

# Optimierung

---

## Vorlesung 2 Optimierung ohne Nebenbedingungen Gradientenverfahren

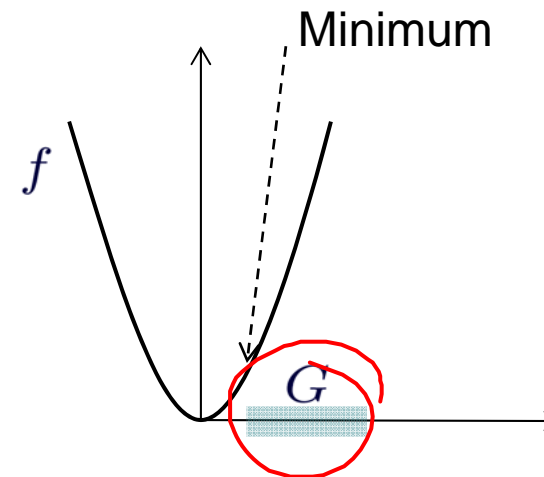
- Ein Optimierungsproblem besteht aus einer **zulässigen Menge**  $G$  und einer **Zielfunktion**  $f : G \rightarrow \mathbb{R}$

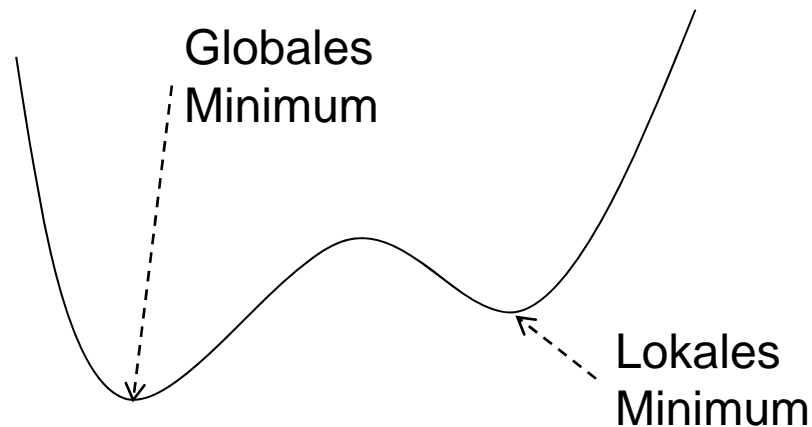
- Beispiel:

$$f(x) = x^2$$

$$G = \{x \in \mathbb{R} \mid 2 \leq x \leq 5\}$$

- **Minimum:**  $\min_x f(x) = 4$
- **Minimierer:**  $\operatorname{argmin}_x f(x) = 2$
- Die nächsten Vorlesungen:  $G = \mathbb{R}^n$ 
  - Kontinuierliche Variablen (beliebiger Dimensionalität)
  - Keine Nebenbedingungen





- Für ein **lokales Minimum**  $f(x^L)$  muss gelten

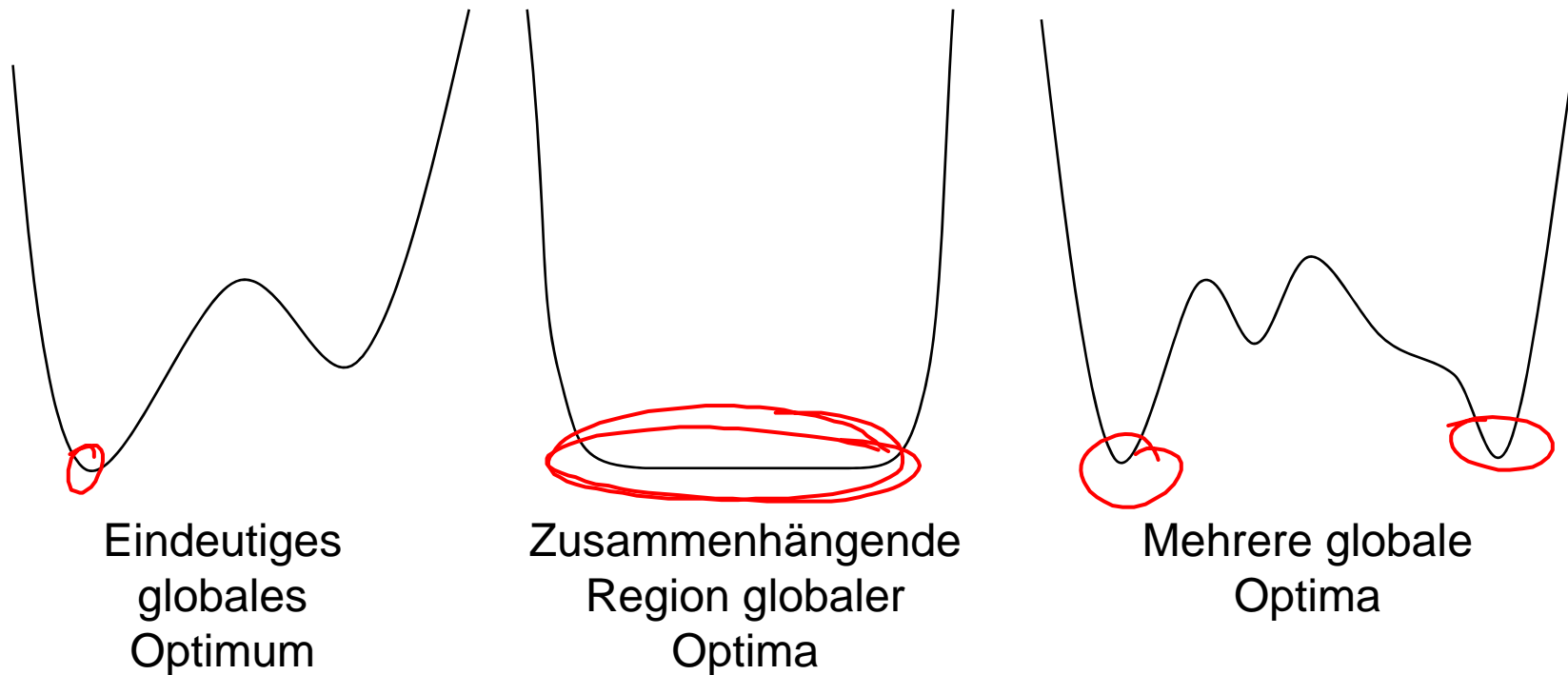
$$f(x^L) \leq f(x) \quad \forall x \in U(x^L)$$

Dabei ist  $U(x^L)$  eine Normkugel mit einem hinreichend kleinen Radius  $\epsilon > 0$ :

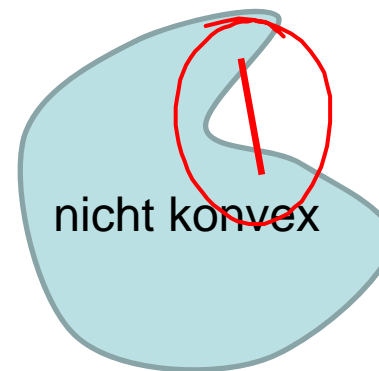
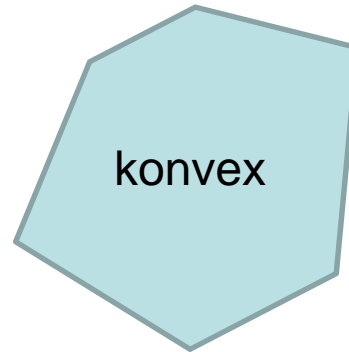
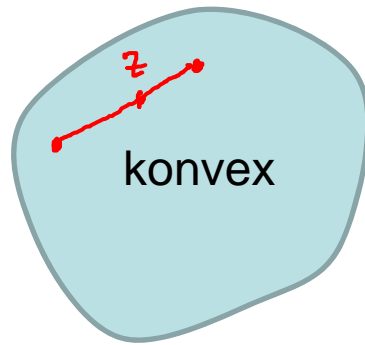
$$U(x^L) = \{x \in \mathbb{R}^n \mid \|x - x^L\| < \epsilon\}$$

- Für ein **globales Minimum**  $f(x^*)$  muss gelten

$$\underline{f(x^*) \leq f(x)} \quad \forall x \in \mathbb{R}^n$$



Kann man feststellen, ob eine Optimierungsaufgabe “gutmütig” ist, also ein eindeutiges globales Minimum und keine weiteren lokalen Minima hat?

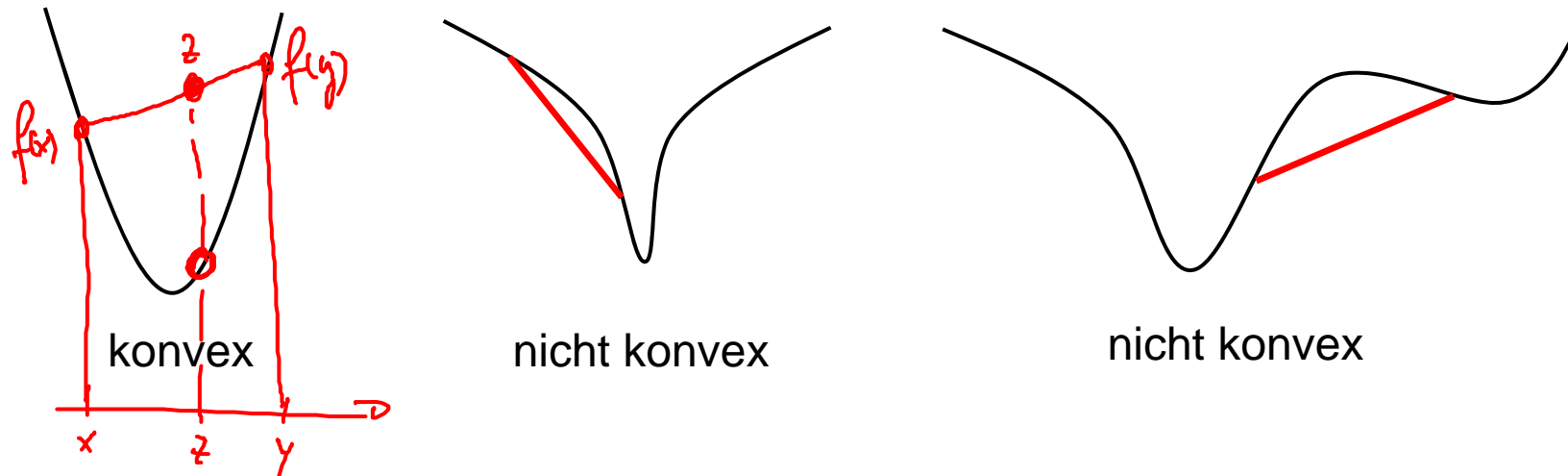


- Eine Menge  $G \subset \mathbb{R}^n$  ist konvex, wenn für beliebige Punkte  $x, y \in G$  auch die Verbindungslinie

$$[x, y] := \{z := (1 - \lambda)x + \lambda y \mid \lambda \in [0, 1]\}$$

in  $G$  enthalten ist:

$$x, y \in G \Rightarrow [x, y] \subset G$$



- Eine über einer konvexen Menge  $G$  erklärte Funktion  $f : G \rightarrow \mathbb{R}$  heißt **konvex**, falls

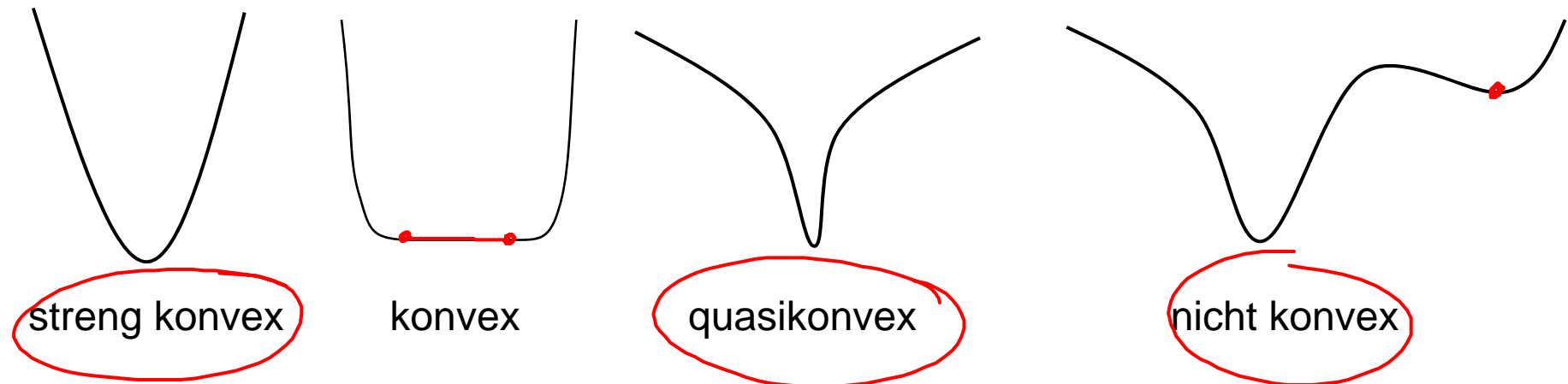
$$\underline{x}, \underline{y} \in \underline{G}; x \neq y \Rightarrow \underline{f((1-\lambda)x + \lambda y)} \leq \underline{(1-\lambda)f(x) + \lambda f(y)} \quad \forall \lambda \in [0, 1]$$

- Sie heißt **streng konvex**, falls

$$x, y \in G; x \neq y \Rightarrow f((1-\lambda)x + \lambda y) < (1-\lambda)f(x) + \lambda f(y) \quad \forall \lambda \in (0, 1)$$

- Eine Funktion  $f$  heisst **(streng) konkav**, falls  $\underline{-f}$  (streng) konvex ist

- Eine streng konvexe Funktion besitzt ein eindeutiges globales Minimum und keine lokalen Minima.
- Eine konvexe Funktion kann mehrere globale Minima besitzen, jedoch keine zusätzlichen lokalen Minima.
- Eine nicht-konvexe Funktion, die keine lokalen Minima besitzt, wird als **quasikonvex** bezeichnet.



- Gegeben: Funktion  $f: \mathbb{R}^n \rightarrow \mathbb{R}$
- Ziel: Berechne

$$\min_{x \in \mathbb{R}^n} f(x) \quad \text{und} \quad \arg \min_{x \in \mathbb{R}^n} f(x)$$

### Beispiel: Eindimensionale (differenzierbare) Funktion $f: \mathbb{R} \rightarrow \mathbb{R}$

- Notwendige Bedingung für lokales Minimum (oder Maximum):

$$\underline{f'(x) = \frac{d}{dx} f(x) = 0} \quad \text{stationärer Punkt}$$

- Hinreichende Bedingung für lokales Minimum:

$$f''(x) = \frac{d^2}{dx^2} f(x) > 0$$

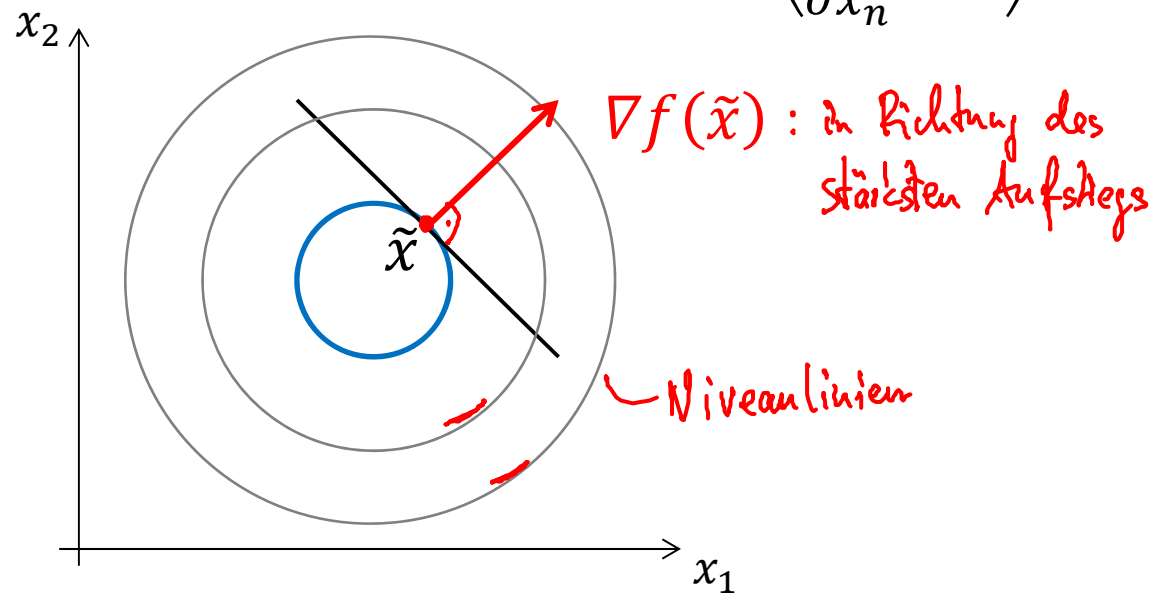
- Funktion  $f(x)$  ist konvex falls  $\underline{f''(x) \geq 0}$  für alle  $x \in \mathbb{R}$ 
  - Funktion  $f(x)$  ist ~~strik~~<sup>streng</sup> konvex falls  $\underline{f''(x) > 0}$  für alle  $x \in \mathbb{R}$
  - konvex: lokales Minimum ist globales Minimum
  - strikt konvex: eindeutiges lokales Minimum



- Gegeben: Funktion  $f: \mathbb{R}^n \rightarrow \mathbb{R}$

- Gradient am Punkt  $\tilde{x} = \begin{pmatrix} \tilde{x}_1 \\ \vdots \\ \tilde{x}_n \end{pmatrix}$ :

$$\underline{\nabla f(\tilde{x})} = \begin{pmatrix} \frac{\partial}{\partial x_1} f(\tilde{x}) \\ \vdots \\ \frac{\partial}{\partial x_n} f(\tilde{x}) \end{pmatrix}$$



- Notwendige Bedingung für lokales Minimum (Max.) an Punkt  $x$ :

$$\text{Gradient } \nabla f(x) = 0$$

- Hinreichende Bedingung für lokales Minimum:

$$\text{Hesse'sche Matrix } \mathbf{H}(x) = \nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \text{ pos. definit}$$

$\geq 0$  (pos. semidefinit)

positiv definit:  $\underline{y^T \mathbf{H} y} > 0$  für alle  $\underline{y} \in \mathbb{R}^n$

- Konvexe Funktionen
  - konvex: lokales Min. = globales Min., strikt konvex: eindeutiges lokales Min.
  - konvex :  $\mathbf{H}(x)$  pos. semidefinit für alle  $x \in \mathbb{R}^n$
  - strikt konvex:  $\mathbf{H}(x)$  positiv definit für alle  $x \in \mathbb{R}^n$

- Nachfolgend gehen wir davon aus, dass die Funktion  $f$  mindestens einmal differenzierbar ist:  $f \in \mathcal{C}^1$

- Gradient von  $f$  :

$$\underline{\nabla f = \left( \frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right)^\top}$$

- Notwendige Bedingung für ein Minimum von  $f$  :

$$\nabla f(x) = 0$$

- Ist  $f$  konvex, so ist dies auch eine hinreichende Bedingung.
  - Und auch sonst ist es oft das primäre Ziel.
- Allgemein stellt die Bedingung  $\nabla f(x) = 0$  ein **nichtlineares Gleichungssystem** dar, das numerisch gelöst werden muss.

- Iterative Abstiegsverfahren:

$$x_{k+1} = \underline{x_k} + \tau_k d_k$$

mit einem Startpunkt  $x_0$ , einer Änderungsrichtung  $d_k$  und einer Schrittweite  $\tau_k$

$x_0, x_1, x_2, \dots$



- Beim Gradientenverfahren entspricht die Änderungsrichtung dem negativen Gradienten der Zielfunktion  $f$  an der aktuellen Stelle  $x_k$

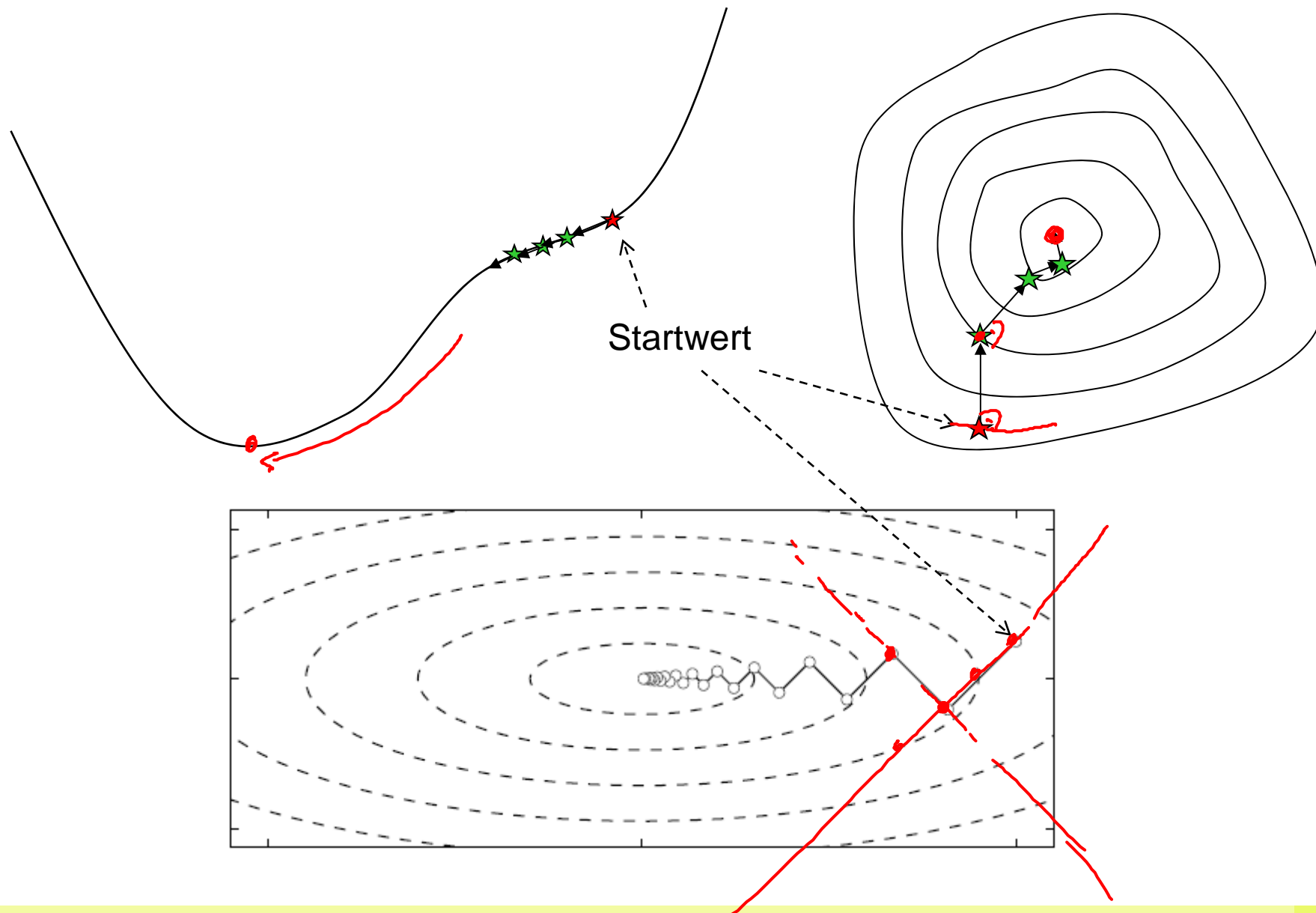
$$\underline{d_k} := \underline{-\nabla f(x_k)}$$

- Ideal: Bestimme **optimale Schrittweite**  $\tau_k$ , so dass

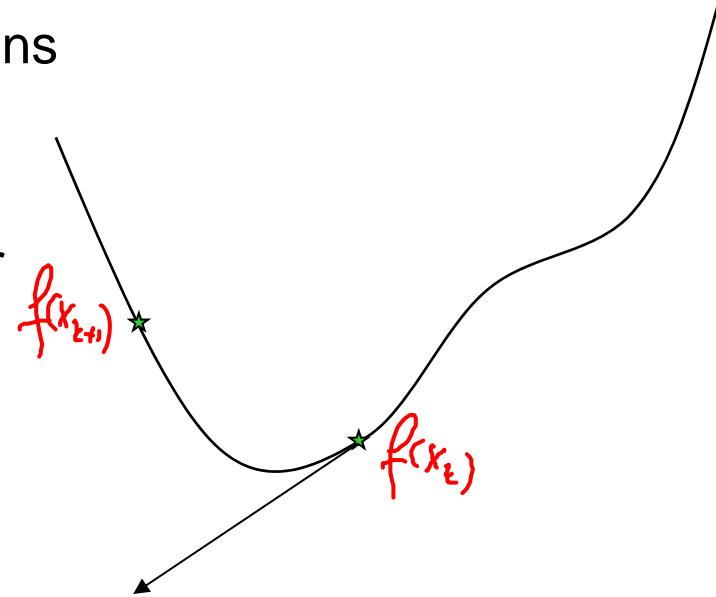
$$f(x_k + \tau_k d_k) \leq f(x_k + \alpha d_k), \quad \forall \alpha \geq 0$$

$$\underline{\tau_k} = \underline{\arg \min_{\alpha} f(x_k + \alpha d_k)}$$

- Mehr zur Schrittweitenbestimmung später...



- Die Wahl der Schrittweite ist entscheidend für Konvergenz und Effizienz des Verfahrens
- Bei zu großen Schrittweiten kann die Lösung in einem Iterationsschritt schlechter werden (keine **Konvergenz**)
- Bei kleinen Schrittweiten benötigt das Verfahren viele Iterationen zur Lösung  
→ hoher Rechenaufwand
- Richtung bereits durch  $d_k$  vorgegeben  
→ eindimensionales Optimierungsproblem über  $\tau_k$   
→ **Line search**
- In einigen Fällen ist eine feste Schrittweite  $\tau$ , die Konvergenz garantiert, effizienter als die Bestimmung einer optimalen Schrittweite.



- Optimierungsaufgabe mit einer eindimensionalen Funktion  $h : \mathbb{R}_+ \rightarrow \mathbb{R}$   
 $\tau^* = \underline{\operatorname{argmin}}_{\tau} h(\tau)$

$$h(\tau) = f(x_k + \tau \cdot d_k)$$

- Falls  $h$  differenzierbar ist, ergibt sich die notwendige Bedingung

$$\underline{h'(\tau) = 0}$$

- Normalerweise nichtlinear und dann nur mit großem Aufwand exakt lösbar
- Daher numerische Lösungsverfahren in einem geschlossenen Intervall unter Einhaltung von Qualitätsbedingungen

- Armijo-Bedingung:

$$\underline{f(x_k + \tau d_k) - f(x_k)} \leq \delta \cdot \tau \underline{|d_k^T \nabla f(x_k)|}, \quad \underline{\delta \in (0,1)}$$

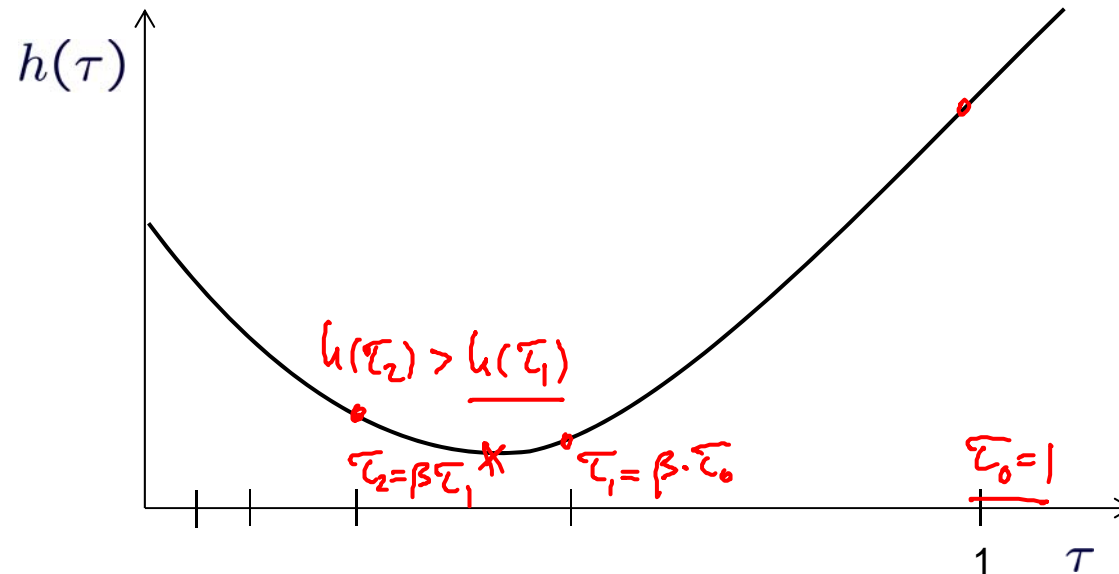
$$f(x+c) \approx f(x) + c \cdot f'(x)$$

$$f(x_k + \tau \cdot d_k) \approx f(x_k) + \tau \cdot d_k^T \nabla f(x_k)$$

$$f(x+c) \approx f(x) + c^T \nabla f(x)$$

Stellt sicher, dass die Zielfunktion  $f$  durch die Schrittweite  $\tau$  “hinreichend” reduziert wird. Überlicherweise  $\underline{\delta = 10^{-4}}$

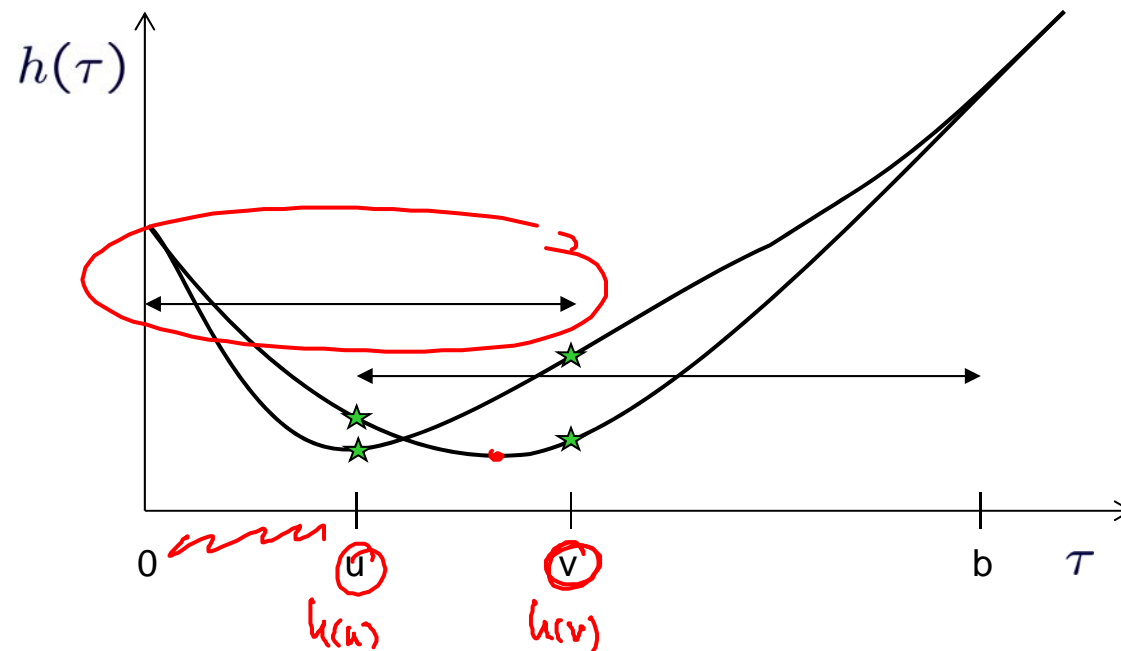
- Ziel: Approximative Minimierung von  $h(\tau) = f(x_k + \tau d_k)$ ,  $\tau \in [0,1]$



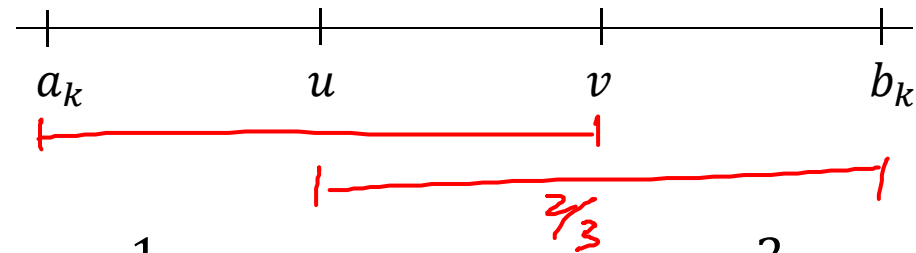
- Starte mit  $\tau_0 = 1$  und reduziere  $\tau_{k+1} = \beta \tau_k$ ,  $\beta \in (0,1)$  solange bis  $h(\tau)$  wieder steigt und die Armijo-Bedingung erfüllt ist.
- Für quasi-konvexe Funktionen  $h(\tau)$  ist die Armijo Bedingung immer erfüllt sobald  $h(\tau)$  wieder steigt.



- Ziel: Minimierung von  $h(\tau) = f(x_k + \tau d_k)$ ,  $\tau \in [0, b]$
- Annahme:  $h(\tau)$  ist streng quasi-konvex (eindeutiges Minimum)



- Grundlegende Idee: Reduziere in jedem Schritt  $k = 1, \dots, N$  das Unsicherheitsintervall  $[a_k, b_k]$   $a_1 = 0, b_1 = b$ .
- Falls  $h(u) > h(v) \Rightarrow a_{k+1} = \underline{u}, b_{k+1} = \underline{b_k}$
- Falls  $h(u) < h(v) \Rightarrow a_{k+1} = \underline{a_k}, b_{k+1} = \underline{v}$

**Einfache Variante:**

$$u = a_k + \frac{1}{3} \cdot (b_k - a_k), \quad v = a_k + \frac{2}{3} \cdot (b_k - a_k)$$

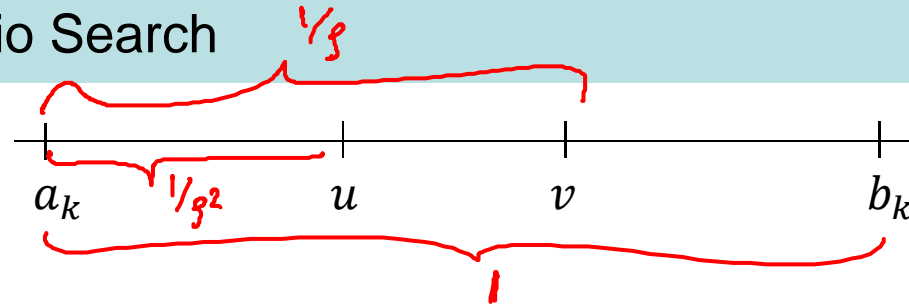
- Suchbereich wird in jedem Schritt um Faktor  $\frac{2}{3}$  verkleinert
- Pro Schritt müssen 2 neue Fkt.-werte (bei  $u$  und  $v$ ) berechnet werden

**Verbesserung:**

- Können wir  $u$  und  $v$  so wählen, dass einer der Werte als neue Grenze weiterverwendet werden kann?



$$\frac{b_k - a_k}{v - a_k} = \frac{v - a_k}{u - a_k} = \frac{b_k - u}{v - u} = \underline{\underline{\rho}}$$



Für  $\underline{\rho = \frac{1+\sqrt{5}}{2}}$  gilt  $\frac{b_k - a_k}{v - a_k} = \frac{v - a_k}{u - a_k} = \frac{b_k - u}{v - u} = \rho.$

Beweis:

$$\frac{1 - 1/\phi^2}{1/\phi - 1/\phi^2} = \frac{(1 + 1/\phi)(1 - 1/\phi)}{1/\phi(1 - 1/\phi)} =$$

- Das Gradientenabstiegsverfahren mit einer Schrittweite, welche die Armijo-Bedingung erfüllt, konvergiert zu einem lokalen Minimum.

~~$$f(x^k + \tau d^k) - f(x^k) \leq \delta \tau \nabla f(x^k)^\top d^k, \quad \delta \in (0, 1)$$~~

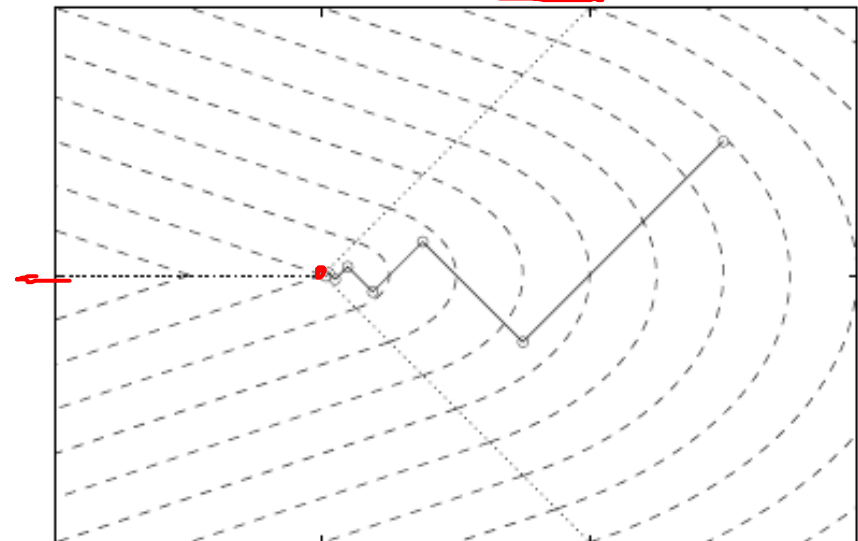
- Grund: Durch die Armijo-Bedingung

$$f(x_k + \tau d_k) - f(x_k) \leq \delta \tau d_k^\top \nabla f(x_k), \quad \delta \in (0, 1)$$

ist sichergestellt, dass der Funktionswert in jeder Iteration sinkt, solange die notwendige Bedingung nicht erfüllt ist, also  $|\nabla f(x_k)| > 0$

- Erinnerung: Wir gehen davon aus, dass  $f$  einmal stetig differenzierbar ist.

Notwendig für die Bestimmung des Gradienten und für Konvergenz zu einem lokalen Minimum.



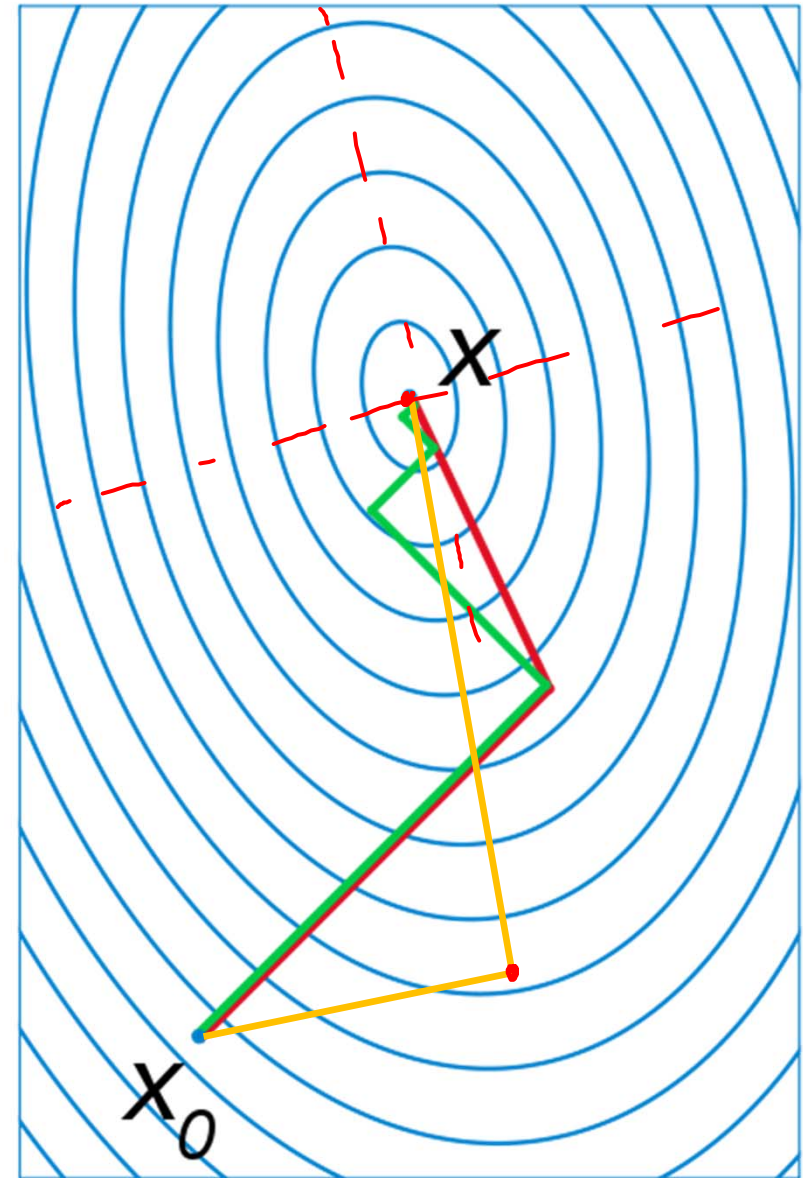
Nicht differenzierbares Beispiel  
Fixpunkt ist kein lokales Minimum

- Auf dem Weg zum Minimum werden immer wieder dieselben Richtungen gewählt  
→ ineffizient
- Mit einer Reihe von orthogonalen Richtungen, wären wir wesentlich schneller beim Minimum.
- Im Falle einer quadratischen Zielfunktion

$$f(x) = \frac{1}{2} x^T A x - b^T x$$

sogar in  $n$  Schritten

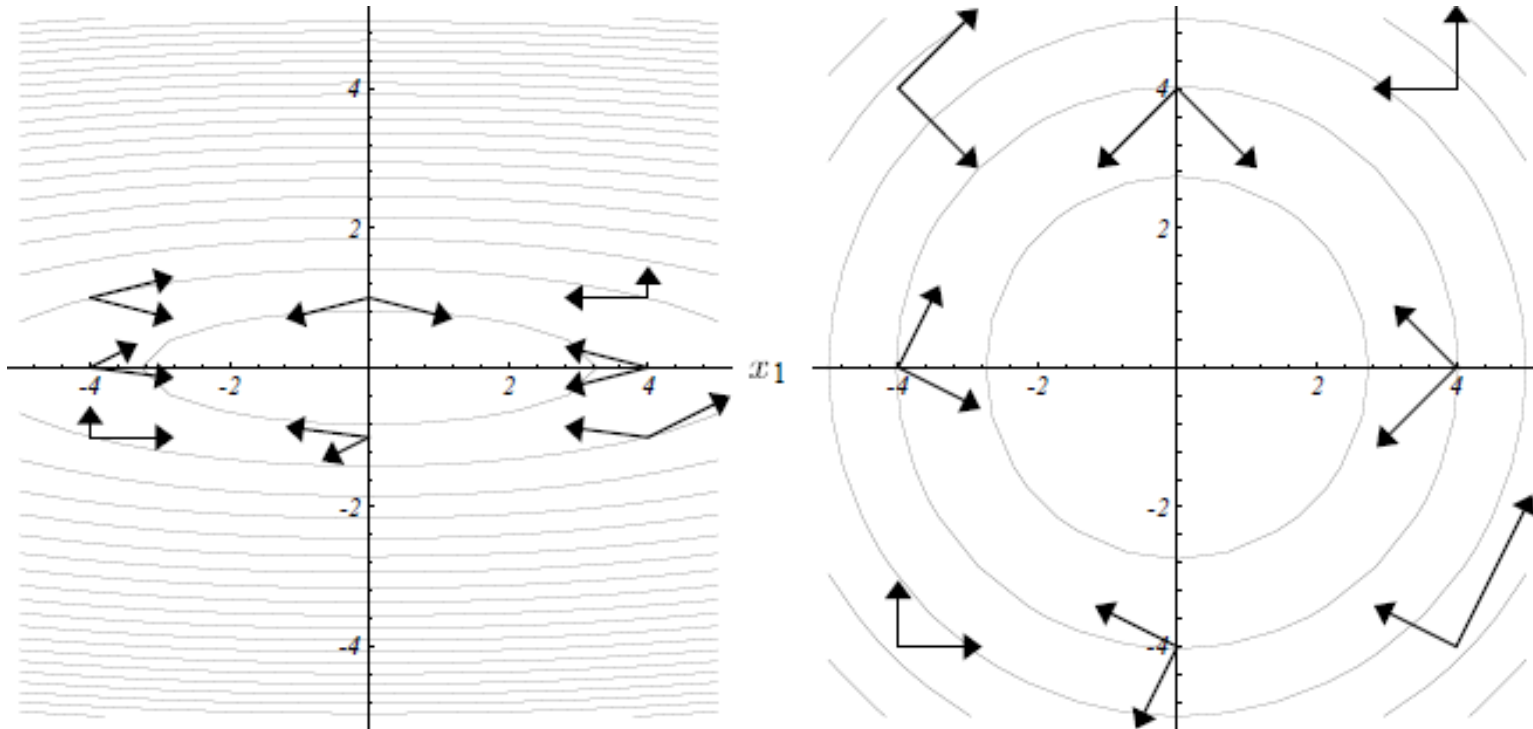
- Hierfür bräuchten wir aber die Eigenvektoren von  $A$   
→ auch ineffizient



- Statt Orthogonalität im Euklidischen Raum, Orthogonalität mit der durch  $A$  definierten Metrik:

$$\underline{x^\top Ax \neq 0}, \quad \underline{x^\top Ay = 0}$$

konjugiert



A-orthogonale Vektoren für zwei unterschiedliche quadratische Zielfunktionen

Autor: J. Shewchuk

- Wähle als erste Richtung den Gradienten an der Startposition  $x_0$ :

$$\underline{p_0} = -\nabla f(x_0) = \underline{b - Ax_0}$$

$$f(x) = \frac{1}{2} x^T A x - b x$$

- Wähle die optimale Schrittweite:

$$\nabla f(x) = Ax - b$$

$$\underline{\tau_k} = \frac{|\nabla f(x_k)|^2}{p_k^T A p_k}$$

analytische Lösung im Fall quadratischer Funktionen, da dann  $h'(\tau) = 0$  eine lineare Gleichung ist

- Neue Lösung:

$$\underline{x_{k+1}} = x_k + \tau_k p_k$$

quadr. Fkt.

- Wähle neue, konjugierte Richtung

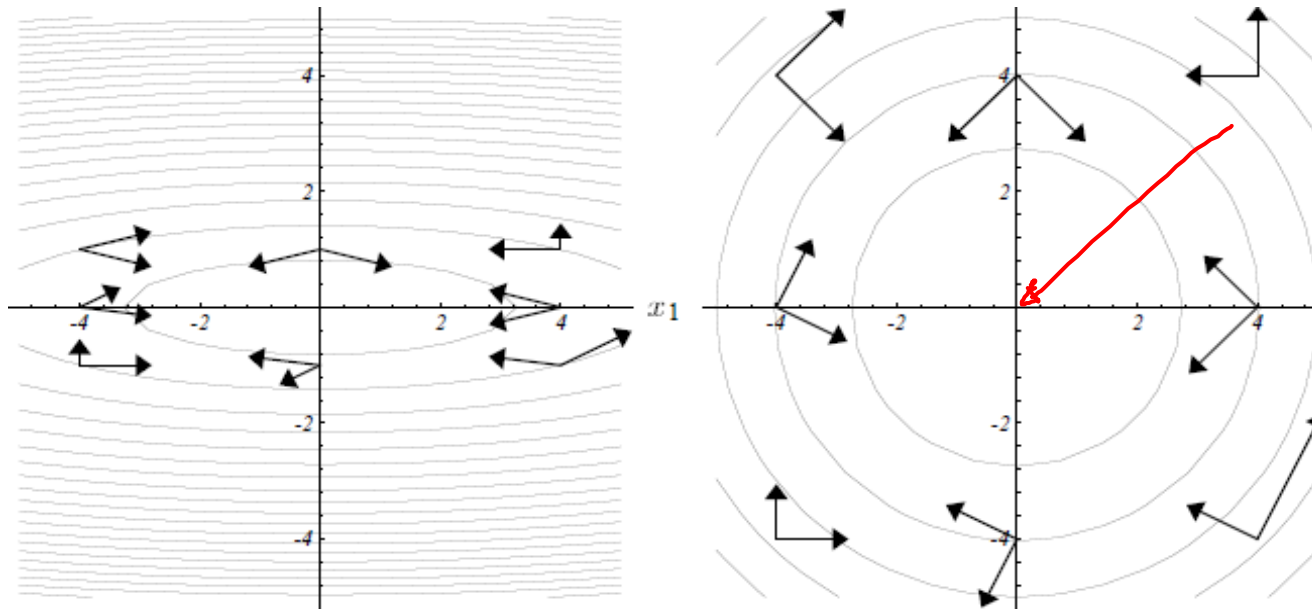
konv. in n Schritten

$$\underline{p_{k+1}} = -\nabla f(x_{k+1}) + \underline{\beta_k p_k}$$

$$\text{mit } \underline{\beta_k} = \frac{|\nabla f(x_{k+1})|^2}{|\nabla f(x_k)|^2} \text{ zur Orthogonalisierung}$$

- Der Minimierer von  $f(x) = \frac{1}{2}x^\top Ax - b^\top x$  entspricht der Lösung des linearen Gleichungssystems  $Ax = b$   $Ax - b = 0$
- Notwendige Bedingung für ein Minimum von  $f(x) = x^\top Ax - b^\top x$ :  
 $\nabla f(x) = Ax - b = 0$
- Das CG Verfahren wird daher meist zum Lösen linearer Gleichungssysteme eingesetzt.
- Die Effizienz hängt von der **Konditionszahl** von  $A \in \mathbb{R}^{n \times n}$  ab.
- Die Konditionszahl einer Matrix ist der größte Eigenwert geteilt durch den kleinsten.
- Bei exakter Zahlendarstellung garantiert das CG Verfahren in  $n$  Schritten die exakte Lösung, bei kleiner Konditionszahl bereits wesentlich schneller.





Links: höhere Konditionszahl. CG Verfahren konvergiert nach 2 Schritten.

Rechts: Konditionszahl = 1. CG Verfahren konvergiert in einem Schritt.

Author: J. Shewchuk

- Das CG Verfahren wird daher oft mit einem Präkonditionierer kombiniert, der vorab die Konditionszahl der Matrix reduziert.  
→ mehr in der Vorlesung Bildverarbeitung

- Wähle als erste Richtung den Gradienten an der Startposition  $x_0$ :

$$p_0 = -\nabla f(x_0)$$

- Wähle die optimale Schrittweite  $\tau_k$ 
  - mithilfe von Line Search (keine analytische Lösung verfügbar)

- Neue Lösung:

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

- Wähle neue, konjugierte Richtung

$$p_{k+1} = -\nabla f(x_{k+1}) + \beta_k p_k$$

$$\text{mit } \beta_k = \frac{|\nabla f(x_{k+1})|^2}{|\nabla f(x_k)|^2} \text{ zur Orthogonalisierung (Fletcher/Reeves)}$$

- Andere Orthogonalisierungsverfahren möglich (z.B. Polak/Ribière)

- Für Minimierungsaufgaben auf konvexen Mengen und konvexen Funktionen lässt sich das Minimum garantieren.
- Gradientenabstieg ist ein einfaches Verfahren zur lokalen Minimierung allgemeiner differenzierbarer Funktionen.
- Die Schrittweitenwahl führt zu einem eindimensionalen Minimierungsproblem.
- Eine effizientere Variante ist das CG Verfahren. Insbesondere quadratische Funktionen lassen sich damit effizient minimieren.