

Materiały wykładowe (fragmenty)

Robert Susmaga

Instytut Informatyki

ul. Piotrowo 2

Poznań

kontakt mail'owy

Robert.Susmaga@CS.PUT.Poznan.PL

kontakt osobisty

Centrum Wykładowe, „blok informatyki”, pok. 7

Techniki optymalizacji

Cz. 1

Wyłączenie odpowiedzialności

Prezentowane materiały, będące dodatkiem pomocniczym do wykładów, z konieczności fragmentarycznym i niedopracowanym, należy wykorzystywać z pełną świadomością faktu, że mogą nie być pozbawione przypadkowych błędów, braków, wypaczeń i przeinaczeń :-)

Autor

...

Gradient i hesjan

- Definicja gradientu ∇_f funkcji $f(\mathbf{x})$ dla $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$
 - gradient (wektor o rozmiarze $n \times 1$)

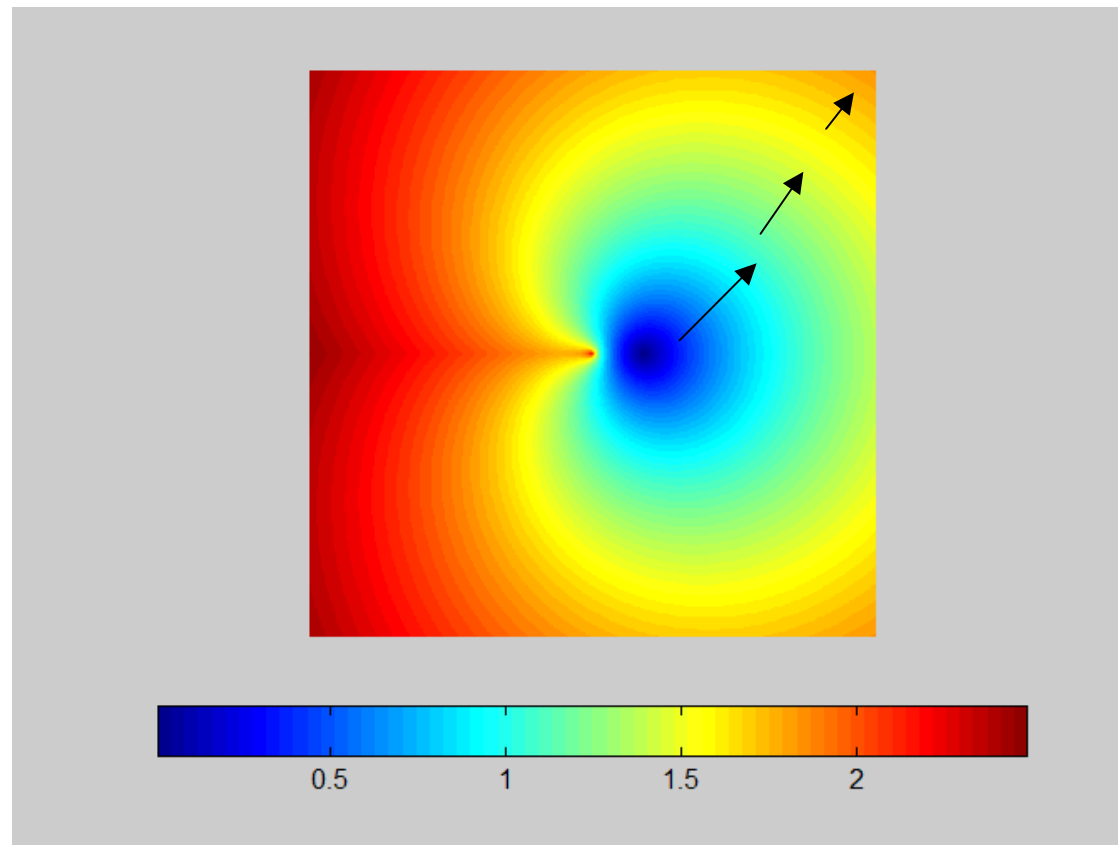
$$\nabla_f = \begin{bmatrix} \partial f / \partial x_1 \\ \partial f / \partial x_2 \\ \dots \\ \partial f / \partial x_n \end{bmatrix}$$

(istnieje, gdy $f(\mathbf{x})$ posiada wszystkie pierwsze pochodne)

Gradient i hesjan

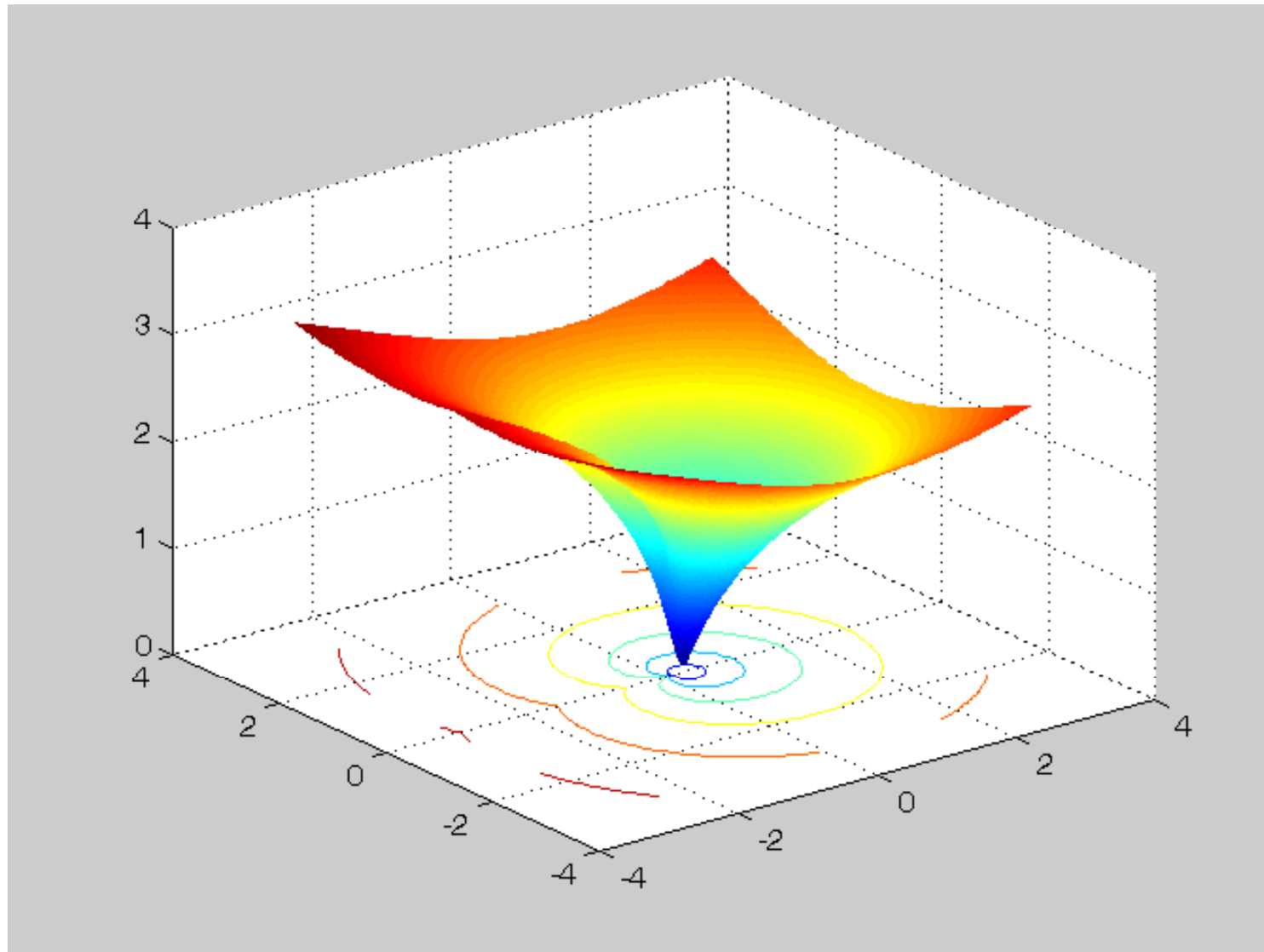
- Cechy gradientu ∇_f funkcji $f(\mathbf{x})$ dla $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$
 - gradient ∇_f funkcji $f(\mathbf{x})$ jest w ogólności wektorem pierwszych pochodnych funkcji $f(\mathbf{x})$ (po odpowiednich elementach wektora \mathbf{x})
 - gradient $\nabla_f(\mathbf{y})$ funkcji $f(\mathbf{x})$ od wektora \mathbf{y} jest wektorem wartości pierwszych pochodnych funkcji $f(\mathbf{x})$ obliczonych dla wektora \mathbf{y}
 - wektor $\nabla_f(\mathbf{y})$ istnieje, gdy ∇_f istnieje i jest określony dla wektora \mathbf{y}
 - interpretacja graficzna: gradient funkcji w punkcie (o ile istnieje i jest niezerowy) jest wektorem wskazującym kierunek, w którym wzrost wartości tej funkcji w otoczeniu tego punktu jest maksymalny
 - w punktach ekstremalnych (ale nie tylko) gradient jest wektorem zerowym (dlatego jest naturalnie wykorzystywany w warunkach stopu)

Gradient i hesjan

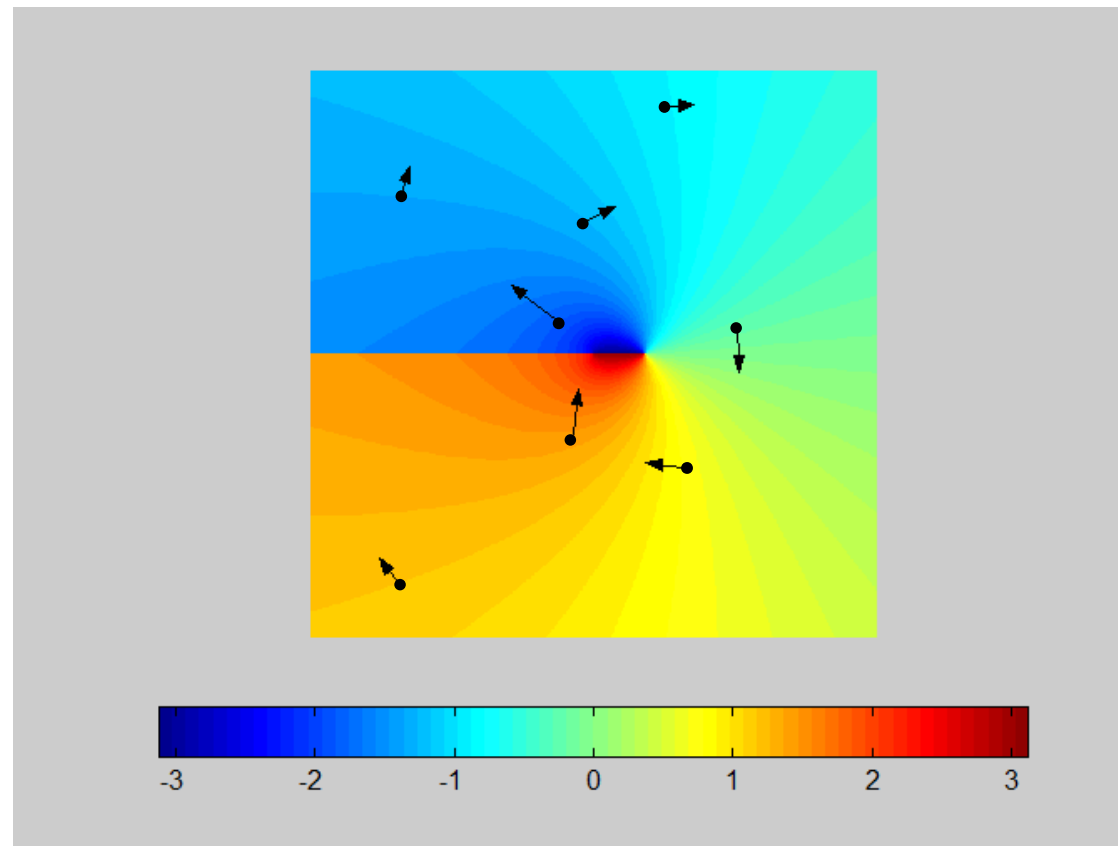


Graficzna interpretacja gradientu

Gradient i hesjan

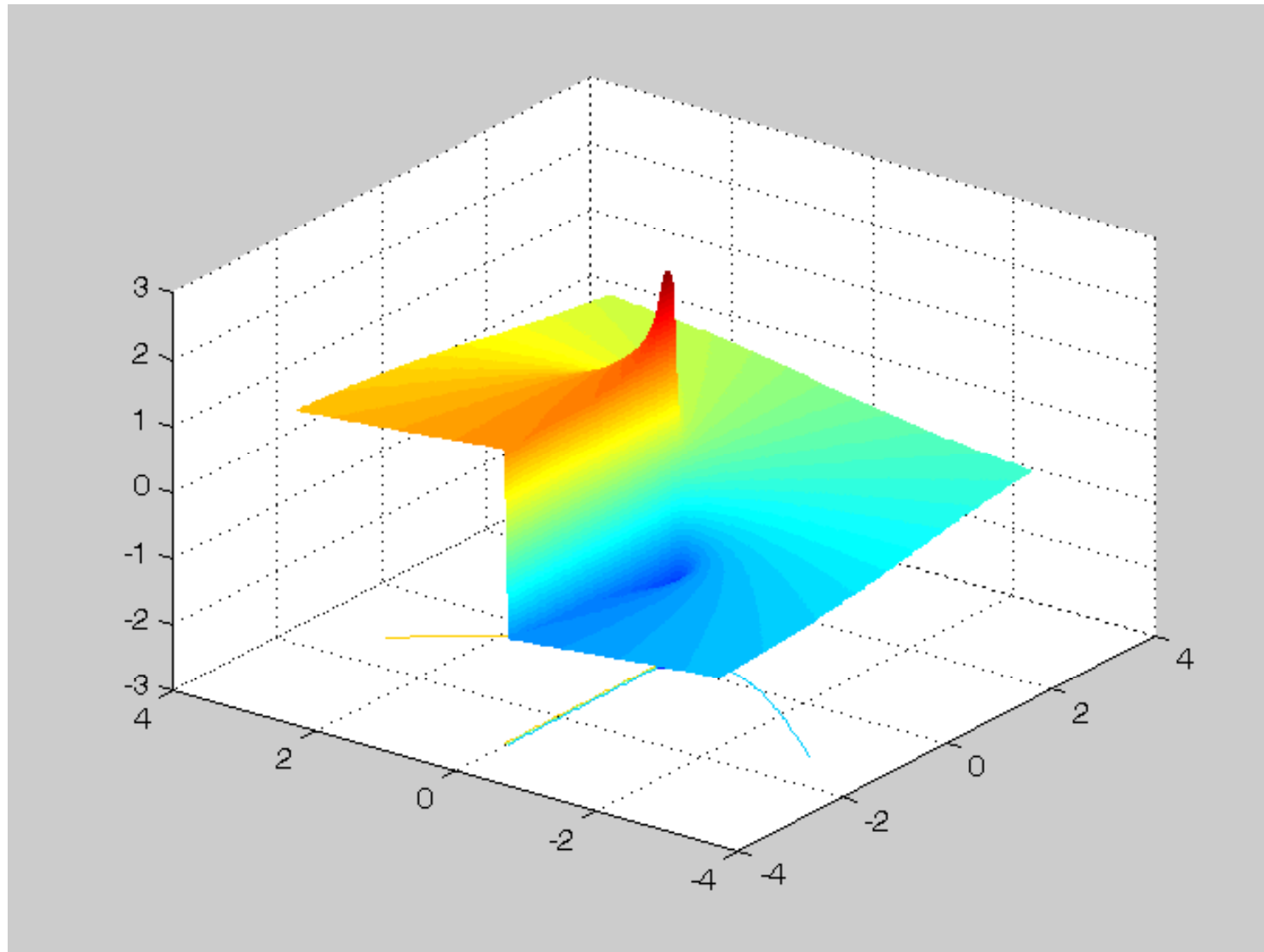


Gradient i hesjan



Graficzna interpretacja gradientu

Gradient i hesjan



Gradient i hesjan

- Definicja hesjanu \mathbf{H}_f funkcji $f(\mathbf{x})$ dla $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$
 - hesjan (macierz o rozmiarach $n \times n$)

$$\mathbf{H}_f = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2 \partial x_2} & \dots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 f}{\partial x_n \partial x_n} \end{bmatrix}$$

(istnieje, gdy $f(\mathbf{x})$ posiada wszystkie drugie pochodne)

Gradient i hesjan

- Cechy hesjanu \mathbf{H}_f funkcji $f(\mathbf{x})$ dla $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$
 - hesjan \mathbf{H}_f funkcji $f(\mathbf{x})$ jest w ogólności macierzą drugich pochodnych funkcji $f(\mathbf{x})$ (po odpowiednich parach elementów wektora \mathbf{x})
 - hesjan $\mathbf{H}_f(\mathbf{w})$ funkcji $f(\mathbf{x})$ od wektora \mathbf{w} jest macierzą wartości drugich pochodnych funkcji $f(\mathbf{x})$ obliczonych dla wektora \mathbf{w}
 - macierz $\mathbf{H}_f(\mathbf{w})$ istnieje, gdy \mathbf{H}_f istnieje i jest określony dla wektora \mathbf{w}
 - uwaga 1: interpretacja graficzna hesjanu jest bardziej złożona niż interpretacja graficzna gradientu, ponieważ gradient jest wektorem (który posiada naturalną interpretację graficzną w postaci strzałki), podczas gdy hesjan jest macierzą (wiele strzałek?)
 - uwaga 2: hesjan jest jacobianem gradientu ($\mathbf{H}_f = \mathbf{J}(\nabla_f)$)

Gradient i hesjan

- Pewne właściwości hesjanu \mathbf{H}_f funkcji $f(\mathbf{x})$
 - ponieważ dla wszystkich i oraz j zachodzi $\partial^2 f / (\partial x_i \partial x_j) = \partial^2 f / (\partial x_j \partial x_i)$, więc hesjan $\mathbf{H}_f(\mathbf{w})$ dla każdego wektora \mathbf{w} jest macierzą symetryczną
 - tw. Schwarz'a (jeżeli pochodne istnieją i są ciągłe, to są sobie równe)
 - jeżeli dwukrotnie różniczkowalna funkcja $f(\mathbf{x})$ jest wypukła w obszarze S , to jej hesjan $\mathbf{H}_f(\mathbf{s})$ jest dla każdego wektora \mathbf{s} z obszaru S macierzą nieujemnie określoną (prawdziwe jest także stwierdzenie odwrotne)
 - oznaczenie $\mathbf{H}_f(\mathbf{s}) \geq 0$

...

Ekstrema (minima/maksima) lokalne

Warunek wystarczający istnienia ekstremum. Jeżeli funkcja $f(x, y)$ jest klasy C^2 w pewnym otoczeniu punktu $P_0(x_0, y_0)$, a ponadto:

$$1^\circ \quad f_x(P_0) = 0 \quad \text{i} \quad f_y(P_0) = 0 \quad (\text{LEI})$$

$$2^\circ \quad W(P_0) \equiv f_{xx}(P_0) \cdot f_{yy}(P_0) - [f_{xy}(P_0)]^2 > 0 \quad (\text{LEI})$$

to funkcja $f(x, y)$ ma w punkcie P_0 *maksimum właściwe*, gdy $f_{xx}(P_0) < 0$, natomiast *minimum właściwe*, gdy $f_{xx}(P_0) > 0$.

W. Żakowski, W. Kołodziej: "Matematyka II",
Podręczniki Akademickie: Elektronika,
WNT, Warszawa, 1977
(str. 74)

(popularne sformułowanie /dla funkcji dwuargumentowej/)

Ekstrema (minima/maksima) lokalne

- To samo z użyciem innych oznaczeń

- funkcja

$$f([x,y]^T) \equiv f(x,y)$$

- założenie: $f(x,y) \in C^2$

- gradient

$$\nabla_f([x_0,y_0]^T) = [\partial f/\partial x, \partial f/\partial y]^T \equiv [g_1, g_2]^T$$

- hesjan

$$\mathbf{H}_f([x_0,y_0]^T) = [\partial^2 f/\partial x^2, \partial^2 f/(\partial x \partial y); \partial^2 f/(\partial y \partial x), \partial^2 f/\partial y^2]^T \equiv [h_{11}, h_{12}; h_{21}, h_{22}]^T$$

- ze względu na symetrię: $h_{12} = h_{21}$

h_{11}	h_{12}
h_{21}	h_{22}

- Twierdzenie o ekstremach (lokalnych)

Jeżeli $g_1 = 0$ i $g_2 = 0$ i $h_{11}h_{22} - h_{12}h_{21} > 0$

- oraz $h_{11} > 0$, to $[x_0,y_0]^T$ stanowi minimum (lokalne) funkcji $f([x,y]^T)$
- oraz $h_{11} < 0$, to $[x_0,y_0]^T$ stanowi maksimum (lokalne) funkcji $f([x,y]^T)$

Ekstrema (minima/maksima) lokalne

- Pojęcia
 - gradient ∇_f
 - gradient w punkcie $\nabla_f(\mathbf{x}_0)$
 - hesjan \mathbf{H}_f
 - hesjan w punkcie $\mathbf{H}_f(\mathbf{x}_0)$

Ekstrema (minima/maksima) lokalne

- Wektor \mathbf{x}_0 dla którego $\nabla_f(\mathbf{x}_0)$ istnieje i spełnia $\nabla_f(\mathbf{x}_0) = \mathbf{0}$ nazywamy punktem stacjonarnym funkcji $f(\mathbf{x})$
- Jeżeli funkcja $f(\mathbf{x}) \in C^2$ posiada ekstremum (lokalne) w \mathbf{x}_0 i $\nabla_f(\mathbf{x}_0)$ istnieje, to \mathbf{x}_0 jest punktem stacjonarnym funkcji $f(\mathbf{x})$
 - wniosek:
 $\nabla_f(\mathbf{x}_0) = \mathbf{0}$ (stacjonarność) jest warunkiem koniecznym istnienia ekstremum (lokalnego) funkcji $f(\mathbf{x})$

Ekstrema (minima/maksima) lokalne

- Twierdzenie o ekstremach (lokalnych) funkcji $f(\mathbf{x}) \in C^2$
Jeżeli \mathbf{x}_0 jest punktem stacjonarnym
 - oraz $\mathbf{H}_f(\mathbf{x}_0) > 0$ ($\mathbf{H}_f(\mathbf{x}_0)$ jest macierzą dodatnio określoną) to \mathbf{x}_0 stanowi minimum (lokalne)
 - oraz $\mathbf{H}_f(\mathbf{x}_0) < 0$ ($\mathbf{H}_f(\mathbf{x}_0)$ jest macierzą ujemnie określoną) to \mathbf{x}_0 stanowi maksimum (lokalne)
- Dodatkowo
 - jeżeli \mathbf{x}_0 jest punktem stacjonarnym funkcji $f(\mathbf{x}) \in C^2$ a wszystkie wartości własne macierzy $\mathbf{H}_f(\mathbf{x}_0)$ są niezerowe i (wszystkie) nie posiadają identycznego znaku, to \mathbf{x}_0 stanowi siodło

Ekstrema (minima/maksima) lokalne

- Twierdzenie o ekstremach (lokalnych) funkcji $f(\mathbf{x}) \in C^2$
(ogólniejsze sformułowanie)

Jeżeli \mathbf{x}_0 jest punktem stacjonarnym i wszystkie wartości własne macierzy $\mathbf{H}_f(\mathbf{x}_0)$ są niezerowe, to

- to \mathbf{x}_0 stanowi ekstremum (lokalne) gdy wszystkie wartości własne mają ten sam znak, przy czym jest to
 - minimum, gdy wartości te są dodatnie
 - maksimum, gdy wartości te są ujemne
- to \mathbf{x}_0 stanowi siodło gdy nie wszystkie wartości własne mają ten sam znak

Ekstrema (minima/maksima) lokalne

- Twierdzenie o ekstremach (lokalnych)

Jeżeli $g_1 = 0$ i $g_2 = 0$ i $h_{11}h_{22} - h_{12}h_{21} > 0$

- oraz $h_{11} > 0$, to $[x_0, y_0]^T$ stanowi minimum (lokalne) funkcji $f([x, y]^T)$
- oraz $h_{11} < 0$, to $[x_0, y_0]^T$ stanowi maksimum (lokalne) funkcji $f([x, y]^T)$

„przetłumaczone”

- $g_1 = 0$ i $g_2 = 0 \Leftrightarrow \nabla_f(\mathbf{x}_0) = \mathbf{0}$

natomiast

- $h_{11} > 0$ i $h_{11}h_{22} - h_{12}h_{21} > 0 \Leftrightarrow \mathbf{H}_f(\mathbf{x}_0) > 0$
- $h_{11} < 0$ i $h_{11}h_{22} - h_{12}h_{21} > 0 \Leftrightarrow \mathbf{H}_f(\mathbf{x}_0) < 0$

h_{11}	h_{12}
h_{21}	h_{22}

...

Metoda Newtona-Raphsona

- Metoda Newtona-Raphsona
 - metoda optymalizacji wielowymiarowej bez ograniczeń (z ewentualnymi ograniczeniami na zakres zmienności zmiennych)
- Dane
 - wielowymiarowy obszar S
 - określona w obszarze S funkcja $f(\mathbf{x})$
- Cel metody
 - znaleźć $\mathbf{x}^* \in S$ taki, że $\forall_{\mathbf{x} \in S} f(\mathbf{x}^*) \leq f(\mathbf{x})$ (minimalizacja funkcji $f(\mathbf{x})$ w obszarze S)

Metoda Newtona-Raphsona

- Naturalne uogólnienie metody Newtona n wymiarów
 - niech
 - $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ będzie wektorem o rozmiarze $n \times 1$
 - $f(\mathbf{x})$ będzie funkcją jednowymiarową od wektora \mathbf{x}
 - trzejelementowe rozwinięcie $q(\mathbf{x})$ funkcji $f(\mathbf{x})$ w szereg Taylora wokół wektora $\mathbf{y} \in S$ dane jest następującym wzorem
$$q(\mathbf{x}) = f(\mathbf{y}) + (\nabla_f(\mathbf{y}))^T(\mathbf{x}-\mathbf{y}) + 1/2 \cdot (\mathbf{x}-\mathbf{y})^T \mathbf{H}_f(\mathbf{y})(\mathbf{x}-\mathbf{y})$$
 - gdzie
 - $\nabla_f(\mathbf{y})$ – gradient funkcji $f(\mathbf{x})$ od wektora \mathbf{y}
 - $\mathbf{H}_f(\mathbf{y})$ – hesjan funkcji $f(\mathbf{x})$ od wektora \mathbf{y}

Metoda Newtona-Raphsona

- Uogólnienie schematu iteracyjnego na n wymiarów, c.d.
 - analogiczna do przypadku jednowymiarowego zasada ustalania następnego wektora na podstawie poprzedniego pozwala na sformułowanie następującego schematu iteracyjnego

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (\mathbf{H}_f(\mathbf{x}_k))^{-1} \nabla_f(\mathbf{x}_k)$$

Algorytm

1. ustal wektor \mathbf{x}_0 i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k - (\mathbf{H}_f(\mathbf{x}_k))^{-1} \nabla_f(\mathbf{x}_k)$
 - podstaw $k = k + 1$

Metoda Newtona-Raphsona

- Potencjalne warunki stopu metody
 - osiągnięcie minimum
 - teoretycznie badamy: $\nabla_f(\mathbf{x}_k) = \mathbf{0}$
 - praktycznie badamy: $\|\nabla_f(\mathbf{x}_k)\| \leq \varepsilon$
 - ustabilizowanie wyniku
 - teoretycznie badamy: $\mathbf{x}_{k+1} = \mathbf{x}_k$
 - praktycznie badamy: $\|\mathbf{x}_{k+1} - \mathbf{x}_k\| \leq \varepsilon$
 - przekroczenie maksymalnej liczby iteracji
 - $k > k_0$

gdzie

- ε jest małą, dodatnią wartością rzeczywistą (dokładność obliczeń)
- k_0 jest dużą, dodatnią wartością całkowitą (maksymalna liczba iteracji)

Metoda Newtona-Raphsona

- Uwaga:
 - w szczególności, tzn. dla $\mathbf{a} = [a]$ zachodzi: $\|\mathbf{a}\| \equiv |a|$
 - (w obu przypadkach: wartość skalarna)
 - w ogólności jednak $\|\mathbf{a}\| \neq |\mathbf{a}|$
 - zastosowanie $\max(|\mathbf{a}|)$ (zamiast $\|\mathbf{a}\|$) byłoby jednak dopuszczalne

Metoda Newtona-Raphsona

- Zbieżność metody
 - metoda nie gwarantuje zbieżności dla każdego wektora początkowego
 - teoretyczne przyczyny ewentualnej niezbieżności
 - osobliwość hesjanu (a więc nie istnieje jego odwrotność)
rezultat: nie można obliczyć \mathbf{x}_{k+1}
 - niewłaściwy krok metody (choć prawidłowo obliczony)
rezultat: $\|\nabla_f(\mathbf{x}_{k+1})\| \geq \|\nabla_f(\mathbf{x}_k)\|$
 - praktyczne przyczyny ewentualnej niezbieżności
 - ...
 - w (korzystnych) przypadkach zbieżnych: (w pobliżu rozwiązania)
zbieżność rzędu drugiego (czyli wysoka!)

Metoda Newtona-Raphsona

- Czy są możliwe sytuacje, w których metoda Newtona-Raphsona nie działa wcale?
 - tak
 - przyczyny
 - gradient/hesjan nieokreślony (nie można zainicjować ciągu $\{\|\mathbf{x}_k\|\}$)
 - hesjan dla pewnego \mathbf{x}_k osobliwy (nie można utworzyć elementu \mathbf{x}_{k+1})
 - ciąg $\{\|\mathbf{x}_k\|\}$ jest niezbieżny, a więc np.:
 - ciąg $\{\|\mathbf{x}_k\|\}$ dąży do $+\infty$
 - ciąg $\{\|\mathbf{x}_k\|\}$ jest cykliczny
 - ciąg $\{\|\mathbf{x}_k\|\}$ przejawia inne powody niezbieżności
 - » np.: +1, 0, +4, 0, +16, 0, +64, 0, +256, ...

...

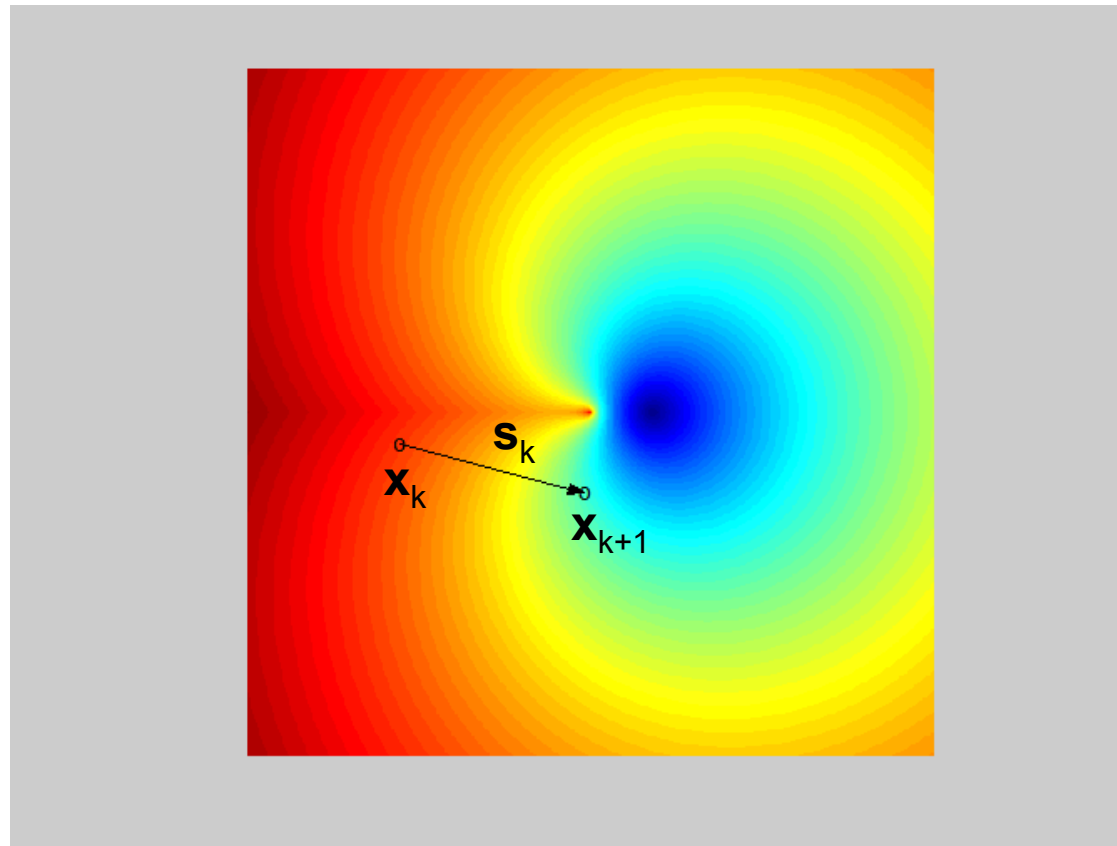
Metoda Newtona-Raphsona

- Metoda Newtona jest szczególnym przypadkiem metody Newtona-Raphsona, ponieważ gdy \mathbf{x} jest wektorem jednoelementowym, czyli $\mathbf{x} = [x]$ (rozmiar 1×1), to
 - $f(\mathbf{x}) = f([x])$ jest funkcją jednowymiarową, którą można zapisywać jako $f(x)$
 - $\nabla_f(\mathbf{x}) = \nabla_f([x]) = [\partial f / \partial x_1] = [f'(x)]$ jest wektorem jednoelementowym, który można zapisywać jako $f'(x)$
 - $\mathbf{H}_f(\mathbf{x}) = \mathbf{H}_f([x]) = [\partial^2 f / (\partial x_1 \partial x_1)] = [f''(x)]$ jest macierzą jednoelementową, którą można zapisywać jako $f''(x)$
- A więc
 - zapis
$$\mathbf{x}_{k+1} = \mathbf{x}_k - (\mathbf{H}_f(\mathbf{y}))^{-1} \nabla_f(\mathbf{y})$$
 - sprowadza się do
$$x_{k+1} = x_k - (f'(x))^{-1} f'(x)$$
 - czyli do
$$x_{k+1} = x_k - f(x)/f'(x)$$

Metoda Newtona-Raphsona

- Krok (wielowymiarowej) metody iteracyjnej
 - wektor dodawany do wektora \mathbf{x}_k w celu przekształcenia go w wektor \mathbf{x}_{k+1} nosi nazwę kroku metody i jest oznaczany przez \mathbf{s}_k
 - w metodzie Newtona-Raphsona $\mathbf{s}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$
(ale w innych metodach wektor ten może być ustalany inaczej)

Metoda Newtona-Raphsona



Ilustracja kroku metody $\mathbf{s}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$
(reprezentowany strzałką)

...

Metoda Newtona-Raphsona

- Przykład minimalizacji funkcji od argumentu dwuwymiarowego (argumenty mają postać $\mathbf{x} = [x_1, x_2]^T$)

– funkcja

$$f(\mathbf{x}) = (x_1)^2 + 2(x_2)^2 + 3x_1$$

– gradient

$$\nabla_f = \begin{bmatrix} \partial f / \partial x_1 \\ \partial f / \partial x_2 \end{bmatrix} = \begin{bmatrix} 2x_1 + 3 \\ 4x_2 \end{bmatrix}$$

– hesjan

$$\mathbf{H}_f = \begin{bmatrix} \partial^2 f / (\partial x_1 \partial x_1) & \partial^2 f / (\partial x_1 \partial x_2) \\ \partial^2 f / (\partial x_2 \partial x_1) & \partial^2 f / (\partial x_2 \partial x_2) \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}$$

Metoda Newtona-Raphsona

- Przykład minimalizacji, c.d.
 - gradient w kolejnych iteracjach
 - ponieważ gradient nie jest wektorem stałych, więc będzie musiał być wyliczany w każdej iteracji
 - hesjan w kolejnych iteracjach
 - ponieważ hesjan jest macierzą stałych, więc będzie taki sam we wszystkich iteracjach
 - odwrotność hesjanu (także identyczna we wszystkich iteracjach)
- $$(\mathbf{H}_f)^{-1} = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/4 \end{bmatrix}$$
- przyjęty warunek stopu: $\nabla_f(\mathbf{x}_k) = \mathbf{0}$

Metoda Newtona-Raphsona

- Przykład minimalizacji, c.d.

- $\mathbf{x}_0 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$

- pierwsza iteracja

- $\nabla_f(\mathbf{x}_0) = \begin{bmatrix} 2 \cdot 1 + 3 \\ 4 \cdot 2 \end{bmatrix} = \begin{bmatrix} 5 \\ 8 \end{bmatrix}$

- ponieważ warunek stopu nie jest spełniony,

- obliczamy $\mathbf{x}_1 = \mathbf{x}_0 - (\mathbf{H}_f(\mathbf{x}_0))^{-1} \nabla_f(\mathbf{x}_0)$

- $\mathbf{x}_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 1/2 & 0 \\ 0 & 1/4 \end{bmatrix} \begin{bmatrix} 5 \\ 8 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 5/2 \\ 2 \end{bmatrix} = \begin{bmatrix} -3/2 \\ 0 \end{bmatrix}$

Metoda Newtona-Raphsona

- Przykład minimalizacji, c.d.
 - $\mathbf{x}_1 = \begin{bmatrix} -3/2 \\ 0 \end{bmatrix}$
 - druga iteracja
$$\nabla_f(\mathbf{x}_1) = \begin{bmatrix} 2 \cdot (-3/2) + 3 \\ 4 \cdot 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$
 - ponieważ warunek stopu jest spełniony, więc wektor \mathbf{x}_1 jest rozwiązaniem (stanowi minimum funkcji $f(\mathbf{x})$)
 - hesjan jest macierzą dodatnio określoną
 - oznaczenie $\mathbf{H}_f(\mathbf{x}_0) > 0$

Metoda Newtona-Raphsona

- Przykład minimalizacji, c.d.
 - dzięki temu, że $\nabla_f(\mathbf{x}_1) = \mathbf{0}$, kolejne rozwiązania, tzn. $\mathbf{x}_2, \mathbf{x}_3, \dots$, spełniałyby $\mathbf{x}_1 = \mathbf{x}_2 = \mathbf{x}_3 = \dots$

Metoda Newtona-Raphsona

- Przykład minimalizacji funkcji od argumentu dwuwymiarowego (argumenty mają postać $\mathbf{x} = [x_1, x_2]^T$)

– funkcja

$$f(\mathbf{x}) = (x_1)^2 + (x_1x_2)^2 + (x_2)^2$$

– gradient

$$\nabla_f = \begin{bmatrix} \partial f / \partial x_1 \\ \partial f / \partial x_2 \end{bmatrix} = \begin{bmatrix} 2x_1 + 2x_1(x_2)^2 \\ 2x_2 + 2(x_1)^2x_2 \end{bmatrix}$$

– hesjan

$$\mathbf{H}_f = \begin{bmatrix} \partial^2 f / (\partial x_1 \partial x_1) & \partial^2 f / (\partial x_1 \partial x_2) \\ \partial^2 f / (\partial x_2 \partial x_1) & \partial^2 f / (\partial x_2 \partial x_2) \end{bmatrix} = \begin{bmatrix} 2 + 2(x_2)^2 & 4x_1x_2 \\ 4x_1x_2 & 2 + 2(x_1)^2 \end{bmatrix}$$

Metoda Newtona-Raphsona

- Przykład minimalizacji, c.d.
 - gradient w kolejnych iteracjach
 - ponieważ gradient nie jest wektorem stałych, więc będzie musiał być wyliczany w każdej iteracji
 - hesjan w kolejnych iteracjach
 - ponieważ hesjan nie jest wektorem stałych, więc będzie musiał być wyliczany w każdej iteracji
 - przyjęty warunek stopu: $\nabla_f(\mathbf{x}_k) = \mathbf{0}$

Metoda Newtona-Raphsona

- Przykład minimalizacji, c.d.

- rozwiązanie początkowe $\mathbf{x}_0 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$

- pierwsza iteracja

$$\nabla_f(\mathbf{x}_0) = \begin{bmatrix} 4 + 4 \\ 2 + 8 \end{bmatrix} = \begin{bmatrix} 8 \\ 10 \end{bmatrix}$$

$$\mathbf{H}_f(\mathbf{x}_0) = \begin{bmatrix} 2 + 2 \cdot 1^2 & 4 \cdot 2 \cdot 1 \\ 4 \cdot 2 \cdot 1 & 2 + 2 \cdot 2^2 \end{bmatrix} = \begin{bmatrix} 4 & 8 \\ 8 & 10 \end{bmatrix}$$

$$(\mathbf{H}_f(\mathbf{x}_0))^{-1} = \begin{bmatrix} -10/24 & 8/24 \\ 8/24 & -4/24 \end{bmatrix}$$

- ponieważ warunek stopu nie jest spełniony, obliczamy $\mathbf{x}_1 = \mathbf{x}_0 - (\mathbf{H}_f(\mathbf{x}_0))^{-1} \nabla_f(\mathbf{x}_0)$

$$\mathbf{x}_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix} - \begin{bmatrix} -10/24 & 8/24 \\ 8/24 & -4/24 \end{bmatrix} \begin{bmatrix} 8 \\ 10 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

Metoda Newtona-Raphsona

- Przykład minimalizacji, c.d.

- druga iteracja

$$\nabla_f(\mathbf{x}_1) = \begin{bmatrix} 4 + 0 \\ 0 + 0 \end{bmatrix} = \begin{bmatrix} 4 \\ 0 \end{bmatrix}$$

$$\mathbf{H}_f(\mathbf{x}_1) = \begin{bmatrix} 2 + 2 \cdot 0^2 & 0 \\ 0 & 2 + 2 \cdot 2^2 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 10 \end{bmatrix}$$

$$(\mathbf{H}_f(\mathbf{x}_1))^{-1} = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/10 \end{bmatrix}$$

- ponieważ warunek stopu nie jest spełniony, obliczamy $\mathbf{x}_2 = \mathbf{x}_1 - (\mathbf{H}_f(\mathbf{x}_1))^{-1} \nabla_f(\mathbf{x}_1)$

$$\mathbf{x}_2 = \begin{bmatrix} 2 \\ 0 \end{bmatrix} - \begin{bmatrix} 1/2 & 0 \\ 0 & 1/10 \end{bmatrix} \begin{bmatrix} 4 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \end{bmatrix} - \begin{bmatrix} 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Metoda Newtona-Raphsona

- Przykład minimalizacji, c.d.
 - $\mathbf{x}_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$
 - trzecia iteracja
 $\nabla_f(\mathbf{x}_2) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$
 - ponieważ warunek stopu jest spełniony, więc wektor \mathbf{x}_2 jest rozwiązaniem (stanowi minimum funkcji $f(\mathbf{x})$)
 - hesjan jest macierzą dodatnio określoną
 - oznaczenie $\mathbf{H}_f(\mathbf{x}_0) > 0$

Metoda Newtona-Raphsona

- Przykład minimalizacji, c.d.
 - uwaga: dzięki temu, że $\nabla_f(\mathbf{x}_2) = \mathbf{0}$, kolejne rozwiązania, tzn. $\mathbf{x}_3, \mathbf{x}_4, \dots$, spełniałyby $\mathbf{x}_2 = \mathbf{x}_3 = \mathbf{x}_4 = \dots$

Metoda Newtona-Raphsona

- Przykład minimalizacji, c.d.

- rozwiązanie początkowe $\mathbf{x}_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

- pierwsza iteracja

$$\nabla_f(\mathbf{x}_0) = \begin{bmatrix} 2 + 2 \\ 2 + 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \end{bmatrix}$$

$$\mathbf{H}_f(\mathbf{x}_0) = \begin{bmatrix} 2 + 2 \cdot 1^2 & 4 \cdot 1 \cdot 1 \\ 4 \cdot 1 \cdot 1 & 2 + 2 \cdot 1^2 \end{bmatrix} = \begin{bmatrix} 4 & 4 \\ 4 & 4 \end{bmatrix}$$

$(\mathbf{H}_f(\mathbf{x}_0))^{-1}$ nie istnieje

...

Przykład ciekawego zastosowania m. Newtona-Raphsona

- Problem najmniejszych kwadratów (PMK) i jego rozwiązanie
 - problem najmniejszych kwadratów może (i zazwyczaj jest) rozwiązywany tzw. metodą najmniejszych kwadratów (MNK) 😊
- PNK jest problemem dopasowania prostej/płaszczyzny/... do zbioru pewnych punktów, czyli problemem odkrycia zależności liniowej pomiędzy ustalonymi zmiennymi wejściowymi a wyjściowymi
- PNK jest (typowym) problemem optymalizacyjnym, jednak na tyle prostym, że posiada jawne rozwiązanie analityczne

Przykład ciekawego zastosowania m. Newtona-Raphsona

- Wywód PNK startujący od rozwiązania układu równań
 - niech dany będzie układ $\mathbf{X}\mathbf{b} = \mathbf{y}$, gdzie \mathbf{X} , \mathbf{b} i \mathbf{y} są macierzami/wektorami o wymiarach $\mathbf{X}_{m \times n}$, $\mathbf{b}_{n \times 1}$, $\mathbf{y}_{m \times 1}$, w którym \mathbf{X} i \mathbf{y} są (odpowiednio) macierzą oraz wektorem stałych, natomiast \mathbf{b} jest wektorem zmiennych
 - rozwiązanie układu polega na znalezieniu wektora \mathbf{b} zapewniającego równość: $\mathbf{X}\mathbf{b} = \mathbf{y}$
 - w ogólności, różne możliwe przypadki opisuje twierdzenie Kroneckera-Capelli'ego; w szczególności (dla \mathbf{X} będącego macierzą pełnego rzędu) możliwe są następujące sytuacje
 - gdy $m < n$, to $\mathbf{X}\mathbf{b} = \mathbf{y}$ może mieć nieskończenie wiele rozwiązań
 - gdy $m = n$, to $\mathbf{X}\mathbf{b} = \mathbf{y}$ może mieć jedno rozwiązanie
 - gdy $m > n$, to $\mathbf{X}\mathbf{b} = \mathbf{y}$ może mieć zero rozwiązań
 - co oznacza, że nie istnieje wektor \mathbf{b} zapewniający równość $\mathbf{X}\mathbf{b} = \mathbf{y}$

Przykład ciekawego zastosowania m. Newtona-Raphsona

- Wywód PNK startujący od rozwiązania układu równań, c.d.
- Rozważamy sytuację, w której nie istnieje wektor \mathbf{b} zapewniający równość $\mathbf{Xb} = \mathbf{y}$
 - wtedy można szukać \mathbf{x} takiego, aby wektor \mathbf{Xb} był jak najbardziej „bliski” wektorowi \mathbf{y} , tzn. aby wektor $\mathbf{Xb} - \mathbf{y}$ był jak najbardziej „bliski” wektorowi $\mathbf{0}$
 - formalnie: szukamy wektora \mathbf{b} minimalizującego wartość wyrażenia (skalarnego) $\|\mathbf{Xb} - \mathbf{y}\|^2$
 - uwaga: wektor \mathbf{b} spełniający $\mathbf{Xb} = \mathbf{y}$ jest szczególnym przypadkiem wektora minimalizującego $\|\mathbf{Xb} - \mathbf{y}\|^2$ (oczywiście wtedy $\|\mathbf{Xb} - \mathbf{y}\|^2 = 0$)
 - znane jest optymalne rozwiązanie analityczne tego problemu, mające postać: $\mathbf{b} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ (jest wynik tzw. metody najmniejszych kwadratów, MNK); rozwiązanie to istnieje, gdy wszystkie kolumny macierzy \mathbf{X} stanowią zbiór wektorów niezależnych

Przykład ciekawego zastosowania m. Newtona-Raphsona

- (Ogólne) rozwiązywanie PNK metodą Newtona-Raphsona
- Niech $s(\mathbf{b}) = \|\mathbf{Xb} - \mathbf{y}\|^2$
 - uwaga: w tym zapisie
 - \mathbf{X} jest macierzą stałych
 - \mathbf{y} jest wektorem stałychnatomiast:
 - \mathbf{b} jest wektorem zmiennych
 - ponieważ dla rzeczywistych \mathbf{w} zachodzi $\|\mathbf{w}\|^2 \equiv \mathbf{w}^T \mathbf{w}$, więc także $\|\mathbf{Xb} - \mathbf{y}\|^2 = (\mathbf{Xb} - \mathbf{y})^T (\mathbf{Xb} - \mathbf{y})$
 - przekształcając to wyrażenie otrzymujemy:
$$\begin{aligned} \|\mathbf{Xb} - \mathbf{y}\|^2 &= (\mathbf{Xb} - \mathbf{y})^T (\mathbf{Xb} - \mathbf{y}) = ((\mathbf{Xb})^T - \mathbf{y}^T) (\mathbf{Xb} - \mathbf{y}) = \\ &= (\mathbf{b}^T \mathbf{X}^T - \mathbf{y}^T) (\mathbf{Xb} - \mathbf{y}) = \mathbf{b}^T \mathbf{X}^T (\mathbf{Xb} - \mathbf{y}) - \mathbf{y}^T (\mathbf{Xb} - \mathbf{y}) = \\ &= \mathbf{b}^T \mathbf{X}^T \mathbf{Xb} - \mathbf{b}^T \mathbf{X}^T \mathbf{y} - (\mathbf{y}^T \mathbf{Xb} - \mathbf{y}^T \mathbf{y}) = \mathbf{b}^T \mathbf{X}^T \mathbf{Xb} - \mathbf{b}^T \mathbf{X}^T \mathbf{y} - \mathbf{y}^T \mathbf{Xb} + \mathbf{y}^T \mathbf{y} = \\ &= \mathbf{b}^T \mathbf{X}^T \mathbf{Xb} - 2\mathbf{y}^T \mathbf{Xb} + \mathbf{y}^T \mathbf{y} = \mathbf{b}^T (\mathbf{X}^T \mathbf{X}) \mathbf{b} - (2\mathbf{y}^T \mathbf{X}) \mathbf{b} + (\mathbf{y}^T \mathbf{y}) \end{aligned}$$
 - postać: $\mathbf{x}^T \mathbf{Ax} + \mathbf{b}^T \mathbf{x} + c$
 - gdzie: \mathbf{A} , \mathbf{b} , c : stałe, \mathbf{x} : zmienna
 - uwaga: stałą $\mathbf{y}^T \mathbf{y}$ zazwyczaj pomijamy w problemie optymalizacji

Przykład ciekawego zastosowania m. Newtona-Raphsona

- (Ogólne) rozwiązywanie PNK metodą Newtona-Raphsona, c.d.

- Zadanie: zminimalizować

$$s(\mathbf{b}) = \|\mathbf{X}\mathbf{b} - \mathbf{y}\|^2 = \mathbf{b}^T(\mathbf{X}^T\mathbf{X})\mathbf{b} - (2\mathbf{y}^T\mathbf{X})\mathbf{b} + (\mathbf{y}^T\mathbf{y})$$

w praktyce

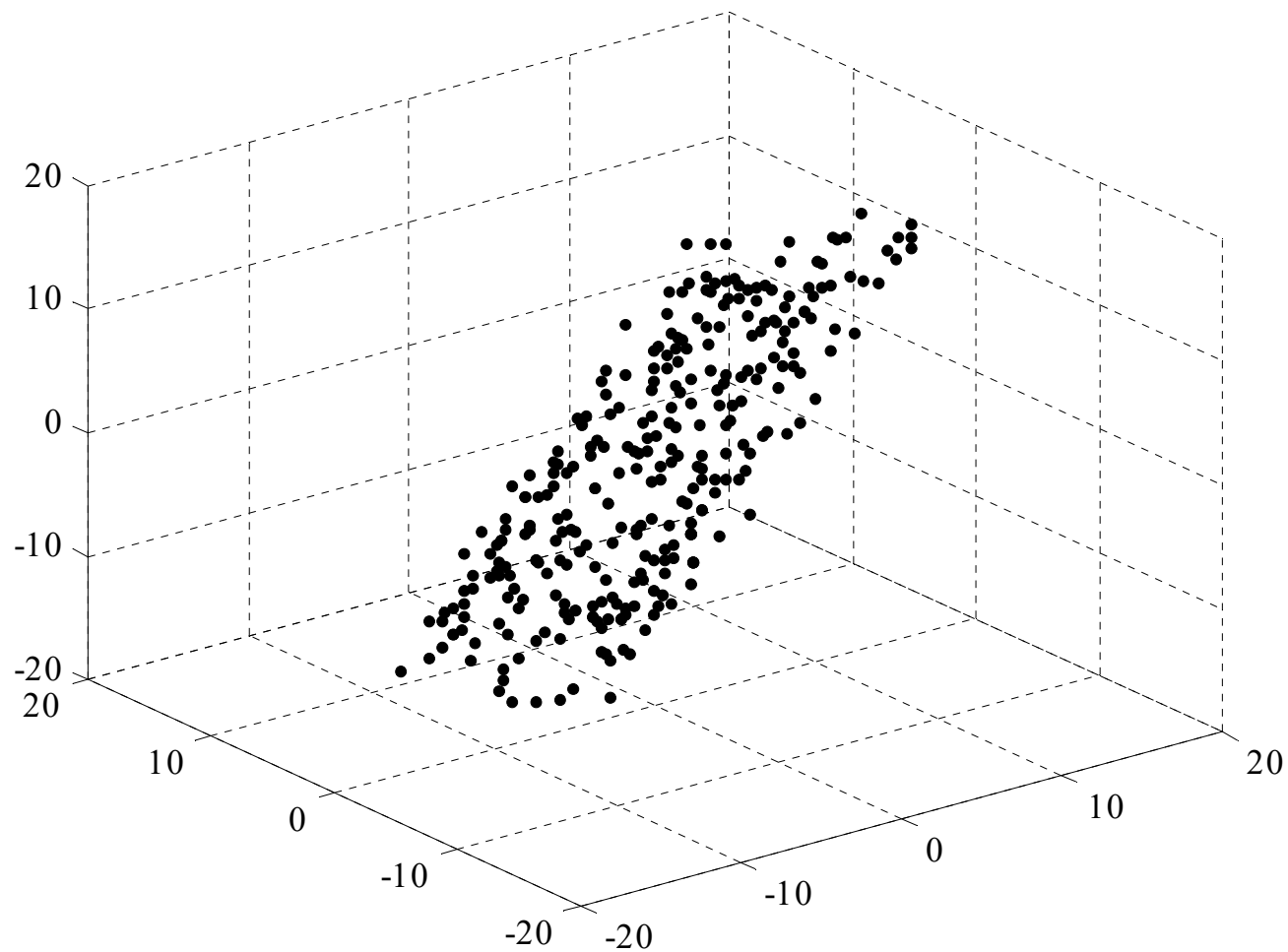
$$s(\mathbf{b}) = \mathbf{b}^T(\mathbf{X}^T\mathbf{X})\mathbf{b} - (2\mathbf{y}^T\mathbf{X})\mathbf{b}$$

- Ustalamy następujące składowe:
 - gradient: $\nabla_s(\mathbf{b}) = 2\mathbf{X}^T\mathbf{X}\mathbf{b} - 2\mathbf{X}^T\mathbf{y} = 2(\mathbf{X}^T\mathbf{X}\mathbf{b} - \mathbf{X}^T\mathbf{y})$ (zależy liniowo od \mathbf{b})
 - hesjan: $\mathbf{H}_s(\mathbf{b}) = 2\mathbf{X}^T\mathbf{X}$ (nie zależy od \mathbf{b})
 - zatem $(\mathbf{H}_s(\mathbf{b}_k))^{-1} = (1/2)(\mathbf{X}^T\mathbf{X})^{-1}$ (nie zależy od \mathbf{b})

Przykład ciekawego zastosowania m. Newtona-Raphsona

- (Szczególne) rozwiązywanie PNK metodą Newtona-Raphsona, c.d.
- Przykład
 - dane wejściowe w postaci 300 „trójek” liczb rzeczywistych x_i, z_i, y_i
 - w praktyce: całkowitych z przedziału $[-9, +9]$
 - $\mathbf{X}_{300 \times 2} = [\mathbf{x}_{300 \times 1} \ \mathbf{z}_{300 \times 1}]$
 - $\mathbf{y}_{300 \times 1}$
 - w praktyce: $\forall_i: y_i = 1.5x_i - 0.5z_i +$ mała wartość losowa
(w praktyce: całkowita z przedziału $[-3, +3]$)

Przykład ciekawego zastosowania m. Newtona-Raphsona



Przykład ciekawego zastosowania m. Newtona-Raphsona

- (Szczególne) rozwiązywanie PNK metodą Newtona-Raphsona, c.d.
 - dla zadanych danych wejściowych
$$\mathbf{X}^T\mathbf{X} = \begin{bmatrix} 8875 & 458 \\ 458 & 9503 \end{bmatrix}$$
$$2\mathbf{y}^T\mathbf{X} = \begin{bmatrix} 26091 \\ -8371 \end{bmatrix}$$
 - ciekawostka
 - pomimo aż 300 „trójek” danych, macierze/wektory są rozmiaru 2x2/2x1
 - tak czy inaczej, (prawie) każda z 300 „trójek” wywarła jakiś wpływ na wyniki
 - wobec dużej liczby „trójek” względny wpływ każdej z nich jest niewielki
 - wniosek: pewne „trójki” można pominąć (co stanowi ideę pewnej rodziny metod)

Przykład ciekawego zastosowania m. Newtona-Raphsona

- (Szczególne) rozwiązywanie PNK metodą Newtona-Raphsona, c.d.

– niech

$$\mathbf{b}_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

– wtedy

$$\nabla_s(\mathbf{b}) = \begin{bmatrix} -7425 \\ 28293 \end{bmatrix}$$

$$\mathbf{H}_s(\mathbf{b}) = \begin{bmatrix} 17750 & 916 \\ 916 & 19006 \end{bmatrix}$$

$$(\mathbf{H}_s(\mathbf{b}))^{-1} = (1/10000) \cdot \begin{bmatrix} 0.5648 & -0.0272 \\ -0.0272 & 0.5275 \end{bmatrix}$$

$$\mathbf{b}_1 = \dots$$

...

Przykład ciekawego zastosowania m. Newtona-Raphsona

- (Szczególne) rozwiązywanie PNK metodą Newtona-Raphsona, c.d.

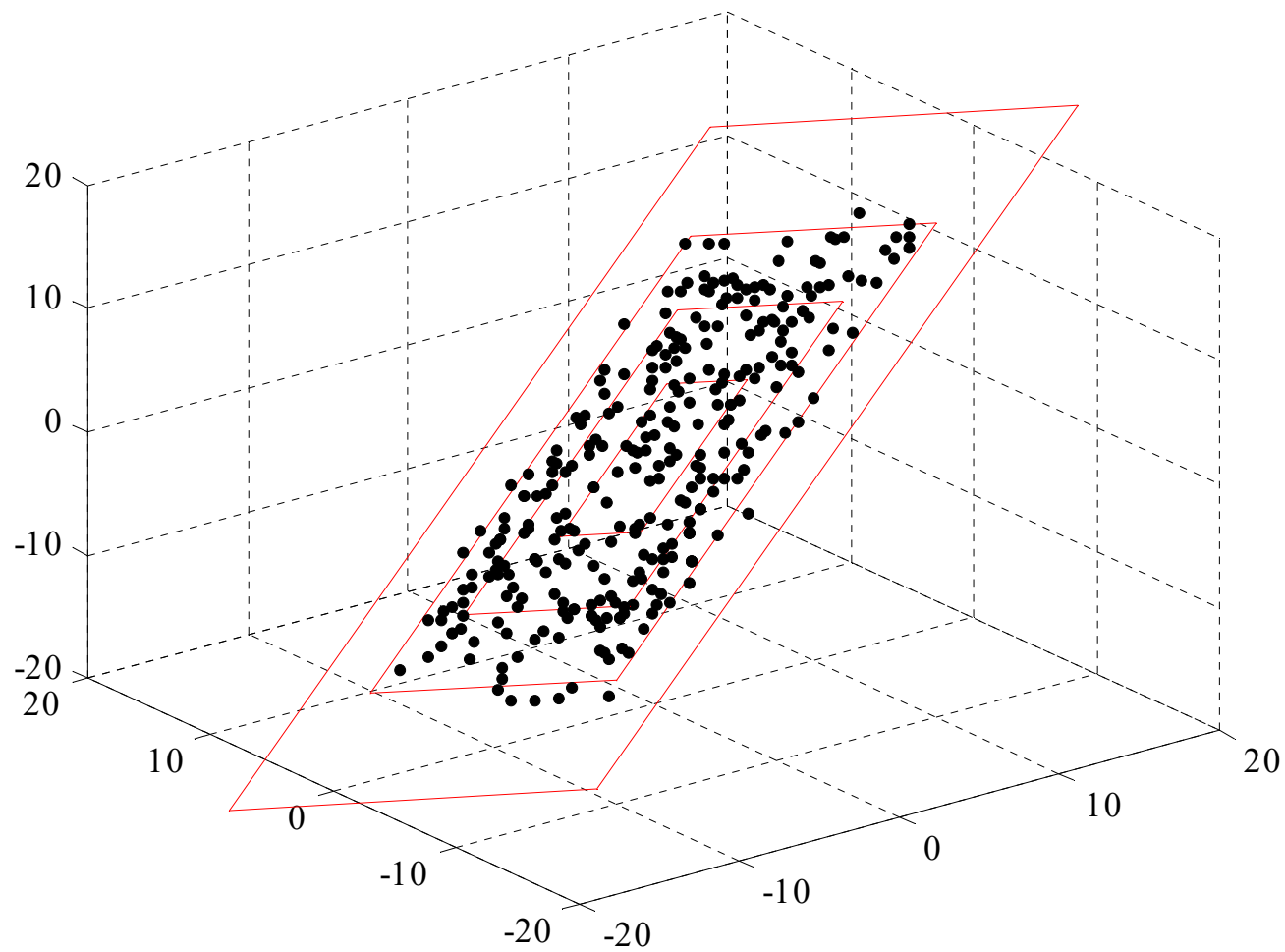
...

– po n iteracjach

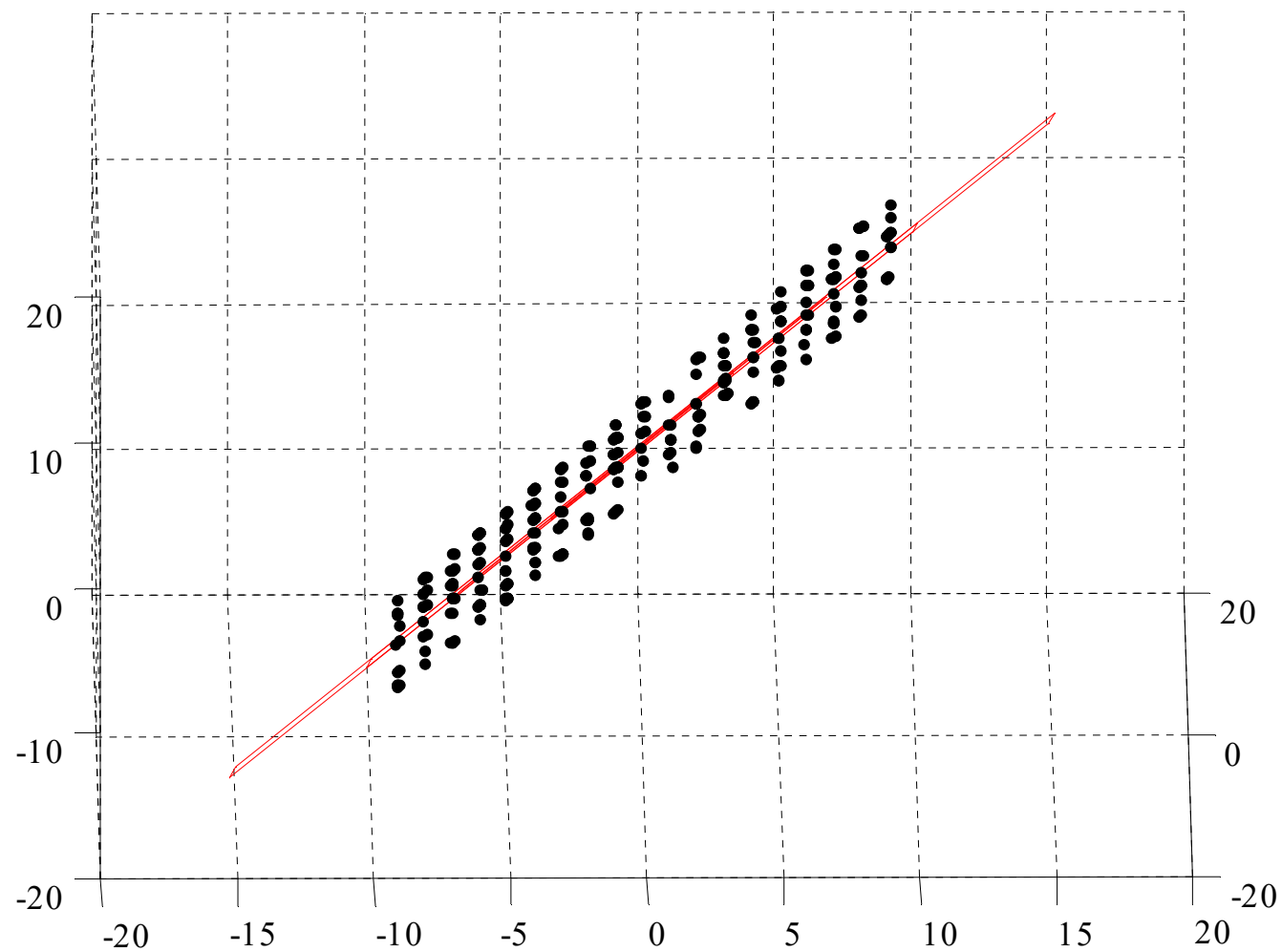
$$\mathbf{b}_n = \begin{bmatrix} 1.4964 \\ -0.5126 \end{bmatrix}$$

- (dla przypomnienia: $\forall_i: z_i = 1.5x_i - 0.5z_i + \text{mała wartość losowa}$)

Przykład ciekawego zastosowania m. Newtona-Raphsona



Przykład ciekawego zastosowania m. Newtona-Raphsona



Przykład ciekawego zastosowania m. Newtona-Raphsona

- (Ogólne) rozwiązywanie PNK metodą Newtona-Raphsona, c.d.
- Znając $\nabla_s(\mathbf{b})$ i $\mathbf{H}_s(\mathbf{b})$ obliczamy

$$\begin{aligned}(\mathbf{H}_s(\mathbf{b}))^{-1}\nabla_s(\mathbf{b}) &= (1/2)(\mathbf{X}^T\mathbf{X})^{-1}\cdot 2(\mathbf{X}^T\mathbf{X}\mathbf{b} - \mathbf{X}^T\mathbf{y}) = \\ &= (\mathbf{X}^T\mathbf{X})^{-1}(\mathbf{X}^T\mathbf{X}\mathbf{b} - \mathbf{X}^T\mathbf{y}) = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}\mathbf{b} - (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = \\ &= \mathbf{I}\mathbf{b} - (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = \mathbf{b} - (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}\end{aligned}$$

Przykład ciekawego zastosowania m. Newtona-Raphsona

- (Ogólne) rozwiązywanie PNK metodą Newtona-Raphsona, c.d.
- Po zaadaptowaniu schematu iteracyjnego metody Newtona-Raphsona $\mathbf{x}_{k+1} = \mathbf{x}_k - (\mathbf{H}_f(\mathbf{x}_k))^{-1} \nabla_f(\mathbf{x}_k)$ do funkcji $s(\mathbf{b})$ (zmienną jest wektor \mathbf{b}) powstaje schemat:

$$\mathbf{b}_{k+1} = \mathbf{b}_k - (\mathbf{H}_s(\mathbf{b}_k))^{-1} \nabla_s(\mathbf{b}_k)$$

- Wykorzystując równość $(\mathbf{H}_s(\mathbf{b}))^{-1} \nabla_s(\mathbf{b}) = \mathbf{b} - (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$ otrzymujemy ostatecznie:

$$\mathbf{b}_{k+1} = \mathbf{b}_k - (\mathbf{b}_k - (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}) = \mathbf{b}_k - \mathbf{b}_k + (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

- Czyli dla dowolnego \mathbf{b}_0 zachodzi:

$$\mathbf{b}_1 = \mathbf{b}_0 - (\mathbf{b}_0 - (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}) = \mathbf{b}_0 - \mathbf{b}_0 + (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

- Jednocześnie dla \mathbf{b}_1 zachodzi:

$$\mathbf{b}_2 = \mathbf{b}_1 - (\mathbf{b}_1 - (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}) = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

itd.

$$\mathbf{b}_n = \mathbf{b}_{n-1} - (\mathbf{b}_{n-1} - (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}) = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

Przykład ciekawego zastosowania m. Newtona-Raphsona

- (Ogólne) rozwiązywanie PNK metodą Newtona-Raphsona, c.d.
- Uzyskany wynik w postaci $\mathbf{b}_1 = \mathbf{b}_2 = \dots = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$
 - powstaje już w wyniku wykonania jednego (pierwszego) kroku
 - nie zależy od \mathbf{b}_0 (tzn. dla każdego \mathbf{b}_0 wektor \mathbf{b}_1 będzie taki sam)
 - stanowi rozwiązanie optymalne (czyli wynik wygenerowany przez MNK)
- Wniosek: dla każdego rozwiązania początkowego metoda Newtona-Raphsona znajduje optymalne rozwiązanie PNK w jednym kroku

Przykład ciekawego zastosowania m. Newtona-Raphsona

- (Szczególne) rozwiązywanie PNK metodą Newtona-Raphsona, c.d.
 - $n = 1$ (wystarczyła jedna iteracja!)

...

Więcej o metodach iteracyjnych

- Ogólna postać (wielowymiarowych) metod iteracyjnych wyrażona z jawnym wykorzystaniem kroku metody

Algorytm

1. ustal wektor \mathbf{x}_0 i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - znajdź \mathbf{s}_k
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
 - podstaw $k = k + 1$

Więcej o metodach iteracyjnych

- Wersja algorytmu Newtona-Raphsona z jawnym wykorzystaniem kroku metody

Algorytm

1. ustal wektor \mathbf{x}_0 i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - oblicz $\mathbf{s}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1} \nabla_f(\mathbf{x}_k)$
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
 - podstaw $k = k + 1$

Więcej o metodach iteracyjnych

- Wyznaczając krok metody możliwe jest rozważenie osobno
 - kierunku poszukiwań: \mathbf{d}_k
 - wektor (zasadniczo niezerowy)
 - reprezentuje wyłącznie kierunek (tzn. kierunek kroku metody)
 - jeżeli jest niezerowy, to jest zwykle przedstawiany postaci unormowanej (tzn. o długości 1)
 - długości kroku: α_k
 - skalar (zasadniczo niezerowy)
 - specyfikuje wyłącznie długość (tzn. długość kroku metody)
 - może być wyznaczany tylko wtedy, gdy $\mathbf{d}_k \neq \mathbf{0}$
 - wtedy: krok metody $\mathbf{s}_k = \alpha_k \mathbf{d}_k$

Więcej o metodach iteracyjnych

- Kierunek poszukiwań i długość kroku na podstawie wektora kroku
 - mając \mathbf{s}_k (który w metodzie Newtona-Raphsona obliczamy jako $\mathbf{s}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$) można (o ile $\mathbf{s}_k \neq \mathbf{0}$) zawsze znaleźć kierunek poszukiwań \mathbf{d}_k i długość kroku α_k wykorzystując zależności
 - $\mathbf{d}_k = \mathbf{s}_k / \|\mathbf{s}_k\|$
 - $\alpha_k = \|\mathbf{s}_k\|$(choć tak naprawdę nie są już wtedy one metodzie potrzebne)
 - uwaga: krok metody, a tym samym kierunek poszukiwań i długość kroku mogą nie istnieć!
 - w niektórych innych metodach kolejność pozyskiwania tych elementów może być jednak inna (najpierw kierunek i długość, a potem krok), co pozwala tym metodom pokonywać pewne słabości metody Newtona-Raphsona

Więcej o metodach iteracyjnych

- Ogólna postać (wielowymiarowych) metod iteracyjnych wyrażona z jawnym wykorzystaniem kierunku poszukiwań oraz długości kroku

Algorytm

1. ustal wektor \mathbf{x}_0 i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - znajdź \mathbf{d}_k
 - znajdź α_k
 - oblicz $\mathbf{s}_k = \alpha_k \mathbf{d}_k$
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
 - podstaw $k = k + 1$

...

Problemy metody Newtona-Raphsona

- Potencjalne problemy
 - kierunek poszukiwań
 - nie istnieje
 - jest bliski wektorowi zerowemu
 - nie jest właściwy
 - w szczególności: nie tworzy kąta rozwartego z gradientem
 - długość kroku
 - jest bliska zeru
 - jest niewłaściwa
- Problemy te nie są wynikiem jakiegoś błędu postępowania, lecz konsekwencją stosowania w metodzie (prawie zawsze*) skończonego przybliżenia funkcji $f(\mathbf{x})$

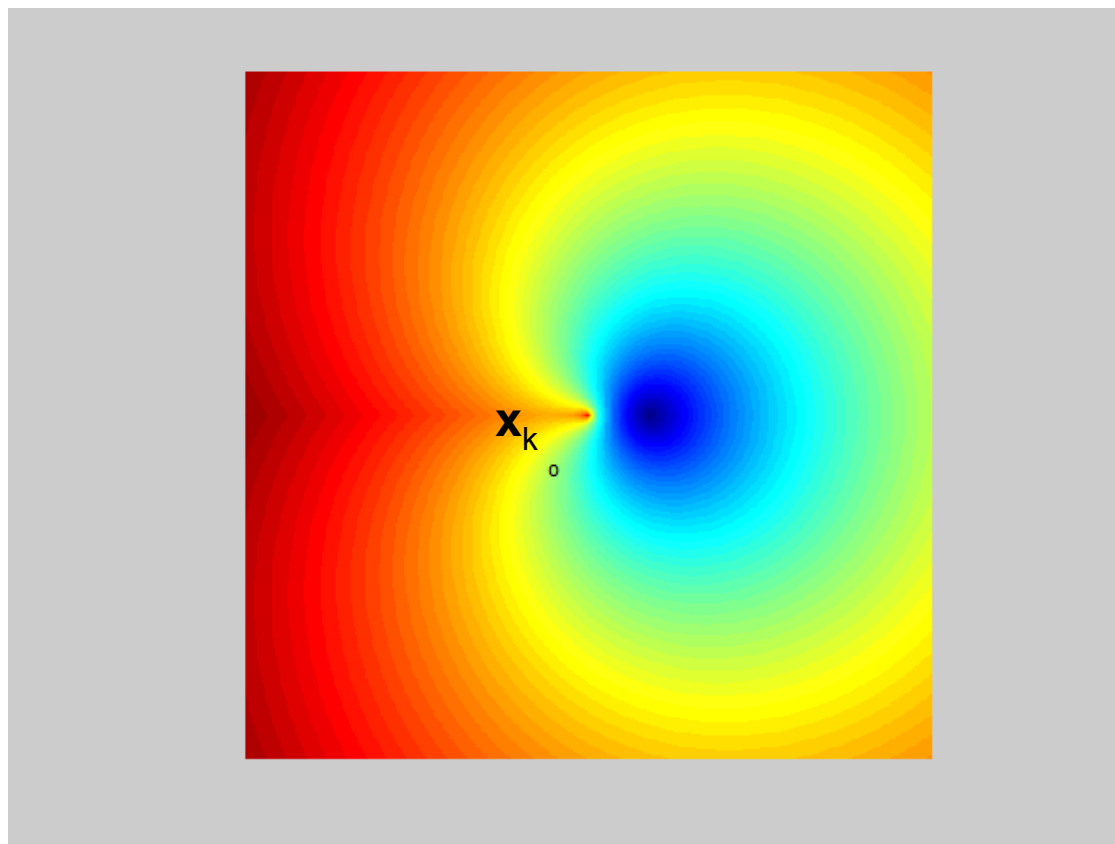
* a więc czasami przybliżenie może być nieskończone?

Idea niezależnego kierunku poszukiwań i długości kroku

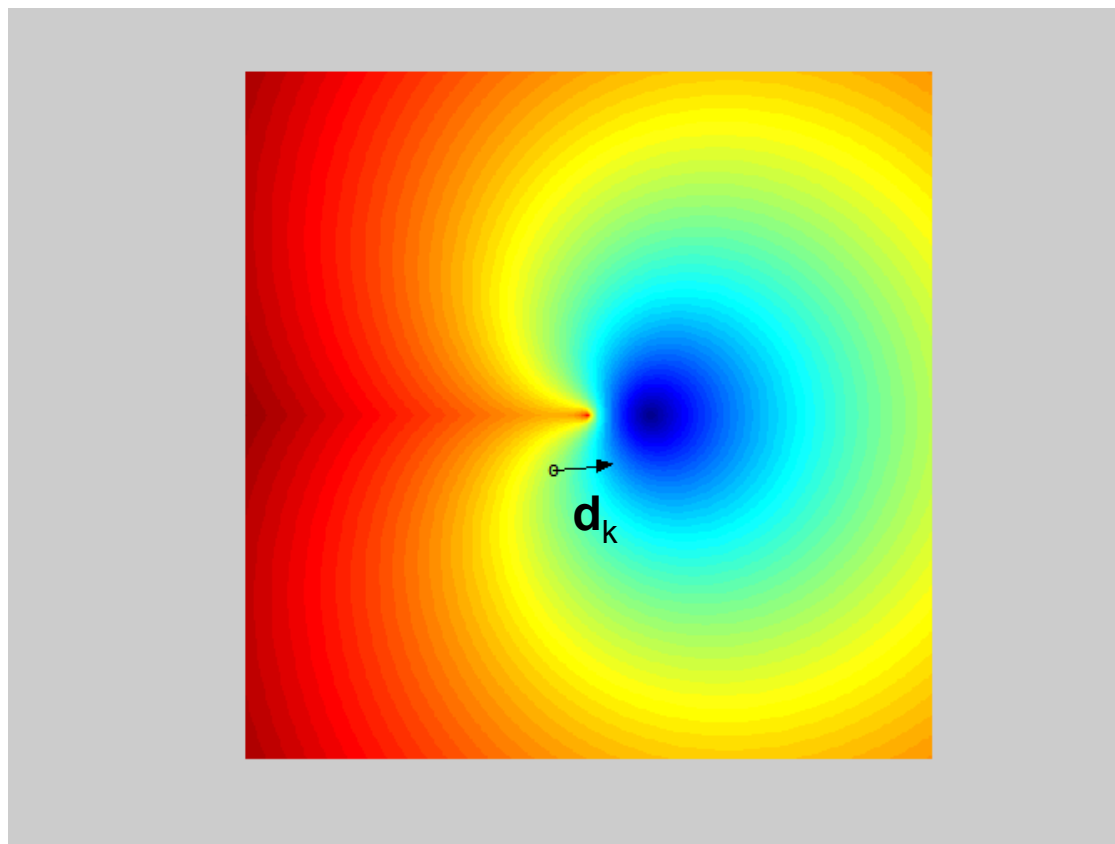
- Problem długości kroku w metodzie Newtona-Raphsona
 - obserwacja: dla znalezionej metody $\mathbf{s}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$ długość tego kroku $\alpha_k = \|\mathbf{s}_k\|$ może być nieoptymalna, tzn. może się okazać, że lepsze przybliżenie minimum funkcji otrzymuje się dla długości kroku większej lub mniejszej od obliczonej

Idea niezależnego kierunku poszukiwań i długości kroku

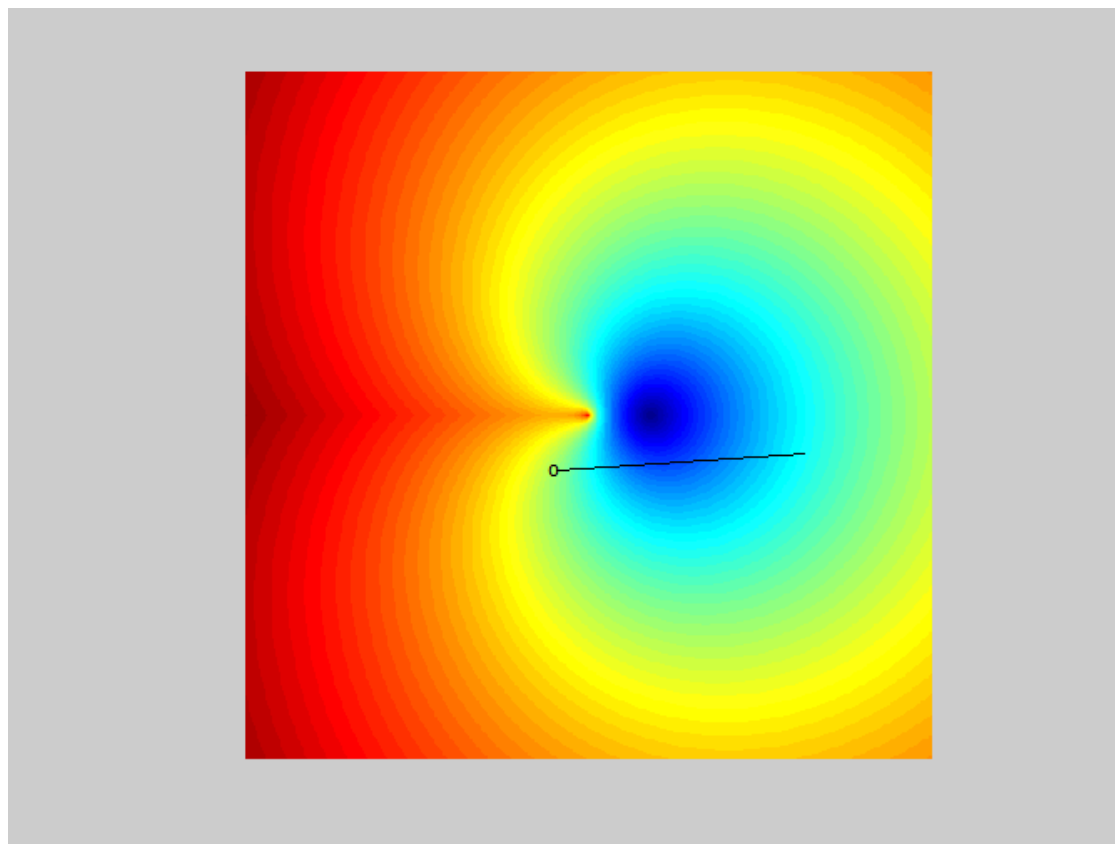
- Inna metoda poszukiwania kroku metody \mathbf{s}_k
 - ustal $\mathbf{d}_k \neq \mathbf{0}$
 - znajdź $\alpha_k \geq 0$ minimalizujące funkcję $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$
 - w praktyce interesują nas wartości $\alpha_k > 0$
 - oblicz $\mathbf{s}_k = \alpha_k \mathbf{d}_k$
- Metoda ta zawiera wewnętrzny problem optymalizacji
 - jest to zawsze problem optymalizacji jednowymiarowej z ograniczeniem na zakres zmiennej (zmienna nieujemna)
 - jako taki może być rozwiązywany np. metodą Newtona (z modyfikacją uwzględniającą nieujemność zmiennej)



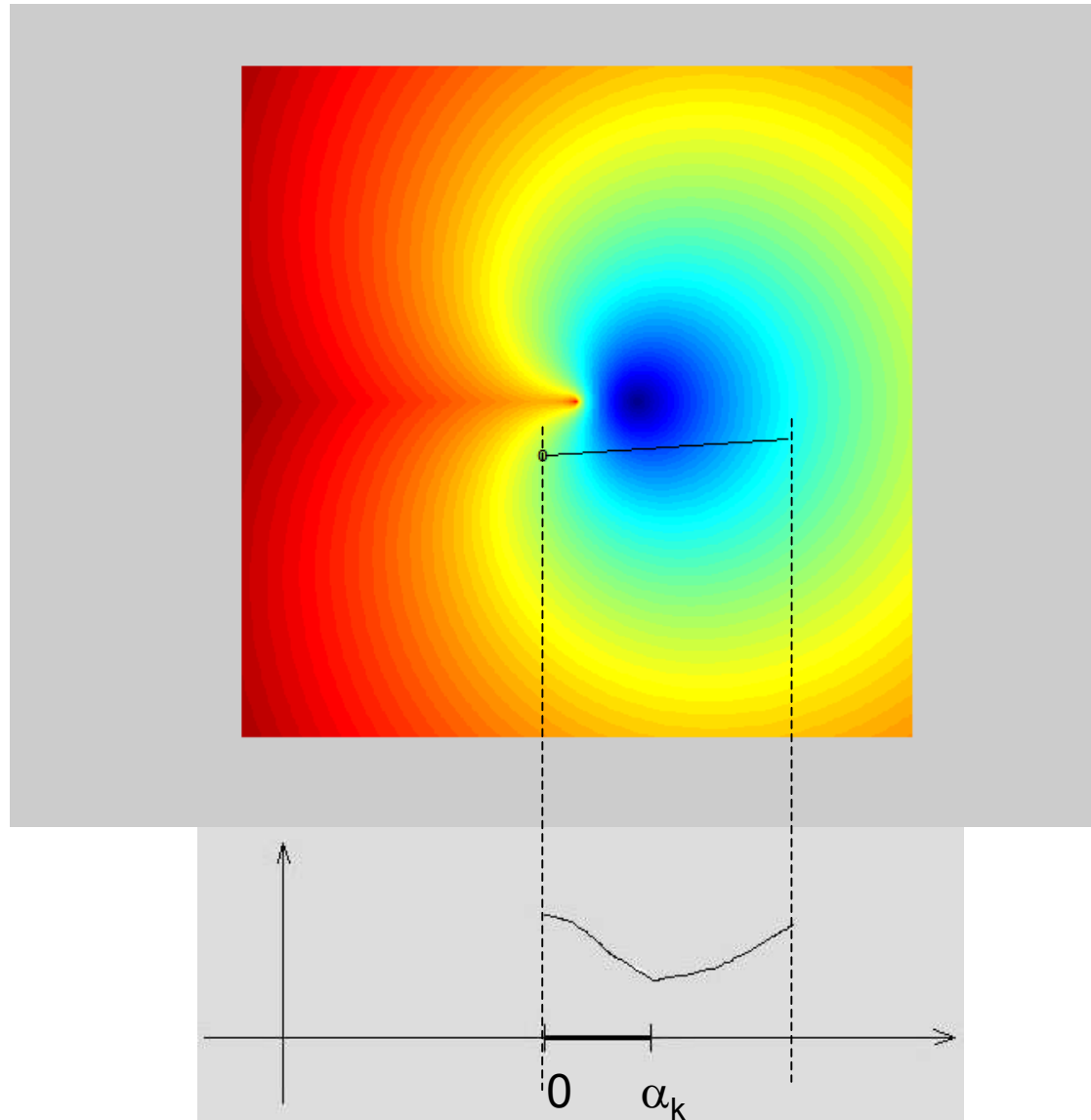
Ilustracja problemu: znajdź $\alpha_k \geq 0$ minimalizujące $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$



Ilustracja problemu: znajdź $\alpha_k \geq 0$ minimalizujące $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$



Ilustracja problemu: znajdź $\alpha_k \geq 0$ minimalizujące $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$



Ilustracja problemu: znajdź $\alpha_k \geq 0$ minimalizujące $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$

Idea niezależnego kierunku poszukiwań i długości kroku

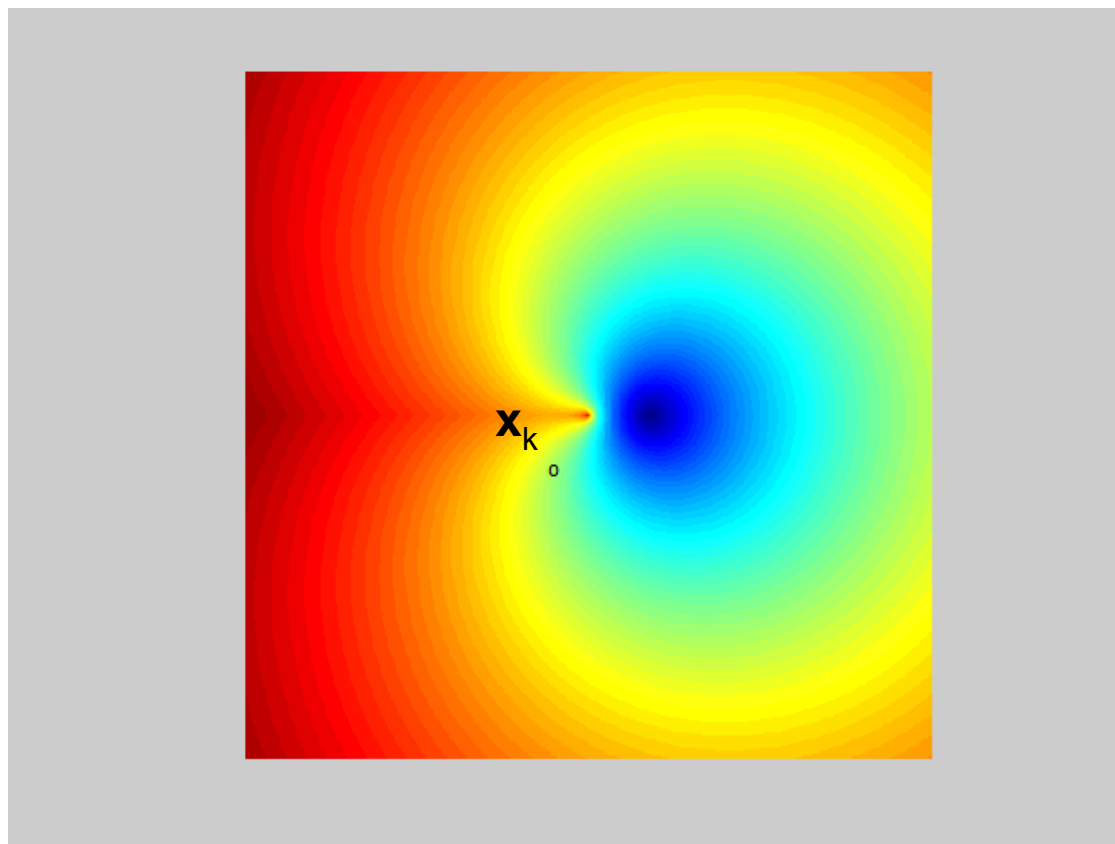
- Przykład tworzenia (jednowymiarowej) funkcji $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$
 - funkcja: $f(\mathbf{x}) = (x_1)^2 + 2(x_2)^2 + 3x_1$
 - $\mathbf{x}_k = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$
 - $\mathbf{d}_k = \begin{bmatrix} -5/2 \\ -2 \end{bmatrix}$
 - wtedy
 - $\mathbf{x}_k + \alpha_k \mathbf{d}_k = \begin{bmatrix} 1 - 5/2 \cdot \alpha_k \\ 2 - 2\alpha_k \end{bmatrix}$
 - $f(\alpha_k) = (1 - 5/2 \cdot \alpha_k)^2 + 2(2 - 2\alpha_k)^2 + 3(1 - 5/2 \cdot \alpha_k) =$
 $= 57/4 \cdot (\alpha_k)^2 - 57/2 \cdot \alpha_k + 12$

Idea niezależnego kierunku poszukiwań i długości kroku

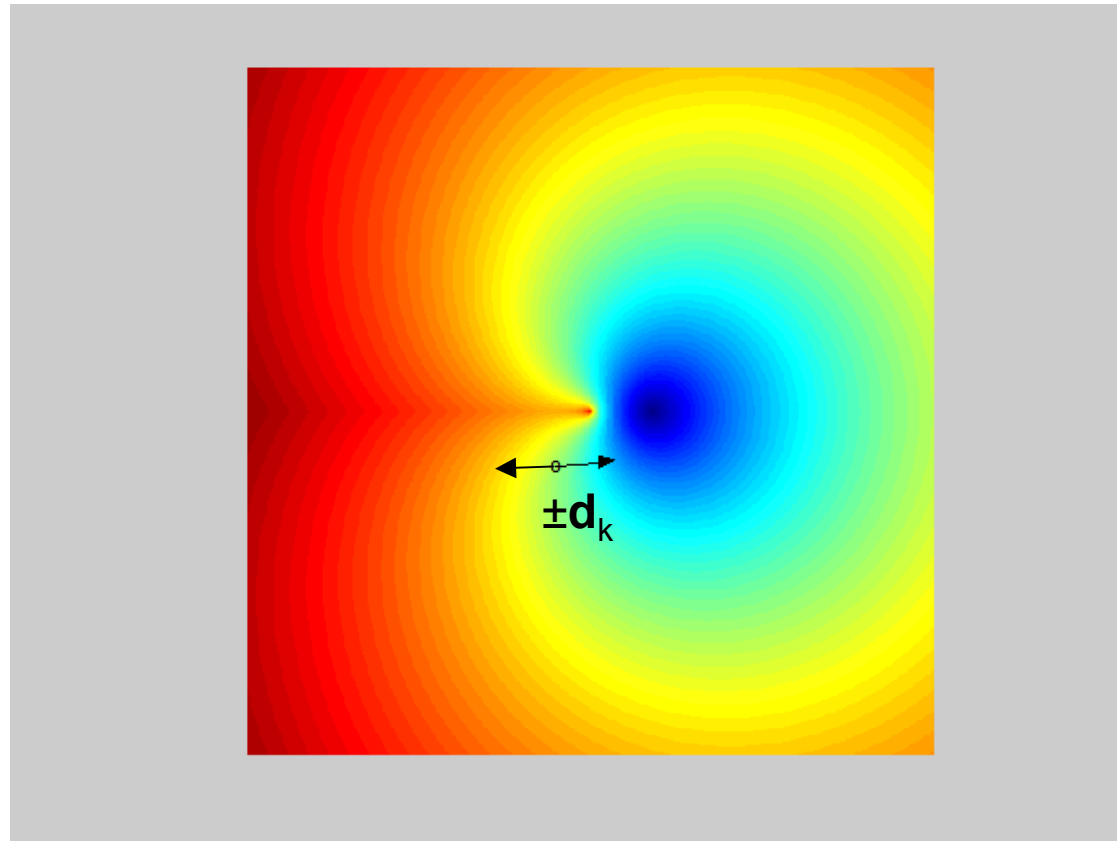
- Ważna implikacja warunku „znajdź $\alpha_k \geq 0 \dots$ ”:
 - \mathbf{d}_k musi być tak dobrany, że aby dotrzeć do minimum funkcji z wektora \mathbf{x}_k należy poruszać się w stronę wyznaczoną wektorem \mathbf{d}_k („do przodu”)
 - a więc przyjęcie np.
 - $\mathbf{d}_k = -\nabla_f(\mathbf{x}_k)$ jest dopuszczalne
 - bo wektor $-\nabla_f(\mathbf{x}_k)$ wskazuje kierunek maksymalnego spadku funkcji
 - znalezione α_k będzie potencjalnie dodatnie
 - $\mathbf{d}_k = \nabla_f(\mathbf{x}_k)$ jest niedopuszczalne
 - bo wektor $\nabla_f(\mathbf{x}_k)$ wskazuje kierunek maksymalnego wzrostu funkcji
 - znalezione α_k będzie zawsze zerem
- Sytuacje szczególne
(powinny być uwzględnione w warunkach stopu metod)
 - $\mathbf{d}_k = \mathbf{0}$
 - $\alpha_k = 0$

Idea niezależnego kierunku poszukiwań i długości kroku

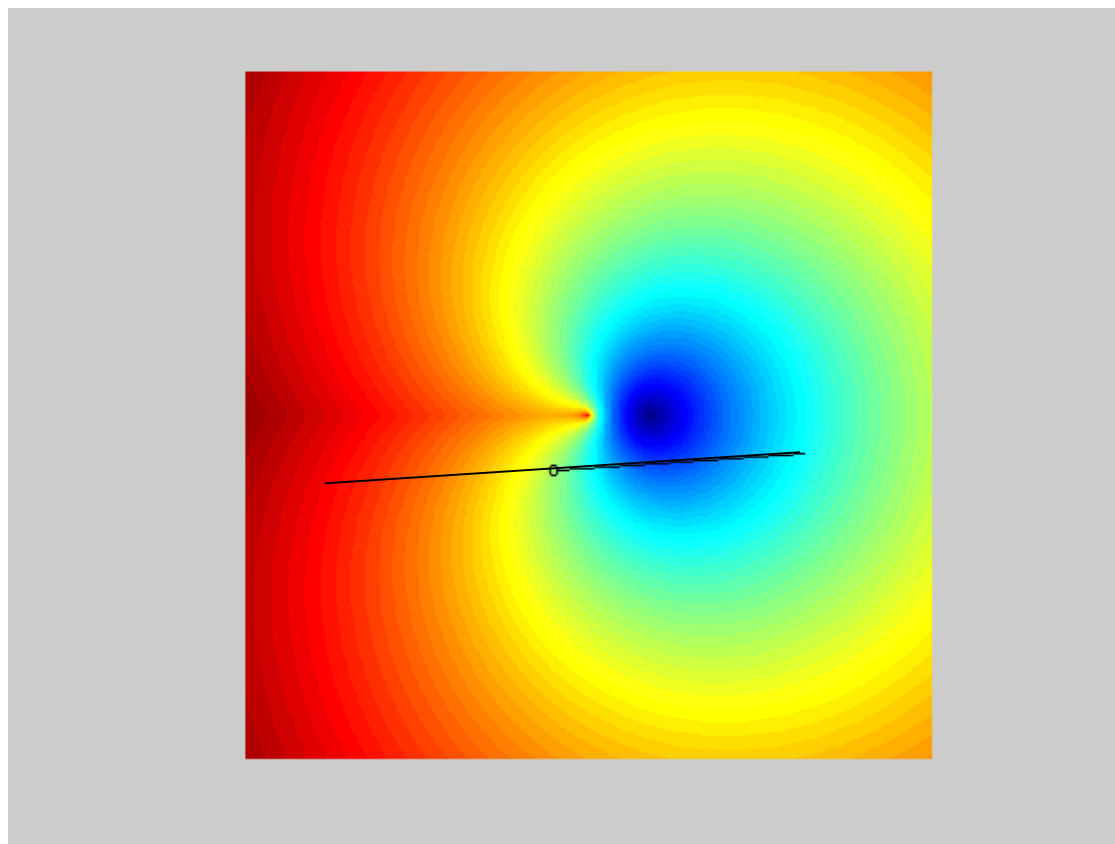
- Poszukiwanie dowolnej (niekoniecznie nieujemnej) długości kroku metody \mathbf{s}_k
 - ustal $\mathbf{d}_k \neq \mathbf{0}$
 - znajdź α_k minimalizujące funkcję $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$
(zamiast: „znajdź $\alpha_k \geq 0$ minimalizujące funkcję $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$ ”)
 - oblicz $\mathbf{s}_k = \alpha_k \mathbf{d}_k$
- Metoda ta zawiera wewnętrzny problem optymalizacji
 - jest to zawsze problem optymalizacji jednowymiarowej bez ograniczeń
 - jako taki może być rozwiązywany np. (optymalizacyjną) metodą Newtona



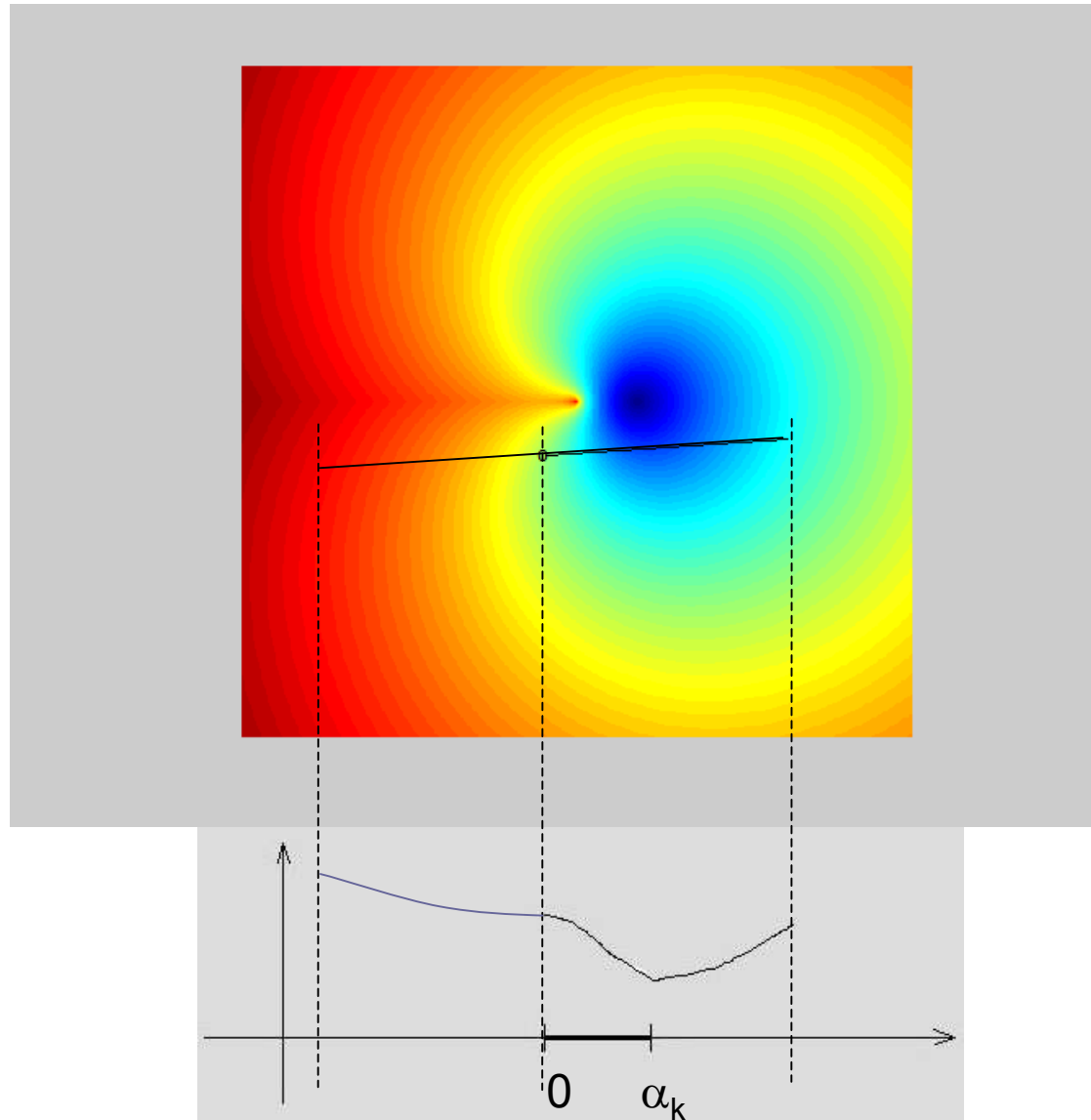
Ilustracja problemu: znajdź α_k minimalizujące $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$



Ilustracja problemu: znajdź α_k minimalizujące $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$



Ilustracja problemu: znajdź α_k minimalizujące $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$



Ilustracja problemu: znajdź α_k minimalizujące $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$

Idea niezależnego kierunku poszukiwań i długości kroku

- Ważna implikacja warunku „znajdź $\alpha_k \dots$ ”:
 - \mathbf{d}_k może być tak dobrany, że aby dotrzeć do minimum funkcji z wektora \mathbf{x}_k należy poruszać się po linii wyznaczonej wektorem \mathbf{d}_k („do przodu” lub „do tyłu”)
 - a więc przyjęcie np.
 - $\mathbf{d}_k = -\nabla_f(\mathbf{x}_k)$ jest dopuszczalne
 - wektor $-\nabla_f(\mathbf{x}_k)$ wskazuje kierunek maksymalnego spadku funkcji
 - znalezione α_k będzie wtedy potencjalnie dodatnie
 - $\mathbf{d}_k = \nabla_f(\mathbf{x}_k)$ jest dopuszczalne
 - wektor $\nabla_f(\mathbf{x}_k)$ wskazuje kierunek maksymalnego wzrostu funkcji
 - znalezione α_k będzie wtedy potencjalnie ujemne
- Sytuacje szczególne
(powinny być uwzględnione w warunkach stopu metod)
 - $\mathbf{d}_k = \mathbf{0}$
 - $\alpha_k = 0$

Idea niezależnego kierunku poszukiwań i długości kroku

- Modyfikacja metody Newtona-Raphsona wykorzystująca optymalizacyjne poszukiwanie długości kroku metody

Algorytm

1. ustal wektor \mathbf{x}_0 i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - ustal \mathbf{d}_k
 - znajdź $\alpha_k \geq 0$ minimalizujące funkcję $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$
 - oblicz $\mathbf{s}_k = \alpha_k \mathbf{d}_k$
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
 - podstaw $k = k + 1$

(podstawowy algorytm tzw. uogólnionych metod newtonowskich)

...

Metoda Newtona-Raphsona

- Metoda Newtona-Raphsona

Algorytm

1. ustal wektor \mathbf{x}_0 i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - oblicz $\mathbf{s}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
 - podstaw $k = k + 1$

Popularne modyfikacje (w skrócie)

- Uogólniona metoda Newtona

Algorytm

1. ustal wektor \mathbf{x}_0 i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - oblicz $\mathbf{d}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1} \nabla_f(\mathbf{x}_k)$
 - znajdź $\alpha_k \geq 0$ minimalizujące funkcję $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$
 - oblicz $\mathbf{s}_k = \alpha_k \mathbf{d}_k$
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
 - podstaw $k = k + 1$

(metoda charakteryzuje się ulepszoną zbieżnością)

Popularne modyfikacje (w skrócie)

- Metoda Cauchy'ego

Algorytm

1. ustal wektor \mathbf{x}_0 i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - oblicz $\mathbf{d}_k = -\nabla_f(\mathbf{x}_k)$
 - znajdź $\alpha_k \geq 0$ minimalizujące funkcję $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$
 - oblicz $\mathbf{s}_k = \alpha_k \mathbf{d}_k$
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
 - podstaw $k = k + 1$

(uproszczona wersja uogólnionej metody Newtona;
nie wymaga znajomości hesjanu ani drugich pochodnych)

Ekstra₁:

- Metoda spadku gradientu

Algorytm

1. ustal wektor \mathbf{x}_0 , długość kroku α i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - oblicz $\mathbf{d}_k = -\nabla_f(\mathbf{x}_k)$
 - oblicz $\mathbf{s}_k = \alpha \mathbf{d}_k$
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
 - podstaw $k = k + 1$

(uproszczona wersja metody Cauchy'ego; brak wewnętrznego problemu optymalizacji jednowymiarowej jednocześnie upodabnia tę metodę do uproszczonej wersji metody Newtona-Raphsona)

Popularne modyfikacje (w skrócie)

- Metoda Levenberga-Marquarda (ujednolicona)

Algorytm

1. ustal wektor \mathbf{x}_0 i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - znajdź $\varepsilon \geq 0$ gwarantujące $\mathbf{H}_f(\mathbf{x}_k) + \varepsilon \mathbf{I} > 0$
 - oblicz $\mathbf{d}_k = -(\mathbf{H}_f(\mathbf{x}_k) + \varepsilon \mathbf{I})^{-1} \nabla_f(\mathbf{x}_k)$
 - znajdź $\alpha_k \geq 0$ minimalizujące funkcję $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$
 - oblicz $\mathbf{s}_k = \alpha_k \mathbf{d}_k$
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
 - podstaw $k = k + 1$

(metoda gwarantuje zbieżność dla każdego wektora początkowego)

Ekstra₂:

- Metoda Levenberga-Marquarda (ujednolicona, właściwa)

Algorytm

1. ustal wektor \mathbf{x}_0 i podstaw $k = 0$
2. dopóki nie zachodzi warunek stopu, wykonuj:
 - znajdź $\varepsilon \geq 0$ gwarantujące $\mathbf{H}_f(\mathbf{x}_k) + \varepsilon \mathbf{I} > 0$
 - oblicz $\mathbf{d}_k = -(\mathbf{H}_f(\mathbf{x}_k) + \varepsilon \mathbf{I} + \mu \text{diag}(\mathbf{H}_f(\mathbf{x}_k)))^{-1} \nabla_f(\mathbf{x}_k)$ dla $\mu \geq 0$
 - znajdź $\alpha_k \geq 0$ minimalizujące funkcję $f(\alpha_k) = f(\mathbf{x}_k + \alpha_k \mathbf{d}_k)$
 - oblicz $\mathbf{s}_k = \alpha_k \mathbf{d}_k$
 - oblicz $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$
 - podstaw $k = k + 1$

(szybsza zbieżność)

...

Metoda Levenberga-Marquarda

- ... ujednolicona, właściwa... czyli jaka?
- Czyli (ogólnie) taka, w której dodaje się coś (konkretnie: macierz diagonalną) do hesjanu (ponieważ dodawana macierz jest diagonalna, operacja ta ma wpływ jedynie na przekątną hesjanu)
- W praktyce dodaje się:
 - przeskalowaną współczynnikiem ε macierz \mathbf{I}
 - przeskalowaną współczynnikiem μ macierz $\text{diag}(\mathbf{H}_f)$
 - obie z powyższych

Metoda Levenberga-Marquarda

- Przykłady

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} + 0.01 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1.01 & 2 & 3 \\ 4 & 5.01 & 6 \\ 7 & 8 & 9.01 \end{bmatrix}$$

Metoda Levenberga-Marquarda

- Przykłady

$$\text{diag}\left(\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}\right) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 9 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} + 0.1 \cdot \text{diag}\left(\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}\right) = \begin{bmatrix} 1.1 & 2 & 3 \\ 4 & 5.5 & 6 \\ 7 & 8 & 9.9 \end{bmatrix}$$

Metoda Levenberga-Marquarda

- Przykłady

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} + 0.01 \cdot \text{diag} \left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) + 0.1 \cdot \text{diag} \left(\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \right) = \begin{bmatrix} 1.11 & 2 & 3 \\ 4 & 5.51 & 6 \\ 7 & 8 & 9.91 \end{bmatrix}$$

Metoda Levenberga-Marquarda

- Pytanie: co się dzieje, gdy do dowolnej symetrycznej macierzy \mathbf{A} dodamy macierz $\varepsilon\mathbf{I}$?
 - przyspieszony kurs algebry/analizy (w tym spektralnej) macierzy (kwadratowych):
 - ...
 - macierz $\mathbf{A}_{n \times n}$ posiada n (niekoniecznie różnych) wartości charakterystycznych, zwanych wartościami własnymi
 - iloczyn wszystkich wartości własnych jest równy wyznacznikowi macierzy, co oznacza, że gdy choć jedna wartość własna jest zerowa, to macierz jest osobliwa
 - macierz osobliwa (zerowy wyznacznik) nie posiada odwrotności
 - jeżeli macierz \mathbf{A} posiada wartości własne λ_i , to macierz $\mathbf{A} + \varepsilon\mathbf{I}$ posiada wartości własne $\lambda_i + \varepsilon$
 - dodanie do przekątnej macierzy dowolnej wartości ε zmienia (gdy $\varepsilon > 0$: zwiększa, gdy $\varepsilon < 0$: zmniejsza) wszystkie wartości własne o ε
 - ...
 - wszystkie wartości własne macierzy symetrycznej są rzeczywiste
 - ...

Metoda Levenberga-Marquarda

- Pytanie: co się dzieje, gdy do dowolnej symetrycznej macierzy \mathbf{A} dodamy macierz $\varepsilon\mathbf{I}$?
- Odpowiedź:
 - dodanie do przekątnej macierzy \mathbf{A} odpowiednio dobranej wartości ε gwarantuje, że macierz staje się nieosobliwa
 - uzasadnienie:
 - (korzystamy z)
„jeżeli macierz \mathbf{A} posiada wartości własne λ_i ,
to macierz $\mathbf{A} + \varepsilon\mathbf{I}$ posiada wartości własne $\lambda_i + \varepsilon$ ”
 - możliwe jest więc takie dobranie ε , aby wszystkie $\lambda_i + \varepsilon$ były niezerowe
 - » np. $\varepsilon = 1 - \min(\lambda_i)$

Metoda Levenberga-Marquarda

- Pytanie: co się dzieje, gdy do dowolnej macierzy nieujemnie określonej \mathbf{A} dodamy macierz $\varepsilon \mathbf{I}$?
 - c.d. przyspieszonego kursu algebry/analizy (w tym spektralnej) macierzy (kwadratowych):
 - ...
 - macierz nazywamy macierzą nieujemnie/dodatnio określoną gdy jest symetryczna i ...
 - (macierz nieujemnie/dodatnio określona jest więc z definicji symetryczna)
 - ...
 - symetryczna macierz jest nieujemnie określona wtedy i tylko wtedy, gdy wszystkie jej wartości własne są nieujemne
 - symetryczna macierz jest dodatnio określona wtedy i tylko wtedy, gdy wszystkie jej wartości własne są dodatnie
 - macierz dodatnio określona jest więc macierzą nieujemnie określoną
 - ...

Metoda Levenberga-Marquarda

- Pytanie: co się dzieje, gdy do dowolnej macierzy nieujemnie określonej \mathbf{A} dodamy macierz $\varepsilon\mathbf{I}$?
- Odpowiedź:
 - dodanie do przekątnej nieujemnie określonej macierzy \mathbf{A} dowolnej dodatniej wartości ε gwarantuje, że macierz staje się dodatnio określona (a tym samym nieosobliwa)
 - uzasadnienie:
 - (korzystamy z)
„jeżeli macierz \mathbf{A} posiada wartości własne λ_i ,
to macierz $\mathbf{A} + \varepsilon\mathbf{I}$ posiada wartości własne $\lambda_i + \varepsilon$ ”
 - wartości własne λ_i macierzy nieujemnie określonej są nieujemne
 - dla dowolnego $\varepsilon > 0$, wszystkie $\lambda_i + \varepsilon$ są dodatnie (a więc niezerowe)

Metoda Levenberga-Marquarda

- Pytanie: co się dzieje, gdy do dowolnej symetrycznej macierzy \mathbf{A} dodamy macierz $\mu \text{diag}(\mathbf{A})$?
 - ...

Wartości własne macierzy kwadratowej

- Dalsze właściwości wartości własnych macierzy kwadratowej
 - (Weyl) jeżeli macierze \mathbf{A} i \mathbf{B} o rozmiarach $n \times n$ są rzeczywiste i symetryczne, a ich wartościami własnymi są (odpowiednio) $\alpha_1 \leq \dots \leq \alpha_n$ oraz $\beta_1 \leq \dots \leq \beta_n$, to $\mathbf{G} = \mathbf{A} + \mathbf{B}$ o rozmiarach $n \times n$ jest rzeczywista i symetryczna, a jej wartościami własnymi są γ_i spełniające
$$\alpha_i + \beta_1 \leq \gamma_i \leq \alpha_i + \beta_n \iff \alpha_i + \min(\beta_i) \leq \gamma_i \leq \alpha_i + \max(\beta_i)$$
oraz
$$\beta_i + \alpha_1 \leq \gamma_i \leq \beta_i + \alpha_n \iff \beta_i + \min(\alpha_i) \leq \gamma_i \leq \beta_i + \max(\alpha_i)$$

...

Ujednolicona a nieujednolicona metoda Levenberga-Marquarda

- Niech \mathbf{A} będzie nieosobliwą macierzą o rozmiarach $n \times n$, a \mathbf{b} wektorem o rozmiarze $n \times 1$; należy obliczyć $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$
 - jak to zrobić?
- Możliwe rozwiązanie
 - znaleźć \mathbf{A}^{-1}
 - przemnożyć \mathbf{A}^{-1} przez \mathbf{b}
- Czy istnieje lepszy (w znaczeniu: mniej pracochłonny) sposób?
 - tak!
 - jaki?

Ujednolicona a nieujednolicona metoda Levenberga-Marquarda

- Idea lepszego sposobu
 - niech $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$
 - twierdzenie
 - wektor \mathbf{x} spełnia zależność $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ wtedy i tylko wtedy, gdy spełnia zależność $\mathbf{Ax} = \mathbf{b}$
 - uzasadnienie
 - $\mathbf{Ax} = \mathbf{b} \Rightarrow \mathbf{A}^{-1}\mathbf{Ax} = \mathbf{A}^{-1}\mathbf{b} \Leftrightarrow \mathbf{Ix} = \mathbf{A}^{-1}\mathbf{b} \Leftrightarrow \mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$
 - $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b} \Rightarrow \mathbf{Ax} = \mathbf{AA}^{-1}\mathbf{b} \Leftrightarrow \mathbf{Ax} = \mathbf{Ib} \Leftrightarrow \mathbf{Ax} = \mathbf{b}$
 - a więc $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ można obliczać znajdując rozwiązanie układu równań $\mathbf{Ax} = \mathbf{b}$, w którym \mathbf{x} jest wektorem zmiennych
 - wynik jest jednoznaczny dla nieosobliwej macierzy \mathbf{A} (czyli gdy istnieje \mathbf{A}^{-1})
 - osiągnąć zysk jest taki, że do rozwiązania układu równań $\mathbf{Ax} = \mathbf{b}$ potrzeba mniej operacji niż do znalezienia odwrotności \mathbf{A}^{-1} macierzy \mathbf{A} i wykonania mnożenia $\mathbf{A}^{-1}\mathbf{b}$
 - dodatkowo, mogą pojawiać się zyski natury numerycznej (im mniej operacji, tym mniejszy błąd numeryczny)

Ujednolicona a nieujednolicona metoda Levenberga-Marquarda

- Podsumowując: w klasycznym sformułowaniu metody Levenberga-Marquarda kierunek poszukiwań \mathbf{d}_k , zadany równością $\mathbf{d}_k = -(\mathbf{H}_f(\mathbf{x}_k) + \varepsilon\mathbf{I})^{-1}\nabla_f(\mathbf{x}_k)$, znajduje się rozwiązując układ równań $(\mathbf{H}_f(\mathbf{x}_k) + \varepsilon\mathbf{I})\mathbf{d}_k = -\nabla_f(\mathbf{x}_k)$, co prowadzi do uzyskania identycznego wektora \mathbf{d}_k
 - gdy $\mathbf{H}_f(\mathbf{x}_k) + \varepsilon\mathbf{I} > 0$, to jest nieosobliwa, a to implikuje jednoznaczność rozwiązania
- Stosowane w ramach wykładu ujednolicenie (w sensie: ujednolicenie zapisu) tej metody oznacza przede wszystkim, że obliczanie kierunku poszukiwań \mathbf{d}_k zapisuje się jako „oblicz $\mathbf{d}_k = -(\mathbf{H}_f(\mathbf{x}_k) + \varepsilon\mathbf{I})^{-1}\nabla_f(\mathbf{x}_k)$ ”, (co ujednolica ten zapis z opisami pozostałych metod) a nie jako „znajdź \mathbf{d}_k rozwiązując układ równań $(\mathbf{H}_f(\mathbf{x}_k) + \varepsilon\mathbf{I})\mathbf{d}_k = -\nabla_f(\mathbf{x}_k)$ ”, (co jest znacznie praktyczniejsze z obliczeniowego punktu widzenia)

Ujednolicona a nieujednolicona metoda Levenberga-Marquarda

- Dodatkowa uwaga: obliczanie wektora zadanego równaniem $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$ poprzez rozwiązanie układu równań $\mathbf{Ax} = \mathbf{b}$ można stosować także w pozostałych metodach, np. w metodzie Newtona-Raphsona
 - schemat iteracyjny: $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$, z krokiem $\mathbf{s}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$
 - wektor \mathbf{s}_k można obliczać mnożąc obie strony równania $\mathbf{s}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$ przez $(\mathbf{H}_f(\mathbf{x}_k))^{-1}$, otrzymując układ równań $(\mathbf{H}_f(\mathbf{x}_k))\mathbf{s}_k = -\nabla_f(\mathbf{x}_k)$, gdzie \mathbf{s}_k jest wektorem zmiennych, i rozwiązując powstały układ
 - w rezultacie nie trzeba znajdować $(\mathbf{H}_f(\mathbf{x}_k))^{-1}$ ani wykonywać mnożenia $(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$
 - wektor \mathbf{x}_{k+1} można obliczać mnożąc obie strony równania $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$, w postaci $\mathbf{x}_{k+1} = \mathbf{x}_k - (\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$ przez $(\mathbf{H}_f(\mathbf{x}_k))^{-1}$, otrzymując układ równań $\mathbf{H}_f(\mathbf{x}_k)\mathbf{x}_{k+1} = \mathbf{H}_f(\mathbf{x}_k)\mathbf{x}_k - \nabla_f(\mathbf{x}_k)$, gdzie \mathbf{x}_{k+1} jest wektorem zmiennych, i rozwiązując powstały układ
 - w rezultacie nie trzeba znajdować $(\mathbf{H}_f(\mathbf{x}_k))^{-1}$, wykonywać mnożenia $(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$, ani odejmowania $\mathbf{x}_k - (\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$

...

Właściwa a niewłaściwa metoda Levenberga-Marquarda

- Wersja 0 (Newtona-Raphsona /ujednolicona/)
 - oblicz $\mathbf{d}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$
- Wersja 1 (Levenberga, tutaj: Levenberga-Marquarda /ujednolicona/)
 - oblicz $\mathbf{d}_k = -(\mathbf{H}_f(\mathbf{x}_k) + \varepsilon\mathbf{I})^{-1}\nabla_f(\mathbf{x}_k)$ dla $\varepsilon \geq 0$
 - element $\varepsilon\mathbf{I}$ gwarantuje odwracalność macierzy
- Wersja 2 (Marquarda /ujednolicona/)
 - oblicz $\mathbf{d}_k = -(\mathbf{H}_f(\mathbf{x}_k) + \mu\text{diag}(\mathbf{H}_f(\mathbf{x}_k)))^{-1}\nabla_f(\mathbf{x}_k)$ dla $\mu \geq 0$
 - element $\mu\text{diag}(\mathbf{H}_f(\mathbf{x}_k))$ poprawia zbieżność
- Wersja 3 (właściwa /ujednolicona/)
 - oblicz $\mathbf{d}_k = -((\mathbf{H}_f(\mathbf{x}_k) + \varepsilon\mathbf{I}) + \mu\text{diag}(\mathbf{H}_f(\mathbf{x}_k)))^{-1}\nabla_f(\mathbf{x}_k)$ dla $\varepsilon \geq 0$ i $\mu \geq 0$

...

Parametryzacja metody Levenberga-Marquarda (ujednoliconej, niewłaściwej)

- Rola parametru $\varepsilon > 0$ przy ustalaniu wartości macierzy $\mathbf{H}_f(\mathbf{x}_k) + \varepsilon \mathbf{I}$
 - jakie wartości może (w ekstremalnych sytuacjach) przyjmować parametr ε ?

Parametryzacja metody Levenberga-Marquarda (ujednoliconej, właściwej)

- Oznaczenia
 - $\mathbf{d} \equiv \mathbf{d}_k$
 - $\mathbf{g} \equiv \nabla_f(\mathbf{x}_k)$
 - $\mathbf{P} \equiv \mathbf{H}_f(\mathbf{x}_k) + \varepsilon \mathbf{I}$
gdzie wartość $\varepsilon \geq 0$ jest tak dobrana, że $\mathbf{P} > 0$ (dodatnio określona)
- Przy tych oznaczeniach kierunek poszukiwań przedstawia się jako
 $\mathbf{d} = -\mathbf{P}^{-1}\mathbf{g}$

Parametryzacja metody Levenberga-Marquarda (ujednoliconej, właściwej)

- Rola parametru ε przy ustalaniu wartości macierzy $\mathbf{P} \equiv \mathbf{H}_f(\mathbf{x}_k) + \varepsilon \mathbf{I}$
 - niech $\varepsilon \rightarrow 0$
 - wtedy $\mathbf{P} \rightarrow \mathbf{H}_f(\mathbf{x}_k)$ oraz $\mathbf{d} = -\mathbf{P}^{-1}\mathbf{g} \rightarrow -(\mathbf{H}_f(\mathbf{x}_k))^{-1}\mathbf{g}$
(stosując pełne oznaczenia: $\mathbf{d}_k = -(\mathbf{H}_f(\mathbf{x}_k))^{-1}\nabla_f(\mathbf{x}_k)$)
 - wniosek: (ujednolicona) metoda Levenberga-Marquarda sprowadza się w tym przypadku do uogólnionej metody Newtona

Parametryzacja metody Levenberga-Marquarda (ujednoliconej, właściwej)

- Rola parametru ε przy ustalaniu wartości macierzy $\mathbf{P} \equiv \mathbf{H}_f(\mathbf{x}_k) + \varepsilon \mathbf{I}$
 - niech $\varepsilon \rightarrow \infty$
 - wtedy $\mathbf{P} \rightarrow \varepsilon \mathbf{I}$ oraz $\mathbf{d} = -\mathbf{P}^{-1} \mathbf{g} \rightarrow -(\varepsilon \mathbf{I})^{-1} \mathbf{g} = -(1/\varepsilon) \mathbf{I}^{-1} \mathbf{g} = -(1/\varepsilon) \mathbf{I} \mathbf{g} = -(1/\varepsilon) \mathbf{g}$
(stosując pełne oznaczenia: $\mathbf{d}_k = -(1/\varepsilon) \nabla_f(\mathbf{x}_k)$)
 - uwaga: ponieważ wektor \mathbf{d}_k definiuje jedynie kierunek poszukiwań, jego długość jest nieistotna, a więc każdy dodatni skalar s w wyrażeniu $\mathbf{d}_k = -s \nabla_f(\mathbf{x}_k)$ może być pominięty
 - wobec powyższej uwagi i warunku $\varepsilon \geq 0$, kierunek poszukiwań można zapisać jako $\mathbf{d}_k = -\nabla_f(\mathbf{x}_k)$
 - wniosek: (ujednolicona) metoda Levenberga-Marquarda sprowadza się w tym przypadku do metody Cauchy'ego

...