Sestupné metody

Vojtěch Franc
Centrum strojového vnímání, ČVUT FEL Praha

Optimalizace ZS 2011

Volný a vázaný extrém, stacionární a kritický bod

Lokální minimum/maximum funkce na množině X se nabývá buď

- ve vnitřním bodě X, pak se jedná o **volný extrém**
- v hraničním bodě X , pak se jedná o **vázaný extrém**

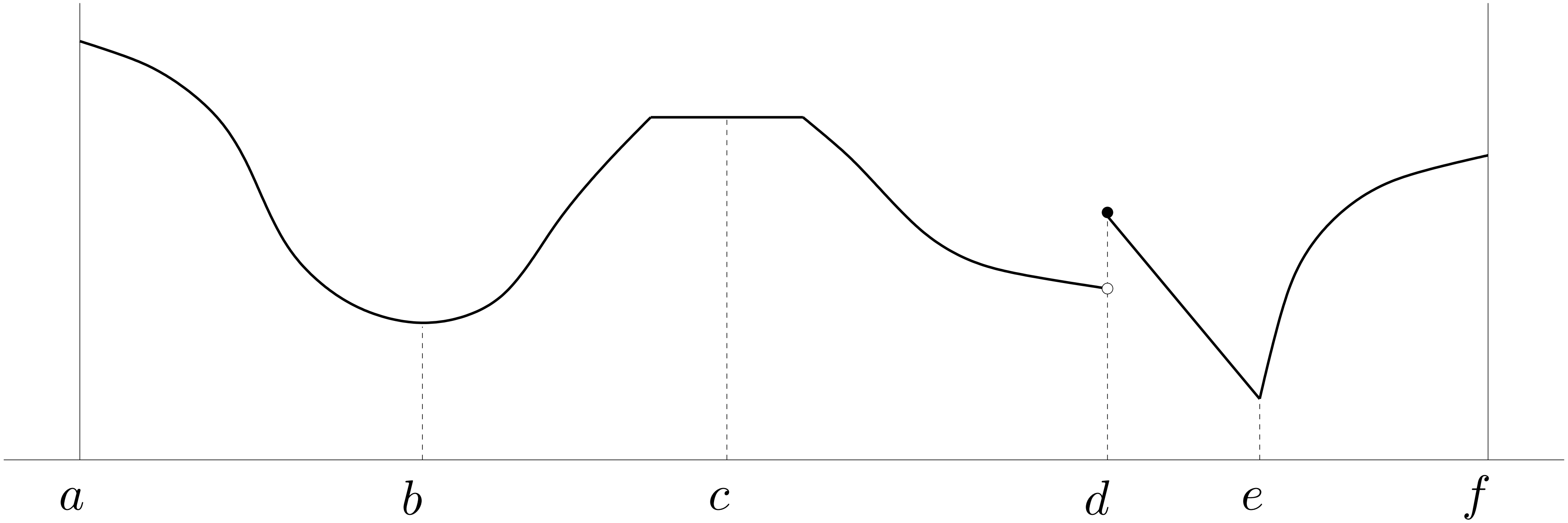
Volný a vázaný extrém, stacionární a kritický bod

Lokální minimum/maximum funkce na množině X se nabývá buď

- ve vnitřním bodě X, pak se jedná o **volný extrém**
- v hraničním bodě X , pak se jedná o **vázaný extrém**

Bod $\mathbf{x}$ z definičního oboru funkce f nazveme

- **stacionární**, pokud v něm je funkce f diferencovatelná a má všechny parciální derivace nulové, tj. $f'(\mathbf{x}) = \mathbf{0}^T$.
- **kritický**, pokud je buď stacionární nebo v něm funkce není diferencovatelná.



Podmínky na volné lokální extrémy

Věta: Nechť $\mathbf{x}$ je vnitřní bod množiny $X \subseteq \mathbb{R}^n$. Nechť $\mathbf{x}$ je lokální extrém funkce $f\colon X \to \mathbb{R}$ na množině X. Pak $\mathbf{x}$ je kritický bod funkce f.

Podmínky na volné lokální extrémy

Věta: Nechť $\mathbf{x}$ je vnitřní bod množiny $X \subseteq \mathbb{R}^n$. Nechť $\mathbf{x}$ je lokální extrém funkce $f\colon X \to \mathbb{R}$ na množině X. Pak $\mathbf{x}$ je kritický bod funkce f.

Poznámka: Věta říká, že kritické body jsou podezřelé z volného lokálního extrému. Kritický bod nemusí být nutně lokální extrém (např. $f(x) = x^3$).

Podmínky na volné lokální extrémy

Věta: Nechť $\mathbf{x}$ je vnitřní bod množiny $X \subseteq \mathbb{R}^n$. Nechť $\mathbf{x}$ je lokální extrém funkce $f\colon X \to \mathbb{R}$ na množině X. Pak $\mathbf{x}$ je kritický bod funkce f.

Poznámka: Věta říká, že kritické body jsou podezřelé z volného lokálního extrému. Kritický bod nemusí být nutně lokální extrém (např. $f(x) = x^3$).

Věta: Nechť $\mathbf{x}$ je vnitřní bod množiny $X \subseteq \mathbb{R}^n$. Nechť je funkce $f\colon X \to \mathbb{R}$ v bodě $\mathbf{x}$ dvakrát diferencovatelná a nechť $\mathbf{x}$ je stacionární bod.

- f má v bodě $\mathbf{x}$ ostré lokální minimum (maximum) na X právě tehdy, když Hessova matice druhých derivací $f''(\mathbf{x})$ je pozitivně (negativně) definitní.
- Je-li $f''(\mathbf{x})$ indefinitní, nemá f v $\mathbf{x}$ lokální minimum ani lokální maximum X ($\mathbf{x}$ se nazývá **sedlový bod**).

Podmínky na volné lokální extrémy

Věta: Nechť $\mathbf{x}$ je vnitřní bod množiny $X \subseteq \mathbb{R}^n$. Nechť $\mathbf{x}$ je lokální extrém funkce $f\colon X \to \mathbb{R}$ na množině X. Pak $\mathbf{x}$ je kritický bod funkce f.

Poznámka: Věta říká, že kritické body jsou podezřelé z volného lokálního extrému. Kritický bod nemusí být nutně lokální extrém (např. $f(x) = x^3$).

Věta: Nechť $\mathbf{x}$ je vnitřní bod množiny $X \subseteq \mathbb{R}^n$. Nechť je funkce $f\colon X \to \mathbb{R}$ v bodě $\mathbf{x}$ dvakrát diferencovatelná a nechť $\mathbf{x}$ je stacionární bod.

- f má v bodě $\mathbf{x}$ ostré lokální minimum (maximum) na X právě tehdy, když Hessova matice druhých derivací $f''(\mathbf{x})$ je pozitivně (negativně) definitní.
- Je-li $f''(\mathbf{x})$ indefinitní, nemá f v $\mathbf{x}$ lokální minimum ani lokální maximum X ($\mathbf{x}$ se nazývá **sedlový bod**).

Poznámka: Věta nic neříká o případu, kdy je $f''(\mathbf{x})$ pozitivně nebo negativně semidefinitní. V tomto případě $\mathbf{x}$ může nebo nemusí být lokální extrém (např. $f(x) = x^3$ a $f(x) = x^4$).

Sestupné metody

- Iterační algoritmy na hledání lokálního minima funkce $f\colon \mathbb{R}^n \to \mathbb{R}$ začínají z odhadu řešení $\mathbf{x}_1 \in \mathbb{R}^n$, které postupně vylepšují aplikací pravidla

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k$$

vektor $\mathbf{s}_k \in \mathbb{R}^n$ je **směr hledání** a skalár $\alpha_k > 0$ je **délka kroku**.

Sestupné metody

- Iterační algoritmy na hledání lokálního minima funkce $f\colon \mathbb{R}^n \to \mathbb{R}$ začínají z odhadu řešení $\mathbf{x}_1 \in \mathbb{R}^n$, které postupně vylepšují aplikací pravidla

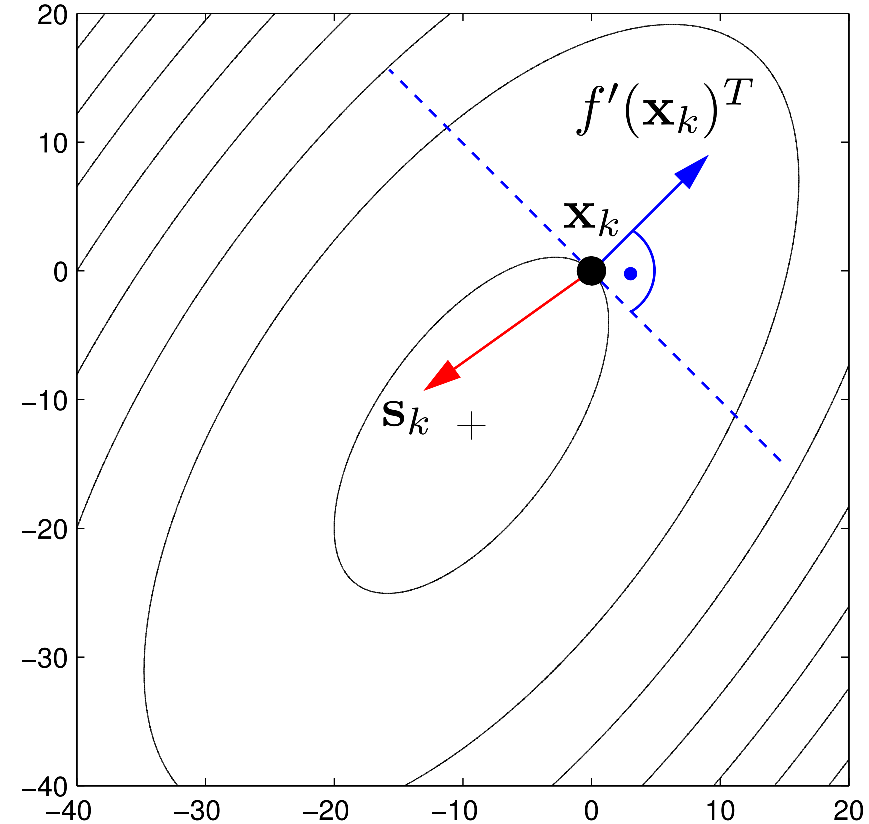
$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k$$

vektor $\mathbf{s}_k \in \mathbb{R}^n$ je **směr hledání** a skalár $\alpha_k > 0$ je **délka kroku**.

- Směr $\mathbf{s}_k$ se nazývá **sestupný**, jestliže

$$f'(\mathbf{x}_k)\mathbf{s}_k < 0$$

směrová derivace ve směru $\mathbf{s}_k$ je záporná.



- Délku kroku α_k můžeme nalézt minimalizací jednorozměrné funkce

$$\min_{\alpha \geq 0} f\left(\mathbf{x}_k + \alpha \mathbf{s}_k\right)$$

Tuto úlohu stačí často řešit přibližně.

Gredientní metoda

- Cíl: chceme nalézt lokální minimum funkce $f\colon \mathbb{R}^n \to \mathbb{R}$

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$

Gredientní metoda

- Cíl: chceme nalézt lokální minimum funkce $f\colon \mathbb{R}^n \to \mathbb{R}$

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$

- Směr hledání se zvolí jako záporný gradient funkce f v bodě $\mathbf{x}_k$:

$$\mathbf{s}_k = -f'(\mathbf{x}_k)^T$$

takže iterační pravidlo má tvar

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k \qquad \Rightarrow \qquad \mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x})^T$$

Gredientní metoda

- Cíl: chceme nalézt lokální minimum funkce $f\colon \mathbb{R}^n \to \mathbb{R}$

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$

- Směr hledání se zvolí jako záporný gradient funkce f v bodě $\mathbf{x}_k$:

$$\mathbf{s}_k = -f'(\mathbf{x}_k)^T$$

takže iterační pravidlo má tvar

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k \qquad \Rightarrow \qquad \mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x})^T$$

- Směr hledání je sestupný

$$f'(\mathbf{x})\mathbf{s}_k = -f'(\mathbf{x})f'(\mathbf{x})^T = -\|f'(\mathbf{x})\|^2 < 0$$

Gredientní metoda

- Cíl: chceme nalézt lokální minimum funkce $f\colon \mathbb{R}^n \to \mathbb{R}$

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$

- Směr hledání se zvolí jako záporný gradient funkce f v bodě $\mathbf{x}_k$:

$$\mathbf{s}_k = -f'(\mathbf{x}_k)^T$$

takže iterační pravidlo má tvar

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k \qquad \Rightarrow \qquad \mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x})^T$$

- Směr hledání je sestupný

$$f'(\mathbf{x})\mathbf{s}_k = -f'(\mathbf{x})f'(\mathbf{x})^T = -\|f'(\mathbf{x})\|^2 < 0$$

- Gradientní metoda je jednoduchá, ale často konverguje pomalu blízko optima.

Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$
kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

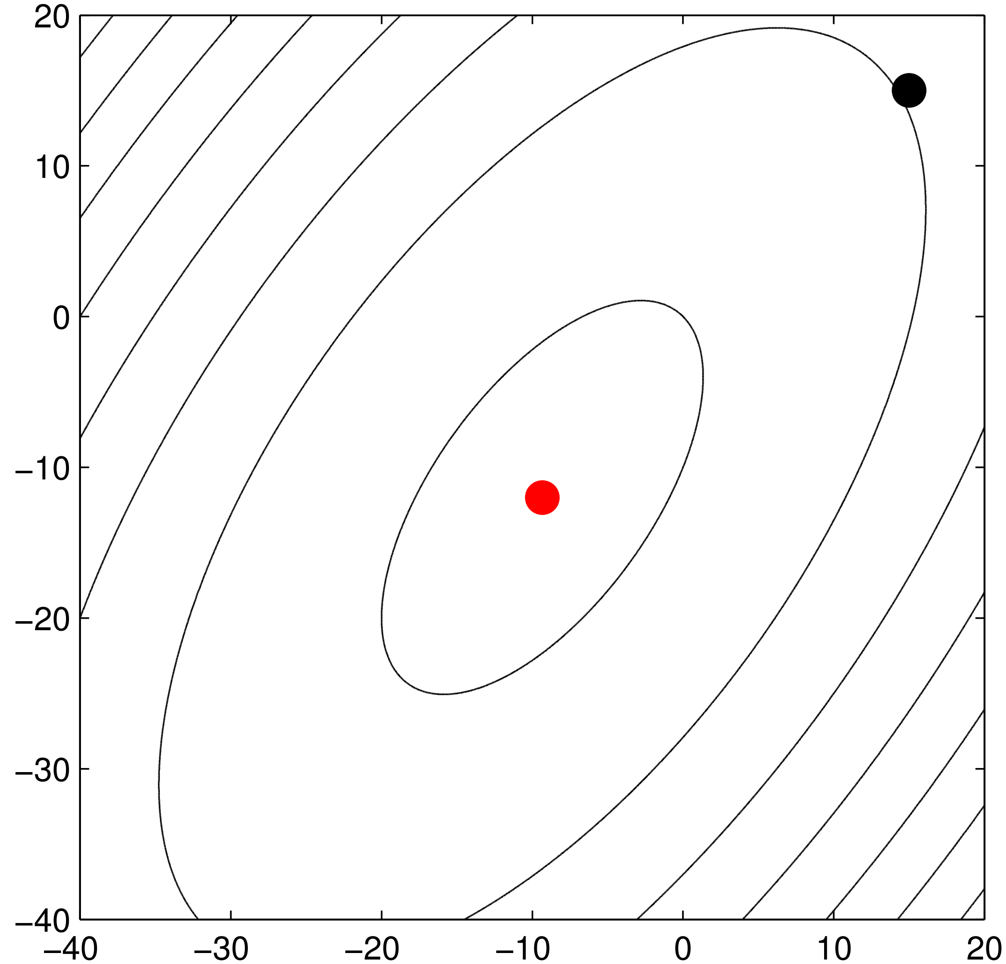
Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat

$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$

$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$

$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$

until



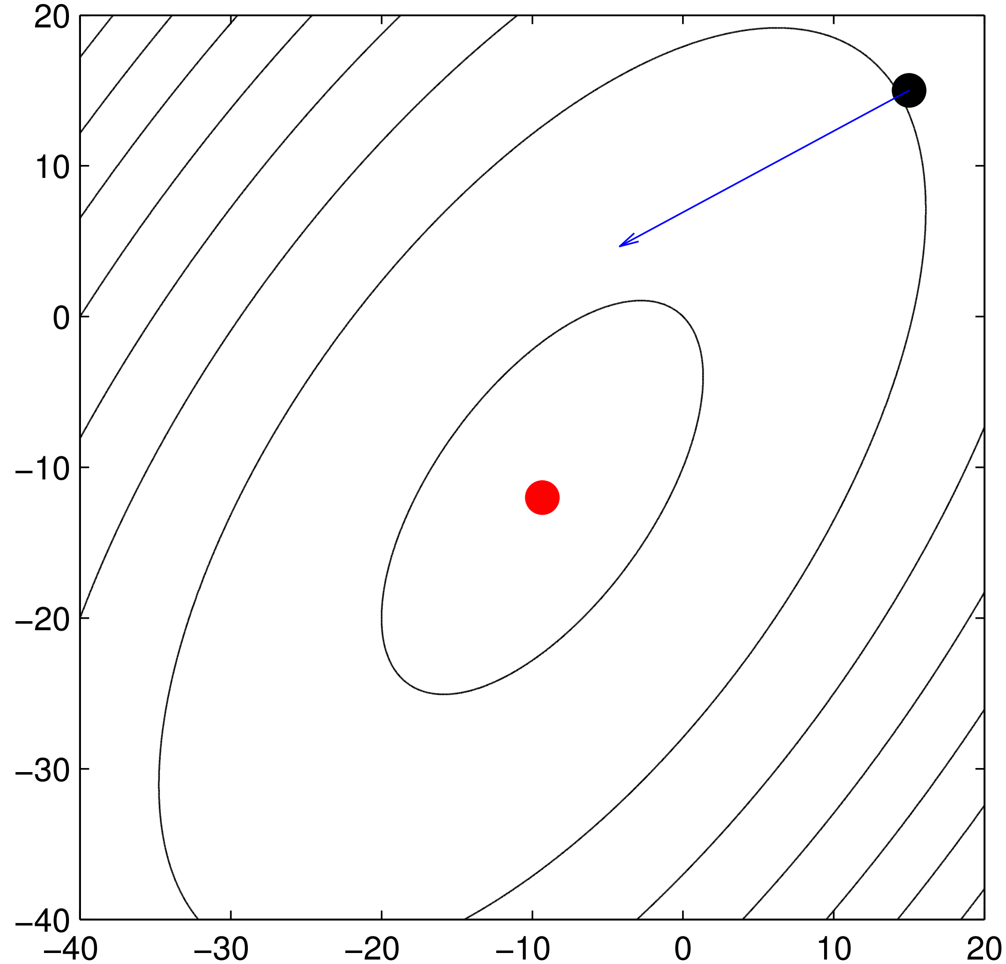
Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$
kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat

$$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$$
$$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$$
$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$$

until



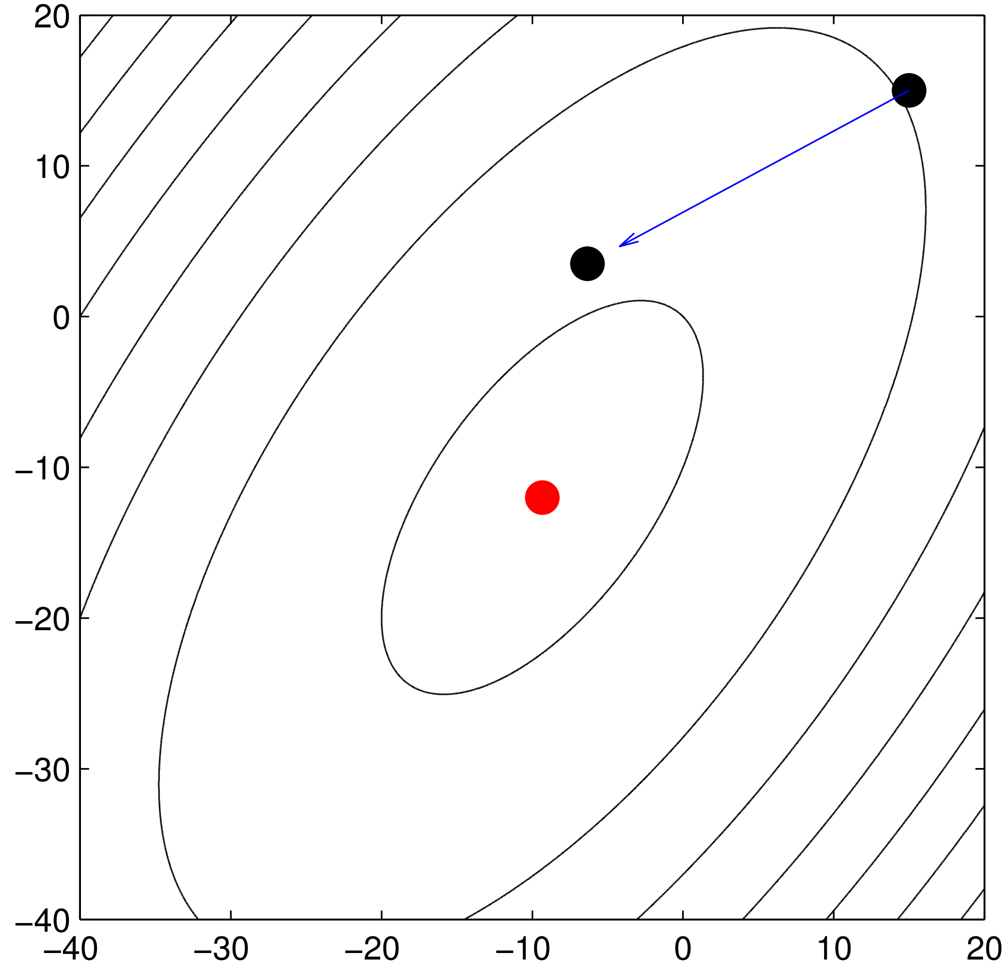
Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$
kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat

$$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$$
$$\alpha_k = \min_{\alpha\geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$$
$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$$

until



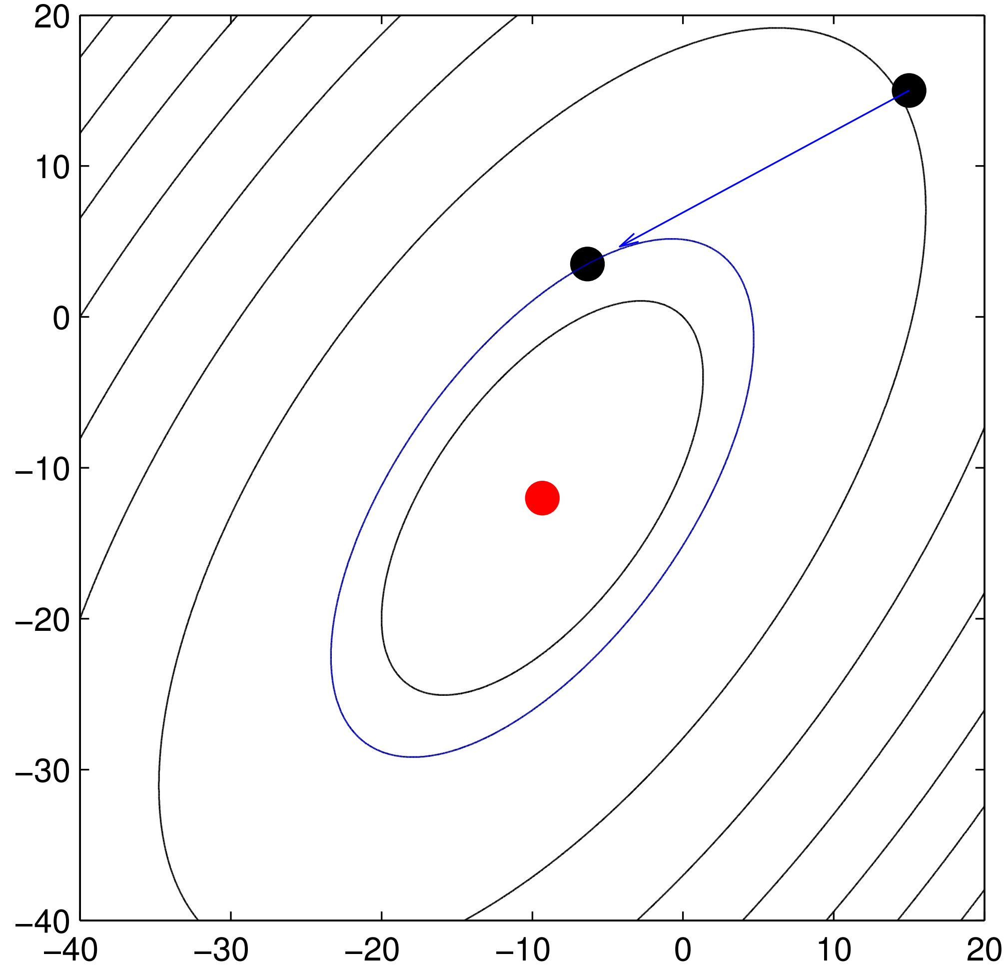
Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$ kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$.

Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat

$$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$$
$$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$$
$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$$

until



Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$
kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

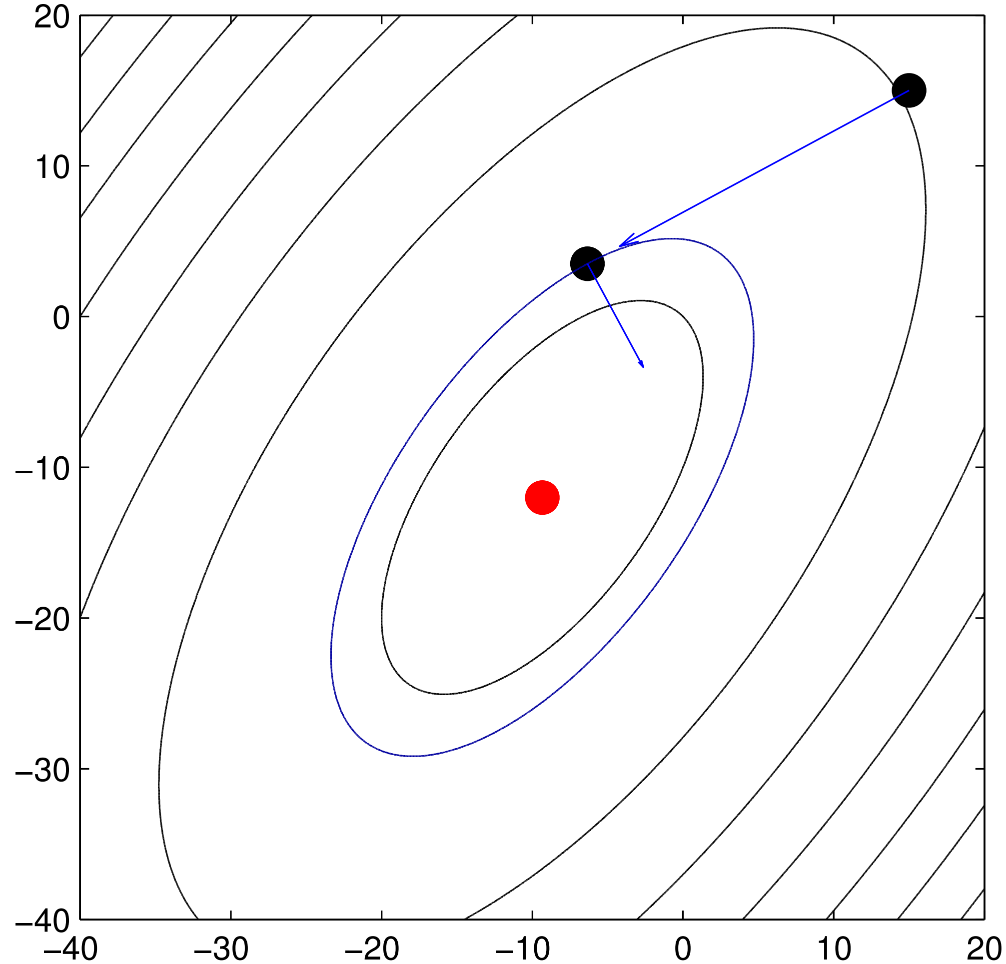
Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat

$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$

$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$

$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$

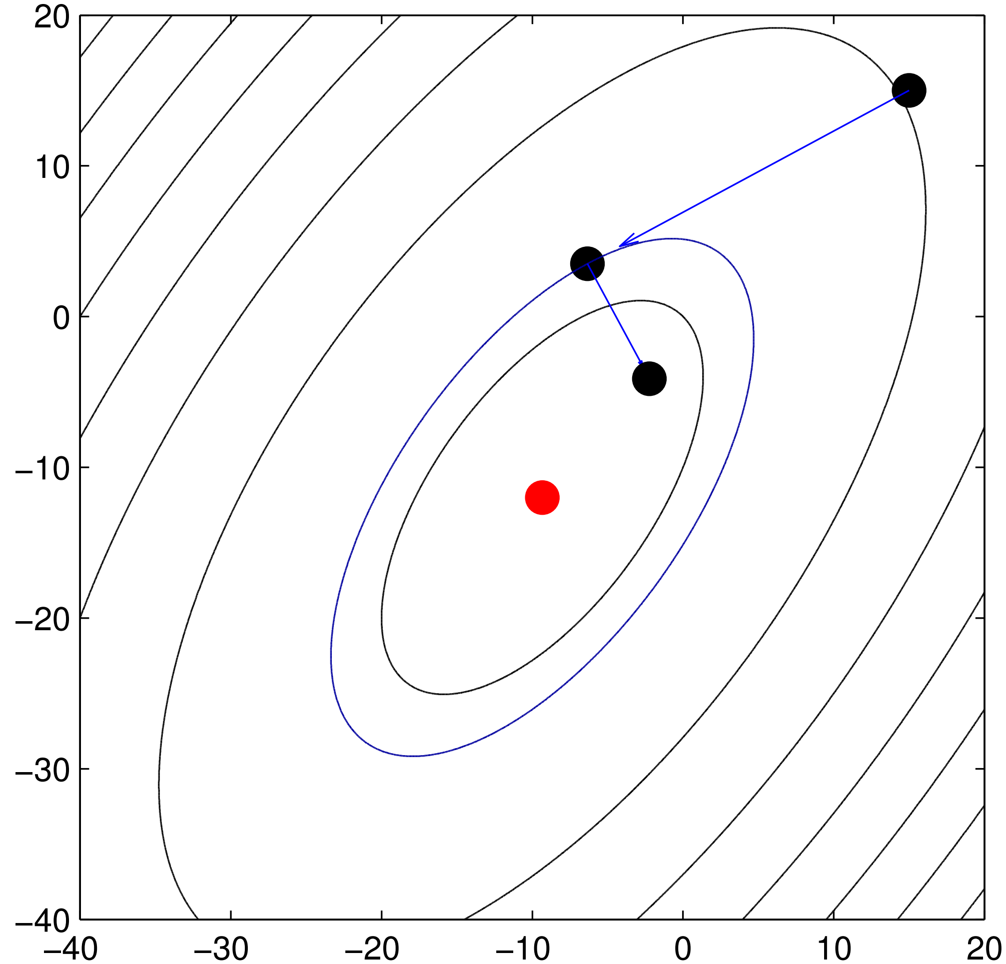
until



Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$
kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right]$.

Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat
$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$
$\alpha_k = \min\limits_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$
$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$
until



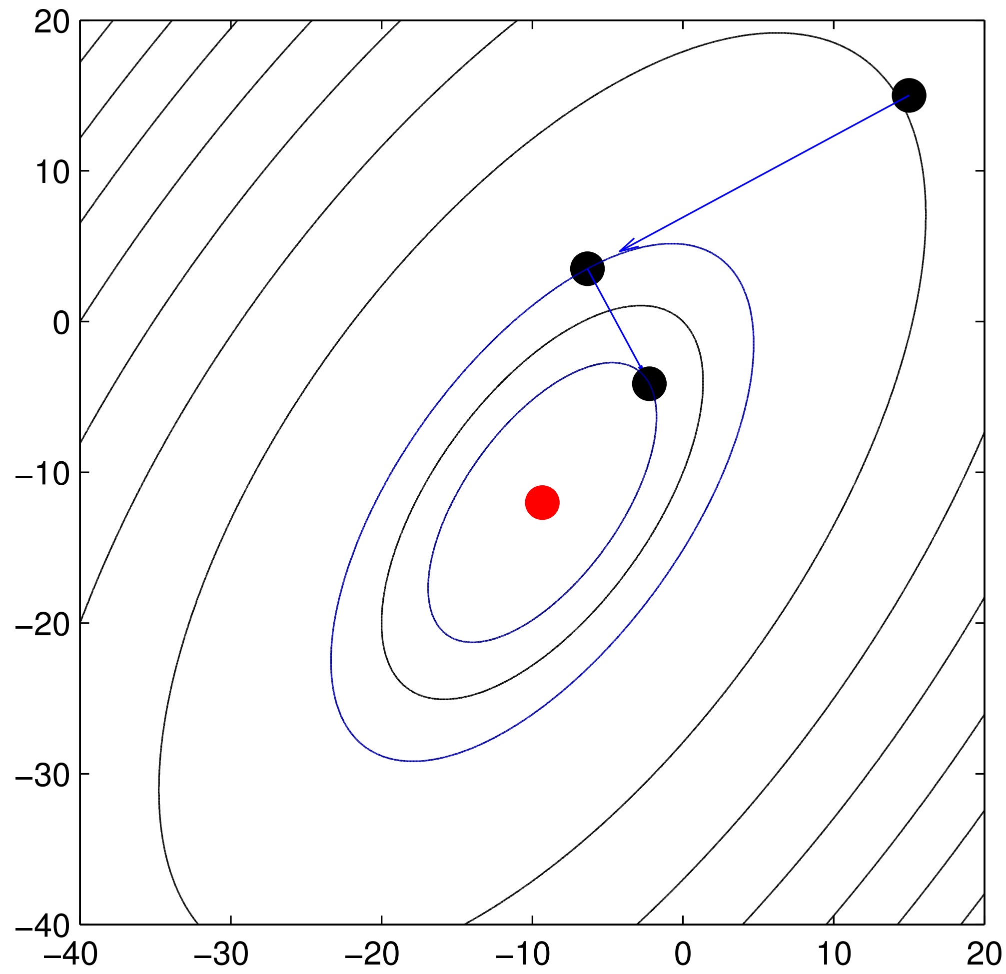
Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$
kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat

$$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$$
$$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$$
$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$$

until



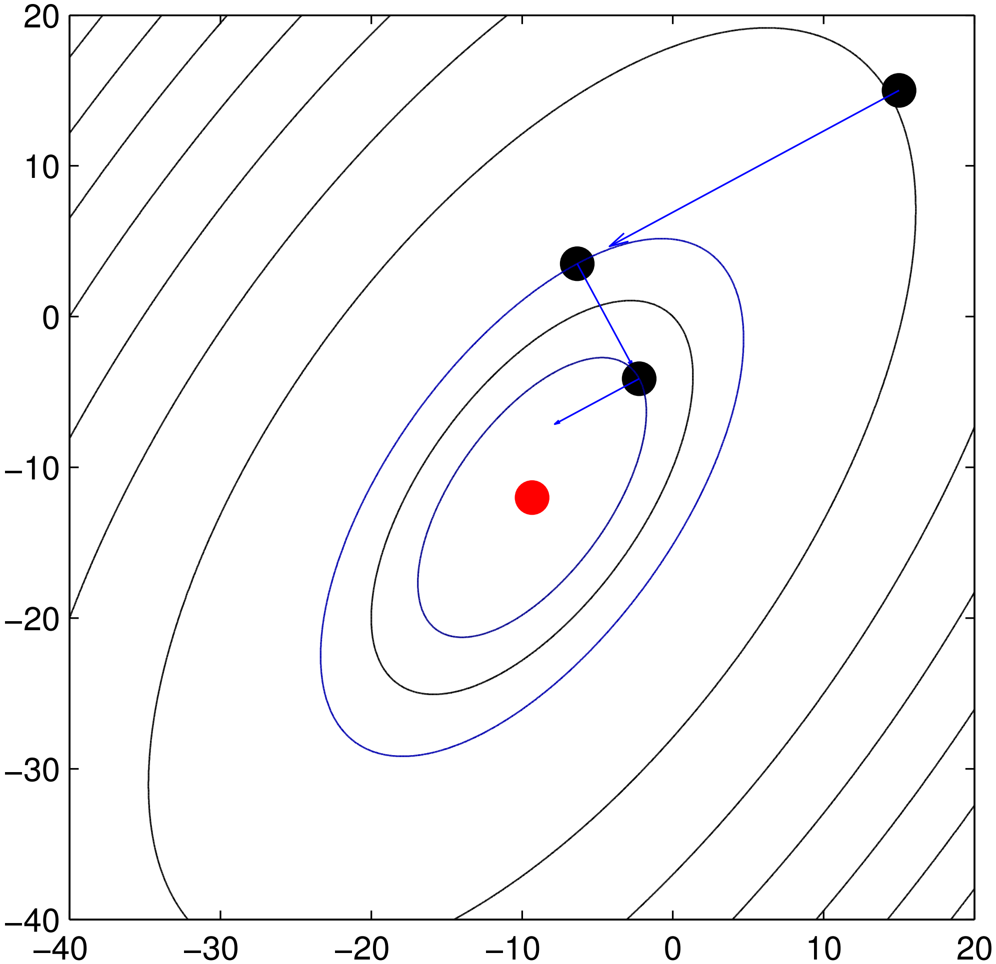
Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$ kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat

$$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$$
$$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$$
$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$$

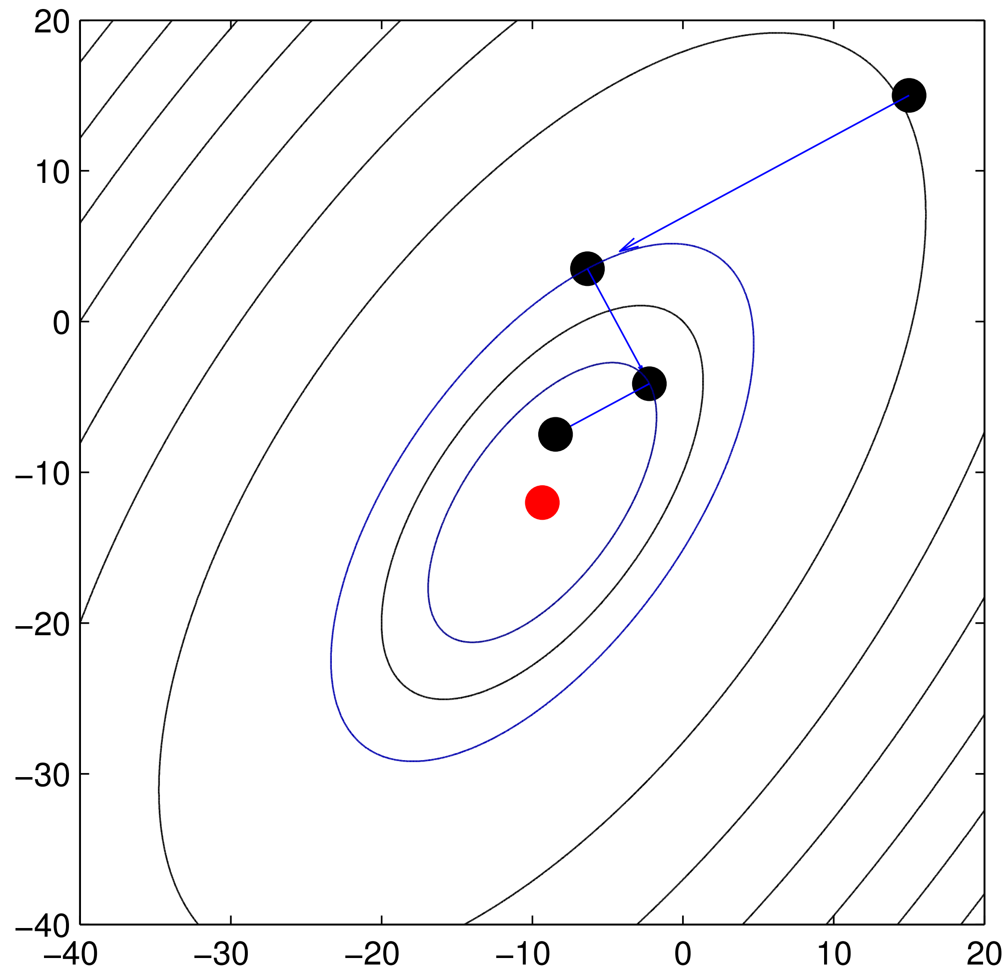
until



Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$ kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat
$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$
$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$
$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$
until



Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$

kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$

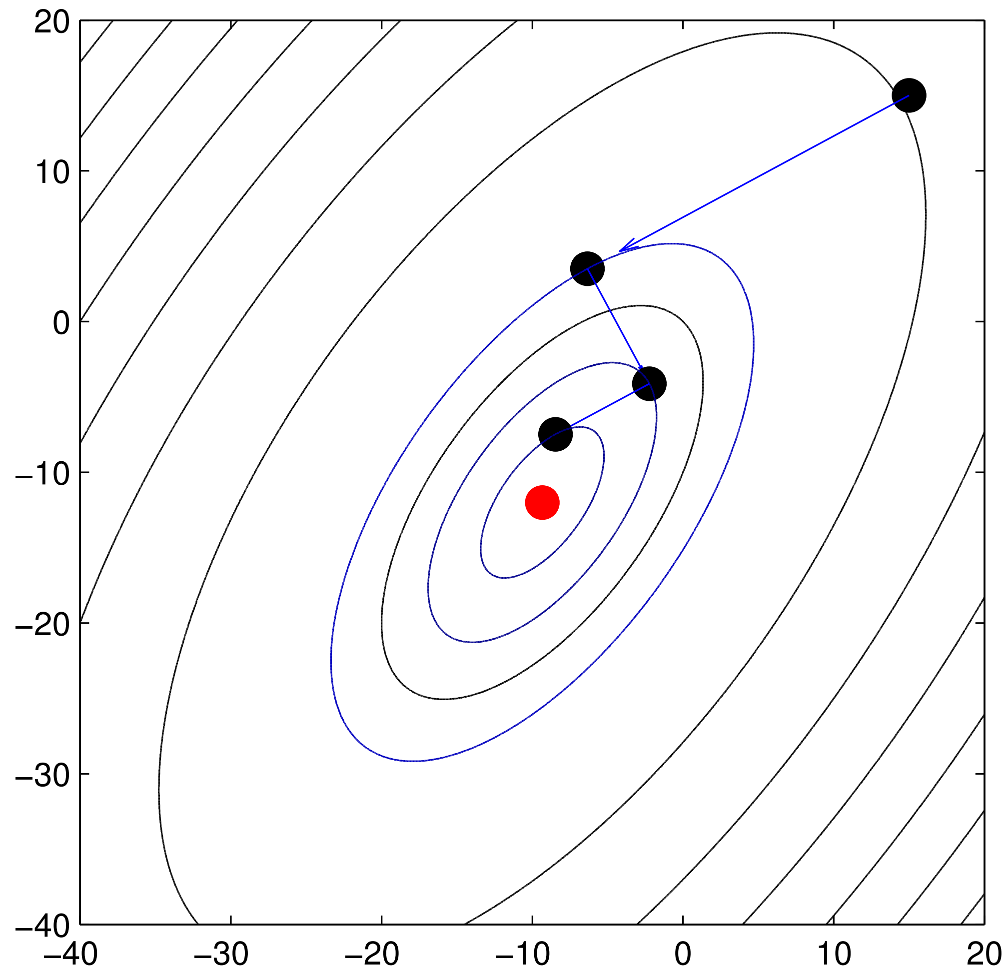
repeat

$$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$$

$$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$$

until



Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$
kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

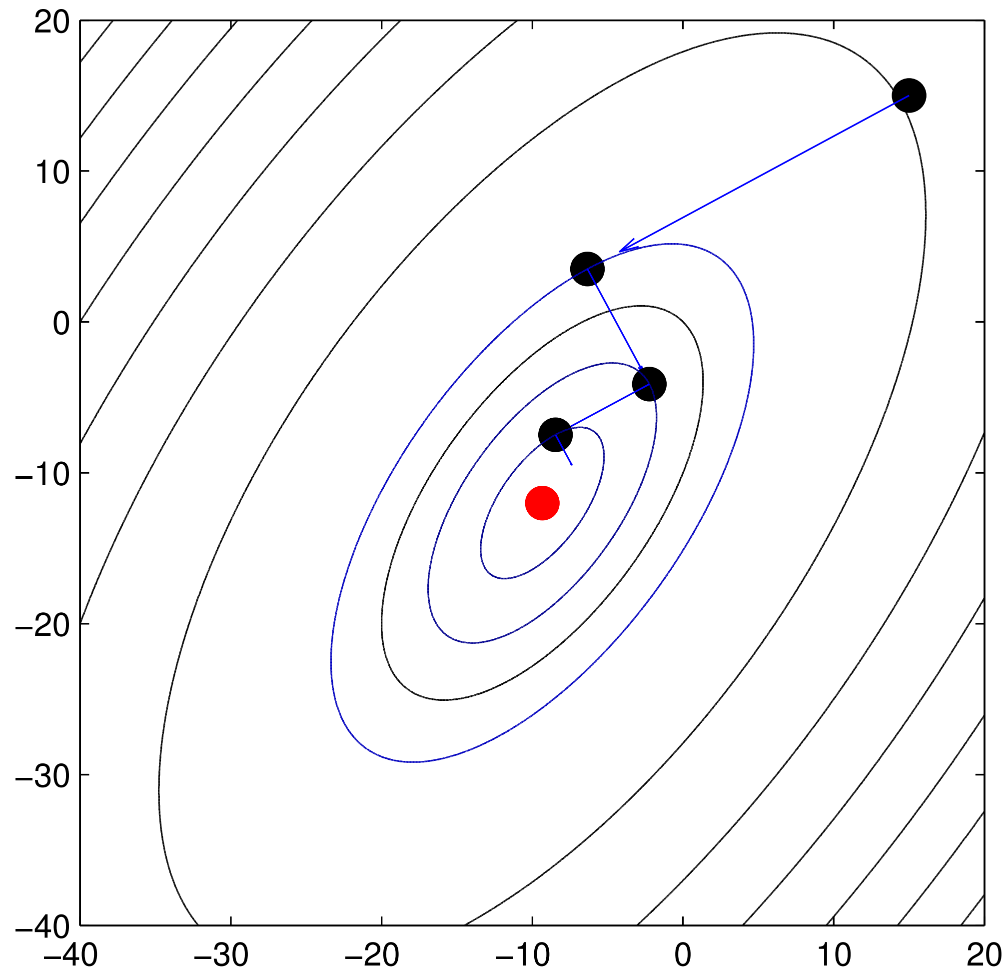
Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat

$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$

$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$

$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$

until



Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$ kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right]$.

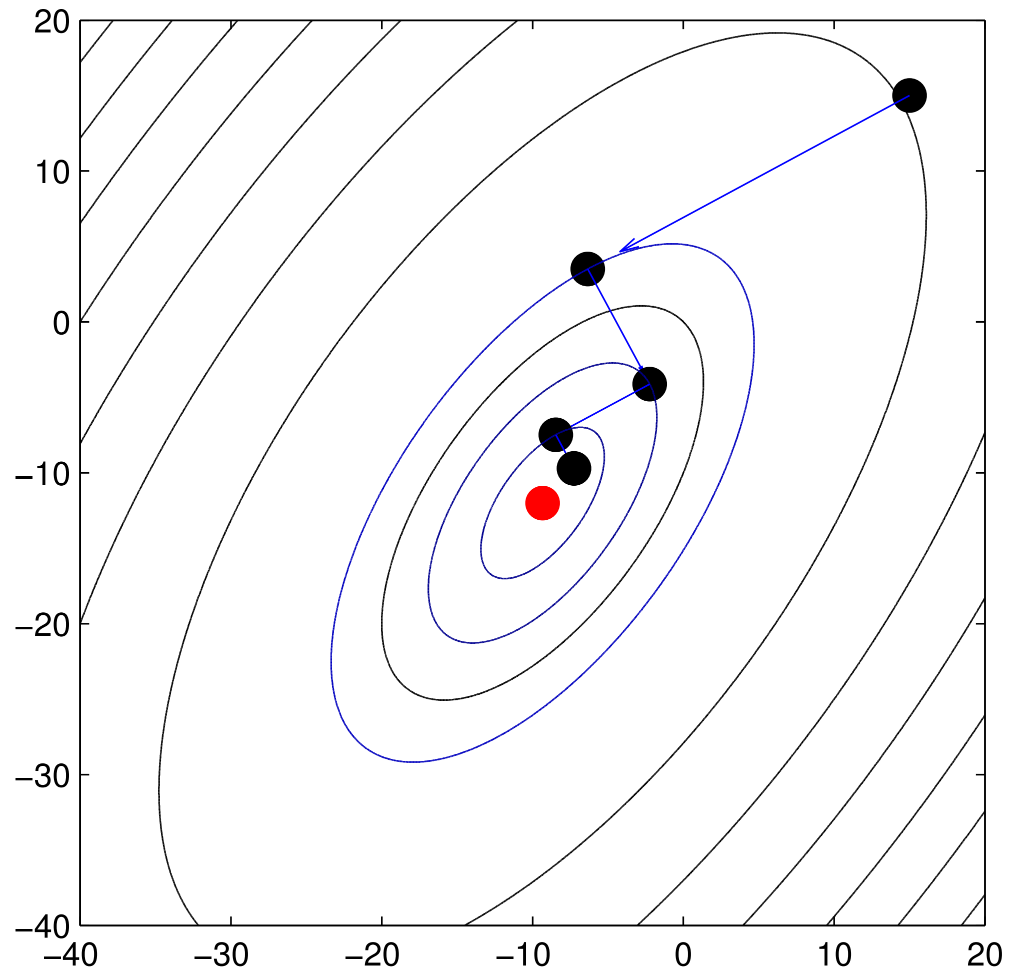
Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat

$$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$$

$$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$$

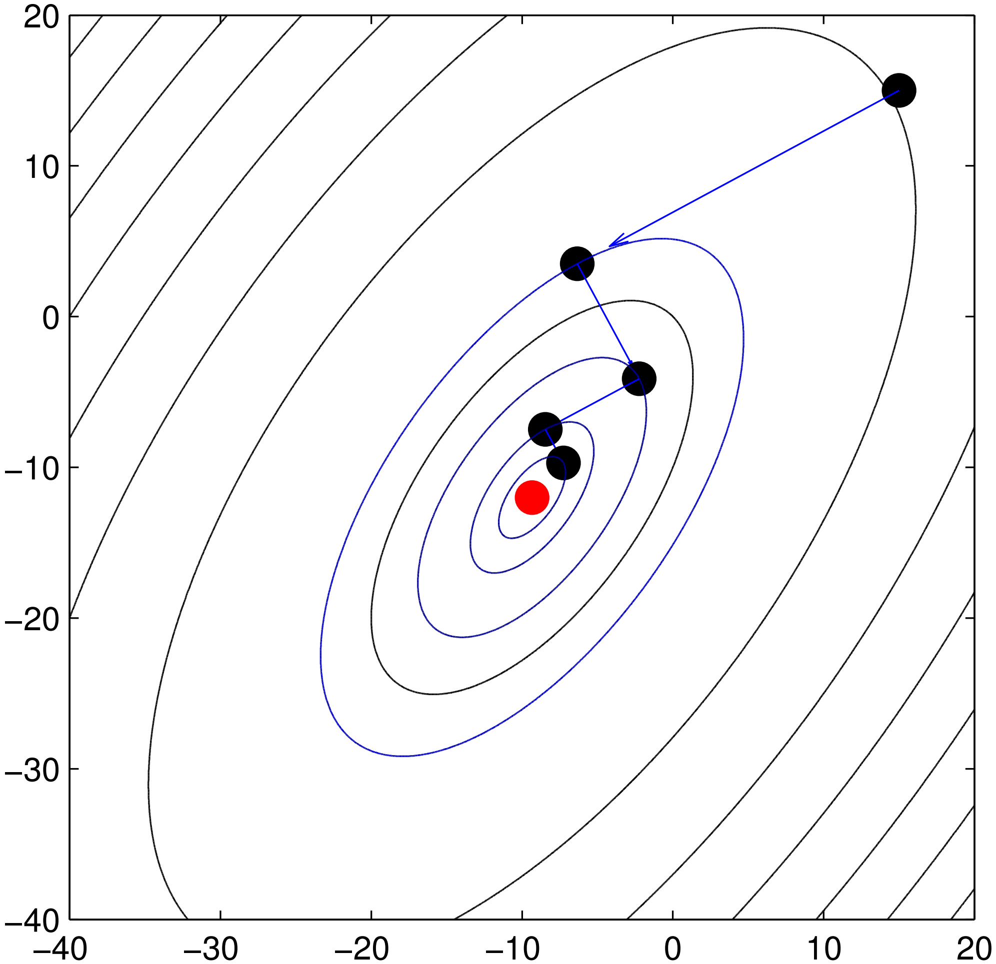
until



Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$
kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

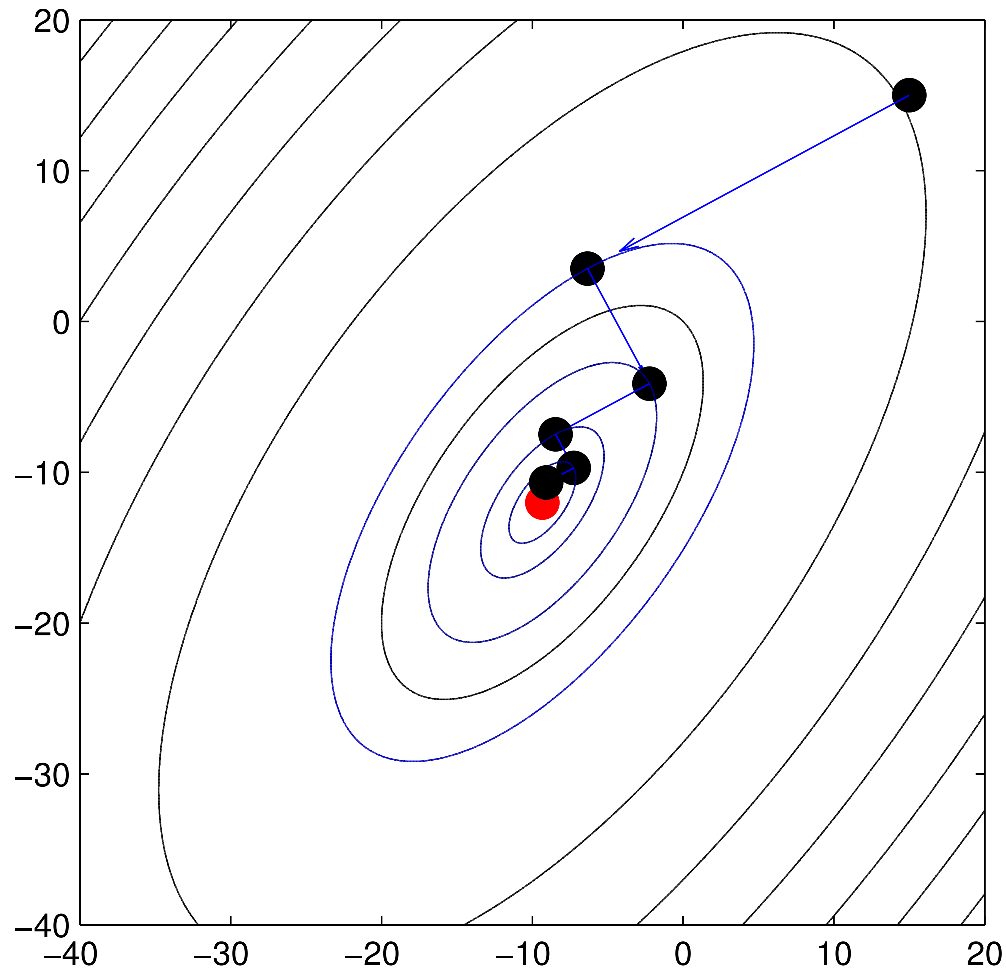
Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat
$$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$$
$$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$$
$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$$
until



Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$
kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat
$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$
$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$
$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$
until



Gredientní metoda

Příklad: Minimalizace kvadratické funkce $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x}$
kde $\mathbf{H} = \begin{bmatrix} 0.3 & -0.15 \\ -0.14 & 0.2 \end{bmatrix}$ a $\mathbf{h} = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

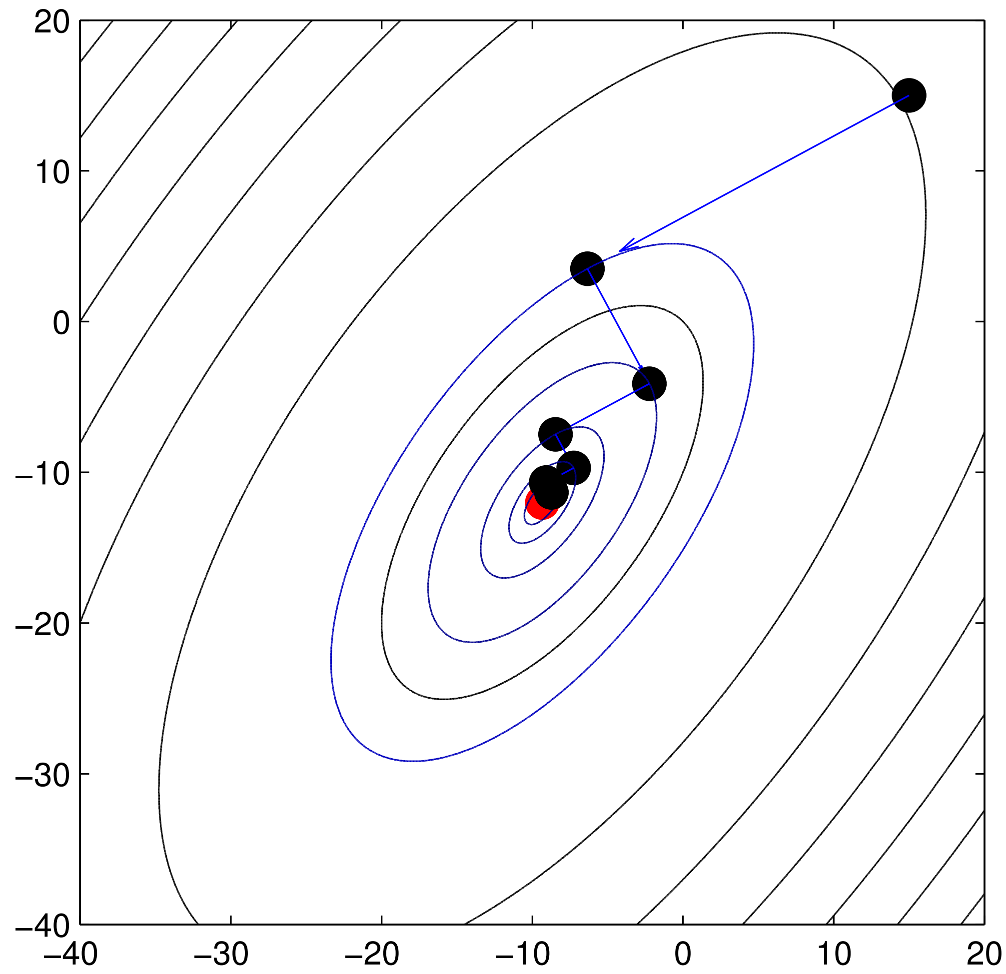
Initial solution $\mathbf{x}_1 \in \mathbb{R}^n$
repeat

$f'(\mathbf{x}_k) = \mathbf{H}\mathbf{x}_k + \mathbf{h}$

$\alpha_k = \min_{\alpha \geq 0} f(\mathbf{x}_k - \alpha f'(\mathbf{x}_k))$

$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k f'(\mathbf{x}_k)$

until



Lineární transformace souřadnic

- Transformujme souřadnice $\mathbf{x}$ lineární transformací $\tilde{\mathbf{x}} = \mathbf{A}\mathbf{x}$ kde $\mathbf{A}$ je regulární.

Lineární transformace souřadnic

- Transformujme souřadnice $\mathbf{x}$ lineární transformací $\tilde{\mathbf{x}} = \mathbf{A}\mathbf{x}$ kde $\mathbf{A}$ je regulární.
- Minimalizační úloha v nové soustavě bude mít stejné optimum

$$\min_{\mathbf{x}} f(\mathbf{x}) = \min_{\tilde{\mathbf{x}}} \tilde{f}(\tilde{\mathbf{x}}) \qquad \text{kde} \qquad f(\mathbf{x}) = f(\mathbf{A}^{-1}\tilde{\mathbf{x}}) = \tilde{f}(\tilde{\mathbf{x}})$$

Lineární transformace souřadnic

- Transformujme souřadnice $\mathbf{x}$ lineární transformací $\tilde{\mathbf{x}} = \mathbf{A}\mathbf{x}$ kde $\mathbf{A}$ je regulární.
- Minimalizační úloha v nové soustavě bude mít stejné optimum

$$\min_{\mathbf{x}} f(\mathbf{x}) = \min_{\tilde{\mathbf{x}}} \tilde{f}(\tilde{\mathbf{x}}) \qquad \text{kde} \qquad f(\mathbf{x}) = f(\mathbf{A}^{-1}\tilde{\mathbf{x}}) = \tilde{f}(\tilde{\mathbf{x}})$$

- Gradientní metoda v nové soustavě má tvar

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k - \alpha_k \tilde{f}'(\tilde{\mathbf{x}}_k)^T$$

Lineární transformace souřadnic

- Transformujme souřadnice $\mathbf{x}$ lineární transformací $\tilde{\mathbf{x}} = \mathbf{A}\mathbf{x}$ kde $\mathbf{A}$ je regulární.
- Minimalizační úloha v nové soustavě bude mít stejné optimum

$$\min_{\mathbf{x}} f(\mathbf{x}) = \min_{\tilde{\mathbf{x}}} \tilde{f}(\tilde{\mathbf{x}}) \qquad \text{kde} \qquad f(\mathbf{x}) = f(\mathbf{A}^{-1}\tilde{\mathbf{x}}) = \tilde{f}(\tilde{\mathbf{x}})$$

- Gradientní metoda v nové soustavě má tvar

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k - \alpha_k \tilde{f}'(\tilde{\mathbf{x}}_k)^T$$

Po dosazení $\tilde{f}'(\tilde{\mathbf{x}}) = f'(\mathbf{x})\mathbf{A}^{-1}$ a $\tilde{\mathbf{x}} = \mathbf{A}\mathbf{x}$ dostaneme iteraci v původních souřadnicích

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k(\mathbf{A}^T\mathbf{A})^{-1} f'(\mathbf{x}_k)^T$$

Lineární transformace souřadnic

- Transformujme souřadnice $\mathbf{x}$ lineární transformací $\tilde{\mathbf{x}} = \mathbf{A}\mathbf{x}$ kde $\mathbf{A}$ je regulární.
- Minimalizační úloha v nové soustavě bude mít stejné optimum

$$\min_{\mathbf{x}} f(\mathbf{x}) = \min_{\tilde{\mathbf{x}}} \tilde{f}(\tilde{\mathbf{x}}) \qquad \text{kde} \qquad f(\mathbf{x}) = f(\mathbf{A}^{-1}\tilde{\mathbf{x}}) = \tilde{f}(\tilde{\mathbf{x}})$$

- Gradientní metoda v nové soustavě má tvar

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k - \alpha_k \tilde{f}'(\tilde{\mathbf{x}}_k)^T$$

Po dosazení $\tilde{f}'(\tilde{\mathbf{x}}) = f'(\mathbf{x})\mathbf{A}^{-1}$ a $\tilde{\mathbf{x}} = \mathbf{A}\mathbf{x}$ dostaneme iteraci v původních souřadnicích

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k (\mathbf{A}^T\mathbf{A})^{-1} f'(\mathbf{x}_k)^T$$

Směr $\mathbf{s}_k = -(\mathbf{A}^T\mathbf{A})^{-1} f'(\mathbf{x}_k)^T$ je sestupný, protože $-f'(\mathbf{x}_k)(\mathbf{A}^T\mathbf{A})^{-1} f'(\mathbf{x}_k)^T < 0$

Lineární transformace souřadnic

- Transformujme souřadnice $\mathbf{x}$ lineární transformací $\tilde{\mathbf{x}} = \mathbf{A}\mathbf{x}$ kde $\mathbf{A}$ je regulární.
- Minimalizační úloha v nové soustavě bude mít stejné optimum

$$\min_{\mathbf{x}} f(\mathbf{x}) = \min_{\tilde{\mathbf{x}}} \tilde{f}(\tilde{\mathbf{x}}) \qquad \text{kde} \qquad f(\mathbf{x}) = f(\mathbf{A}^{-1}\tilde{\mathbf{x}}) = \tilde{f}(\tilde{\mathbf{x}})$$

- Gradientní metoda v nové soustavě má tvar

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k - \alpha_k \tilde{f}'(\tilde{\mathbf{x}}_k)^T$$

Po dosazení $\tilde{f}'(\tilde{\mathbf{x}}) = f'(\mathbf{x})\mathbf{A}^{-1}$ a $\tilde{\mathbf{x}} = \mathbf{A}\mathbf{x}$ dostaneme iteraci v původních souřadnicích

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k (\mathbf{A}^T\mathbf{A})^{-1} f'(\mathbf{x}_k)^T$$

Směr $\mathbf{s}_k = -(\mathbf{A}^T\mathbf{A})^{-1} f'(\mathbf{x}_k)^T$ je sestupný, protože $-f'(\mathbf{x}_k)(\mathbf{A}^T\mathbf{A})^{-1} f'(\mathbf{x}_k)^T < 0$

- Zobecnění: každá sestupná metoda lze zapsat

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{C}_k f'(\mathbf{x}_k)^T$$

Newtonova metoda pro řešení soustavy nelineárních rovnic

- Cíl: pro zadané diferencovatelné zobrazení $\mathbf{g}\colon \mathbb{R}^n \to \mathbb{R}^n$ chceme vyřešit rovnici

$$\mathbf{g}(\mathbf{x}) = \mathbf{0}$$

Tj. řešíme soustavu n rovnic (mohou být nelineární) s n neznámými.

Newtonova metoda pro řešení soustavy nelineárních rovnic

- Cíl: pro zadané diferencovatelné zobrazení $\mathbf{g}\colon \mathbb{R}^n \to \mathbb{R}^n$ chceme vyřešit rovnici

$$\mathbf{g}(\mathbf{x}) = \mathbf{0}$$

  Tj. řešíme soustavu n rovnic (mohou být nelineární) s n neznámými.

- Aproximujeme zobrazení $\mathbf{g}$ v okolí $\mathbf{x}_k$ Taylorovým polynomem prvního řádu

$$\mathbf{g}(\mathbf{x}) \approx \tilde{\mathbf{g}}(\mathbf{x}) = \mathbf{g}(\mathbf{x}_k) + \mathbf{g}'(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k)$$

Newtonova metoda pro řešení soustavy nelineárních rovnic

- Cíl: pro zadané diferencovatelné zobrazení $\mathbf{g}\colon \mathbb{R}^n \to \mathbb{R}^n$ chceme vyřešit rovnici

$$\mathbf{g}(\mathbf{x}) = \mathbf{0}$$

  Tj. řešíme soustavu n rovnic (mohou být nelineární) s n neznámými.

- Aproximujeme zobrazení $\mathbf{g}$ v okolí $\mathbf{x}_k$ Taylorovým polynomem prvního řádu

$$\mathbf{g}(\mathbf{x}) \approx \tilde{\mathbf{g}}(\mathbf{x}) = \mathbf{g}(\mathbf{x}_k) + \mathbf{g}'(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k)$$

- Začneme z odhadu $\mathbf{x}_1$ a další iteraci najdeme řešením lineární soustavy

$$\tilde{\mathbf{g}}(\mathbf{x}_{k+1}) = \mathbf{0}$$

Newtonova metoda pro řešení soustavy nelineárních rovnic

- Cíl: pro zadané diferencovatelné zobrazení $\mathbf{g}\colon \mathbb{R}^n \to \mathbb{R}^n$ chceme vyřešit rovnici

$$\mathbf{g}(\mathbf{x}) = \mathbf{0}$$

  Tj. řešíme soustavu n rovnic (mohou být nelineární) s n neznámými.

- Aproximujeme zobrazení $\mathbf{g}$ v okolí $\mathbf{x}_k$ Taylorovým polynomem prvního řádu

$$\mathbf{g}(\mathbf{x}) \approx \tilde{\mathbf{g}}(\mathbf{x}) = \mathbf{g}(\mathbf{x}_k) + \mathbf{g}'(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k)$$

- Začneme z odhadu $\mathbf{x}_1$ a další iteraci najdeme řešením lineární soustavy

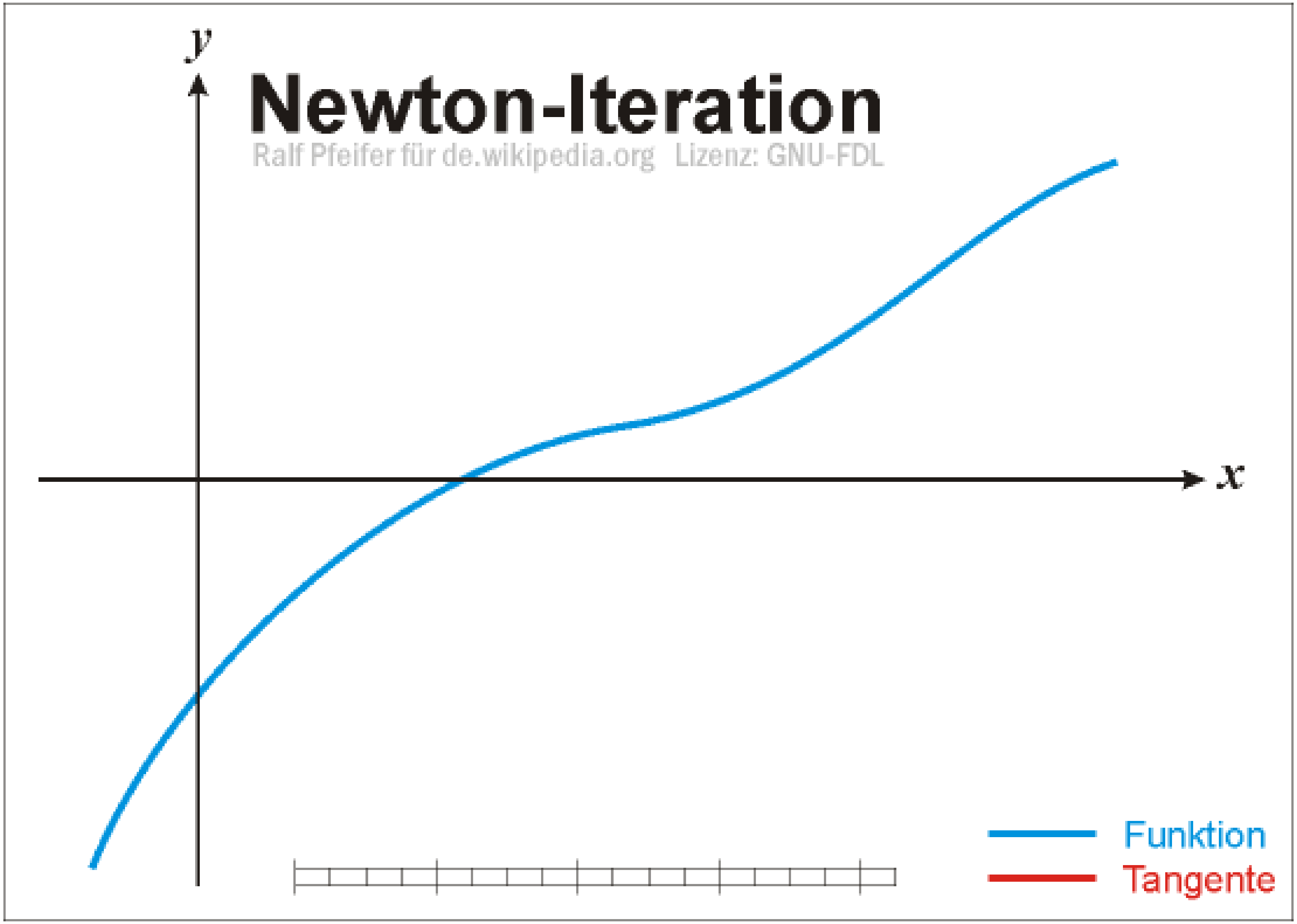
$$\tilde{\mathbf{g}}(\mathbf{x}_{k+1}) = \mathbf{0}$$

- Pokud je Jacobiho matice $\mathbf{g}'(\mathbf{x}_k) \in \mathbb{R}^{n \times n}$ regulární iterujeme

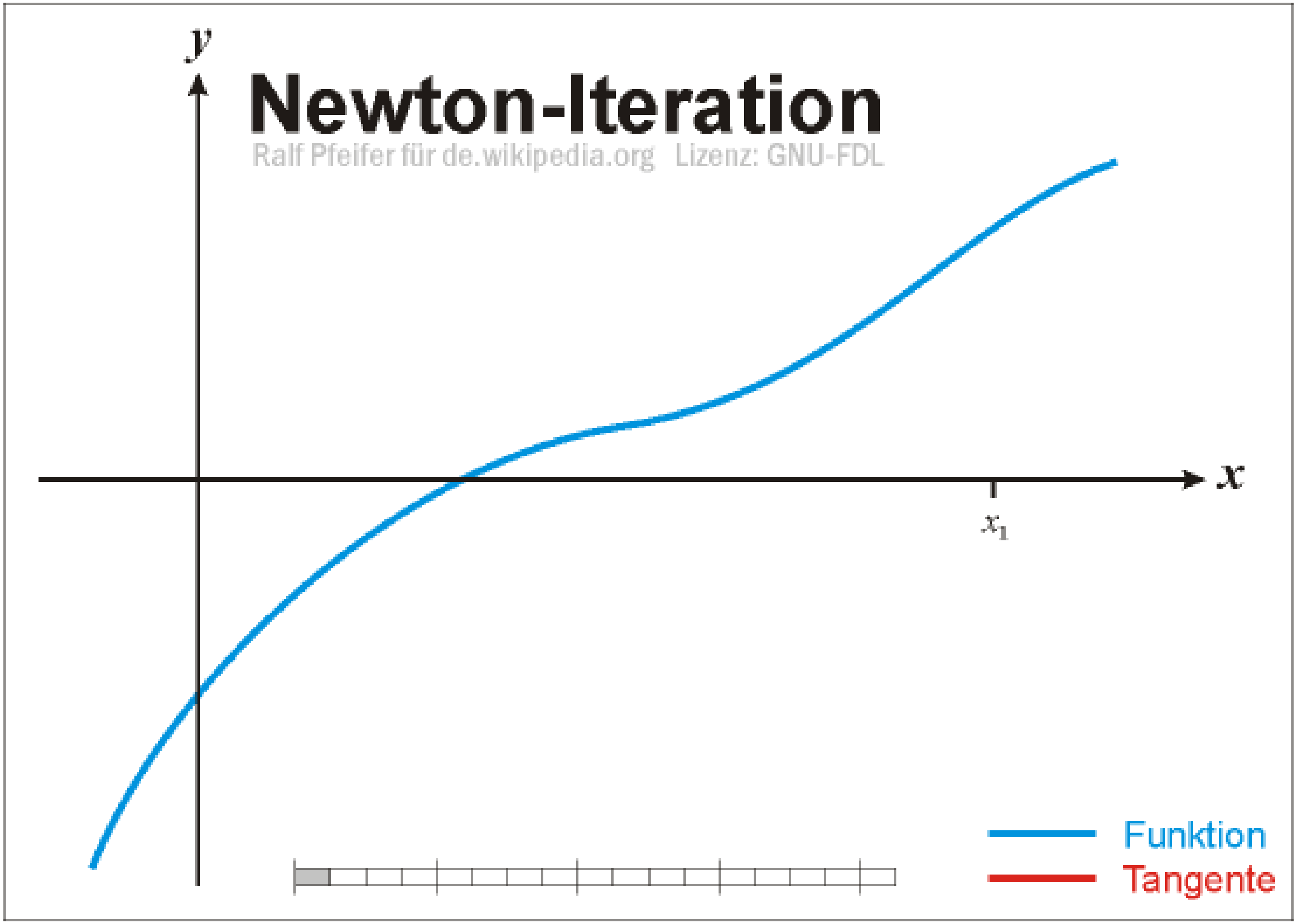
$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{g}'(\mathbf{x}_k)^{-1}\mathbf{g}(\mathbf{x}_k)$$

Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$

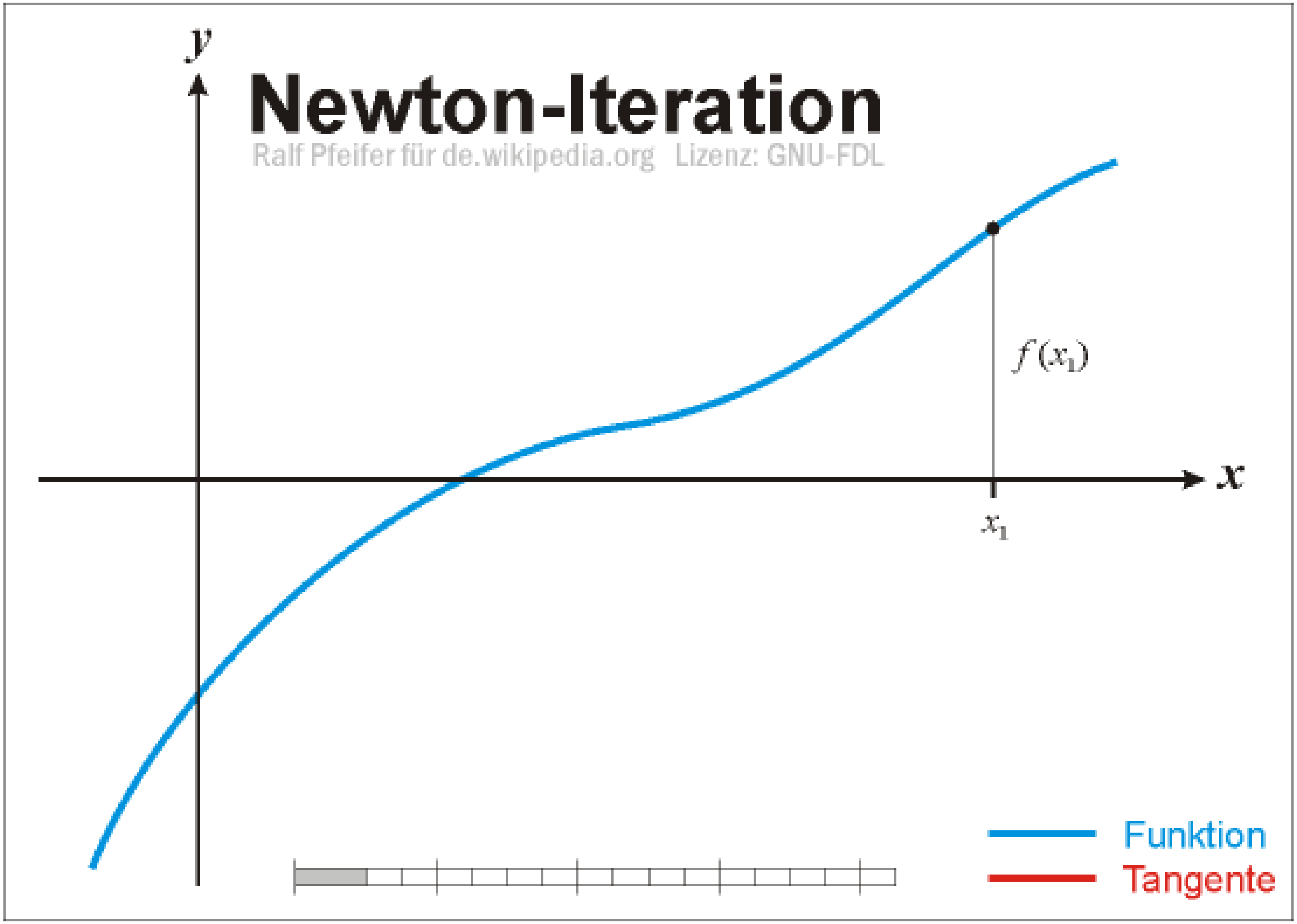


Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



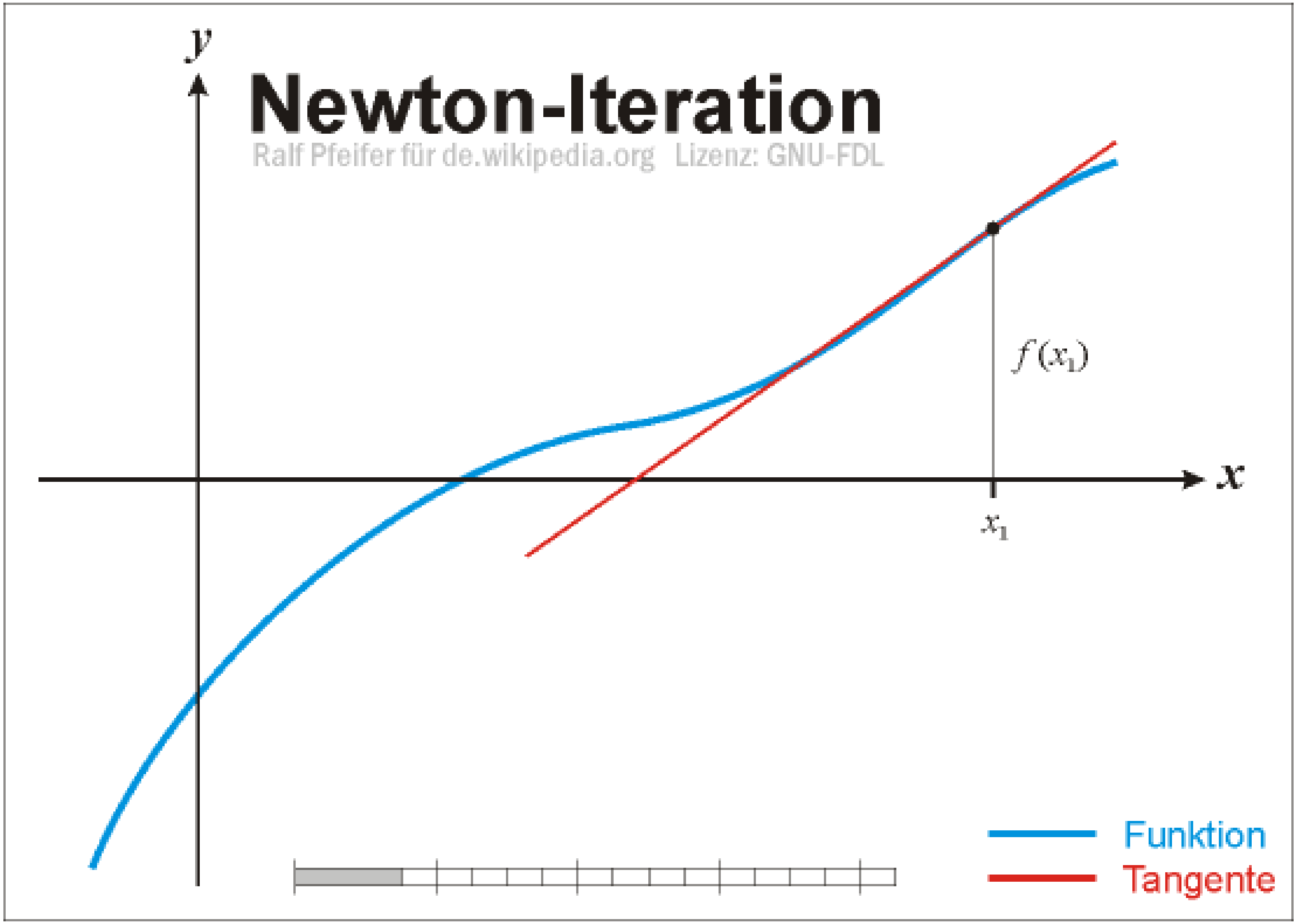
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



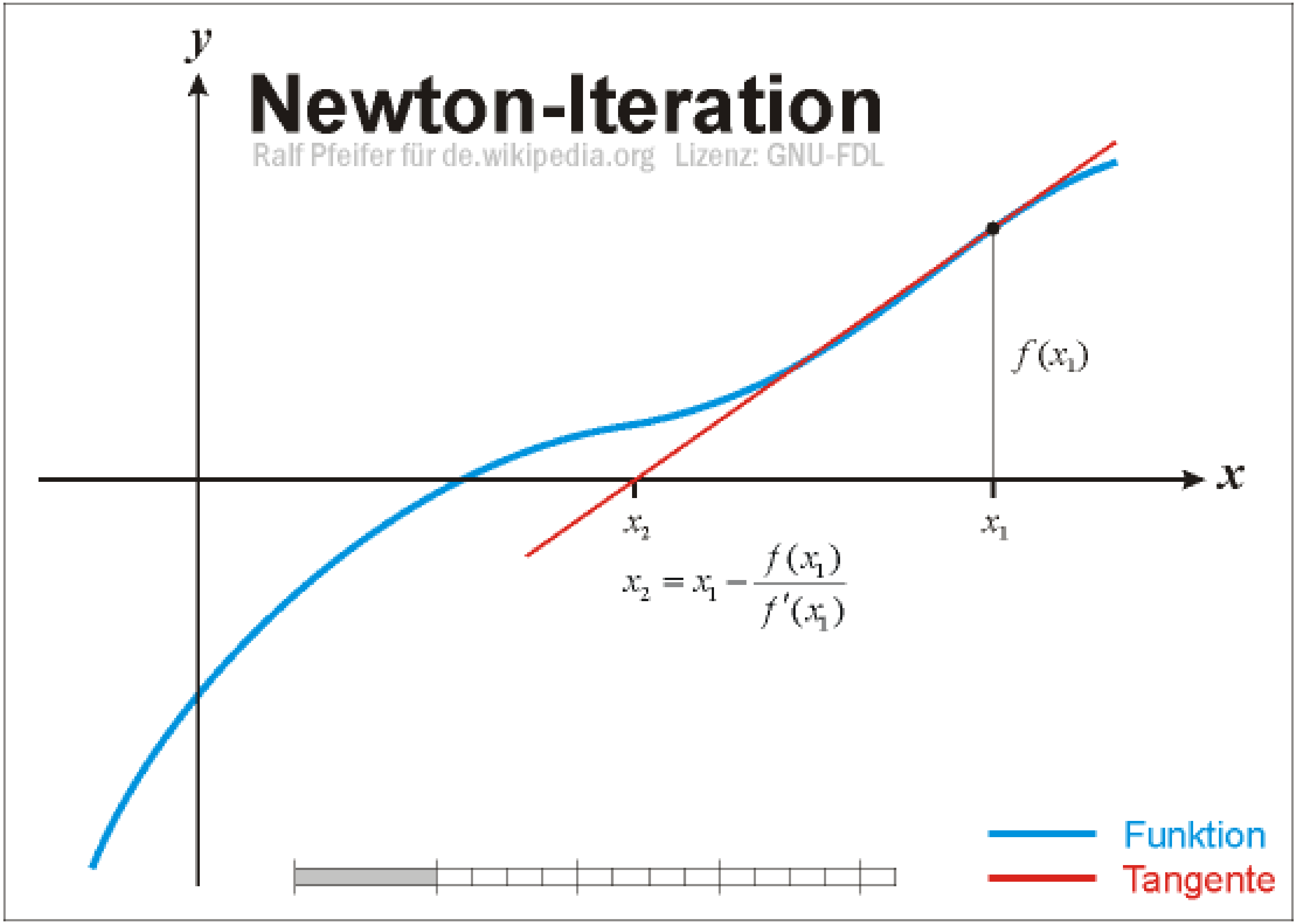
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



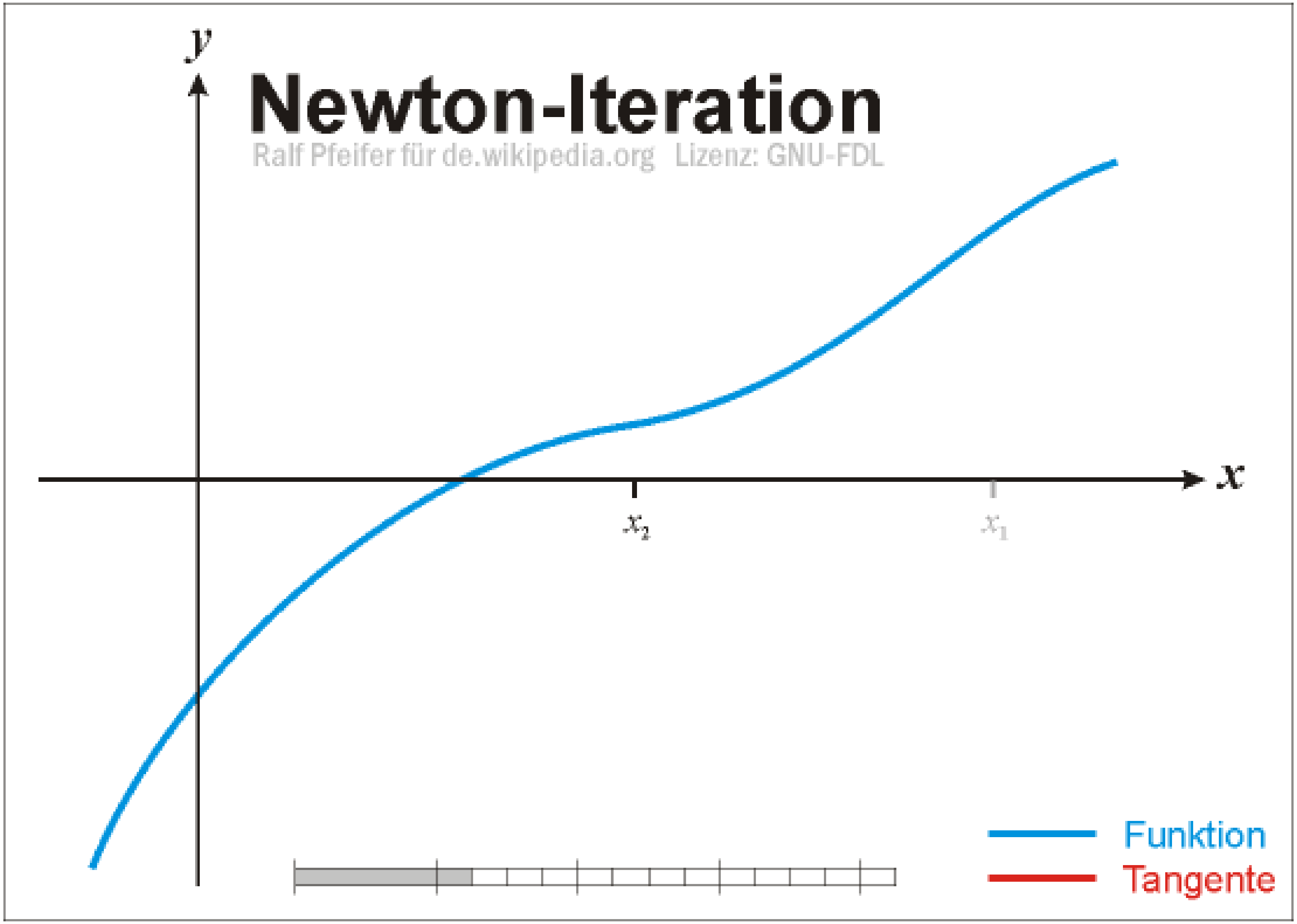
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



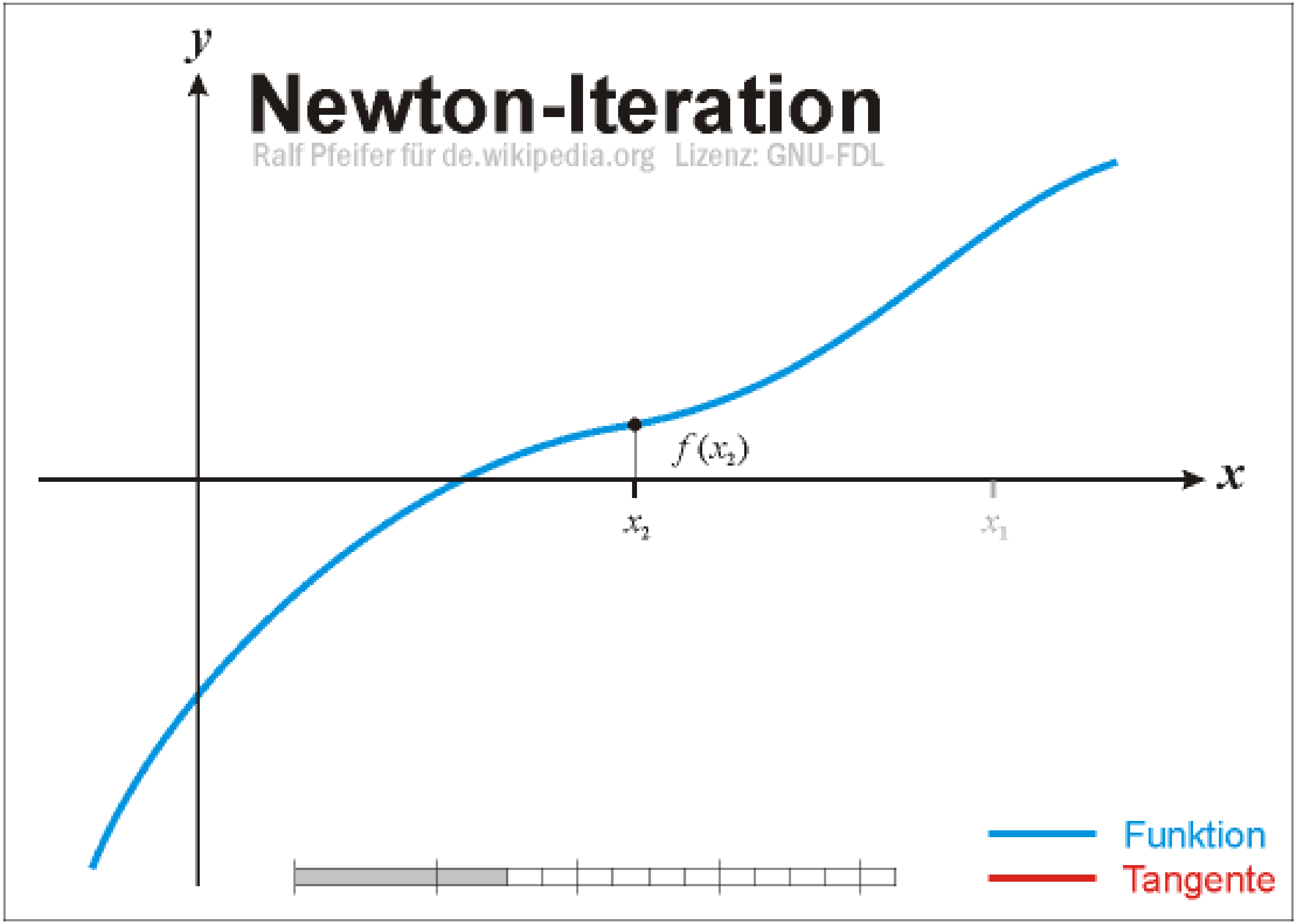
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



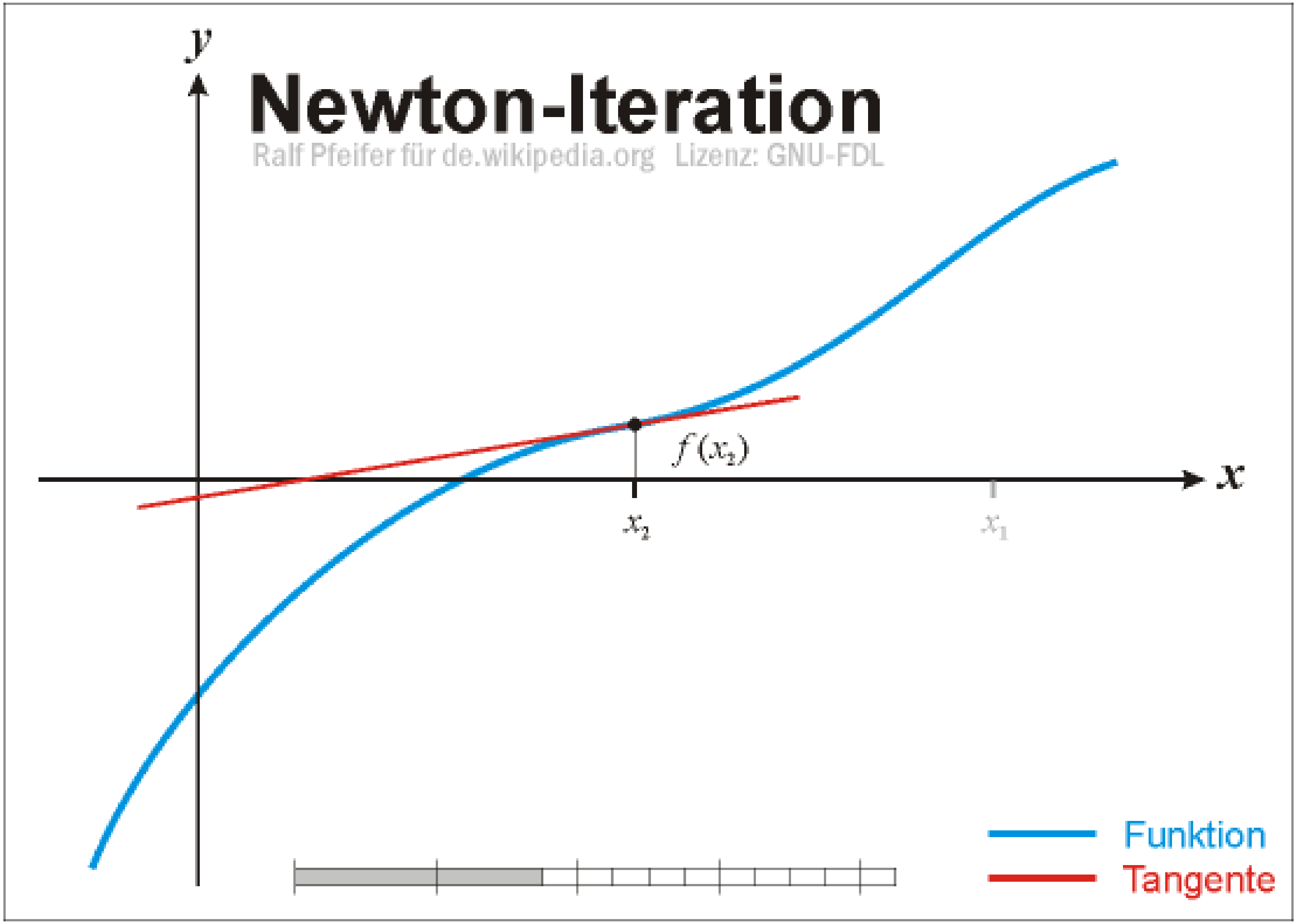
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



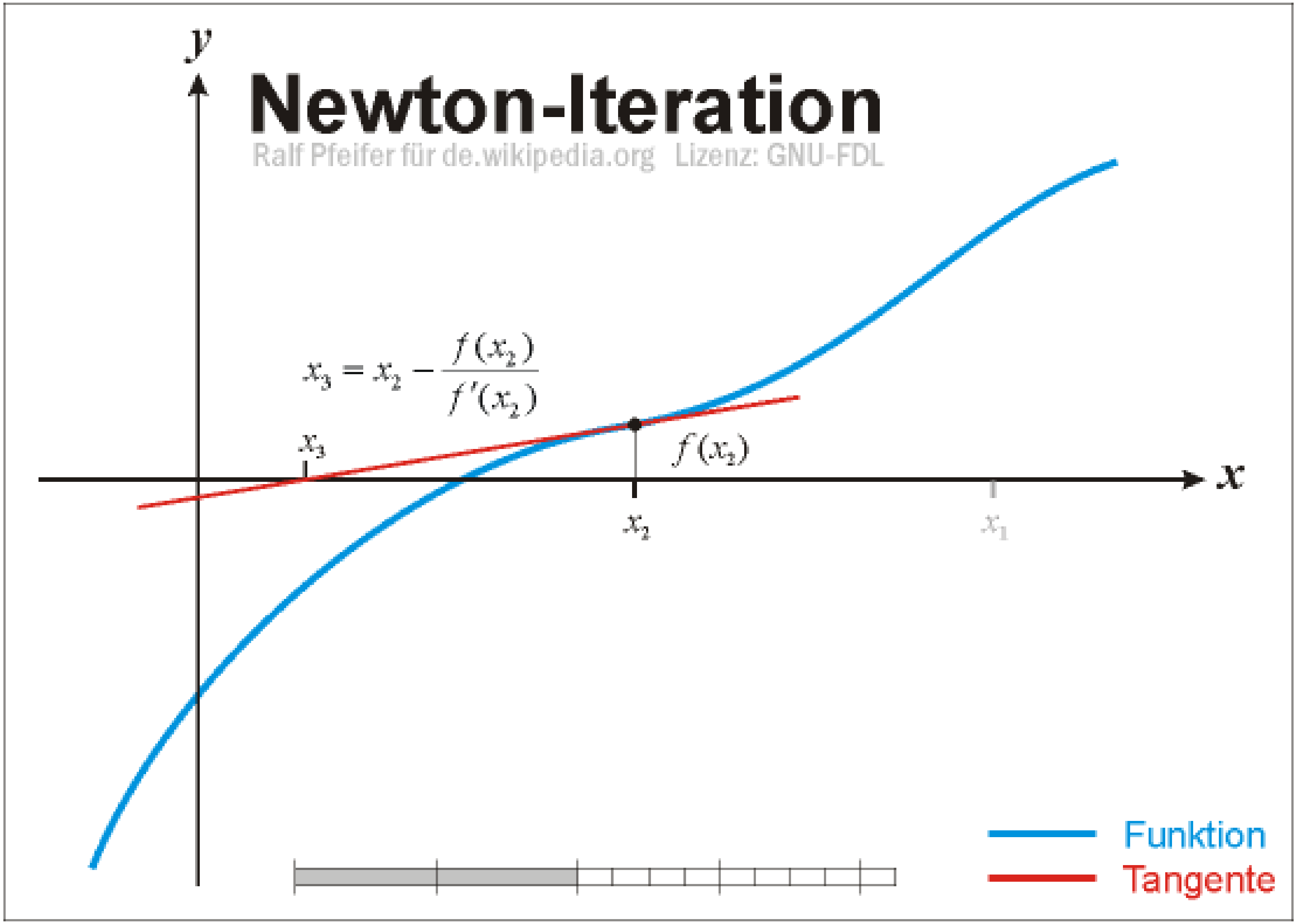
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



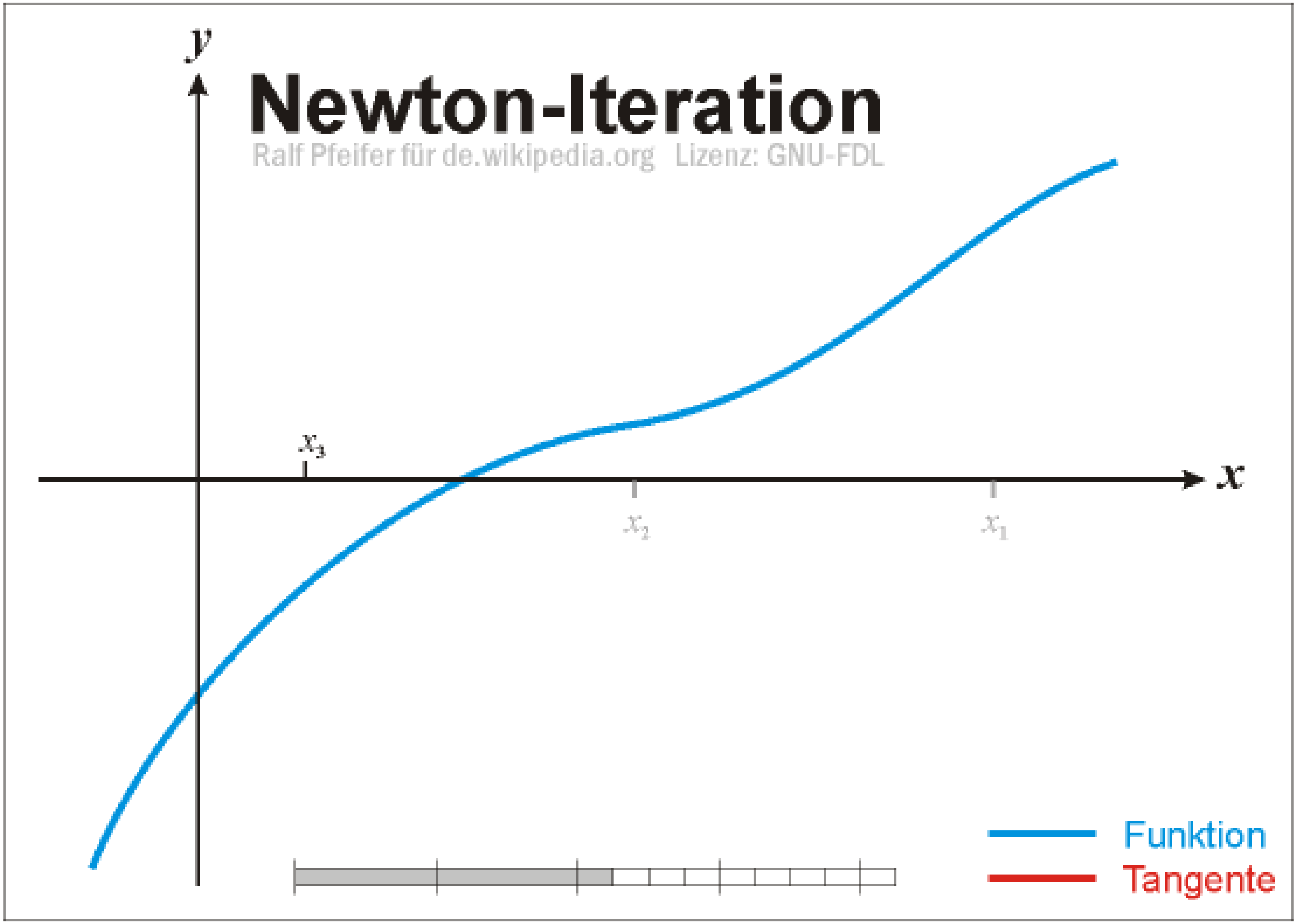
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



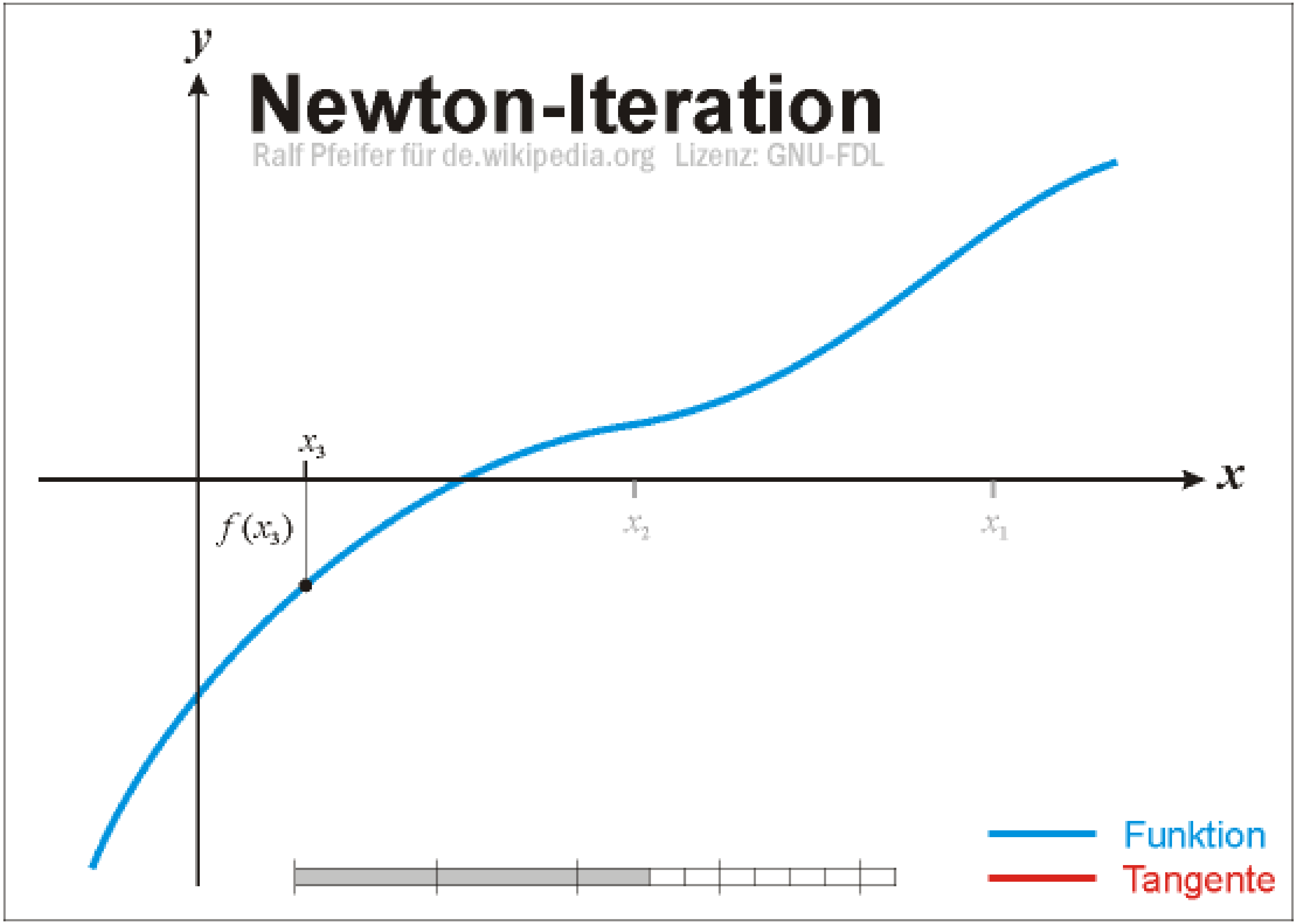
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



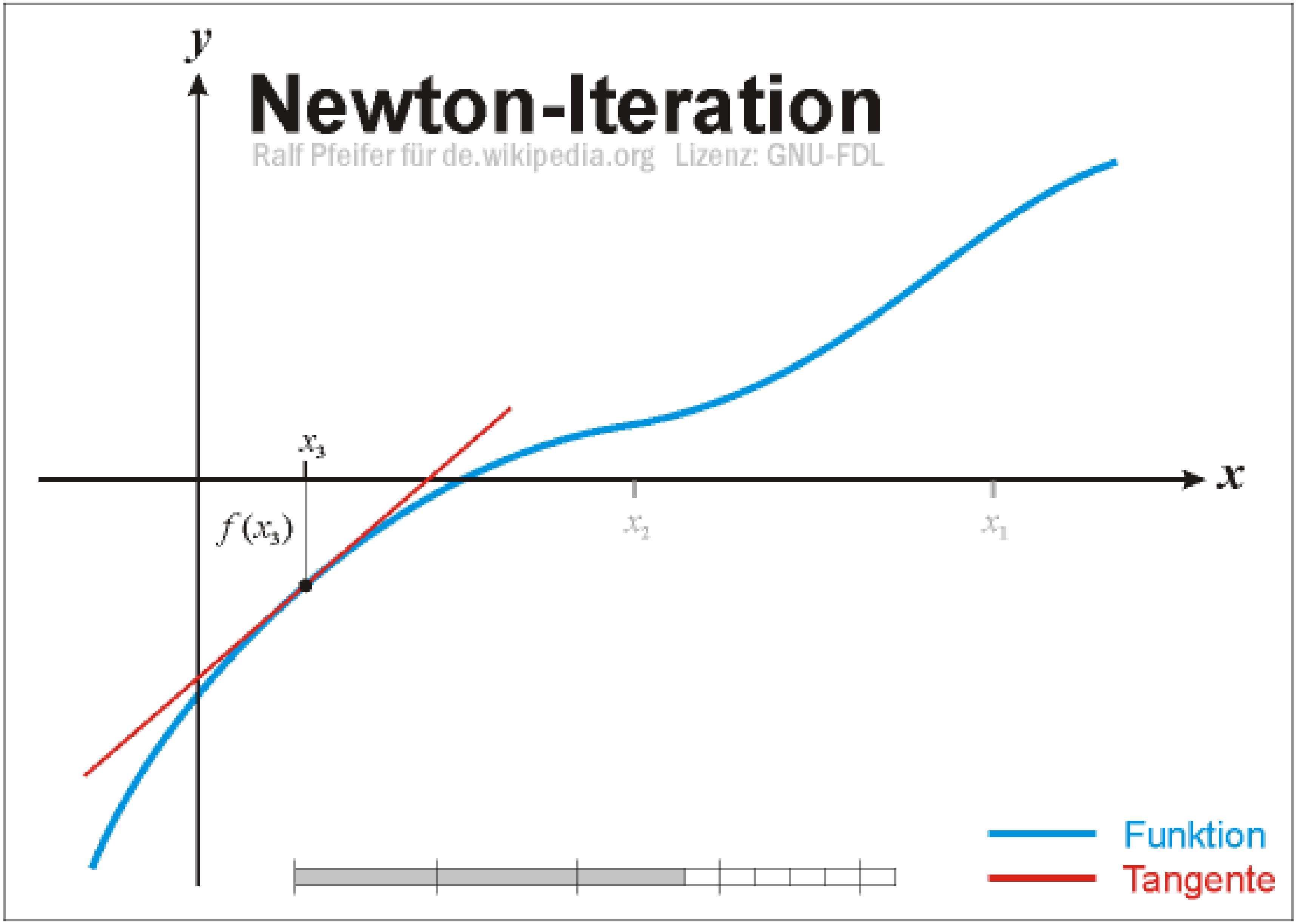
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



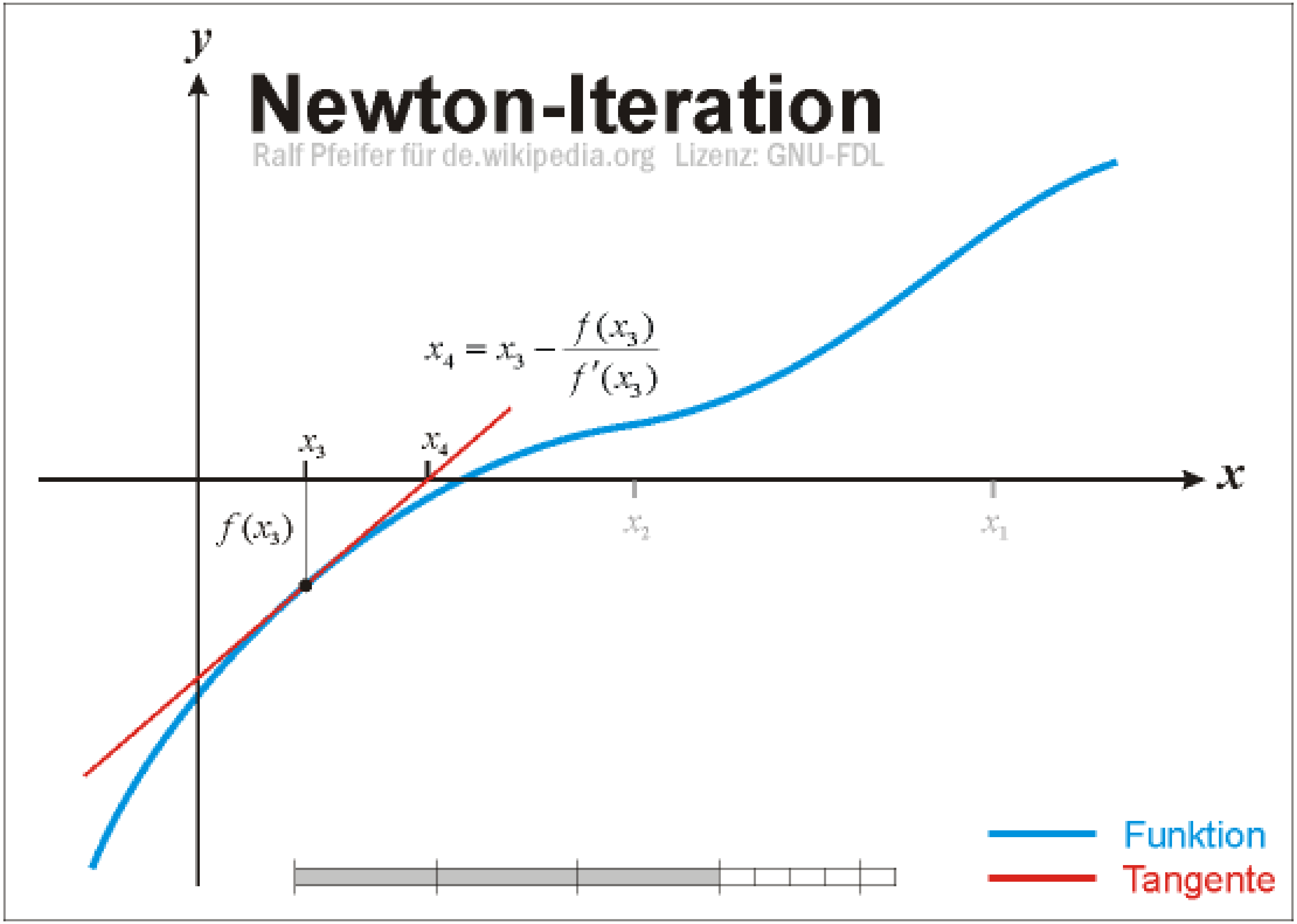
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



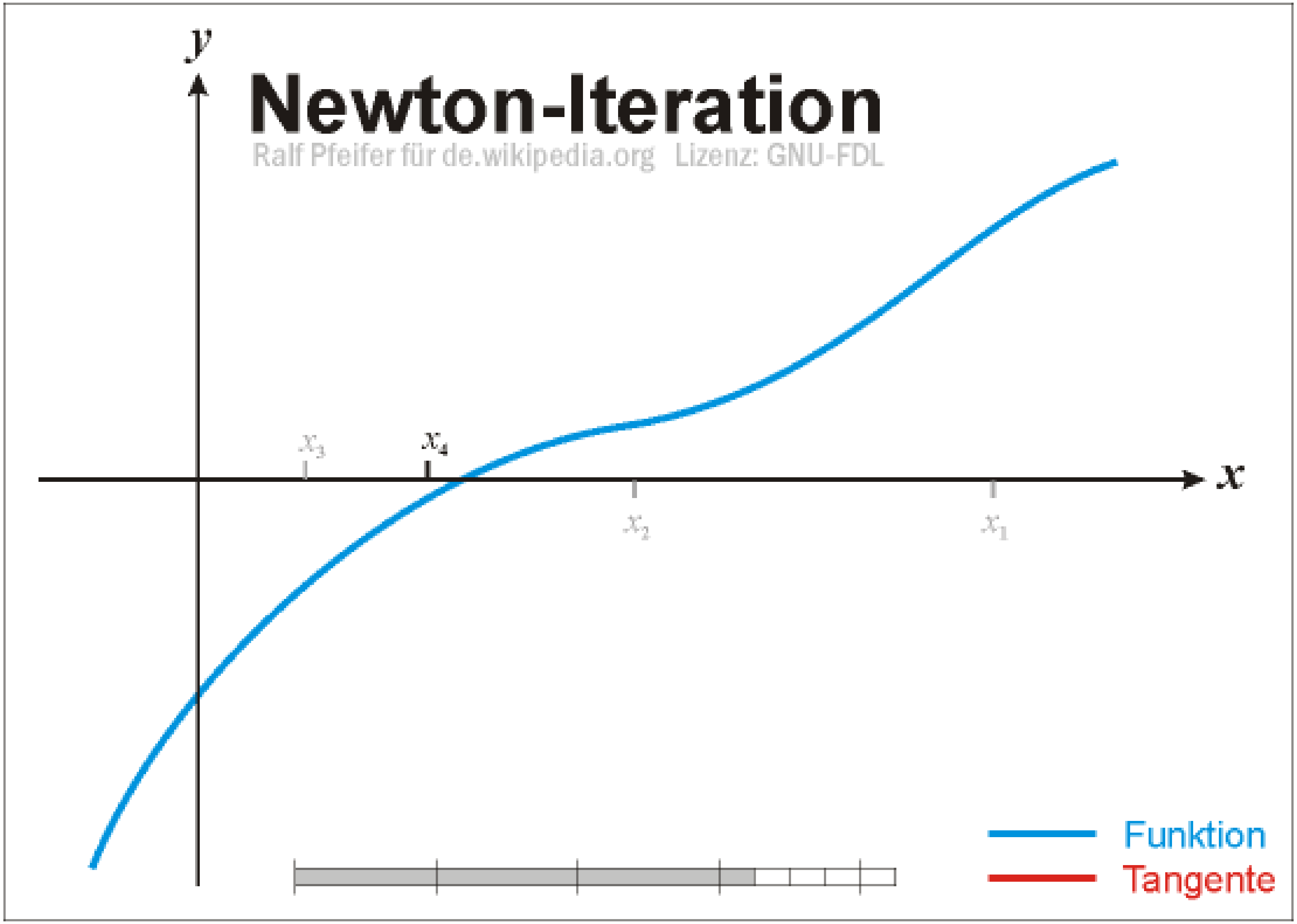
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



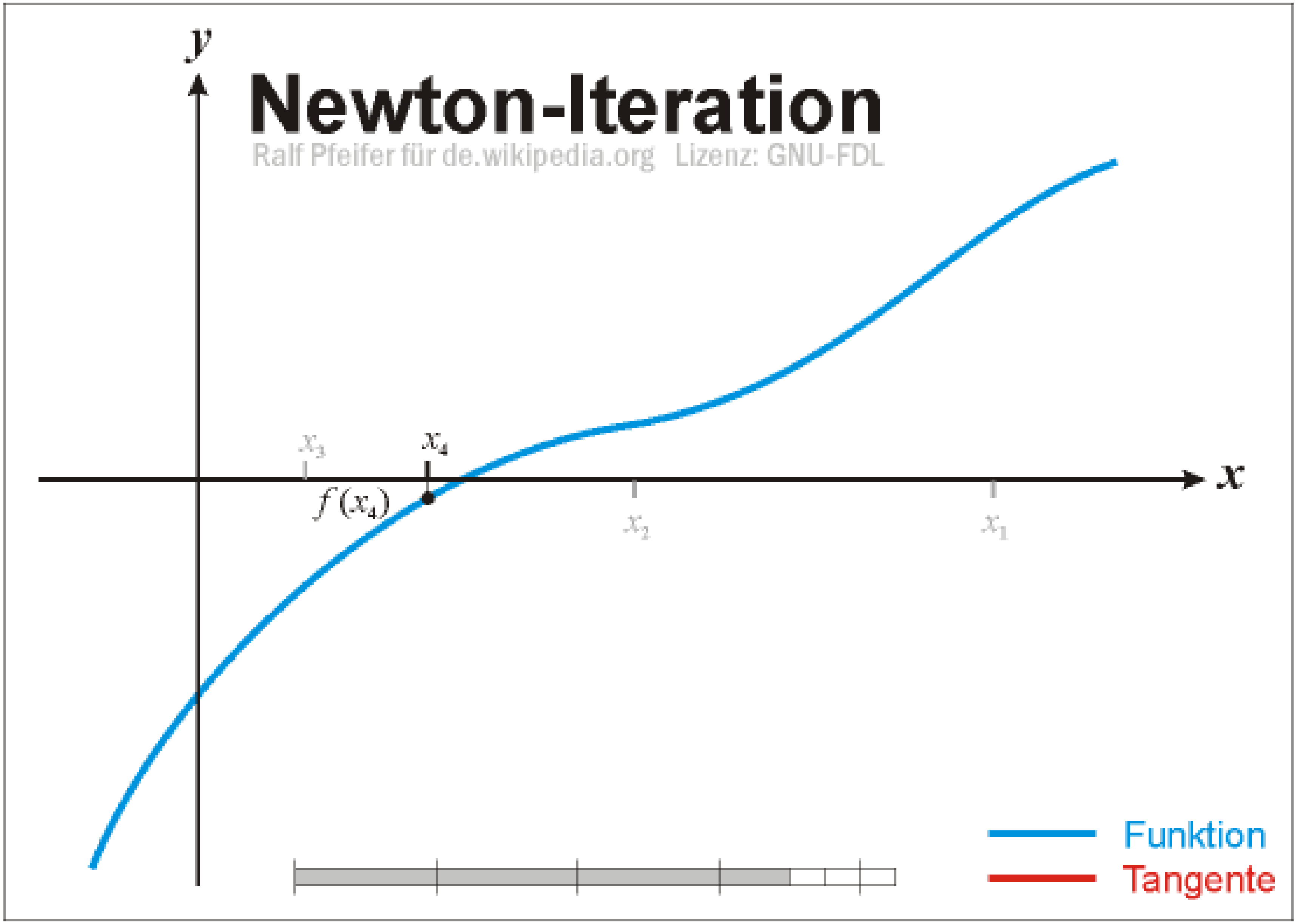
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



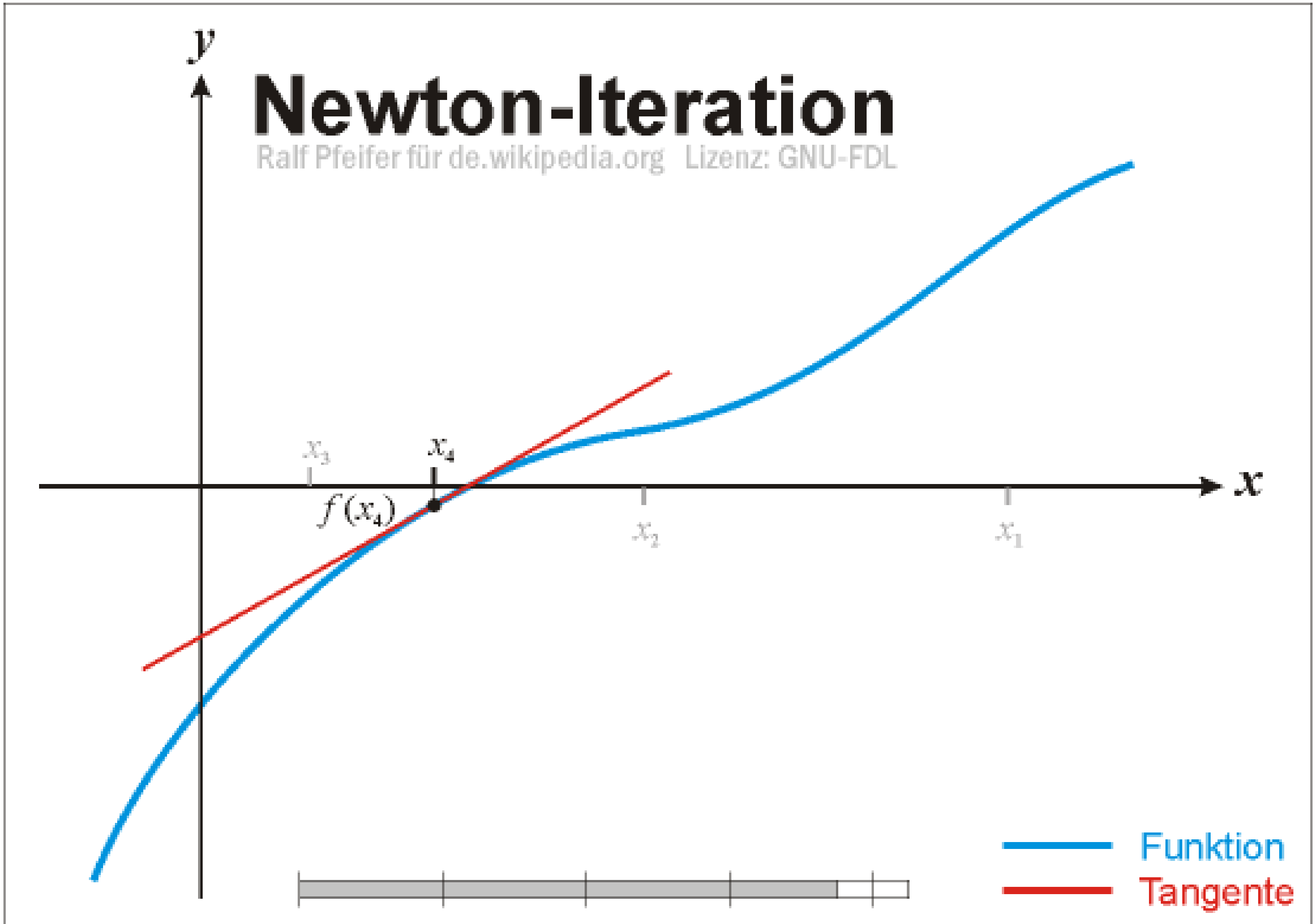
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



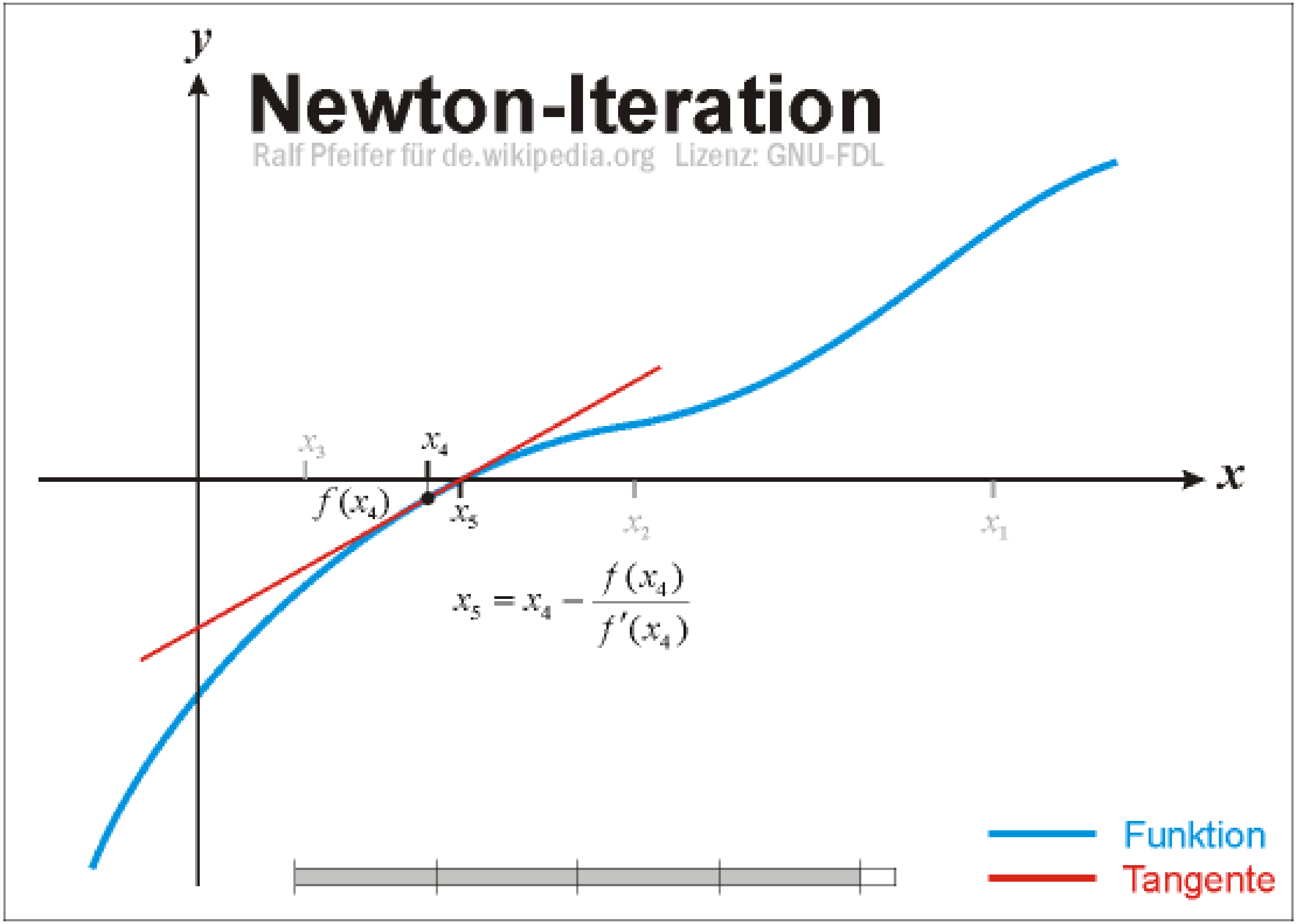
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



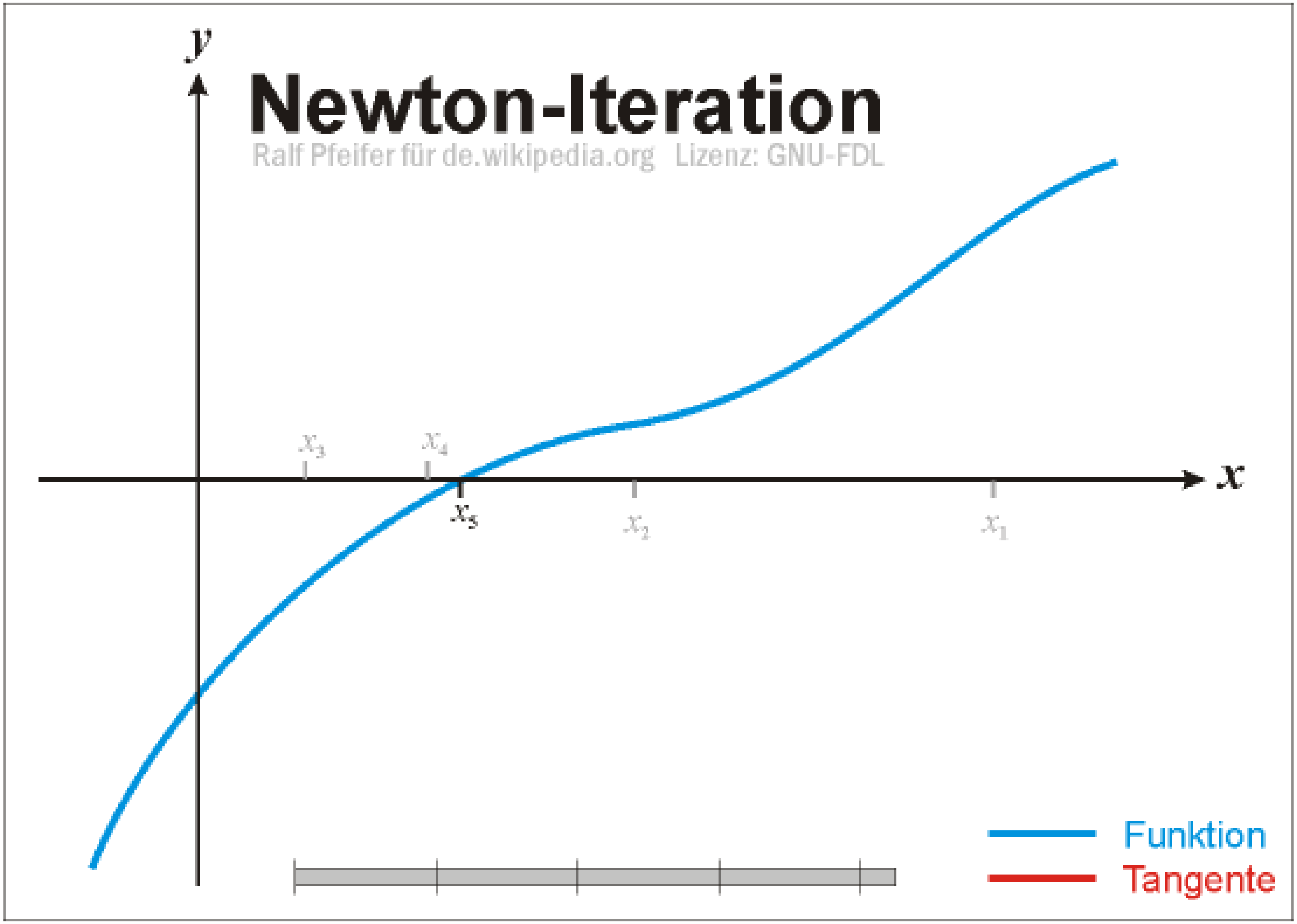
Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



Newtonova metoda pro řešení soustavy nelineárních rovnic

Příklad: Hledání kořenu funkce jedné proměnné vede na iteraci: $x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$



Newtonova metoda pro minimalizaci funkce

- Cíl: chceme nalézt lokální minimum funkce $f\colon \mathbb{R}^n \to \mathbb{R}$

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$

Newtonova metoda pro minimalizaci funkce

- Cíl: chceme nalézt lokální minimum funkce $f\colon \mathbb{R}^n \to \mathbb{R}$

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$

- Položíme $\mathbf{g}(\mathbf{x}) = f'(\mathbf{x})^T$ a pomocí Newtonovy metody řešíme $\mathbf{g}(\mathbf{x}) = \mathbf{0}$, tedy iterujeme

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{g}'(\mathbf{x}_k)^{-1}\mathbf{g}(\mathbf{x}_k) \qquad \text{neboli} \qquad \mathbf{x}_{k+1} = \mathbf{x}_k - f''(\mathbf{x}_k)^{-1} f'(\mathbf{x})^T$$

Newtonova metoda pro minimalizaci funkce

- Cíl: chceme nalézt lokální minimum funkce $f\colon \mathbb{R}^n \to \mathbb{R}$

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$

- Položíme $\mathbf{g}(\mathbf{x}) = f'(\mathbf{x})^T$ a pomocí Newtonovy metody řešíme $\mathbf{g}(\mathbf{x}) = \mathbf{0}$, tedy iterujeme

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{g}'(\mathbf{x}_k)^{-1}\mathbf{g}(\mathbf{x}_k) \qquad \text{neboli} \qquad \mathbf{x}_{k+1} = \mathbf{x}_k - f''(\mathbf{x}_k)^{-1} f'(\mathbf{x})^T$$

to odpovídá sestupné metodě

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k \qquad \text{kde} \qquad \mathbf{s}_k = -f''(\mathbf{x}_k)^{-1} f'(\mathbf{x})^T$$

Newtonova metoda pro minimalizaci funkce

- Cíl: chceme nalézt lokální minimum funkce $f\colon \mathbb{R}^n \to \mathbb{R}$

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$

- Položíme $\mathbf{g}(\mathbf{x}) = f'(\mathbf{x})^T$ a pomocí Newtonovy metody řešíme $\mathbf{g}(\mathbf{x}) = \mathbf{0}$, tedy iterujeme

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{g}'(\mathbf{x}_k)^{-1}\mathbf{g}(\mathbf{x}_k) \qquad \text{neboli} \qquad \mathbf{x}_{k+1} = \mathbf{x}_k - f''(\mathbf{x}_k)^{-1} f'(\mathbf{x})^T$$

to odpovídá sestupné metodě

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k \qquad \text{kde} \qquad \mathbf{s}_k = -f''(\mathbf{x}_k)^{-1} f'(\mathbf{x})^T$$

- Pokud $\alpha_k = 1$ jedná se o čistou Newtonovu metodu.

Newtonova metoda pro minimalizaci funkce

- Cíl: chceme nalézt lokální minimum funkce $f\colon \mathbb{R}^n \to \mathbb{R}$

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$

- Položíme $\mathbf{g}(\mathbf{x}) = f'(\mathbf{x})^T$ a pomocí Newtonovy metody řešíme $\mathbf{g}(\mathbf{x}) = \mathbf{0}$, tedy iterujeme

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{g}'(\mathbf{x}_k)^{-1}\mathbf{g}(\mathbf{x}_k) \qquad \text{neboli} \qquad \mathbf{x}_{k+1} = \mathbf{x}_k - f''(\mathbf{x}_k)^{-1} f'(\mathbf{x})^T$$

to odpovídá sestupné metodě

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k \qquad \text{kde} \qquad \mathbf{s}_k = -f''(\mathbf{x}_k)^{-1} f'(\mathbf{x})^T$$

- Pokud $\alpha_k = 1$ jedná se o čistou Newtonovu metodu.
- Postačující podmínkou pro sestupný směr, je aby Hessian $f''(\mathbf{x}_k)$ byl pozitivně definitní

$$f'(\mathbf{x}_k)\mathbf{s}_k = -f'(\mathbf{x}_k) f''(\mathbf{x}_k)^{-1} f'(\mathbf{x})^T < 0$$

Newtonova metoda pro minimalizaci funkce

- Newtonovu metodu pro minimalizaci funkce $f\colon \mathbb{R}^n \to \mathbb{R}$ lze odvodit i jinak.

Newtonova metoda pro minimalizaci funkce

- Newtonovu metodu pro minimalizaci funkce $f\colon \mathbb{R}^n \to \mathbb{R}$ lze odvodit i jinak.
- Aproximujeme funkci $f(\mathbf{x})$ kolem bodu $\mathbf{x}_k$ Taylorovým polynomem 2. řadu

$$f(\mathbf{x}) \approx \tilde{f}(\mathbf{x}) = f(\mathbf{x}_k) + f'(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_k)^T f''(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k)$$

Newtonova metoda pro minimalizaci funkce

- Newtonovu metodu pro minimalizaci funkce $f\colon \mathbb{R}^n \to \mathbb{R}$ lze odvodit i jinak.
- Aproximujeme funkci $f(\mathbf{x})$ kolem bodu $\mathbf{x}_k$ Taylorovým polynomem 2. řadu

$$f(\mathbf{x}) \approx \tilde{f}(\mathbf{x}) = f(\mathbf{x}_k) + f'(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_k)^T f''(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k)$$

- Novou iteraci $\mathbf{x}_{k+1}$ získáme minimalizací

$$\min_{\mathbf{x} \in \mathbb{R}^n} \tilde{f}(\mathbf{x})$$

Newtonova metoda pro minimalizaci funkce

- Newtonovu metodu pro minimalizaci funkce $f\colon \mathbb{R}^n \to \mathbb{R}$ lze odvodit i jinak.
- Aproximujeme funkci $f(\mathbf{x})$ kolem bodu $\mathbf{x}_k$ Taylorovým polynomem 2. řadu

$$f(\mathbf{x}) \approx \tilde{f}(\mathbf{x}) = f(\mathbf{x}_k) + f'(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_k)^T f''(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k)$$

- Novou iteraci $\mathbf{x}_{k+1}$ získáme minimalizací

$$\min_{\mathbf{x} \in \mathbb{R}^n} \tilde{f}(\mathbf{x})$$

což můžeme řešit jako soustavu lineáních rovnic

$$\tilde{f}'(\mathbf{x}_{k+1}) = \mathbf{0} \qquad \Rightarrow \qquad f'(\mathbf{x}_k)^T + f''(\mathbf{x}_k)(\mathbf{x}_{k+1} - \mathbf{x}_k) = \mathbf{0}$$

Newtonova metoda pro minimalizaci funkce

- Newtonovu metodu pro minimalizaci funkce $f\colon \mathbb{R}^n \to \mathbb{R}$ lze odvodit i jinak.
- Aproximujeme funkci $f(\mathbf{x})$ kolem bodu $\mathbf{x}_k$ Taylorovým polynomem 2. řadu

$$f(\mathbf{x}) \approx \tilde{f}(\mathbf{x}) = f(\mathbf{x}_k) + f'(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_k)^T f''(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k)$$

- Novou iteraci $\mathbf{x}_{k+1}$ získáme minimalizací

$$\min_{\mathbf{x} \in \mathbb{R}^n} \tilde{f}(\mathbf{x})$$

což můžeme řešit jako soustavu lineáních rovnic

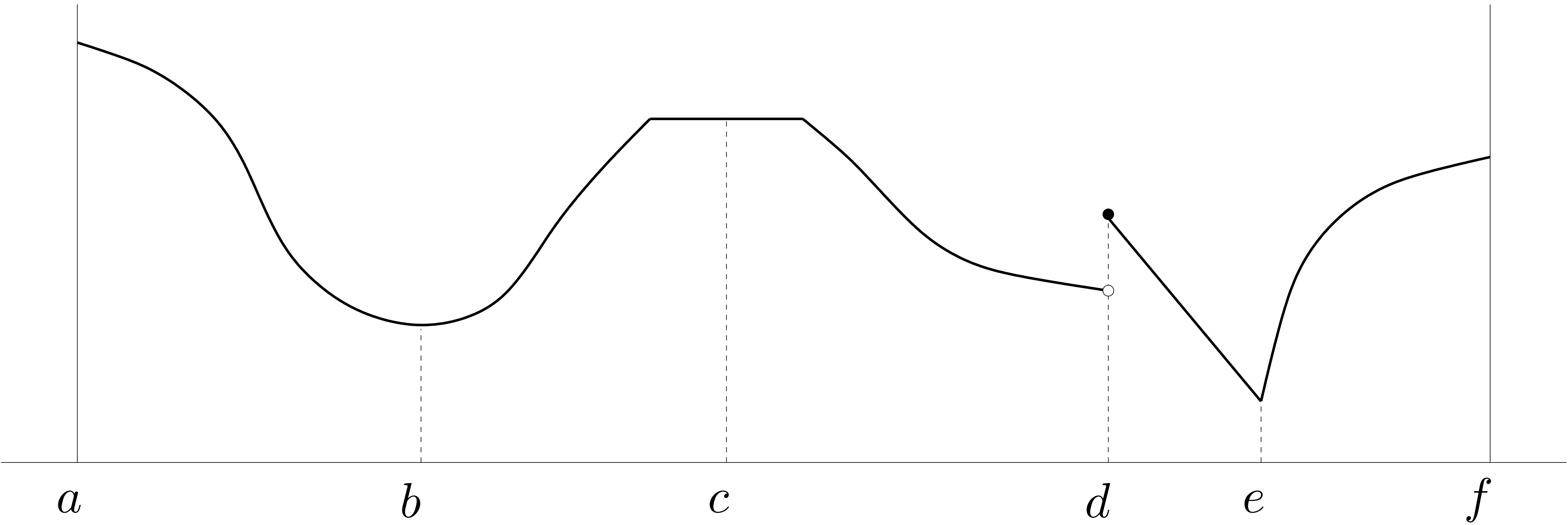
$$\tilde{f}'(\mathbf{x}_{k+1}) = \mathbf{0} \qquad \Rightarrow \qquad f'(\mathbf{x}_k)^T + f''(\mathbf{x}_k)(\mathbf{x}_{k+1} - \mathbf{x}_k) = \mathbf{0}$$

a pokud je Hessian $f''(\mathbf{x}_k)$ regulární dostaneme řešení

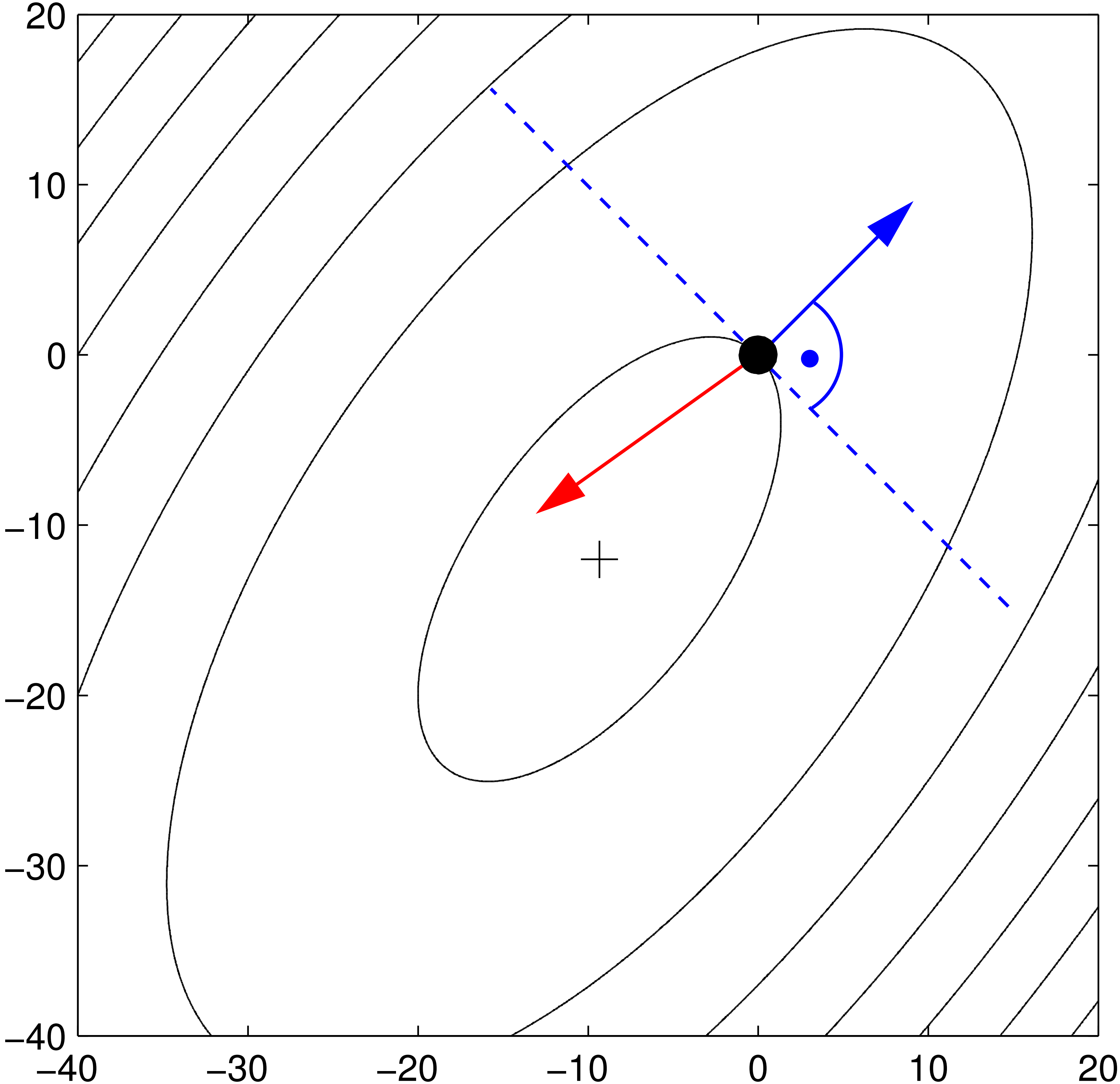
$$\mathbf{x}_{k+1} = \mathbf{x}_k - f''(\mathbf{x}_k)^{-1} f'(\mathbf{x}_k)^T$$

Konec

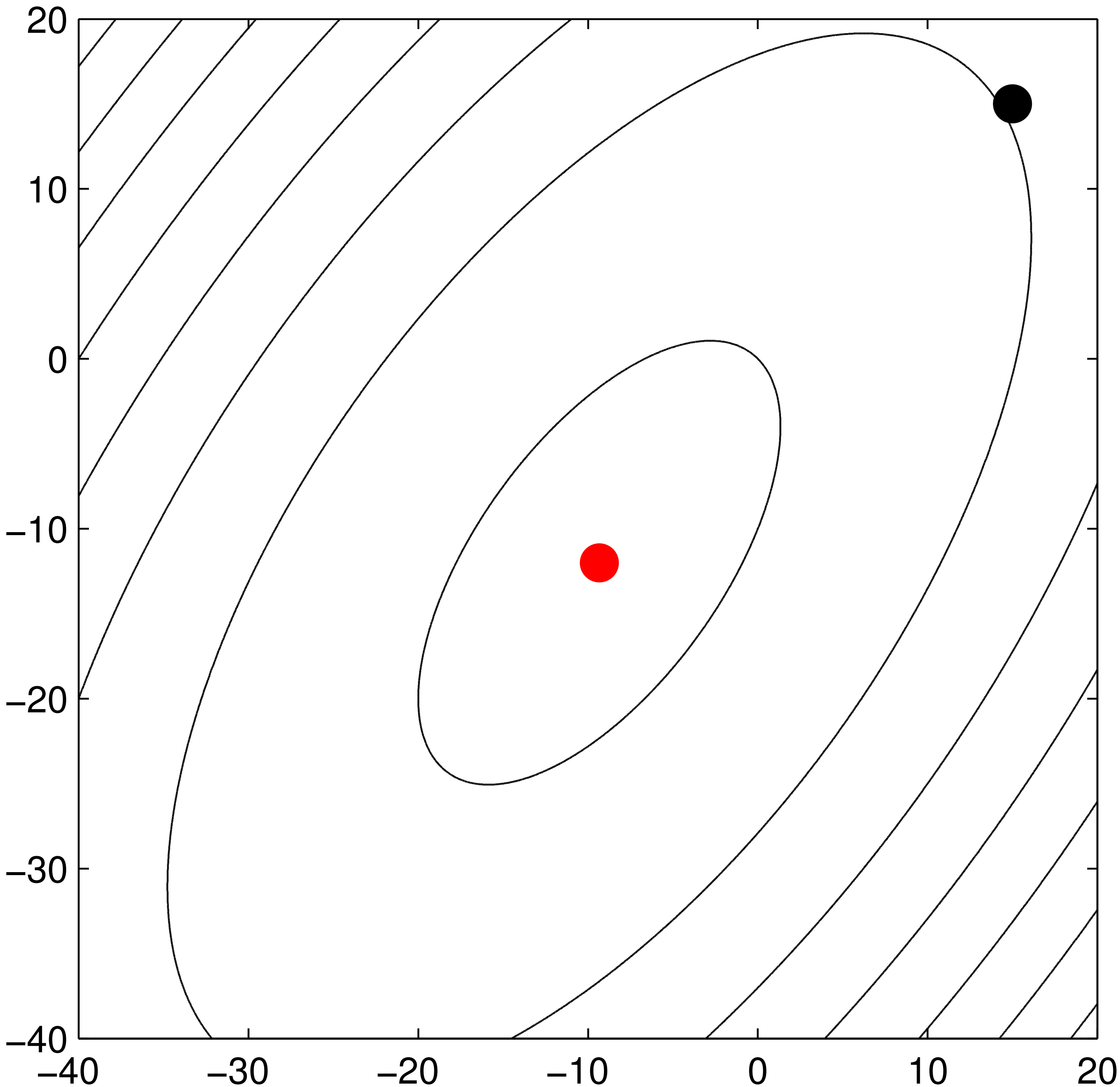
m p



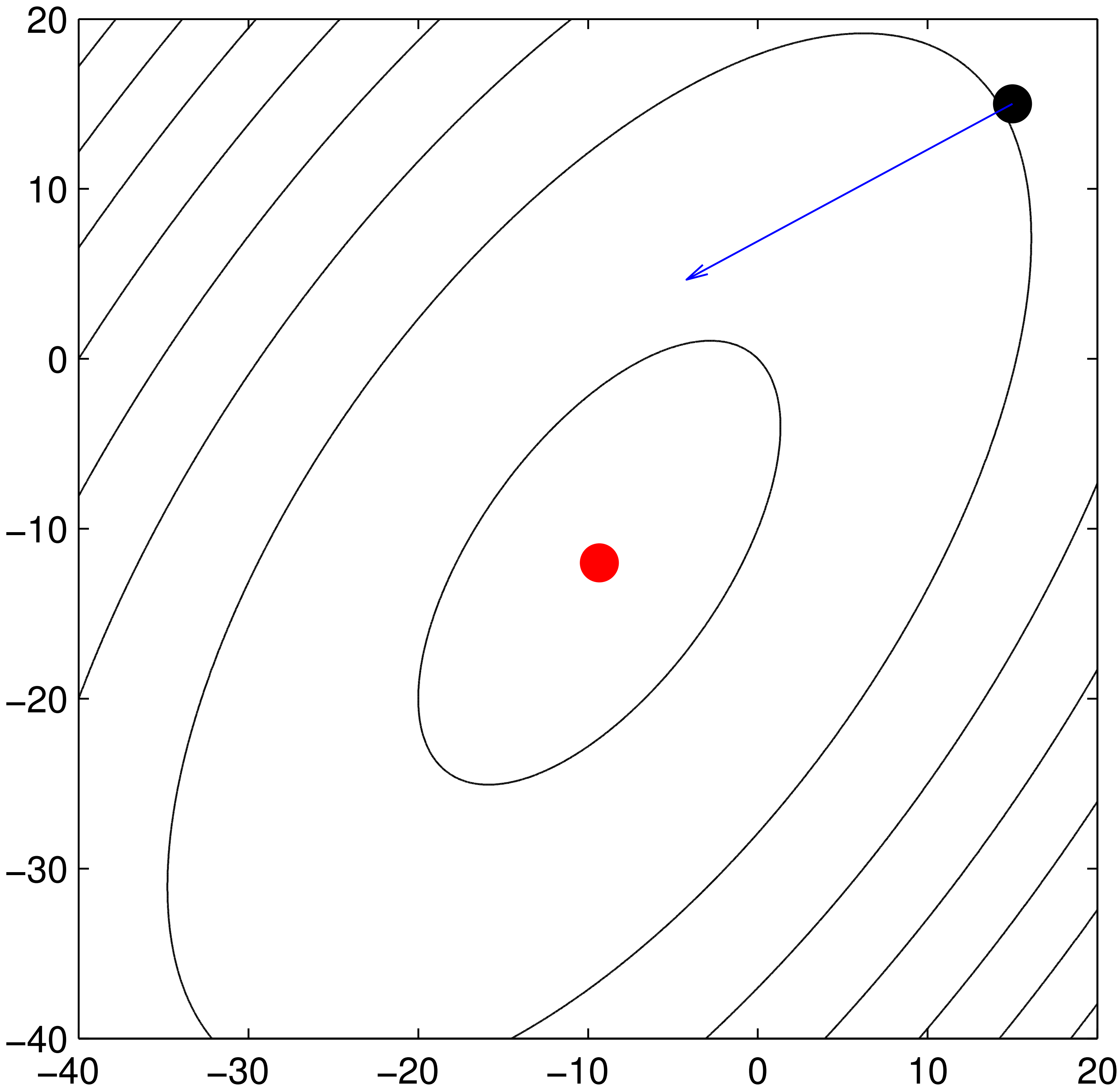
a
b
c
d
e
f

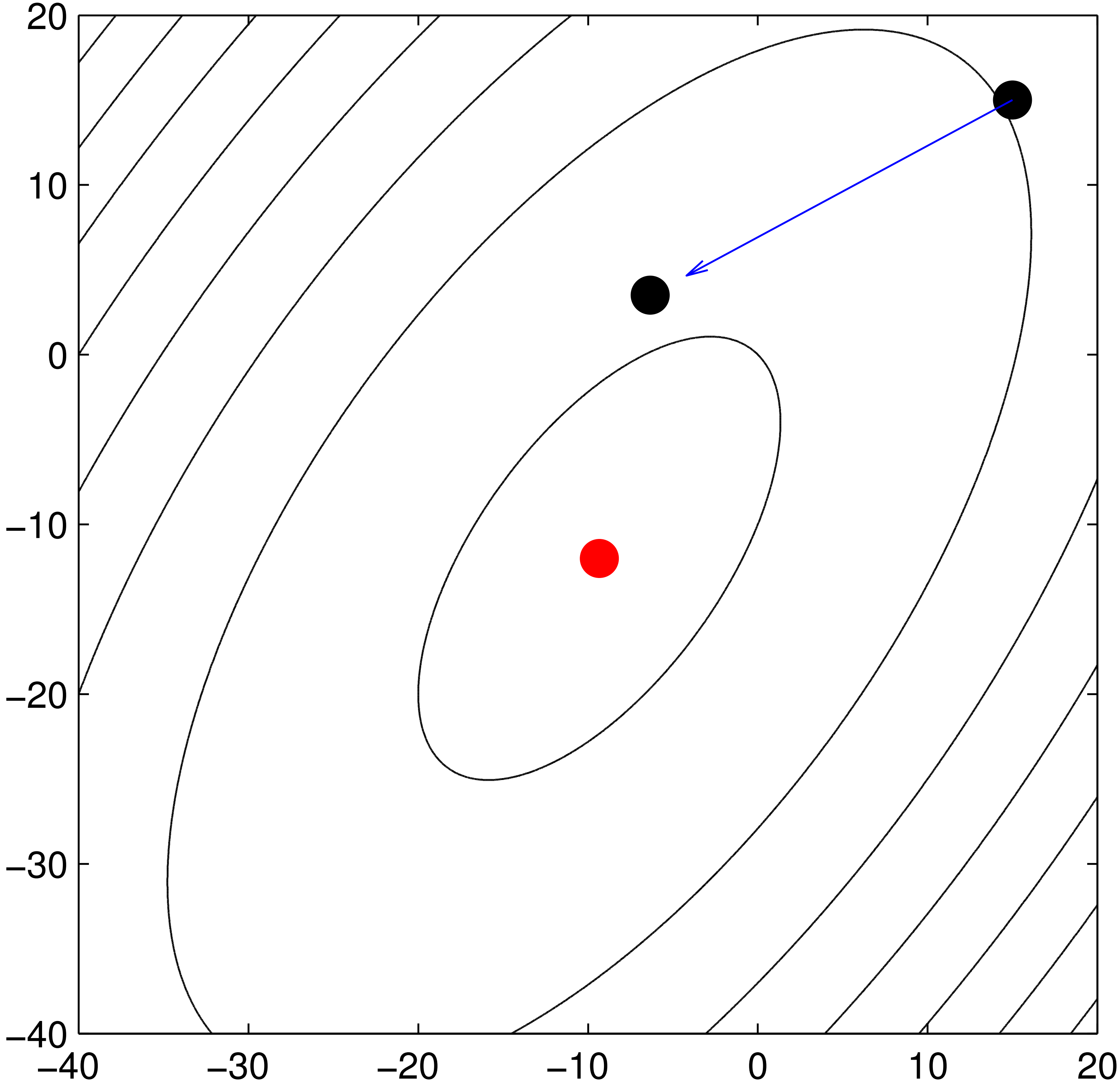


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

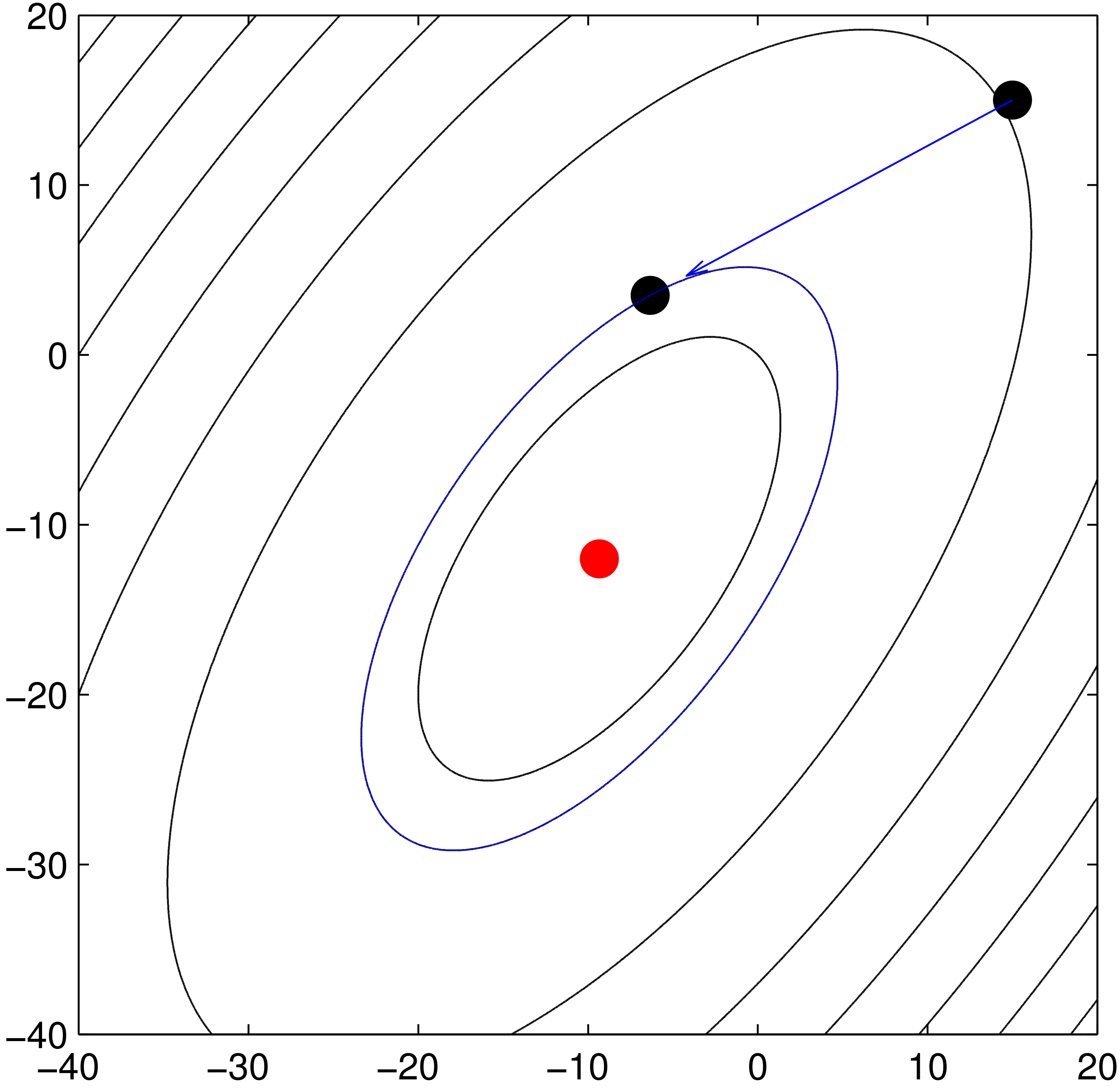


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

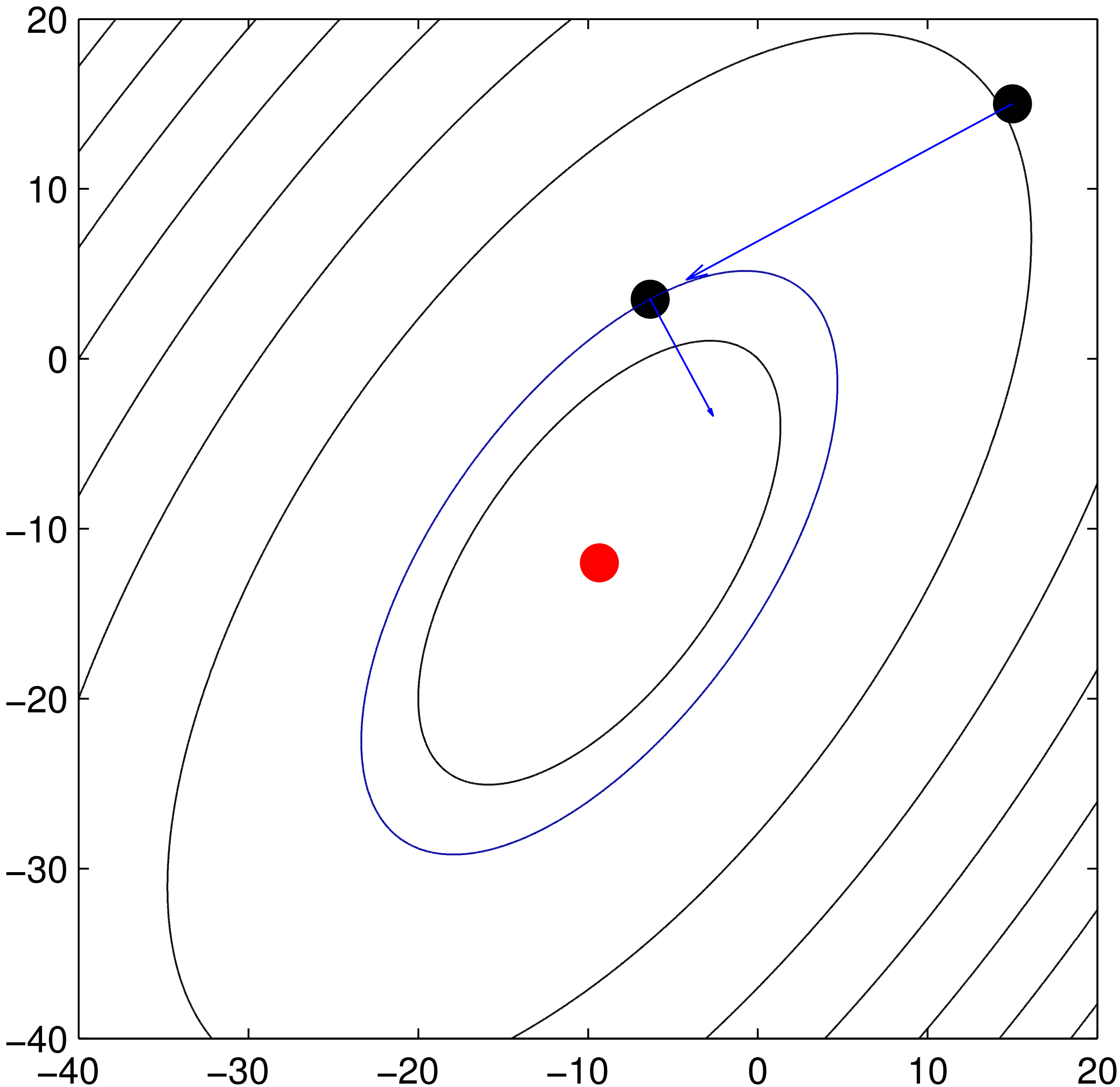

20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

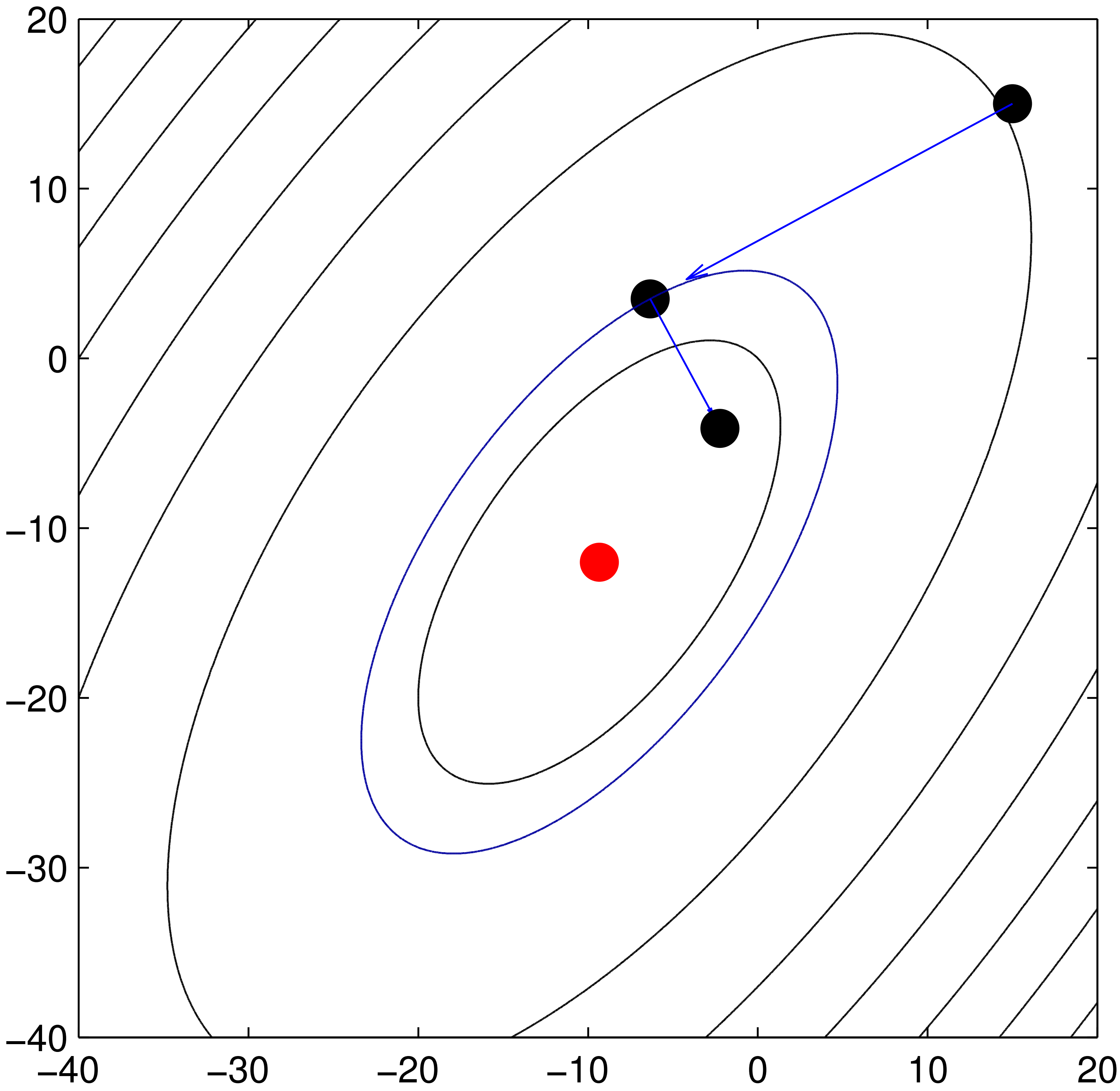


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

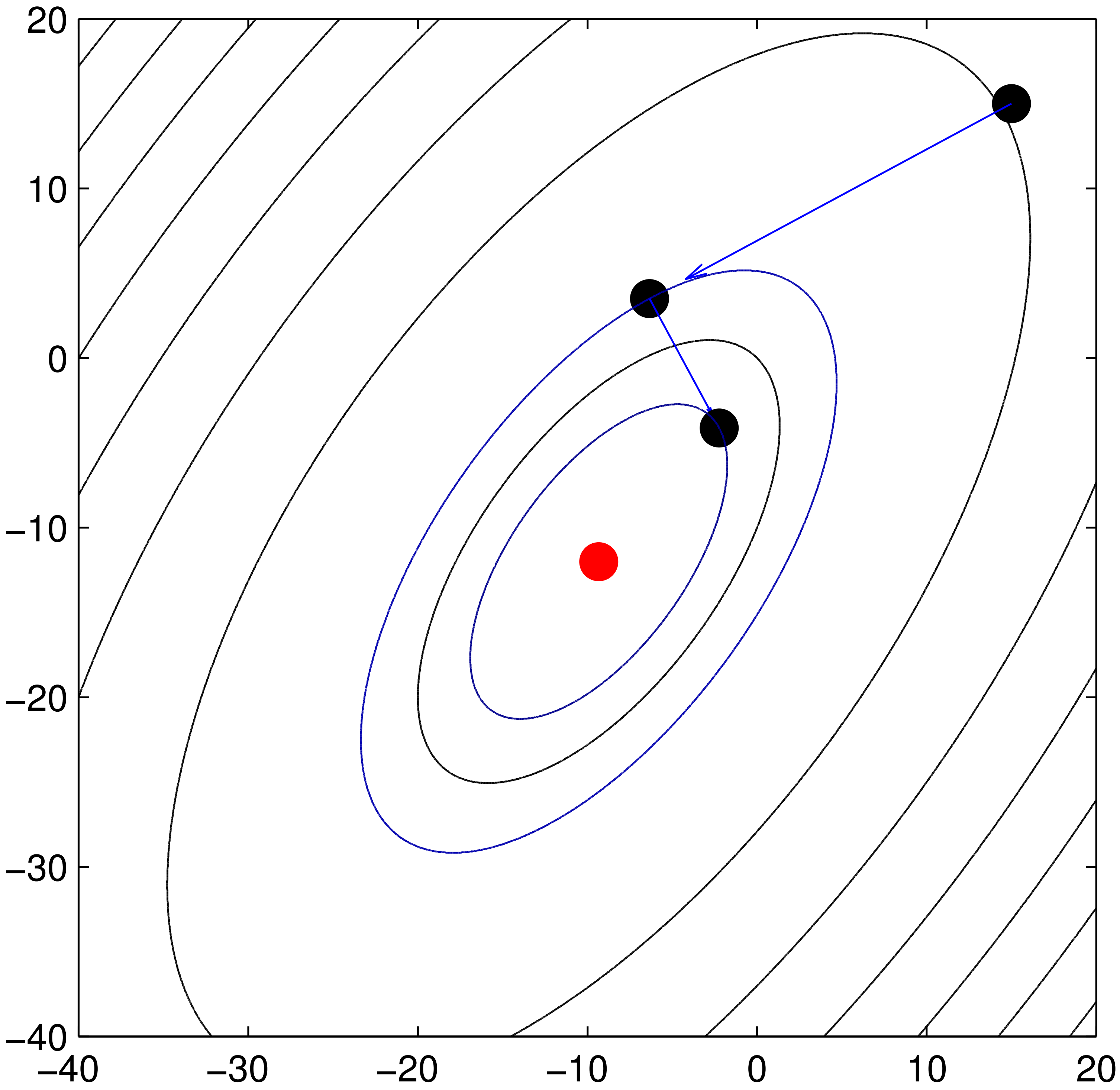


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

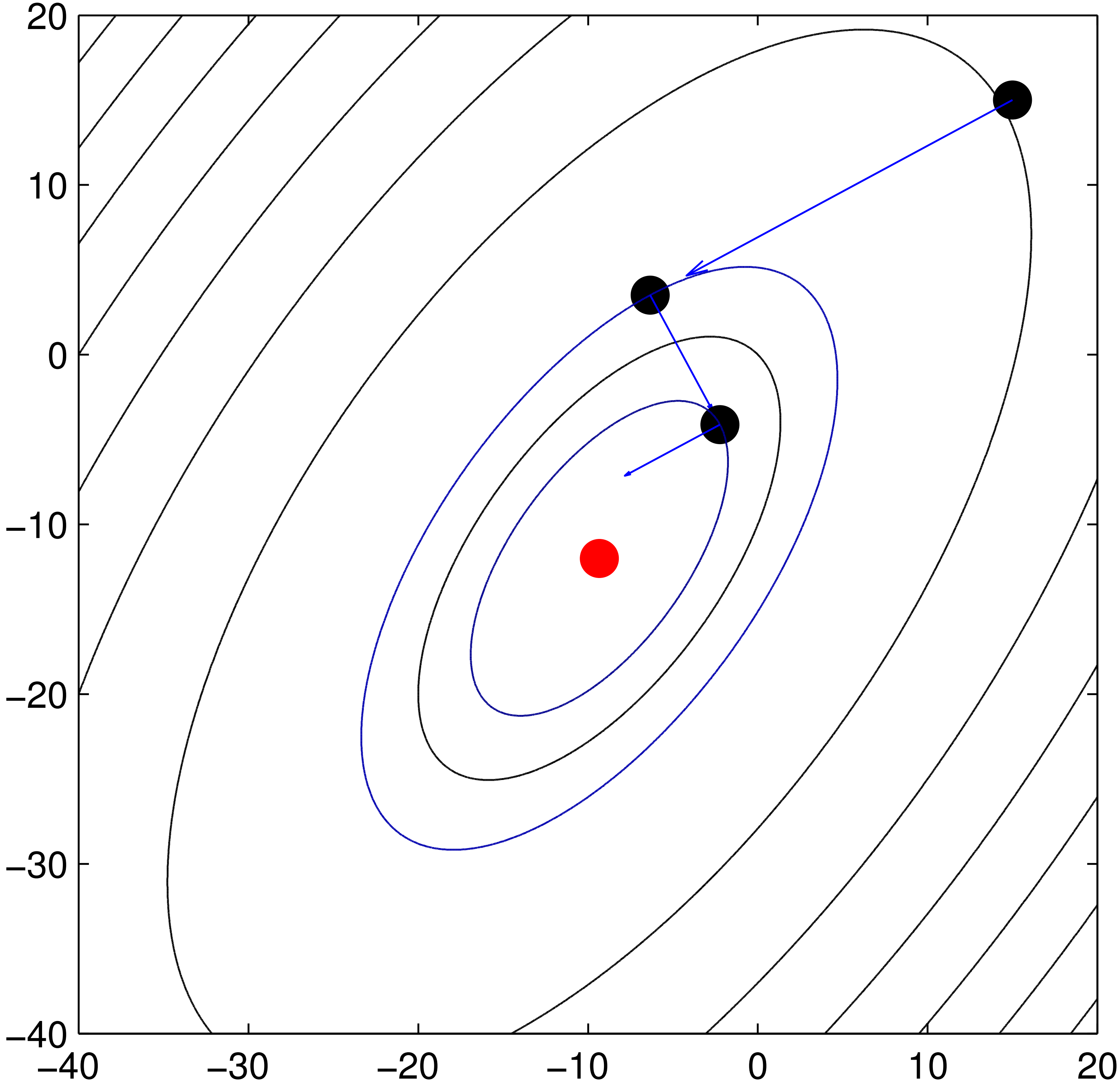

20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

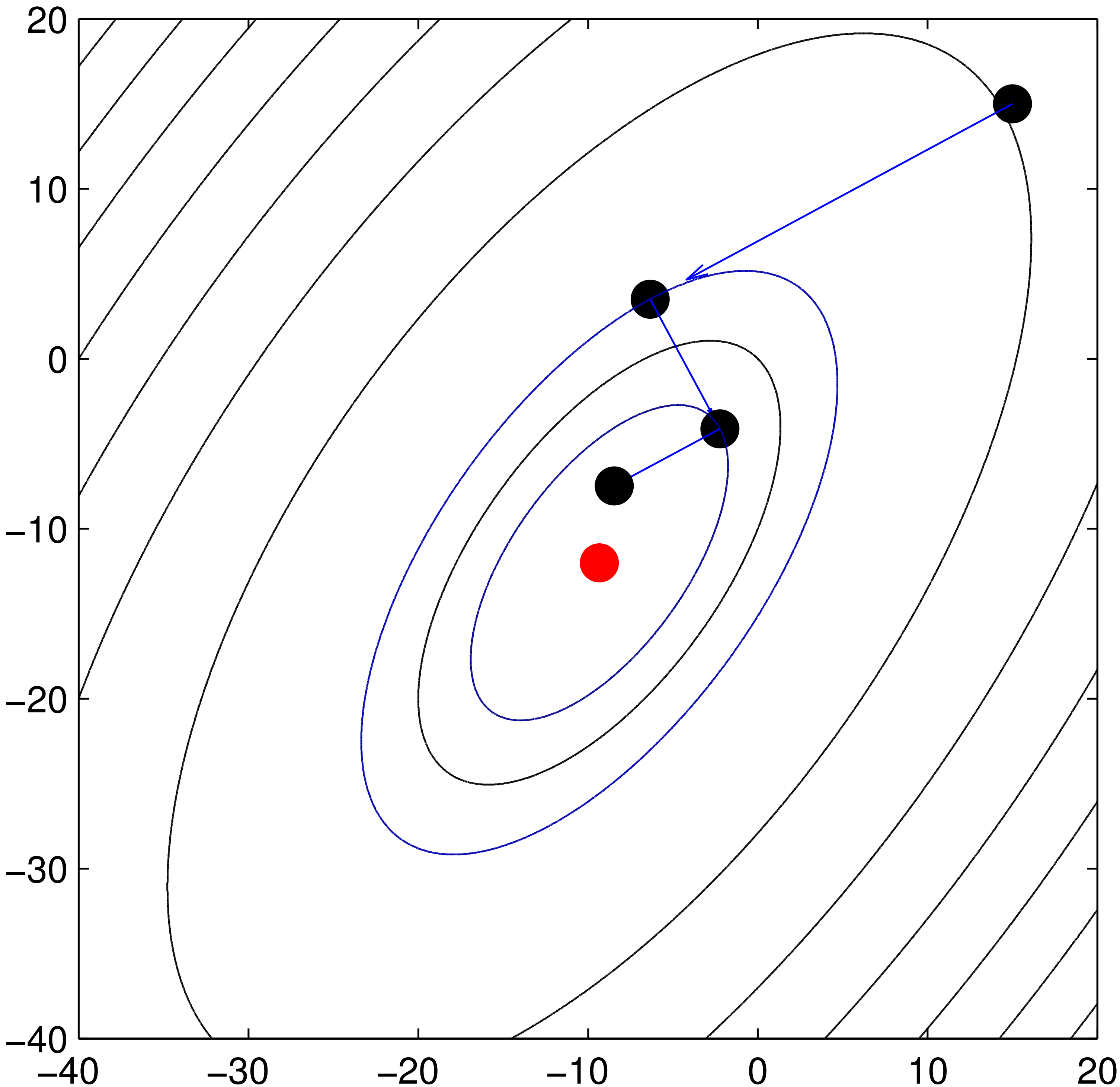


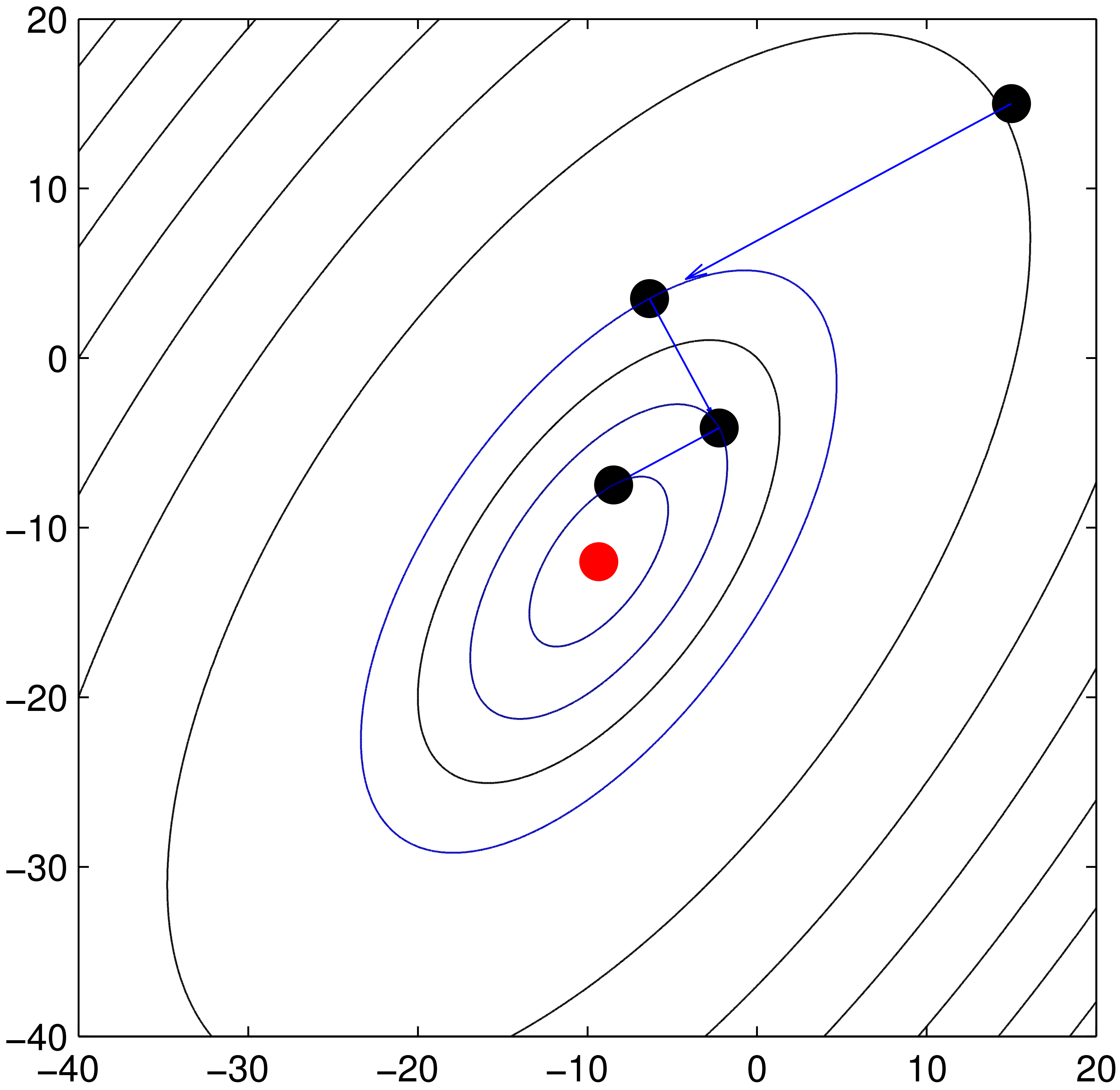
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

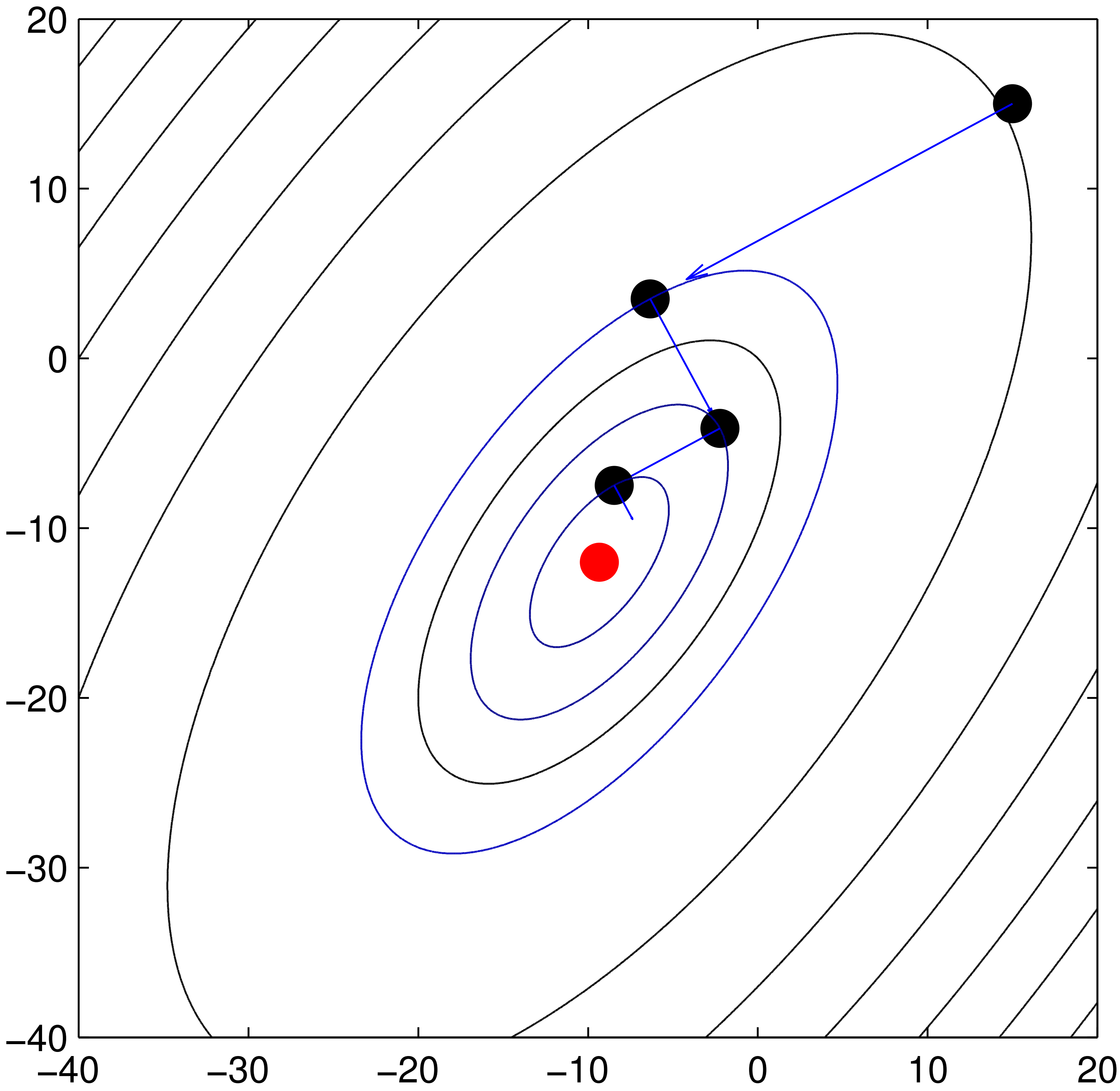


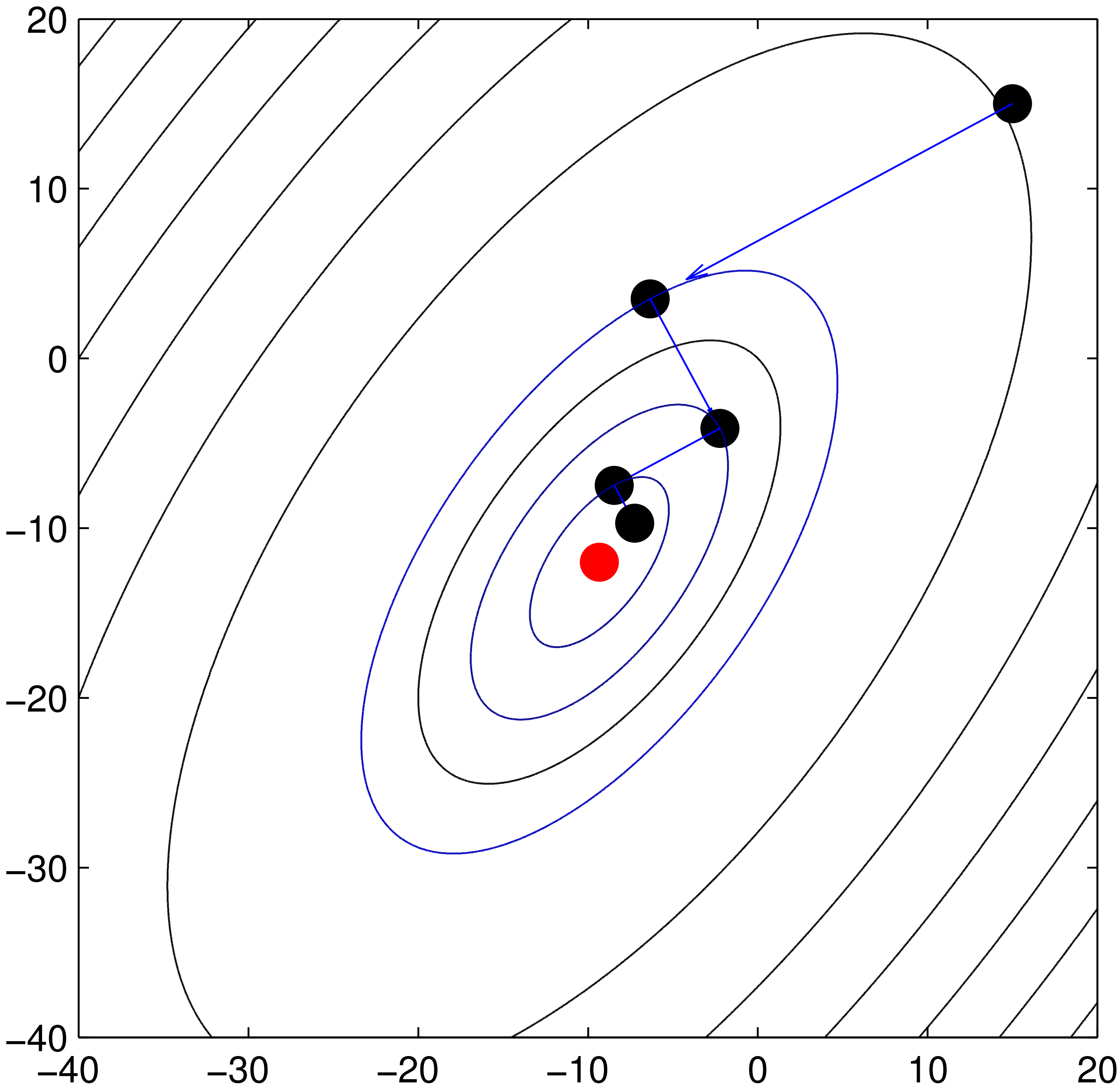
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

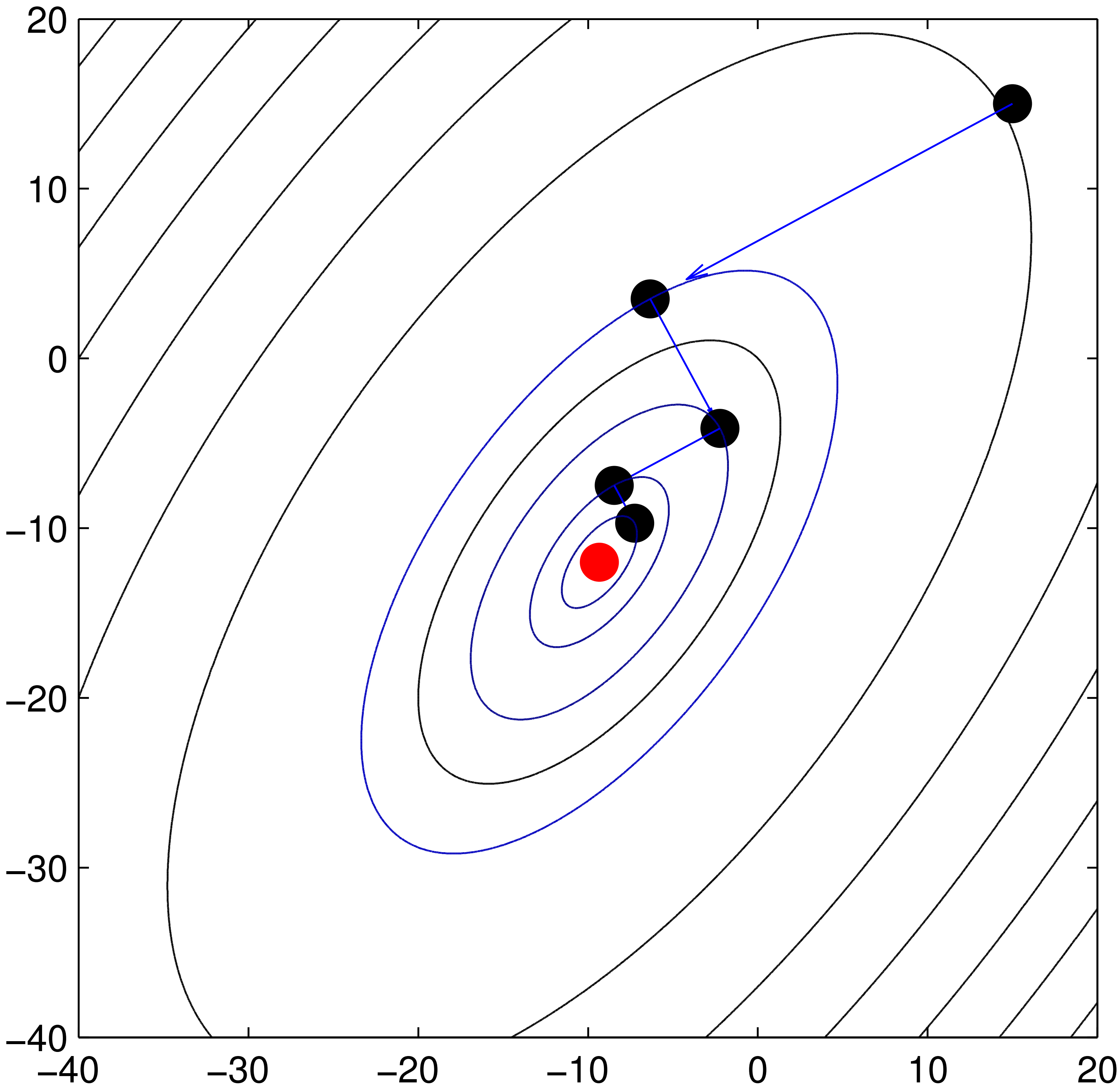

20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

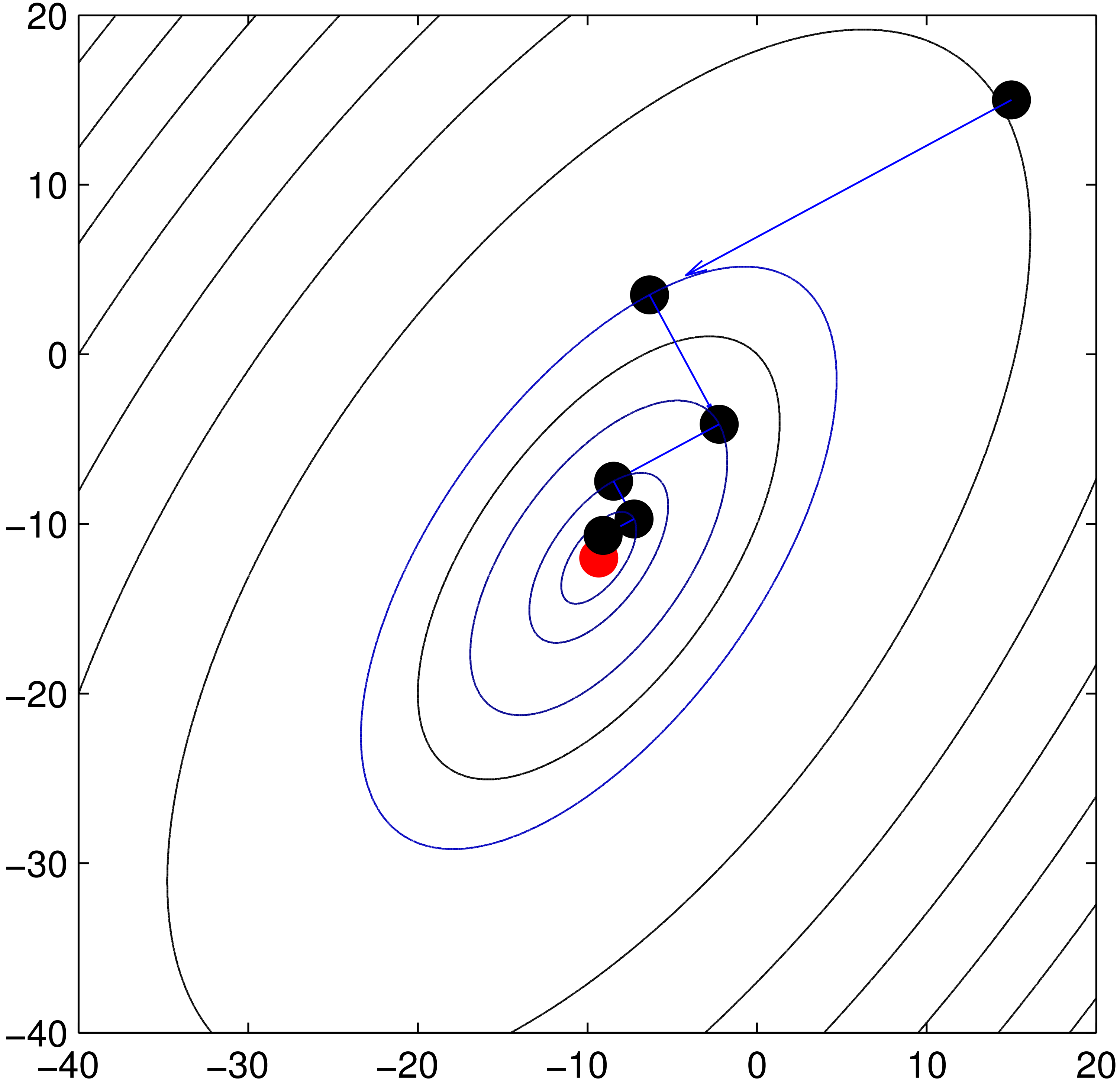

20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

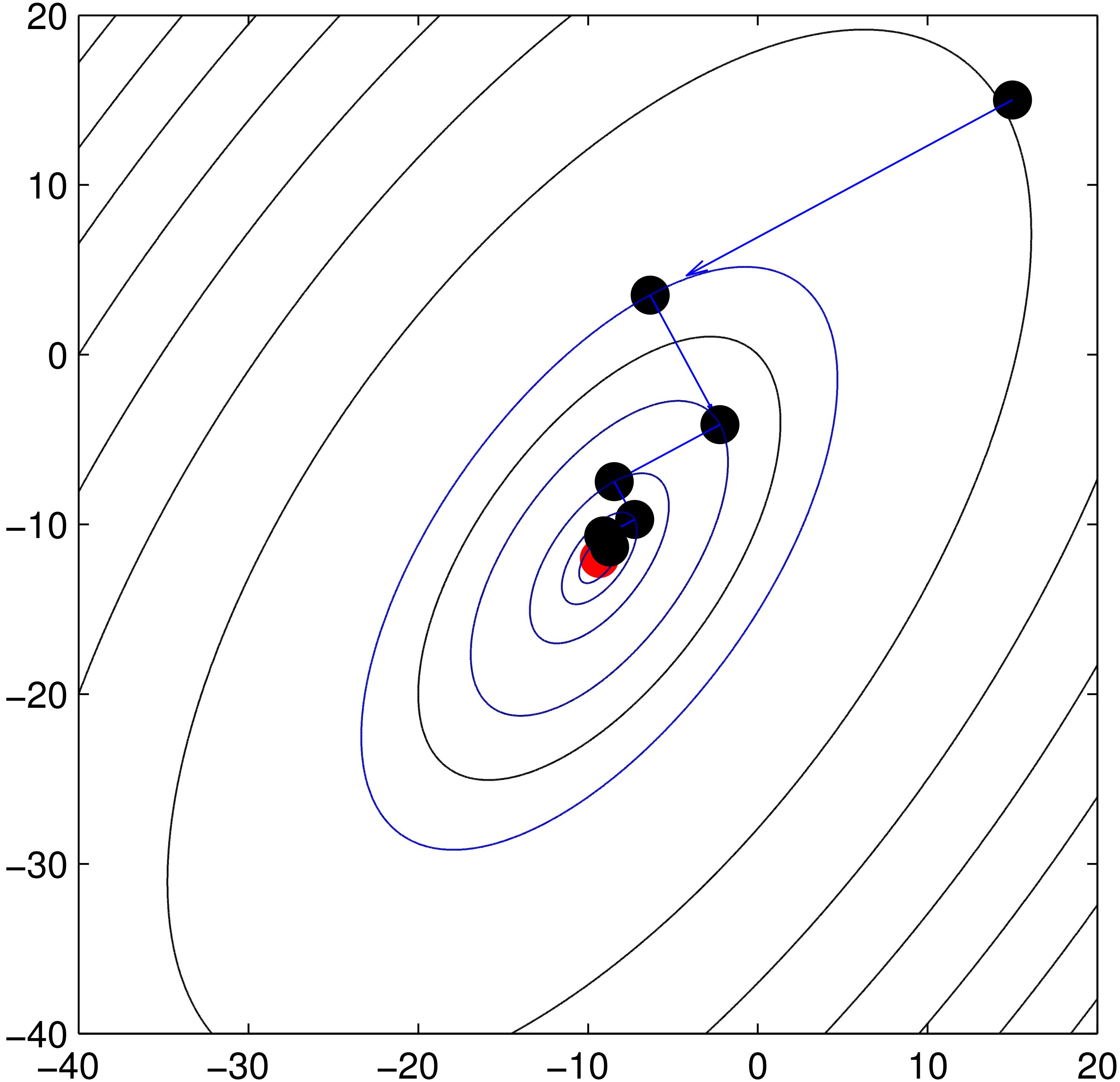

20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20



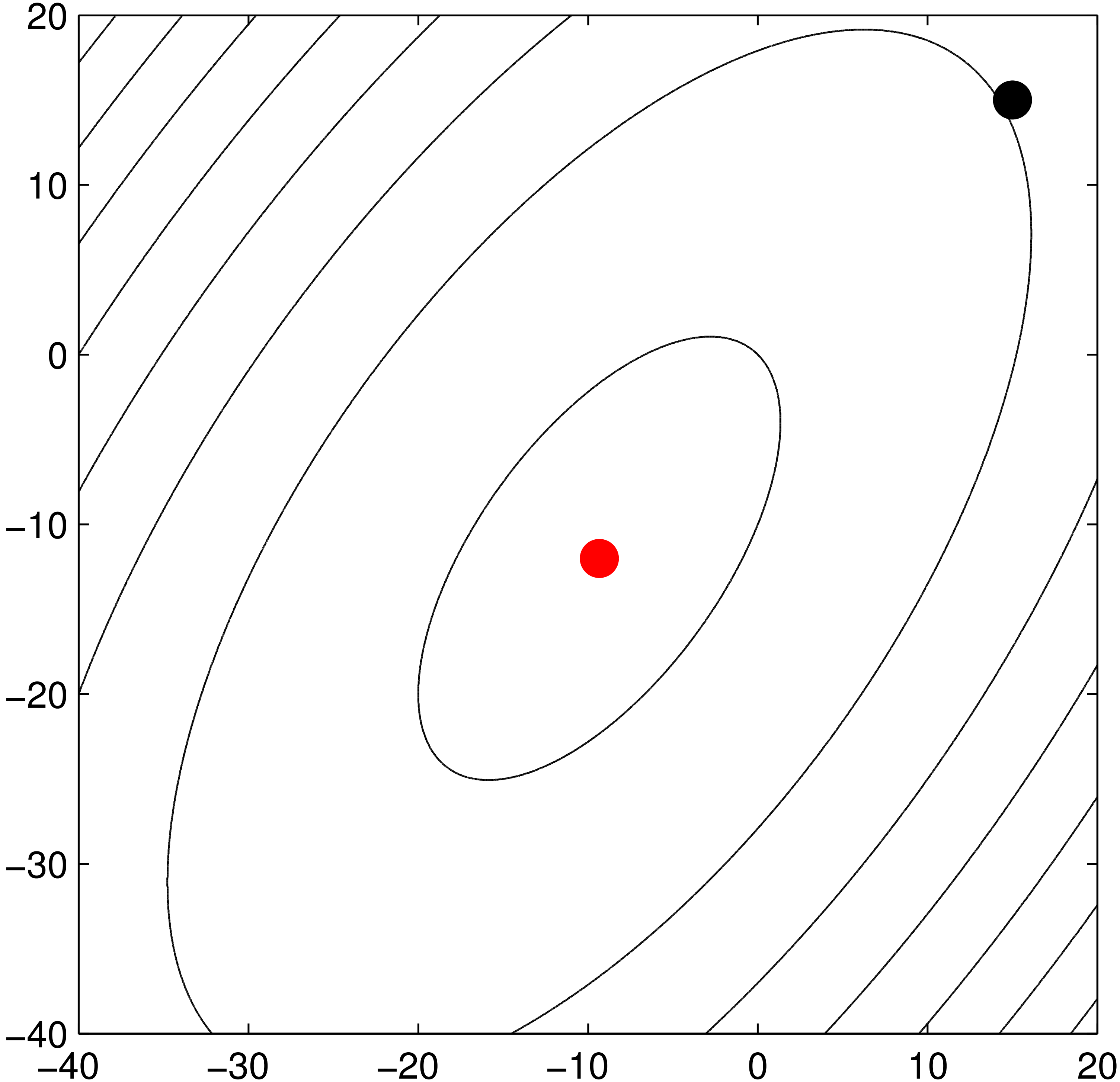
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20



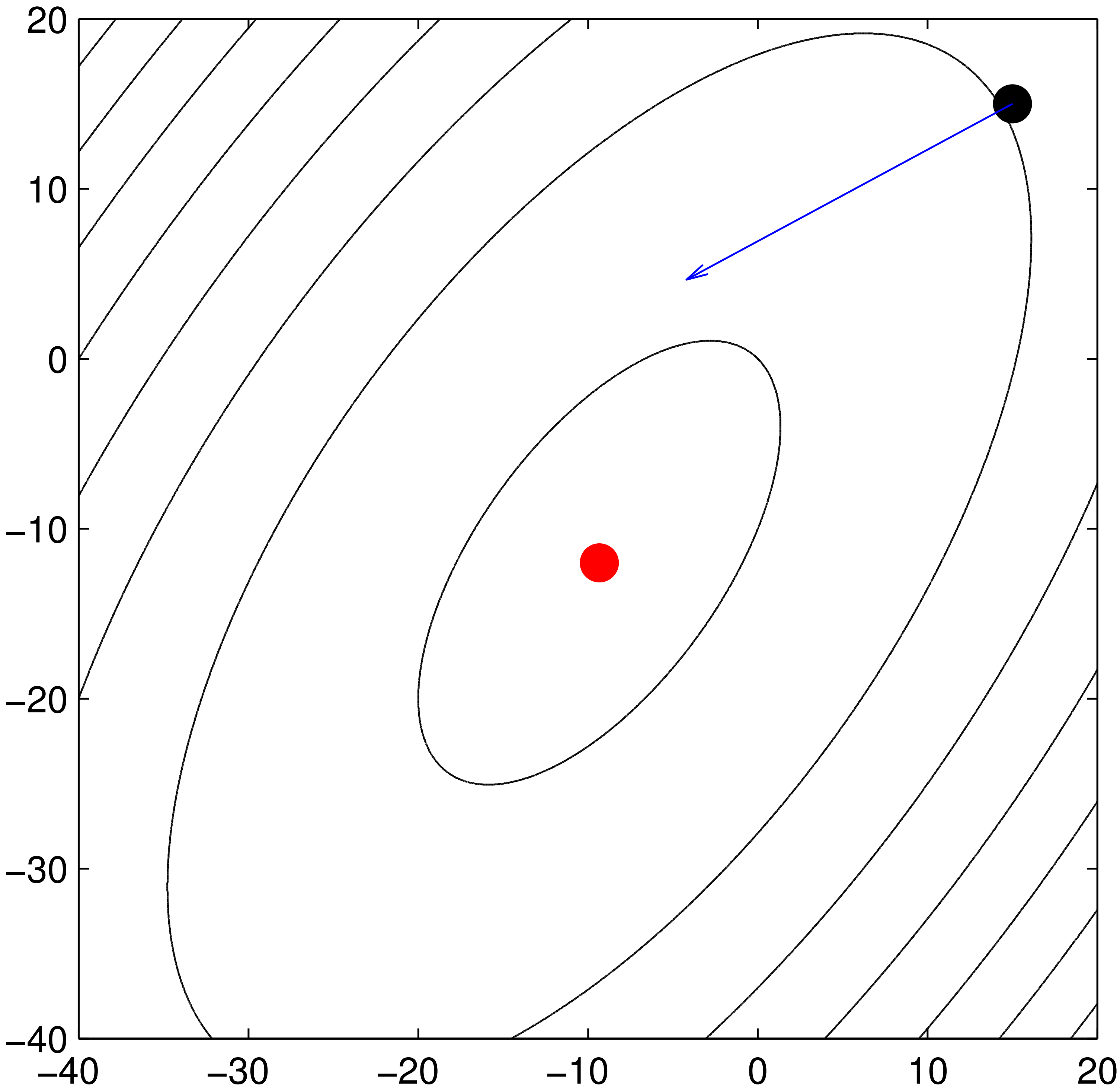
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

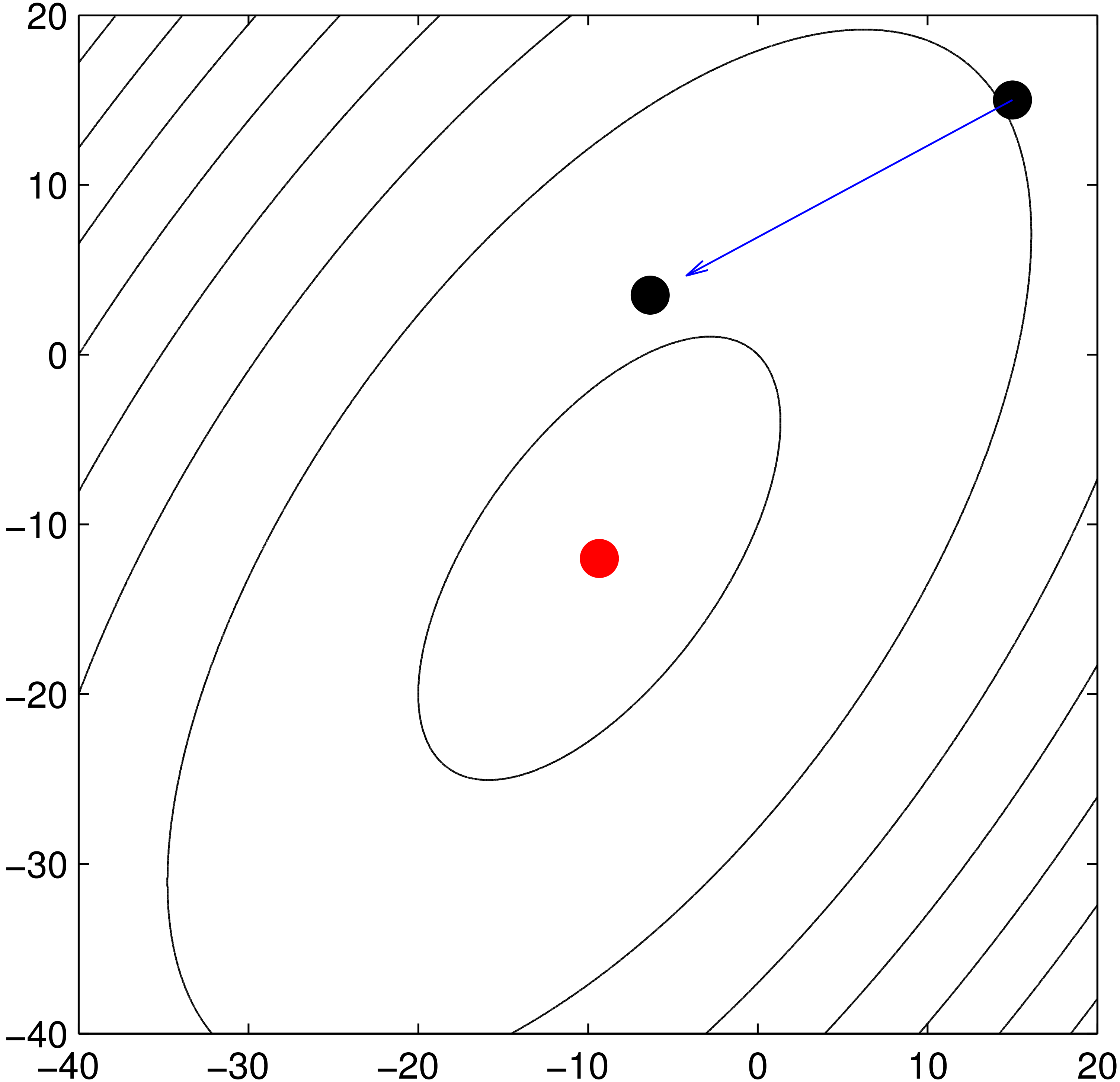


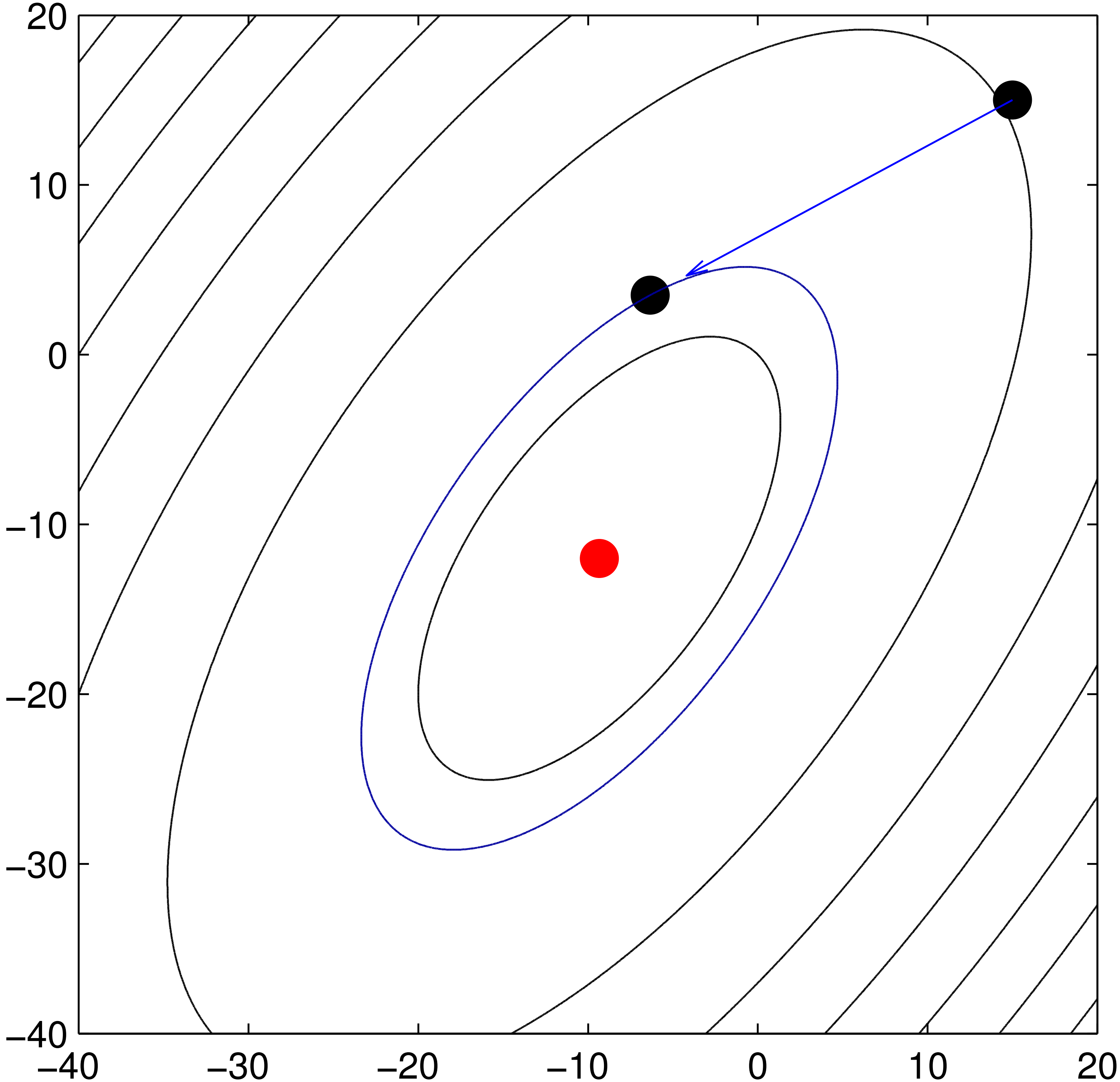
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20



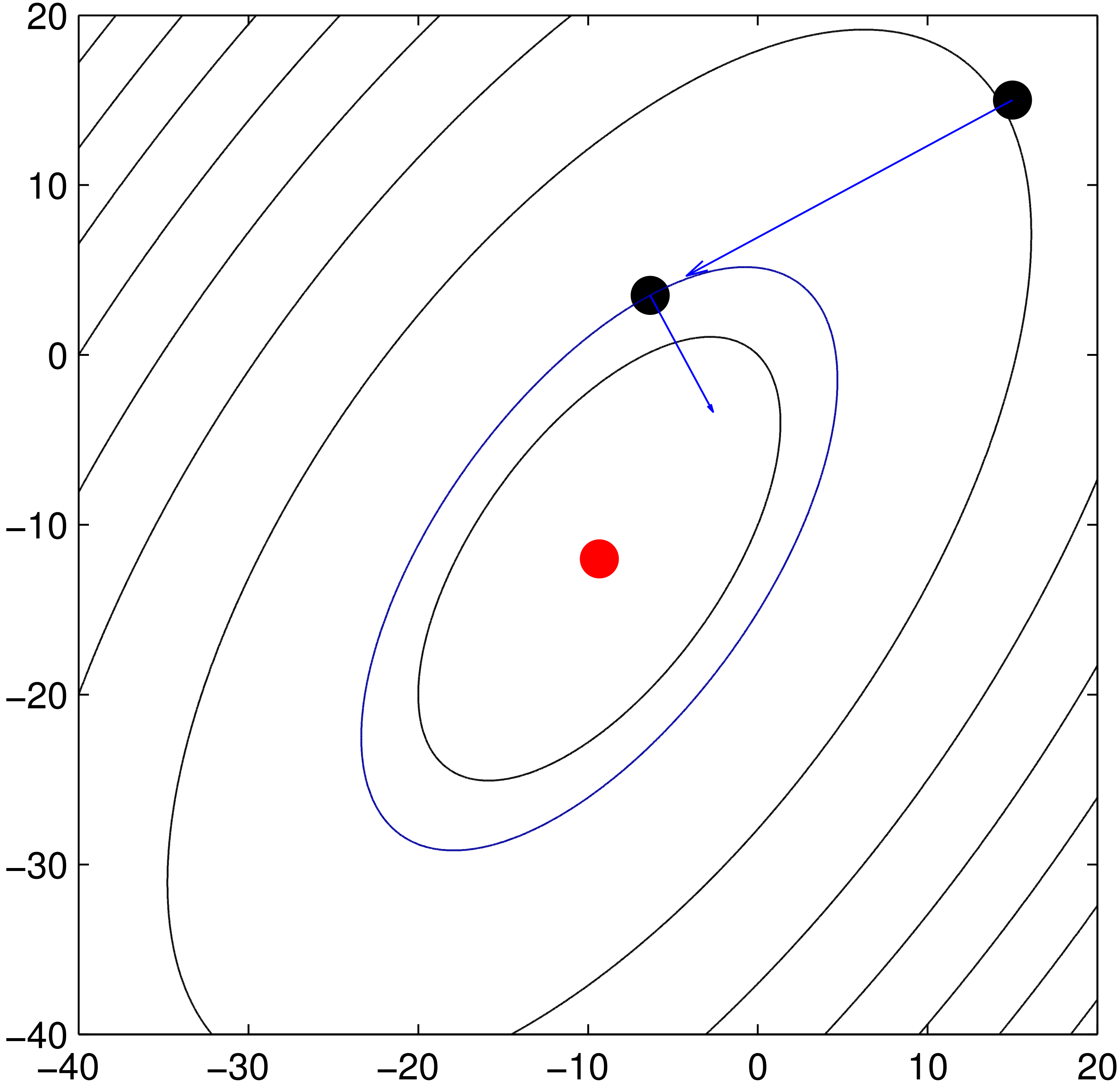
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20
20

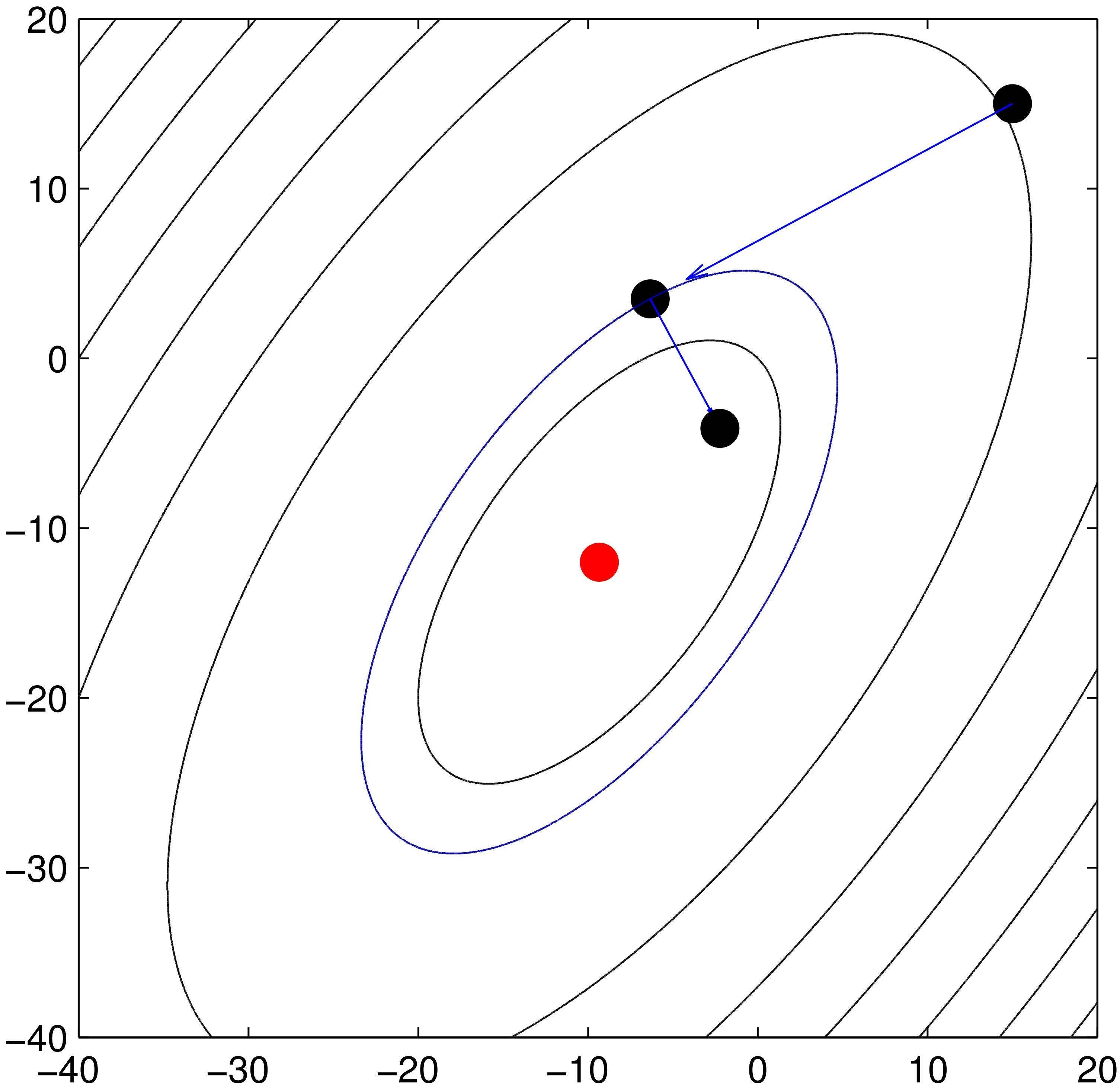

20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

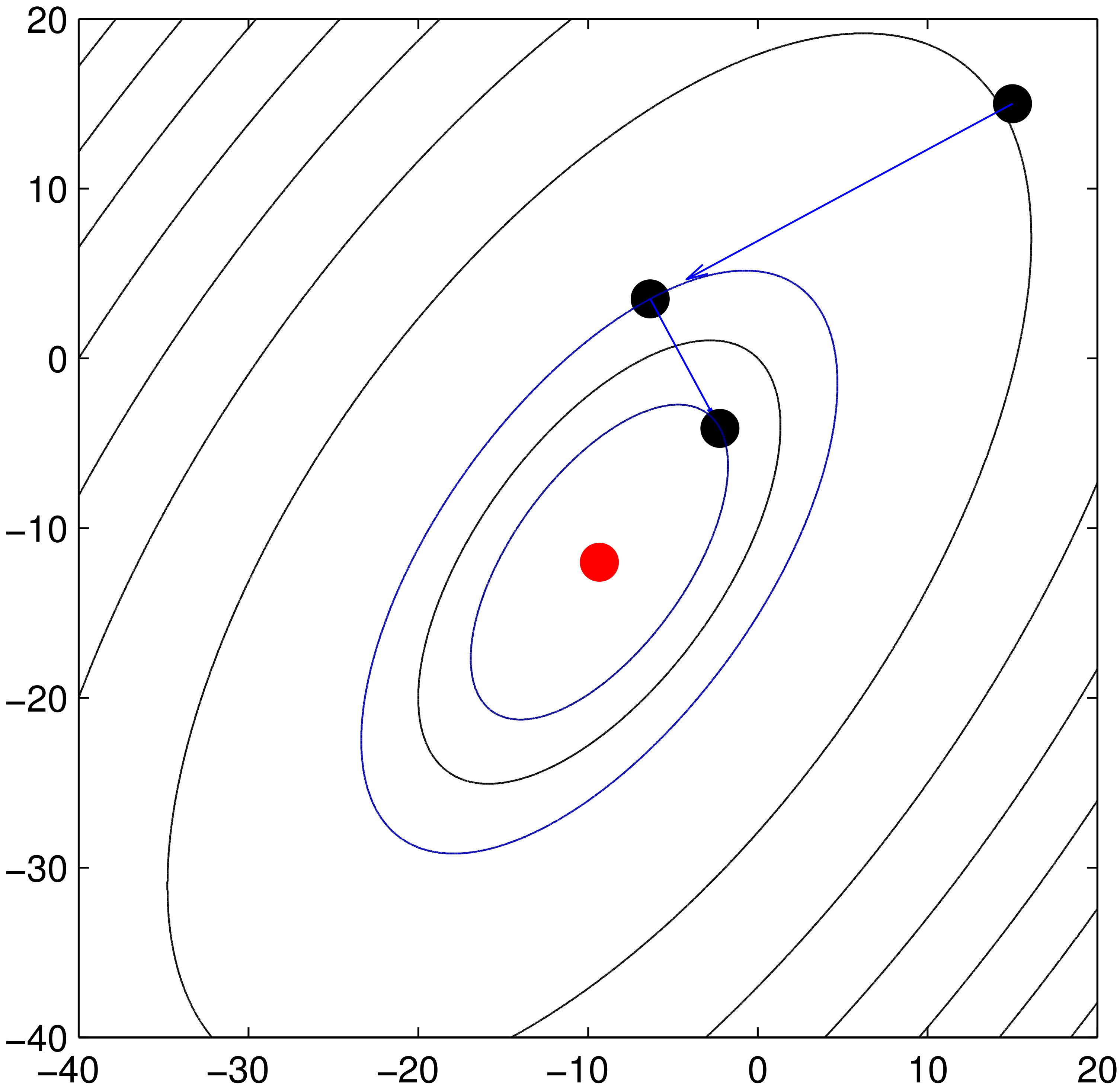


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

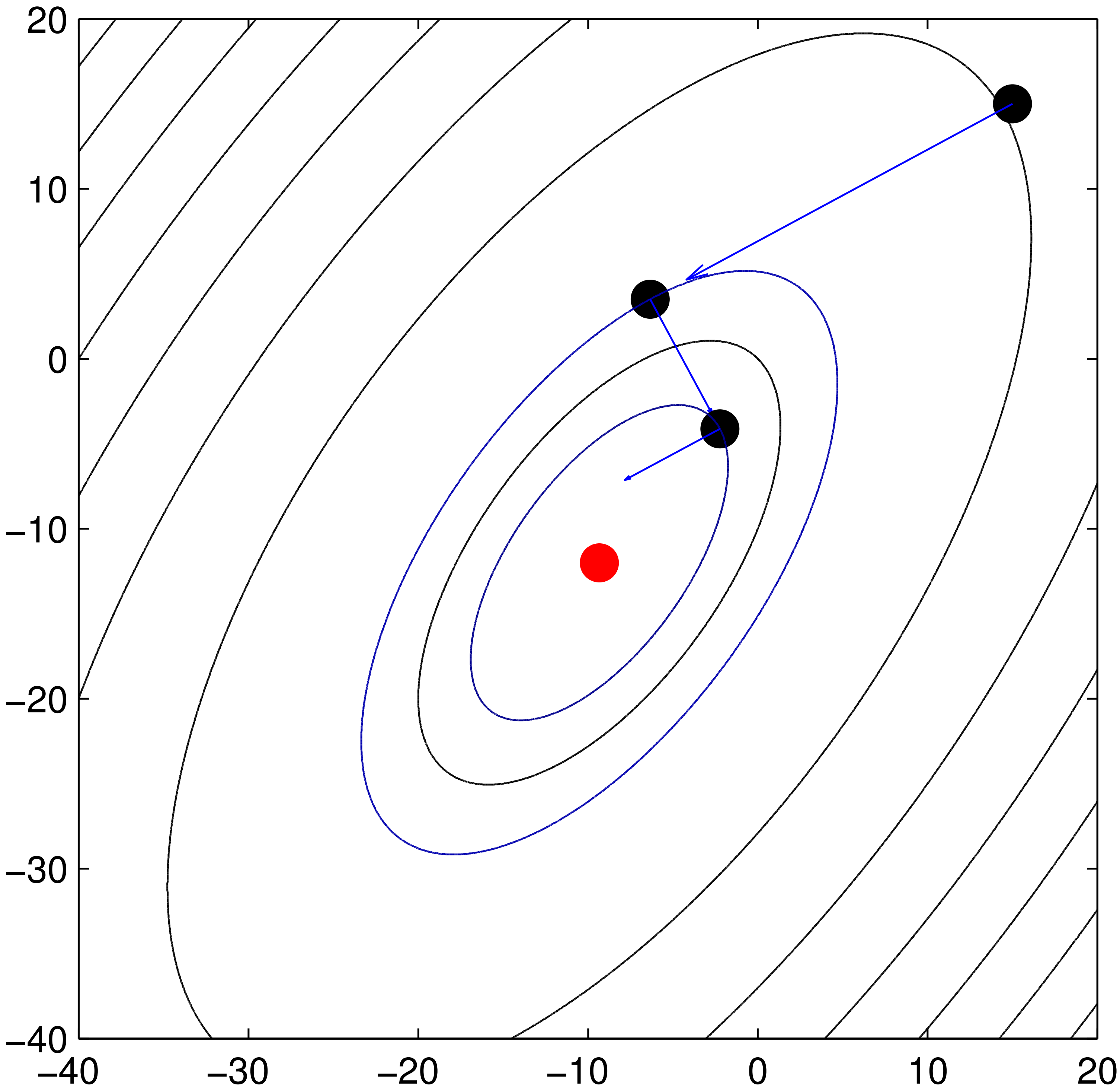


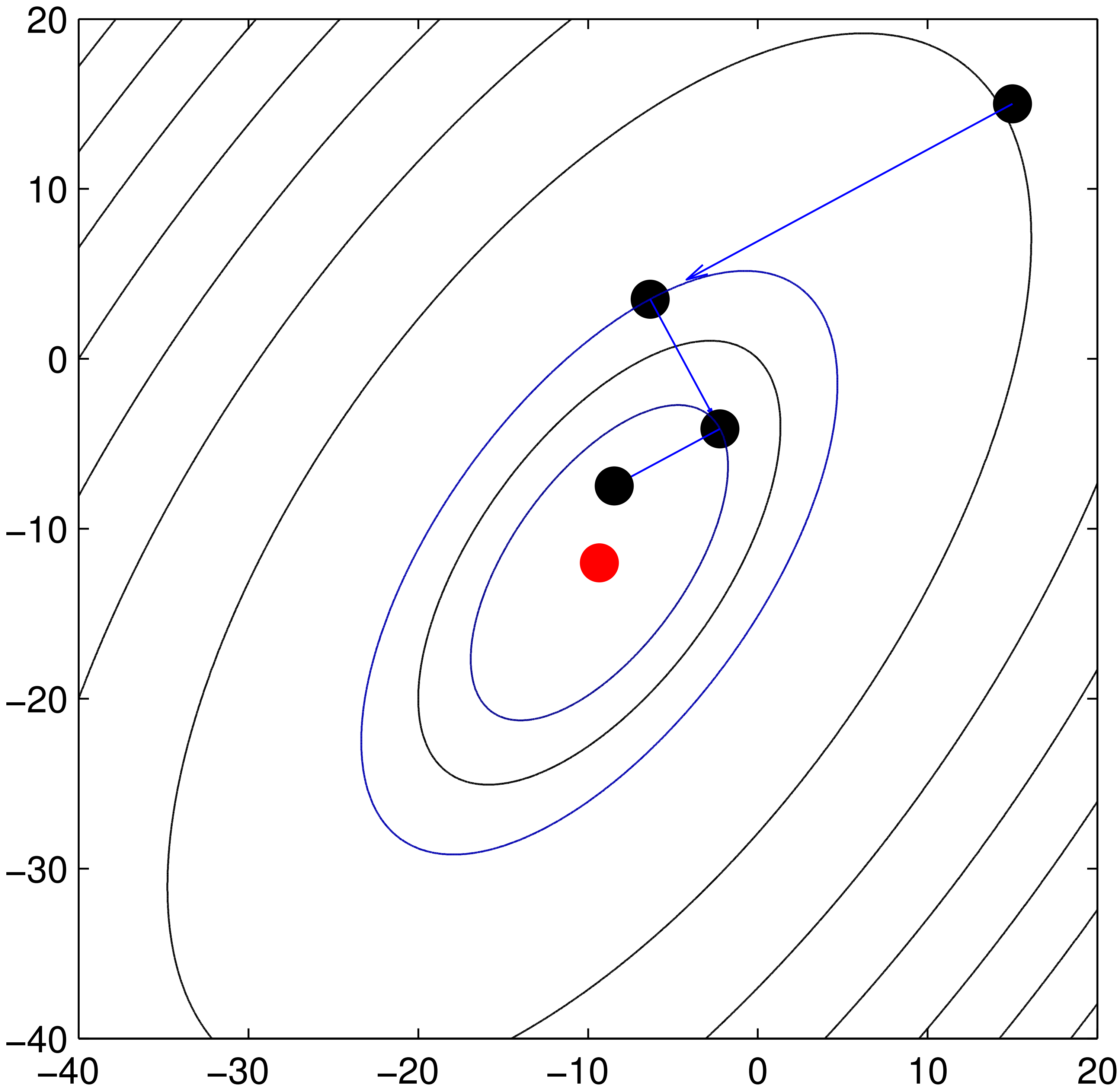
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

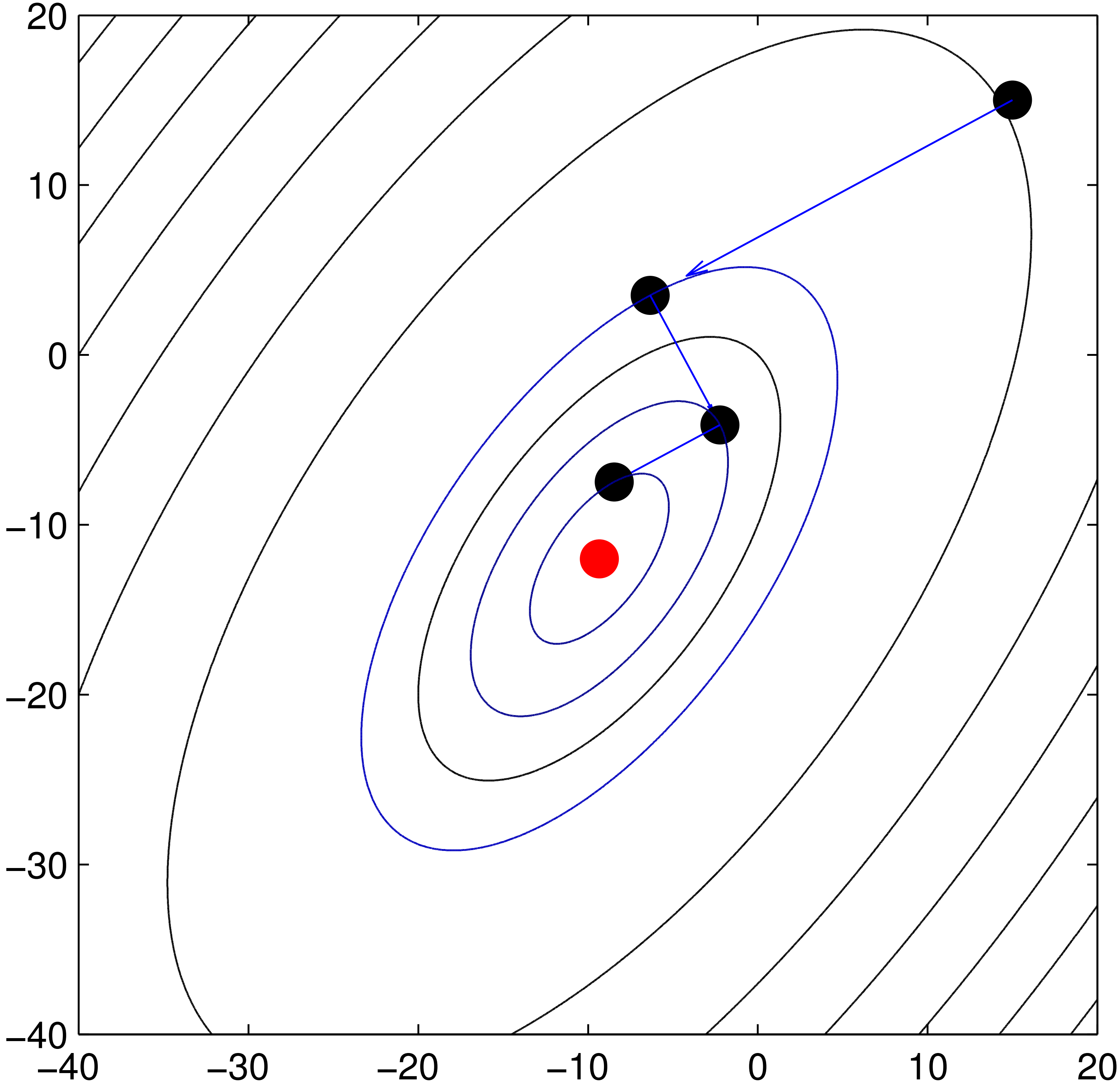

20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

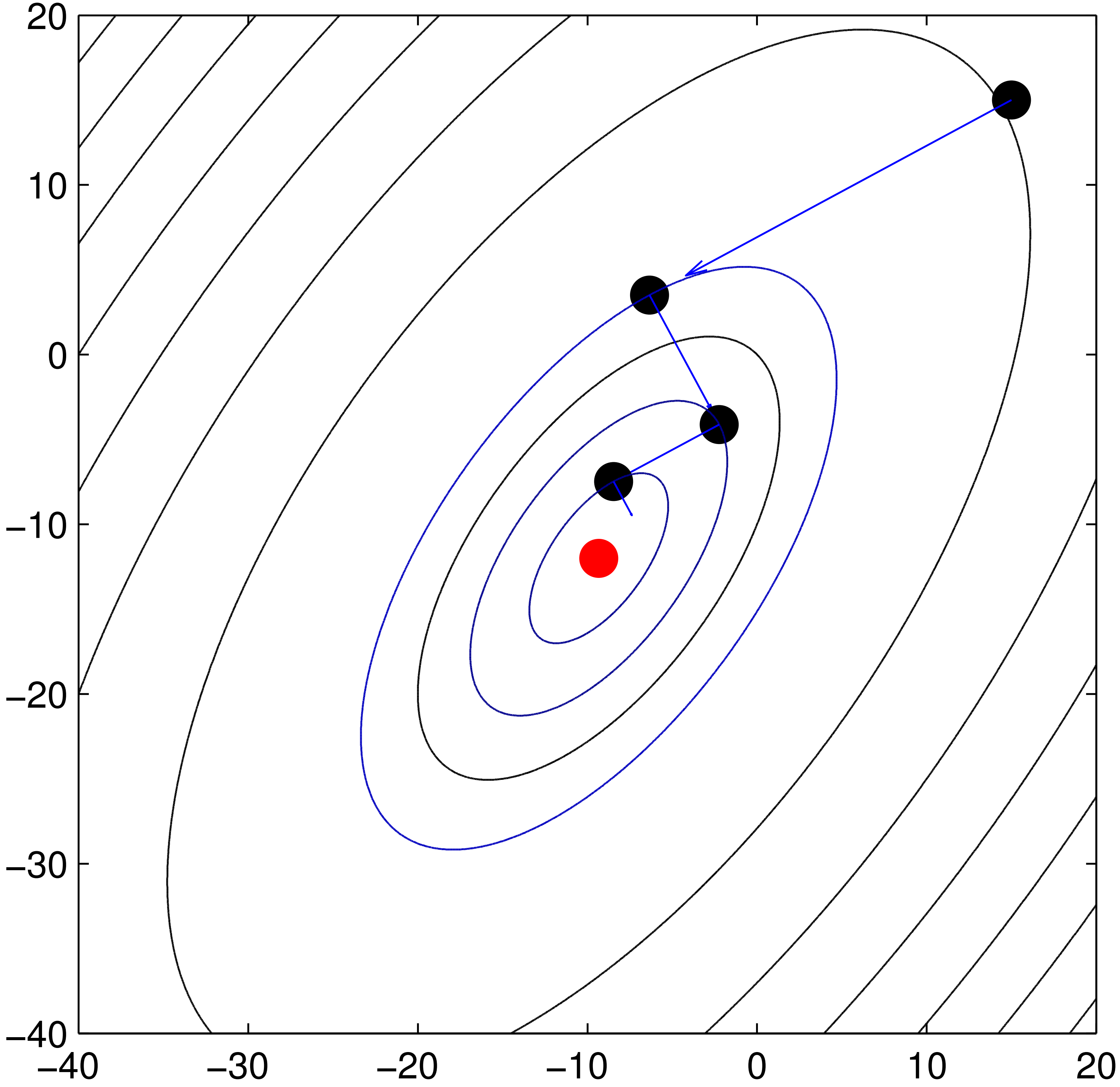


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

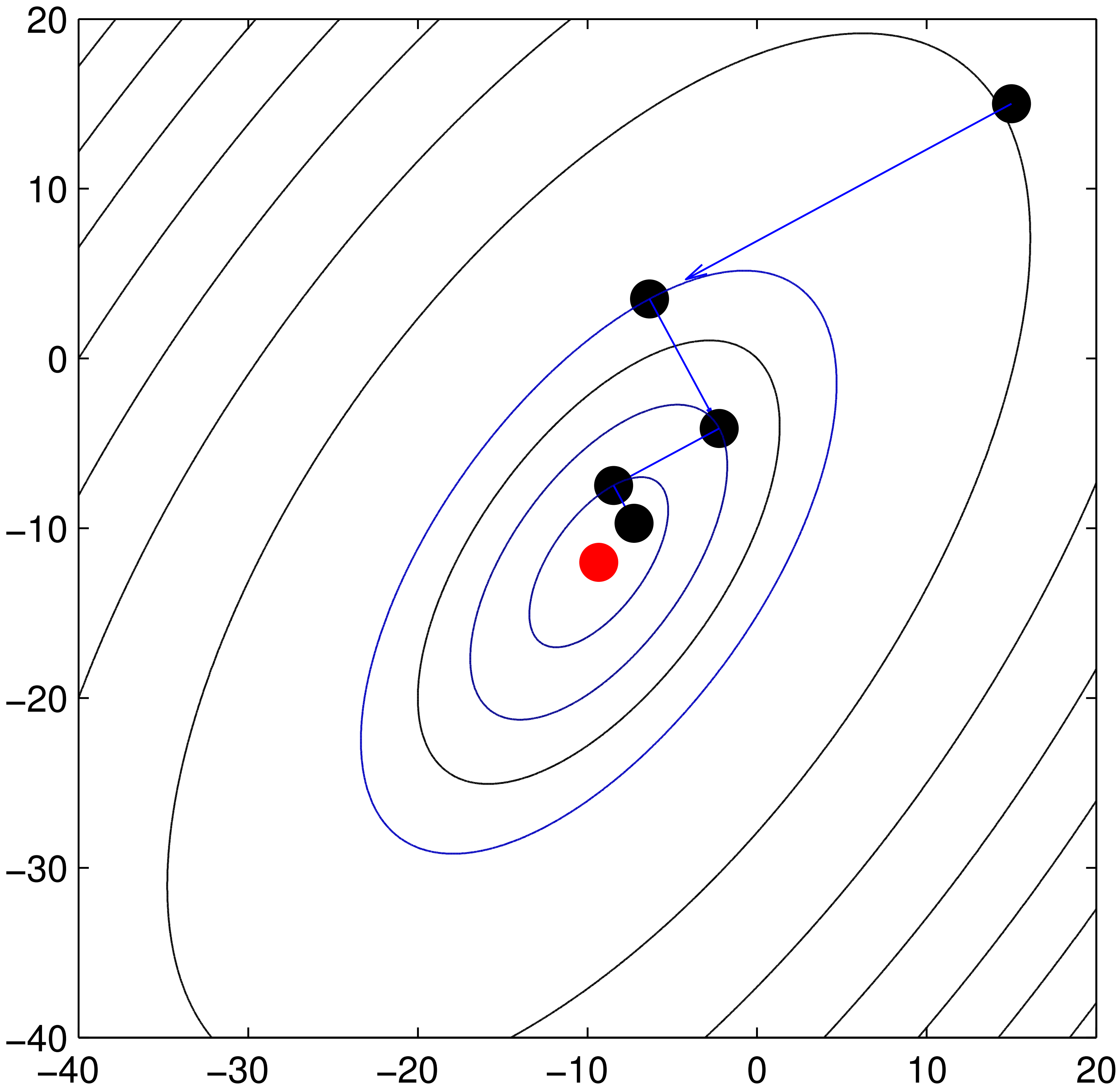

20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

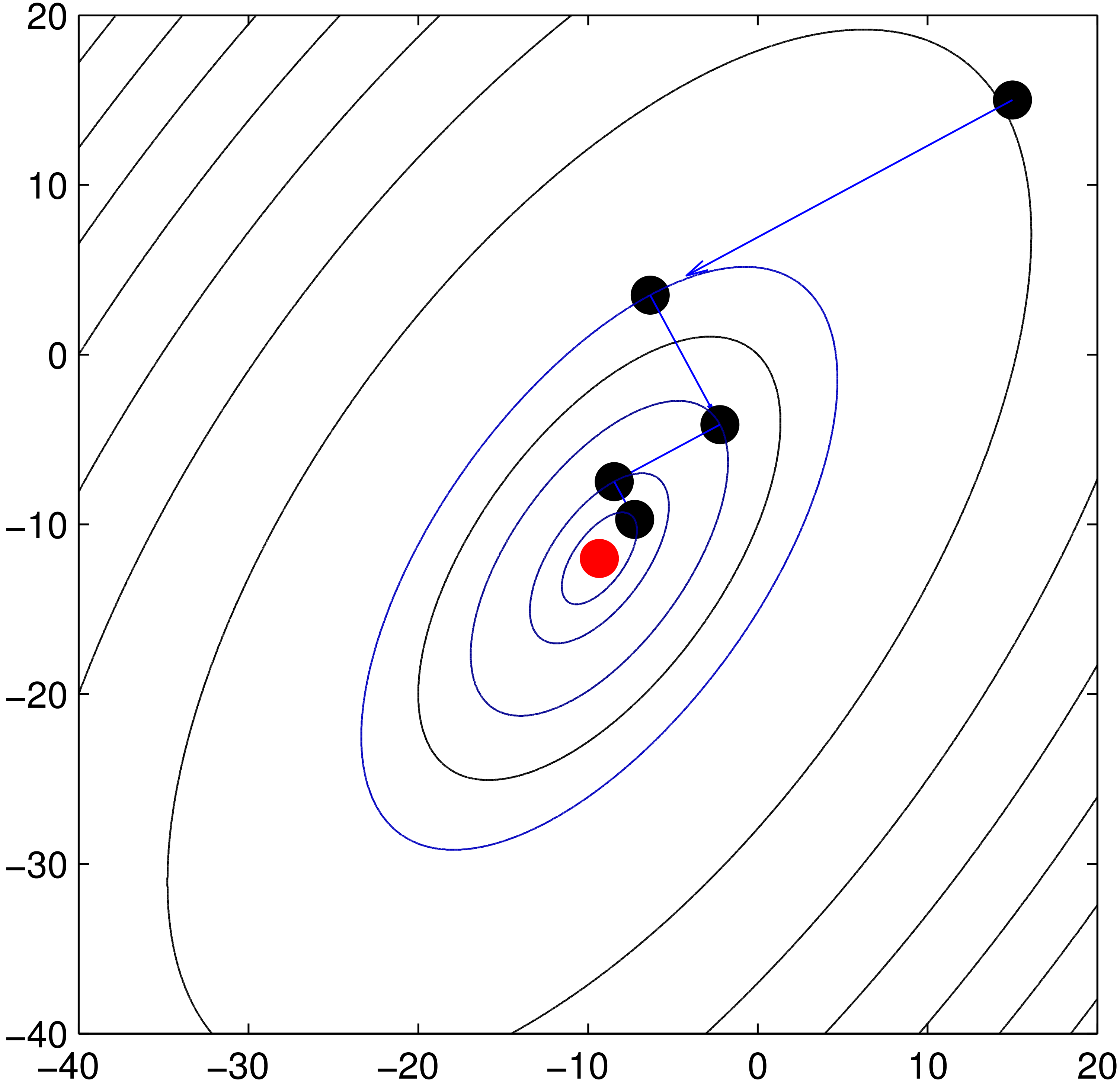

20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20



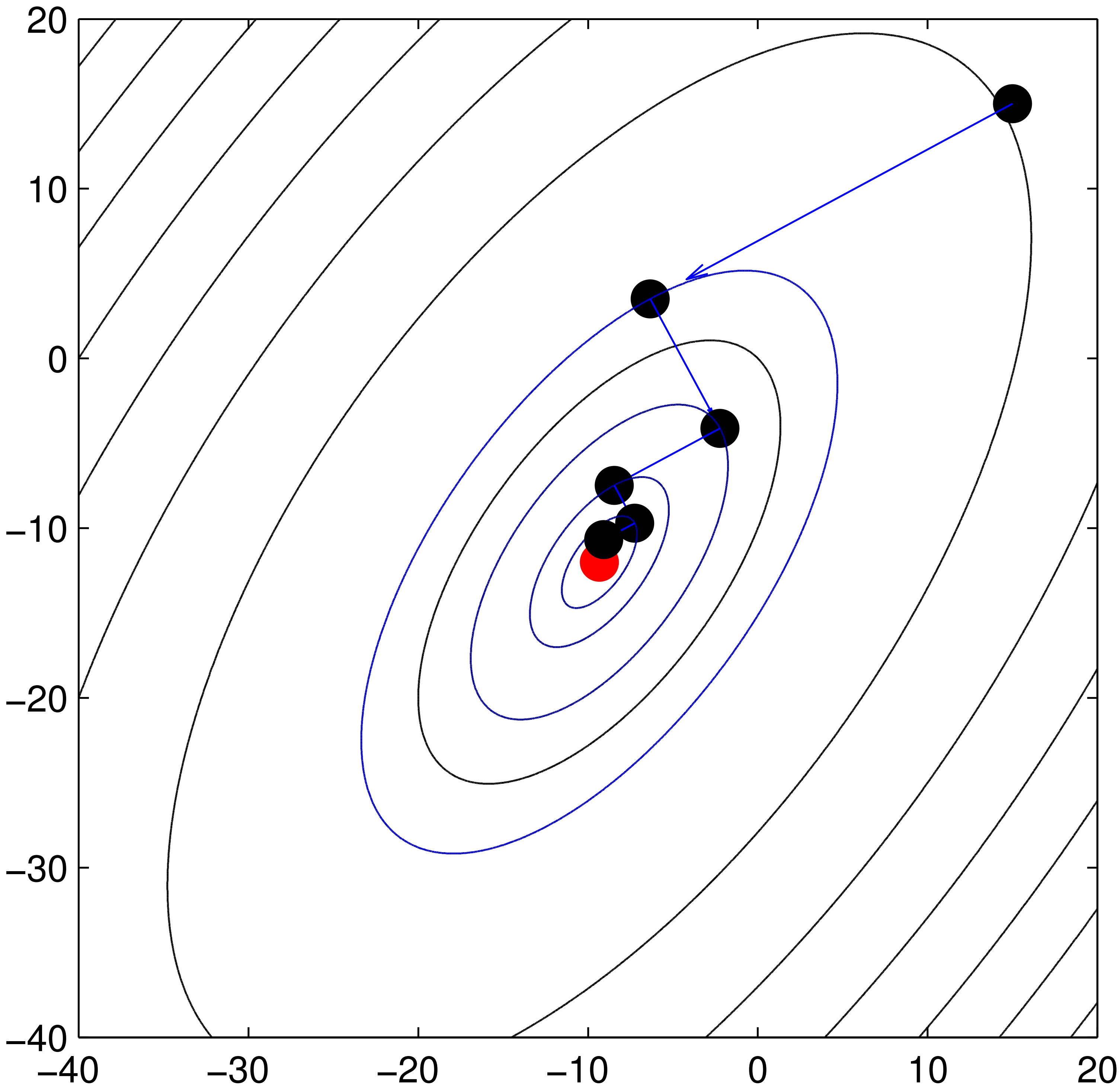
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

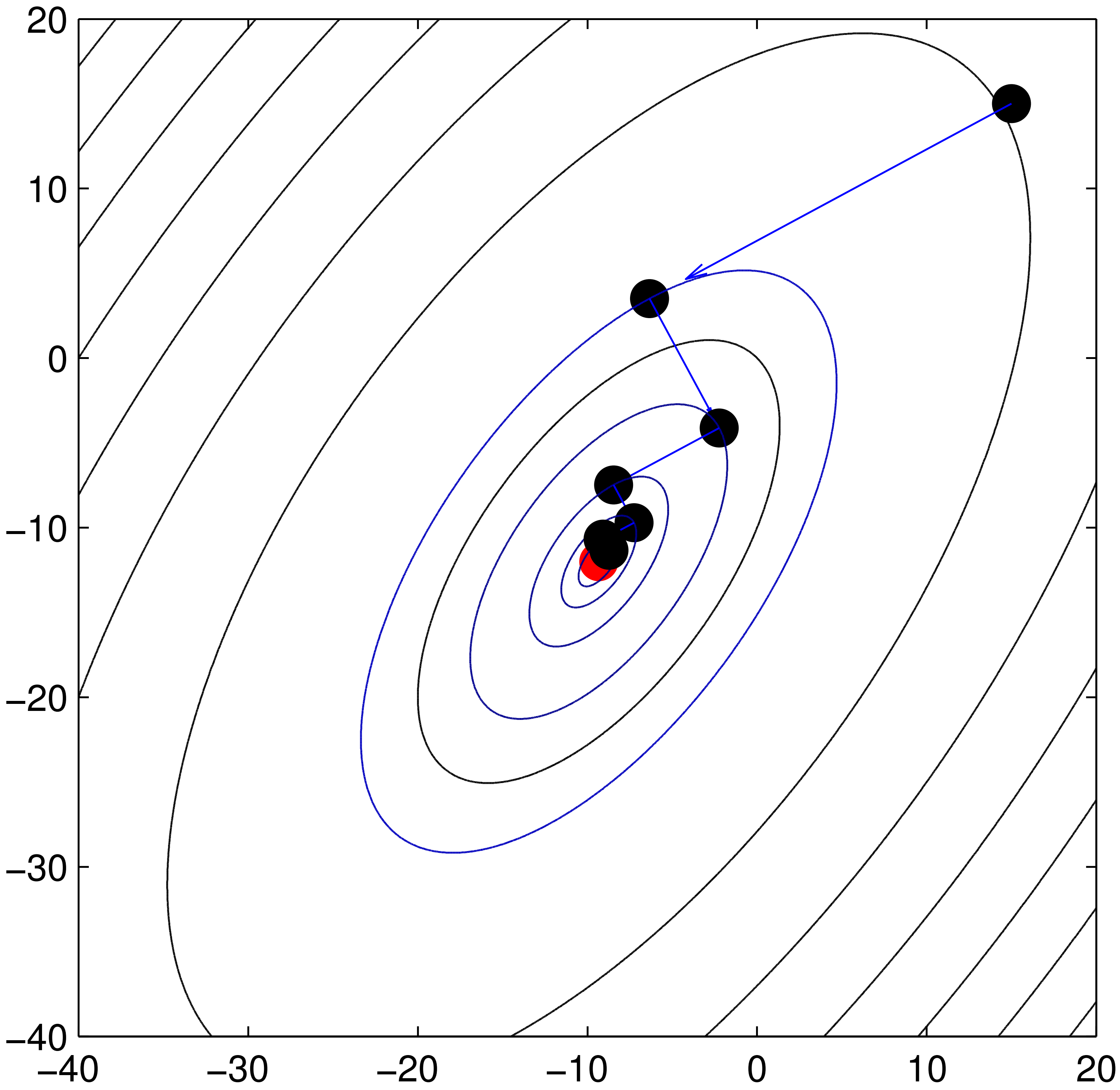


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

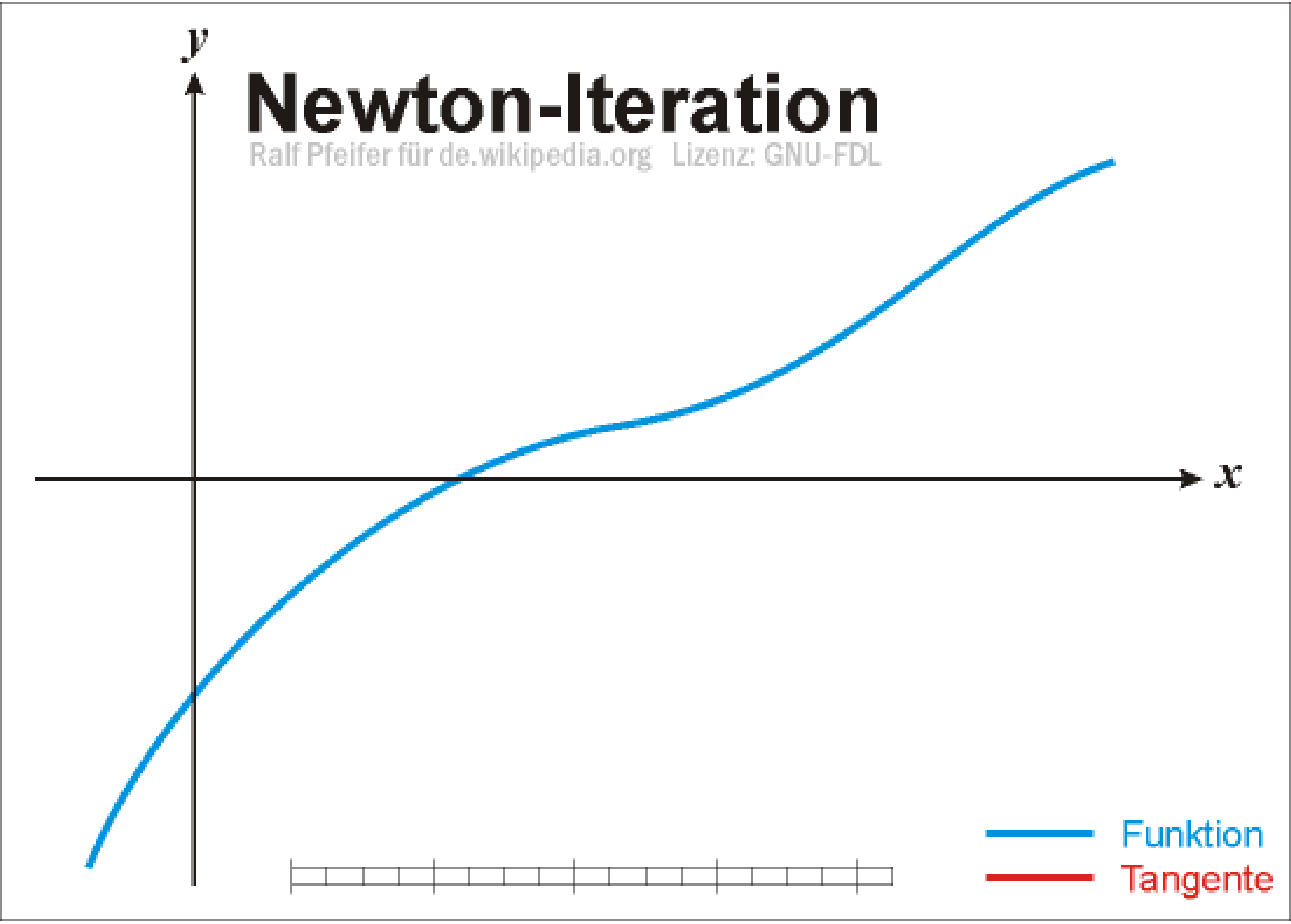


20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20

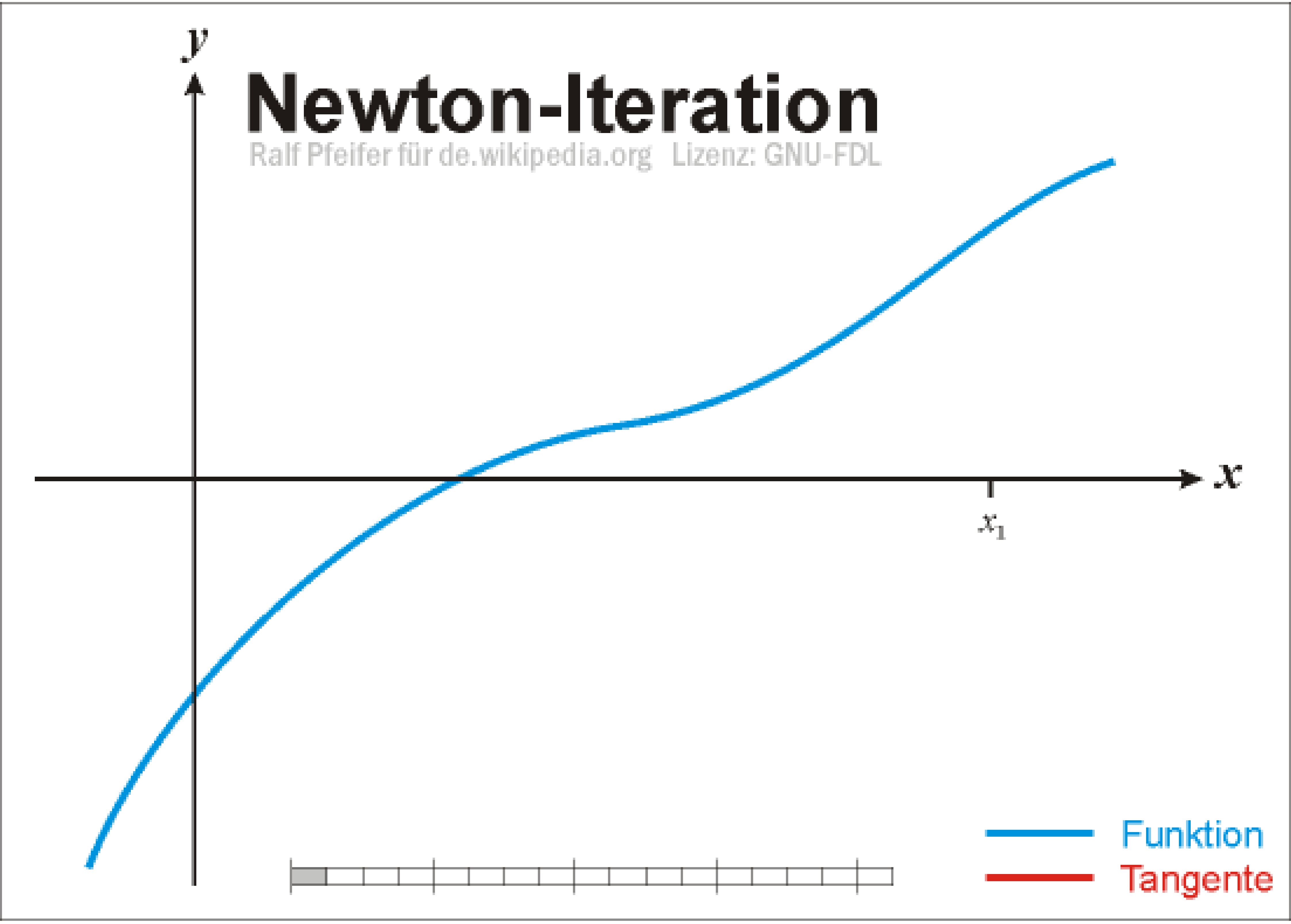

20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20



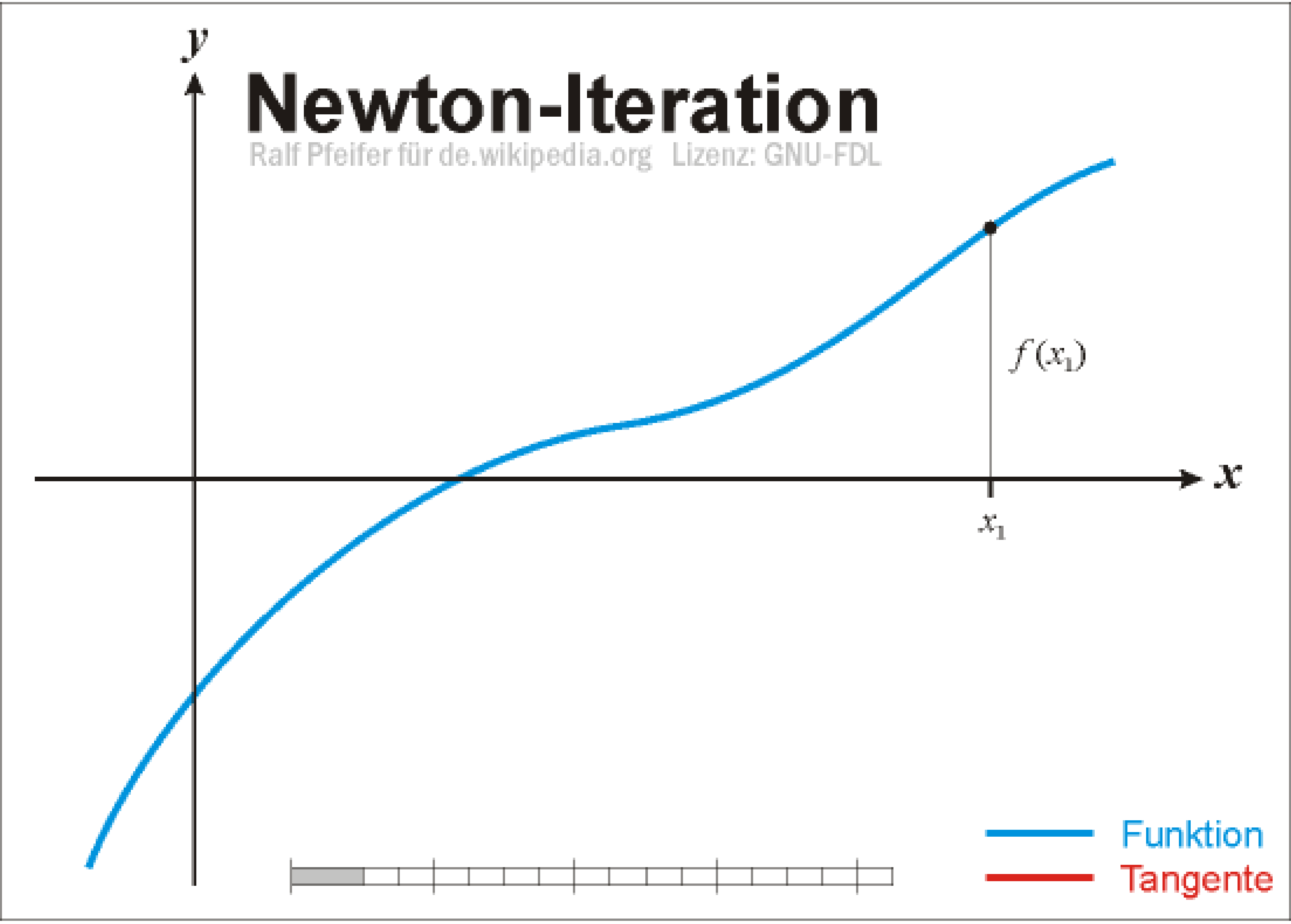
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20



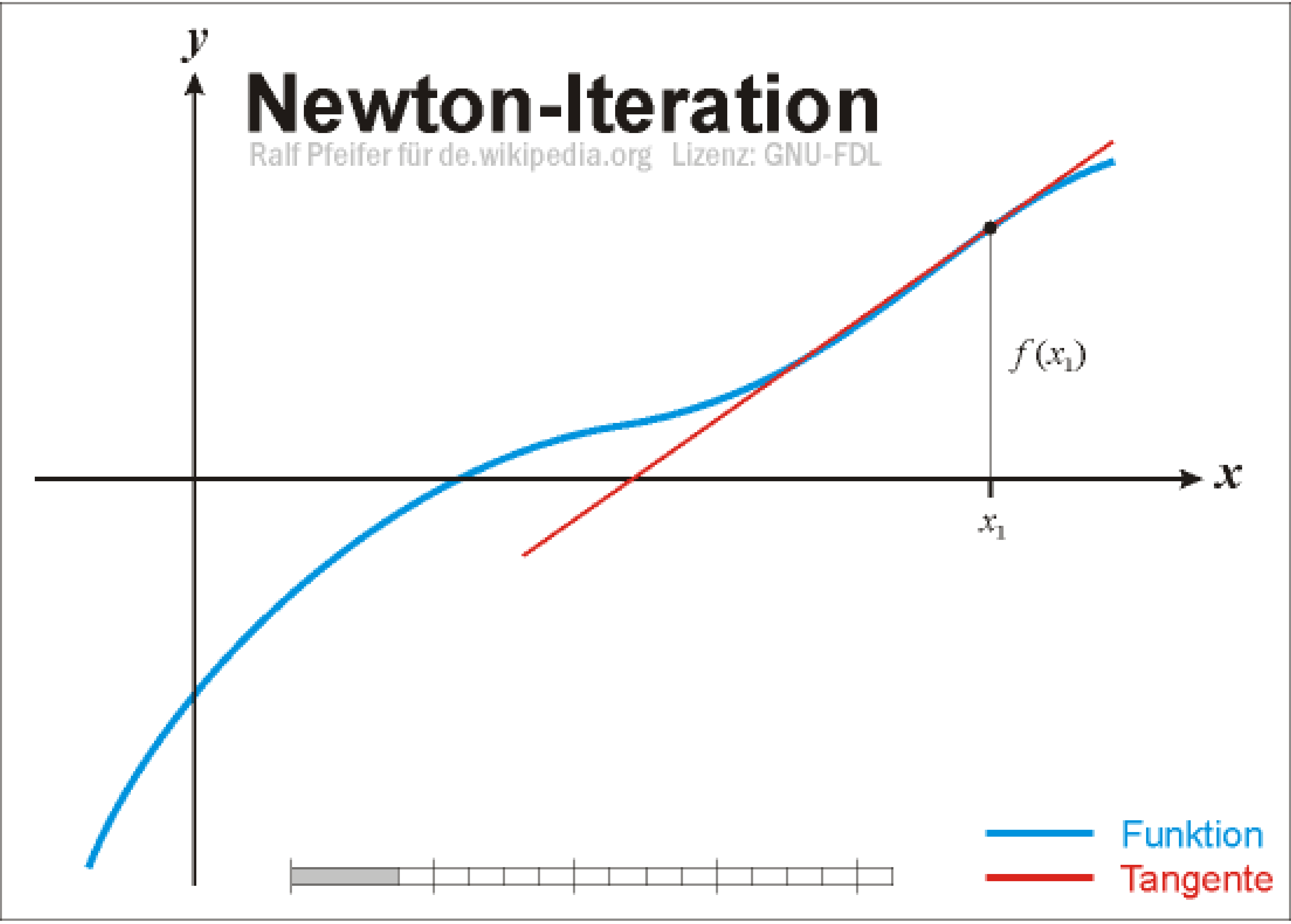
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
Funktion
Tangente



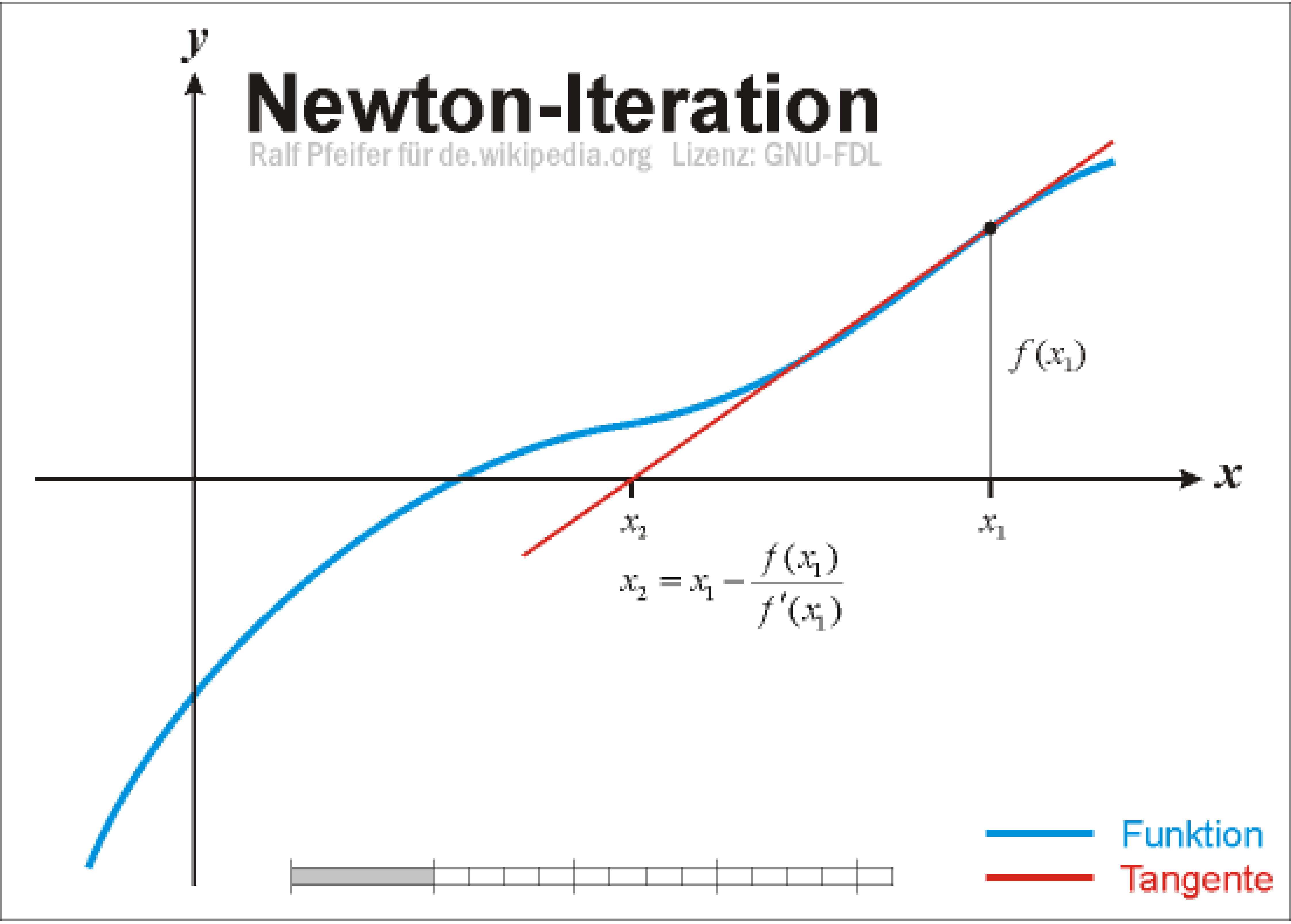
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
x_1
Funktion
Tangente



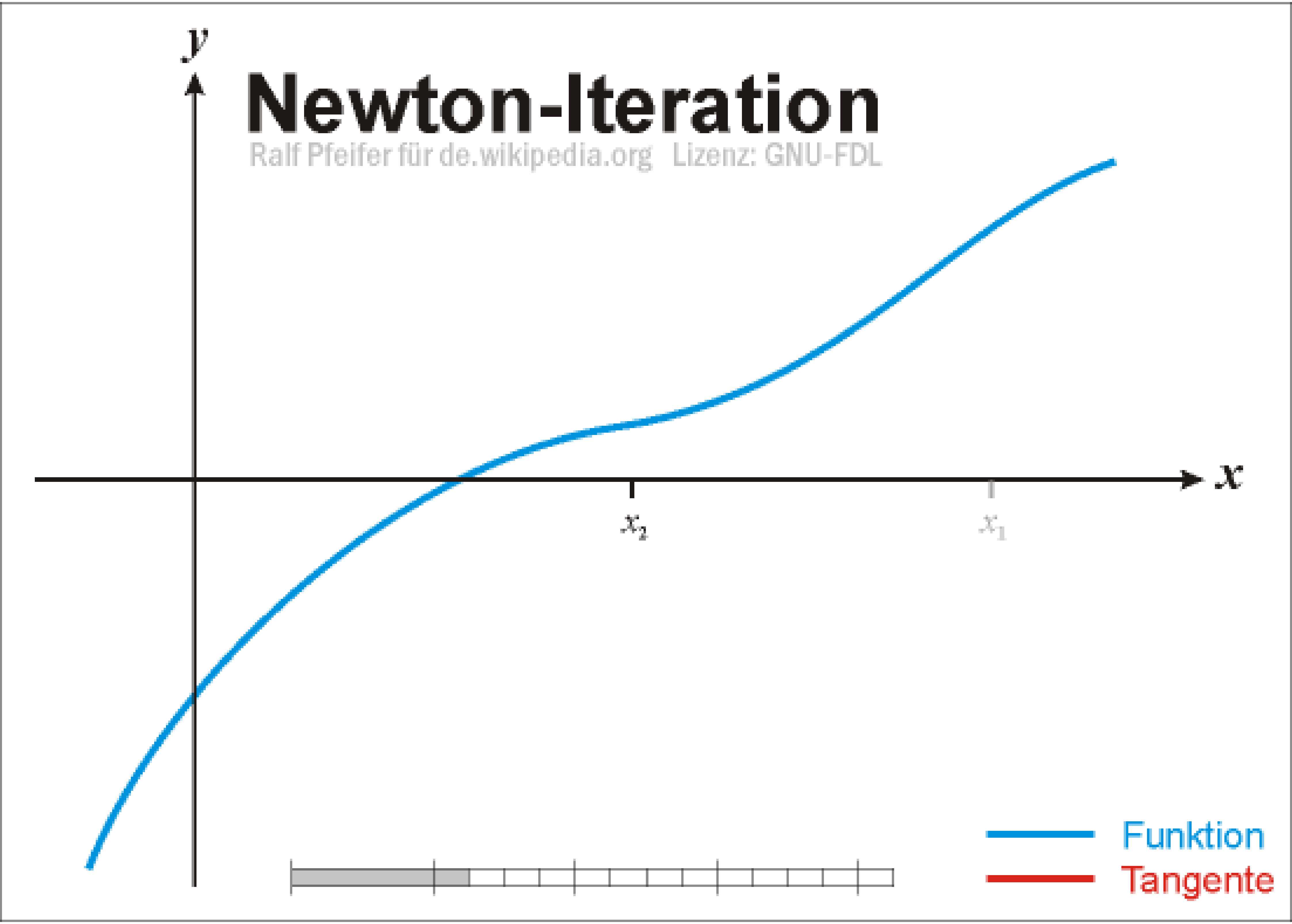
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
$f(x_1)$
x
x_1
Funktion
Tangente

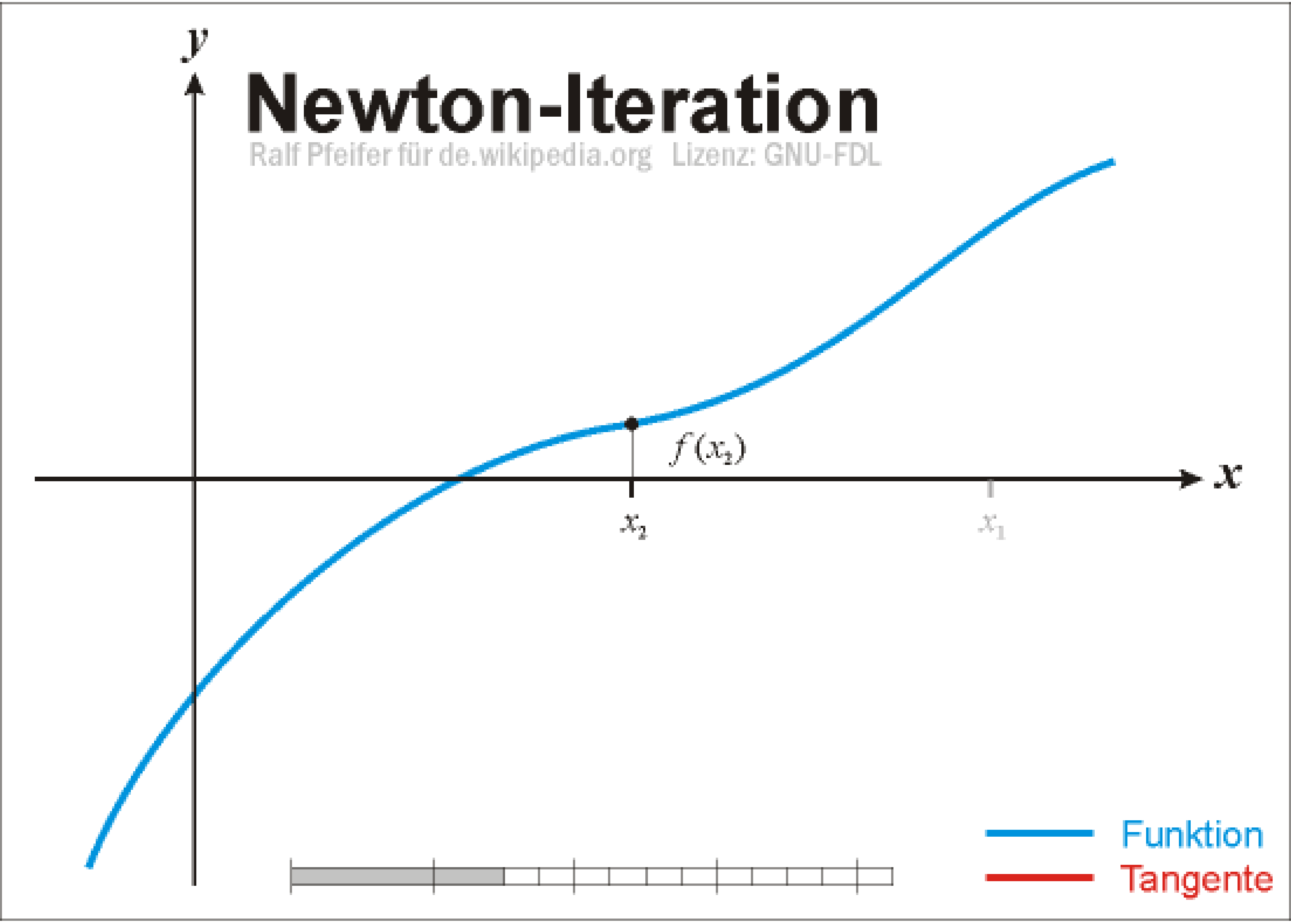


Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
$f(x_1)$
x_1
Funktion
Tangente

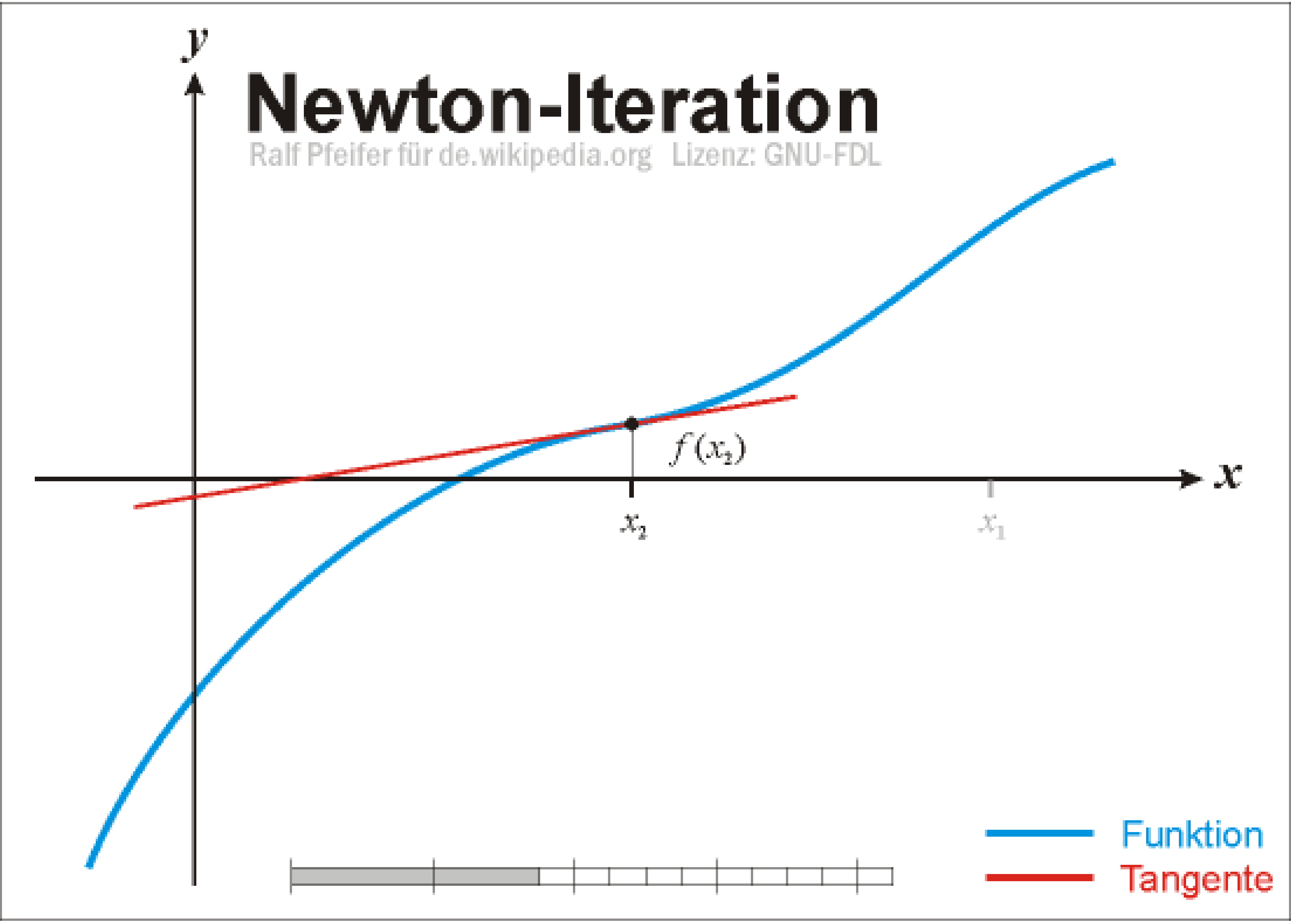


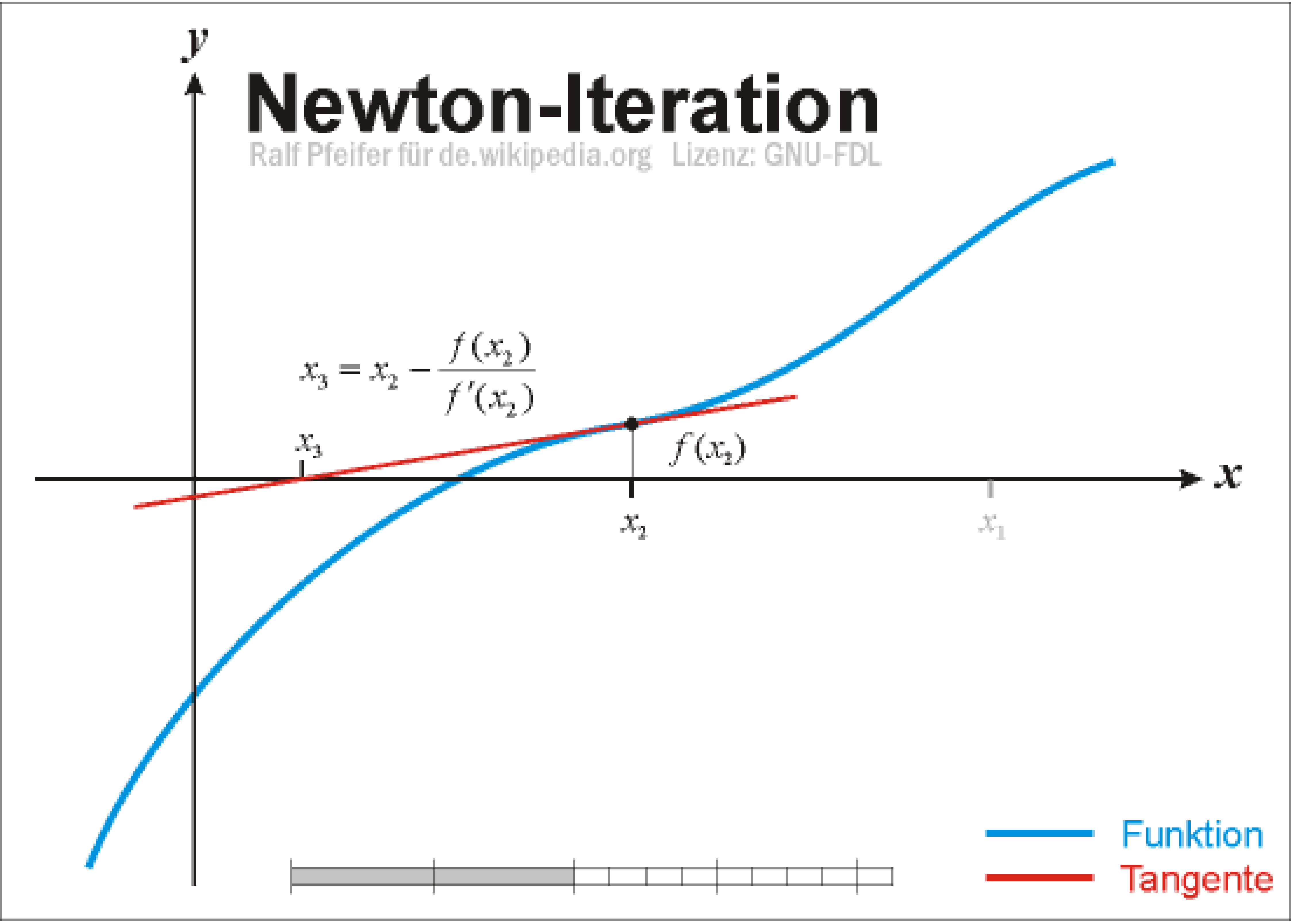
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
$f(x_1)$
x
x_2
x_1
$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$
Funktion
Tangente


Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
x_2
x_1
Funktion
Tangente

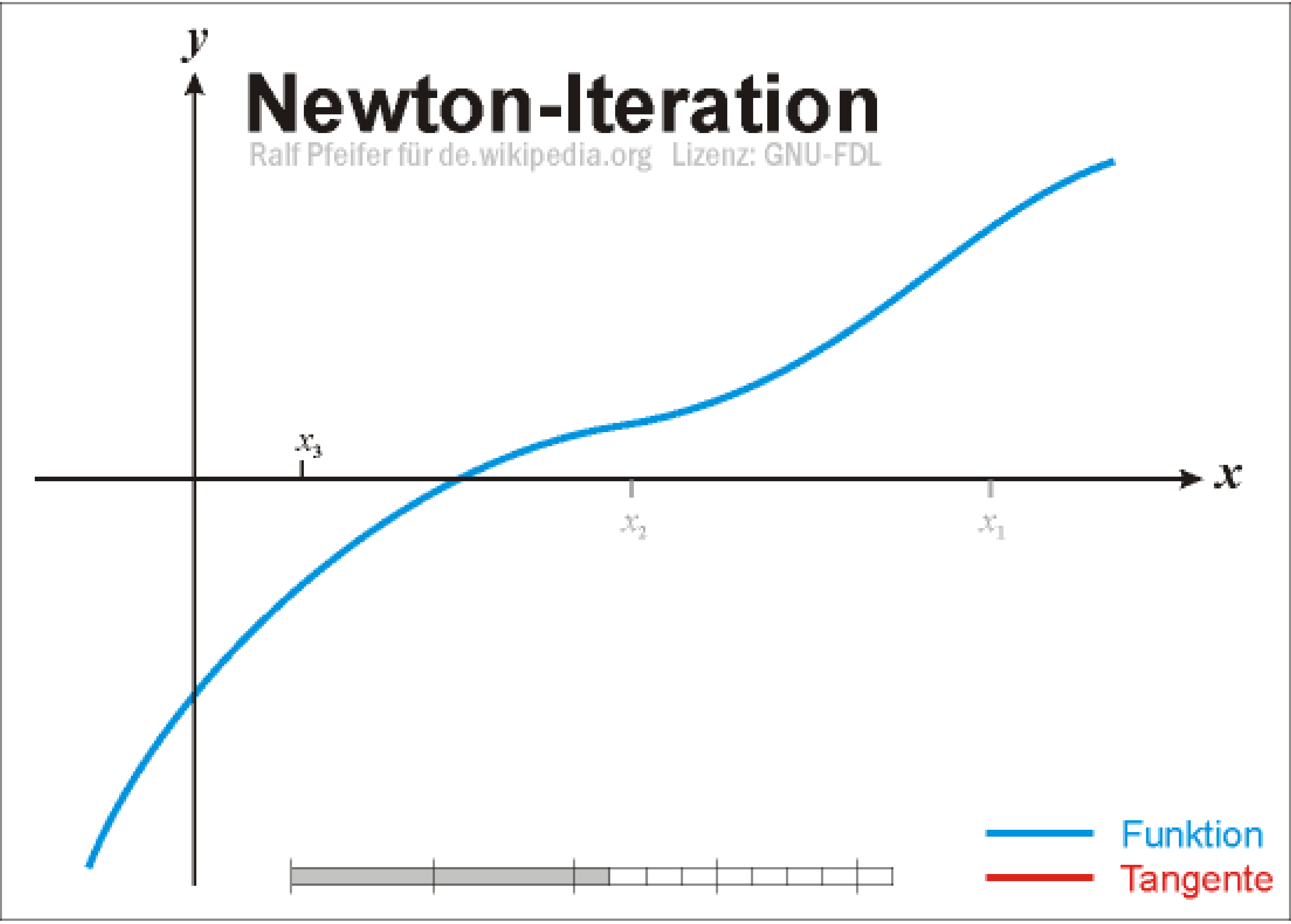


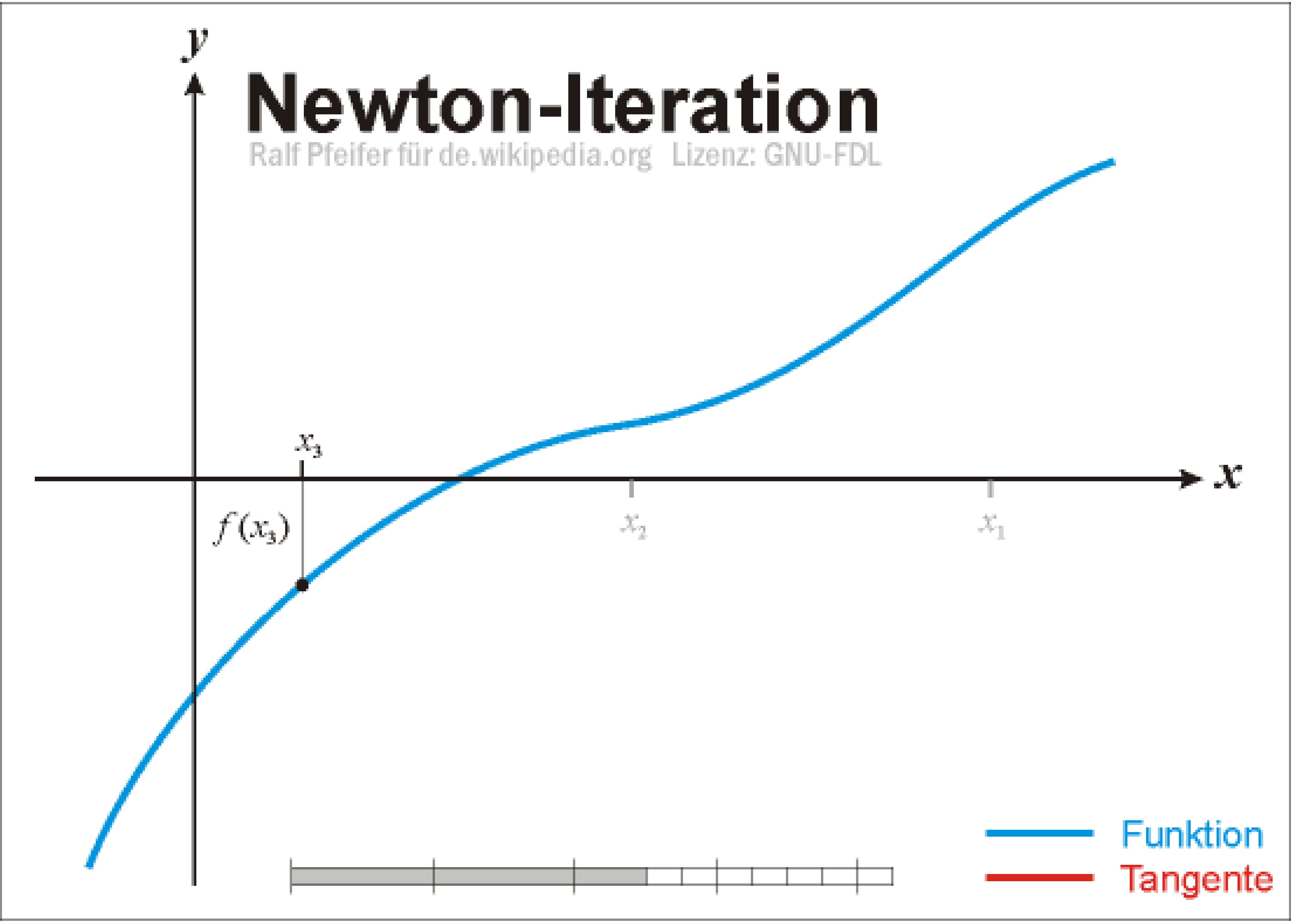
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
$f(x_2)$
x
x_2
x_1
Funktion
Tangente


Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
f(x2)
x2
x1
Funktion
Tangente

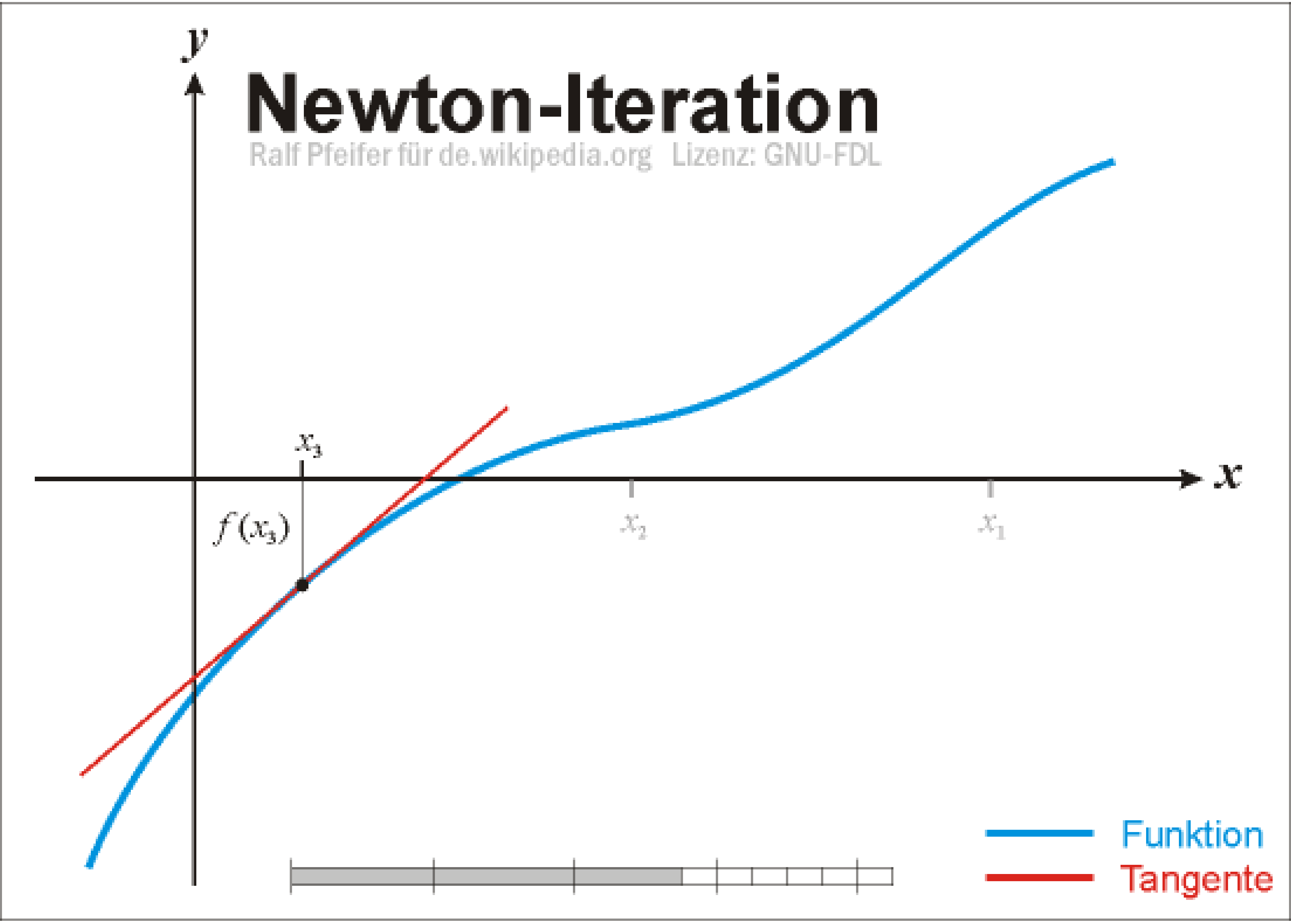


Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
$x_3 = x_2 - \frac{f(x_2)}{f'(x_2)}$
x_3
$f(x_2)$
x_2
x_1
Funktion
Tangente

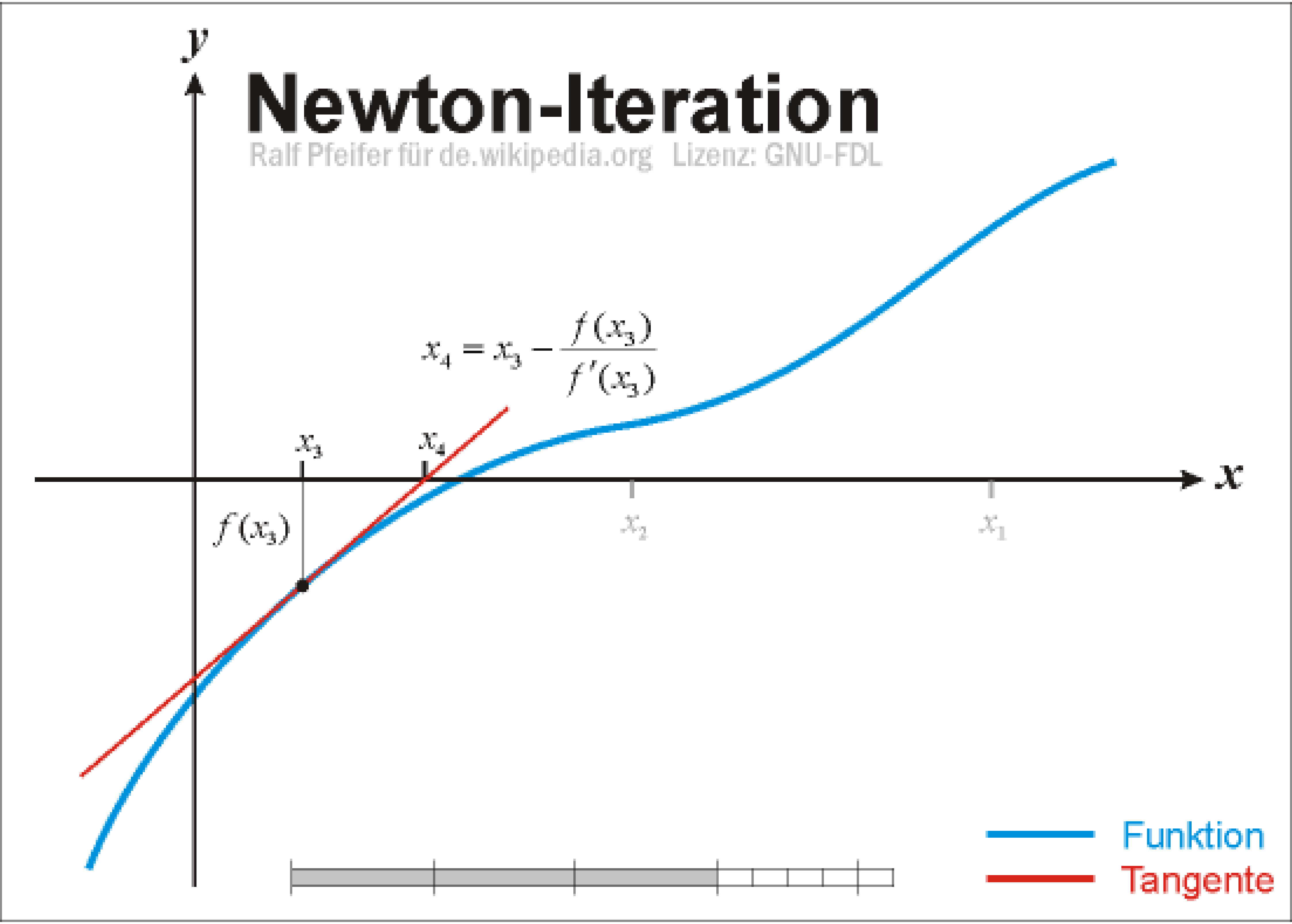

y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
x_3
x_2
x_1
Funktion
Tangente



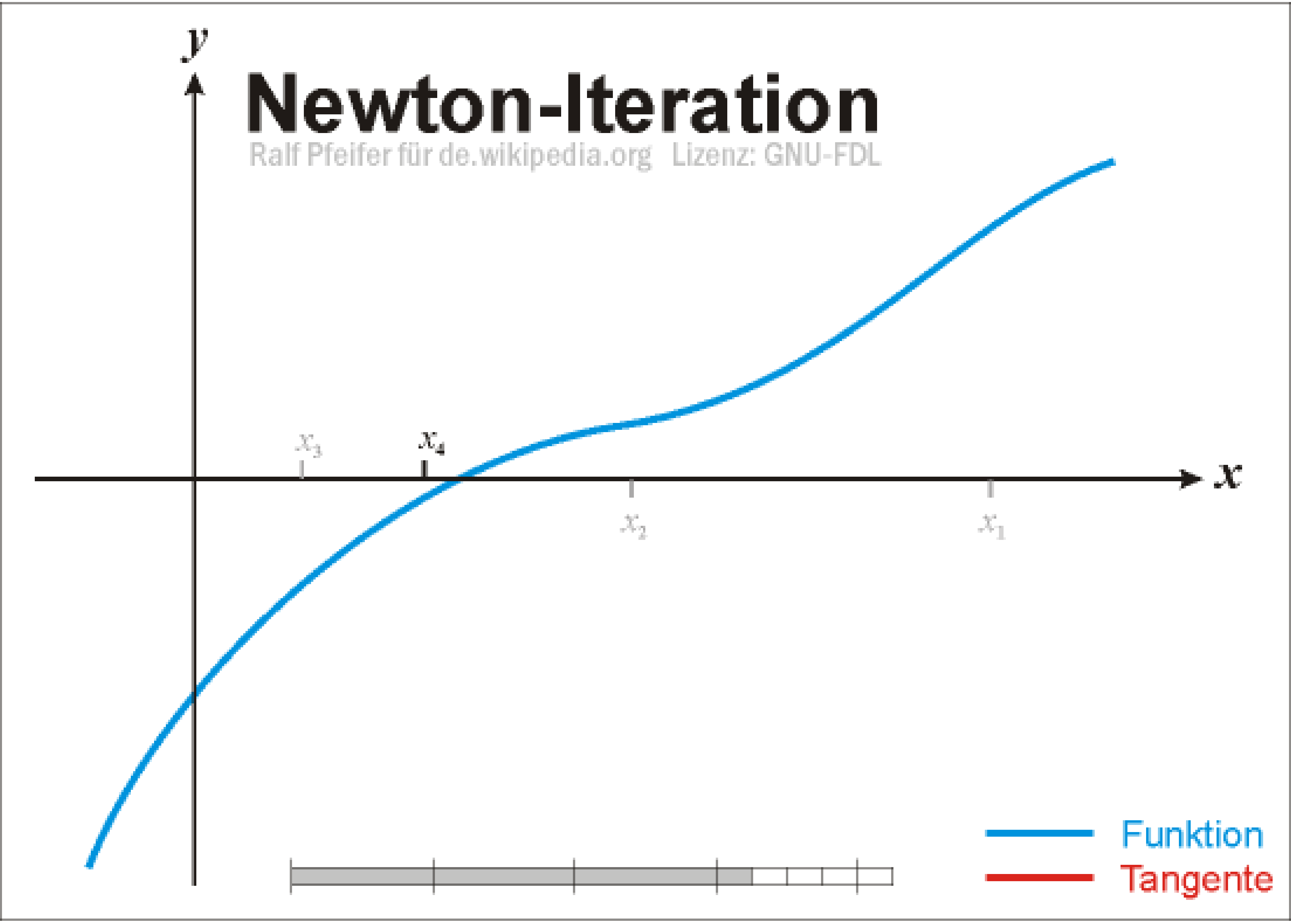
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
x_3
$f(x_3)$
x_2
x_1
Funktion
Tangente

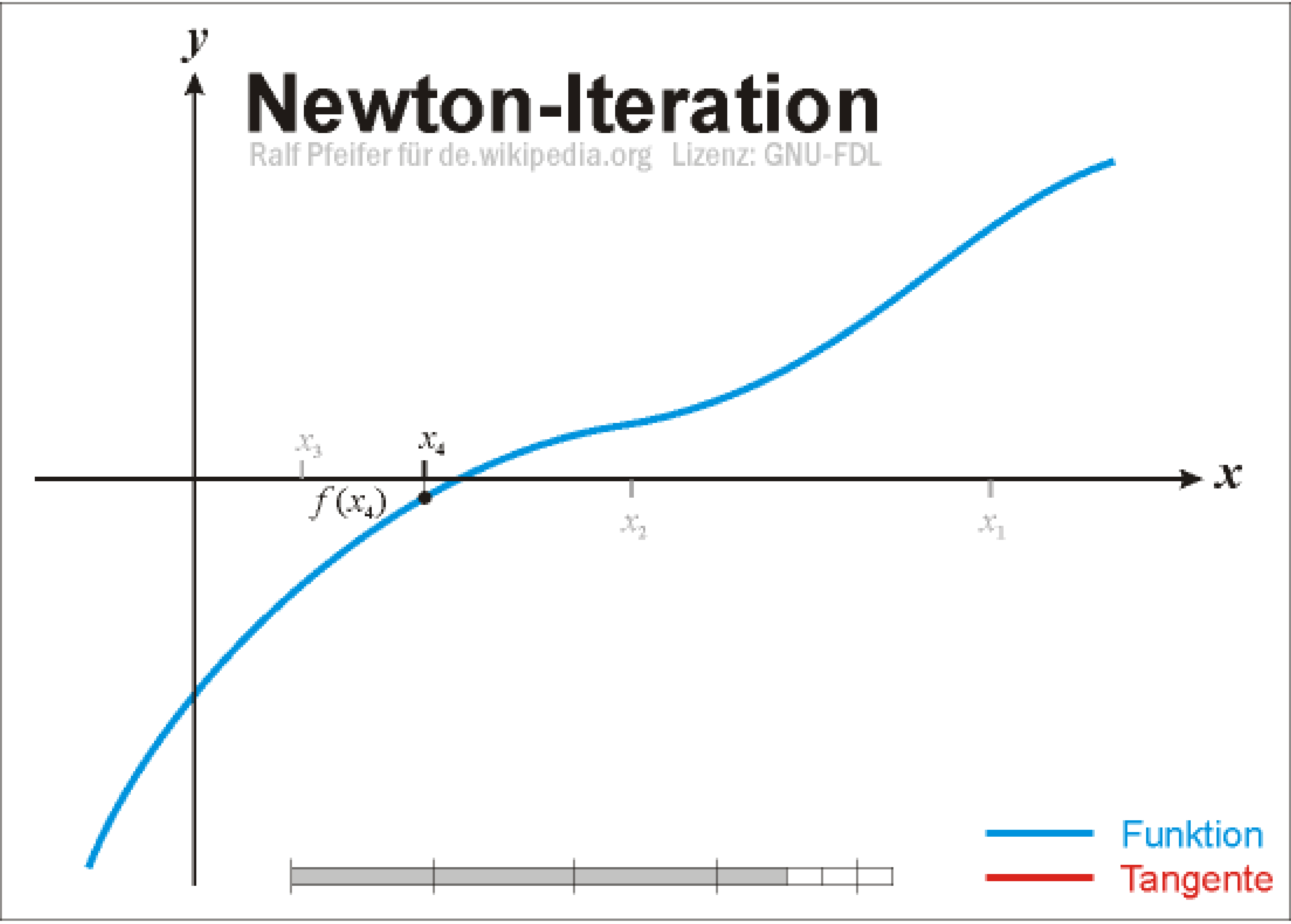


y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x_3
x
$f(x_3)$
x_2
x_1
Funktion
Tangente

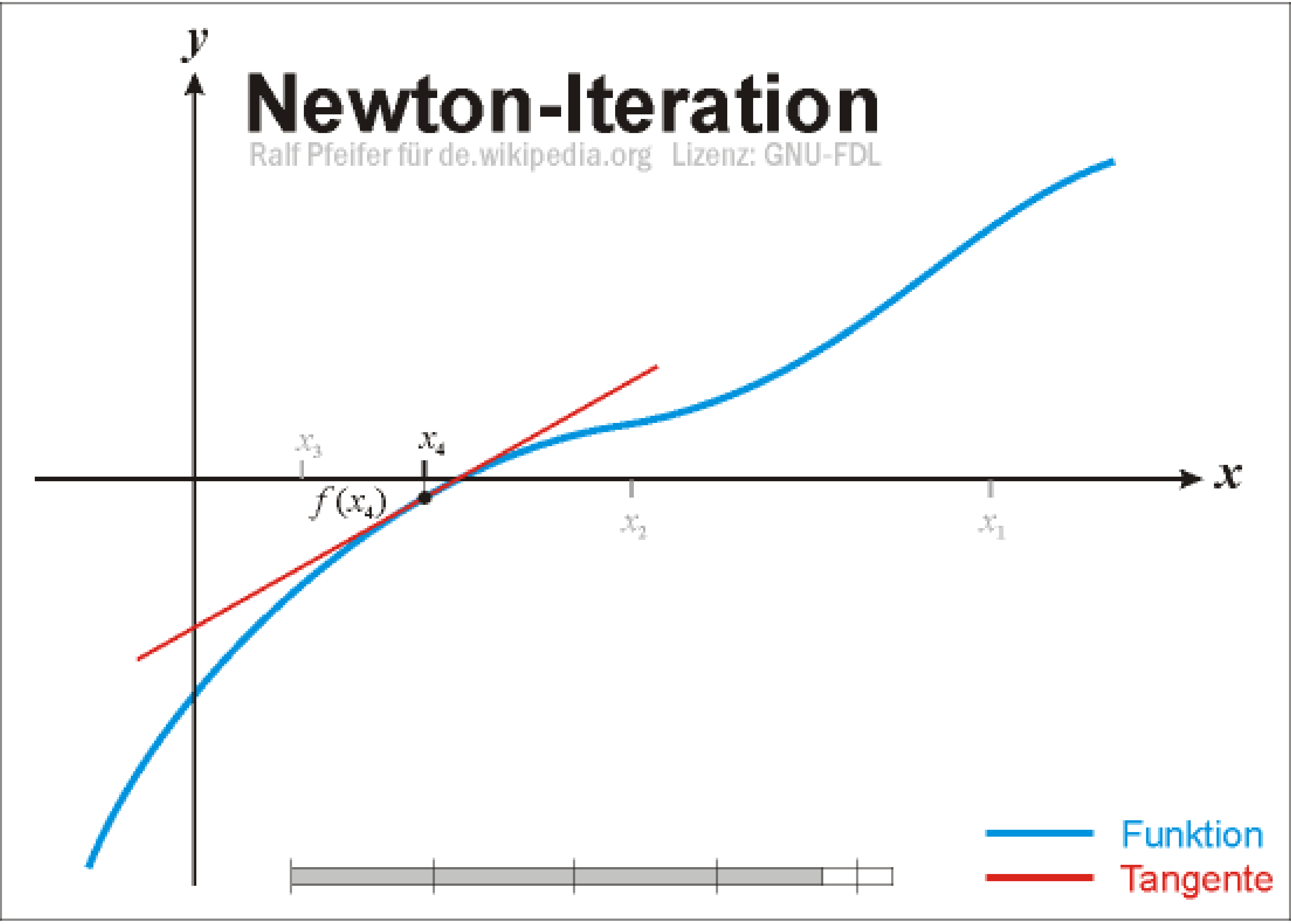


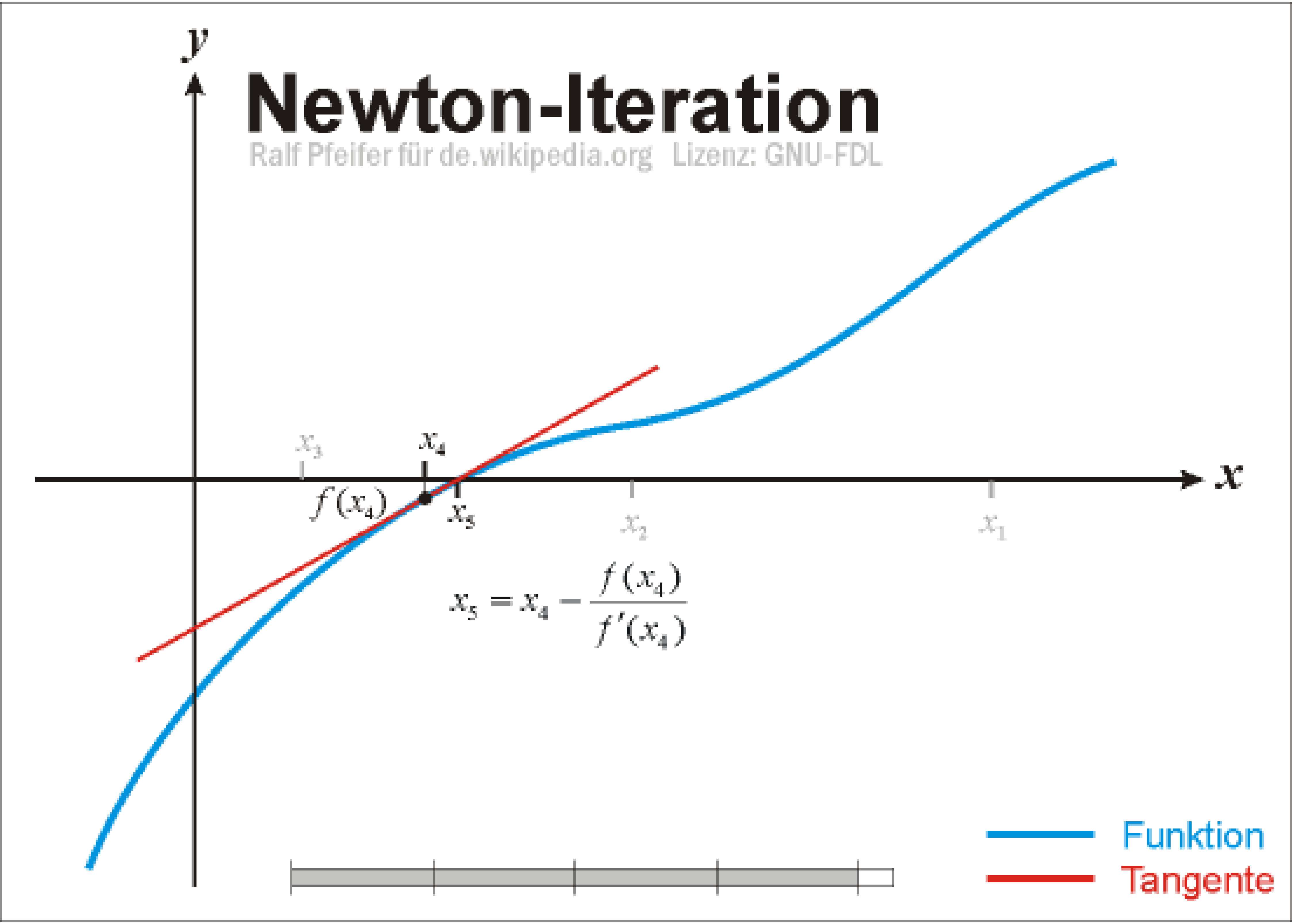
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
$x_4 = x_3 - \frac{f(x_3)}{f'(x_3)}$
x_3
x_4
$f(x_3)$
x_2
x_1
Funktion
Tangente


Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
x3
x4
x2
x1
Funktion
Tangente

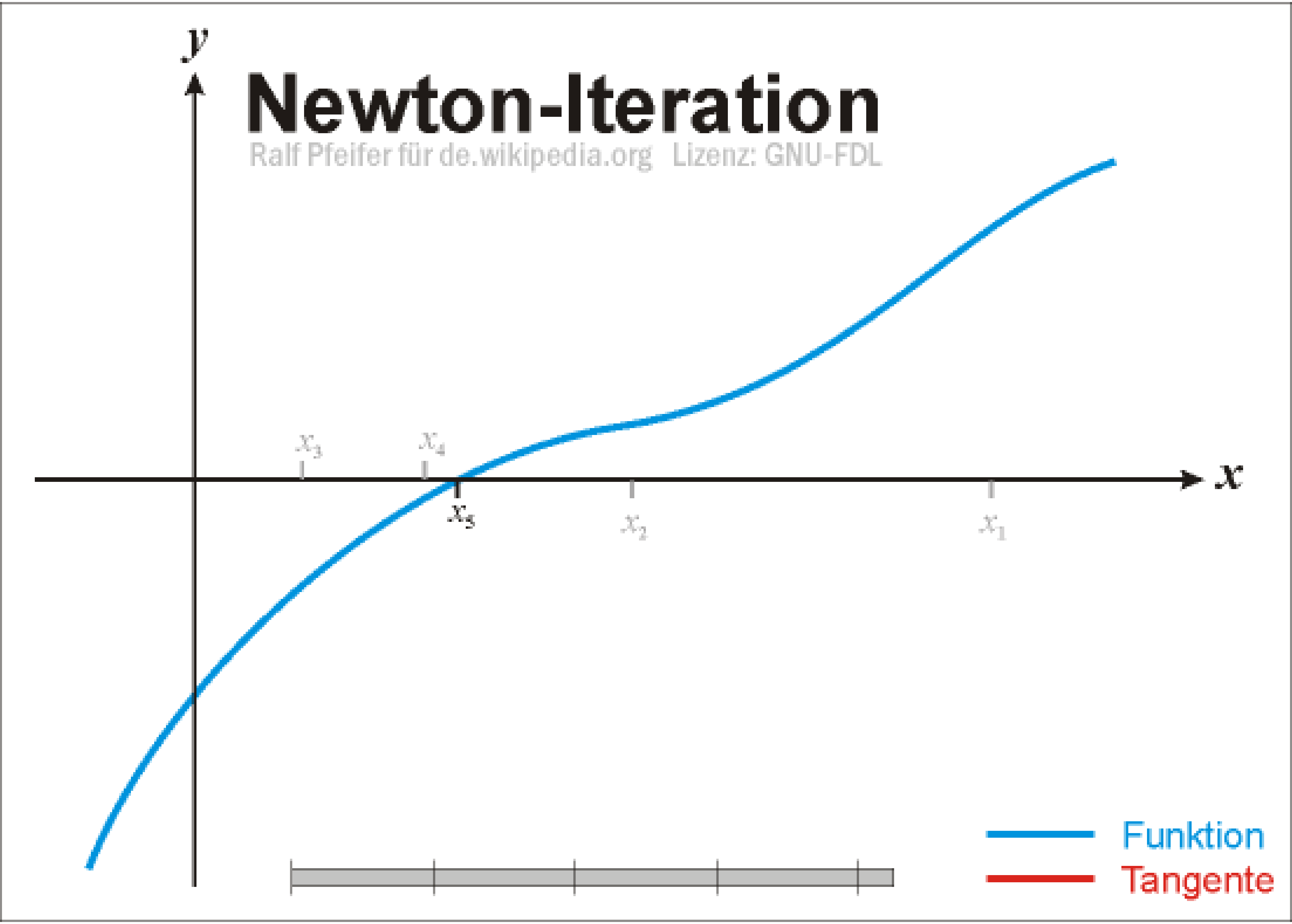


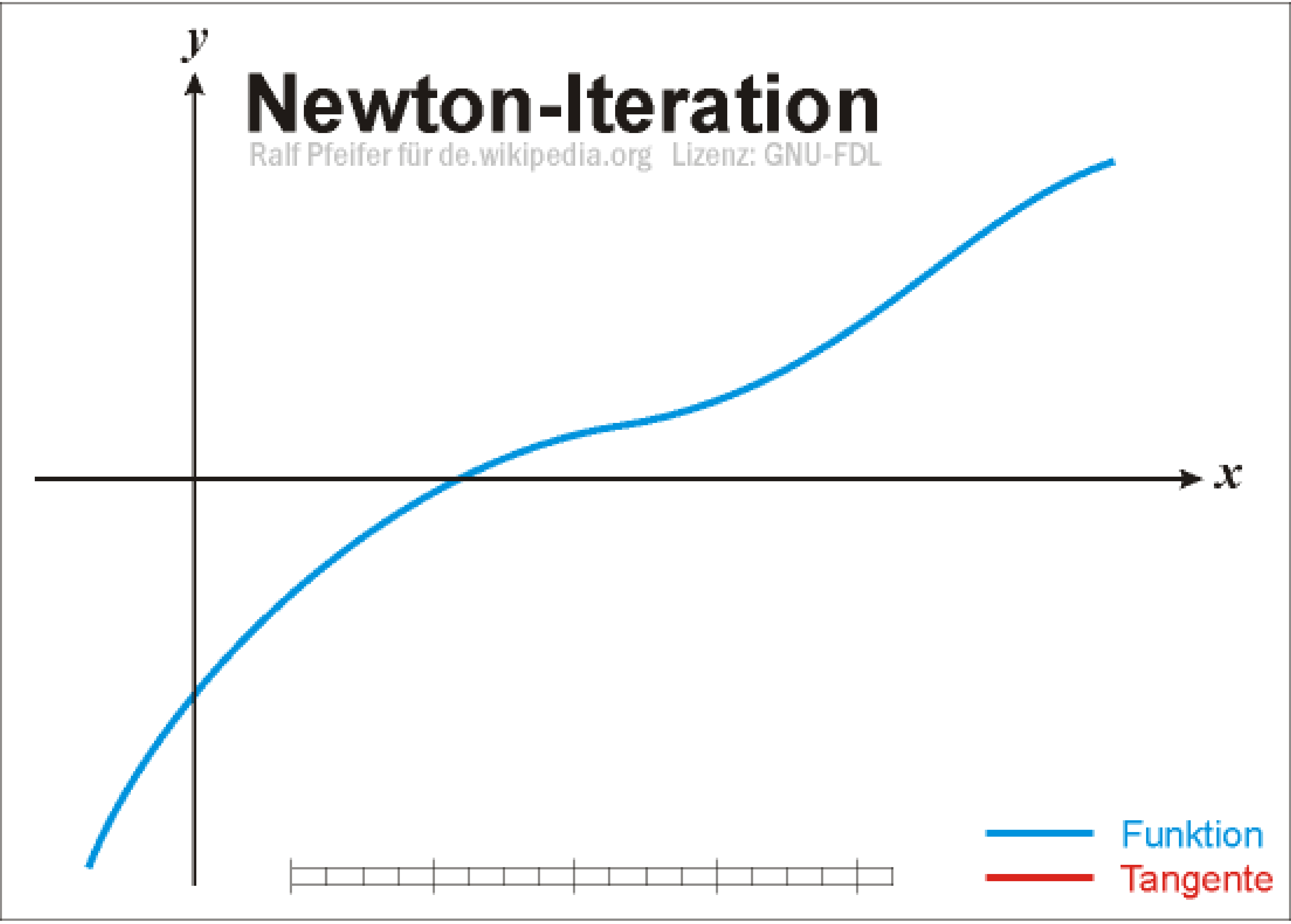
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
x_3
x_4
$f(x_4)$
x_2
x_1
Funktion
Tangente


Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
x_3
x_4
$f(x_4)$
x_2
x_1
Funktion
Tangente

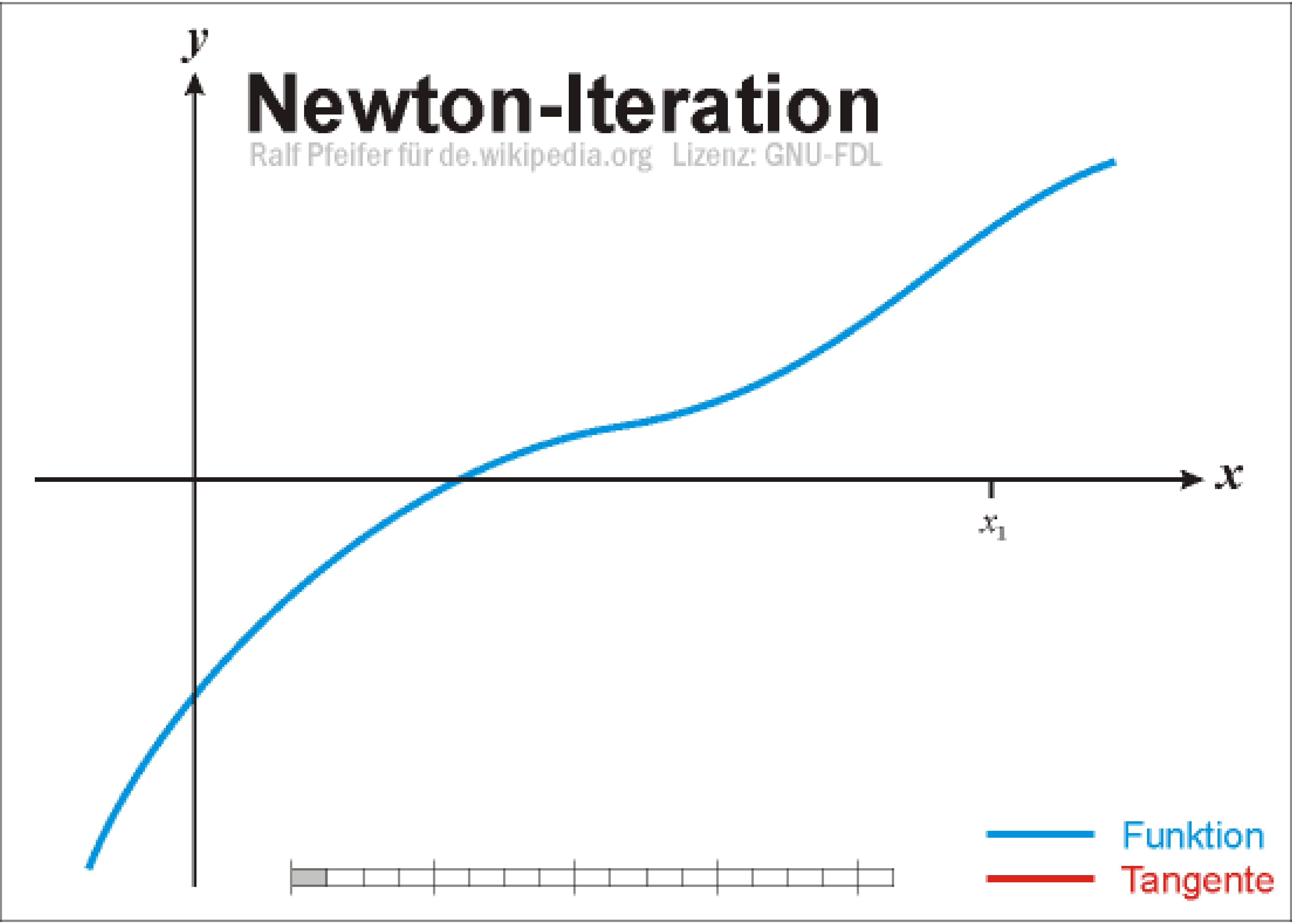


Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
x_3
x_4
$f(x_4)$
x_5
x_2
x_1
$x_5 = x_4 - \frac{f(x_4)}{f'(x_4)}$
Funktion
Tangente

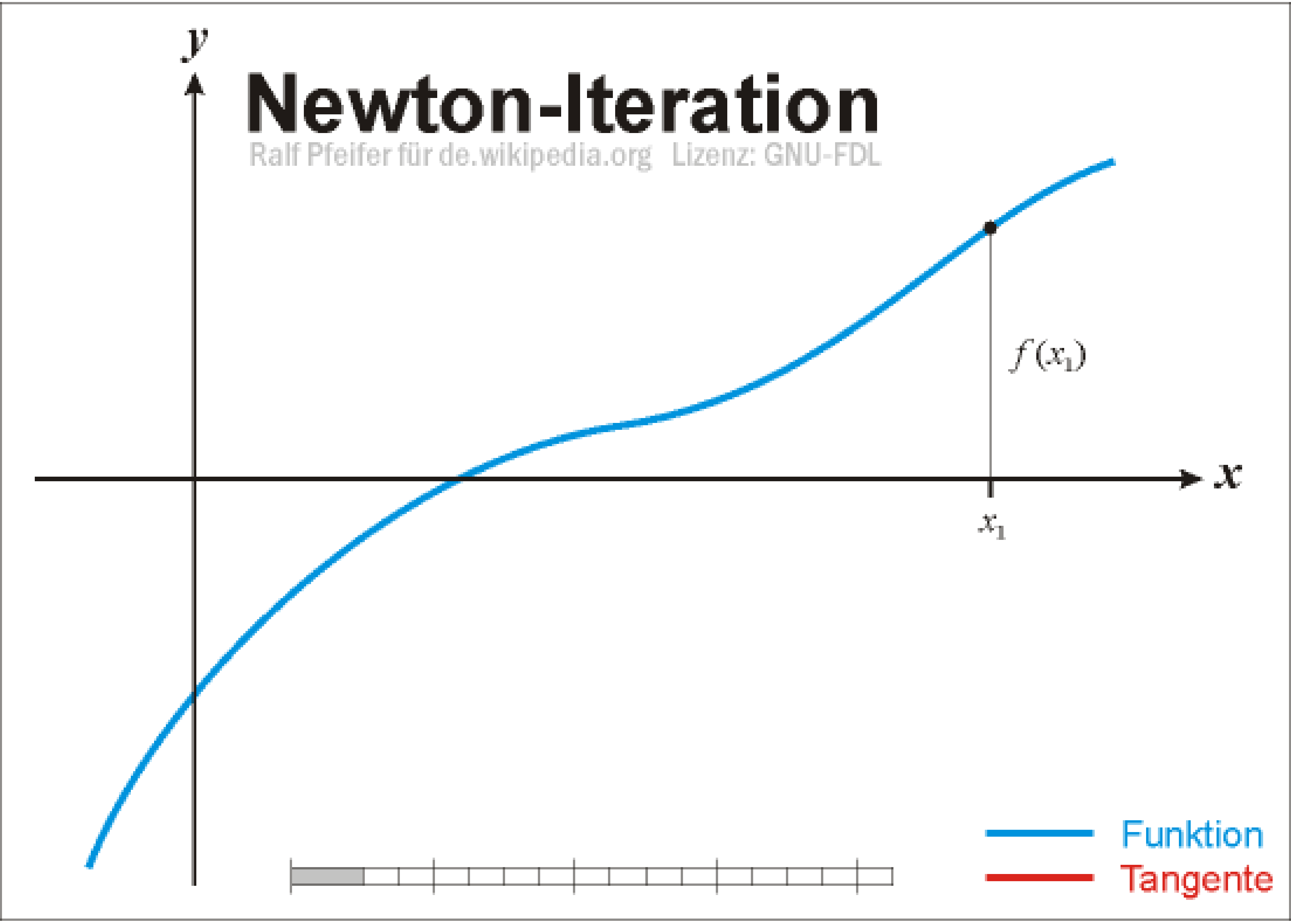

Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
x_3
x_4
x_5
x_2
x_1
Funktion
Tangente



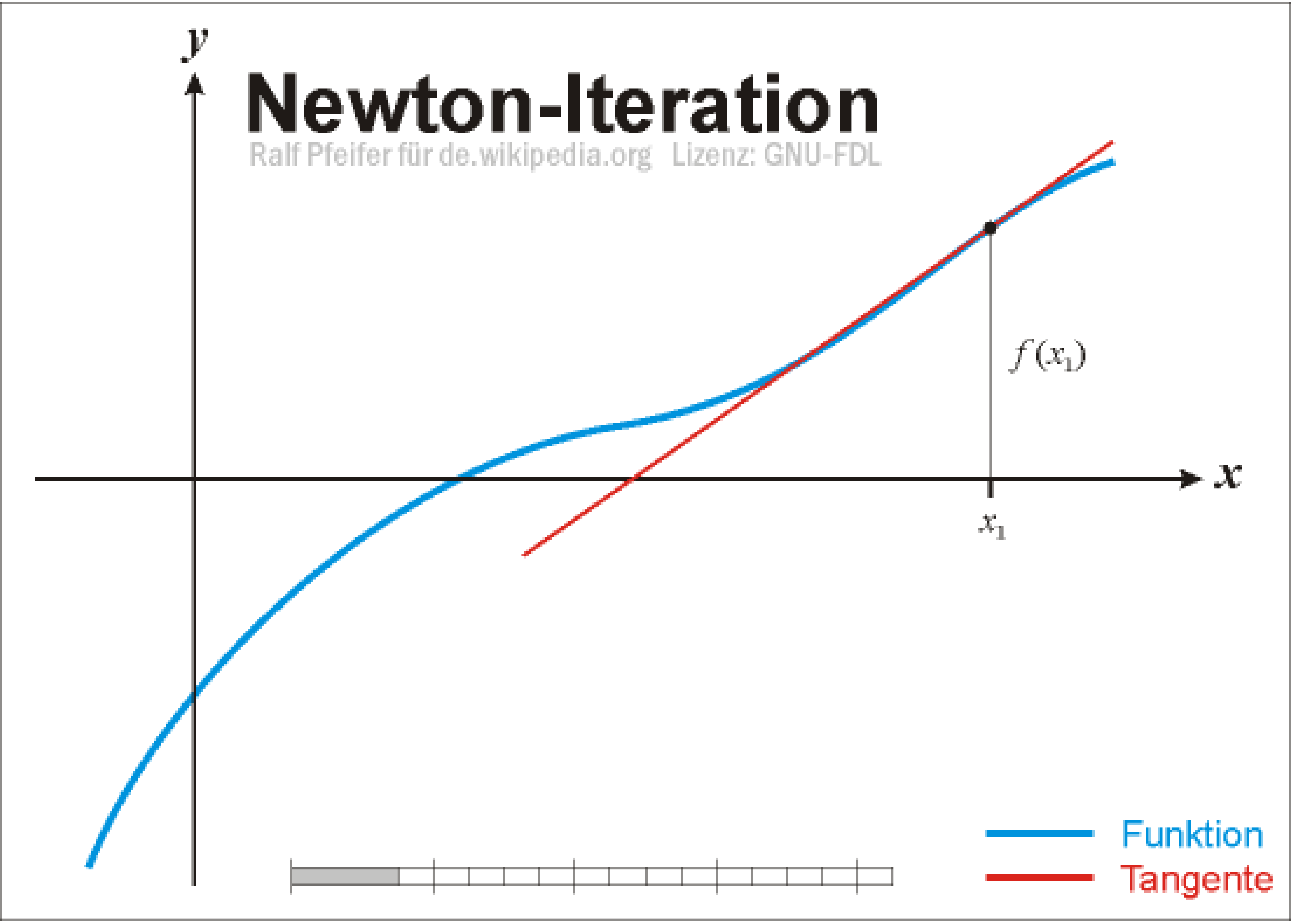
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
Funktion
Tangente



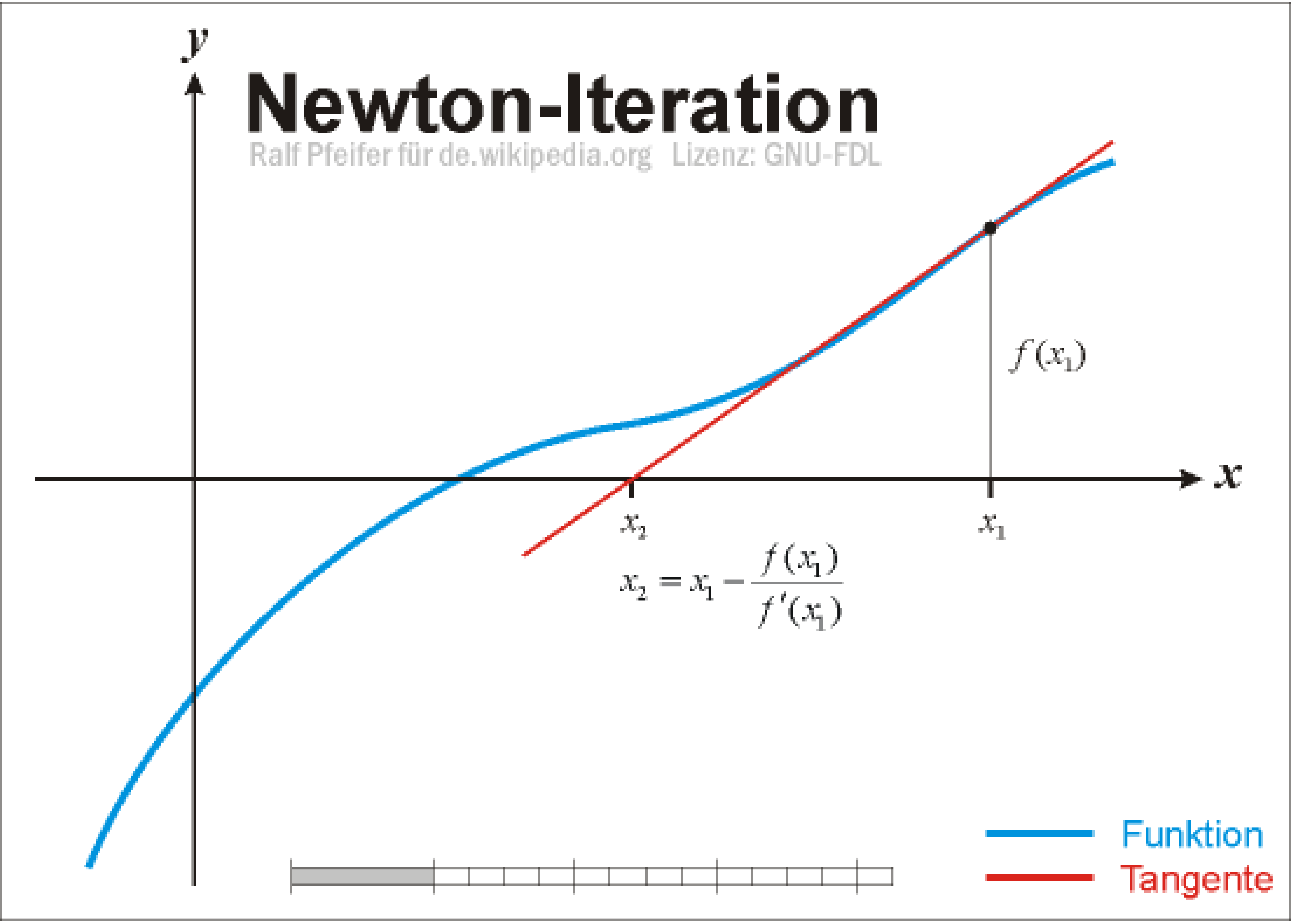
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
x_1
Funktion
Tangente



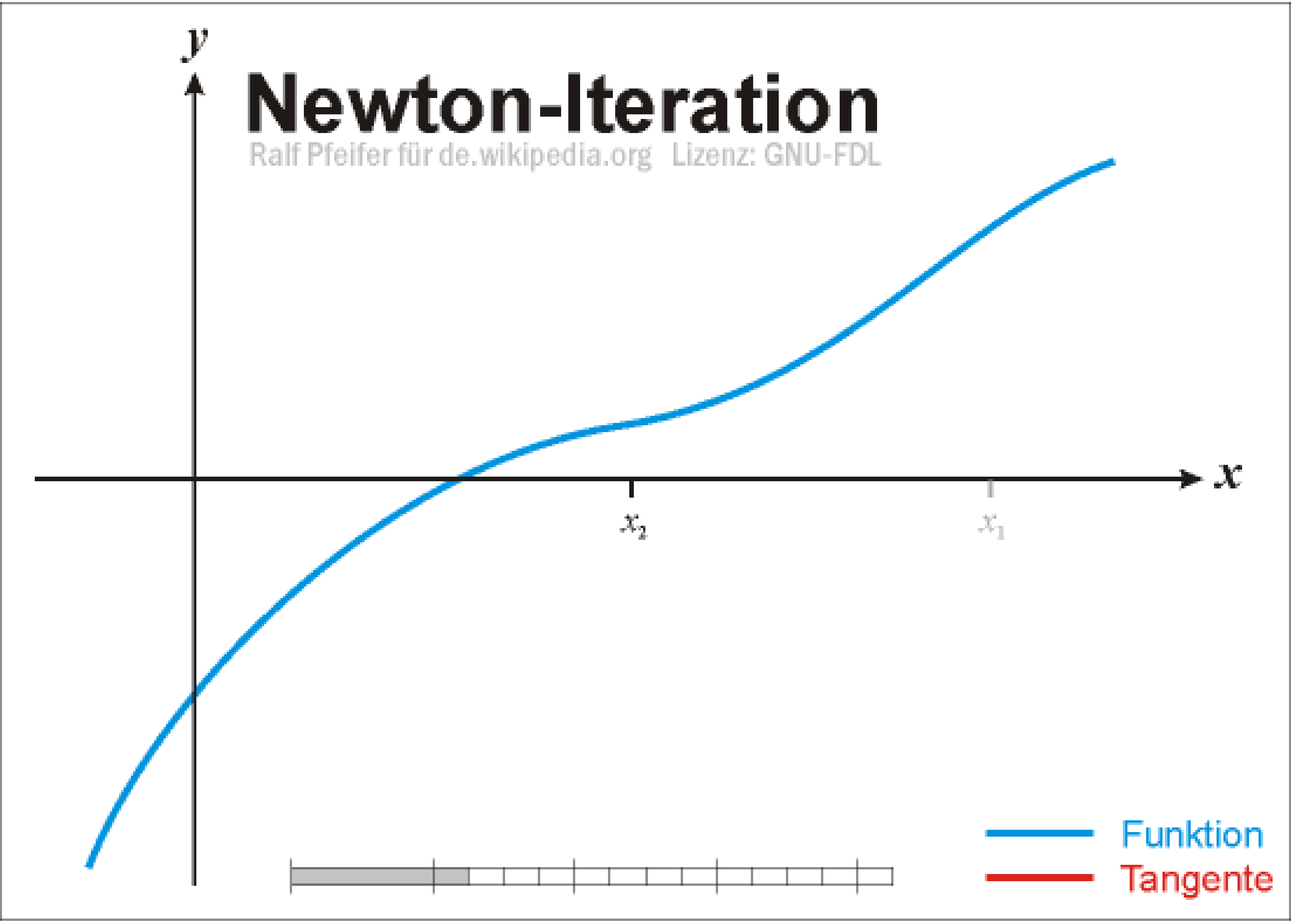
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
f (x1)
x1
Funktion
Tangente



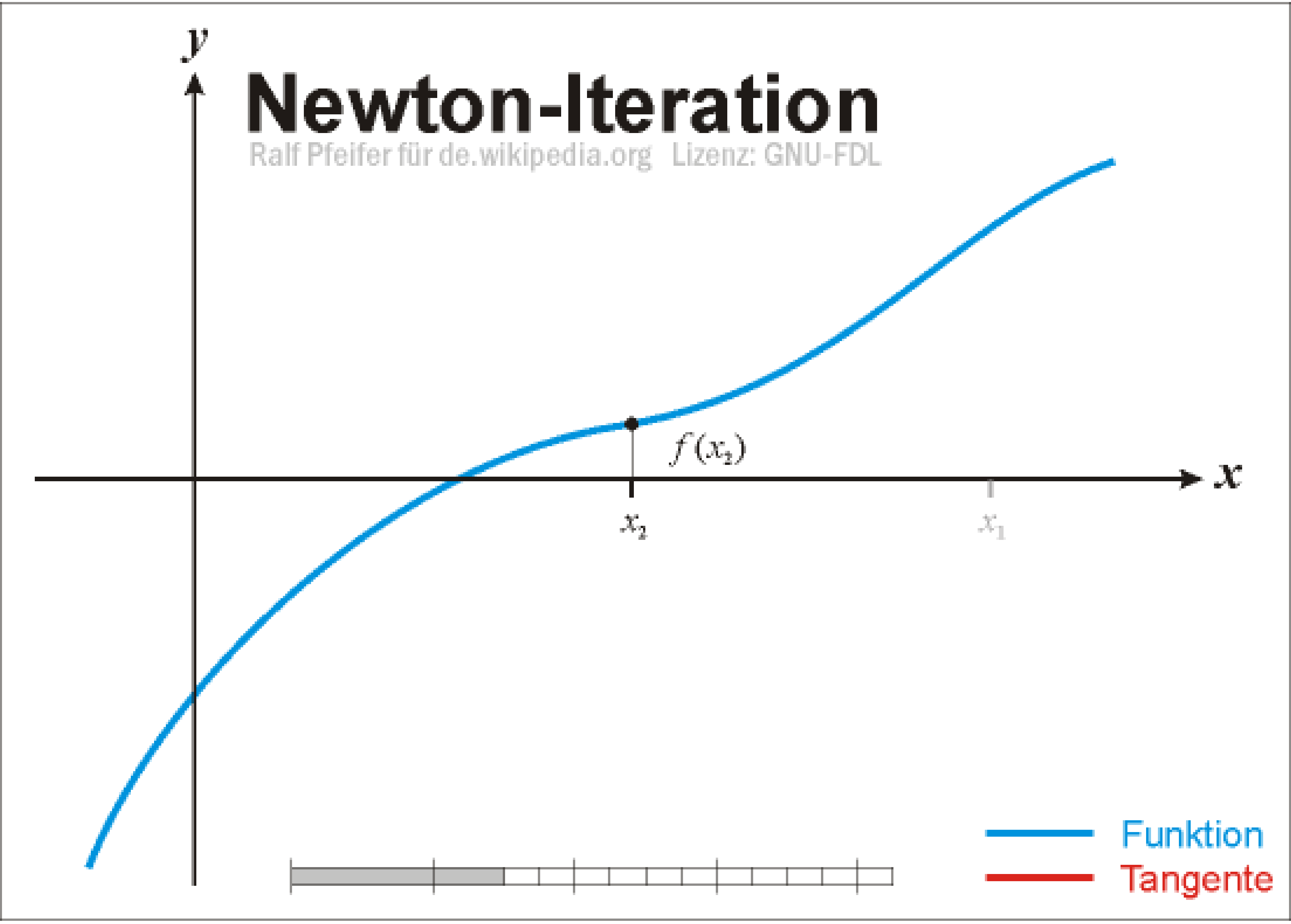
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
$f(x_1)$
x_1
Funktion
Tangente



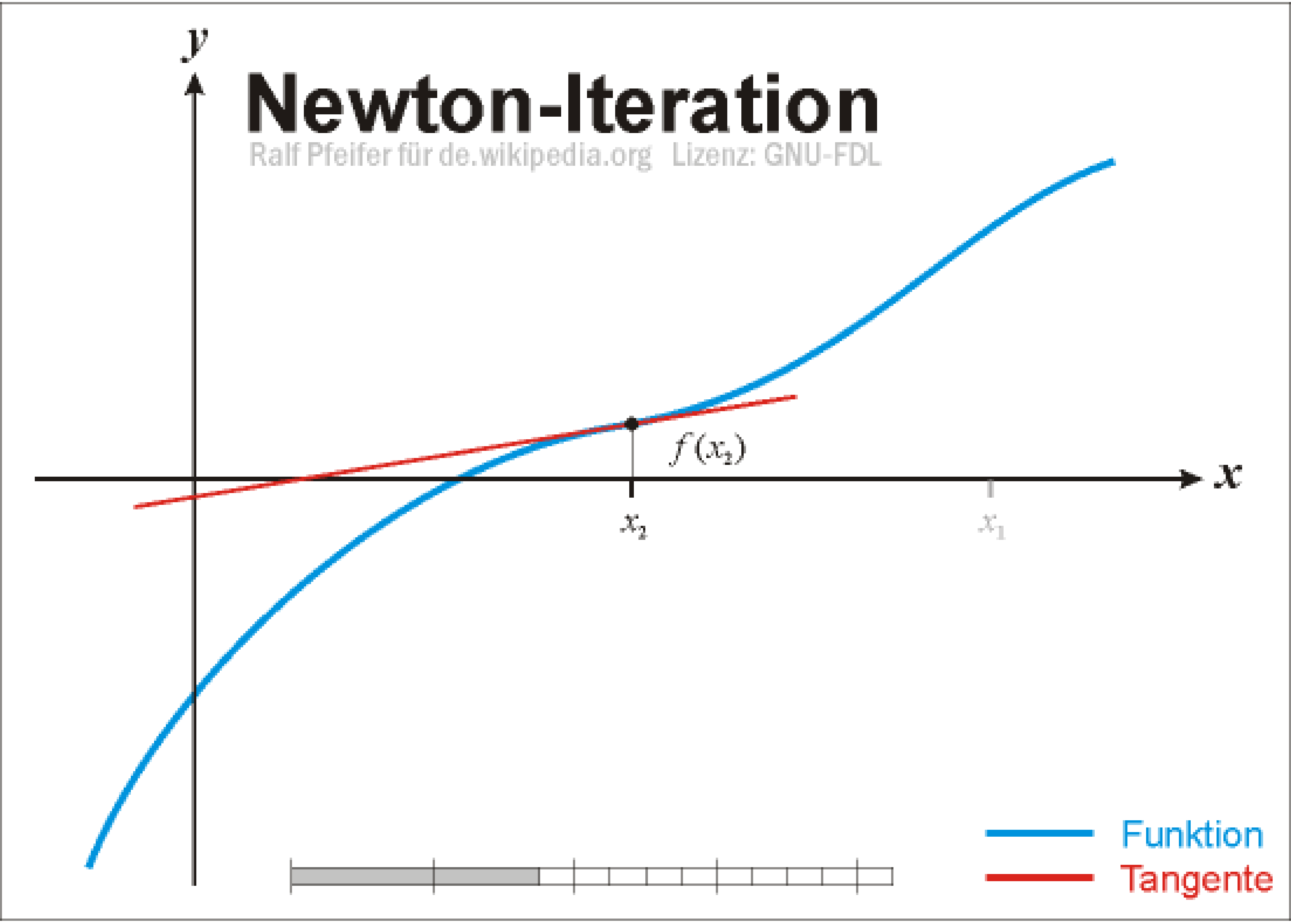
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
$f(x_1)$
x
x_2
x_1
$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$
Funktion
Tangente

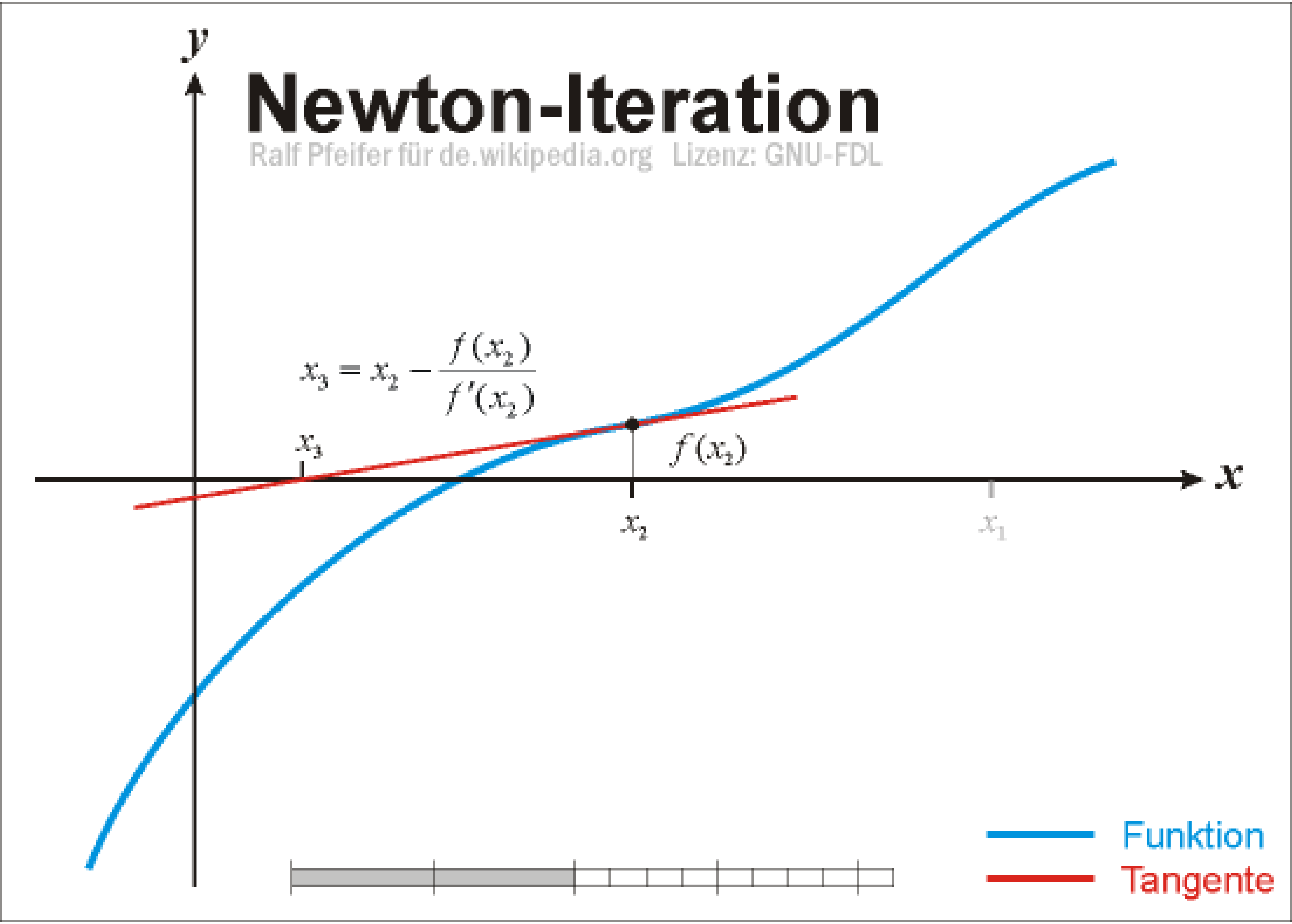


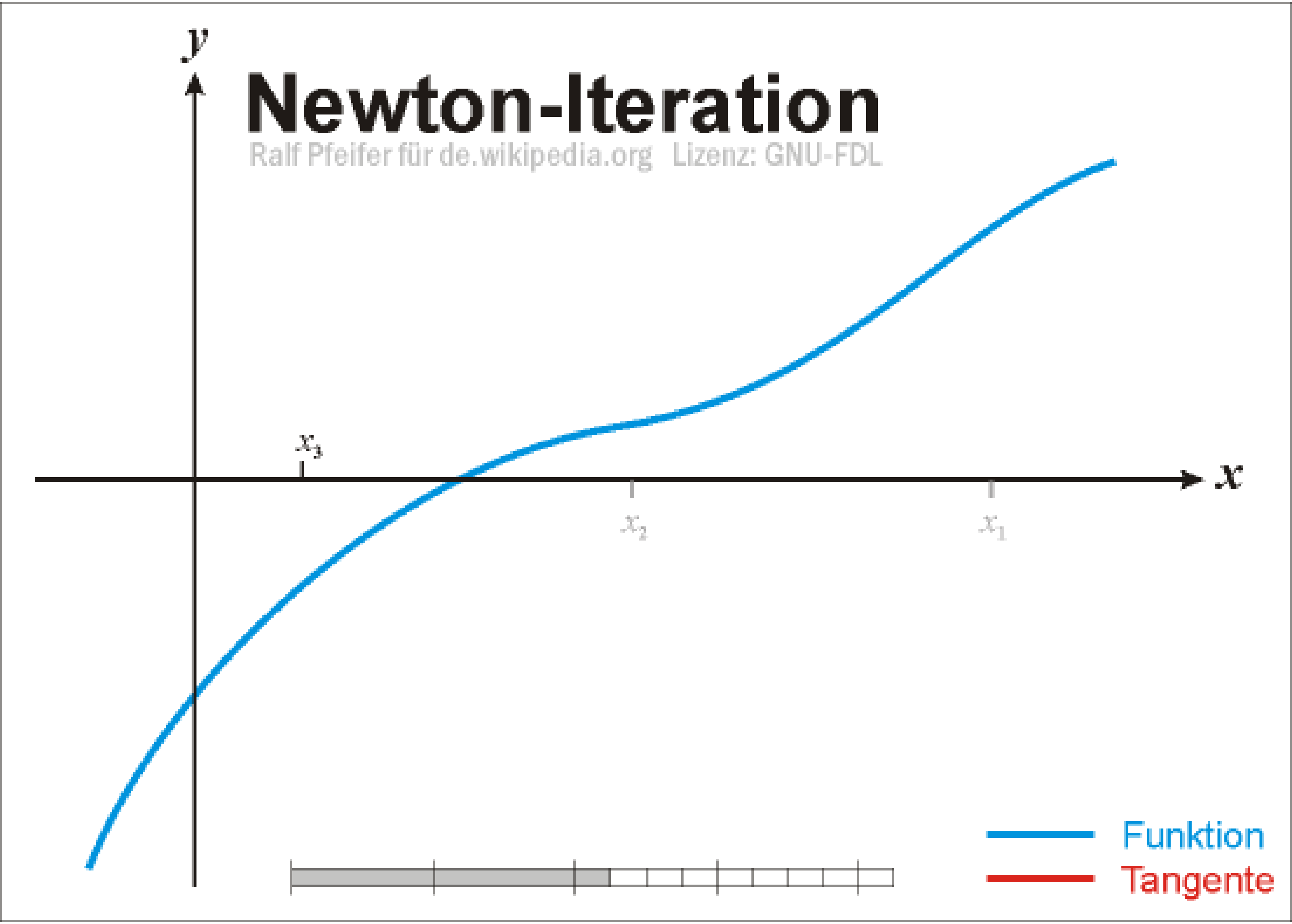
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
x_2
x_1
Funktion
Tangente



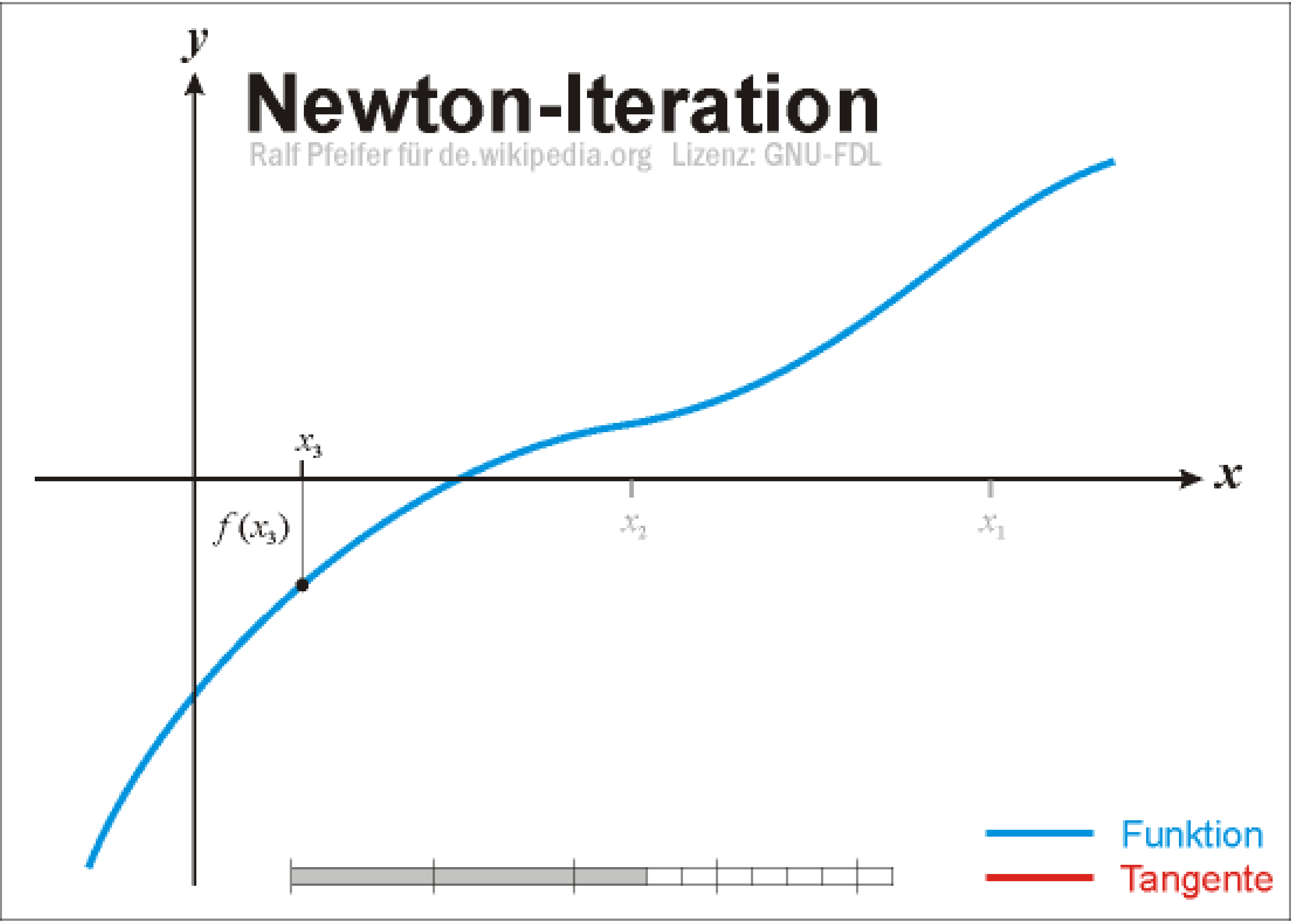
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
$f(x_2)$
x
x_2
x_1
Funktion
Tangente


Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
$f(x_2)$
x_2
x_1
Funktion
Tangente

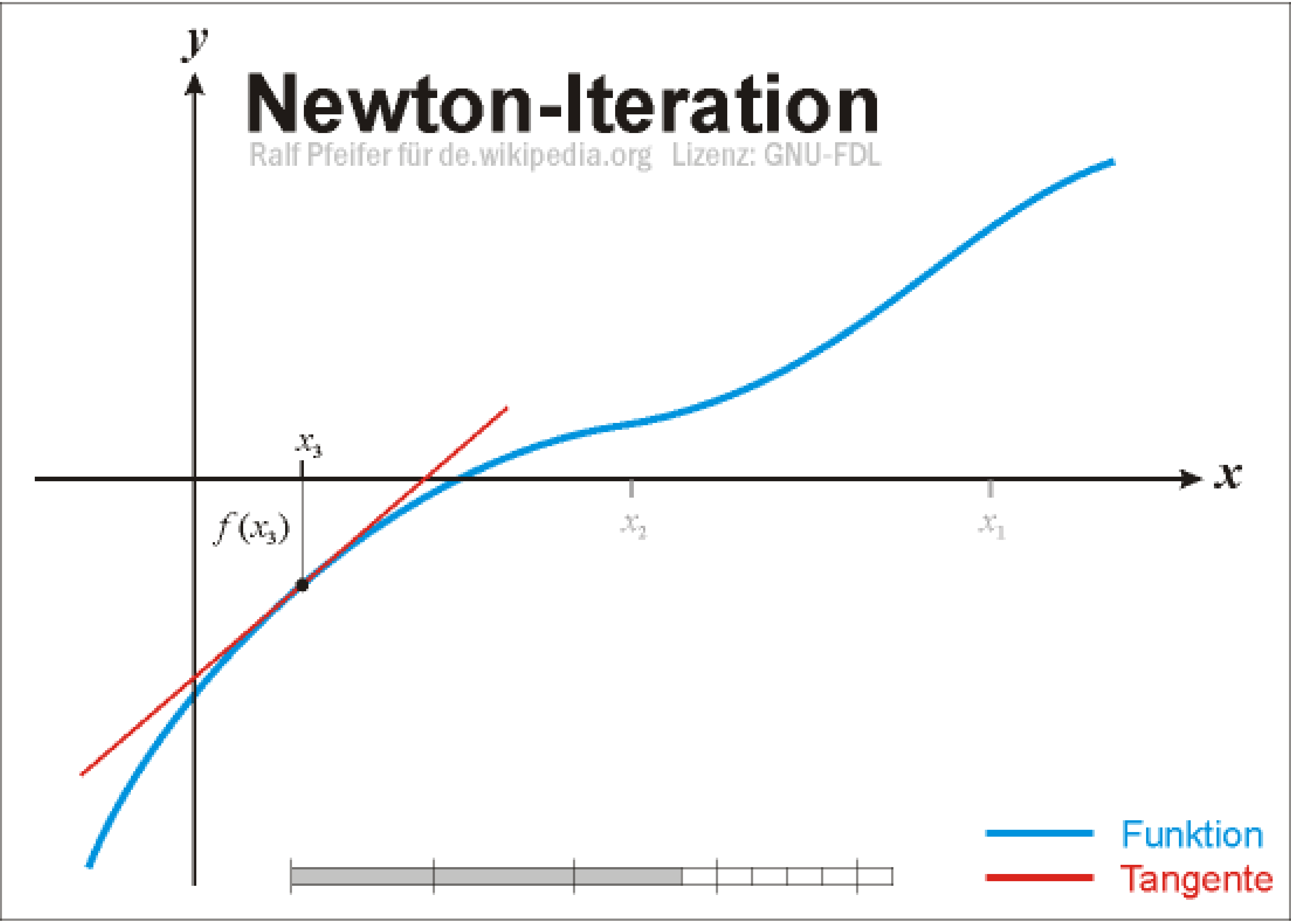

Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
$x_3 = x_2 - \frac{f(x_2)}{f'(x_2)}$
x_3
$f(x_2)$
x_2
x_1
Funktion
Tangente



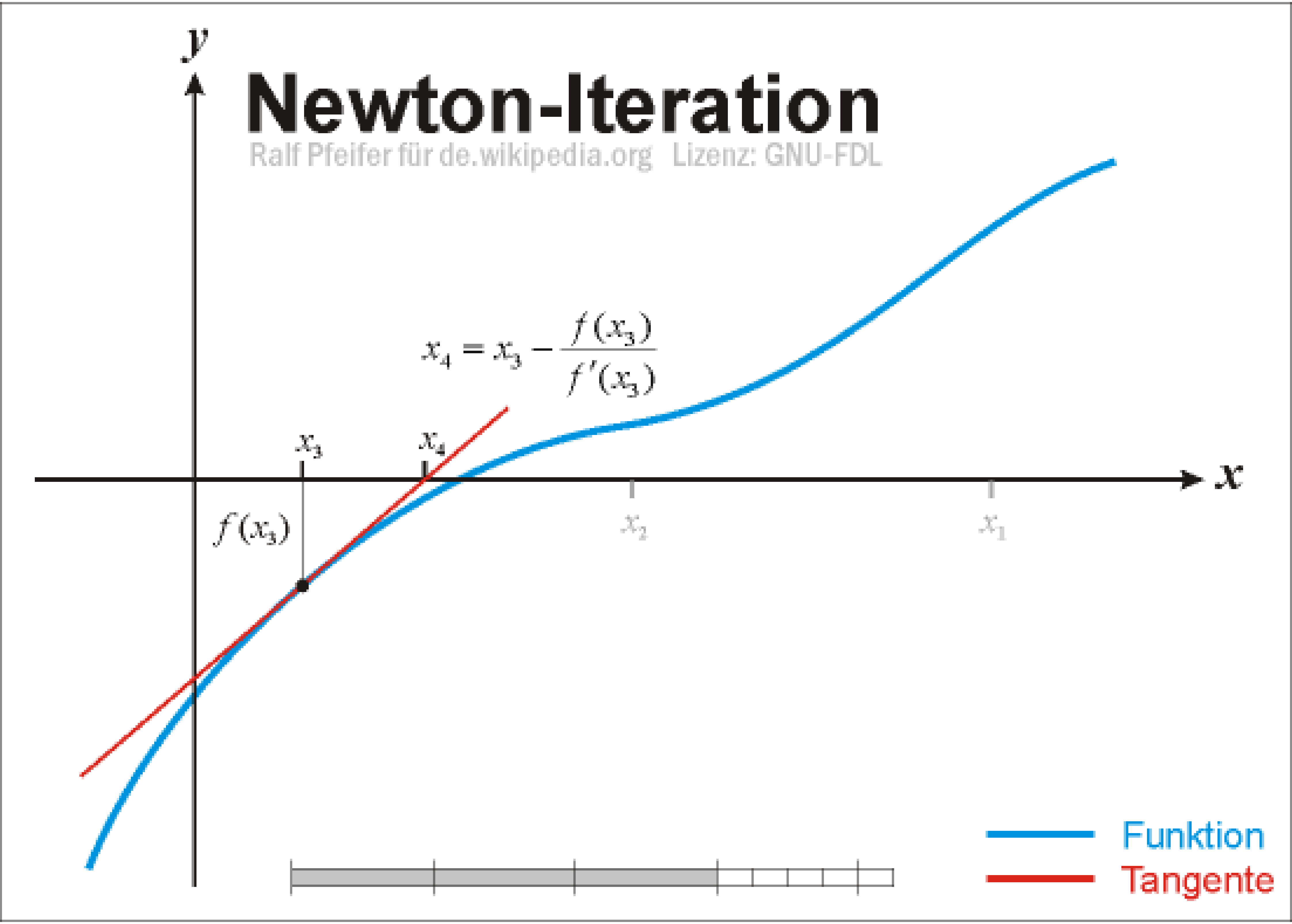
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
x3
x2
x1
Funktion
Tangente



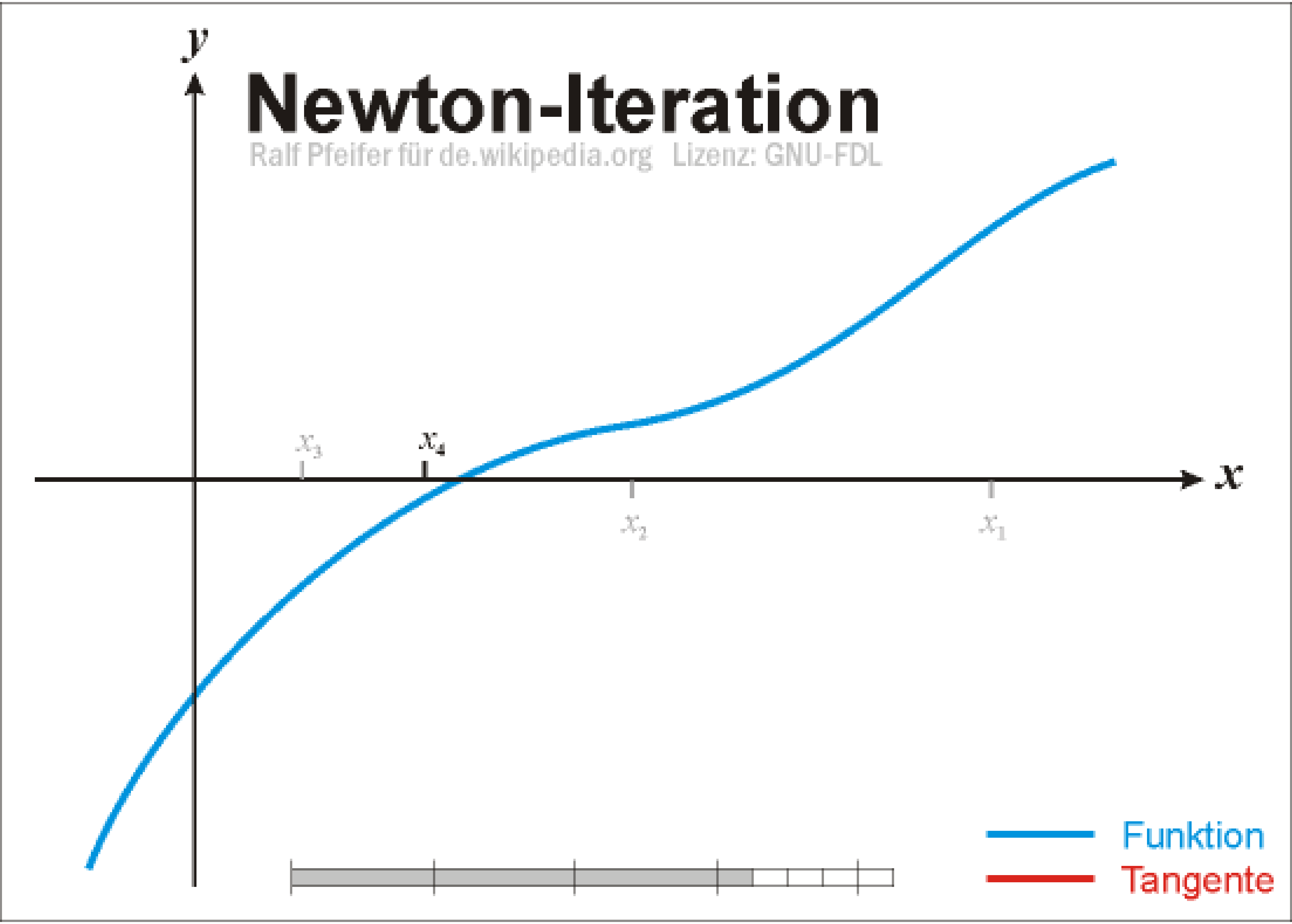
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x_3
x
$f(x_3)$
x_2
x_1
Funktion
Tangente



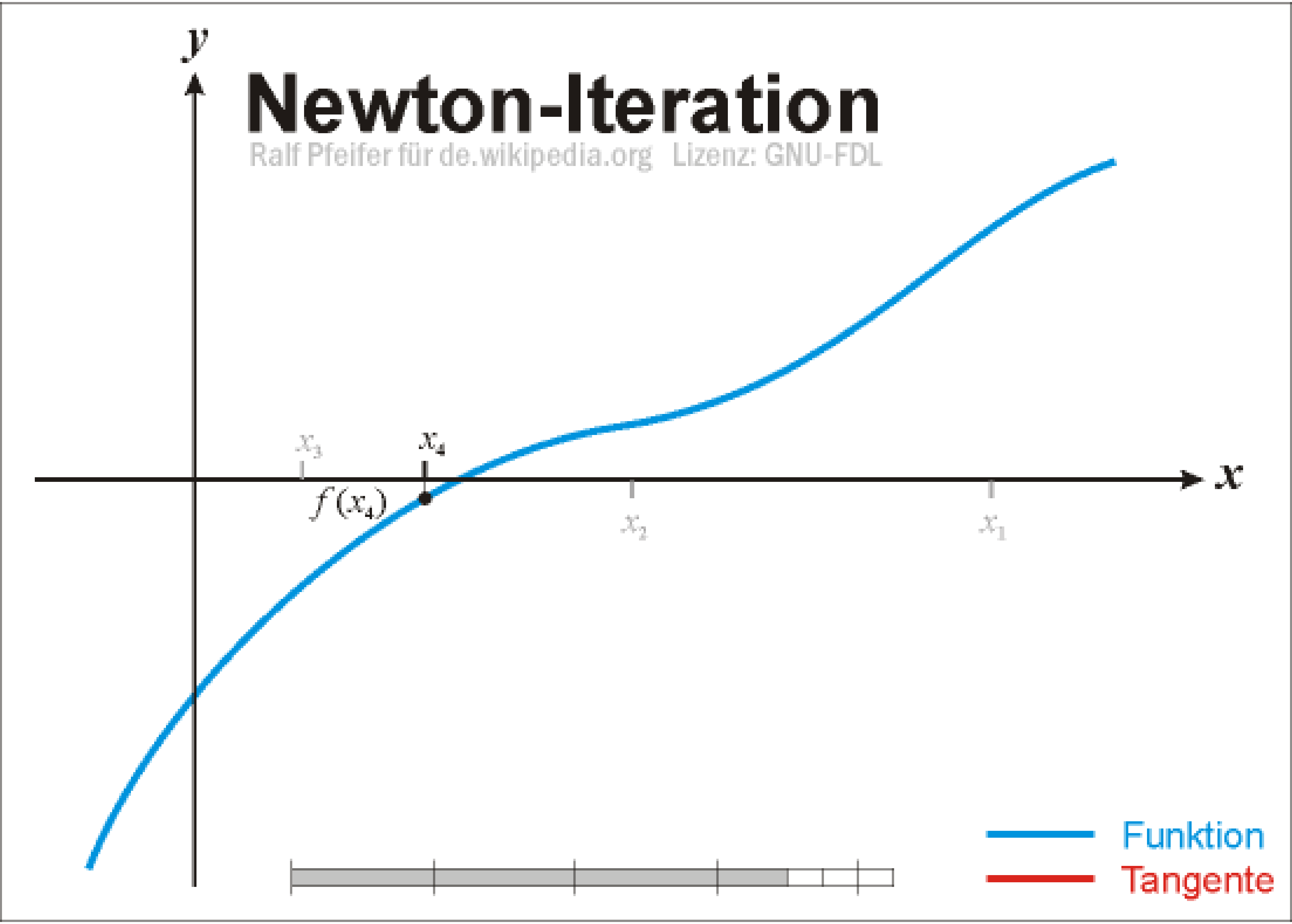
y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
x_3
$f(x_3)$
x_2
x_1
Funktion
Tangente



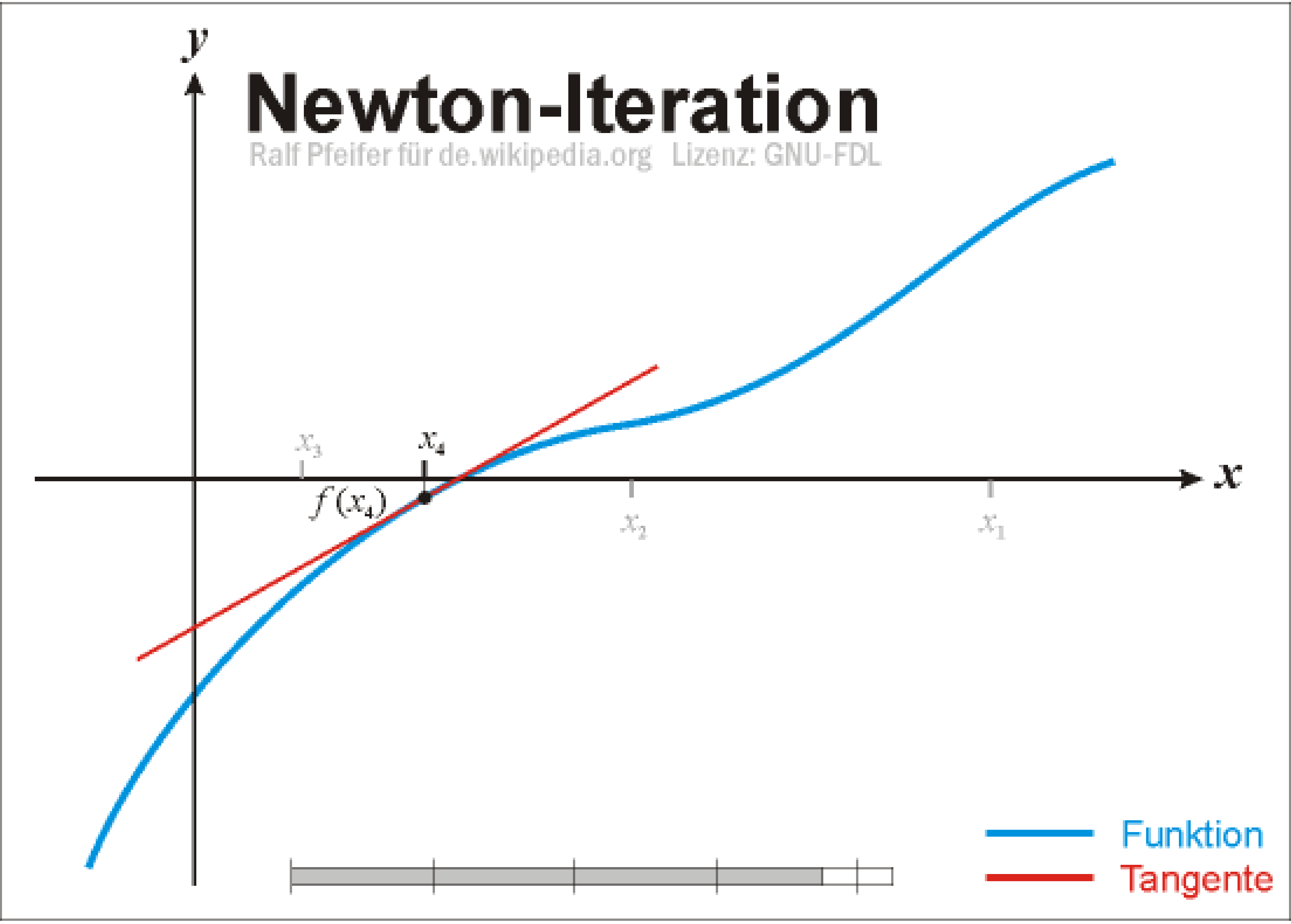
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
$x_4 = x_3 - \frac{f(x_3)}{f'(x_3)}$
x_3
x_4
$f(x_3)$
x_2
x_1
Funktion
Tangente

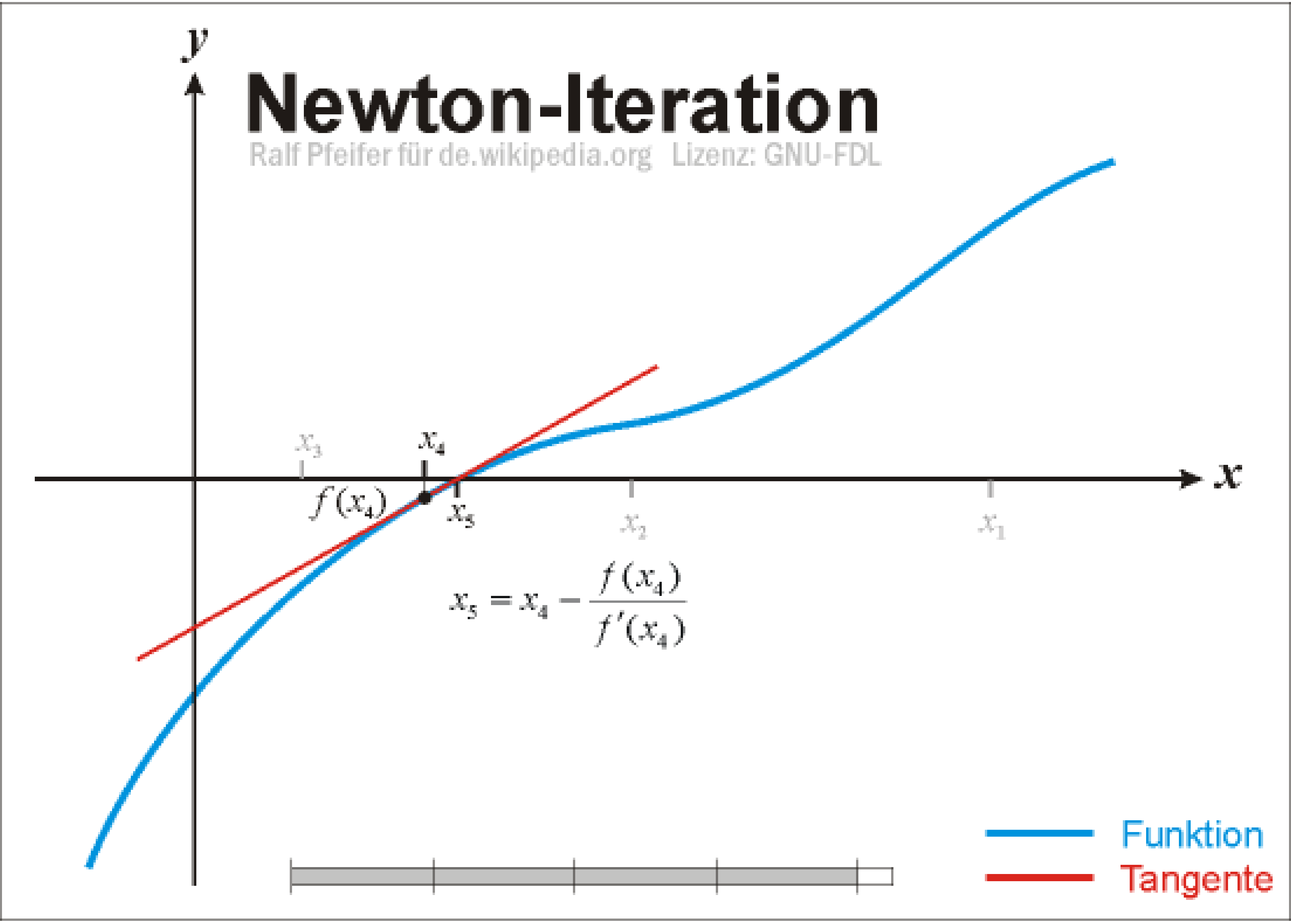


y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
x_3
x_4
x_2
x_1
Funktion
Tangente

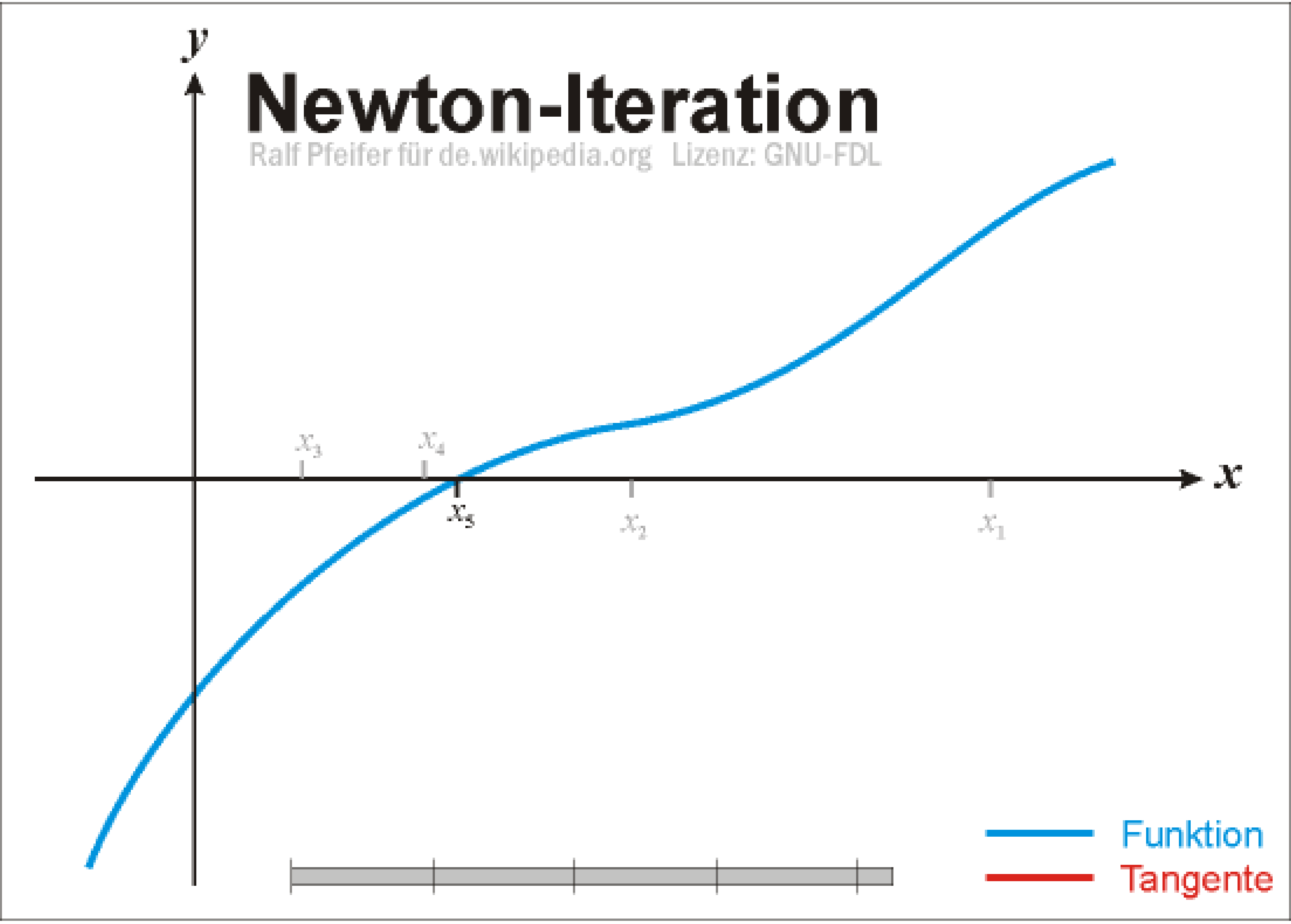


y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
x_3
x_4
$f(x_4)$
x_2
x_1
Funktion
Tangente


Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
y
x
x_3
x_4
$f(x_4)$
x_2
x_1
Funktion
Tangente



y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
x_3
x_4
$f(x_4)$
x_5
x_2
x_1
$x_5 = x_4 - \frac{f(x_4)}{f'(x_4)}$
Funktion
Tangente



y
Newton-Iteration
Ralf Pfeifer für de.wikipedia.org Lizenz: GNU-FDL
x
x_3
x_4
x_5
x_2
x_1
Funktion
Tangente