\lambda_1 = \frac{1}{2}, \lambda_2 = 2$: $\rho = \frac{1}{3}, \alpha = \frac{8}{9}, \beta = \frac{1}{9}$.



(Accelerated) steepest ascent: convergence

- Convergence rate for $\kappa = \frac{\lambda_{max}}{\lambda_{min}}$:
  - Steepest ascent: $\rho = \frac{\kappa-1}{\kappa+1}$
  - Accelerated steepest ascent: $\rho = \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)$

- $\lim_{t\to\infty} \|\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^*\| / \|\boldsymbol{x}^{(t)} - \boldsymbol{x}^*\|^q = \rho$

convergence order; here $q = 1$

convergence rate

- Example $\kappa = 100$ ("ill-conditioned"):
  - $\frac{\kappa-1}{\kappa+1} = \frac{99}{101}$; $\left(\frac{\kappa-1}{\kappa+1}\right)^t = 1, 0.98, \dots, 0.82, \dots, 0.14, \dots$ (**t=10** at 0.82, **t=100** at 0.14)
  - $\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} = \frac{9}{11}$; $\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^t = 1, 0.82, \dots, 0.13, \dots, 1.9 \cdot 10^{-9}, \dots$

Today's schedule

- Analytical optimisation
- Iterative optimisation
  - Bi-section method (univariate optimisation)
  - Convergence speed and stopping criteria
  - Newton
  - Steepest ascent
  - Accelerated steepest ascent
  - Quasi-Newton

Quasi-Newton

- Steepest ascent and Newton method have iteration
$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \left(\boldsymbol{M}^{(t)}\right)^{-1} \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$$
with $\boldsymbol{M}^{(t)} = \boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)$ for the Newton method and with $\left(\boldsymbol{M}^{(t)}\right)^{-\mathbf{1}} = -\alpha_t \boldsymbol{I}$ for the steepest ascent method
- A disadvantage of Newton is the need to calculate the Hessian $\boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)$ in each iteration
- A disadvantage of steepest ascent is that no information about the curvature is used
- We can monitor the computed gradients $\boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$ and their change gives information about the curvature of g

Quasi-Newton

- Steepest ascent and Newton method have iteration

$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \left(\boldsymbol{M}^{(t)}\right)^{-1}\boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$$

- Newton ($\boldsymbol{M}^{(t)} = \boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)$) was motivated with the multidimensional Taylor expansion

$$\boldsymbol{g}'(\boldsymbol{x}^*) \approx \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right) + \boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)\left(\boldsymbol{x}^* - \boldsymbol{x}^{(t)}\right)$$

  or

$$\boldsymbol{g}'(\boldsymbol{x}^*) - \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right) \approx \boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)\left(\boldsymbol{x}^* - \boldsymbol{x}^{(t)}\right)$$

- We want to use approximations $\boldsymbol{M}^{(t+1)}$ to $\boldsymbol{g}''\left(\boldsymbol{x}^{(t)}\right)$ which fulfil this relation when $\boldsymbol{x}^*$ is replaced by $\boldsymbol{x}^{(t+1)}$:

$$\boldsymbol{g}'\left(\boldsymbol{x}^{(t+1)}\right) - \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right) = \boldsymbol{M}^{(t+1)}\left(\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)}\right)$$

- This condition is called secant condition
- There are multiple solutions to the secant condition

Quasi-Newton

- Steepest ascent and Newton method have iteration
$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \left(\boldsymbol{M}^{(t)}\right)^{-1} \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$$
- Secant condition:
$$\boldsymbol{g}'\left(\boldsymbol{x}^{(t+1)}\right) - \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right) = \boldsymbol{M}^{(t+1)}\left(\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)}\right)$$
- Or, with $\boldsymbol{y}^{(t)} = \boldsymbol{g}'\left(\boldsymbol{x}^{(t+1)}\right) - \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$ and $\boldsymbol{z}^{(t)} = \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)}$:
$$\boldsymbol{y}^{(t)} = \boldsymbol{M}^{(t+1)} \boldsymbol{z}^{(t)}$$
- Suggestion from Broyden, Fletcher, Goldfarb, and Shanno (BFGS; 4 publications in 1970) fulfilling secant condition:

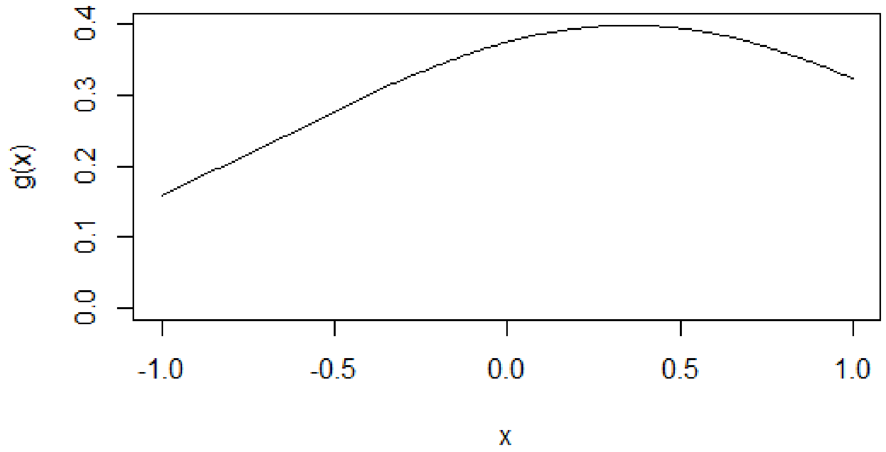
$$\boldsymbol{M}^{(t+1)} = \boldsymbol{M}^{(t)} - \frac{\boldsymbol{M}^{(t)}\boldsymbol{z}^{(t)}\left(\boldsymbol{M}^{(t)}\boldsymbol{z}^{(t)}\right)^T}{\boldsymbol{z}^{(t)^T}\boldsymbol{M}^{(t)}\boldsymbol{z}^{(t)}} + \frac{\boldsymbol{y}^{(t)}\boldsymbol{y}^{(t)^T}}{\boldsymbol{y}^{(t)^T}\boldsymbol{z}^{(t)}}$$

Quasi-Newton

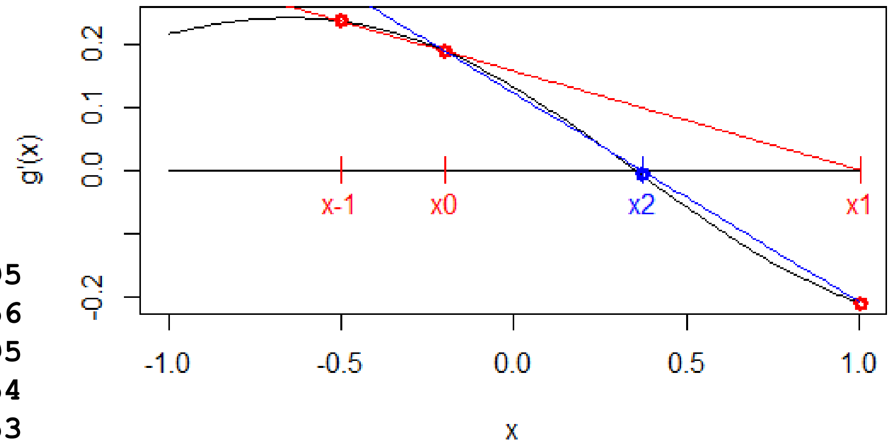
- The BFGS (quasi-Newton) method has iteration
$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \left(\boldsymbol{M}^{(t)}\right)^{-1} \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$$
and
$$\boldsymbol{M}^{(t+1)} = \boldsymbol{M}^{(t)} - \frac{\boldsymbol{M}^{(t)}\boldsymbol{z}^{(t)}\left(\boldsymbol{M}^{(t)}\boldsymbol{z}^{(t)}\right)^{T}}{\boldsymbol{z}^{(t)^{T}}\boldsymbol{M}^{(t)}\boldsymbol{z}^{(t)}} + \frac{\boldsymbol{y}^{(t)}\boldsymbol{y}^{(t)^{T}}}{\boldsymbol{y}^{(t)^{T}}\boldsymbol{z}^{(t)}}$$
- Ascent is not ensured but backtracking (stepsize-halving) can be used as for steepest ascent to ensure it:
$$\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} - \alpha^{(t)}\left(\boldsymbol{M}^{(t)}\right)^{-1} \boldsymbol{g}'\left(\boldsymbol{x}^{(t)}\right)$$
- The **R** function **`optim`** includes the quasi-Newton BFGS
- Convergence of quasi-Newton methods are faster than linear but slower than quadratic (some assumptions necessary; see e.g. Nocedal and Wright, 2006, Theorem 3.7)

Univariate secant method

- $x^{(t+1)} = x^{(t)} - g'\left(x^{(t)}\right)\frac{x^{(t)}-x^{(t-1)}}{g'\left(x^{(t)}\right)-g'\left(x^{(t-1)}\right)}$
- Start with $x^{(0)}$ and $x^{(-1)}$
- Secant through $x^{(0)}$ and $x^{(-1)}$ determines $x^{(1)}$
- Secant through $x^{(1)}$ and $x^{(0)}$ determines $x^{(2)}$
- …
- until stopping crit. fulfilled




```
x0  -0.2
x1   1.006995
x2   0.371656
x3   0.349095
x4   0.353554
x5   0.353553
x6   0.353553
STOP
```



Convergence order for deterministic algorithms

- Recall: Convergence order and convergence rate

$$\frac{\|\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^*\|}{\|\boldsymbol{x}^{(t)} - \boldsymbol{x}^*\|^q} \to c \text{ (for } t \to \infty)$$

- q is convergence order ($q = 1, 0 < c < 1$ linear; $q = 2, c > 0$ quadratic)
- c is convergence rate
- Under certain assumption, we have following orders:

Uni-dimensional	**Bisection** order = roughly 1*		**Secant** order = $(1 + \sqrt{5})/2$	**Newton** order = 2
Multi-dimensional		**Steepest ascent** order = 1	**Quasi-Newton** order > 1**	**Newton** order = 2

*strictly, the above criterion cannot be proven for bisection
**criterion above fulfilled for $q = 1$ and $c = 0$; "superlinear"

Convergence speed for an example function

- The convergence of BFGS and Newton can be extremely fast in praxis compared to steepest ascent/descent
- Example from Nocedal and Wright (2006), chapter 6: Rosenbrock function $g(\boldsymbol{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$, starting point $(-1.2, 1)$, optimum at (1,1).
  #iterations until error $< 10^{-5}$:
  - Steepest descent 5264
  - BFGS 34
  - Newton 21

Assignments

- Topic 1: March 12 until March 31*
- Topic 2: March 12 until March 31 (peer assessment until April 14)
- Topic 3: April 1 until April 14*
- Topic 4: April 15 until April 28 (peer assessment until May 14)
- Topic 5: April 29 until May 14*
- Topic 6: May 16 until June 7*
- Topic 7: May 16 until June 7 (peer assessment until June 30)

*teacher assessment

- Second chance for Topic 1-7: until **September 30 (no extension!)**