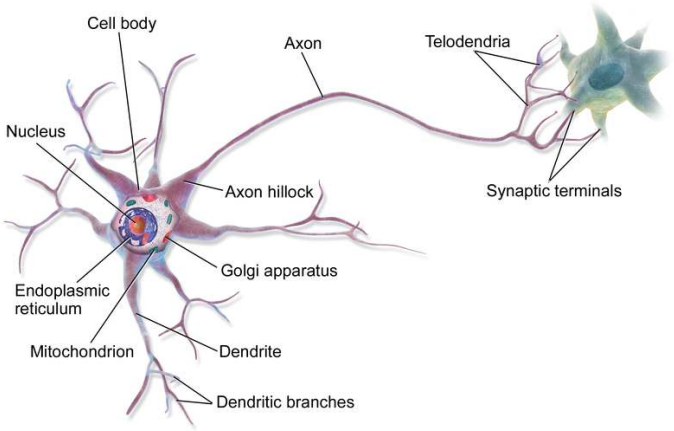
Perceptron

Przemysław Głomb

Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk

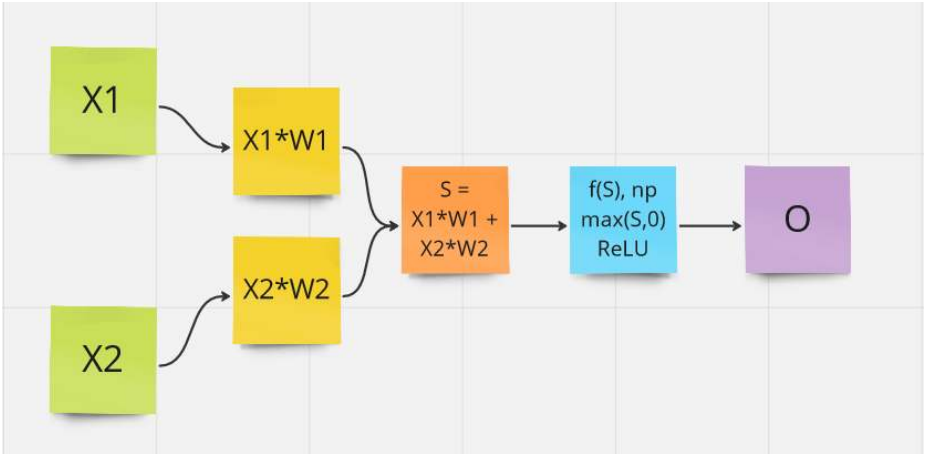
redaktor pomocniczy: Bartłomiej Gardas

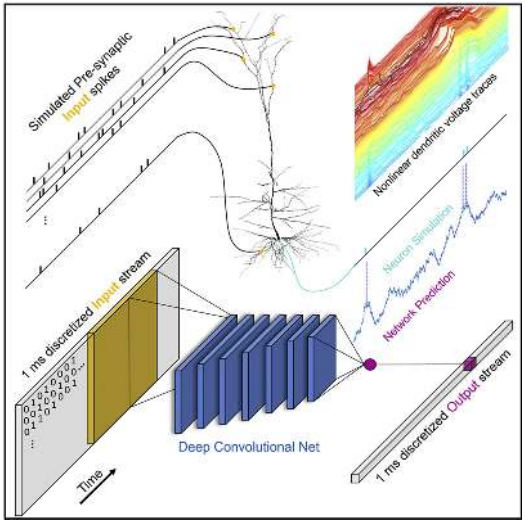
Neuron



https://en.wikipedia.org/wiki/File:Blausen_0657_MultipolarNeuron.png, User:BruceBlaus, CC 3

Sztuczny neuron (perceptron)

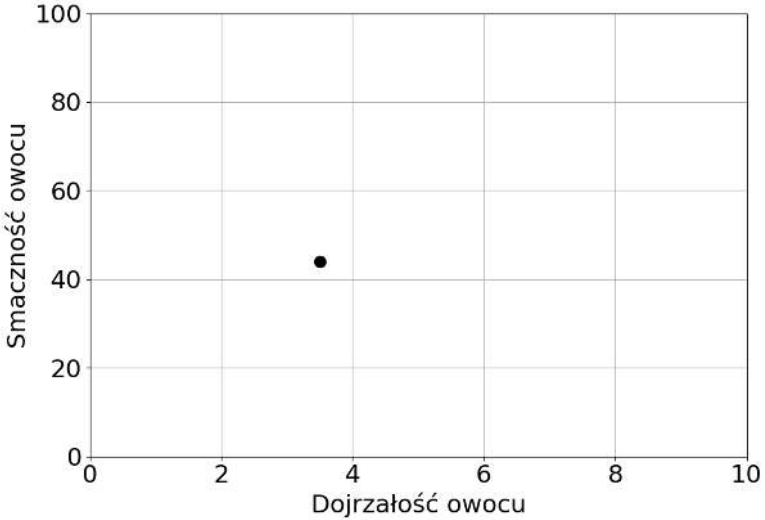


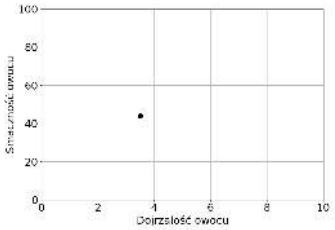


D. Beniaguev, I. Segev, M. London, Single cortical neurons as deep artificial neural networks, Neuron, Volume 109, Issue 17, 2021, Pages 2727-2739.e3,

Przykład („toy example”) owoce

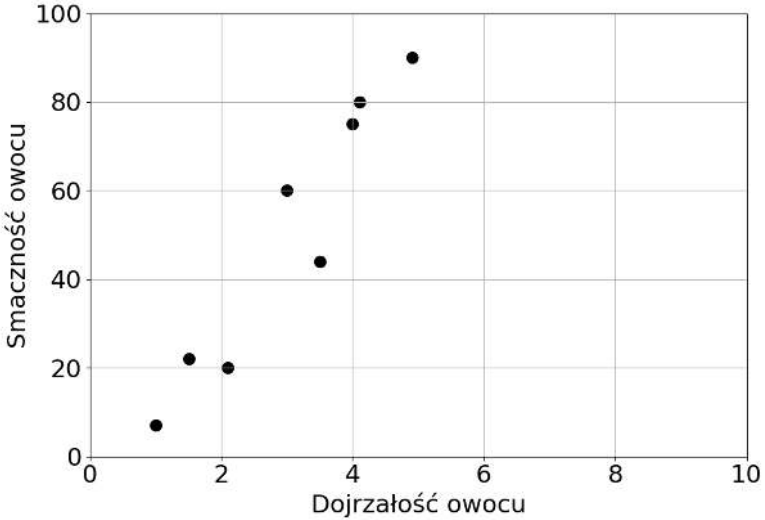
https://en.wikipedia.org/wiki/File:Veraison_Cabernet_Sauvignon.jpg, User:Dominique Hessel, CC 3


100
80
60
40
20
0
Smaczność owocu
0
2
4
6
8
10
Dojrzałość owocu

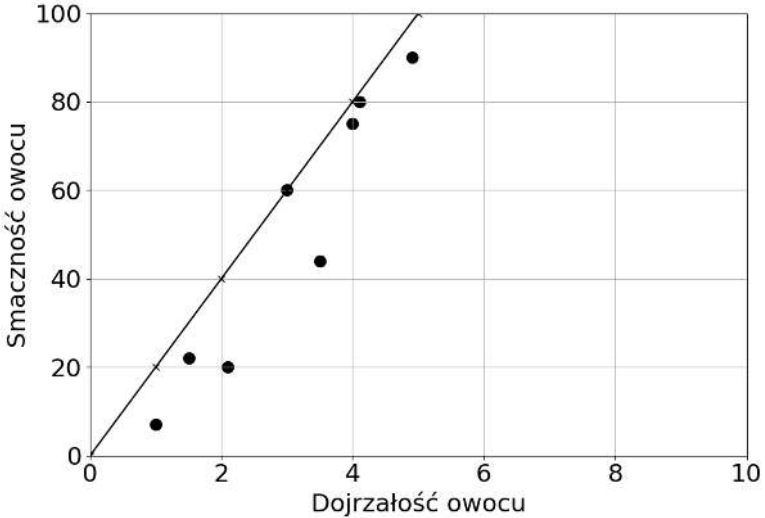


$y = f(wx + b)$, x dojrzałość, y smaczność

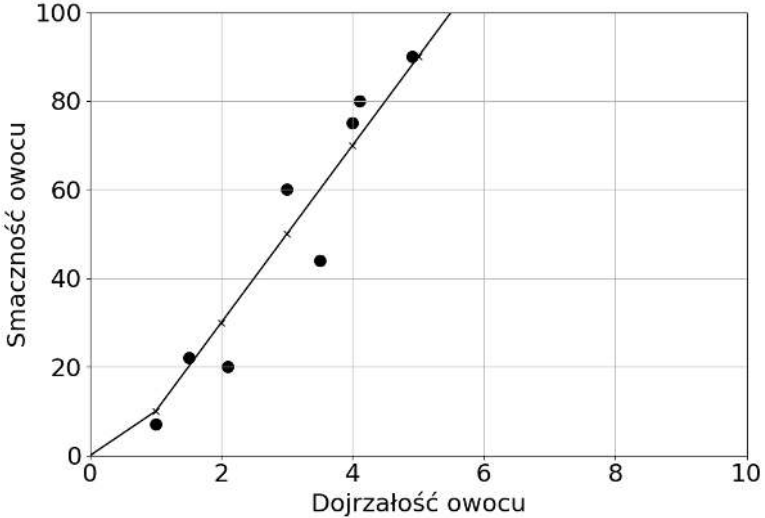
np. $f(a) = \max(a, 0)$ (ReLU), $f(a) = \frac{1}{1+e^{-a}}$ (sigmoid), $f(a) = \frac{(e^a - e^{-a})}{(e^a + e^{-a})}$ (tanh)


Smaczność owocu
Dojrzałość owocu
0
20
40
60
80
100
0
2
4
6
8
10

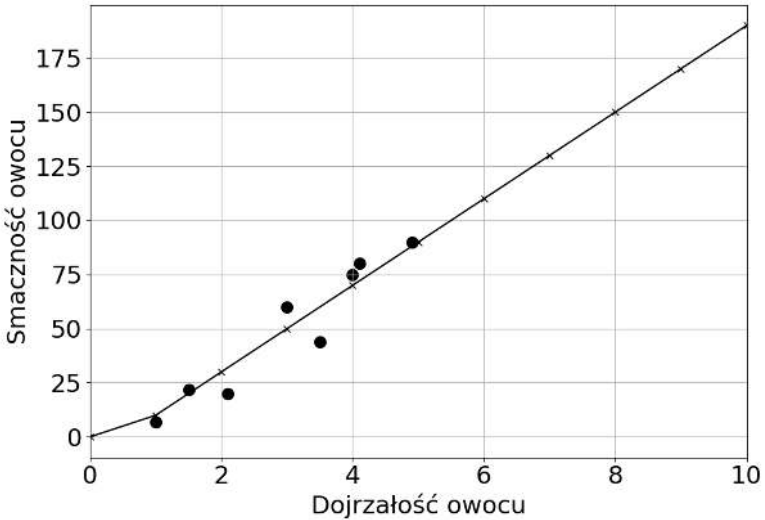
$$y = 20x$$



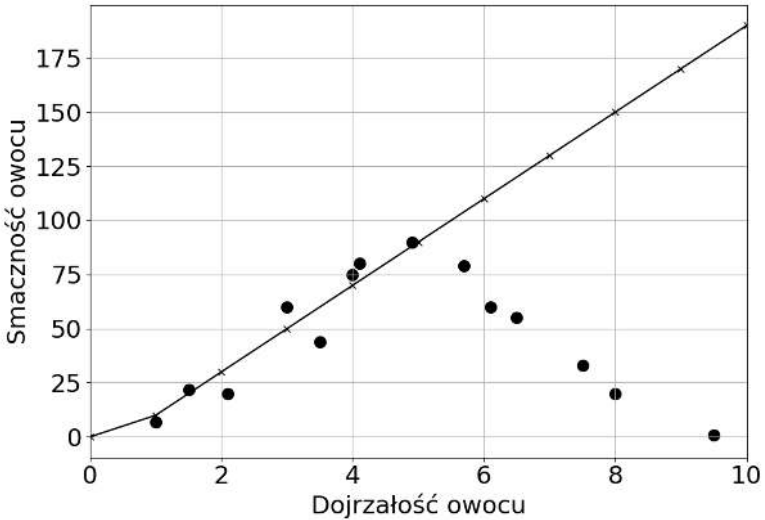
$$y = \max(20x - 10, 0)$$



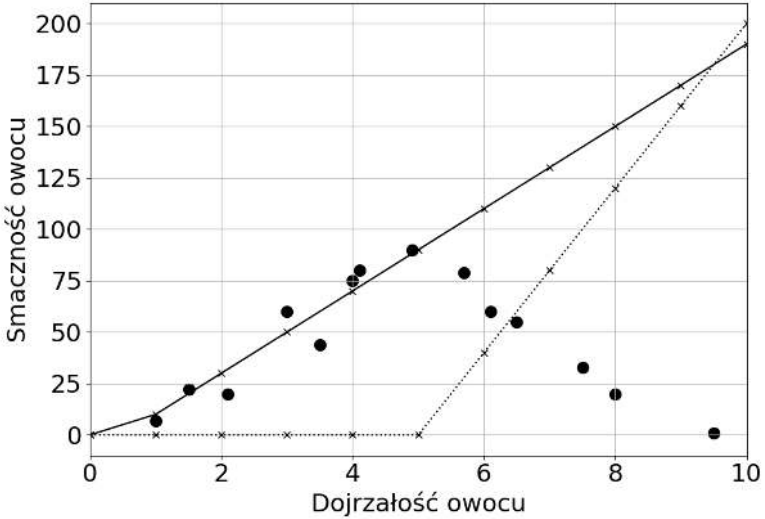
$$y = \max(20x - 10, 0)$$


Smaczność owocu
Dojrzałość owocu
0
25
50
75
100
125
150
175
0
2
4
6
8
10

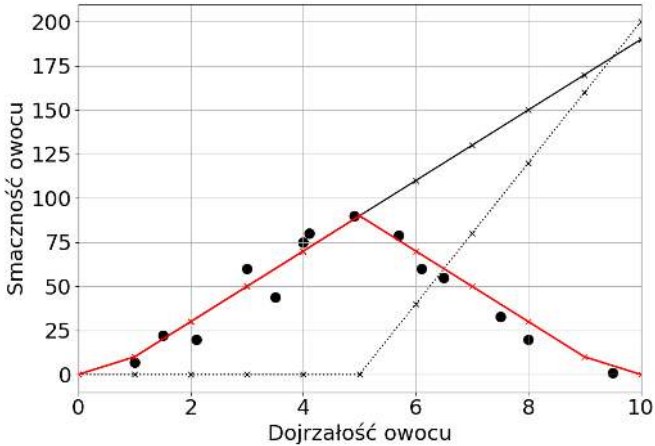
$$y = \max(20x - 10, 0)$$


Smaczność owocu
Dojrzałość owocu
0
25
50
75
100
125
150
175
0
2
4
6
8
10

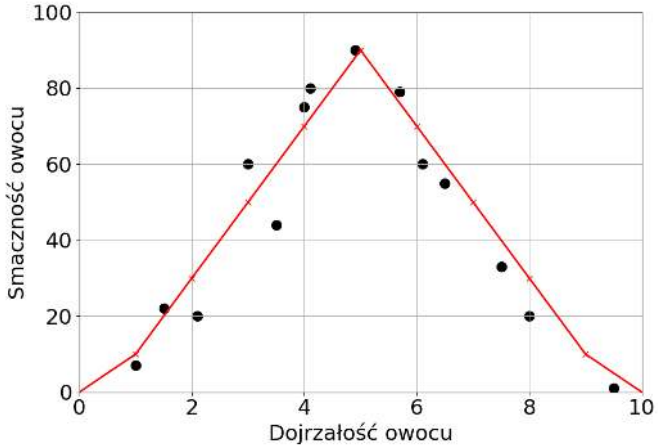
$$y = \max(20x - 10, 0) \quad y_2 = \max(40x - 200, 0)$$



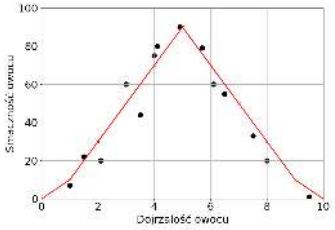
$$y = \max(20x - 10, 0) \quad y_2 = \max(40x - 200, 0)$$
$$y_{\text{out}} = \max(y - y_2, 0)$$



$$y = \max(20x - 10, 0) \quad y_2 = \max(40x - 200, 0)$$
$$y_{\text{out}} = \max(y - y_2, 0)$$



$$y = \max(20x - 10, 0) \quad y_2 = \max(40x - 200, 0)$$
$$y_{\text{out}} = \max(1y - 1y_2 + 0, 0)$$



wynik działa :)

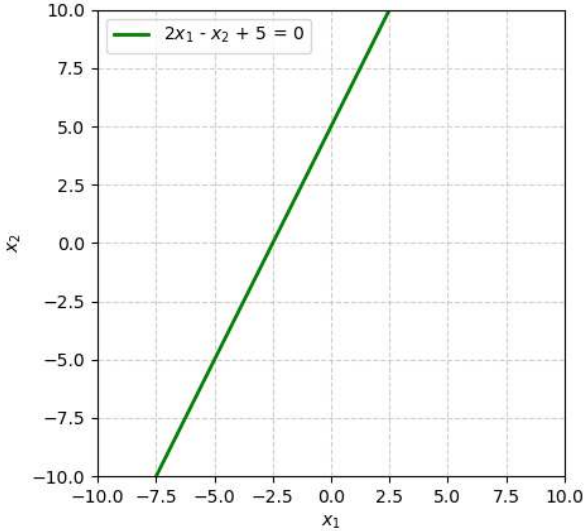
$x \to y \qquad x \to y_2$

$y \to y_{\text{out}} \leftarrow y_2$

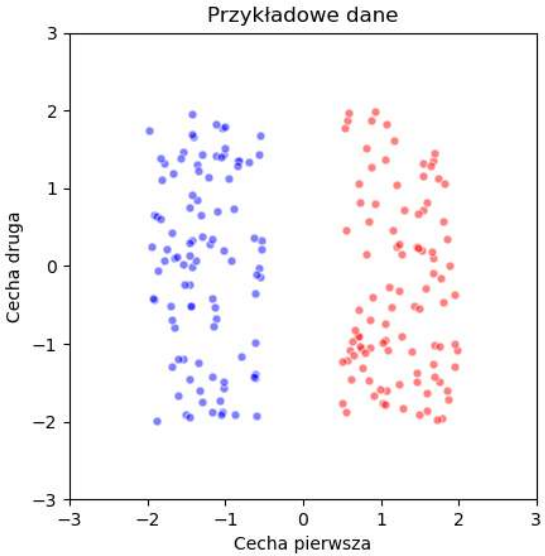
model architektura (schemat połączeń neuronów) oraz wagi $(20, -10, 40, -200, 1, -1, 0)$

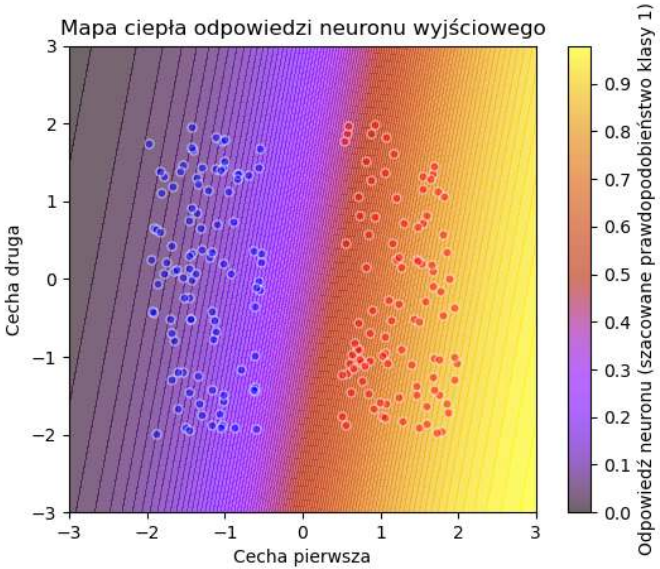
dodatkowo zbiór danych / model dopasowany do zbioru danych; hierarchie cech

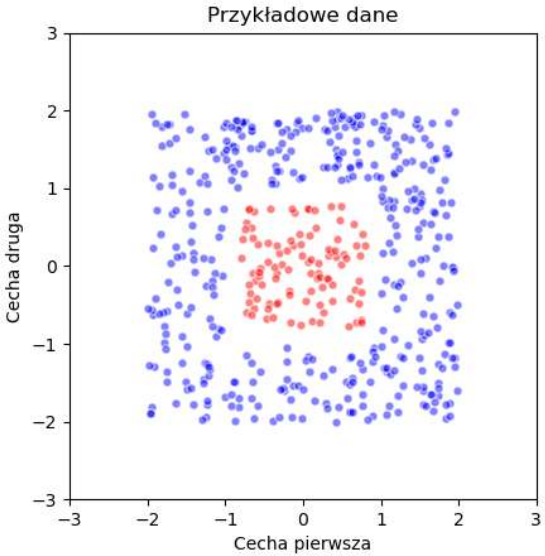
Hierarchie cech

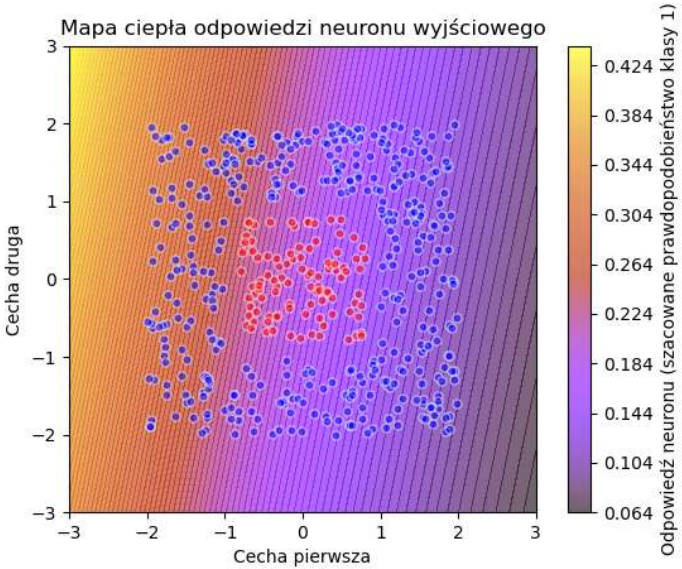

$2x_1 - x_2 + 5 = 0$
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
x_2
−10.0 −7.5 −5.0 −2.5 0.0 2.5 5.0 7.5 10.0
x_1

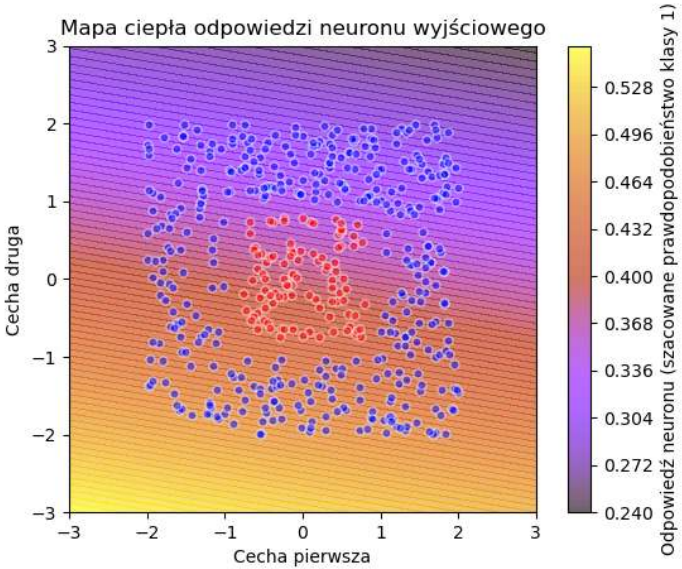
Jeden neuron wyjściowy, ⊛⊛ → ◎


Przykładowe dane
Cecha druga
Cecha pierwsza
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3

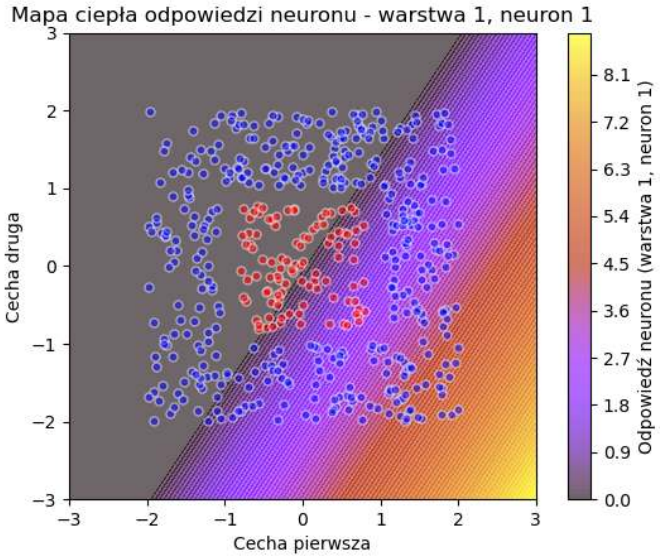

Mapa ciepła odpowiedzi neuronu wyjściowego
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (szacowane prawdopodobieństwo klasy 1)
3
2
1
0
−1
−2
−3
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0

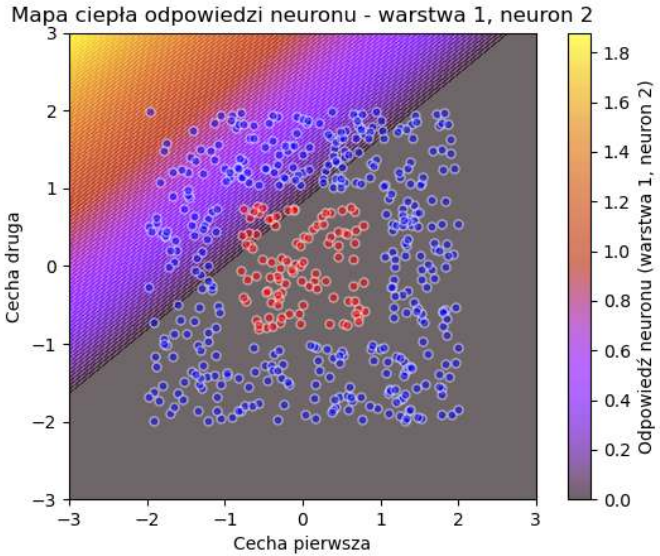

Przykładowe dane
Cecha druga
Cecha pierwsza
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3

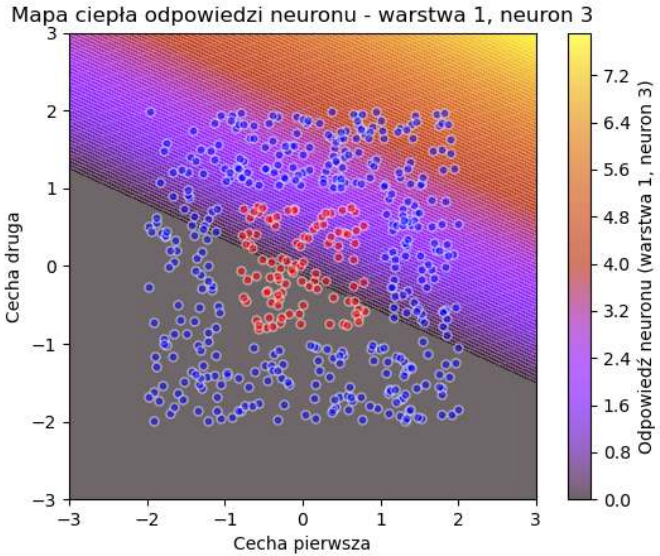

Mapa ciepła odpowiedzi neuronu wyjściowego
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (szacowane prawdopodobieństwo klasy 1)
3
2
1
0
−1
−2
−3
0.424
0.384
0.344
0.304
0.264
0.224
0.184
0.144
0.104
0.064

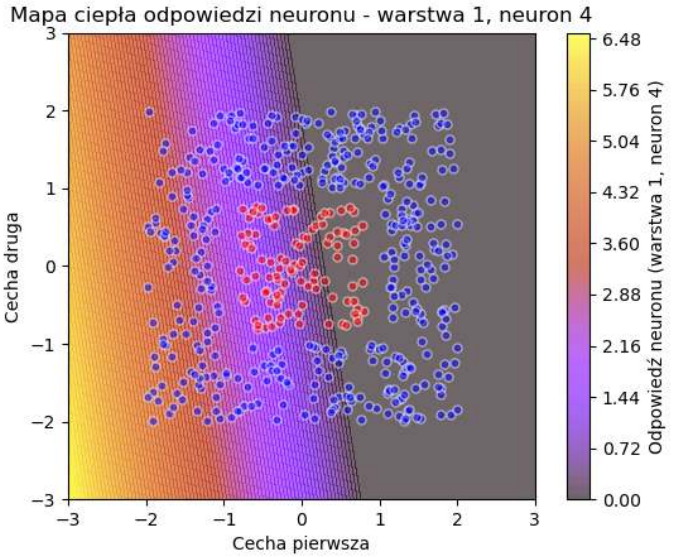

Mapa ciepła odpowiedzi neuronu wyjściowego
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (szacowane prawdopodobieństwo klasy 1)
3
2
1
0
−1
−2
−3
0.528
0.496
0.464
0.432
0.400
0.368
0.336
0.304
0.272
0.240

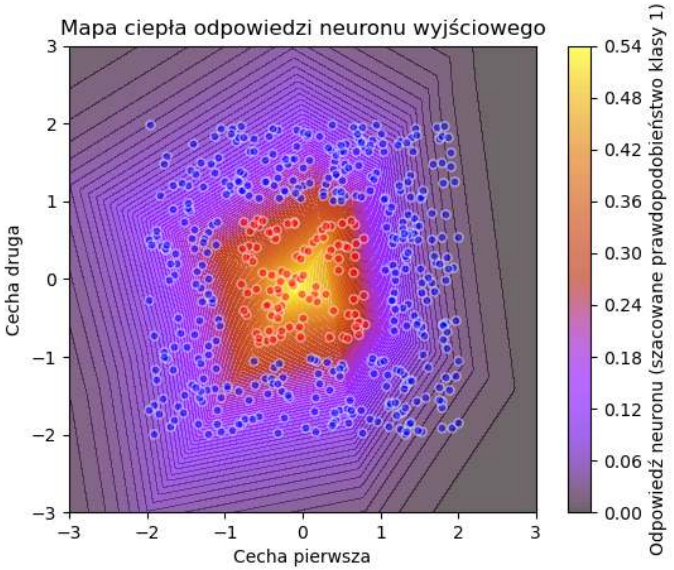
Jeden neuron wyjściowy, cztery w warstwie ukrytej ⊛⊛ → ⊖ ⊖ ⊖⊖ → ◎

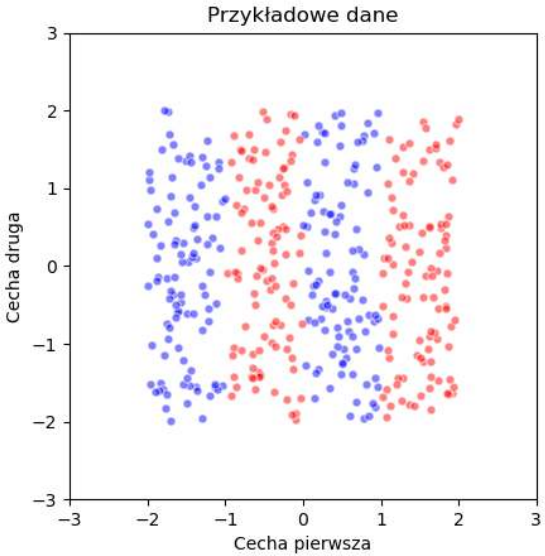

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 1
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 1)
3
2
1
0
−1
−2
−3
8.1
7.2
6.3
5.4
4.5
3.6
2.7
1.8
0.9
0.0

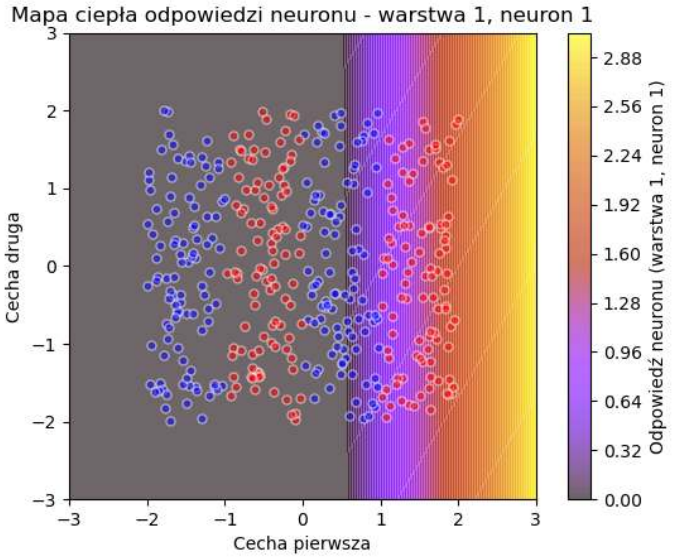

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 2
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 2)
3
2
1
0
−1
−2
−3
1.8
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0

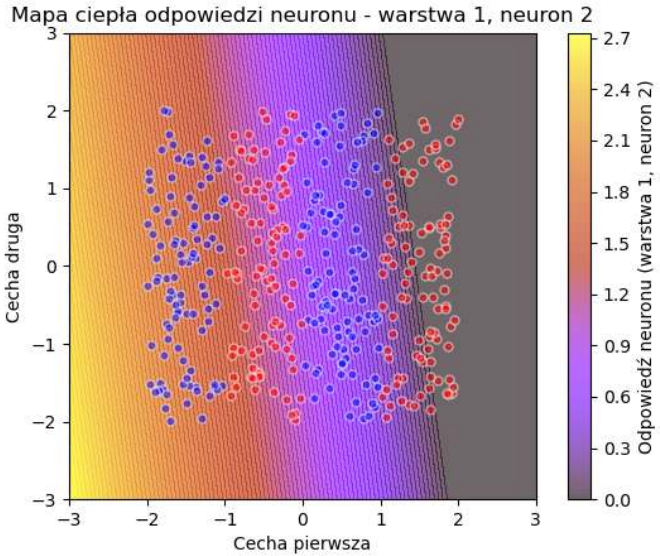

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 3
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 3)
3
2
1
0
−1
−2
−3
7.2
6.4
5.6
4.8
4.0
3.2
2.4
1.6
0.8
0.0

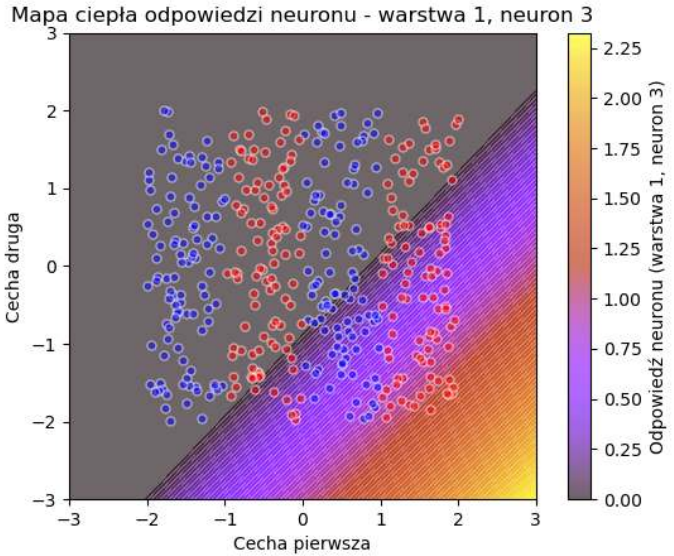

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 4
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 4)
3
2
1
0
−1
−2
−3
6.48
5.76
5.04
4.32
3.60
2.88
2.16
1.44
0.72
0.00

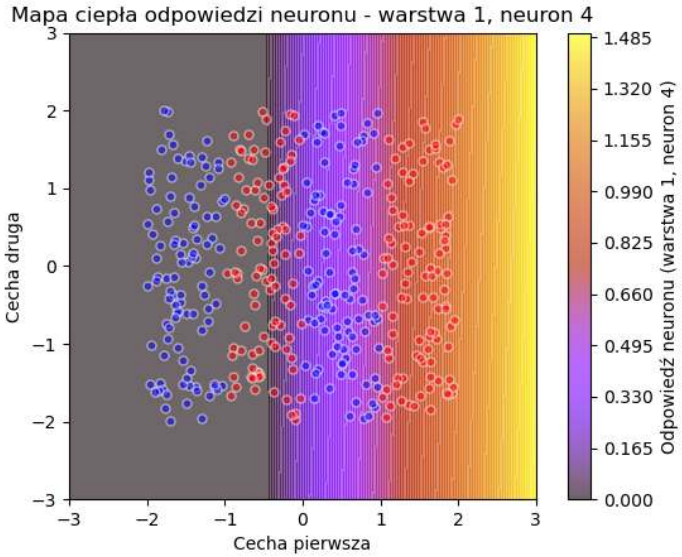

Mapa ciepła odpowiedzi neuronu wyjściowego
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (szacowane prawdopodobieństwo klasy 1)
3
2
1
0
−1
−2
−3
0.54
0.48
0.42
0.36
0.30
0.24
0.18
0.12
0.06
0.00

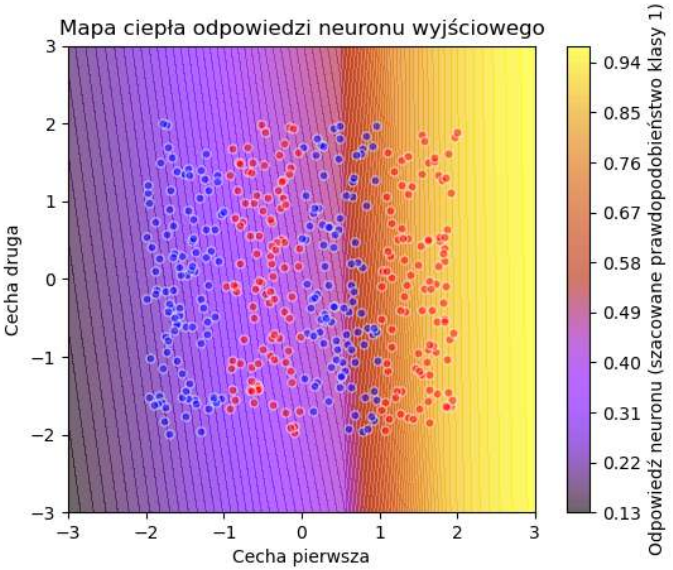

Przykładowe dane
Cecha druga
Cecha pierwsza
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3


Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 1
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 1)
3
2
1
0
−1
−2
−3
2.88
2.56
2.24
1.92
1.60
1.28
0.96
0.64
0.32
0.00

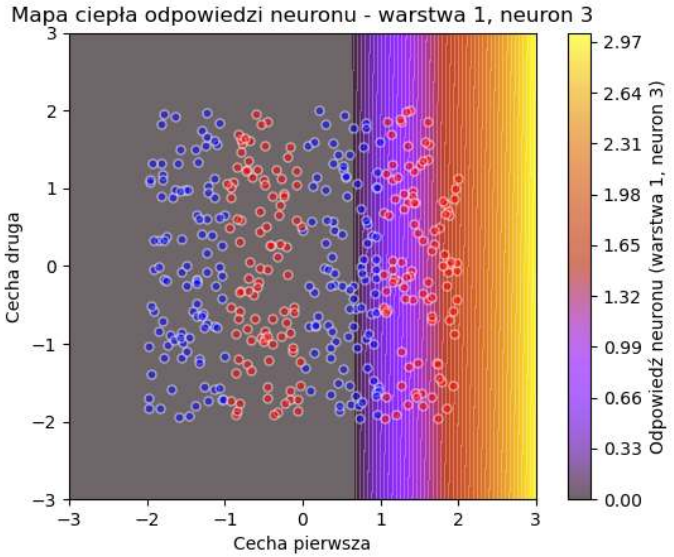

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 2
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 2)
3
2
1
0
−1
−2
−3
2.7
2.4
2.1
1.8
1.5
1.2
0.9
0.6
0.3
0.0

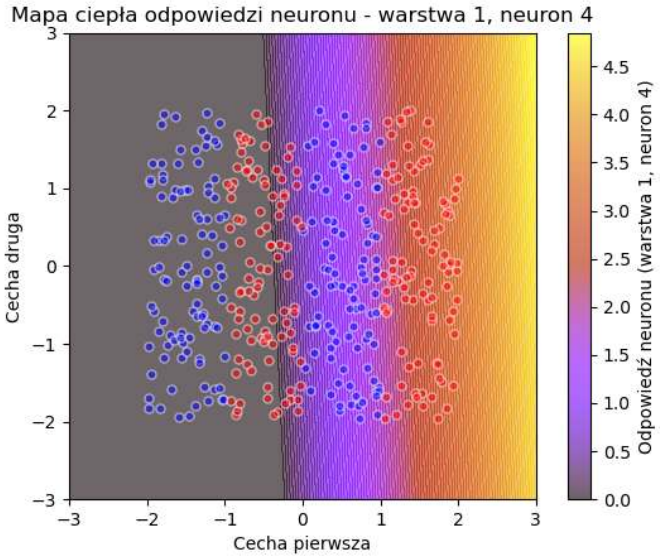

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 3
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 3)
3
2
1
0
−1
−2
−3
2.25
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00

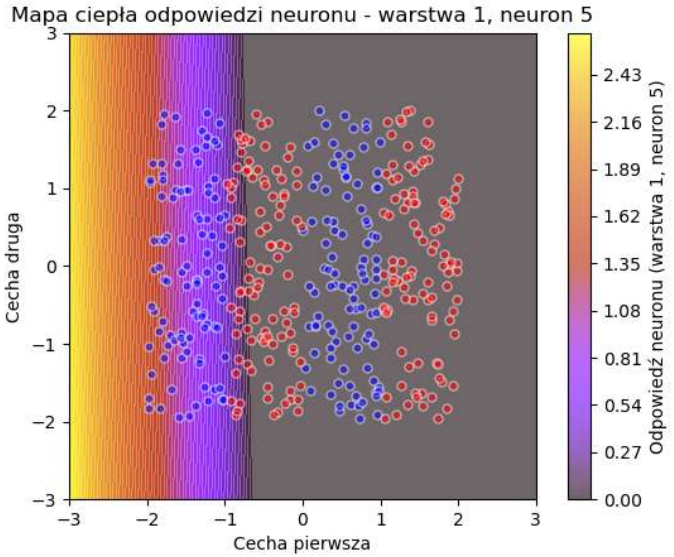

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 4
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 4)
3
2
1
0
−1
−2
−3
1.485
1.320
1.155
0.990
0.825
0.660
0.495
0.330
0.165
0.000

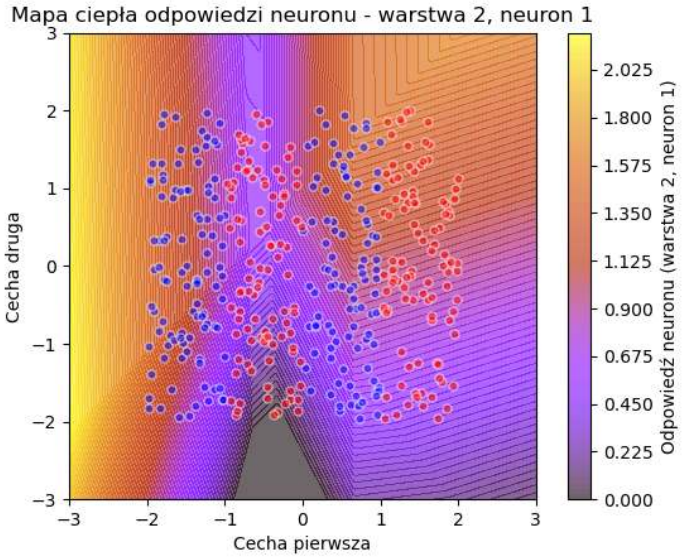

Mapa ciepła odpowiedzi neuronu wyjściowego
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (szacowane prawdopodobieństwo klasy 1)
3
2
1
0
−1
−2
−3
0.94
0.85
0.76
0.67
0.58
0.49
0.40
0.31
0.22
0.13

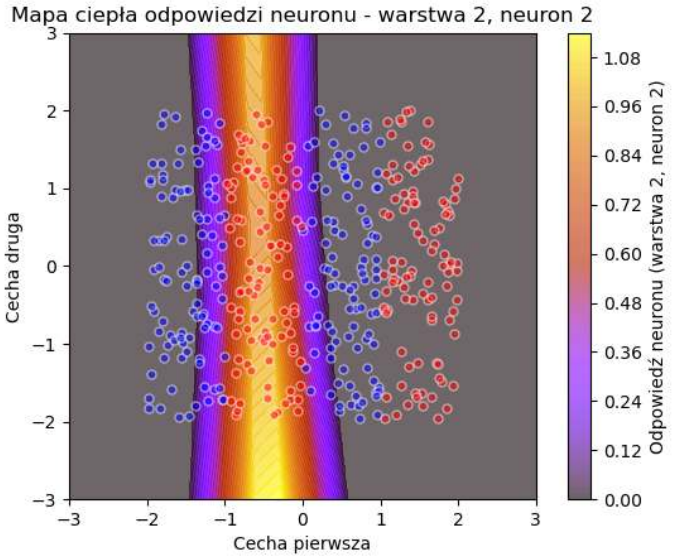
2 warstwy ukryte (5, 5), ⊛⊛ → ⊖ ⊖ ⊖ ⊖ ⊖ → ⊖ ⊖ ⊖ ⊖ ⊖ → ◎

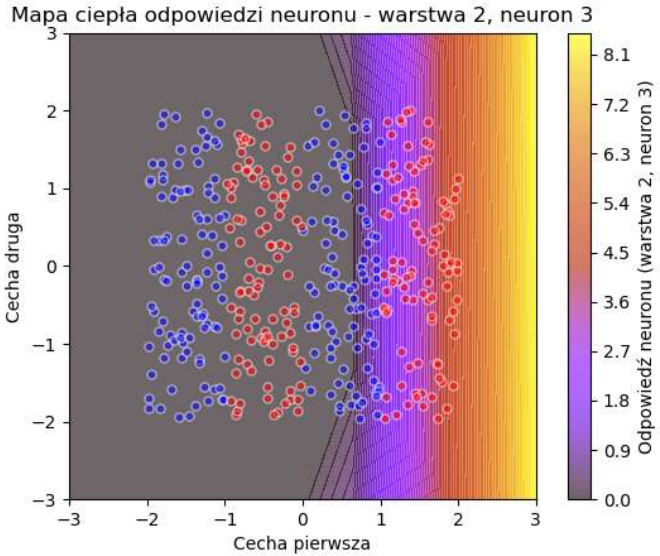

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 3
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 3)
3
2
1
0
−1
−2
−3
2.97
2.64
2.31
1.98
1.65
1.32
0.99
0.66
0.33
0.00

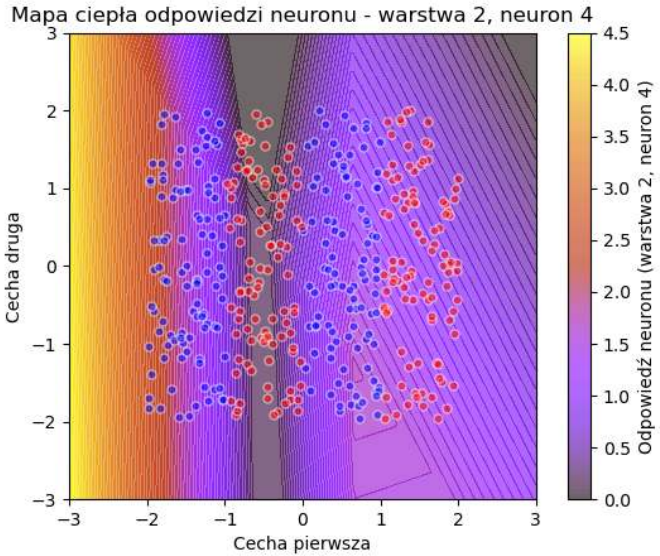

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 4
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 4)
3
2
1
0
−1
−2
−3
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0

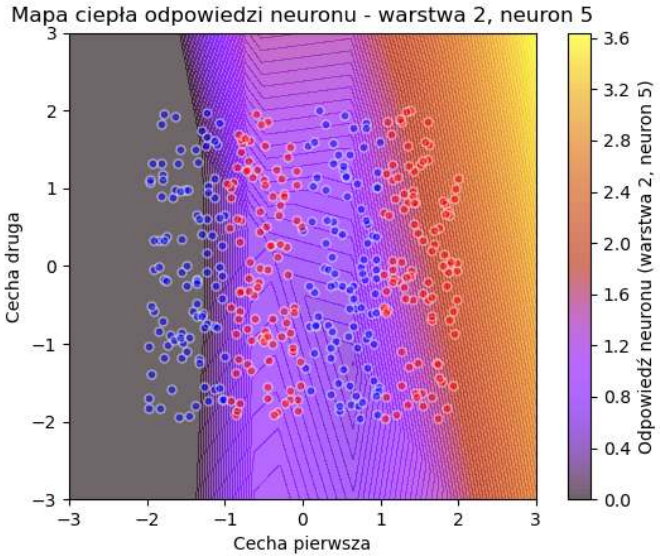

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 5
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 5)
3
2
1
0
−1
−2
−3
2.43
2.16
1.89
1.62
1.35
1.08
0.81
0.54
0.27
0.00

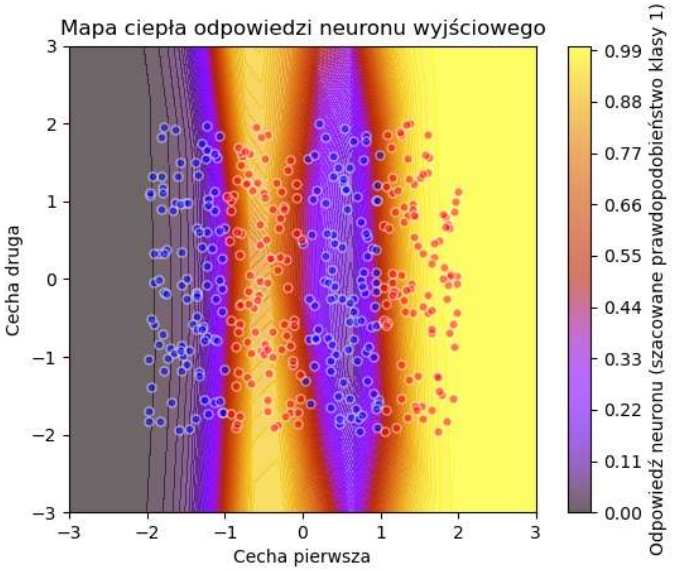

Mapa ciepła odpowiedzi neuronu - warstwa 2, neuron 1
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 2, neuron 1)
3
2
1
0
−1
−2
−3
2.025
1.800
1.575
1.350
1.125
0.900
0.675
0.450
0.225
0.000

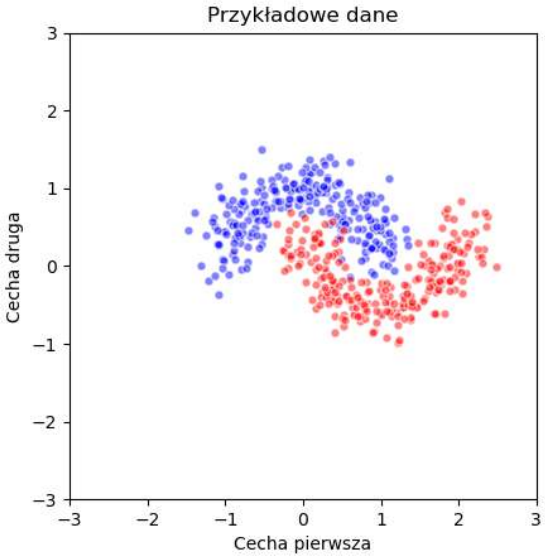

Mapa ciepła odpowiedzi neuronu - warstwa 2, neuron 2
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 2, neuron 2)
3
2
1
0
−1
−2
−3
1.08
0.96
0.84
0.72
0.60
0.48
0.36
0.24
0.12
0.00

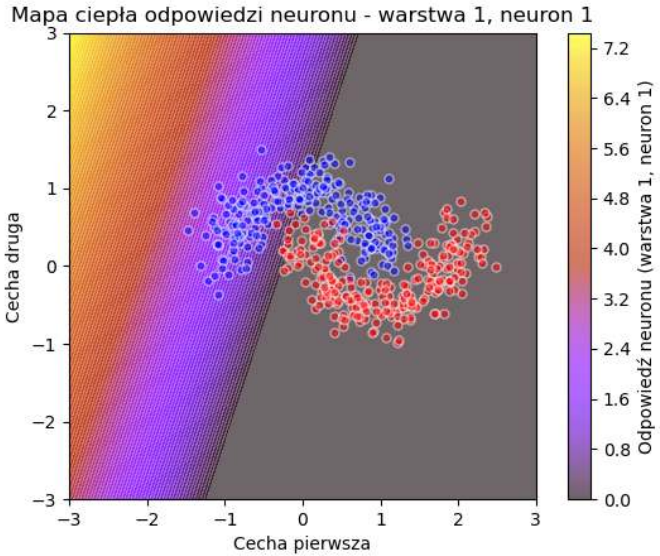

Mapa ciepła odpowiedzi neuronu - warstwa 2, neuron 3
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 2, neuron 3)
3
2
1
0
−1
−2
−3
8.1
7.2
6.3
5.4
4.5
3.6
2.7
1.8
0.9
0.0

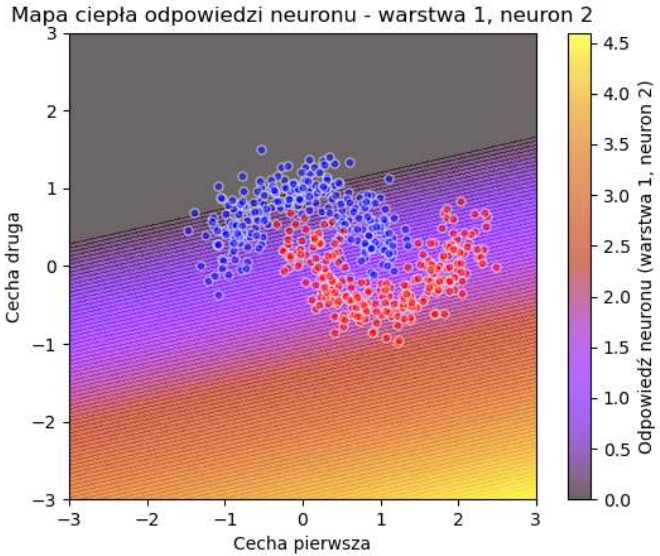

Mapa ciepła odpowiedzi neuronu - warstwa 2, neuron 4
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 2, neuron 4)
3
2
1
0
−1
−2
−3
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0

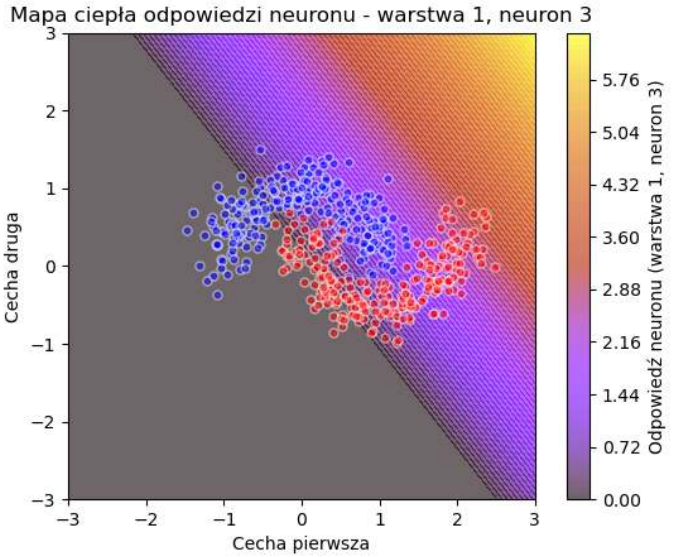

Mapa ciepła odpowiedzi neuronu - warstwa 2, neuron 5
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 2, neuron 5)
3
2
1
0
−1
−2
−3
3.6
3.2
2.8
2.4
2.0
1.6
1.2
0.8
0.4
0.0

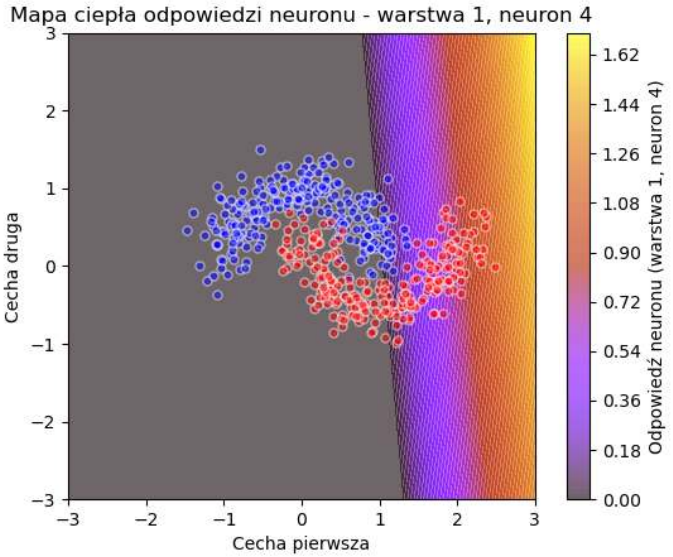

Mapa ciepła odpowiedzi neuronu wyjściowego
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (szacowane prawdopodobieństwo klasy 1)
3
2
1
0
−1
−2
−3
0.99
0.88
0.77
0.66
0.55
0.44
0.33
0.22
0.11
0.00

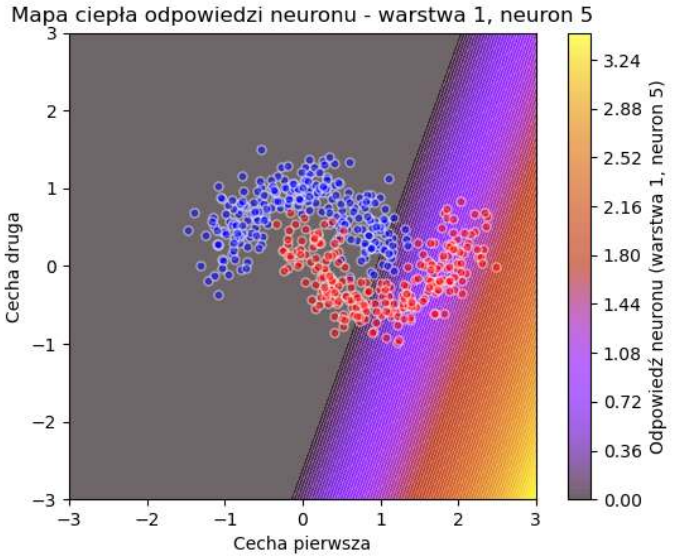

Przykładowe dane
Cecha druga
Cecha pierwsza
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3

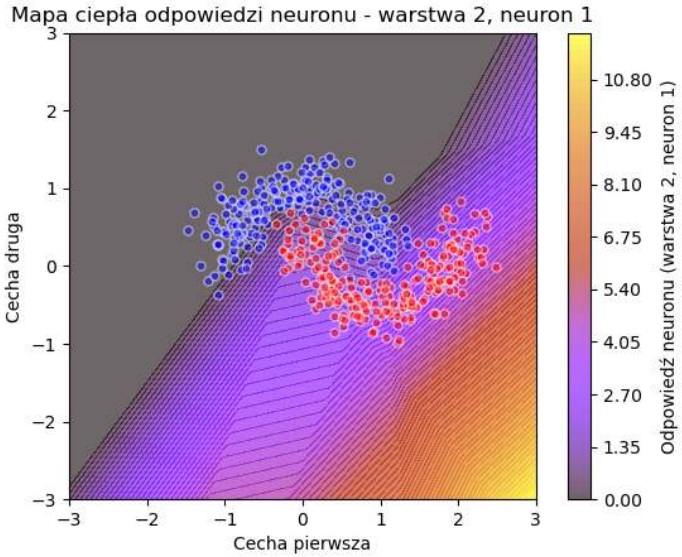

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 1
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 1)
3
2
1
0
−1
−2
−3
7.2
6.4
5.6
4.8
4.0
3.2
2.4
1.6
0.8
0.0

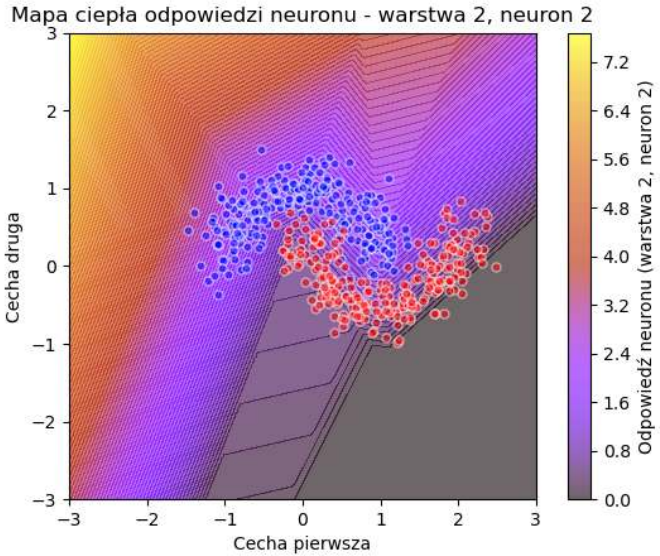

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 2
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 2)
3
2
1
0
−1
−2
−3
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0

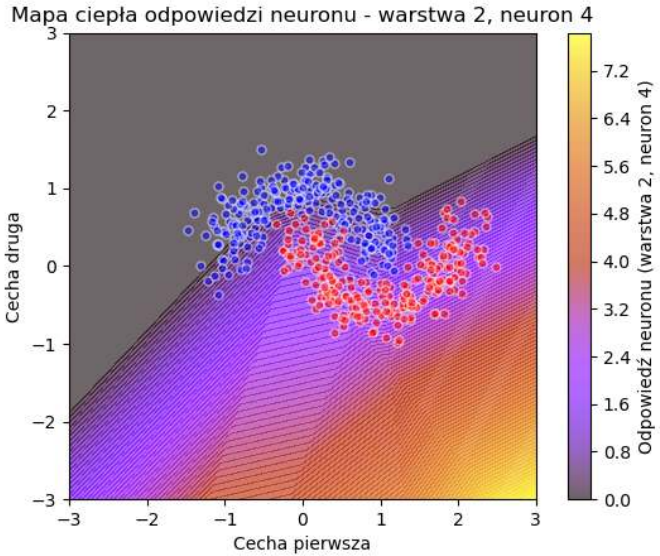

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 3
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 3)
3
2
1
0
−1
−2
−3
5.76
5.04
4.32
3.60
2.88
2.16
1.44
0.72
0.00

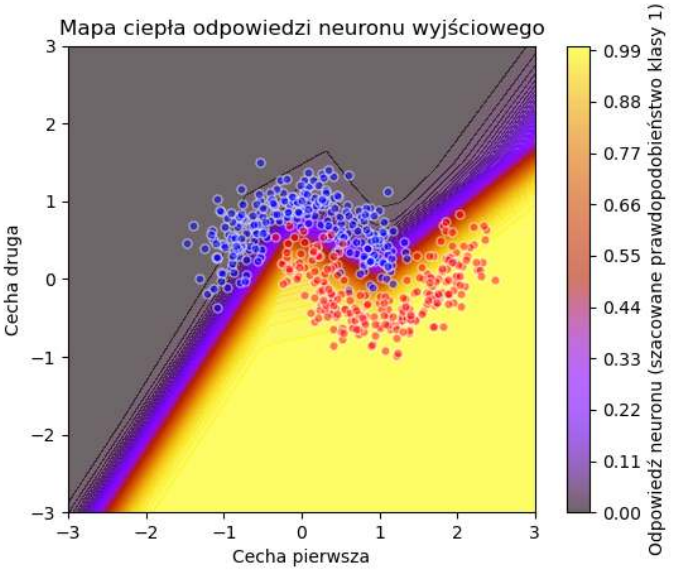

Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 4
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 4)
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3
1.62
1.44
1.26
1.08
0.90
0.72
0.54
0.36
0.18
0.00


Mapa ciepła odpowiedzi neuronu - warstwa 1, neuron 5
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 1, neuron 5)
3
2
1
0
−1
−2
−3
3.24
2.88
2.52
2.16
1.80
1.44
1.08
0.72
0.36
0.00

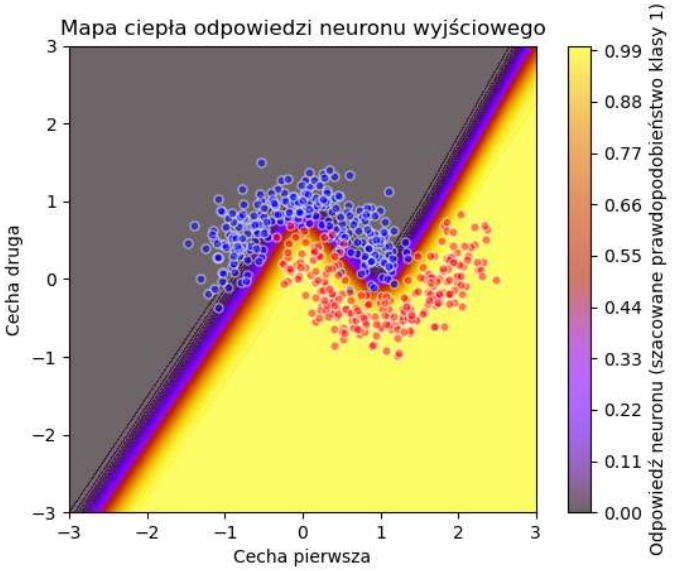

Mapa ciepła odpowiedzi neuronu - warstwa 2, neuron 1
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 2, neuron 1)
3
2
1
0
−1
−2
−3
10.80
9.45
8.10
6.75
5.40
4.05
2.70
1.35
0.00

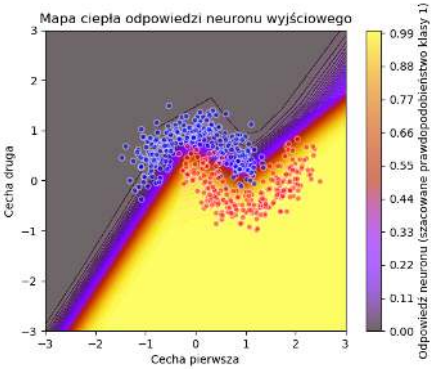

Mapa ciepła odpowiedzi neuronu - warstwa 2, neuron 2
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 2, neuron 2)
3
2
1
0
−1
−2
−3
7.2
6.4
5.6
4.8
4.0
3.2
2.4
1.6
0.8
0.0

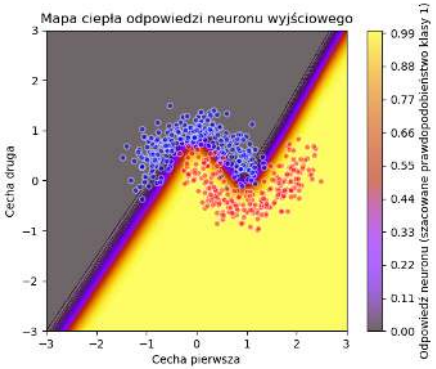

Mapa ciepła odpowiedzi neuronu - warstwa 2, neuron 4
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (warstwa 2, neuron 4)
3
2
1
0
−1
−2
−3
7.2
6.4
5.6
4.8
4.0
3.2
2.4
1.6
0.8
0.0


Mapa ciepła odpowiedzi neuronu wyjściowego
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (szacowane prawdopodobieństwo klasy 1)
3
2
1
0
−1
−2
−3
0.99
0.88
0.77
0.66
0.55
0.44
0.33
0.22
0.11
0.00

2 warstwy ukryte (10, 10), $\circledast\circledast \longrightarrow 10 \times \ominus \longrightarrow 10 \times \ominus \longrightarrow \circledcirc$


Mapa ciepła odpowiedzi neuronu wyjściowego
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (szacowane prawdopodobieństwo klasy 1)
3
2
1
0
−1
−2
−3
0.99
0.88
0.77
0.66
0.55
0.44
0.33
0.22
0.11
0.00


Mapa ciepła odpowiedzi neuronu wyjściowego
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (szacowane prawdopodobieństwo klasy 1)
0.99
0.88
0.77
0.66
0.55
0.44
0.33
0.22
0.11
0.00


Mapa ciepła odpowiedzi neuronu wyjściowego
Cecha druga
Cecha pierwsza
Odpowiedź neuronu (szacowane prawdopodobieństwo klasy 1)
0.99
0.88
0.77
0.66
0.55
0.44
0.33
0.22
0.11
0.00

Sieci MLP (Multi Layer Perceptron)

Przemysław Głomb

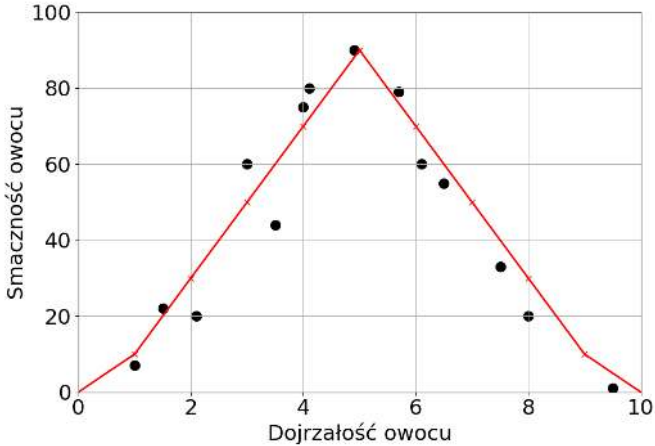
Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk

redaktor pomocniczy: Bartłomiej Gardas

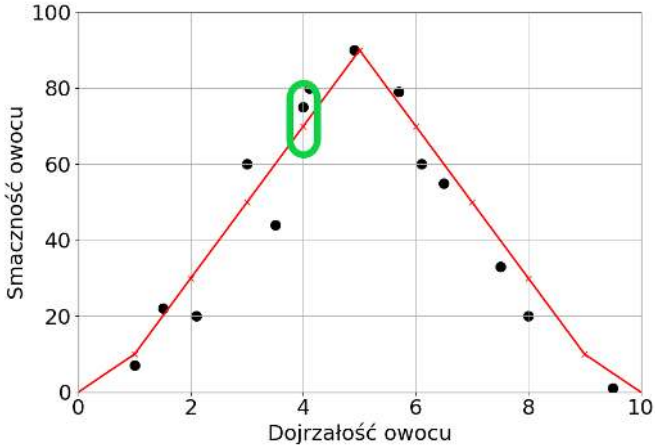
Uczenie / trening sieci

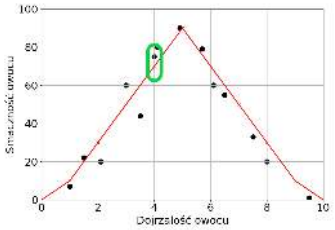
Backpropagation

$$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0)$$
$$y_{21} = \max(y_{11} - y_{12}, 0)$$



$$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0)$$
$$y_{21} = \max(y_{11} - y_{12}, 0)$$





$(x = 4, y^{\text{true}} = 75)$

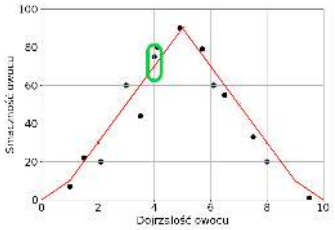
$y_{11} = \max(20 \times 4 - 10, 0) = \max(70, 0) = 70$ $\quad w_{111} = 20, b_{11} = -10$

$y_{12} = \max(40 \times 4 - 200, 0) = \max(-40, 0) = 0$ $\quad w_{121} = 40, b_{12} = -200$

$y_{21} = \max(70 - 0, 0) = 70$ $\quad w_{211} = 1, w_{212} = -1, b_{21} = 0$

błąd (loss) $(75 - 70)^2 = 25$ $\quad L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2$

błąd $\sim w_{ijk}$?



$(x = 4, y^{\text{true}} = 75)$

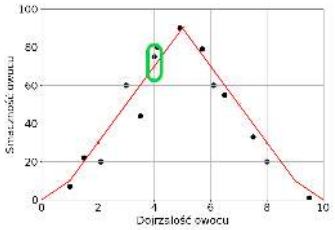
$y_{11} = \max(20 \times 4 - 10, 0) = \max(70, 0) = 70$ $\quad w_{111} = 20, b_{11} = -10$

$y_{12} = \max(40 \times 4 - 200, 0) = \max(-40, 0) = 0$ $\quad w_{121} = 40, b_{12} = -200$

$y_{21} = \max(70 - 0, 0) = 70$ $\quad w_{211} = 1, w_{212} = -1, b_{21} = 0$

błąd (loss) $(75 - 70)^2 = 25$ $\quad L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2$

błąd $\sim w_{ijk}$?



$(x = 4, y^{\text{true}} = 75)$

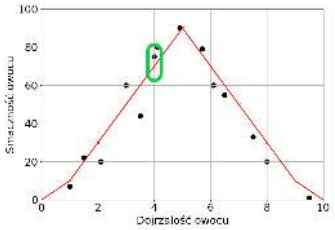
$y_{11} = \max(20 \times 4 - 10, 0) = \max(70, 0) = 70$ $\quad w_{111} = 20, b_{11} = -10$

$y_{12} = \max(40 \times 4 - 200, 0) = \max(-40, 0) = 0$ $\quad w_{121} = 40, b_{12} = -200$

$y_{21} = \max(70 - 0, 0) = 70$ $\quad w_{211} = 1, w_{212} = -1, b_{21} = 0$

błąd (loss) $(75 - 70)^2 = 25$ $\quad L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2$

błąd $\sim w_{ijk}$?



$(x = 4, y^{\text{true}} = 75)$

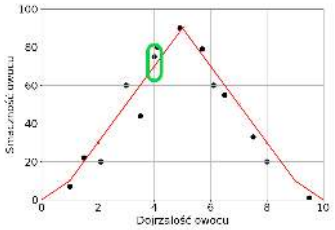
$y_{11} = \max(20 \times 4 - 10, 0) = \max(70, 0) = 70$ $\quad w_{111} = 20, b_{11} = -10$

$y_{12} = \max(40 \times 4 - 200, 0) = \max(-40, 0) = 0$ $\quad w_{121} = 40, b_{12} = -200$

$y_{21} = \max(70 - 0, 0) = 70$ $\quad w_{211} = 1, w_{212} = -1, b_{21} = 0$

błąd (loss) $(75 - 70)^2 = 25$ $\quad L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2$

błąd $\sim w_{ijk}$?



$(x = 4, y^{\text{true}} = 75)$

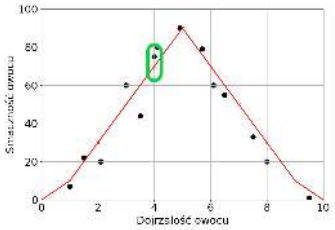
$y_{11} = \max(20 \times 4 - 10, 0) = \max(70, 0) = 70$ $\quad w_{111} = 20, b_{11} = -10$

$y_{12} = \max(40 \times 4 - 200, 0) = \max(-40, 0) = 0$ $\quad w_{121} = 40, b_{12} = -200$

$y_{21} = \max(70 - 0, 0) = 70$ $\quad w_{211} = 1, w_{212} = -1, b_{21} = 0$

błąd (loss) $(75 - 70)^2 = 25$ $\quad L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2$

błąd $\sim w_{ijk}$?



$(x = 4, y^{\text{true}} = 75)$

$y_{11} = \max(20 \times 4 - 10, 0) = \max(70, 0) = 70$ $\quad w_{111} = 20, b_{11} = -10$

$y_{12} = \max(40 \times 4 - 200, 0) = \max(-40, 0) = 0$ $\quad w_{121} = 40, b_{12} = -200$

$y_{21} = \max(70 - 0, 0) = 70$ $\quad w_{211} = 1, w_{212} = -1, b_{21} = 0$

błąd (loss) $(75 - 70)^2 = 25$ $\quad L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2$

błąd $\sim w_{ijk}$?

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$

$y_{11} = \max(z_{11}, 0) \quad z_{11} = w_{111}x + b_{11}$

$y_{12} = \max(z_{12}, 0) \quad z_{12} = w_{121}x + b_{12}$

$y_{21} = \max(z_{21}, 0) \quad z_{21} = w_{211}y_{11} + w_{212}y_{12} + b_{21}$

$\frac{\partial L}{\partial y_{21}} = \frac{\partial}{\partial y_{21}}(y^{\text{true}^2} - 2y^{\text{true}}y_{21} + y_{21}^2) = 2y_{21} - 2y^{\text{true}}$

$\frac{\partial L}{\partial z_{21}} = \frac{\partial L}{\partial y_{21}} \frac{\partial y_{21}}{\partial z_{21}} \quad \frac{\partial y_{21}}{\partial z_{21}} = \{0 \text{ if } z_{21} < 0; 1 \text{ if } z_{21} > 0\}$

(Reguła łańcuchowa $\frac{\partial L}{\partial w} = \frac{\partial L}{\partial y} \frac{\partial y}{\partial z} \frac{\partial z}{\partial w}$)

$\frac{\partial L}{\partial w_{211}} = \frac{\partial L}{\partial y_{21}} \frac{\partial y_{21}}{\partial z_{21}} \frac{\partial z_{21}}{\partial w_{211}} \quad \frac{\partial z_{21}}{\partial w_{211}} = y_{11}$

...

$\frac{\partial L}{\partial w_{ijk}}$ (zależność błędu od wagi)

$\Delta w_{ijk} = -\gamma \frac{\partial L}{\partial w_{ijk}}$ (aktualizacja wagi)

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$

$y_{11} = \max(z_{11}, 0) \quad z_{11} = w_{111}x + b_{11}$

$y_{12} = \max(z_{12}, 0) \quad z_{12} = w_{121}x + b_{12}$

$y_{21} = \max(z_{21}, 0) \quad z_{21} = w_{211}y_{11} + w_{212}y_{12} + b_{21}$

$\frac{\partial L}{\partial y_{21}} = \frac{\partial}{\partial y_{21}}(y^{\text{true}^2} - 2y^{\text{true}}y_{21} + y_{21}^2) = 2y_{21} - 2y^{\text{true}}$

$\frac{\partial L}{\partial z_{21}} = \frac{\partial L}{\partial y_{21}} \frac{\partial y_{21}}{\partial z_{21}} \quad \frac{\partial y_{21}}{\partial z_{21}} = \{0 \text{ if } z_{21} < 0; 1 \text{ if } z_{21} > 0\}$

(Reguła łańcuchowa $\frac{\partial L}{\partial w} = \frac{\partial L}{\partial y} \frac{\partial y}{\partial z} \frac{\partial z}{\partial w}$)

$\frac{\partial L}{\partial w_{211}} = \frac{\partial L}{\partial y_{21}} \frac{\partial y_{21}}{\partial z_{21}} \frac{\partial z_{21}}{\partial w_{211}} \quad \frac{\partial z_{21}}{\partial w_{211}} = y_{11}$

...

$\frac{\partial L}{\partial w_{ijk}}$ (zależność błędu od wagi)

$\Delta w_{ijk} = -\gamma \frac{\partial L}{\partial w_{ijk}}$ (aktualizacja wagi)

$$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$$
$$y_{11} = \max(z_{11}, 0) \quad z_{11} = w_{111}x + b_{11}$$
$$y_{12} = \max(z_{12}, 0) \quad z_{12} = w_{121}x + b_{12}$$
$$y_{21} = \max(z_{21}, 0) \quad z_{21} = w_{211}y_{11} + w_{212}y_{12} + b_{21}$$

$$\frac{\partial L}{\partial y_{21}} = \frac{\partial}{\partial y_{21}}(y^{\text{true}^2} - 2y^{\text{true}}y_{21} + y_{21}^2) = 2y_{21} - 2y^{\text{true}}$$

$$\frac{\partial L}{\partial z_{21}} = \frac{\partial L}{\partial y_{21}}\frac{\partial y_{21}}{\partial z_{21}} \quad \frac{\partial y_{21}}{\partial z_{21}} = \{0 \text{ if } z_{21} < 0; 1 \text{ if } z_{21} > 0\}$$

(Reguła łańcuchowa $\frac{\partial L}{\partial w} = \frac{\partial L}{\partial y}\frac{\partial y}{\partial z}\frac{\partial z}{\partial w}$)

$$\frac{\partial L}{\partial w_{211}} = \frac{\partial L}{\partial y_{21}}\frac{\partial y_{21}}{\partial z_{21}}\frac{\partial z_{21}}{\partial w_{211}} \quad \frac{\partial z_{21}}{\partial w_{211}} = y_{11}$$

...

$\frac{\partial L}{\partial w_{ijk}}$ (zależność błędu od wagi)

$\Delta w_{ijk} = -\gamma\frac{\partial L}{\partial w_{ijk}}$ (aktualizacja wagi)

$$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$$
$$y_{11} = \max(z_{11}, 0) \quad z_{11} = w_{111}x + b_{11}$$
$$y_{12} = \max(z_{12}, 0) \quad z_{12} = w_{121}x + b_{12}$$
$$y_{21} = \max(z_{21}, 0) \quad z_{21} = w_{211}y_{11} + w_{212}y_{12} + b_{21}$$

$$\frac{\partial L}{\partial y_{21}} = \frac{\partial}{\partial y_{21}}(y^{\text{true}^2} - 2y^{\text{true}}y_{21} + y_{21}^2) = 2y_{21} - 2y^{\text{true}}$$

$$\frac{\partial L}{\partial z_{21}} = \frac{\partial L}{\partial y_{21}}\frac{\partial y_{21}}{\partial z_{21}} \quad \frac{\partial y_{21}}{\partial z_{21}} = \{0 \text{ if } z_{21} < 0; 1 \text{ if } z_{21} > 0\}$$

(Reguła łańcuchowa $\frac{\partial L}{\partial w} = \frac{\partial L}{\partial y}\frac{\partial y}{\partial z}\frac{\partial z}{\partial w}$)

$$\frac{\partial L}{\partial w_{211}} = \frac{\partial L}{\partial y_{21}}\frac{\partial y_{21}}{\partial z_{21}}\frac{\partial z_{21}}{\partial w_{211}} \quad \frac{\partial z_{21}}{\partial w_{211}} = y_{11}$$

...

$\frac{\partial L}{\partial w_{ijk}}$ (zależność błędu od wagi)

$\Delta w_{ijk} = -\gamma\frac{\partial L}{\partial w_{ijk}}$ (aktualizacja wagi)

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$

$y_{11} = \max(z_{11}, 0) \quad z_{11} = w_{111}x + b_{11}$

$y_{12} = \max(z_{12}, 0) \quad z_{12} = w_{121}x + b_{12}$

$y_{21} = \max(z_{21}, 0) \quad z_{21} = w_{211}y_{11} + w_{212}y_{12} + b_{21}$

$\frac{\partial L}{\partial y_{21}} = \frac{\partial}{\partial y_{21}}(y^{\text{true}^2} - 2y^{\text{true}}y_{21} + y_{21}^2) = 2y_{21} - 2y^{\text{true}}$

$\frac{\partial L}{\partial z_{21}} = \frac{\partial L}{\partial y_{21}} \frac{\partial y_{21}}{\partial z_{21}} \quad \frac{\partial y_{21}}{\partial z_{21}} = \{0 \text{ if } z_{21} < 0; 1 \text{ if } z_{21} > 0\}$

(Reguła łańcuchowa $\frac{\partial L}{\partial w} = \frac{\partial L}{\partial y} \frac{\partial y}{\partial z} \frac{\partial z}{\partial w}$)

$\frac{\partial L}{\partial w_{211}} = \frac{\partial L}{\partial y_{21}} \frac{\partial y_{21}}{\partial z_{21}} \frac{\partial z_{21}}{\partial w_{211}} \quad \frac{\partial z_{21}}{\partial w_{211}} = y_{11}$

...

$\frac{\partial L}{\partial w_{ijk}}$ (zależność błędu od wagi)

$\Delta w_{ijk} = -\gamma \frac{\partial L}{\partial w_{ijk}}$ (aktualizacja wagi)

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$
$y_{11} = \max(z_{11}, 0) \quad z_{11} = w_{111}x + b_{11}$
$y_{12} = \max(z_{12}, 0) \quad z_{12} = w_{121}x + b_{12}$
$y_{21} = \max(z_{21}, 0) \quad z_{21} = w_{211}y_{11} + w_{212}y_{12} + b_{21}$

$$\frac{\partial L}{\partial y_{21}} = \frac{\partial}{\partial y_{21}}(y^{\text{true}^2} - 2y^{\text{true}}y_{21} + y_{21}^2) = 2y_{21} - 2y^{\text{true}}$$

$$\frac{\partial L}{\partial z_{21}} = \frac{\partial L}{\partial y_{21}}\frac{\partial y_{21}}{\partial z_{21}} \quad \frac{\partial y_{21}}{\partial z_{21}} = \{0 \text{ if } z_{21} < 0; 1 \text{ if } z_{21} > 0\}$$

(Reguła łańcuchowa $\frac{\partial L}{\partial w} = \frac{\partial L}{\partial y}\frac{\partial y}{\partial z}\frac{\partial z}{\partial w}$)

$$\frac{\partial L}{\partial w_{211}} = \frac{\partial L}{\partial y_{21}}\frac{\partial y_{21}}{\partial z_{21}}\frac{\partial z_{21}}{\partial w_{211}} \quad \frac{\partial z_{21}}{\partial w_{211}} = y_{11}$$

...

$\frac{\partial L}{\partial w_{ijk}}$ (zależność błędu od wagi)

$\Delta w_{ijk} = -\gamma\frac{\partial L}{\partial w_{ijk}}$ (aktualizacja wagi)

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$
$y_{11} = \max(z_{11}, 0) \quad z_{11} = w_{111}x + b_{11}$
$y_{12} = \max(z_{12}, 0) \quad z_{12} = w_{121}x + b_{12}$
$y_{21} = \max(z_{21}, 0) \quad z_{21} = w_{211}y_{11} + w_{212}y_{12} + b_{21}$

$\frac{\partial L}{\partial y_{21}} = \frac{\partial}{\partial y_{21}}(y^{\text{true}^2} - 2y^{\text{true}}y_{21} + y_{21}^2) = 2y_{21} - 2y^{\text{true}}$

$\frac{\partial L}{\partial z_{21}} = \frac{\partial L}{\partial y_{21}} \frac{\partial y_{21}}{\partial z_{21}} \quad \frac{\partial y_{21}}{\partial z_{21}} = \{0 \text{ if } z_{21} < 0; 1 \text{ if } z_{21} > 0\}$

(Reguła łańcuchowa $\frac{\partial L}{\partial w} = \frac{\partial L}{\partial y} \frac{\partial y}{\partial z} \frac{\partial z}{\partial w}$)

$\frac{\partial L}{\partial w_{211}} = \frac{\partial L}{\partial y_{21}} \frac{\partial y_{21}}{\partial z_{21}} \frac{\partial z_{21}}{\partial w_{211}} \quad \frac{\partial z_{21}}{\partial w_{211}} = y_{11}$

...

$\frac{\partial L}{\partial w_{ijk}}$ (zależność błędu od wagi)

$\Delta w_{ijk} = -\gamma \frac{\partial L}{\partial w_{ijk}}$ (aktualizacja wagi)

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$
$y_{11} = \max(z_{11}, 0) \quad z_{11} = w_{111}x + b_{11}$
$y_{12} = \max(z_{12}, 0) \quad z_{12} = w_{121}x + b_{12}$
$y_{21} = \max(z_{21}, 0) \quad z_{21} = w_{211}y_{11} + w_{212}y_{12} + b_{21}$

$\frac{\partial L}{\partial y_{21}} = \frac{\partial}{\partial y_{21}}(y^{\text{true}^2} - 2y^{\text{true}}y_{21} + y_{21}^2) = 2y_{21} - 2y^{\text{true}}$

$\frac{\partial L}{\partial z_{21}} = \frac{\partial L}{\partial y_{21}} \frac{\partial y_{21}}{\partial z_{21}} \quad \frac{\partial y_{21}}{\partial z_{21}} = \{0 \text{ if } z_{21} < 0; 1 \text{ if } z_{21} > 0\}$

(Reguła łańcuchowa $\frac{\partial L}{\partial w} = \frac{\partial L}{\partial y} \frac{\partial y}{\partial z} \frac{\partial z}{\partial w}$)

$\frac{\partial L}{\partial w_{211}} = \frac{\partial L}{\partial y_{21}} \frac{\partial y_{21}}{\partial z_{21}} \frac{\partial z_{21}}{\partial w_{211}} \quad \frac{\partial z_{21}}{\partial w_{211}} = y_{11}$
...
$\frac{\partial L}{\partial w_{ijk}}$ (zależność błędu od wagi)

$\Delta w_{ijk} = -\gamma \frac{\partial L}{\partial w_{ijk}}$ (aktualizacja wagi)

1. Sieć neuronowa: wagi (+bias, architektura)
2. Przykład (sample) i wartość wymagana dla niego (ground truth)
3. Wyliczenie wartości częściowych i końcowych dla przykładu (forward pass)
4. Miara błędu (loss)
5. Wsteczna propagacja błędu (backpropagation) – wyznaczenie gradientów
6. Aktualizacja wag sieci

[Stochastic] gradient descent

Mamy:

1. Adaptacja (aktualizacja) wag dla konkretnego przykładu (punktu danych)

Dalsze zagadnienia:

1. Wiele punktów danych – optymalizacja
2. Model danych i skuteczna predykcja z modelu
3. Współczynnik szybkości uczenia (learning rate), zanikające i eksplodujące gradienty, zagadnienia optymalizacji, inicjalizacja, ...

Mamy:

1. Adaptacja (aktualizacja) wag dla konkretnego przykładu (punktu danych)

Dalsze zagadnienia:

1. Wiele punktów danych – optymalizacja
2. Model danych i skuteczna predykcja z modelu
3. Współczynnik szybkości uczenia (learning rate), zanikające i eksplodujące gradienty, zagadnienia optymalizacji, inicjalizacja, ...

„Gradient prosty" (gradient descent):

1. Mamy $\frac{\partial L}{\partial w_{ijk}}$
2. Minimalizujemy L – błąd zmniejsza się najszybciej jeżeli idziemy w kierunku przeciwnym do gradientu
3. Iteracyjnie modyfikujemy wagi $w_{ijk}^{n+1} = w_{ijk}^{n} - \gamma \frac{\partial L}{\partial w_{ijk}}$
4. Zejdziemy wzdłuż gradientu do zera i problem rozwiązany? Niestety, bardziej skomplikowane...
5. Na początek – jak wyliczyć gradient – mamy $m = 12$ przykładów

(wykorzystujemy wszystkie) batch gradient descent

(wykorzystujemy jeden) stochastic gradient descent

(wykorzystujemy podzbiór) minibatch gradient descent

„Gradient prosty” (gradient descent):

1. Mamy $\frac{\partial L}{\partial w_{ijk}}$
2. Minimalizujemy L – błąd zmniejsza się najszybciej jeżeli idziemy w kierunku przeciwnym do gradientu
3. Iteracyjnie modyfikujemy wagi $w_{ijk}^{n+1} = w_{ijk}^{n} - \gamma \frac{\partial L}{\partial w_{ijk}}$
4. Zejdziemy wzdłuż gradientu do zera i problem rozwiązany? Niestety, bardziej skomplikowane...
5. Na początek – jak wyliczyć gradient – mamy $m = 12$ przykładów

●●●●●●●●●●●●	(wykorzystujemy wszystkie)	batch gradient descent
○●○○○○○○○○○○	(wykorzystujemy jeden)	stochastic gradient descent
○○○○●●●●○○○○	(wykorzystujemy podzbiór)	minibatch gradient descent

1. Gradient descent – iteracyjne poprawianie sieci
   - 1.1 **batch gradient descent** – aktualizacja modelu dopiero po przejściu przez wszystkie przykłady treningowe – kosztowne obliczeniowo i pamięciowo
   - 1.2 **stochastic gradient descent** – aktualizacja modelu po analizie pojedynczego przykładu – mniejsze wymagania obliczeniowe i pamięciowe, mogą pojawić się fluktuacje błędu
   - 1.3 **minibatch gradient descent** – praktyczny kompromis
2. Przejście przez wszystkie przykłady zbioru treningowego – epoka (**epoch**)
3. Przygody na szlaku z gradientem
   - 3.1 Lokalne minima (**local minima**)
   - 3.2 Punkt siodłowy (**saddle point**)
   - 3.3 Zanikający gradient (**vanishing gradient**)
   - 3.4 Eksplodujący gradient (**exploding gradient**)

1. Gradient descent – iteracyjne poprawianie sieci
   - 1.1 **batch gradient descent** – aktualizacja modelu dopiero po przejściu przez wszystkie przykłady treningowe – kosztowne obliczeniowo i pamięciowo
   - 1.2 **stochastic gradient descent** – aktualizacja modelu po analizie pojedynczego przykładu – mniejsze wymagania obliczeniowe i pamięciowe, mogą pojawić się fluktuacje błędu
   - 1.3 **minibatch gradient descent** – praktyczny kompromis
2. Przejście przez wszystkie przykłady zbioru treningowego – epoka (**epoch**)
3. Przygody na szlaku z gradientem
   - 3.1 Lokalne minima (**local minima**)
   - 3.2 Punkt siodłowy (**saddle point**)
   - 3.3 Zanikający gradient (**vanishing gradient**)
   - 3.4 Eksplodujący gradient (**exploding gradient**)

1. Gradient descent – iteracyjne poprawianie sieci
   - 1.1 **batch gradient descent** – aktualizacja modelu dopiero po przejściu przez wszystkie przykłady treningowe – kosztowne obliczeniowo i pamięciowo
   - 1.2 **stochastic gradient descent** – aktualizacja modelu po analizie pojedynczego przykładu – mniejsze wymagania obliczeniowe i pamięciowe, mogą pojawić się fluktuacje błędu
   - 1.3 **minibatch gradient descent** – praktyczny kompromis
2. Przejście przez wszystkie przykłady zbioru treningowego – epoka (**epoch**)
3. Przygody na szlaku z gradientem
   - 3.1 Lokalne minima (**local minima**)
   - 3.2 Punkt siodłowy (**saddle point**)
   - 3.3 Zanikający gradient (**vanishing gradient**)
   - 3.4 Eksplodujący gradient (**exploding gradient**)

1. Gradient descent – iteracyjne poprawianie sieci
   - 1.1 **batch gradient descent** – aktualizacja modelu dopiero po przejściu przez wszystkie przykłady treningowe – kosztowne obliczeniowo i pamięciowo
   - 1.2 **stochastic gradient descent** – aktualizacja modelu po analizie pojedynczego przykładu – mniejsze wymagania obliczeniowe i pamięciowe, mogą pojawić się fluktuacje błędu
   - 1.3 **minibatch gradient descent** – praktyczny kompromis
2. Przejście przez wszystkie przykłady zbioru treningowego – epoka (**epoch**)
3. Przygody na szlaku z gradientem
   - 3.1 Lokalne minima (**local minima**)
   - 3.2 Punkt siodłowy (**saddle point**)
   - 3.3 Zanikający gradient (**vanishing gradient**)
   - 3.4 Eksplodujący gradient (**exploding gradient**)

1. Gradient descent – iteracyjne poprawianie sieci
   - 1.1 **batch gradient descent** – aktualizacja modelu dopiero po przejściu przez wszystkie przykłady treningowe – kosztowne obliczeniowo i pamięciowo
   - 1.2 **stochastic gradient descent** – aktualizacja modelu po analizie pojedynczego przykładu – mniejsze wymagania obliczeniowe i pamięciowe, mogą pojawić się fluktuacje błędu
   - 1.3 **minibatch gradient descent** – praktyczny kompromis
2. Przejście przez wszystkie przykłady zbioru treningowego – epoka (**epoch**)
3. Przygody na szlaku z gradientem
   - 3.1 Lokalne minima (**local minima**)
   - 3.2 Punkt siodłowy (**saddle point**)
   - 3.3 Zanikający gradient (**vanishing gradient**)
   - 3.4 Eksplodujący gradient (**exploding gradient**)

1. Gradient descent – iteracyjne poprawianie sieci
   - 1.1 **batch gradient descent** – aktualizacja modelu dopiero po przejściu przez wszystkie przykłady treningowe – kosztowne obliczeniowo i pamięciowo
   - 1.2 **stochastic gradient descent** – aktualizacja modelu po analizie pojedynczego przykładu – mniejsze wymagania obliczeniowe i pamięciowe, mogą pojawić się fluktuacje błędu
   - 1.3 **minibatch gradient descent** – praktyczny kompromis
2. Przejście przez wszystkie przykłady zbioru treningowego – epoka (**epoch**)
3. Przygody na szlaku z gradientem
   - 3.1 Lokalne minima (**local minima**)
   - 3.2 Punkt siodłowy (**saddle point**)
   - 3.3 Zanikający gradient (**vanishing gradient**)
   - 3.4 Eksplodujący gradient (**exploding gradient**)

1. Gradient descent – iteracyjne poprawianie sieci
   - 1.1 **batch gradient descent** – aktualizacja modelu dopiero po przejściu przez wszystkie przykłady treningowe – kosztowne obliczeniowo i pamięciowo
   - 1.2 **stochastic gradient descent** – aktualizacja modelu po analizie pojedynczego przykładu – mniejsze wymagania obliczeniowe i pamięciowe, mogą pojawić się fluktuacje błędu
   - 1.3 **minibatch gradient descent** – praktyczny kompromis
2. Przejście przez wszystkie przykłady zbioru treningowego – epoka (**epoch**)
3. Przygody na szlaku z gradientem
   - 3.1 Lokalne minima (**local minima**)
   - 3.2 Punkt siodłowy (**saddle point**)
   - 3.3 Zanikający gradient (**vanishing gradient**)
   - 3.4 Eksplodujący gradient (**exploding gradient**)

Rysunek: Gradient descent w typowych ilustracjach

Rysunek: Gradient descent w prakt

źródło ilustracji:DALL·E

Rysunek: Gradient descent w typowych ilustracjach

Rysunek: Gradient descent w praktyce

źródło ilustracji:DALL·E

Learning rate

Mamy:

1. Adaptacja (aktualizacja) wag dla konkretnego przykładu (punktu danych)
2. Optymalizacja – iteracyjne przejście przez kolejne punkty danych i modyfikacje wag

Dalsze zagadnienia:

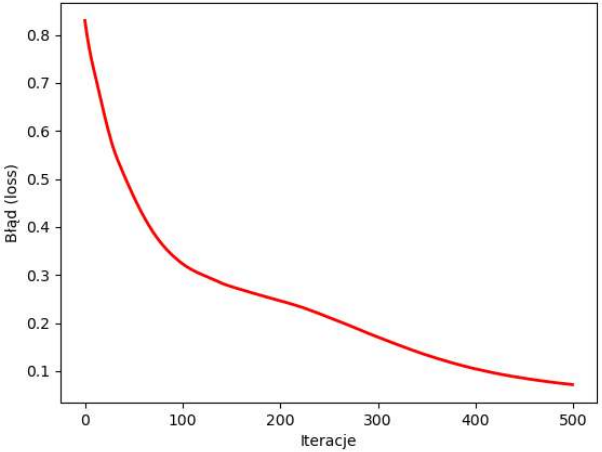
1. Krzywa błędu (**loss curve**)
2. Learning rate
3. Optymalizacja++

Mamy:

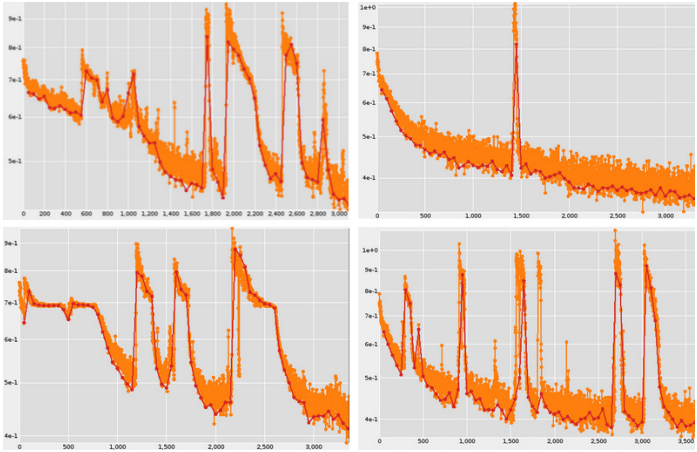
1. Adaptacja (aktualizacja) wag dla konkretnego przykładu (punktu danych)
2. Optymalizacja – iteracyjne przejście przez kolejne punkty danych i modyfikacje wag

Dalsze zagadnienia:

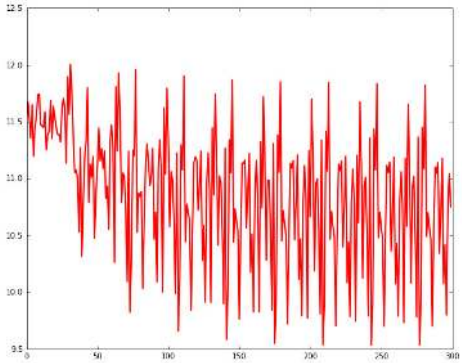
1. Krzywa błędu (**loss curve**)
2. Learning rate
3. Optymalizacja++



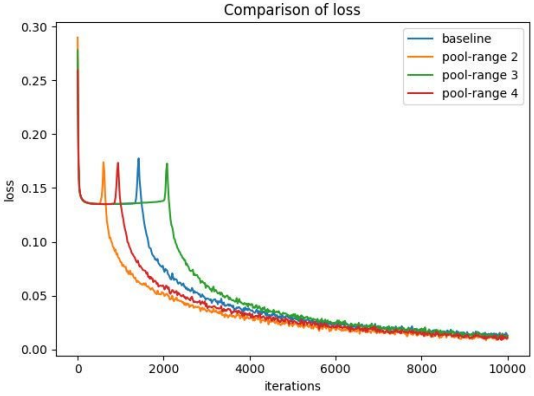
Rysunek: Przykładowa krzywa błędu



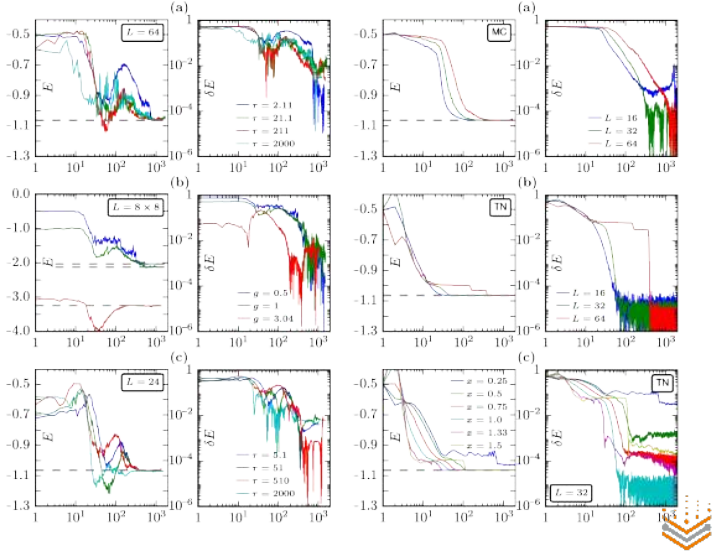
Rysunek: *Taming Spatial Transformer Networks, contributed by Diogo. For the record, it's not supposed to look like that.* (Pobrane z `https://lossfunctions.tumblr.com/`)



Rysunek: *A heart rate or a loss function? :)*
This one of a custom implementation of an RNN, graciously contributed by Ray Zhang. (Pobrane z `https://lossfunctions.tumblr.com/`)



Rysunek: *A highly amusing specimen from @_karfly . Truly baffles the mind.* (Pobrane z `https://lossfunctions.tumblr.com/`)



B. Gardas et al, Quantum neural networks to simulate many-body quantum systems, Phys. Rev. B 98,

redback.bjfu.ws: Why is my loss function going down and jumping?
J_Johnson: It's not always a straight path to the best fit. In fact, usually it's pretty bumpy.

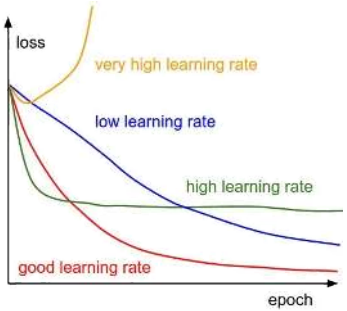
(z: `https://discuss.pytorch.org/t/why-is-my-loss-function-going-down-and-jumping/170508`)

Kontrola procesu treningu:

1. Współczynnik szybkości uczenia (**learning rate**) γ, $w_{ijk}^{n+1} = w_{ijk}^{n} - \gamma \frac{\partial L}{\partial w_{ijk}}$

Learning rate:

- Typowe wartości $0.1 - 0.001$
- Zbyt niskie wartości (zwykle) – wydłużony czas uczenia; wysokie – zatrzymanie uczenia, oscylacje
- Algorytmy: np. Adagrad, RMSprop, Adam...



Źródło ilustracji: CS231n Convolutional Neural Networks for Visual Recognition

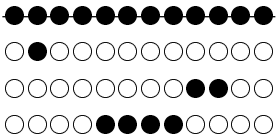
- Algorytmy: np. Adagrad, RMSprop, Adam...
  - Pęd gradientu (**momentum**) – przejście przez punkty siodłowe (gdzie gradient ma niewielkie wartości) i lokalne minima
    $v^n = \mu v^{n-1} - \gamma \frac{\partial L}{\partial w_{ijk}} \qquad w_{ijk}^{n+1} = w_{ijk}^n + v^n$
  - Skalowanie (**learning rate annealing/decay**), np. $\gamma_n = \gamma_0 e^{-dn}$ – zmniejszanie wartości w trakcie uczenia

redback.bjfu.ws: Why is my loss function going down and jumping?
J_Johnson: It's not always a straight path to the best fit. In fact, usually it's pretty bumpy.
(z: `https://discuss.pytorch.org/t/why-is-my-loss-function-going-down-and-jumping/170508`)

Kontrola procesu treningu:

1. Współczynnik szybkości uczenia (**learning rate**) γ, $w_{ijk}^{n+1} = w_{ijk}^{n} - \gamma \frac{\partial L}{\partial w_{ijk}}$
2. Rozmiar podzbioru danych (**batch size**)

Batch size (BS):

 (niepraktyczne)

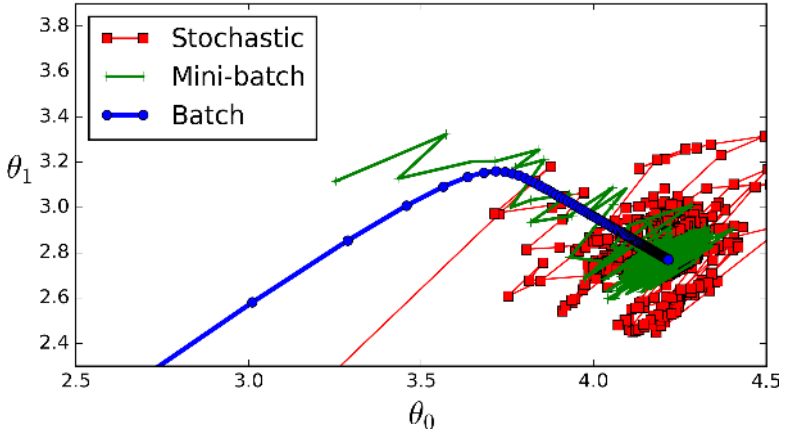
popularne wartości np. 32, 64 / 128, 256

małe wartości np. 1, 2, 4, 8 – szybsze zmiany w sieci (więcej na epokę)

duże wartości wymagają więcej pamięci, mniej szumu

Przy małej wartości BS mogą pojawić się fluktuacje (widoczne w krzywej błędu), niekoniecznie negatywne – mogą uodparniać przed przetrenowaniem

Stochastic vs minibatch vs batch gradient descent



Źródło ilustracji: Odpowiedź *itdxer* w pytaniu *What is batch size in neural network?* CrossValidated StackExchange

Wartości gradientu, inicjalizacja – wartości początkowe

$$y_{11} = \max(20x - 10, 0)$$

$$y_{21} = \max(20y_{11} - 10, 0) \quad y_{31} = \max(20y_{21} - 10, 0) \quad y_{41} = \max(20y_{31} - 10, 0)$$

$$x = 1 \rightarrow y_{11} = 10 \quad y_{21} = 190 \quad y_{31} = 3790 \quad y_{41} = 75790$$

$$y_{11} = \max(0.2x, 0) \quad y_{21} = \max(0.2y_{11}, 0) \quad \ldots$$

$$x = 1 \rightarrow y_{11} = 0.2 \quad y_{21} = 0.04 \quad y_{31} = 0.008 \quad y_{41} = 0.0016$$

- Wartości „przepływają" przez sieć, podczas wyliczania wyjść neuronów ($\rightarrow$) i gradientów ($\leftarrow$)
- Wartości wyliczanych wyjść/gradientów wrażliwe na zakres wartości wag
- Eksplodujący gradient (**exploding gradients**) – duże modyfikacje do wag, niestabilności/NaN
- Zanikający gradient (**vanishing gradient**) – małe (lub brak) modyfikacji do wag, spowolniony/zatrzymany proces uczenia

$y_{11} = \max(20x - 10, 0)$

$y_{21} = \max(20y_{11} - 10, 0) \quad y_{31} = \max(20y_{21} - 10, 0) \quad y_{41} = \max(20y_{31} - 10, 0)$

$x = 1 \rightarrow y_{11} = 10 \quad y_{21} = 190 \quad y_{31} = 3790 \quad y_{41} = 75790$

$y_{11} = \max(0.2x, 0) \quad y_{21} = \max(0.2y_{11}, 0) \quad \ldots$

$x = 1 \rightarrow y_{11} = 0.2 \quad y_{21} = 0.04 \quad y_{31} = 0.008 \quad y_{41} = 0.0016$

- Wartości „przepływają" przez sieć, podczas wyliczania wyjść neuronów ($\rightarrow$) i gradientów ($\leftarrow$)
- Wartości wyliczanych wyjść/gradientów wrażliwe na zakres wartości wag
- Eksplodujący gradient (**exploding gradients**) – duże modyfikacje do wag, niestabilności/NaN
- Zanikający gradient (**vanishing gradient**) – małe (lub brak) modyfikacji do wag, spowolniony/zatrzymany proces uczenia

$y_{11} = \max(20x - 10, 0)$
$y_{21} = \max(20y_{11} - 10, 0) \quad y_{31} = \max(20y_{21} - 10, 0) \quad y_{41} = \max(20y_{31} - 10, 0)$
$x = 1 \rightarrow y_{11} = 10 \quad y_{21} = 190 \quad y_{31} = 3790 \quad y_{41} = 75790$

$y_{11} = \max(0.2x, 0) \quad y_{21} = \max(0.2y_{11}, 0) \quad \ldots$
$x = 1 \rightarrow y_{11} = 0.2 \quad y_{21} = 0.04 \quad y_{31} = 0.008 \quad y_{41} = 0.0016$

- Wartości „przepływają" przez sieć, podczas wyliczania wyjść neuronów ($\rightarrow$) i gradientów ($\leftarrow$)
- Wartości wyliczanych wyjść/gradientów wrażliwe na zakres wartości wag
- Eksplodujący gradient (**exploding gradients**) – duże modyfikacje do wag, niestabilności/NaN
- Zanikający gradient (**vanishing gradient**) – małe (lub brak) modyfikacji do wag, spowolniony/zatrzymany proces uczenia

$$y_{11} = \max(20x - 10, 0)$$
$$y_{21} = \max(20y_{11} - 10, 0) \quad y_{31} = \max(20y_{21} - 10, 0) \quad y_{41} = \max(20y_{31} - 10, 0)$$
$$x = 1 \rightarrow y_{11} = 10 \quad y_{21} = 190 \quad y_{31} = 3790 \quad y_{41} = 75790$$

$$y_{11} = \max(0.2x, 0) \quad y_{21} = \max(0.2y_{11}, 0) \quad \ldots$$
$$x = 1 \rightarrow y_{11} = 0.2 \quad y_{21} = 0.04 \quad y_{31} = 0.008 \quad y_{41} = 0.0016$$

- Wartości „przepływają" przez sieć, podczas wyliczania wyjść neuronów ($\rightarrow$) i gradientów ($\leftarrow$)
- Wartości wyliczanych wyjść/gradientów wrażliwe na zakres wartości wag
- Eksplodujący gradient (**exploding gradients**) – duże modyfikacje do wag, niestabilności/NaN
- Zanikający gradient (**vanishing gradient**) – małe (lub brak) modyfikacji do wag, spowolniony/zatrzymany proces uczenia

$$y_{11} = \max(20x - 10, 0)$$
$$y_{21} = \max(20y_{11} - 10, 0) \quad y_{31} = \max(20y_{21} - 10, 0) \quad y_{41} = \max(20y_{31} - 10, 0)$$
$$x = 1 \rightarrow y_{11} = 10 \quad y_{21} = 190 \quad y_{31} = 3790 \quad y_{41} = 75790$$

$$y_{11} = \max(0.2x, 0) \quad y_{21} = \max(0.2y_{11}, 0) \quad \ldots$$
$$x = 1 \rightarrow y_{11} = 0.2 \quad y_{21} = 0.04 \quad y_{31} = 0.008 \quad y_{41} = 0.0016$$

- Wartości „przepływają" przez sieć, podczas wyliczania wyjść neuronów ($\rightarrow$) i gradientów ($\leftarrow$)
- Wartości wyliczanych wyjść/gradientów wrażliwe na zakres wartości wag
- Eksplodujący gradient (**exploding gradients**) – duże modyfikacje do wag, niestabilności/NaN
- Zanikający gradient (**vanishing gradient**) – małe (lub brak) modyfikacji do wag, spowolniony/zatrzymany proces uczenia

$$y_{11} = \max(20x - 10, 0)$$
$$y_{21} = \max(20y_{11} - 10, 0) \quad y_{31} = \max(20y_{21} - 10, 0) \quad y_{41} = \max(20y_{31} - 10, 0)$$
$$x = 1 \rightarrow y_{11} = 10 \quad y_{21} = 190 \quad y_{31} = 3790 \quad y_{41} = 75790$$

$$y_{11} = \max(0.2x, 0) \quad y_{21} = \max(0.2y_{11}, 0) \quad \ldots$$
$$x = 1 \rightarrow y_{11} = 0.2 \quad y_{21} = 0.04 \quad y_{31} = 0.008 \quad y_{41} = 0.0016$$

- Wartości „przepływają" przez sieć, podczas wyliczania wyjść neuronów ($\rightarrow$) i gradientów ($\leftarrow$)
- Wartości wyliczanych wyjść/gradientów wrażliwe na zakres wartości wag
- Eksplodujący gradient (**exploding gradients**) – duże modyfikacje do wag, niestabilności/NaN
- Zanikający gradient (**vanishing gradient**) – małe (lub brak) modyfikacji do wag, spowolniony/zatrzymany proces uczenia

1. Przygotowanie danych

   Normalizacja kontrola zakresu danych, np. $\hat{x}_{ij} = \frac{x_{ij} - \min \mathbf{x}_i}{\max \mathbf{x}_i - \min \mathbf{x}_i}$

   Standaryzacja korekta względem średniej i wariancji $\hat{x}_{ij} = \frac{x_{ij} - \mu_i}{\sigma_i}$

2. Przygotowanie sieci

   Inicjalizacja początkowe wartości wag losowane np. $w_{ijk} = \mathcal{N}(\mu = 0, \sigma = 0.01)$

   Xavier Glorot uwzględnia średnią liczbę połączeń do neuronu

   $$w_{ijk} = \mathcal{N}\left(\mu = 0, \sigma = \sqrt{\frac{1}{\text{fan}_{\text{avg}}}}\right) \text{ lub } w_{ijk} = \mathcal{U}\left(\sqrt{\frac{3}{\text{fan}_{\text{avg}}}}\right)$$

   Kaming He rozwija inicjalizację XG pod kątem ReLU $w_{ijk} = \mathcal{N}\left(\mu = 0, \sigma = \sqrt{\frac{2}{\text{fan}_{\text{in}}}}\right)$

   lub $w_{ijk} = \mathcal{U}\left(\sqrt{\frac{6}{\text{fan}_{\text{avg}}}}\right)$

3. Losowa inicjalizacja + optymalizacja lokalna = możliwe różne wyniki treningu w zależności od stanu początkowego

1. Przygotowanie danych

   Normalizacja kontrola zakresu danych, np. $\hat{x}_{ij} = \frac{x_{ij} - \min \mathbf{x}_i}{\max \mathbf{x}_i - \min \mathbf{x}_i}$

   Standaryzacja korekta względem średniej i wariancji $\hat{x}_{ij} = \frac{x_{ij} - \mu_i}{\sigma_i}$

2. Przygotowanie sieci

   Inicjalizacja początkowe wartości wag losowane np. $w_{ijk} = \mathcal{N}(\mu = 0, \sigma = 0.01)$

   Xavier Glorot uwzględnia średnią liczbę połączeń do neuronu

   $$w_{ijk} = \mathcal{N}\left(\mu = 0, \sigma = \sqrt{\frac{1}{\text{fan}_{\text{avg}}}}\right) \text{ lub } w_{ijk} = \mathcal{U}\left(\sqrt{\frac{3}{\text{fan}_{\text{avg}}}}\right)$$

   Kaming He rozwija inicjalizację XG pod kątem ReLU $w_{ijk} = \mathcal{N}\left(\mu = 0, \sigma = \sqrt{\frac{2}{\text{fan}_{\text{in}}}}\right)$

   lub $w_{ijk} = \mathcal{U}\left(\sqrt{\frac{6}{\text{fan}_{\text{avg}}}}\right)$

3. Losowa inicjalizacja + optymalizacja lokalna = możliwe różne wyniki treningu w zależności od stanu początkowego

1. Przygotowanie danych

   Normalizacja kontrola zakresu danych, np. $\hat{x}_{ij} = \frac{x_{ij} - \min \mathbf{x}_i}{\max \mathbf{x}_i - \min \mathbf{x}_i}$

   Standaryzacja korekta względem średniej i wariancji $\hat{x}_{ij} = \frac{x_{ij} - \mu_i}{\sigma_i}$

2. Przygotowanie sieci

   Inicjalizacja początkowe wartości wag losowane np. $w_{ijk} = \mathcal{N}(\mu = 0, \sigma = 0.01)$

   Xavier Glorot uwzględnia średnią liczbę połączeń do neuronu

   $$w_{ijk} = \mathcal{N}\left(\mu = 0, \sigma = \sqrt{\frac{1}{\text{fan}_{\text{avg}}}}\right) \text{ lub } w_{ijk} = \mathcal{U}\left(\sqrt{\frac{3}{\text{fan}_{\text{avg}}}}\right)$$

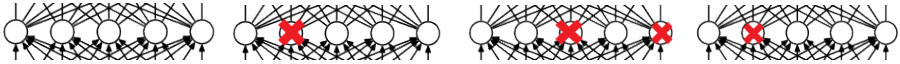
   Kaming He rozwija inicjalizację XG pod kątem ReLU $w_{ijk} = \mathcal{N}\left(\mu = 0, \sigma = \sqrt{\frac{2}{\text{fan}_{\text{in}}}}\right)$ lub $w_{ijk} = \mathcal{U}\left(\sqrt{\frac{6}{\text{fan}_{\text{avg}}}}\right)$

3. Losowa inicjalizacja + optymalizacja lokalna = możliwe różne wyniki treningu w zależności od stanu początkowego

Regularyzacja

- Wsparcie treningu – kontrola wartości wag

  Dropout z określonym prawdopodobieństwem (np. $p = 0.2$ albo $p = 0.5$) zerujemy wyjście z neuronu w trakcie treningu



  Batch Normalization normalizacja i przeskalowanie wartości pomiędzy warstwami

  $y = \gamma \frac{x - \mu_B}{\sigma_B} + \beta$ (standaryzacja + przeskalowanie + przesunięcie)

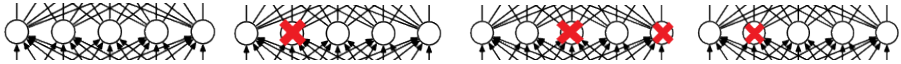
  L2/L1 regularization „kara" dla wysokich wartości wag

  $L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2 + \lambda \sum_{i=1}^{n} w_i^2$

  $L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2 + \lambda \sum_{i=1}^{n} |w_i|$

- Wsparcie treningu – kontrola wartości wag

  Dropout z określonym prawdopodobieństwem (np. $p = 0.2$ albo $p = 0.5$) zerujemy wyjście z neuronu w trakcie treningu



  Batch Normalization normalizacja i przeskalowanie wartości pomiędzy warstwami

  $y = \gamma \frac{x - \mu_B}{\sigma_B} + \beta$ (standaryzacja + przeskalowanie + przesunięcie)

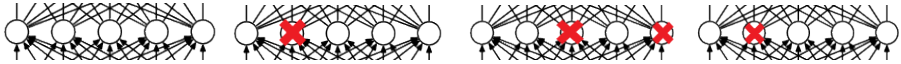
  L2/L1 regularization „kara" dla wysokich wartości wag

  $L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2 + \lambda \sum_{i=1}^{n} w_i^2$

  $L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2 + \lambda \sum_{i=1}^{n} |w_i|$

- Wsparcie treningu – kontrola wartości wag

  Dropout z określonym prawdopodobieństwem (np. $p = 0.2$ albo $p = 0.5$) zerujemy wyjście z neuronu w trakcie treningu



  Batch Normalization normalizacja i przeskalowanie wartości pomiędzy warstwami

  $y = \gamma \frac{x - \mu_B}{\sigma_B} + \beta$ (standaryzacja + przeskalowanie + przesunięcie)

  L2/L1 regularization „kara" dla wysokich wartości wag

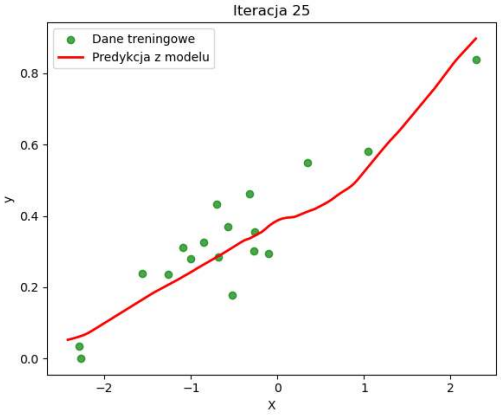
  $L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2 + \lambda \sum_{i=1}^{n} w_i^2$

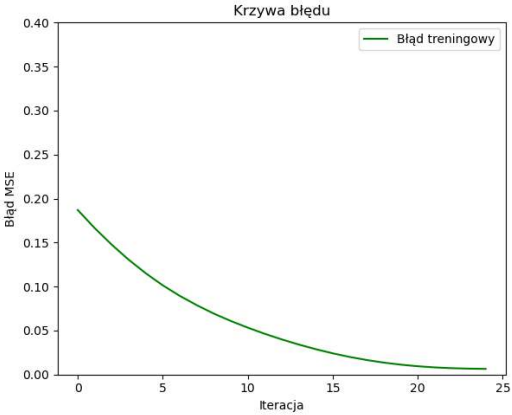
  $L(y^{\text{true}}, y^{\text{out}}) = (y^{\text{true}} - y_{21})^2 + \lambda \sum_{i=1}^{n} |w_i|$

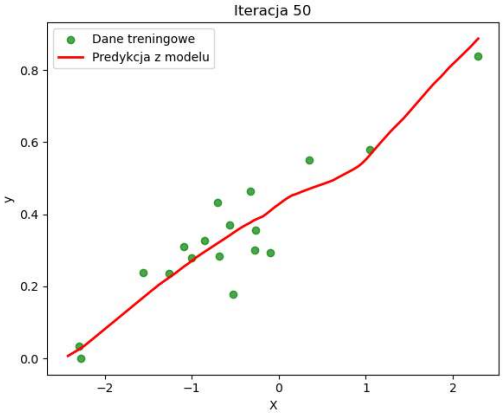
Mamy:

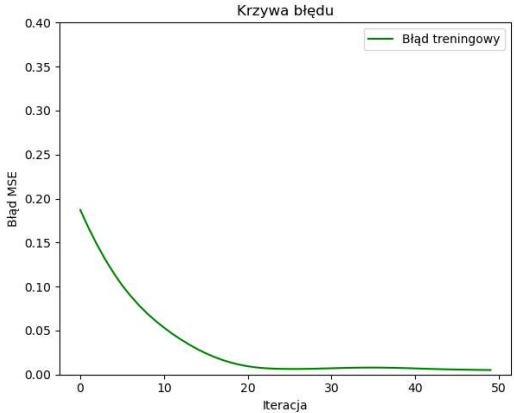
1. Adaptacja (aktualizacja) wag dla konkretnego przykładu (punktu danych)
2. Optymalizacja – iteracyjne przejście przez kolejne punkty danych i modyfikacje wag
3. Sterowanie procesem uczenia – learning rate, batch size, wybór optymalizatora
4. Inicjalizacja
5. Regularyzacja

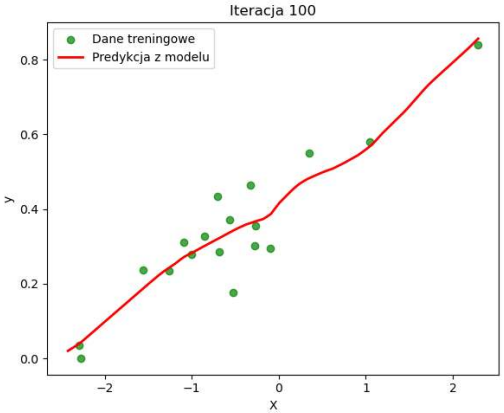
Zbiory testowe, walidacyjne, przetrenowanie, hiperparametry

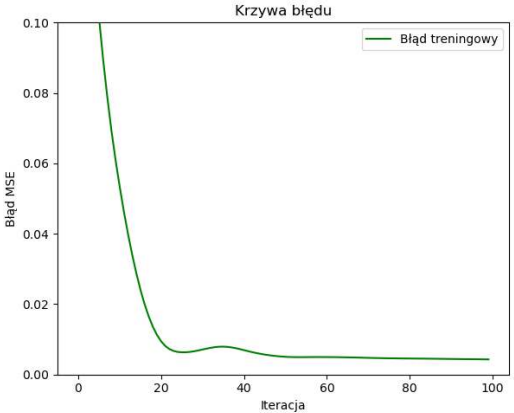

Iteracja 25
Dane treningowe
Predykcja z modelu
y
X
0.0
0.2
0.4
0.6
0.8
−2
−1
0
1
2

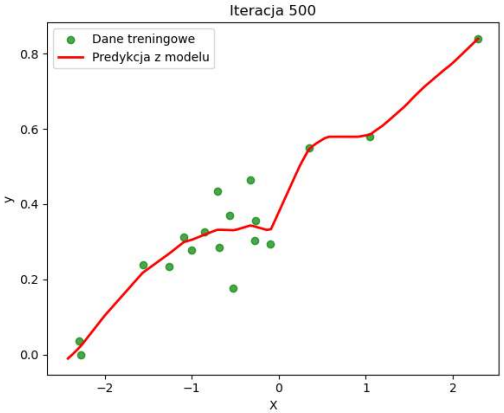

Krzywa błędu
Błąd treningowy
Błąd MSE
Iteracja
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0
5
10
15
20
25

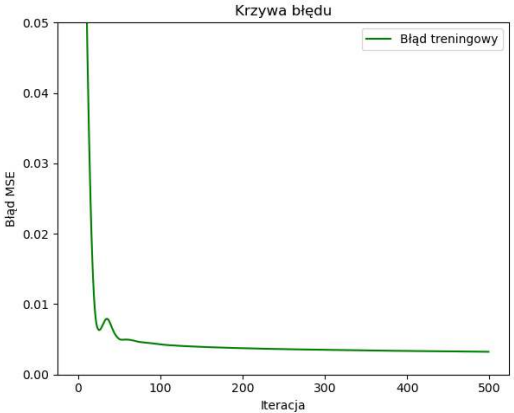

Iteracja 50
Dane treningowe
Predykcja z modelu
y
X
0.0
0.2
0.4
0.6
0.8
−2
−1
0
1
2

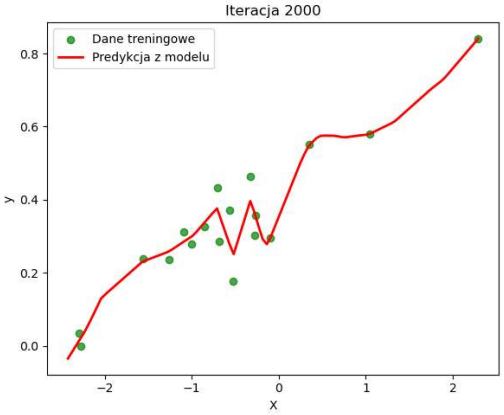

Krzywa błędu
Błąd treningowy
Błąd MSE
Iteracja
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0
10
20
30
40
50

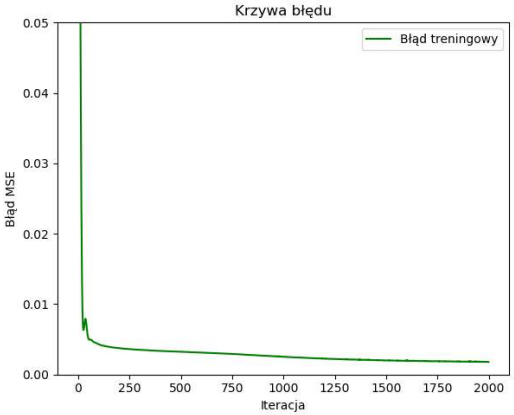

Iteracja 100
Dane treningowe
Predykcja z modelu
y
X
0.0
0.2
0.4
0.6
0.8
−2
−1
0
1
2

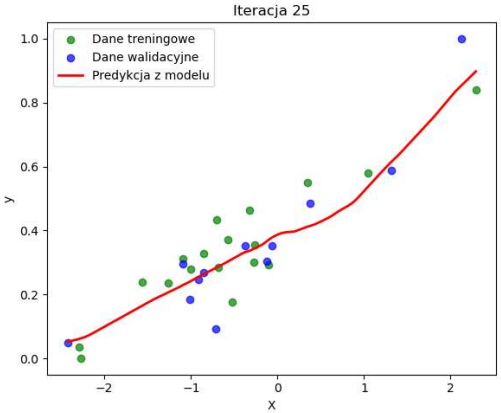

Krzywa błędu
Błąd treningowy
Błąd MSE
Iteracja
0.00
0.02
0.04
0.06
0.08
0.10
0
20
40
60
80
100

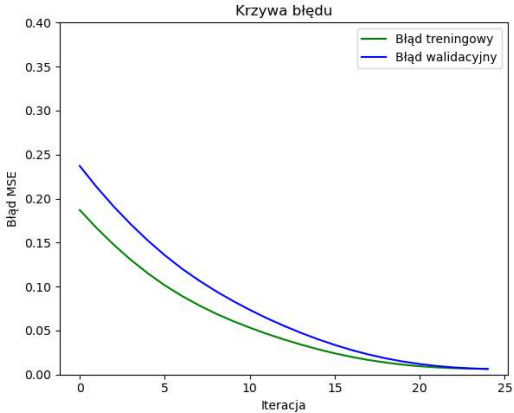

Iteracja 500
Dane treningowe
Predykcja z modelu
y
X
0.0
0.2
0.4
0.6
0.8
−2
−1
0
1
2

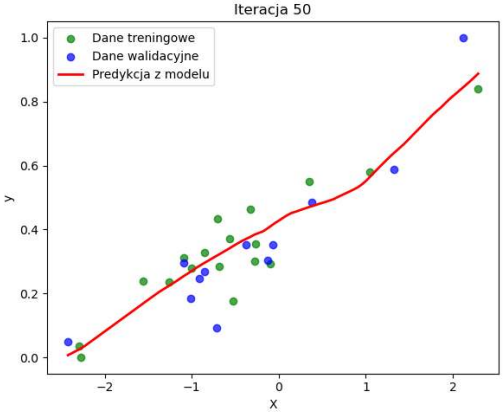

Krzywa błędu
Błąd treningowy
Błąd MSE
Iteracja
0.00
0.01
0.02
0.03
0.04
0.05
0
100
200
300
400
500

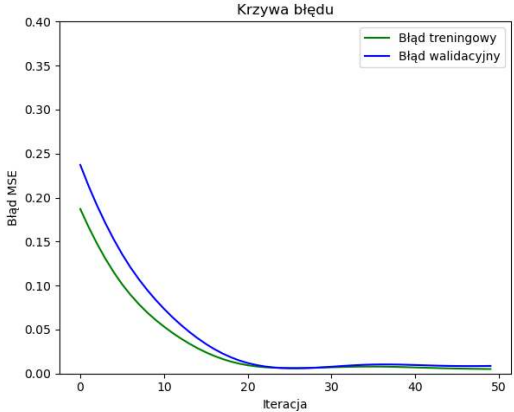

Iteracja 2000
Dane treningowe
Predykcja z modelu
y
X
0.0
0.2
0.4
0.6
0.8
−2
−1
0
1
2

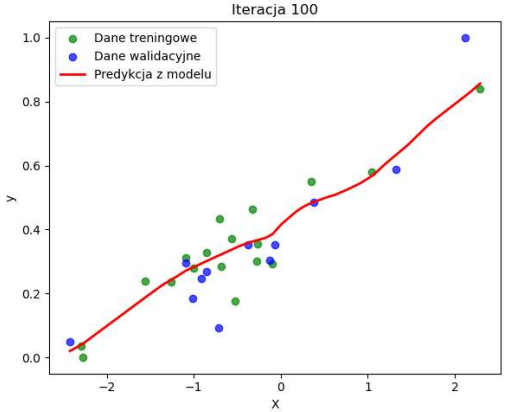

Krzywa błędu
Błąd treningowy
Błąd MSE
Iteracja
0.00
0.01
0.02
0.03
0.04
0.05
0
250
500
750
1000
1250
1500
1750
2000

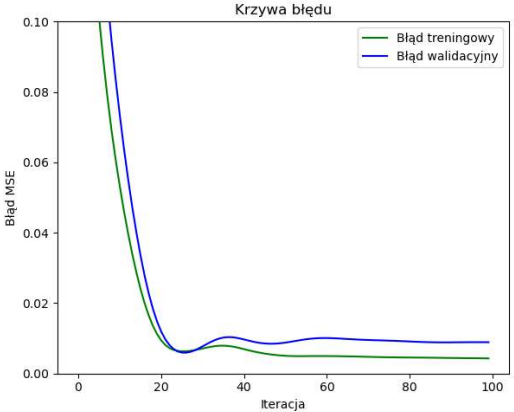

Iteracja 25
Dane treningowe
Dane walidacyjne
Predykcja z modelu
y
X
1.0
0.8
0.6
0.4
0.2
0.0
−2
−1
0
1
2

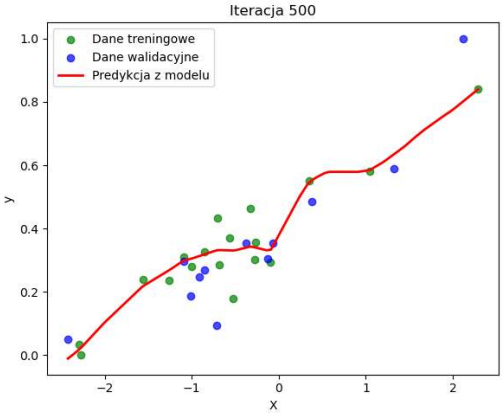

Krzywa błędu
Błąd treningowy
Błąd walidacyjny
Błąd MSE
Iteracja
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
5
10
15
20
25

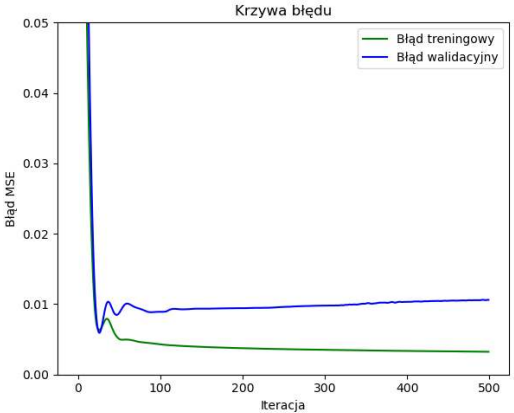

Iteracja 50
Dane treningowe
Dane walidacyjne
Predykcja z modelu
y
X
1.0
0.8
0.6
0.4
0.2
0.0
−2
−1
0
1
2

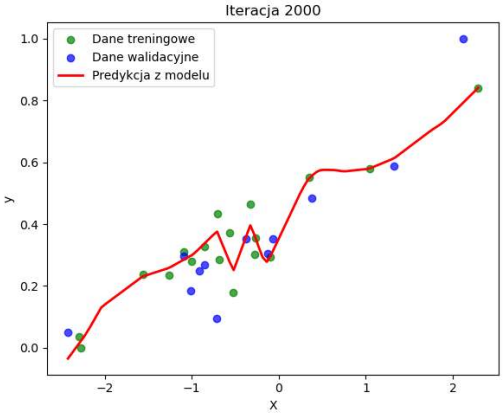

Krzywa błędu
Błąd treningowy
Błąd walidacyjny
Błąd MSE
Iteracja
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
10
20
30
40
50

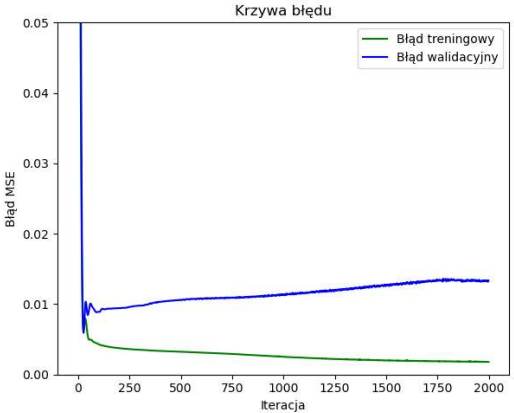

Iteracja 100
Dane treningowe
Dane walidacyjne
Predykcja z modelu
y
X
1.0
0.8
0.6
0.4
0.2
0.0
−2
−1
0
1
2


Krzywa błędu
Błąd treningowy
Błąd walidacyjny
Błąd MSE
Iteracja
0.10
0.08
0.06
0.04
0.02
0.00
0
20
40
60
80
100


Iteracja 500
Dane treningowe
Dane walidacyjne
Predykcja z modelu
y
X
1.0
0.8
0.6
0.4
0.2
0.0
−2
−1
0
1
2


Krzywa błędu
Błąd treningowy
Błąd walidacyjny
Błąd MSE
Iteracja
0.05
0.04
0.03
0.02
0.01
0.00
0
100
200
300
400
500


Iteracja 2000
Dane treningowe
Dane walidacyjne
Predykcja z modelu
y
X


Krzywa błędu
Błąd treningowy
Błąd walidacyjny
Błąd MSE
Iteracja

zbiór treningowy optymalizacja wag modelu

zbiór walidacyjny zabezpiecza przed przetrenowaniem, pozwala wybrać hiperparametry (np. learning rate 0.01 czy 0.001?), ułatwia decyzję o zakończeniu treningu, diagnostyka procesu treningu

zbiór testowy oszacowanie rzeczywistej wydajności (skuteczności, wyniku) wytrenowanego modelu

Zbiory rozłączne!

- **Przetrenowanie** – przesadne dopasowanie do zbioru treningowego
- **Generalizacja** – dopasowanie do charakterystyki danych, którą zbiór treningowy reprezentuje

Zmienne procesu treningu:

1. Dane
   - 1.1 Rozmiar zbioru, liczba przykładów – czy jest wystarczający?
   - 1.2 Diagnostyka przykładów – podobieństwo/różnorodność? Outliery?
2. Model
   - 2.1 Architekura – liczba warstw, liczba neuronów
   - 2.2 Dobór hiperparametrów, regularyzacja
   - 2.3 Rozmiar modelu względem rozmiaru zbioru danych
3. Trening
   - 3.1 Dostępne zasoby obliczeniowe – przeszukiwanie przestrzeni hiperparametrów, sprawdzenie różnych stanów początkowych
   - 3.2 Weryfikacja zbieżności na podzbiorze przykładów
   - 3.3 Diagnostyka wyników

Co jeżeli zbiór danych jest niewystarczający? Augmentacja danych (**data augmentation**)

- Wprowadzenie szumu (różnego rodzaju)
- Edycja strukturalna (wycięcie fragmentu, obroty, skalowanie, ...)
- Składanie nowego przykładu z kilku przykładów
- ...

Źródło ilustracji: T. Kumar et al, Image Data Augmentation Approaches: A Comprehensive Survey and Future directions, arXiv:2301.02830

Co jeżeli zbiór danych jest niewystarczający? Augmentacja danych (**data augmentation**)

- Wprowadzenie szumu (różnego rodzaju)
- Edycja strukturalna (wycięcie fragmentu, obroty, skalowanie, ...)
- Składanie nowego przykładu z kilku przykładów
- ...

Źródło ilustracji: T. Kumar et al, Image Data Augmentation Approaches: A Comprehensive Survey and Future directions, arXiv:2301.02830

Co jeżeli zbiór danych jest niewystarczający? Augmentacja danych (**data augmentation**)

- Wprowadzenie szumu (różnego rodzaju)
- Edycja strukturalna (wycięcie fragmentu, obroty, skalowanie, . . .)
- Składanie nowego przykładu z kilku przykładów
- . . .

Źródło ilustracji: T. Kumar et al, Image Data Augmentation Approaches: A Comprehensive Survey and Future directions, arXiv:2301.02830

Konstrukcja sieci

Softmax

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$

Regresja – otrzymujemy wartość liczbową – jak dobry jest owoc w skali 0-10

A podział na „dobry" i „zły" – klasyfikacja?

Modyfikacja wyjścia sieci, tak żeby na końcu

- y_{21} wartość liczbowa określająca jak bardzo przykład należy do klasy „dobry"
- y_{22} [...] do klasy „zły"

Problematyczne uczenie, bo np.

- $y_{21} = 0.02, y_{22} = 0.6 \rightarrow$ „zły owoc"
- $y_{21} = 1.7, y_{22} = 0.6 \rightarrow$ „dobry owoc"

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$

Regresja – otrzymujemy wartość liczbową – jak dobry jest owoc w skali 0-10

A podział na „dobry" i „zły" – klasyfikacja?

Modyfikacja wyjścia sieci, tak żeby na końcu

- y_{21} wartość liczbowa określająca jak bardzo przykład należy do klasy „dobry"
- y_{22} [...] do klasy „zły"

Problematyczne uczenie, bo np.

- $y_{21} = 0.02, y_{22} = 0.6 \rightarrow$ „zły owoc"
- $y_{21} = 1.7, y_{22} = 0.6 \rightarrow$ „dobry owoc"

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$

Regresja – otrzymujemy wartość liczbową – jak dobry jest owoc w skali 0-10

A podział na „dobry" i „zły" – klasyfikacja?

Modyfikacja wyjścia sieci, tak żeby na końcu

- y_{21} wartość liczbowa określająca jak bardzo przykład należy do klasy „dobry"
- y_{22} [...] do klasy „zły"

Problematyczne uczenie, bo np.

- $y_{21} = 0.02, y_{22} = 0.6 \rightarrow$ „zły owoc"
- $y_{21} = 1.7, y_{22} = 0.6 \rightarrow$ „dobry owoc"

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$

Regresja – otrzymujemy wartość liczbową – jak dobry jest owoc w skali 0-10

A podział na „dobry" i „zły" – klasyfikacja?

Modyfikacja wyjścia sieci, tak żeby na końcu

- y_{21} wartość liczbowa określająca jak bardzo przykład należy do klasy „dobry"
- y_{22} [...] do klasy „zły"

Problematyczne uczenie, bo np.

- $y_{21} = 0.02, y_{22} = 0.6 \rightarrow$ „zły owoc"
- $y_{21} = 1.7, y_{22} = 0.6 \rightarrow$ „dobry owoc"

$y_{11} = \max(20x - 10, 0) \quad y_{12} = \max(40x - 200, 0) \quad y_{21} = \max(y_{11} - y_{12}, 0)$

Regresja – otrzymujemy wartość liczbową – jak dobry jest owoc w skali 0-10

A podział na „dobry" i „zły" – klasyfikacja?

Modyfikacja wyjścia sieci, tak żeby na końcu

- y_{21} wartość liczbowa określająca jak bardzo przykład należy do klasy „dobry"
- y_{22} [...] do klasy „zły"

Problematyczne uczenie, bo np.

- $y_{21} = 0.02, y_{22} = 0.6 \rightarrow$ „zły owoc"
- $y_{21} = 1.7, y_{22} = 0.6 \rightarrow$ „dobry owoc"

Softmax: $\sigma(\mathbf{y})_i = \frac{e^{y_i}}{\sum_{j=1}^{N} e^{y_j}}$

$\mathbf{y}_a = \left[0.02, 0.6\right]$

$\sigma\left(\left[0.02, 0.6\right]\right)_1 = \frac{e^{0.02}}{e^{0.02}+e^{0.6}} \approx 0.36$

$\sigma\left(\left[0.02, 0.6\right]\right)_2 = \frac{e^{0.6}}{e^{0.02}+e^{0.6}} \approx 0.64$

$\sigma\left(\mathbf{y}_a\right) \approx \left[0.36, 0.64\right]$

$\mathbf{y}_b = \left[1.7, 0.6\right]$

$\sigma\left(\mathbf{y}_b\right) \approx \left[0.75, 0.25\right]$

Szacowane prawdopodobieństwo przynależności do poszczególnych klas

Softmax: $\sigma(\mathbf{y})_i = \frac{e^{y_i}}{\sum_{j=1}^{N} e^{y_j}}$

$\mathbf{y}_a = \left[0.02, 0.6\right]$

$\sigma\left(\left[0.02, 0.6\right]\right)_1 = \frac{e^{0.02}}{e^{0.02}+e^{0.6}} \approx 0.36$

$\sigma\left(\left[0.02, 0.6\right]\right)_2 = \frac{e^{0.6}}{e^{0.02}+e^{0.6}} \approx 0.64$

$\sigma\left(\mathbf{y}_a\right) \approx \left[0.36, 0.64\right]$

$\mathbf{y}_b = \left[1.7, 0.6\right]$

$\sigma\left(\mathbf{y}_b\right) \approx \left[0.75, 0.25\right]$

Szacowane prawdopodobieństwo przynależności do poszczególnych klas

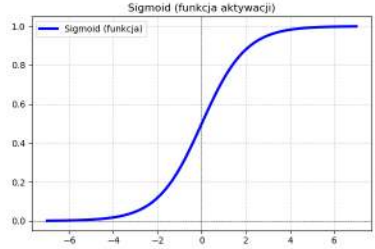
Softmax: $\sigma(\mathbf{y})_i = \frac{e^{y_i}}{\sum_{j=1}^{N} e^{y_j}}$

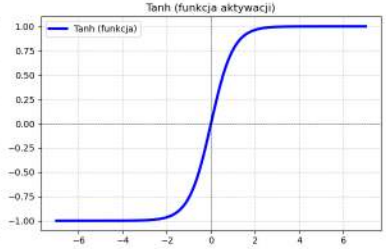
$$\mathbf{y}_a = \left[0.02, 0.6\right]$$
$$\sigma\left(\left[0.02, 0.6\right]\right)_1 = \frac{e^{0.02}}{e^{0.02}+e^{0.6}} \approx 0.36$$
$$\sigma\left(\left[0.02, 0.6\right]\right)_2 = \frac{e^{0.6}}{e^{0.02}+e^{0.6}} \approx 0.64$$
$$\sigma\left(\mathbf{y}_a\right) \approx \left[0.36, 0.64\right]$$

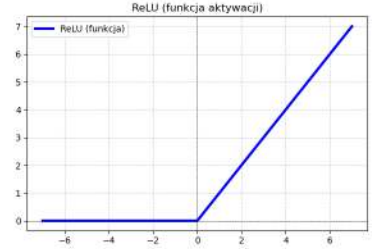
$$\mathbf{y}_b = \left[1.7, 0.6\right]$$
$$\sigma\left(\mathbf{y}_b\right) \approx \left[0.75, 0.25\right]$$

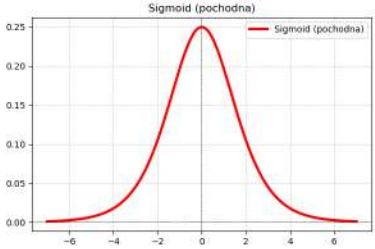
Szacowane prawdopodobieństwo przynależności do poszczególnych klas

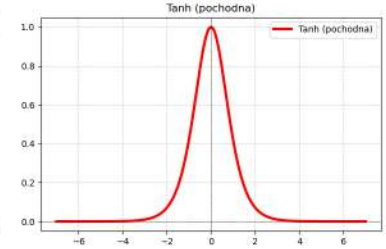
Funkcje aktywacji

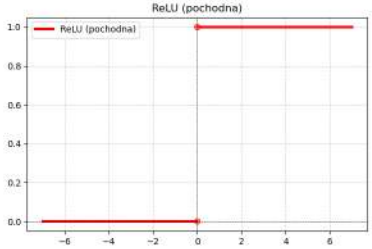

Sigmoid (funkcja aktywacji)
Sigmoid (funkcja)


Tanh (funkcja aktywacji)
Tanh (funkcja)


ReLU (funkcja aktywacji)
ReLU (funkcja)


Sigmoid (pochodna)
Sigmoid (pochodna)


Tanh (pochodna)
Tanh (pochodna)


ReLU (pochodna)
ReLU (pochodna)

Rozważmy zbiór danych (fragment zbioru Iris):

```
[[5.1 3.5 1.4 0.2]
 [4.9 3.  1.4 0.2]
 [4.7 3.2 1.3 0.2]
 [4.6 3.1 1.5 0.2]
 [5.  3.6 1.4 0.2]
 [5.4 3.9 1.7 0.4]
 [4.6 3.4 1.4 0.3]
 [5.  3.4 1.5 0.2]
 [4.4 2.9 1.4 0.2]
 [4.9 3.1 1.5 0.1]]
```

i neuron ReLU z wagami $\left[0.2, -0.4, 0.1, -0.5\right]$

$5.1 \times 0.2 + 3.5 \times -0.4 + 1.4 \times 0.1 + 0.2 \times -0.5 = 1.02 - 1.4 + 0.14 - 0.1 = -0.34 \ldots$

. . . i tak dla każdego – neuron jest „martwy” (brak niezerowej wartości, zerowy gradient)

Rozważmy zbiór danych (fragment zbioru Iris):

```
[[5.1 3.5 1.4 0.2]
 [4.9 3.  1.4 0.2]
 [4.7 3.2 1.3 0.2]
 [4.6 3.1 1.5 0.2]
 [5.  3.6 1.4 0.2]
 [5.4 3.9 1.7 0.4]
 [4.6 3.4 1.4 0.3]
 [5.  3.4 1.5 0.2]
 [4.4 2.9 1.4 0.2]
 [4.9 3.1 1.5 0.1]]
```

i neuron ReLU z wagami $\left[0.2, -0.4, 0.1, -0.5\right]$

$5.1 \times 0.2 + 3.5 \times -0.4 + 1.4 \times 0.1 + 0.2 \times -0.5 = 1.02 - 1.4 + 0.14 - 0.1 = -0.34 \ldots$

. . . i tak dla każdego – neuron jest „martwy” (brak niezerowej wartości, zerowy gradient)

$\mathrm{ReLU}(x) = \max(x, 0)$

Alternatywy:

- Leaky ReLU $f(x) = \begin{cases} x & x > 0, \\ \alpha x & x \leqslant 0 \end{cases}$, np. $\alpha = 0.1$
- Parametric ReLU (α jest parametrem wyznaczanym w trakcie treningu)
- SiLU $f(x) = x \times \mathrm{sigmoid}(x)$
- ELU $f(x) = \begin{cases} x & x > 0, \\ \alpha(e^x - 1) & x \leqslant 0 \end{cases}$, $\alpha \geqslant 0$ jest hiperparametrem
- Softplus $f(x) = \ln(1 + e^x)$
- ...

Funkcje błędu (loss)

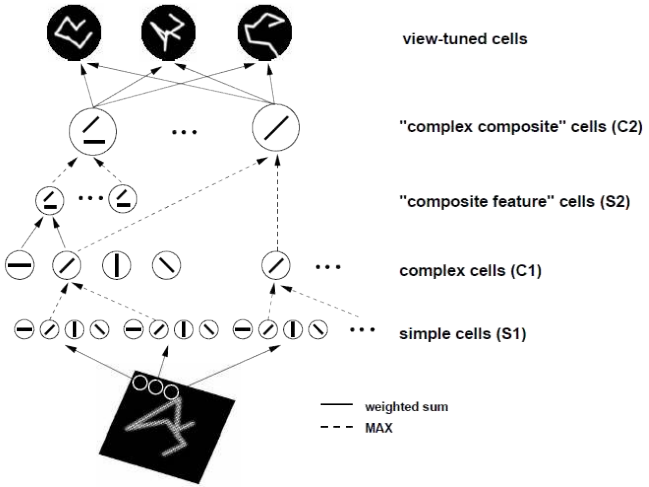
- Mean Squared Error (MSE) $L(x, y) = (x - y)^2$
- Mean Absolute Error (MAE) $L(x, y) = |x - y|$
- Negative Log-Likelihood Loss function (NLL) $L(x, y) = -y \log x - (1 - y) \log(1 - x)$
- Cross-Entropy Loss ($\approx$ softmax + NLL)
- Margin Ranking Loss $L(x_1, x_2, y) = \max(0, -y(x_1 - x_2) + m)$, ($m$ margines)
- Kullback-Leibler Divergence Loss $L(x, y) = y(\log y - x)$
- ...

Zaawansowane architektury – sieci konwolucyjne

Problem: rozpoznawanie obiektów w obrazach

- Obiekt przesunięty, obrócony, przeskalowany, zdeformowany (operacja rzutowania)
- Cechy charakterystyczne – złożone, rzadkie, różne dla różnych rodzajów obiektów

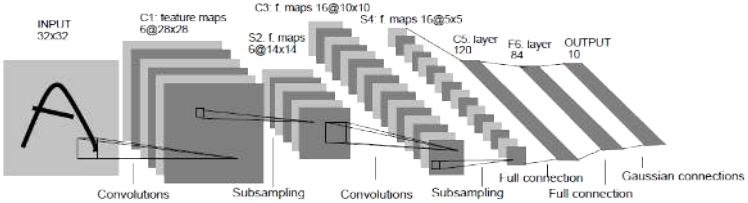
Model HMAX:



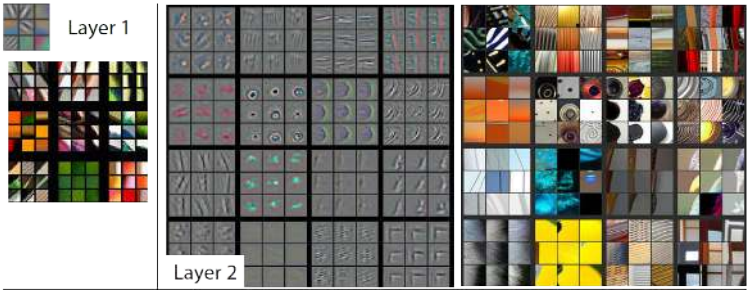
Źródło ilustracji: Riesenhuber, M. & Poggio, T. Hierarchical models of object recognition in cortex. Nat. Neurosci. 2, 1019-1025

Architektura konwolucyjna:

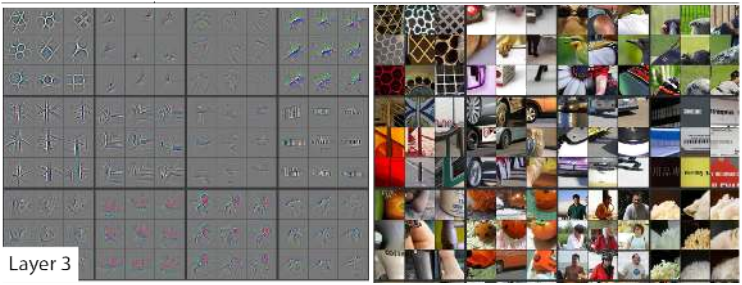
- Filtry konwolucyjne (uczone/trenowane)
- Operacja „zbierania” (pooling)
- Pary konwolucja+pooling powtórzone kilka razy
- Na końcu kilka warstw sieci MLP



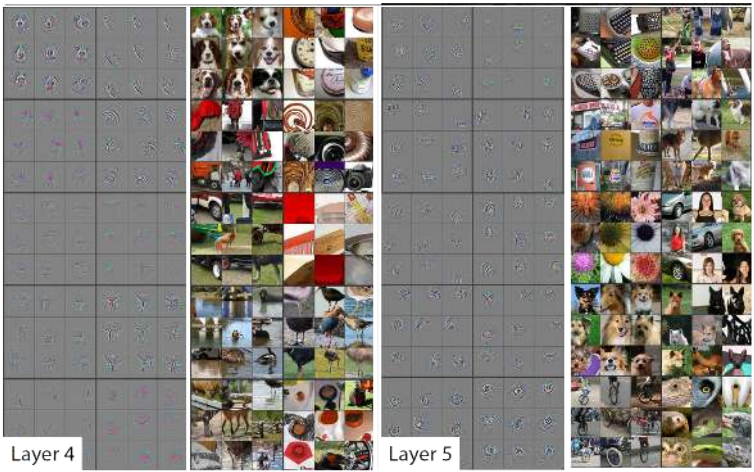
Źródło ilustracji: Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. (1998). "Gradient-based learning applied to document recognition"(PDF). Proceedings of the IEEE. 86 (11): 2278–2324



Źródło ilustracji: Zeiler, M.D., Fergus, R. (2014). Visualizing and Understanding Convolutional Networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds) Computer Vision – ECCV 2014

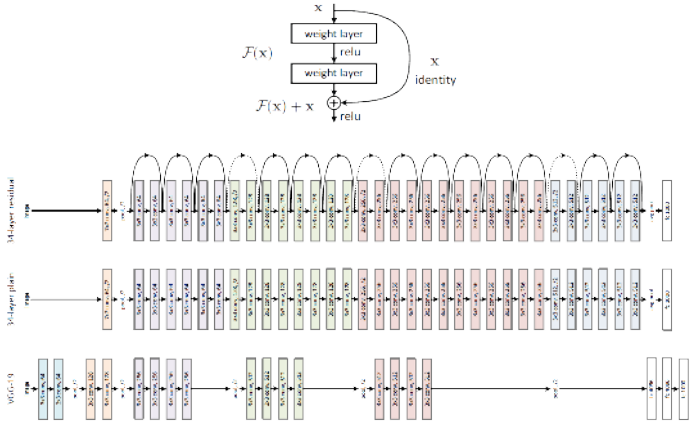


Źródło ilustracji: Zeiler, M.D., Fergus, R. (2014). Visualizing and Understanding Convolutional Networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds) Computer Vision – ECCV 2014



Źródło ilustracji: Zeiler, M.D., Fergus, R. (2014). Visualizing and Understanding Convolutional Networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds) Computer Vision – ECCV 2014

Sieci ResNet



Źródło ilustracji: K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition,"2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778

Zaawansowane architektury – sieci transformers

ChatGPT 4o

Share PR

Why is a Raven Like a Writing Desk?

"Why is a raven like a writing desk?" is a riddle posed by the Mad Hatter in Lewis Carroll's "Alice's Adventures in Wonderland." This famous riddle is known for its lack of a definitive answer. Carroll himself later suggested an answer in a preface to an 1896 edition of the book: "Because it can produce a few notes, though they are very flat; and it is nevar put with the wrong end in front!" This answer is a pun, as "nevar" is "raven" spelled backward, though it was later corrected to "never" in subsequent editions, losing the pun.

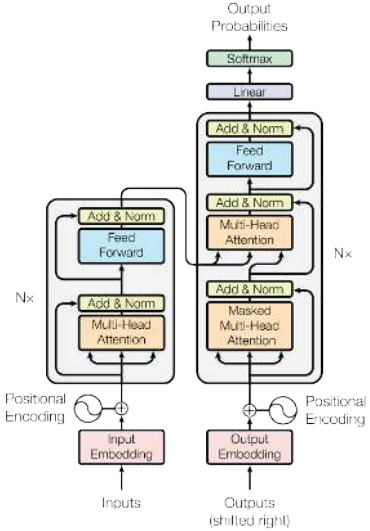
The riddle has been a subject of much debate and has inspired numerous answers and interpretations over the years, both serious and humorous. It reflects the whimsical and nonsensical tone of the book, where logic and conventional answers are often turned on their head.

Problem „sekwencja w sekwencję" (sequence to sequence), z odległymi zależnościami między słowami

```
                  Mary had a little lamb.
In the hospital, Mary had a little lamb.
```

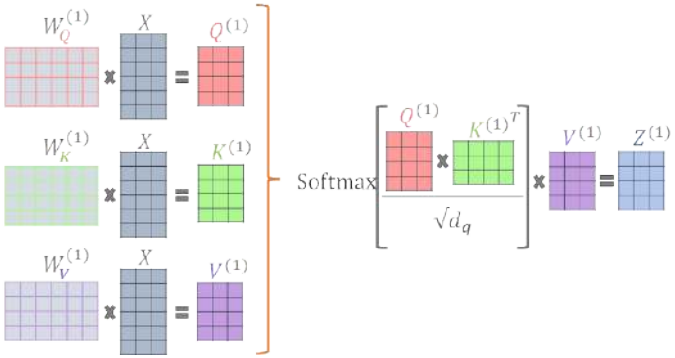
Krótka (i fragmentaryczna) historia prób rozwiązania:

- Sieci rekurencyjne
- Sieci LSTM (Long-Short Term Memory)
- Sieci transformers



Źródło ilustracji: Vaswani, A. et al (2017). "Attention is All you Need". Advances in Neural Information Processing Systems. 30

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$$



Źródło ilustracji: Ahmed, S., Nielsen, I.E., Tripathi, A. et al. Transformers in Time-Series Analysis: A Tutorial. Circuits Syst Signal Process 42, 7433-7466 (2023).

Źródło ilustracji: Ahmed, S., Nielsen, I.E., Tripathi, A. et al. Transformers in Time-Series Analysis: A Tutorial. Circuits Syst Signal Process 42, 7433-7466 (2023)

Sieć MLP – analiza działania dla przykładowych danych

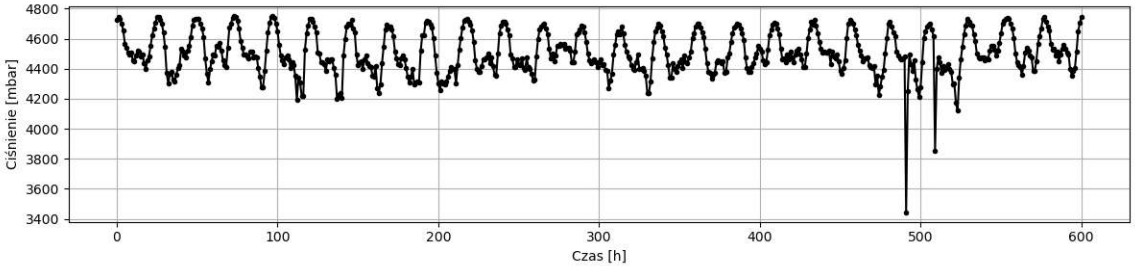
Przemysław Głomb

Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk

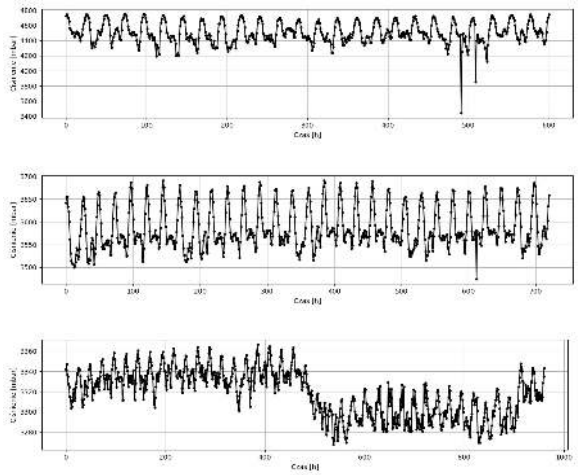
redaktor pomocniczy: Bartłomiej Gardas

Sieci neuronowe – „case study” weryfikacji danych

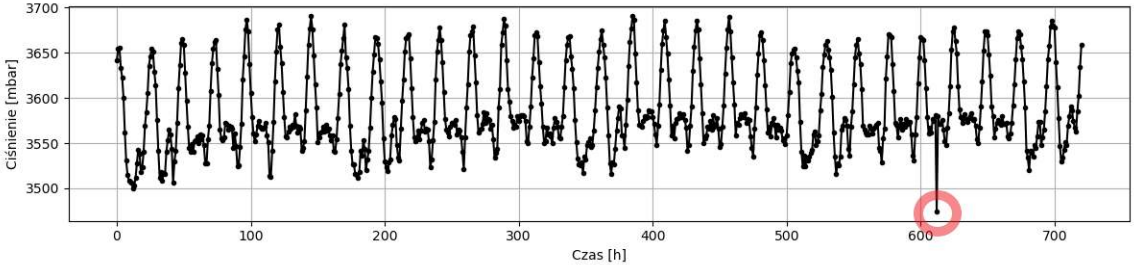
Pomiar ciśnienia na wejściu do strefy wodociągowej



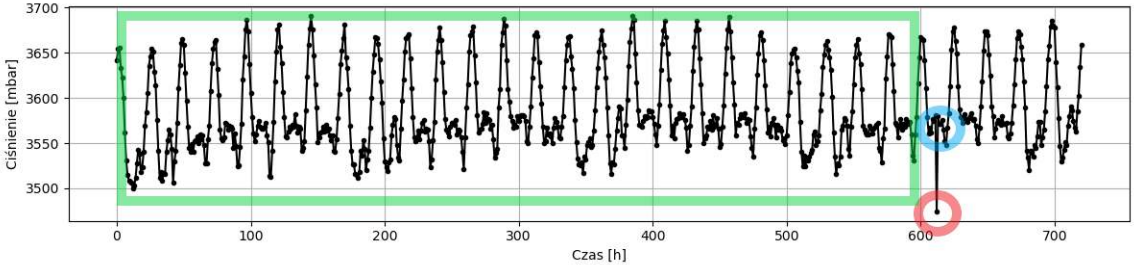
Pomiar ciśnienia na wejściu do strefy wodociągowej



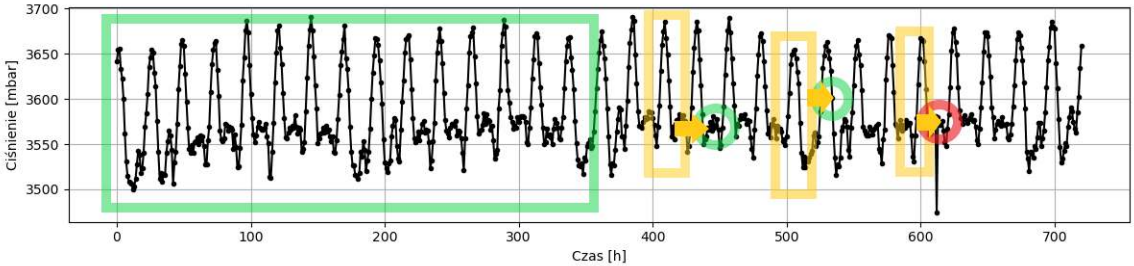
Weryfikacja danych



Anomalia → „nietypowe zachowanie”

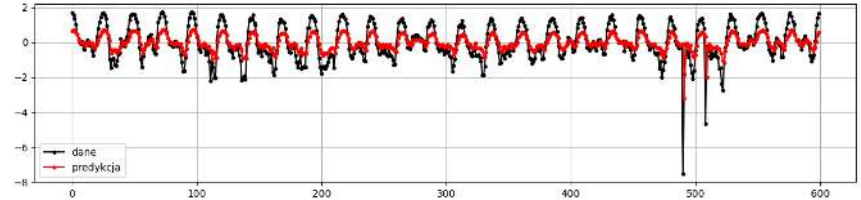


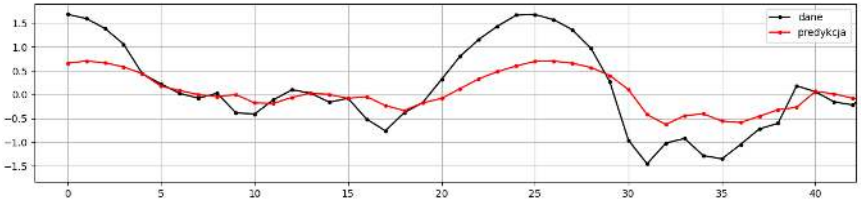
Weryfikacja na podstawie predykcji z modelu



Jeden, wyjściowy neuron

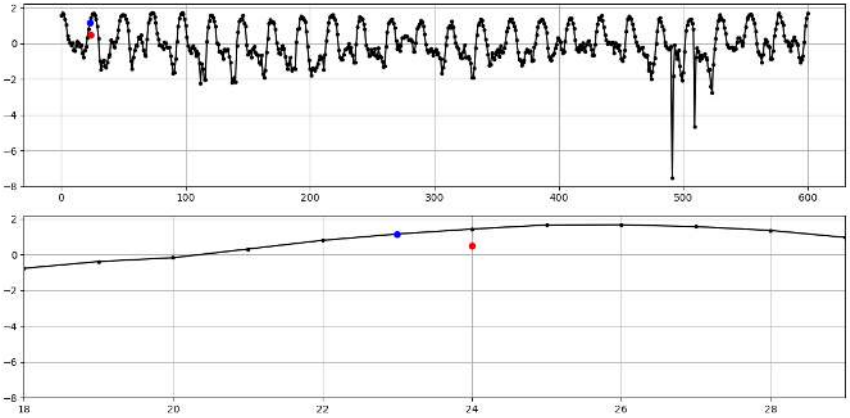
```
n_window, layers = 1, []
```





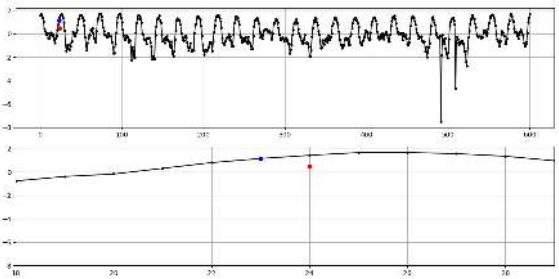
Jeden, wyjściowy neuron

```
n_window, layers = 1, []
```



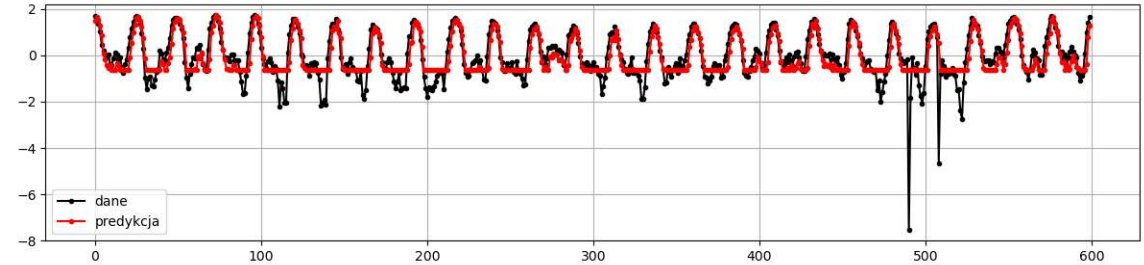
Jeden, wyjściowy neuron

```
n_window, layers = 1, []
```



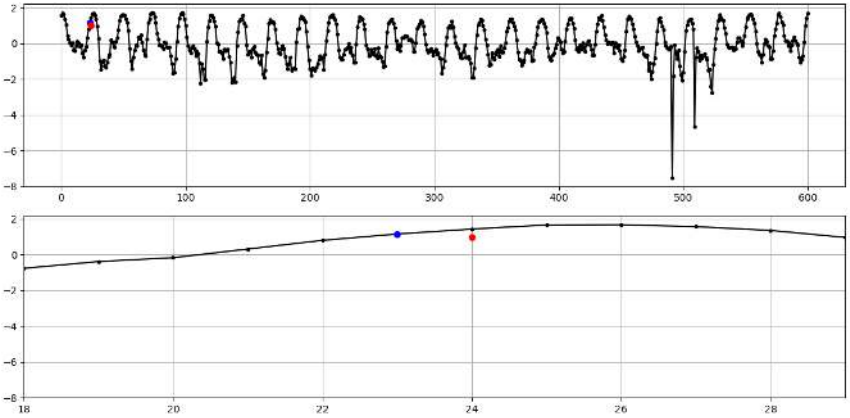
Jeden neuron w warstwie ukrytej

```
n_window, layers = 1, [1]
```



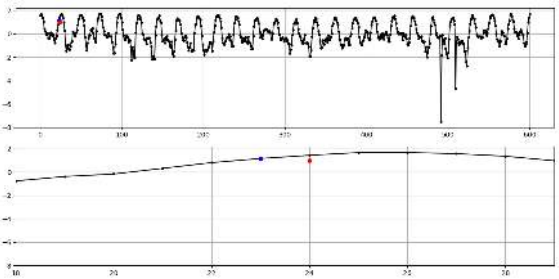
Jeden neuron w warstwie ukrytej

```
n_window, layers = 1, [1]
```



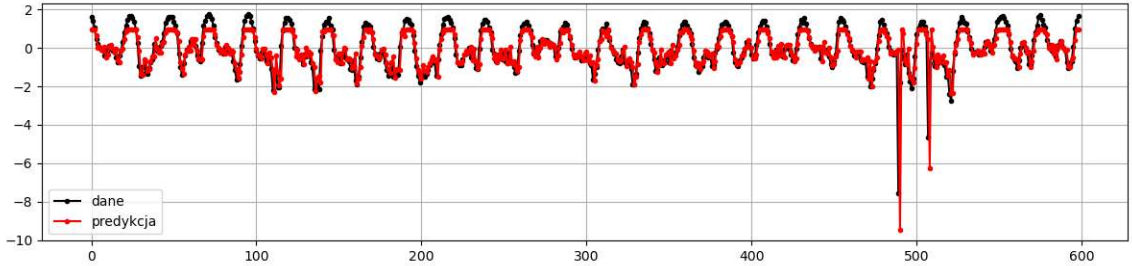
Jeden neuron w warstwie ukrytej

```
n_window, layers = 1, [1]
```



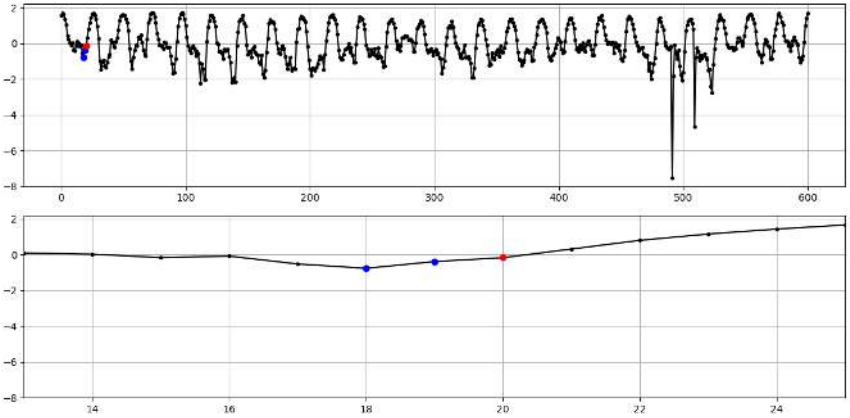
Dwa pomiary w historii

```
n_window, layers = 2, [1]
```



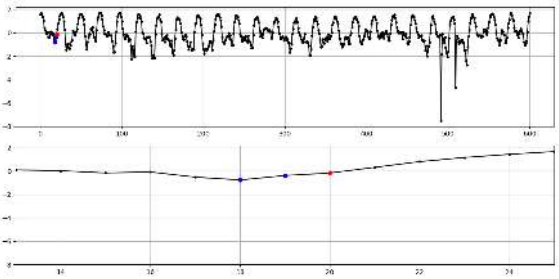
Dwa pomiary w historii

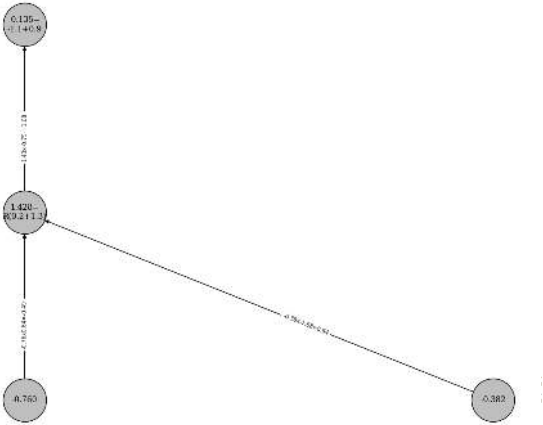
```
n_window, layers = 2, [1]
```



Dwa pomiary w historii

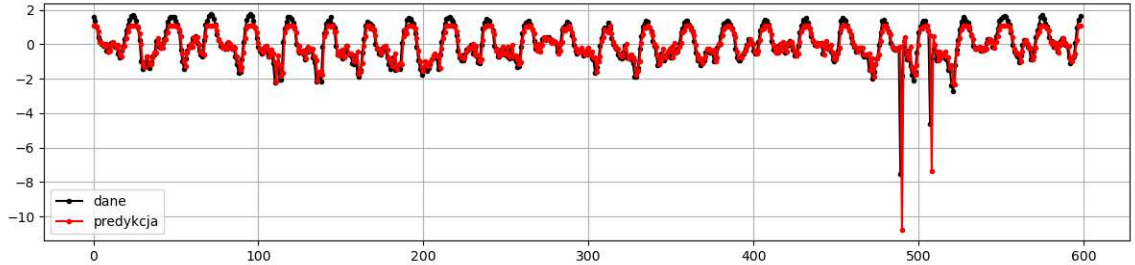
```
n_window, layers = 2, [1]
```





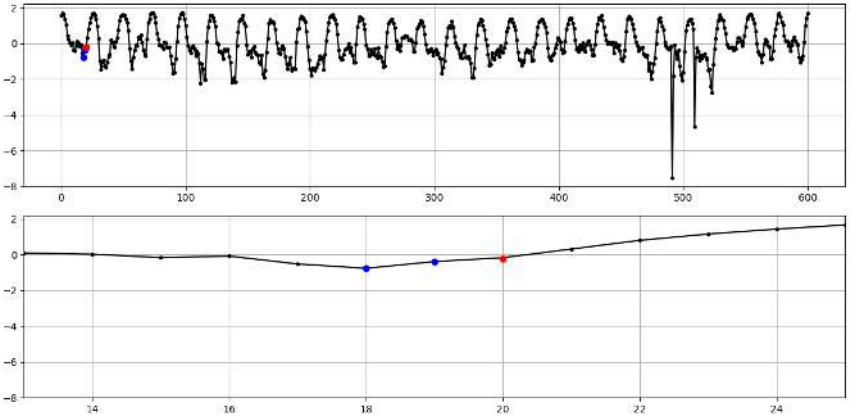
2 pomiary w historii i 2 neurony w w. ukrytej

```
n_window, layers = 2, [2]
```



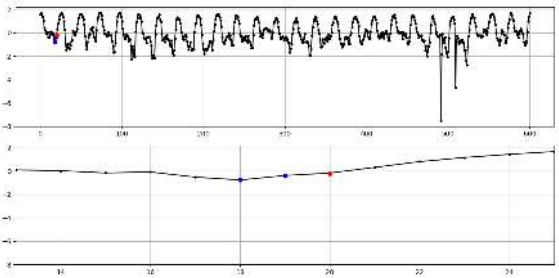
2 pomiary w historii i 2 neurony w w. ukrytej

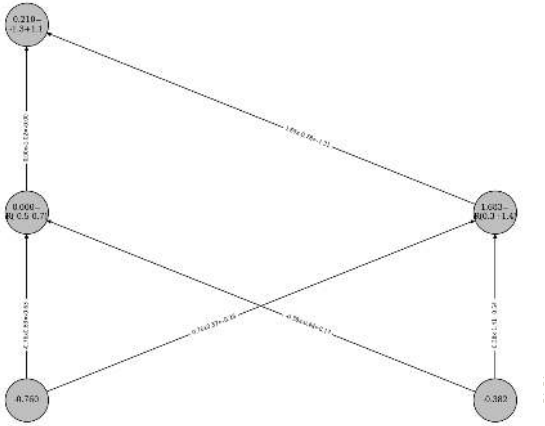
```
n_window, layers = 2, [2]
```



2 pomiary w historii i 2 neurony w w. ukrytej

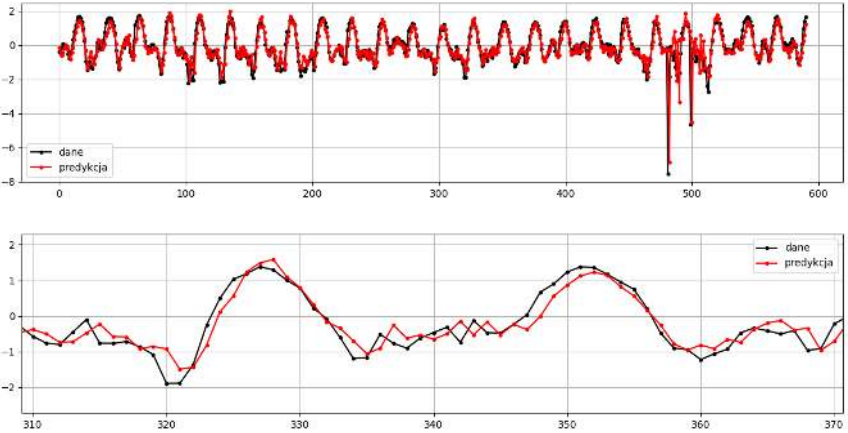
```
n_window, layers = 2, [2]
```





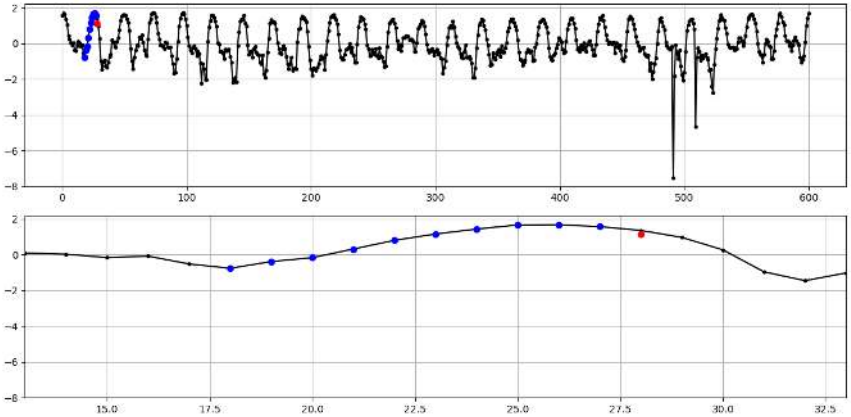
10 pomiarów w historii i 2 neurony w w. ukrytej

```
n_window, layers = 10, [2]
```



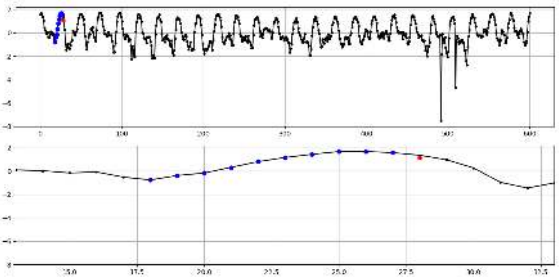
10 pomiarów w historii i 2 neurony w w. ukrytej

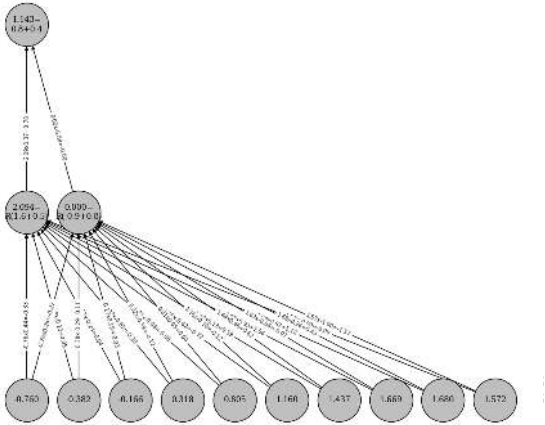
```
n_window, layers = 10, [2]
```



10 pomiarów w historii i 2 neurony w w. ukrytej

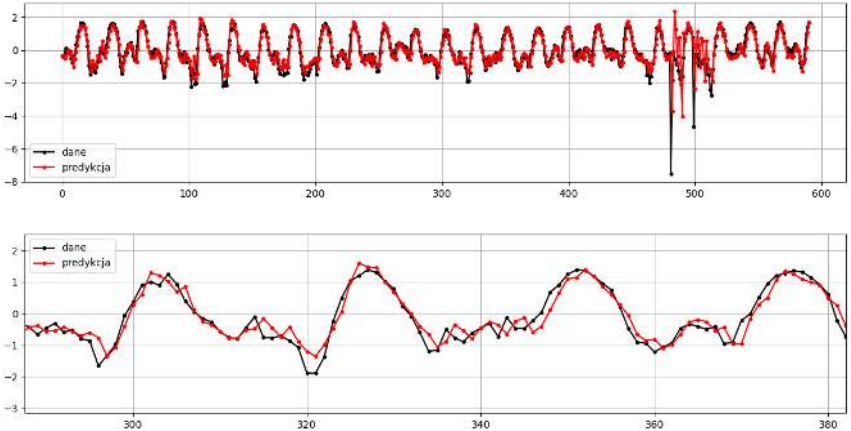
```
n_window, layers = 10, [2]
```





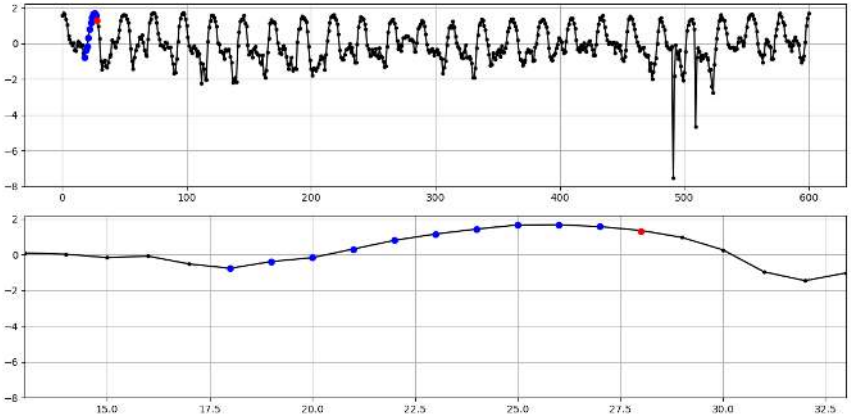
10 pomiarów w historii i 3 neurony w w. ukrytej

```
n_window, layers = 10, [3]
```



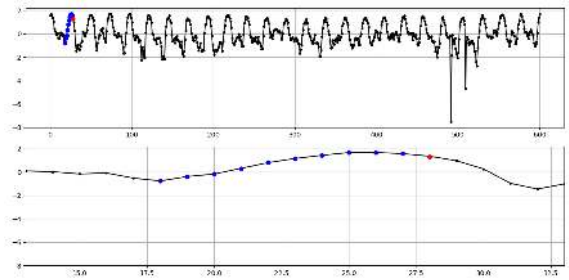
10 pomiarów w historii i 3 neurony w w. ukrytej

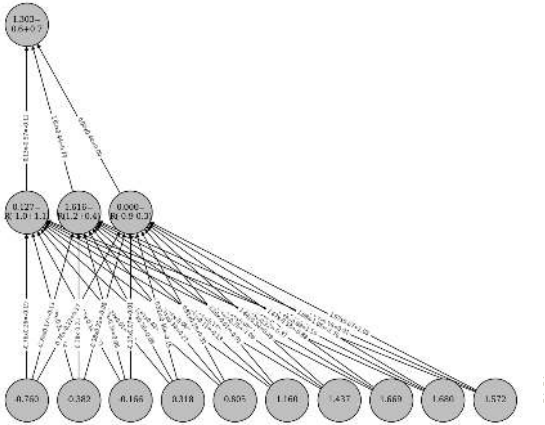
```
n_window, layers = 10, [3]
```



10 pomiarów w historii i 3 neurony w w. ukrytej

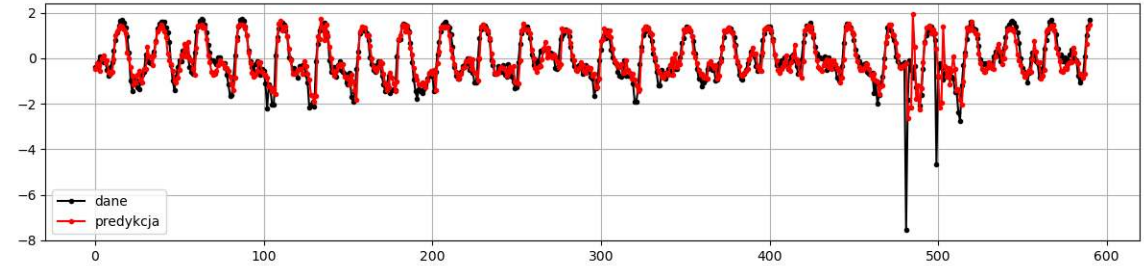
```
n_window, layers = 10, [3]
```





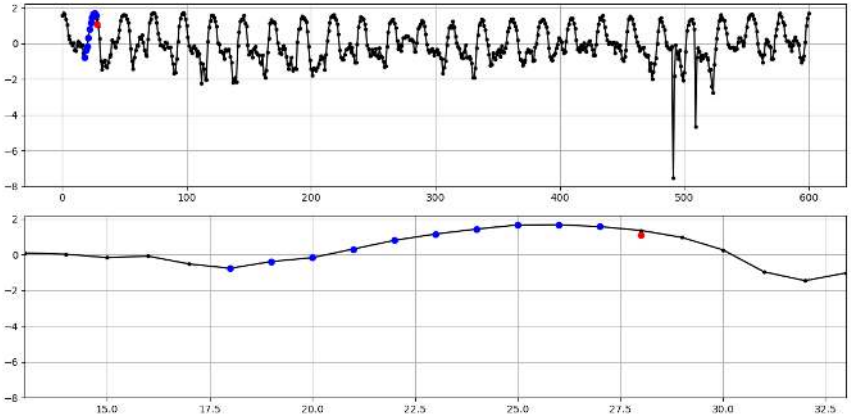
2 warstwy ukryte

```
n_window, layers = 10, [3, 2]
```



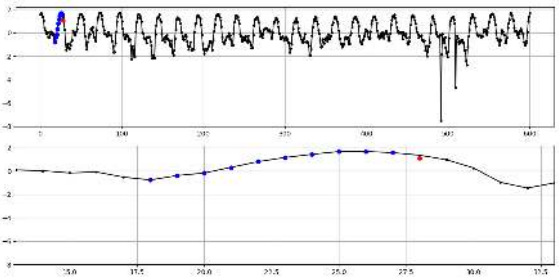
2 warstwy ukryte

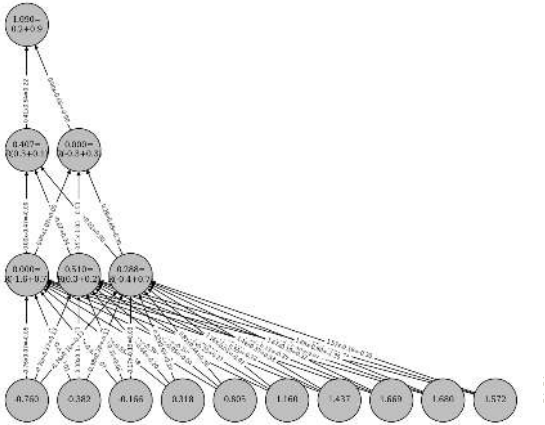
```
n_window, layers = 10, [3, 2]
```



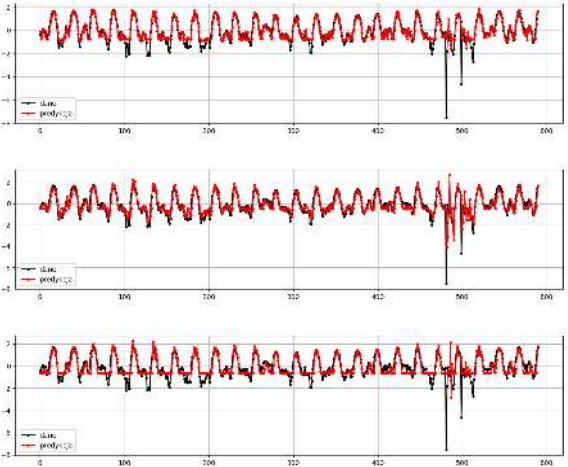
2 warstwy ukryte

```
n_window, layers = 10, [3, 2]
```

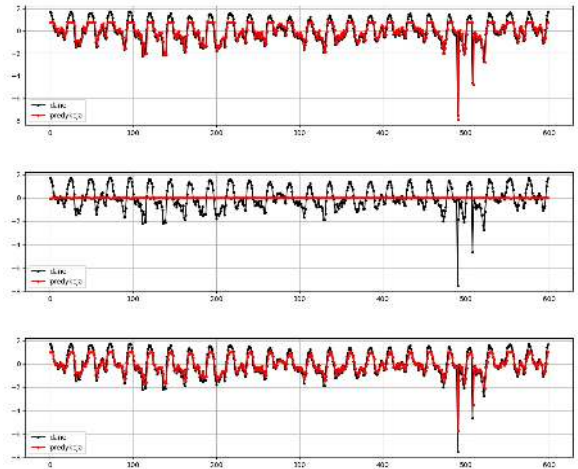




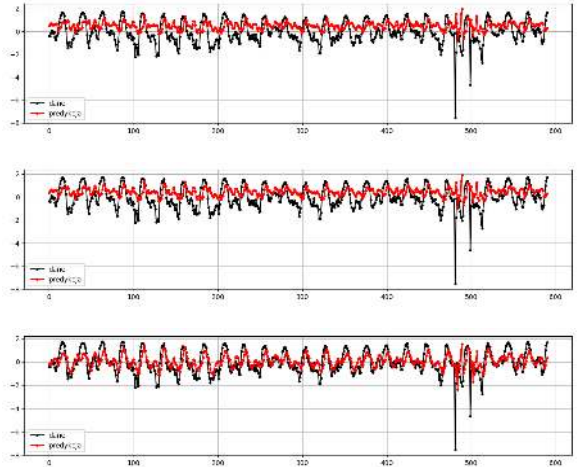
Wpływ wartości początkowych (10, [2])



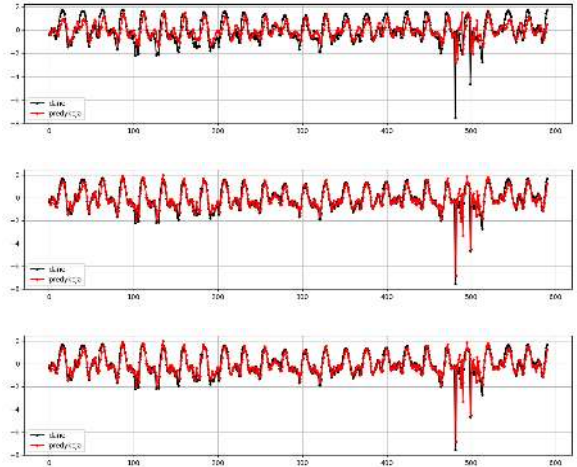
Wpływ wartości początkowych (1, [1])



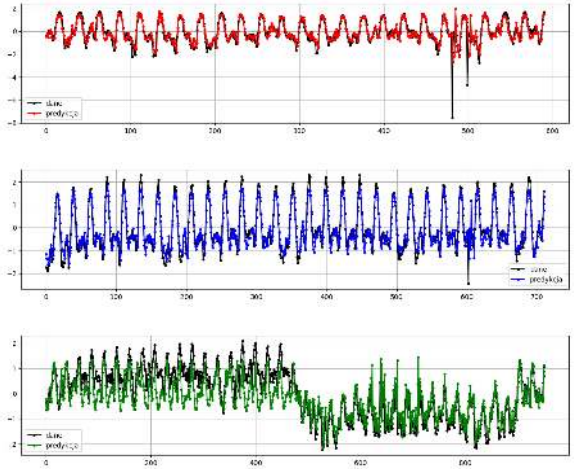
Wpływ liczby iteracji: 1, 10, 100



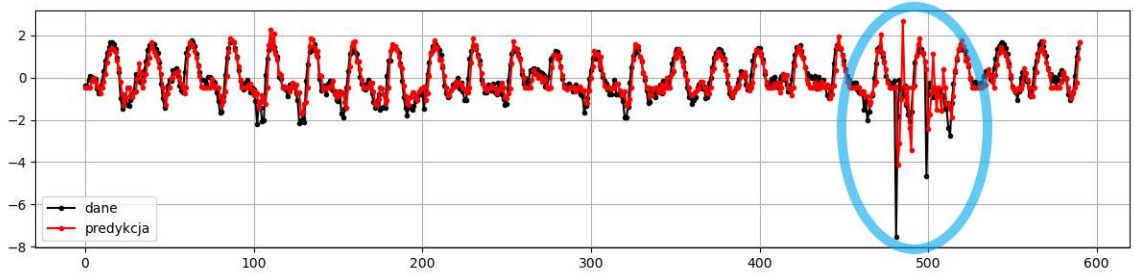
Wpływ liczby iteracji: 200, 1000, 10000



Inny sygnał?



Podsumowanie – wykrywanie anomalii



Statystyczne metody uczenia maszynowego – wprowadzenie

Przemysław Głomb

Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk

redaktor pomocniczy: Bartłomiej Gardas

AKADEMIA SZTUKI KWANTOWEJ

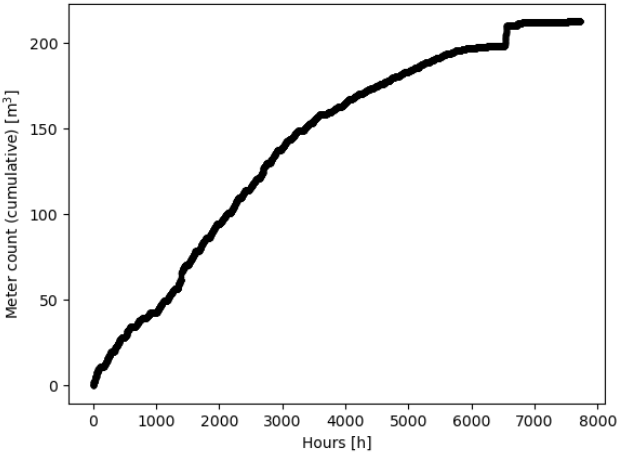
Ministerstwo Nauki i Szkolnictwa Wyższego

NAUKA DLA SPOŁECZEŃSTWA

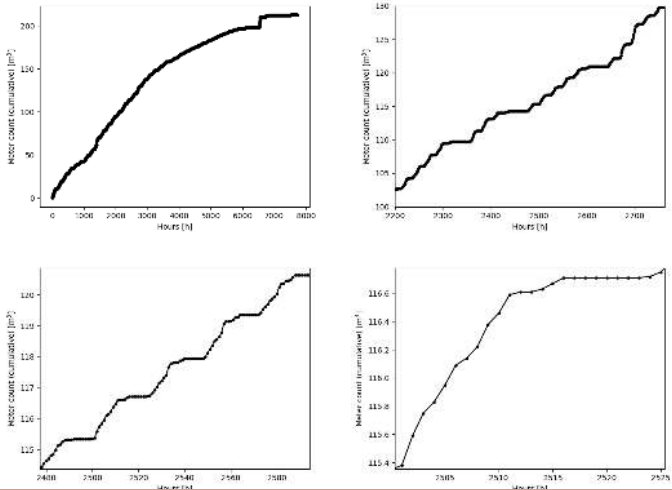
IITiS

Analiza przykładowego sygnału

Dane – przykładowy sygnał



Dane – przykładowy sygnał (zoom)



Dane – przykładowy sygnał – obserwacje/cechy

1. Szereg czasowy
2. Kumulatywne zliczanie impulsów – sygnał kumulatywny (ang. integrative) – różnicowanie (ang. differentiation)

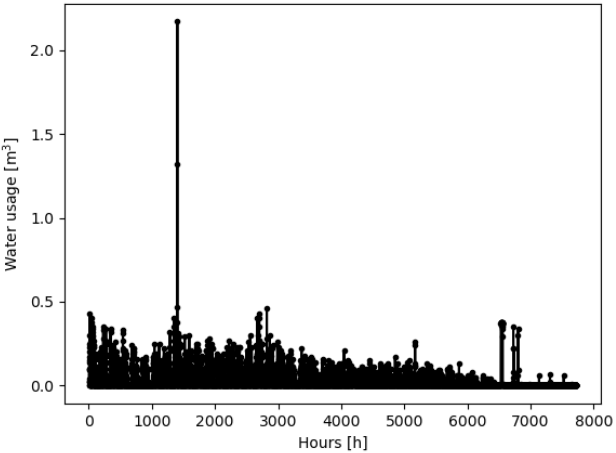
→ Sygnał, sygnał po różnicowaniu, sygnał po scałkowaniu

Dane – przykładowy sygnał – obserwacje/cechy

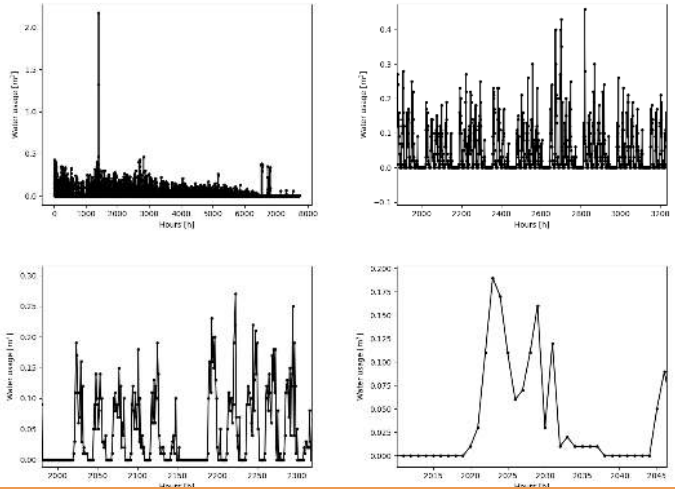
1. Szereg czasowy
2. Kumulatywne zliczanie impulsów – sygnał kumulatywny (ang. integrative) – różnicowanie (ang. differentiation)

   → Sygnał, sygnał po różnicowaniu, sygnał po scałkowaniu

Dane – przykładowy sygnał – różnicowanie

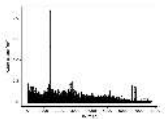


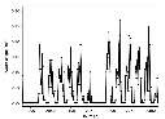
Dane – przykładowy sygnał – różnicowanie



Dane – przykładowy sygnał – obserwacje/cechy

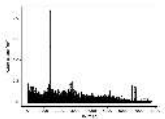
1. Szereg czasowy
2. Kumulatywne zliczanie impulsów – sygnał kumulatywny (ang. integrative) – różnicowanie (ang. differentiation)
   → Sygnał, sygnał po różnicowaniu, sygnał po scałkowaniu
3. Cykl – dobowy, tygodniowy
   → Cykle, sezonowość
4. Trend – zmienny
5. Pojedyncze zdarzenia – anomalie
6. Wartości liczbowe (min/max, typowe, rozkład wartości)

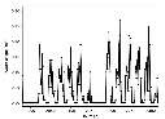




Dane – przykładowy sygnał – obserwacje/cechy

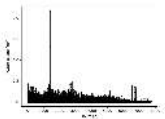
1. Szereg czasowy
2. Kumulatywne zliczanie impulsów – sygnał kumulatywny (ang. integrative) – różnicowanie (ang. differentiation)
   → Sygnał, sygnał po różnicowaniu, sygnał po scałkowaniu
3. Cykl – dobowy, tygodniowy
   → Cykle, sezonowość
4. Trend – zmienny
5. Pojedyncze zdarzenia – anomalie
6. Wartości liczbowe (min/max, typowe, rozkład wartości)

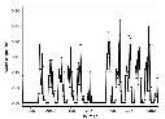




Dane – przykładowy sygnał – obserwacje/cechy

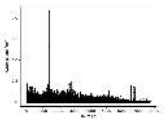
1. Szereg czasowy
2. Kumulatywne zliczanie impulsów – sygnał kumulatywny (ang. integrative) – różnicowanie (ang. differentiation)
   → Sygnał, sygnał po różnicowaniu, sygnał po scałkowaniu
3. Cykl – dobowy, tygodniowy
   → Cykle, sezonowość
4. Trend – zmienny
5. Pojedyncze zdarzenia – anomalie
6. Wartości liczbowe (min/max, typowe, rozkład wartości)

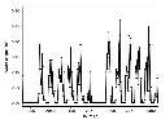




Dane – przykładowy sygnał – obserwacje/cechy

1. Szereg czasowy
2. Kumulatywne zliczanie impulsów – sygnał kumulatywny (ang. integrative) – różnicowanie (ang. differentiation)
   → Sygnał, sygnał po różnicowaniu, sygnał po scałkowaniu
3. Cykl – dobowy, tygodniowy
   → Cykle, sezonowość
4. Trend – zmienny
5. Pojedyncze zdarzenia – anomalie
6. Wartości liczbowe (min/max, typowe, rozkład wartości)





Bazy danych, weryfikacja danych

1. Dane, informacja, wiedza
2. Reprezentacja danych – sposób zapisu w pamięci komputera
3. Baza danych – zorganizowany zbiór danych
4. Weryfikacja
   - 4.1 Poprawność i spójność
   - 4.2 Weryfikacja merytoryczna – „sens" danych, znaczenie, kontekst, ...
     - 4.2.1 Anomalie – odstępstwa od „normy"
     - 4.2.2 Wystąpienie znanych problemów – detekcja/klasyfikacja

Bazy danych, weryfikacja danych

1. Dane, informacja, wiedza
2. Reprezentacja danych – sposób zapisu w pamięci komputera
3. Baza danych – zorganizowany zbiór danych
4. Weryfikacja
   - 4.1 Poprawność i spójność
   - 4.2 Weryfikacja merytoryczna – „sens" danych, znaczenie, kontekst, ...
     - 4.2.1 Anomalie – odstępstwa od „normy"
     - 4.2.2 Wystąpienie znanych problemów – detekcja/klasyfikacja

Bazy danych, weryfikacja danych

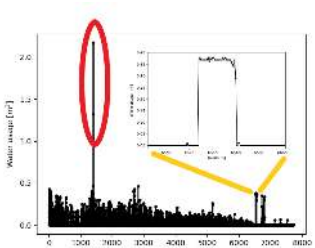
1. Dane, informacja, wiedza
2. Reprezentacja danych – sposób zapisu w pamięci komputera
3. Baza danych – zorganizowany zbiór danych
4. Weryfikacja
   - 4.1 Poprawność i spójność
   - 4.2 Weryfikacja merytoryczna – „sens" danych, znaczenie, kontekst, ...
     - 4.2.1 Anomalie – odstępstwa od „normy"
     - 4.2.2 Wystąpienie znanych problemów – detekcja/klasyfikacja

Bazy danych, weryfikacja danych

1. Dane, informacja, wiedza
2. Reprezentacja danych – sposób zapisu w pamięci komputera
3. Baza danych – zorganizowany zbiór danych
4. Weryfikacja
   - 4.1 Poprawność i spójność
   - 4.2 Weryfikacja merytoryczna – „sens" danych, znaczenie, kontekst, ...
     - 4.2.1 Anomalie – odstępstwa od „normy"
     - 4.2.2 Wystąpienie znanych problemów – detekcja/klasyfikacja

Bazy danych, weryfikacja danych

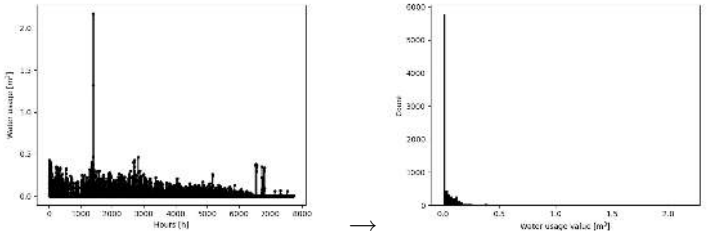
1. Dane, informacja, wiedza
2. Reprezentacja danych – sposób zapisu w pamięci komputera
3. Baza danych – zorganizowany zbiór danych
4. Weryfikacja
   - 4.1 Poprawność i spójność
   - 4.2 Weryfikacja merytoryczna – „sens" danych, znaczenie, kontekst, ...
     - 4.2.1 Anomalie – odstępstwa od „normy"
     - 4.2.2 Wystąpienie znanych problemów – detekcja/klasyfikacja



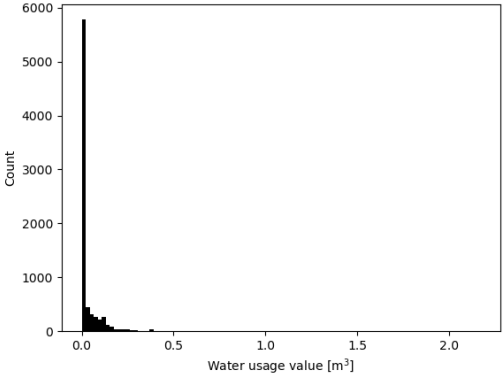
Weryfikacja statystyczna – MAD

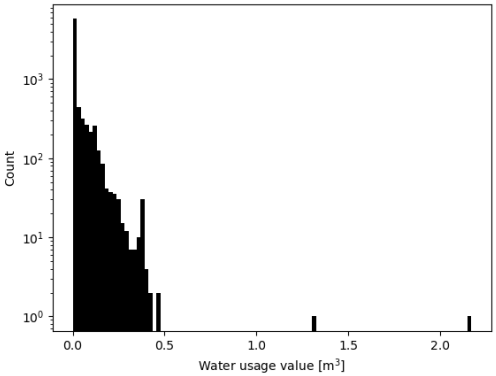
Statystyki opisowe – opis sygnału

1. ~~Szereg czasowy~~ → histogram



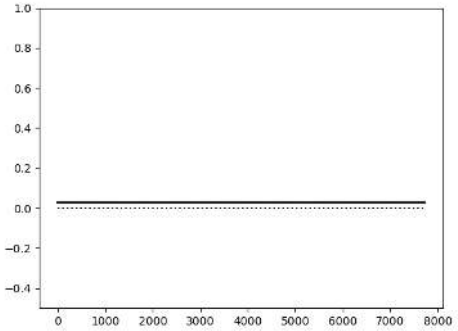
Statystyki opisowe – opis sygnału – histogram





Statystyki opisowe – opis sygnału

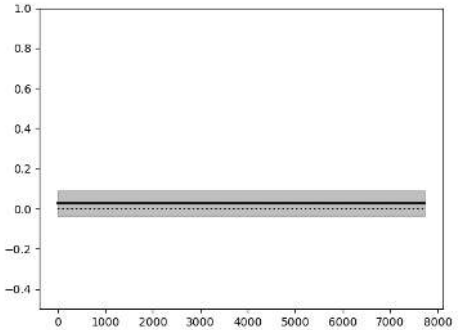
średnia $\mu = 0.028$ $\mu = \frac{1}{n}\sum_i^n x_i$



Statystyki opisowe – opis sygnału

średnia $\mu = 0.028$ $\quad \mu = \frac{1}{n}\sum_i^n x_i$

wariancja $\sigma^2 = 0.004$ $\quad \sigma^2 = \frac{1}{n}\sum_i^n (\mu - x_i)^2$ $\quad \sigma = 0.064$

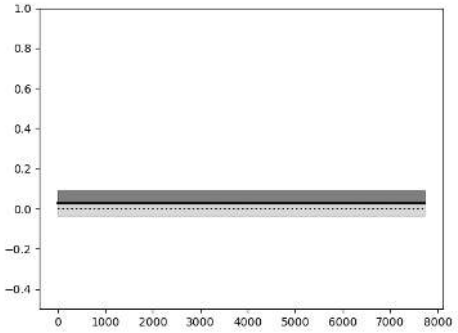


Statystyki opisowe – opis sygnału

średnia $\mu = 0.028$ $\quad \mu = \frac{1}{n}\sum_i^n x_i$

wariancja $\sigma^2 = 0.004$ $\quad \sigma^2 = \frac{1}{n}\sum_i^n (\mu - x_i)^2$ $\quad \sigma = 0.064$

skośność $s = 8.234$



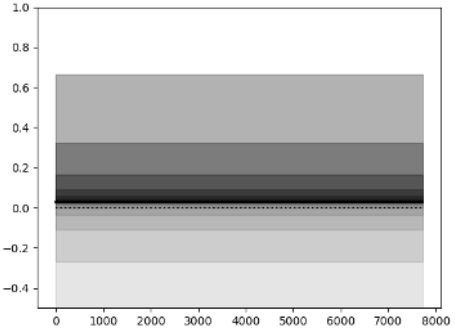
Statystyki opisowe – opis sygnału

średnia $\mu = 0.028$ $\quad \mu = \frac{1}{n}\sum_i^n x_i$

wariancja $\sigma^2 = 0.004$ $\quad \sigma^2 = \frac{1}{n}\sum_i^n (\mu - x_i)^2$ $\quad \sigma = 0.064$

skośność $s = 8.234$

kurtoza $k = 193.664$



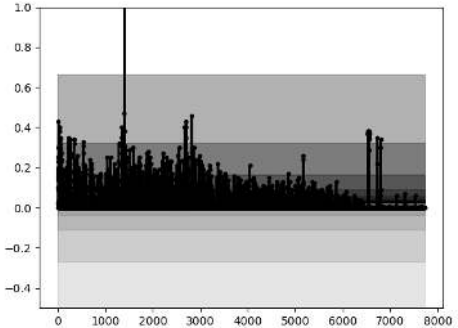
Statystyki opisowe – opis sygnału

średnia $\mu = 0.028$ $\quad \mu = \frac{1}{n}\sum_i^n x_i$

wariancja $\sigma^2 = 0.004$ $\quad \sigma^2 = \frac{1}{n}\sum_i^n (\mu - x_i)^2$ $\quad \sigma = 0.064$

skośność $s = 8.234$

kurtoza $k = 193.664$

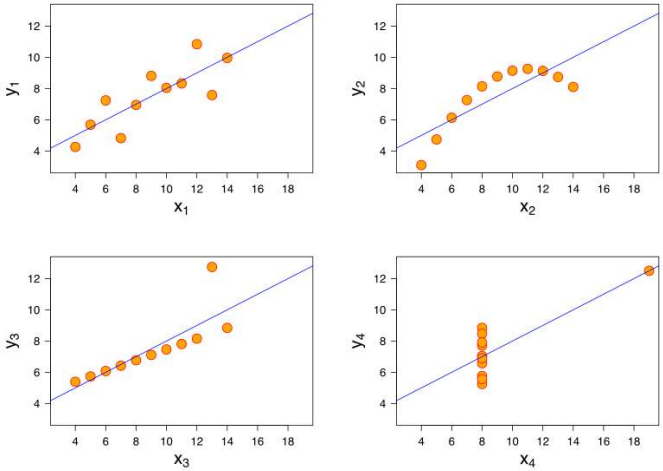


Kwartet Anscombe'a

Danych jest 11 par punktów (x, y) które mają następujące parametry statystyczne:

średnia x	9
wariancja x	11
średnia y	7.5
wariancja y	4.125
korelacja x i y	0.816
prosta regresji	$y = 3 + 0.5x$

Kwartet Anscombe'a



Statystki porządkowe (ang. order statistics)

1. Mediana
2. Kwartyle
3. Percentyle

$$\begin{bmatrix} -2. & 0.5 & 0.71 & 0.6 & 0.7 & -2.1 & 0.59 & 0.51 \end{bmatrix}$$

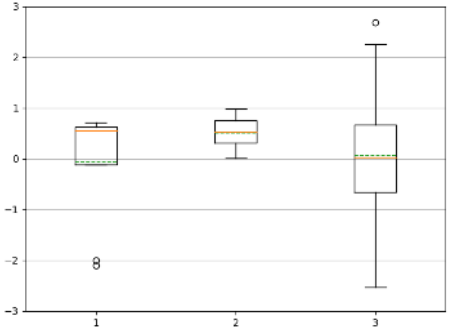
$$\begin{bmatrix} -2.1 & -2. & 0.5 & 0.51 & 0.59 & 0.6 & 0.7 & 0.71 \end{bmatrix}$$

średnia -0.06125

mediana 0.55

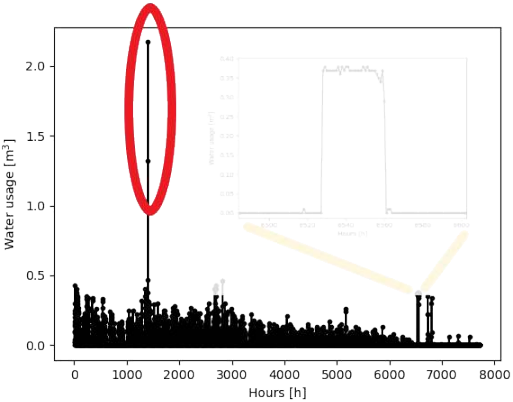
Statystki porządkowe (ang. order statistics) – wykres pudełkowy (ang. boxplot)

1. $\begin{bmatrix} -2. & 0.5 & 0.71 & 0.6 & 0.7 & -2.1 & 0.59 & 0.51 \end{bmatrix}$
2. 100 punktów z rozkładu równomiernego (ang. uniform) $\langle 0, 1 \rangle$
3. 100 punktów ze standardowego rozkładu normalnego (ang. standard normal distribution)



Weryfikacja z wykorzystaniem statystyk

→ Algorytm Median Absolute Deviation (MAD)



Algorytm Median Absolute Deviation (MAD)

1. Dane (przykładowe odczyty) $\mathbf{x} = \left[x_1, x_2, \ldots, x_n\right]$

```
x = [1383.464 1384.9871 1386.6281 1388.142 1389.766 1391.188 1392.796 1394.144 1395.4139 1396.488 1398.209 1500.011 1230.793
1549.475 1399.6281 1238.033 1400.318 1249.4 1400.764 1363.806 1356.909 1287.488 1402.229 1403.588 1405.167 1406.427 ]
```

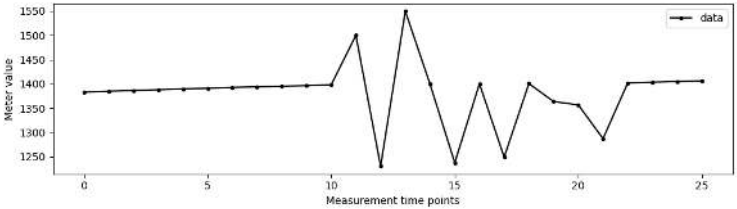
2. $m = \text{median}(\mathbf{x})$
3. $d_i = |x_i - m| \quad \mathbf{d} = \left[d_1, d_2, \ldots, d_n\right]$
4. $T_{\text{MAD}} = \text{median}(\mathbf{d})$
5. $T = 3 \times T_{\text{MAD}}$, $d_i > T \rightarrow$ błędny pomiar

Algorytm Median Absolute Deviation (MAD)

1. Dane (przykładowe odczyty) $\mathbf{x} = \left[x_1, x_2, \ldots, x_n\right]$

```
x = [1383.464 1384.9871 1386.6281 1388.142 1389.766 1391.188 1392.796 1394.144 1395.4139 1396.488 1398.209 1500.011 1230.793
1549.475 1399.6281 1238.033 1400.318 1249.4 1400.764 1363.806 1356.909 1287.488 1402.229 1403.588 1405.167 1406.427 ]
```

2. $m = \text{median}(\mathbf{x})$
3. $d_i = |x_i - m| \quad \mathbf{d} = \left[d_1, d_2, \ldots, d_n\right]$
4. $T_{\text{MAD}} = \text{median}(\mathbf{d})$
5. $T = 3 \times T_{\text{MAD}}$, $d_i > T \rightarrow$ błędny pomiar

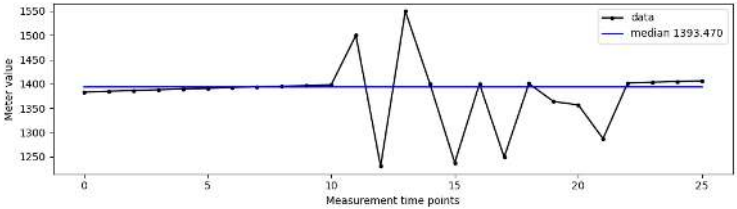


Algorytm Median Absolute Deviation (MAD)

1. Dane (przykładowe odczyty) $\mathbf{x} = \left[x_1, x_2, \ldots, x_n\right]$

```
x = [1383.464 1384.9871 1386.6281 1388.142 1389.766 1391.188 1392.796 1394.144 1395.4139 1396.488 1398.209 1500.011 1230.793
1549.475 1399.6281 1238.033 1400.318 1249.4 1400.764 1363.806 1356.909 1287.488 1402.229 1403.588 1405.167 1406.427 ]
```

2. $m = \text{median}(\mathbf{x})$
3. $d_i = |x_i - m| \quad \mathbf{d} = \left[d_1, d_2, \ldots, d_n\right]$
4. $T_{\text{MAD}} = \text{median}(\mathbf{d})$
5. $T = 3 \times T_{\text{MAD}}$, $d_i > T \rightarrow$ błędny pomiar

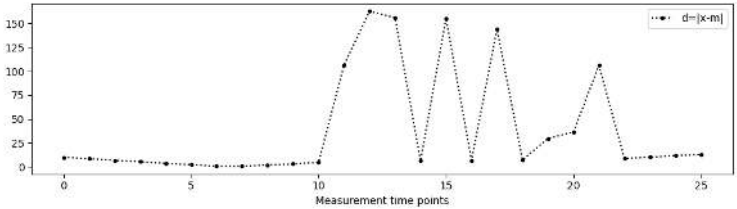


Algorytm Median Absolute Deviation (MAD)

1. Dane (przykładowe odczyty) $\mathbf{x} = \left[x_1, x_2, \ldots, x_n\right]$

```
x = [1383.464 1384.9871 1386.6281 1388.142 1389.766 1391.188 1392.796 1394.144 1395.4139 1396.488 1398.209 1500.011 1230.793
1549.475 1399.6281 1238.033 1400.318 1249.4 1400.764 1363.806 1356.909 1287.488 1402.229 1403.588 1405.167 1406.427 ]
```

2. $m = \text{median}(\mathbf{x})$
3. $d_i = |x_i - m| \quad \mathbf{d} = \left[d_1, d_2, \ldots, d_n\right]$
4. $T_{\text{MAD}} = \text{median}(\mathbf{d})$
5. $T = 3 \times T_{\text{MAD}}$, $d_i > T \rightarrow$ błędny pomiar

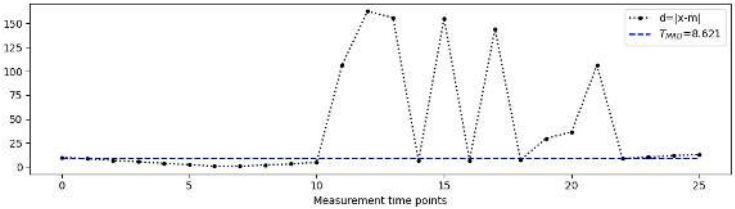


Algorytm Median Absolute Deviation (MAD)

1. Dane (przykładowe odczyty) $\mathbf{x} = \left[x_1, x_2, \ldots, x_n\right]$

```
x = [1383.464 1384.9871 1386.6281 1388.142 1389.766 1391.188 1392.796 1394.144 1395.4139 1396.488 1398.209 1500.011 1230.793
1549.475 1399.6281 1238.033 1400.318 1249.4 1400.764 1363.806 1356.909 1287.488 1402.229 1403.588 1405.167 1406.427 ]
```

2. $m = \text{median}(\mathbf{x})$
3. $d_i = |x_i - m| \quad \mathbf{d} = \left[d_1, d_2, \ldots, d_n\right]$
4. $T_{\text{MAD}} = \text{median}(\mathbf{d})$
5. $T = 3 \times T_{\text{MAD}}$, $d_i > T \rightarrow$ błędny pomiar

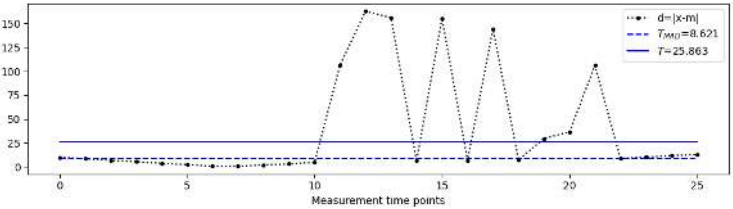


Algorytm Median Absolute Deviation (MAD)

1. Dane (przykładowe odczyty) $\mathbf{x} = \left[x_1, x_2, \ldots, x_n\right]$

```
x = [1383.464 1384.9871 1386.6281 1388.142 1389.766 1391.188 1392.796 1394.144 1395.4139 1396.488 1398.209 1500.011 1230.793
1549.475 1399.6281 1238.033 1400.318 1249.4 1400.764 1363.806 1356.909 1287.488 1402.229 1403.588 1405.167 1406.427 ]
```

2. $m = \text{median}(\mathbf{x})$
3. $d_i = |x_i - m| \quad \mathbf{d} = \left[d_1, d_2, \ldots, d_n\right]$
4. $T_{\text{MAD}} = \text{median}(\mathbf{d})$
5. $T = 3 \times T_{\text{MAD}}$, $d_i > T \rightarrow$ błędny pomiar

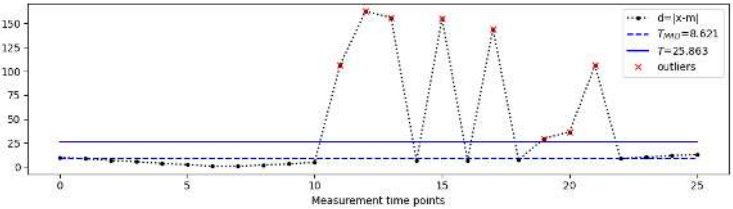


Algorytm Median Absolute Deviation (MAD)

1. Dane (przykładowe odczyty) $\mathbf{x} = \left[x_1, x_2, \ldots, x_n\right]$

```
x = [1383.464 1384.9871 1386.6281 1388.142 1389.766 1391.188 1392.796 1394.144 1395.4139 1396.488 1398.209 1500.011 1230.793
1549.475 1399.6281 1238.033 1400.318 1249.4 1400.764 1363.806 1356.909 1287.488 1402.229 1403.588 1405.167 1406.427 ]
```

2. $m = \text{median}(\mathbf{x})$
3. $d_i = |x_i - m| \quad \mathbf{d} = \left[d_1, d_2, \ldots, d_n\right]$
4. $T_{\text{MAD}} = \text{median}(\mathbf{d})$
5. $T = 3 \times T_{\text{MAD}}$, $d_i > T \rightarrow$ błędny pomiar

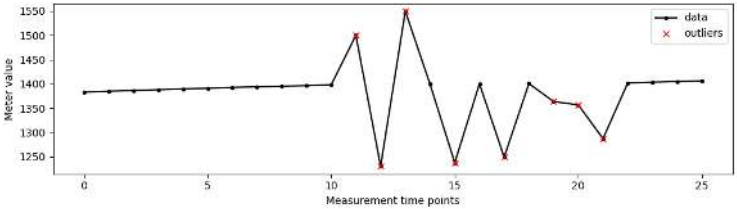


Algorytm Median Absolute Deviation (MAD)

1. Dane (przykładowe odczyty) $\mathbf{x} = \left[x_1, x_2, \ldots, x_n\right]$

```
x = [1383.464 1384.9871 1386.6281 1388.142 1389.766 1391.188 1392.796 1394.144 1395.4139 1396.488 1398.209 1500.011 1230.793
1549.475 1399.6281 1238.033 1400.318 1249.4 1400.764 1363.806 1356.909 1287.488 1402.229 1403.588 1405.167 1406.427 ]
```

2. $m = \text{median}(\mathbf{x})$
3. $d_i = |x_i - m| \quad \mathbf{d} = \left[d_1, d_2, \ldots, d_n\right]$
4. $T_{\text{MAD}} = \text{median}(\mathbf{d})$
5. $T = 3 \times T_{\text{MAD}}$, $d_i > T \rightarrow$ błędny pomiar

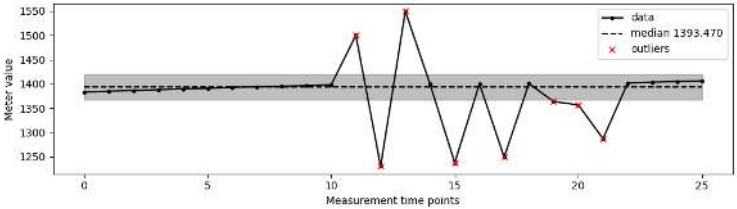


Algorytm Median Absolute Deviation (MAD)

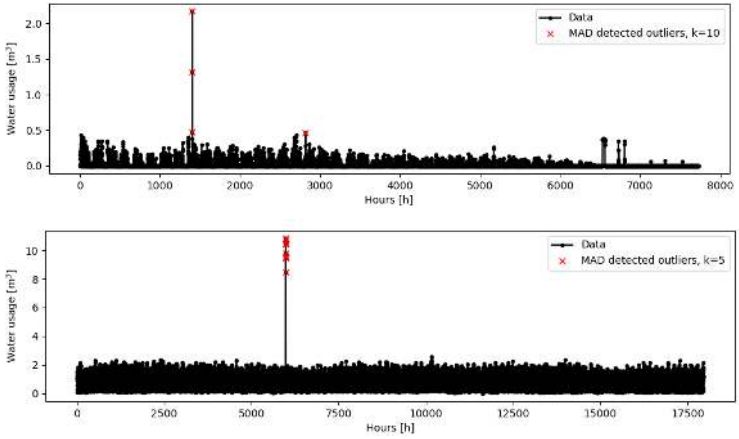
1. Dane (przykładowe odczyty) $\mathbf{x} = \left[x_1, x_2, \ldots, x_n\right]$

```
x = [1383.464 1384.9871 1386.6281 1388.142 1389.766 1391.188 1392.796 1394.144 1395.4139 1396.488 1398.209 1500.011 1230.793
1549.475 1399.6281 1238.033 1400.318 1249.4 1400.764 1363.806 1356.909 1287.488 1402.229 1403.588 1405.167 1406.427 ]
```

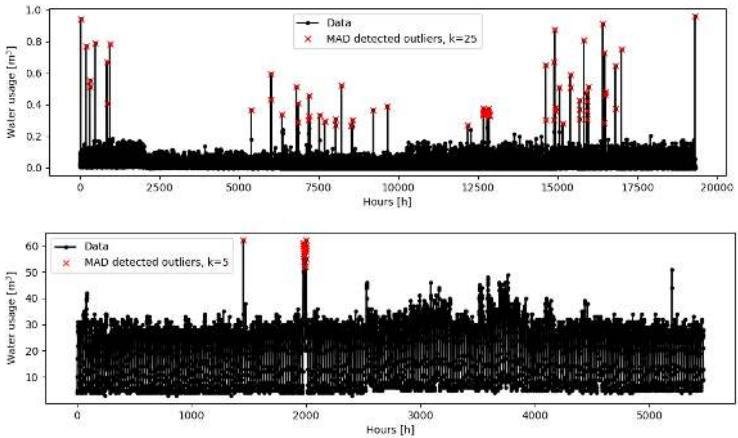
2. $m = \text{median}(\mathbf{x})$
3. $d_i = |x_i - m| \quad \mathbf{d} = \left[d_1, d_2, \ldots, d_n\right]$
4. $T_{\text{MAD}} = \text{median}(\mathbf{d})$
5. $T = 3 \times T_{\text{MAD}}$, $d_i > T \rightarrow$ błędny pomiar



Algorytm MAD – przykłady

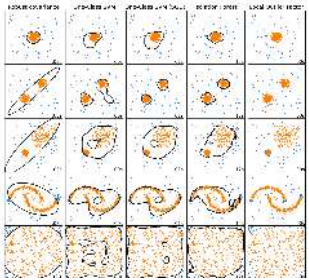


Algorytm MAD – przykłady



Weryfikacja statystyczna – outliery

1. Wykrywanie danych odstających (ang. outliers)
   - 1.1 „odstających” $\approx$ spoza zakresu danych
   - 1.2 MAD (i wiele innych algorytmów)[1]
2. Weryfikacja
   - 2.1 Wyizolowanie pomiarów błędnych – błędy urządzeń
   - 2.2 Wykrycie anomalii i zmian trendu

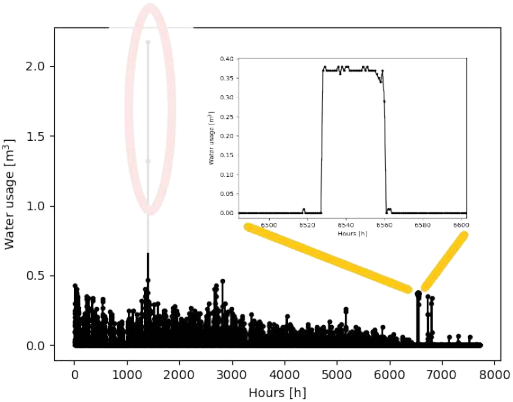


[1] scikit-learn.org/stable/auto_examples/miscellaneous/plot_anomaly_comparison.html, scikit-learn devs, BSD

Weryfikacja statystyczna – RX

Weryfikacja z wykorzystaniem statystyk „strikes back”

→ Algorytm Reed-Xiaoli (RX)



Dlaczego średnia (i inne statystyki np. wariancja) jest dobra?

1. Łatwa do policzenia
2. Znane właściwości
3. Element wnioskowania statystycznego, część bardziej złożonych metod

Dlaczego średnia (i inne statystyki np. wariancja) jest dobra?

1. Łatwa do policzenia
2. Znane właściwości
3. Element wnioskowania statystycznego, część bardziej złożonych metod

Dlaczego średnia (i inne statystyki np. wariancja) jest dobra?

1. Łatwa do policzenia
2. Znane właściwości
3. Element wnioskowania statystycznego, część bardziej złożonych metod

Dlaczego średnia (i inne statystyki np. wariancja) jest dobra?

1. Łatwa do policzenia
2. Znane właściwości
3. Element wnioskowania statystycznego, część bardziej złożonych metod

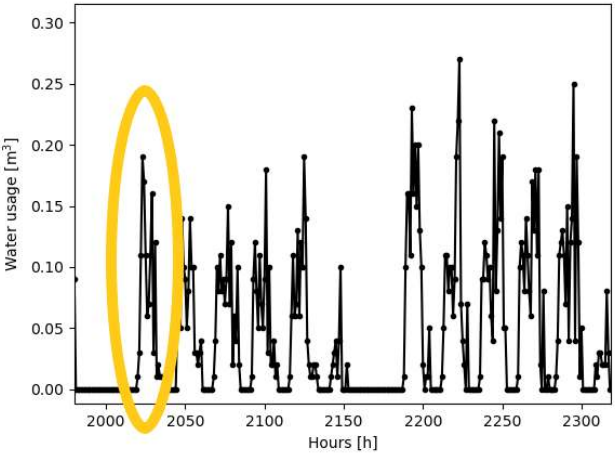
Dlaczego średnia (i inne statystyki np. wariancja) jest dobra?

1. Łatwa do policzenia
2. Znane właściwości
3. Element wnioskowania statystycznego, część bardziej złożonych metod

outlier wartość spoza rozkładu danych, prawdopodobnie błędna - np. błąd urządzenia rejestrującego

anomalia wartość z rozkładu danych, ale rzadka – np. wyciek z instalacji

Wyjście poza pojedyncze punkty danych



Okresy dobowe

1. Wykorzystujemy właściwości sygnału – okresowość dobową

```
[[4.43 4.77 6.1  ... 7.97 4.85 4.29]
[4.22 4.18 5.18 ... 9.1  6.37 5.06]
[4.86 5.9  6.54 ... 8.92 6.54 5.19]
...
[2.72 2.52 2.66 ... 9.17 6.59 4.8 ]
[3.51 2.9  2.52 ... 8.79 5.34 3.59]
[2.62 2.72 2.57 ... 9.71 6.01 3.41]]
```

   $X = \left[x_{ij}\right]$ i dni j godziny, $\mathbf{x}_i$ i-ta doba

2. Anomalia → „nietypowe zachowanie"
3. „Zachowanie" → średnia dobowa $\bar{\mathbf{x}} = \left[\bar{x}_j\right]$ $\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$
4. „Nietypowe" → duża odległość od średniej $\|\bar{\mathbf{x}} - \mathbf{x}_i\| = \sqrt{\sum_{j=1}^{24}(\bar{x}_j - x_{ij})^2}$

Okresy dobowe

1. Wykorzystujemy właściwości sygnału – okresowość dobową

```
[[4.43 4.77 6.1  ... 7.97 4.85 4.29]
[4.22 4.18 5.18 ... 9.1  6.37 5.06]
[4.86 5.9  6.54 ... 8.92 6.54 5.19]
...
[2.72 2.52 2.66 ... 9.17 6.59 4.8 ]
[3.51 2.9  2.52 ... 8.79 5.34 3.59]
[2.62 2.72 2.57 ... 9.71 6.01 3.41]]
```

   $X = \left[x_{ij}\right]$ i dni j godziny, $\mathbf{x}_i$ i-ta doba
2. Anomalia $\rightarrow$ „nietypowe zachowanie”
3. „Zachowanie” $\rightarrow$ średnia dobowa $\bar{\mathbf{x}} = \left[\bar{x}_j\right]$ $\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$
4. „Nietypowe” $\rightarrow$ duża odległość od średniej $\|\bar{\mathbf{x}} - \mathbf{x}_i\| = \sqrt{\sum_{j=1}^{24}(\bar{x}_j - x_{ij})^2}$

Okresy dobowe

1. Wykorzystujemy właściwości sygnału – okresowość dobową

```
[[4.43 4.77 6.1  ... 7.97 4.85 4.29]
[4.22 4.18 5.18 ... 9.1  6.37 5.06]
[4.86 5.9  6.54 ... 8.92 6.54 5.19]
...
[2.72 2.52 2.66 ... 9.17 6.59 4.8 ]
[3.51 2.9  2.52 ... 8.79 5.34 3.59]
[2.62 2.72 2.57 ... 9.71 6.01 3.41]]
```

   $X = \left[x_{ij}\right]$ i dni j godziny, $\mathbf{x}_i$ i-ta doba
2. Anomalia $\rightarrow$ „nietypowe zachowanie”
3. „Zachowanie” $\rightarrow$ średnia dobowa $\bar{\mathbf{x}} = \left[\bar{x}_j\right]$ $\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$
4. „Nietypowe” $\rightarrow$ duża odległość od średniej $\|\bar{\mathbf{x}} - \mathbf{x}_i\| = \sqrt{\sum_{j=1}^{24}(\bar{x}_j - x_{ij})^2}$

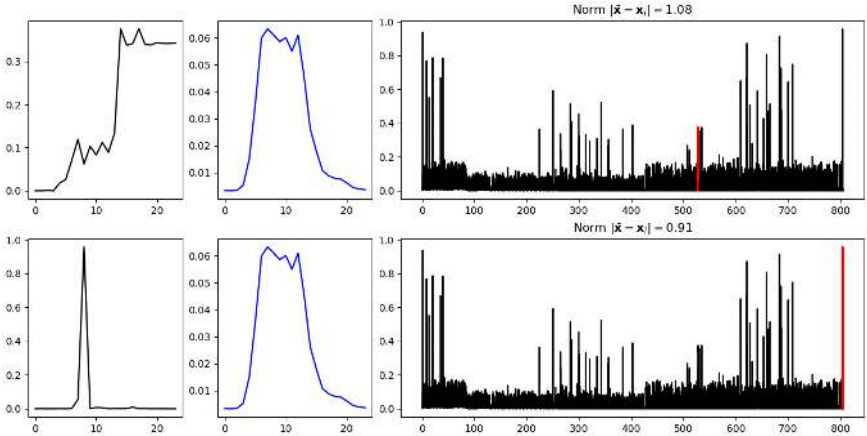
Okresy dobowe

1. Wykorzystujemy właściwości sygnału – okresowość dobową

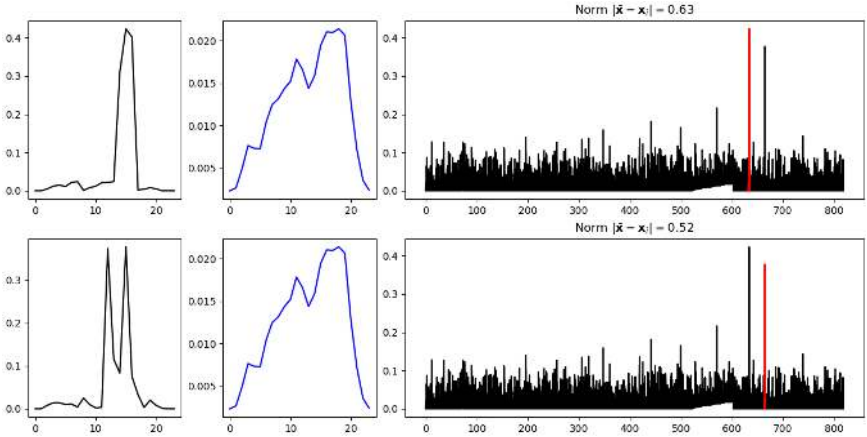
```
[[4.43 4.77 6.1  ... 7.97 4.85 4.29]
[4.22 4.18 5.18 ... 9.1  6.37 5.06]
[4.86 5.9  6.54 ... 8.92 6.54 5.19]
...
[2.72 2.52 2.66 ... 9.17 6.59 4.8 ]
[3.51 2.9  2.52 ... 8.79 5.34 3.59]
[2.62 2.72 2.57 ... 9.71 6.01 3.41]]
```

   $X = \left[x_{ij}\right]$ i dni j godziny, $\mathbf{x}_i$ i-ta doba
2. Anomalia $\rightarrow$ „nietypowe zachowanie"
3. „Zachowanie" $\rightarrow$ średnia dobowa $\bar{\mathbf{x}} = \left[\bar{x}_j\right]$ $\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$
4. „Nietypowe" $\rightarrow$ duża odległość od średniej $\|\bar{\mathbf{x}} - \mathbf{x}_i\| = \sqrt{\sum_{j=1}^{24}(\bar{x}_j - x_{ij})^2}$

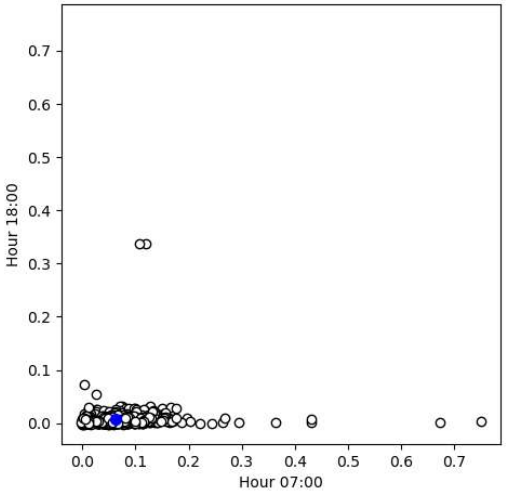
Problem?


Norm $|\bar{\mathbf{x}} - \mathbf{x}_i| = 1.08$
Norm $|\bar{\mathbf{x}} - \mathbf{x}_i| = 0.91$

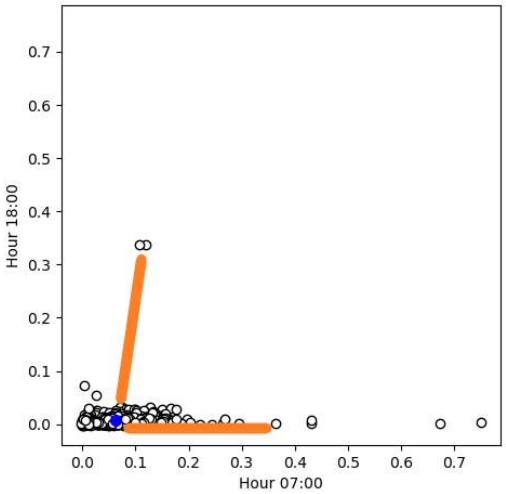
Problem?


Norm $|\tilde{\mathbf{x}} - \mathbf{x}_i| = 0.63$
Norm $|\tilde{\mathbf{x}} - \mathbf{x}_i| = 0.52$

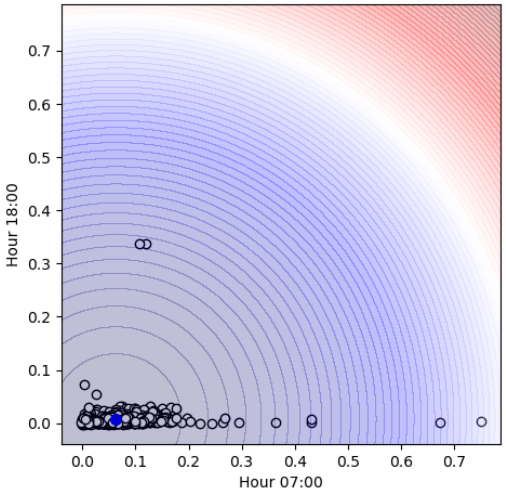
Odległość od średniej a rozkład danych



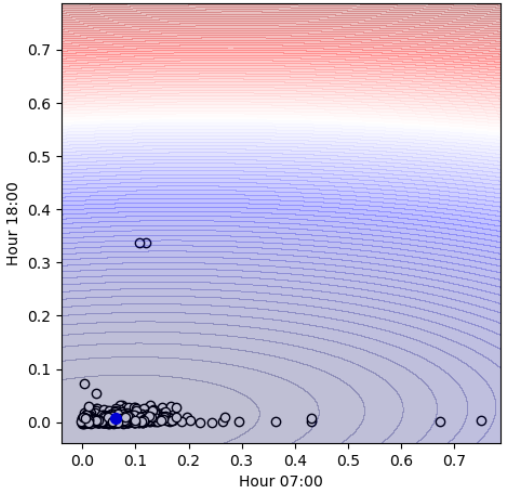
Odległość od średniej a rozkład danych



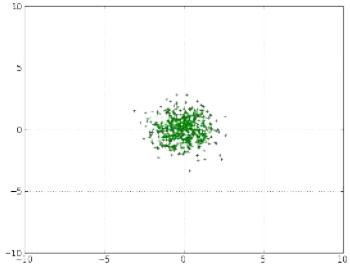
Odległość od średniej a rozkład danych



Odległość od średniej a rozkład danych



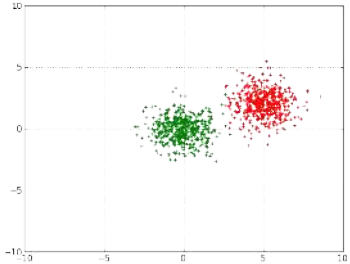
Opis rozkładu – macierz kowariancji



Parametry rozkładu normalnego $\mathcal{N}(\mu, \Sigma)$

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

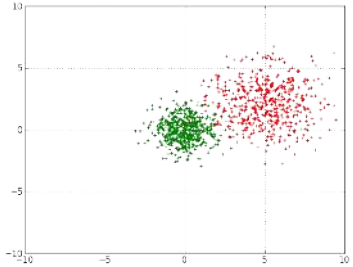
Opis rozkładu – macierz kowariancji



Parametry rozkładu normalnego $\mathcal{N}(\mu, \Sigma)$

$$\mu = \begin{bmatrix} 5 \\ 2 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

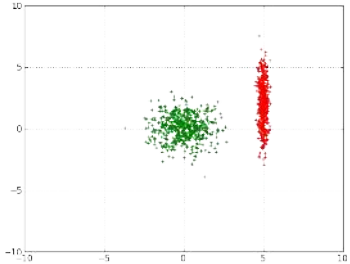
Opis rozkładu – macierz kowariancji



Parametry rozkładu normalnego $\mathcal{N}(\mu, \Sigma)$

$$\mu = \begin{bmatrix} 5 \\ 2 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$$

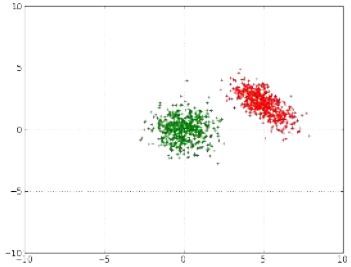
Opis rozkładu – macierz kowariancji



Parametry rozkładu normalnego $\mathcal{N}(\mu, \Sigma)$

$$\mu = \begin{bmatrix} 5 \\ 2 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 0.03 & 0 \\ 0 & 3 \end{bmatrix}$$

Opis rozkładu – macierz kowariancji



Parametry rozkładu normalnego $\mathcal{N}(\mu, \Sigma)$

$$\mu = \begin{bmatrix} 5 \\ 2 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 1 & -0.69 \\ -0.69 & 1 \end{bmatrix}$$

Odległość Mahalanobisa

$$d_E(\bar{\mathbf{x}}, \mathbf{x}_i) = \|\bar{\mathbf{x}} - \mathbf{x}_i\| = \sqrt{\sum_{j=1}^{24} (\bar{x}_j - x_{ij})^2} =$$

Odległość Mahalanobisa

$$d_E(\bar{\mathbf{x}}, \mathbf{x}_i) = \|\bar{\mathbf{x}} - \mathbf{x}_i\| = \sqrt{\sum_{j=1}^{24} (\bar{x}_j - x_{ij})^2} = \sqrt{(\bar{\mathbf{x}} - \mathbf{x}_i)(\bar{\mathbf{x}} - \mathbf{x}_i)^\top} =$$

Odległość Mahalanobisa

$$d_E(\bar{\mathbf{x}}, \mathbf{x}_i) = \|\bar{\mathbf{x}} - \mathbf{x}_i\| = \sqrt{\sum_{j=1}^{24} (\bar{x}_j - x_{ij})^2} = \sqrt{(\bar{\mathbf{x}} - \mathbf{x}_i)(\bar{\mathbf{x}} - \mathbf{x}_i)^\top} = \sqrt{(\bar{\mathbf{x}} - \mathbf{x}_i) I (\bar{\mathbf{x}} - \mathbf{x}_i)^\top}$$

I – macierz identyczności

Odległość Mahalanobisa

$d_E(\bar{\mathbf{x}}, \mathbf{x}_i) = \|\bar{\mathbf{x}} - \mathbf{x}_i\| = \sqrt{\sum_{j=1}^{24}(\bar{x}_j - x_{ij})^2} = \sqrt{(\bar{\mathbf{x}} - \mathbf{x}_i)(\bar{\mathbf{x}} - \mathbf{x}_i)^\top} = \sqrt{(\bar{\mathbf{x}} - \mathbf{x}_i)I(\bar{\mathbf{x}} - \mathbf{x}_i)^\top}$

I – macierz identyczności

$C = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})^\top(\mathbf{x}_i - \bar{\mathbf{x}})$ – macierz kowariancji danych (ang. sample covariance)

Odległość Mahalanobisa

$$d_E(\bar{\mathbf{x}}, \mathbf{x}_i) = \|\bar{\mathbf{x}} - \mathbf{x}_i\| = \sqrt{\sum_{j=1}^{24} (\bar{x}_j - x_{ij})^2} = \sqrt{(\bar{\mathbf{x}} - \mathbf{x}_i)(\bar{\mathbf{x}} - \mathbf{x}_i)^\top} = \sqrt{(\bar{\mathbf{x}} - \mathbf{x}_i) I (\bar{\mathbf{x}} - \mathbf{x}_i)^\top}$$

I – macierz identyczności

$C = \frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}})^\top (\mathbf{x}_i - \bar{\mathbf{x}})$ – macierz kowariancji danych (ang. sample covariance)

C^{-1} – odwrotność macierzy kowariancji

Odległość Mahalanobisa

$d_E(\bar{\mathbf{x}}, \mathbf{x}_i) = \|\bar{\mathbf{x}} - \mathbf{x}_i\| = \sqrt{\sum_{j=1}^{24}(\bar{x}_j - x_{ij})^2} = \sqrt{(\bar{\mathbf{x}} - \mathbf{x}_i)(\bar{\mathbf{x}} - \mathbf{x}_i)^\top} = \sqrt{(\bar{\mathbf{x}} - \mathbf{x}_i)I(\bar{\mathbf{x}} - \mathbf{x}_i)^\top}$

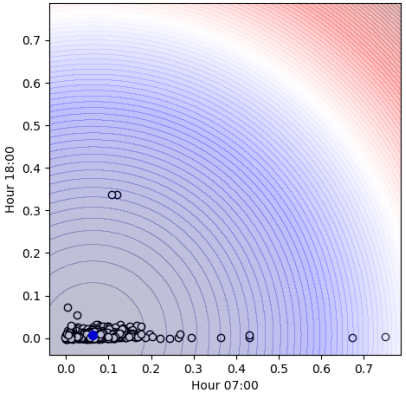
I – macierz identyczności

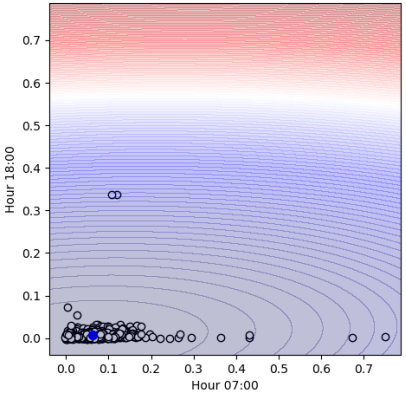
$C = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})^\top(\mathbf{x}_i - \bar{\mathbf{x}})$ – macierz kowariancji danych (ang. sample covariance)

C^{-1} – odwrotność macierzy kowariancji

$d_M(\bar{\mathbf{x}}, \mathbf{x}_i) = \sqrt{(\bar{\mathbf{x}} - \mathbf{x}_i)C^{-1}(\bar{\mathbf{x}} - \mathbf{x}_i)^\top}$

Odległość Mahalanobisa





Wykrywanie anomalii – odległość Mahalanobisa

1. Wykorzystujemy właściwości sygnału – okresowość dobową

```
[[4.43 4.77 6.1 ... 7.97 4.85 4.29]
[4.22 4.18 5.18 ... 9.1  6.37 5.06]
[4.86 5.9  6.54 ... 8.92 6.54 5.19]
...
[2.72 2.52 2.66 ... 9.17 6.59 4.8 ]
[3.51 2.9  2.52 ... 8.79 5.34 3.59]
[2.62 2.72 2.57 ... 9.71 6.01 3.41]]
```

   $X = \left[x_{ij}\right]$ i dni j godziny, $\mathbf{x}_i$ i-ta doba
2. Anomalia $\rightarrow$ „nietypowe zachowanie"
3. „Zachowanie" $\rightarrow$ średnia dobowa $\bar{\mathbf{x}} = \left[\bar{x}_j\right]$ $\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$ i macierz kowariancji
4. „Nietypowe" $\rightarrow$ duża odległość od średniej liczona odległością Mahalanobisa $d_M(\bar{\mathbf{x}}, \mathbf{x}_i)$

Wykrywanie anomalii – odległość Mahalanobisa → algorytm RX

1. Mamy dane $X \rightarrow$ średnia $\bar{\mathbf{x}}$, macierz kowariancji C
2. $\mathbf{x}_i$ jest anomalią jeżeli $d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T$
3. Chcielibyśmy wyrazić próg w postaci prawdopodobieństwa, np. anomalia jest jeżeli $\mathbf{x}_i$ ma prawdopodobieństwo np. $\alpha = 1\%$, albo 0.1%
4. Wiemy, że jeżeli punkty w X mają rozkład normalny, to $d_M(\bar{\mathbf{x}}, \mathbf{x}_i)$ ma rozkład χ^2 z d-stopniami swobody (w naszym wypadku $d = 24$)
5. Znając rozkład – w tym wypadku χ^2 – możemy wyznaczyć T, dla którego prawdopodobieństwo przekroczenia wartości jest α, $P(d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T) = \alpha$

Wykrywanie anomalii – odległość Mahalanobisa → algorytm RX

1. Mamy dane $X \rightarrow$ średnia $\bar{\mathbf{x}}$, macierz kowariancji C
2. $\mathbf{x}_i$ jest anomalią jeżeli $d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T$
3. Chcielibyśmy wyrazić próg w postaci prawdopodobieństwa, np. anomalia jest jeżeli $\mathbf{x}_i$ ma prawdopodobieństwo np. $\alpha = 1\%$, albo 0.1%
4. Wiemy, że jeżeli punkty w X mają rozkład normalny, to $d_M(\bar{\mathbf{x}}, \mathbf{x}_i)$ ma rozkład χ^2 z d-stopniami swobody (w naszym wypadku $d = 24$)
5. Znając rozkład – w tym wypadku χ^2 – możemy wyznaczyć T, dla którego prawdopodobieństwo przekroczenia wartości jest α, $P(d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T) = \alpha$

Wykrywanie anomalii – odległość Mahalanobisa → algorytm RX

1. Mamy dane $X \rightarrow$ średnia $\bar{\mathbf{x}}$, macierz kowariancji C
2. $\mathbf{x}_i$ jest anomalią jeżeli $d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T$ ... ale jak wyliczyć T?
3. Chcielibyśmy wyrazić próg w postaci prawdopodobieństwa, np. anomalia jest jeżeli $\mathbf{x}_i$ ma prawdopodobieństwo np. $\alpha = 1\%$, albo 0.1%
4. Wiemy, że jeżeli punkty w X mają rozkład normalny, to $d_M(\bar{\mathbf{x}}, \mathbf{x}_i)$ ma rozkład χ^2 z d-stopniami swobody (w naszym wypadku $d = 24$)
5. Znając rozkład – w tym wypadku χ^2 – możemy wyznaczyć T, dla którego prawdopodobieństwo przekroczenia wartości jest α, $P(d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T) = \alpha$

Wykrywanie anomalii – odległość Mahalanobisa → algorytm RX

1. Mamy dane $X \rightarrow$ średnia $\bar{\mathbf{x}}$, macierz kowariancji C
2. $\mathbf{x}_i$ jest anomalią jeżeli $d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T$ ... ale jak wyliczyć T?
   ... dla różnych wartości średniej i m. kowariancji różne T :(
3. Chcielibyśmy wyrazić próg w postaci prawdopodobieństwa, np. anomalia jest jeżeli $\mathbf{x}_i$ ma prawdopodobieństwo np. $\alpha = 1\%$, albo 0.1%
4. Wiemy, że jeżeli punkty w X mają rozkład normalny, to $d_M(\bar{\mathbf{x}}, \mathbf{x}_i)$ ma rozkład χ^2 z d-stopniami swobody (w naszym wypadku $d = 24$)
5. Znając rozkład – w tym wypadku χ^2 – możemy wyznaczyć T, dla którego prawdopodobieństwo przekroczenia wartości jest α, $P(d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T) = \alpha$

Wykrywanie anomalii – odległość Mahalanobisa → algorytm RX

1. Mamy dane $X \rightarrow$ średnia $\bar{\mathbf{x}}$, macierz kowariancji C
2. $\mathbf{x}_i$ jest anomalią jeżeli $d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T$ ... ale jak wyliczyć T?
   ... dla różnych wartości średniej i m. kowariancji różne T :(
3. Chcielibyśmy wyrazić próg w postaci prawdopodobieństwa, np. anomalia jest jeżeli $\mathbf{x}_i$ ma prawdopodobieństwo np. $\alpha = 1\%$, albo 0.1%
4. Wiemy, że jeżeli punkty w X mają rozkład normalny, to $d_M(\bar{\mathbf{x}}, \mathbf{x}_i)$ ma rozkład χ^2 z d-stopniami swobody (w naszym wypadku $d = 24$)
5. Znając rozkład – w tym wypadku χ^2 – możemy wyznaczyć T, dla którego prawdopodobieństwo przekroczenia wartości jest α, $P(d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T) = \alpha$

Wykrywanie anomalii – odległość Mahalanobisa → algorytm RX

1. Mamy dane $X \rightarrow$ średnia $\bar{\mathbf{x}}$, macierz kowariancji C
2. $\mathbf{x}_i$ jest anomalią jeżeli $d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T$ ... ale jak wyliczyć T?
   ... dla różnych wartości średniej i m. kowariancji różne T :(
3. Chcielibyśmy wyrazić próg w postaci prawdopodobieństwa, np. anomalia jest jeżeli $\mathbf{x}_i$ ma prawdopodobieństwo np. $\alpha = 1\%$, albo 0.1% ... a dla α wyznaczyć T automatycznie
4. Wiemy, że jeżeli punkty w X mają rozkład normalny, to $d_M(\bar{\mathbf{x}}, \mathbf{x}_i)$ ma rozkład χ^2 z d-stopniami swobody (w naszym wypadku $d = 24$)
5. Znając rozkład – w tym wypadku χ^2 – możemy wyznaczyć T, dla którego prawdopodobieństwo przekroczenia wartości jest α, $P(d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T) = \alpha$

Wykrywanie anomalii – odległość Mahalanobisa → algorytm RX

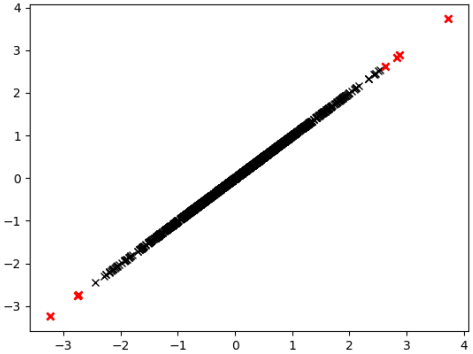
1. Mamy dane $X \rightarrow$ średnia $\bar{\mathbf{x}}$, macierz kowariancji C
2. $\mathbf{x}_i$ jest anomalią jeżeli $d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T$ ... ale jak wyliczyć T?
   ... dla różnych wartości średniej i m. kowariancji różne T :(
3. Chcielibyśmy wyrazić próg w postaci prawdopodobieństwa, np. anomalia jest jeżeli $\mathbf{x}_i$ ma prawdopodobieństwo np. $\alpha = 1\%$, albo 0.1% ... a dla α wyznaczyć T automatycznie
4. Wiemy, że jeżeli punkty w X mają rozkład normalny, to $d_M(\bar{\mathbf{x}}, \mathbf{x}_i)$ ma rozkład χ^2 z d-stopniami swobody (w naszym wypadku $d = 24$)
5. Znając rozkład – w tym wypadku χ^2 – możemy wyznaczyć T, dla którego prawdopodobieństwo przekroczenia wartości jest α, $P(d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T) = \alpha$

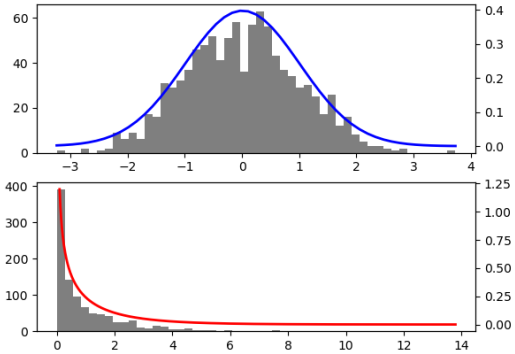
Wykrywanie anomalii – odległość Mahalanobisa → algorytm RX

1. Mamy dane $X \rightarrow$ średnia $\bar{\mathbf{x}}$, macierz kowariancji C
2. $\mathbf{x}_i$ jest anomalią jeżeli $d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T$ ... ale jak wyliczyć T?
   ... dla różnych wartości średniej i m. kowariancji różne T :(
3. Chcielibyśmy wyrazić próg w postaci prawdopodobieństwa, np. anomalia jest jeżeli $\mathbf{x}_i$ ma prawdopodobieństwo np. $\alpha = 1\%$, albo 0.1% ... a dla α wyznaczyć T automatycznie
4. Wiemy, że jeżeli punkty w X mają rozkład normalny, to $d_M(\bar{\mathbf{x}}, \mathbf{x}_i)$ ma rozkład χ^2 z d-stopniami swobody (w naszym wypadku $d = 24$)
5. Znając rozkład – w tym wypadku χ^2 – możemy wyznaczyć T, dla którego prawdopodobieństwo przekroczenia wartości jest α, $P(d_M(\bar{\mathbf{x}}, \mathbf{x}_i) > T) = \alpha$

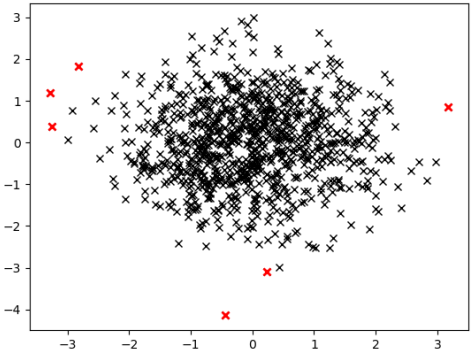
Algorytm RX – przykłady

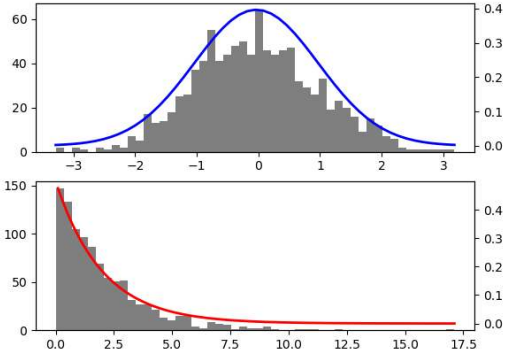
Dane sztucznie wygenerowane, rozkład normalny, $d = 1$



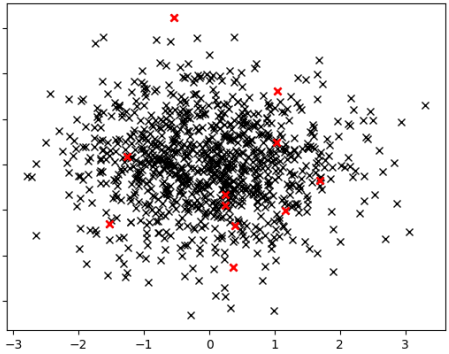


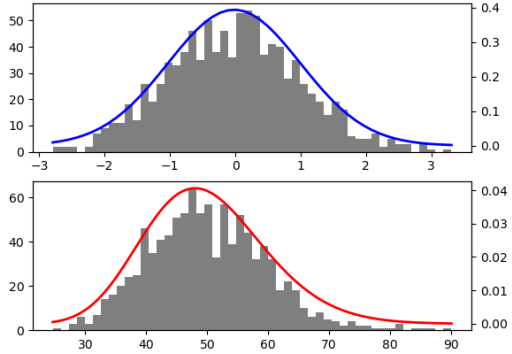
Dane sztucznie wygenerowane, rozkład normalny, $d = 2$



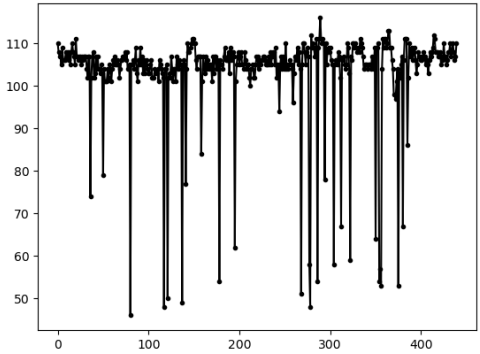


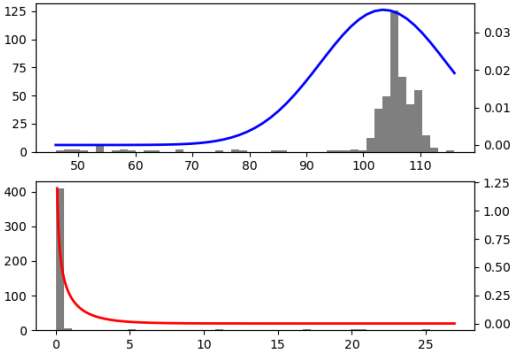
Dane sztucznie wygenerowane, rozkład normalny, $d = 50$



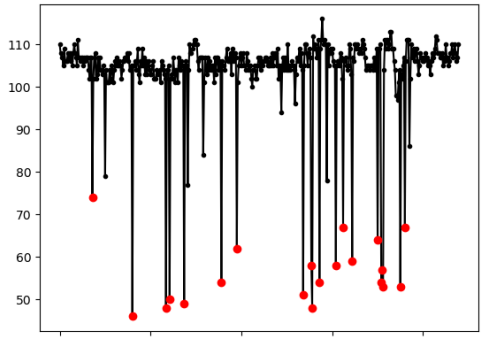


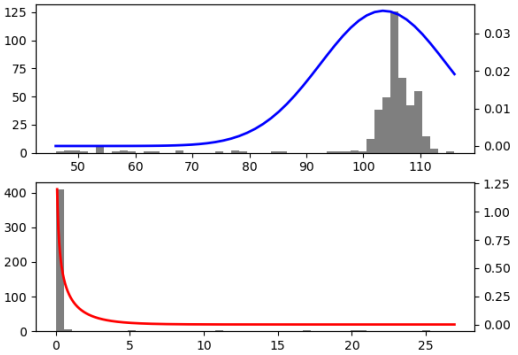
Dane rzeczywiste (monitoring przemysłowy), rozkład – ?, $d = 1$



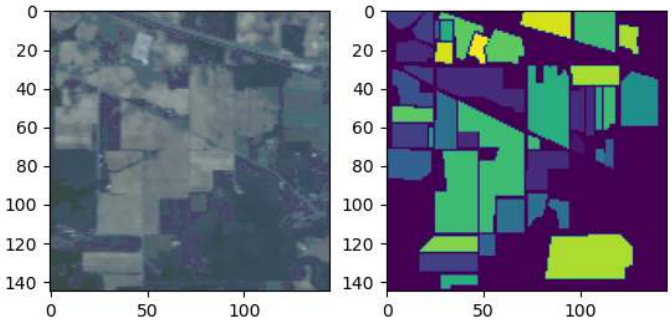


Dane rzeczywiste (monitoring przemysłowy), rozkład – ?, $d = 1$

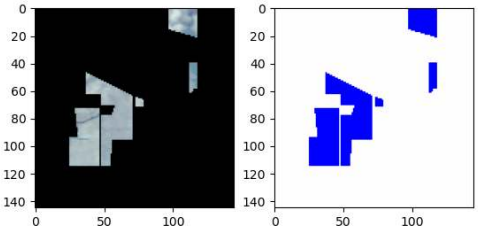


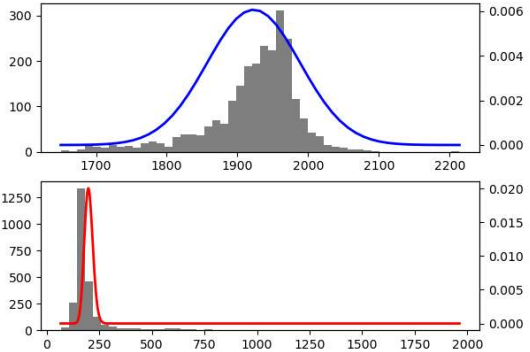


Dane rzeczywiste (zdjęcie hiperspektralne), rozkład – ?, $d = 100$

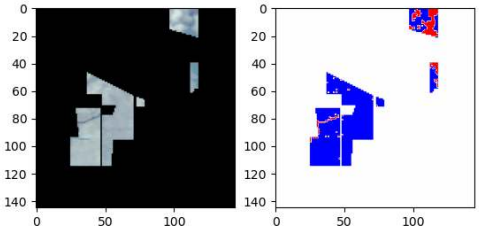


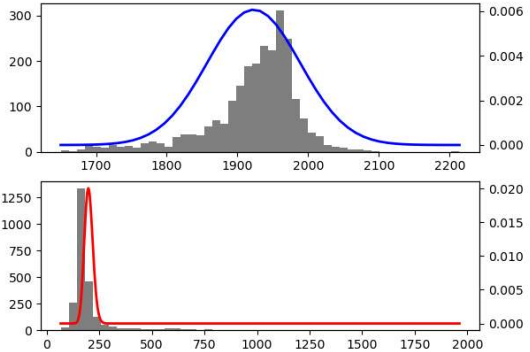
Dane rzeczywiste (zdjęcie hiperspektralne), rozkład – ?, $d = 100$





Dane rzeczywiste (zdjęcie hiperspektralne), rozkład – ?, $d = 100$





Wykrywanie anomalii, algorytm RX, podsumowanie

1. Anomalie a outliery
2. Elementy (RX):
   - 2.1 Model – rozkład normalny, opisany średnią i macierzą kowariancji (parametry)
   - 2.2 Algorytm – odległość Mahalanobisa, wykorzystanie znanego rozkładu χ^2 (parametry wejściowe, wyjście)
   - 2.3 Teoria – prawdopodobieństwo i statystyka
3. Rozwinięcia
   - 3.1 Mixture of Gaussians (MoG)
   - 3.2 Principal Component Analysis (PCA)
   - 3.3 Hidden Markov Models (HMM)
4. Alternatywy – np. sieci neuronowe

Wykrywanie anomalii, algorytm RX, podsumowanie

1. Anomalie a outliery
2. Elementy (RX):
   - 2.1 Model – rozkład normalny, opisany średnią i macierzą kowariancji (parametry)
   - 2.2 Algorytm – odległość Mahalanobisa, wykorzystanie znanego rozkładu χ^2 (parametry wejściowe, wyjście)
   - 2.3 Teoria – prawdopodobieństwo i statystyka
3. Rozwinięcia
   - 3.1 Mixture of Gaussians (MoG)
   - 3.2 Principal Component Analysis (PCA)
   - 3.3 Hidden Markov Models (HMM)
4. Alternatywy – np. sieci neuronowe

Wykrywanie anomalii, algorytm RX, podsumowanie

1. Anomalie a outliery
2. Elementy (RX):
   - 2.1 Model – rozkład normalny, opisany średnią i macierzą kowariancji (parametry)
   - 2.2 Algorytm – odległość Mahalanobisa, wykorzystanie znanego rozkładu χ^2 (parametry wejściowe, wyjście)
   - 2.3 Teoria – prawdopodobieństwo i statystyka
3. Rozwinięcia
   - 3.1 Mixture of Gaussians (MoG)
   - 3.2 Principal Component Analysis (PCA)
   - 3.3 Hidden Markov Models (HMM)
4. Alternatywy – np. sieci neuronowe

Wykrywanie anomalii, algorytm RX, podsumowanie

1. Anomalie a outliery
2. Elementy (RX):
   - 2.1 Model – rozkład normalny, opisany średnią i macierzą kowariancji (parametry)
   - 2.2 Algorytm – odległość Mahalanobisa, wykorzystanie znanego rozkładu χ^2 (parametry wejściowe, wyjście)
   - 2.3 Teoria – prawdopodobieństwo i statystyka
3. Rozwinięcia
   - 3.1 Mixture of Gaussians (MoG)
   - 3.2 Principal Component Analysis (PCA)
   - 3.3 Hidden Markov Models (HMM)
4. Alternatywy – np. sieci neuronowe

Statystyczne metody uczenia maszynowego – redukcja wymiarowości i klasyfikacja

Przemysław Głomb

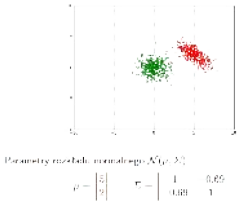
Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk

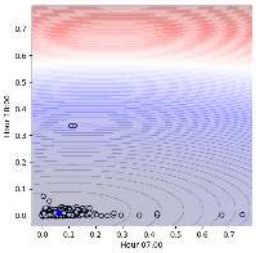
redaktor pomocniczy: Bartłomiej Gardas

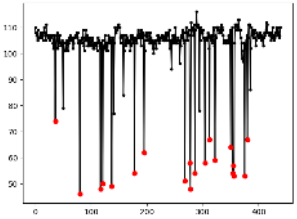
Metody liniowe

Analiza składowych głównych

Macierz kowariancji, odległość Mahalanobisa, RX, ...







Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów
10 cech	predykcja właściwości rozbłysków słonecznych
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych
3+ mln cech	reputacja adresów URL

Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)[1]
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów
10 cech	predykcja właściwości rozbłysków słonecznych
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych
3+ mln cech	reputacja adresów URL

[1] Grisoni, F., Neuhaus, C.S., Hishinuma, M., Gabernet, G., Hiss, J.A., Kotera, M. and Schneider, G., 2019. De novo design of anticancer peptides by ensemble artificial neural networks. Journal of Molecular Modeling, 25(5), 112

Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów[1]
10 cech	predykcja właściwości rozbłysków słonecznych
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych
3+ mln cech	reputacja adresów URL

[1] Scalabrini Sampaio, G., Vallim Filho, A. R. d. A., Santos da Silva, L., & Augusto da Silva, L. (2019). Prediction of Motor Failure Time Using An Artificial Neural Network. Sensors, 19(19), 4342

Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów
10 cech	predykcja właściwości rozbłysków słonecznych[1]
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych
3+ mln cech	reputacja adresów URL

[1] https://archive.ics.uci.edu/dataset/89/solar+flare

Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów
10 cech	predykcja właściwości rozbłysków słonecznych
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych[1]
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych
3+ mln cech	reputacja adresów URL

[1] https://archive.ics.uci.edu/dataset/336/chronic+kidney+disease

Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów
10 cech	predykcja właściwości rozbłysków słonecznych
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych[1]
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych
3+ mln cech	reputacja adresów URL

[1] Redmond, Michael and Alok Baveja. "A data-driven software tool for enabling cooperative information sharing among police departments." Eur. J. Oper. Res. 141 (2002): 660-678.

Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów
10 cech	predykcja właściwości rozbłysków słonecznych
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny[1]
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych
3+ mln cech	reputacja adresów URL

[1] Gul, S., Rahim, F., Isin, S. et al. Structure-based design and classifications of small molecules regulating the circadian rhythm period. Sci Rep 11, 18510 (2021)

Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów
10 cech	predykcja właściwości rozbłysków słonecznych
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych
3+ mln cech	reputacja adresów URL

Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów
10 cech	predykcja właściwości rozbłysków słonecznych
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych[1]
3+ mln cech	reputacja adresów URL

[1] https://archive.ics.uci.edu/dataset/309/gas+sensor+array+exposed+to+turbulent+gas+mixtures

Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów
10 cech	predykcja właściwości rozbłysków słonecznych
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych
3+ mln cech	reputacja adresów URL[1]

[1] https://archive.ics.uci.edu/dataset/187/url+reputation

Zbiory danych są wielowymiarowe

1 cecha	przykład z wykładu o perceptronie - dojrzałość owocu
2 cechy	właściwości przeciwrakowe peptydów (sekwencja dna, aktywność)
5 cech	predykcja awarii silników wentylatora na podstawie danych z akcelerometrów
10 cech	predykcja właściwości rozbłysków słonecznych
24 cechy	klasyfikacja choroby nerek na podstawie parametrów biochemicznych
127 cech	predykcja parametrów przestępstw i wykroczeń na podstawie danych socjoekonomicznych
1203 cechy	Toksyczność cząsteczek mających wpływ na zegar biologiczny
65 tys. cech	obrazek 256×256
150 tys. cech	Zespół czujników gazowych wystawiony na działanie turbulentnych mieszanek gazowych
3+ mln cech	reputacja adresów URL

1203 cechy? 150 tys. cech? Miliony cech?

- Trudność inspekcji (tysiące → miliony przykładów)
- Złożoność – wymagana „pojemność" modelu (zdolność do reprezentacji)
  - Trudność doboru modelu, architektury
  - Czas trwania przygotowania modelu
  - Zasoby pamięciowe i obliczeniowe na przygotowanie i testowanie modelu
- Właściwości przestrzeni wielowymiarowych
  - Przekleństwo wymiarowości (**curse of dimensionality**)

1203 cechy? 150 tys. cech? Miliony cech?

- Trudność inspekcji (tysiące → miliony przykładów)
- Złożoność – wymagana „pojemność" modelu (zdolność do reprezentacji)
  - Trudność doboru modelu, architektury
  - Czas trwania przygotowania modelu
  - Zasoby pamięciowe i obliczeniowe na przygotowanie i testowanie modelu
- Właściwości przestrzeni wielowymiarowych
  - Przekleństwo wymiarowości (**curse of dimensionality**)

1203 cechy? 150 tys. cech? Miliony cech?

- Trudność inspekcji (tysiące → miliony przykładów)
- Złożoność – wymagana „pojemność" modelu (zdolność do reprezentacji)
  - Trudność doboru modelu, architektury
  - Czas trwania przygotowania modelu
  - Zasoby pamięciowe i obliczeniowe na przygotowanie i testowanie modelu
- Właściwości przestrzeni wielowymiarowych
  - Przekleństwo wymiarowości (**curse of dimensionality**)

1203 cechy? 150 tys. cech? Miliony cech?

- Trudność inspekcji (tysiące → miliony przykładów)
- Złożoność – wymagana „pojemność" modelu (zdolność do reprezentacji)
  - Trudność doboru modelu, architektury
  - Czas trwania przygotowania modelu
  - Zasoby pamięciowe i obliczeniowe na przygotowanie i testowanie modelu
- Właściwości przestrzeni wielowymiarowych
  - Przekleństwo wymiarowości (**curse of dimensionality**)

1203 cechy? 150 tys. cech? Miliony cech?

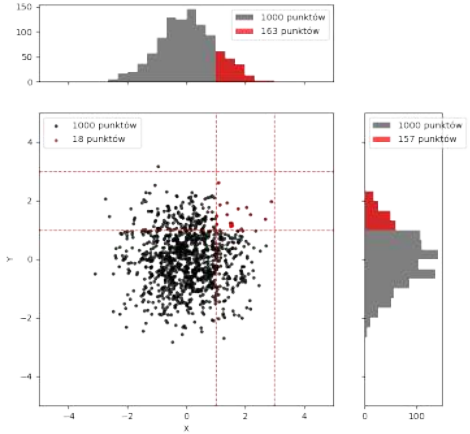
- Trudność inspekcji (tysiące → miliony przykładów)
- Złożoność – wymagana „pojemność" modelu (zdolność do reprezentacji)
  - Trudność doboru modelu, architektury
  - Czas trwania przygotowania modelu
  - Zasoby pamięciowe i obliczeniowe na przygotowanie i testowanie modelu
- Właściwości przestrzeni wielowymiarowych
  - Przekleństwo wymiarowości (**curse of dimensionality**)

Przekleństwo wymiarowości

- Rzadkość danych w przestrzeniach wielowymiarowych
- „Ucentrowienie" (hubness), np. przy szukaniu podobnych przykładów
- Wymagania liczby punktów danych $n > d, n \geqslant 10d, n \gg d$
- Wymagania obliczeniowe i pamięciowe
- Trudność oszacowania parametrów statystycznych (korelacje, outliery)

Przekleństwo wymiarowości

- Rzadkość danych w przestrzeniach wielowymiarowych



- „Ucentrowienie” (hubness), np. przy szukaniu podobnych przykładów

Przekleństwo wymiarowości

- Rzadkość danych w przestrzeniach wielowymiarowych
- „Ucentrowienie" (hubness), np. przy szukaniu podobnych przykładów
- Wymagania liczby punktów danych $n > d, n \geqslant 10d, n \gg d$
- Wymagania obliczeniowe i pamięciowe
- Trudność oszacowania parametrów statystycznych (korelacje, outliery)

Korelacje między cechami

Intuicja:

- Dane z urządzeń IoT mierzących temperaturę, wilgotność itp. w mieście
- Film z kamery monitoringu parkingu
- Dane mieszkań: cena, rozmiar, liczba pokojów, liczba łazienek, miejsce dodatkowe (piwnica/garaż)
- Zużycie wody w okresie godzinowym w ciągu tygodnia

Reprezentacja korelacji:

- Nazwane cechy (np. cena mieszkania i rozmiar w m^2)
- Nie nazwane cechy (np. piksele obrazka)

Korelacje między cechami

Intuicja:

- Dane z urządzeń IoT mierzących temperaturę, wilgotność itp. w mieście
- Film z kamery monitoringu parkingu
- Dane mieszkań: cena, rozmiar, liczba pokojów, liczba łazienek, miejsce dodatkowe (piwnica/garaż)
- Zużycie wody w okresie godzinowym w ciągu tygodnia

Reprezentacja korelacji:

- Nazwane cechy (np. cena mieszkania i rozmiar w m^2)
- Nie nazwane cechy (np. piksele obrazka)

Korelacje między cechami

Intuicja:

- Dane z urządzeń IoT mierzących temperaturę, wilgotność itp. w mieście
- Film z kamery monitoringu parkingu
- Dane mieszkań: cena, rozmiar, liczba pokojów, liczba łazienek, miejsce dodatkowe (piwnica/garaż)
- Zużycie wody w okresie godzinowym w ciągu tygodnia

Reprezentacja korelacji:

- Nazwane cechy (np. cena mieszkania i rozmiar w m^2)
- Nie nazwane cechy (np. piksele obrazka)

Korelacje między cechami

Intuicja:

- Dane z urządzeń IoT mierzących temperaturę, wilgotność itp. w mieście
- Film z kamery monitoringu parkingu
- Dane mieszkań: cena, rozmiar, liczba pokojów, liczba łazienek, miejsce dodatkowe (piwnica/garaż)
- Zużycie wody w okresie godzinowym w ciągu tygodnia

Reprezentacja korelacji:

- Nazwane cechy (np. cena mieszkania i rozmiar w m^2)
- Nie nazwane cechy (np. piksele obrazka)

Korelacje między cechami

Intuicja:

- Dane z urządzeń IoT mierzących temperaturę, wilgotność itp. w mieście
- Film z kamery monitoringu parkingu
- Dane mieszkań: cena, rozmiar, liczba pokojów, liczba łazienek, miejsce dodatkowe (piwnica/garaż)
- Zużycie wody w okresie godzinowym w ciągu tygodnia

Reprezentacja korelacji:

- Nazwane cechy (np. cena mieszkania i rozmiar w m^2)
- Nie nazwane cechy (np. piksele obrazka)

Korelacje między cechami

Intuicja:

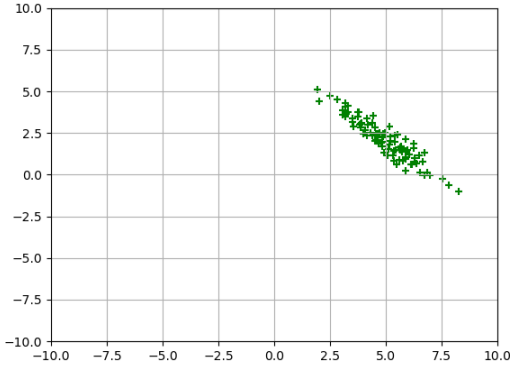
- Dane z urządzeń IoT mierzących temperaturę, wilgotność itp. w mieście
- Film z kamery monitoringu parkingu
- Dane mieszkań: cena, rozmiar, liczba pokojów, liczba łazienek, miejsce dodatkowe (piwnica/garaż)
- Zużycie wody w okresie godzinowym w ciągu tygodnia

Reprezentacja korelacji:

- Nazwane cechy (np. cena mieszkania i rozmiar w m^2)
- Nie nazwane cechy (np. piksele obrazka)

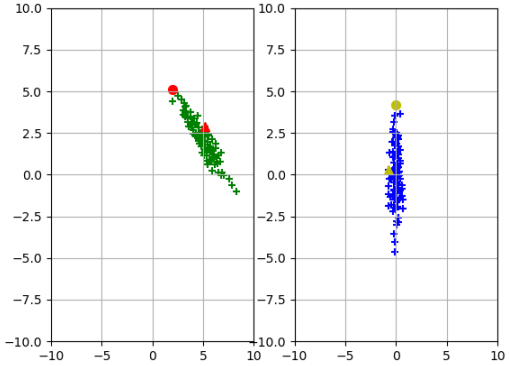
Dekorelacja

Intuicja:



Dekorelacja

Intuicja:



Analiza składowych głównych – wyznaczanie

1. Zbiór przykładów $\mathcal{X}$
   $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$
   $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $x_{ij} \in \mathbb{R}$
2. Zbiór po odjęciu średniej
   $\mathbf{x}'_i = \mathbf{x}_i - \frac{1}{m}\sum_{i=1}^{m} \mathbf{x}_i = \mathbf{x}_i - \bar{\mathbf{x}}$
   zapisujemy w postaci macierzy
   $$X = \begin{bmatrix} x'_{11} & x'_{12} & \ldots \\ x'_{21} & x'_{22} & \ldots \\ \ldots & \ldots & x'_{mn} \end{bmatrix}$$
3. Wyznaczamy macierz kowariancji
   $S = \frac{1}{m-1} X X^\top$
   i jej wektory własne (posortowane po wartościach własnych) układamy w macierz A
4. Dla dowolnego (w ramach rodziny danych) wektora x uzyskujemy składowe z równania
   $\mathbf{c} = A(\mathbf{x} - \bar{\mathbf{x}})$

Analiza składowych głównych – wyznaczanie

1. Zbiór przykładów $\mathcal{X}$
   $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$
   $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $x_{ij} \in \mathbb{R}$
2. Zbiór po odjęciu średniej
   $\mathbf{x}'_i = \mathbf{x}_i - \frac{1}{m}\sum_{i=1}^{m} \mathbf{x}_i = \mathbf{x}_i - \bar{\mathbf{x}}$
   zapisujemy w postaci macierzy
   $$X = \begin{bmatrix} x'_{11} & x'_{12} & \ldots \\ x'_{21} & x'_{22} & \ldots \\ \ldots & \ldots & x'_{mn} \end{bmatrix}$$
3. Wyznaczamy macierz kowariancji
   $S = \frac{1}{m-1} X X^\top$
   i jej wektory własne (posortowane po wartościach własnych) układamy w macierz A
4. Dla dowolnego (w ramach rodziny danych) wektora x uzyskujemy składowe z równania
   $\mathbf{c} = A(\mathbf{x} - \bar{\mathbf{x}})$

Analiza składowych głównych – wyznaczanie

1. Zbiór przykładów $\mathcal{X}$
   $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$
   $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $x_{ij} \in \mathbb{R}$
2. Zbiór po odjęciu średniej
   $\mathbf{x}'_i = \mathbf{x}_i - \frac{1}{m}\sum_{i=1}^{m} \mathbf{x}_i = \mathbf{x}_i - \bar{\mathbf{x}}$
   zapisujemy w postaci macierzy
   $$X = \begin{bmatrix} x'_{11} & x'_{12} & \ldots \\ x'_{21} & x'_{22} & \ldots \\ \ldots & \ldots & x'_{mn} \end{bmatrix}$$
3. Wyznaczamy macierz kowariancji
   $S = \frac{1}{m-1} X X^\top$
   i jej wektory własne (posortowane po wartościach własnych) układamy w macierz A
4. Dla dowolnego (w ramach rodziny danych) wektora x uzyskujemy składowe z równania
   $\mathbf{c} = A(\mathbf{x} - \bar{\mathbf{x}})$

Analiza składowych głównych – wyznaczanie

1. Zbiór przykładów $\mathcal{X}$
   $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$
   $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $x_{ij} \in \mathbb{R}$
2. Zbiór po odjęciu średniej
   $\mathbf{x}'_i = \mathbf{x}_i - \frac{1}{m}\sum_{i=1}^{m} \mathbf{x}_i = \mathbf{x}_i - \bar{\mathbf{x}}$
   zapisujemy w postaci macierzy
   $$X = \begin{bmatrix} x'_{11} & x'_{12} & \ldots \\ x'_{21} & x'_{22} & \ldots \\ \ldots & \ldots & x'_{mn} \end{bmatrix}$$
3. Wyznaczamy macierz kowariancji
   $S = \frac{1}{m-1} X X^\top$
   i jej wektory własne (posortowane po wartościach własnych) układamy w macierz A
4. Dla dowolnego (w ramach rodziny danych) wektora x uzyskujemy składowe z równania
   $\mathbf{c} = A(\mathbf{x} - \bar{\mathbf{x}})$

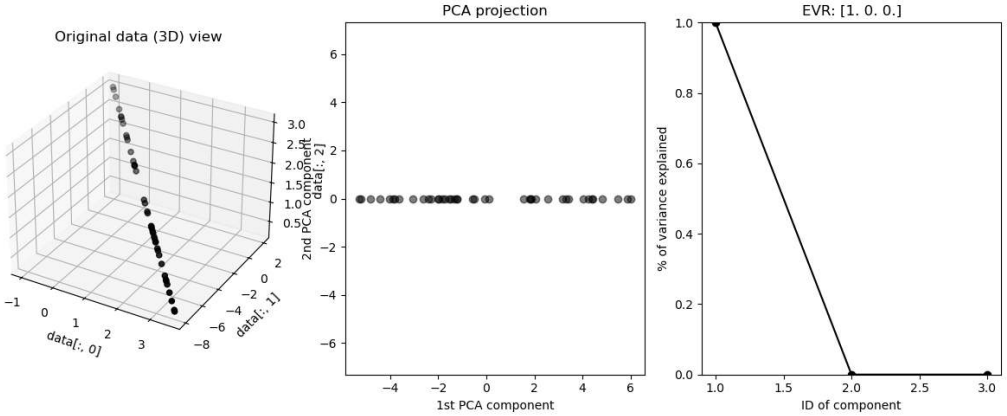
Analiza składowych głównych – wyznaczanie

1. Zbiór przykładów $\mathcal{X}$
   $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$
   $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $x_{ij} \in \mathbb{R}$
2. Zbiór po odjęciu średniej
   $\mathbf{x}'_i = \mathbf{x}_i - \frac{1}{m}\sum_{i=1}^{m} \mathbf{x}_i = \mathbf{x}_i - \bar{\mathbf{x}}$
   zapisujemy w postaci macierzy
   $$X = \begin{bmatrix} x'_{11} & x'_{12} & \ldots \\ x'_{21} & x'_{22} & \ldots \\ \ldots & \ldots & x'_{mn} \end{bmatrix}$$
3. Wyznaczamy macierz kowariancji
   $S = \frac{1}{m-1} X X^\top$
   i jej wektory własne (posortowane po wartościach własnych) układamy w macierz A
4. Dla dowolnego (w ramach rodziny danych) wektora x uzyskujemy składowe z równania
   $\mathbf{c} = A(\mathbf{x} - \bar{\mathbf{x}})$

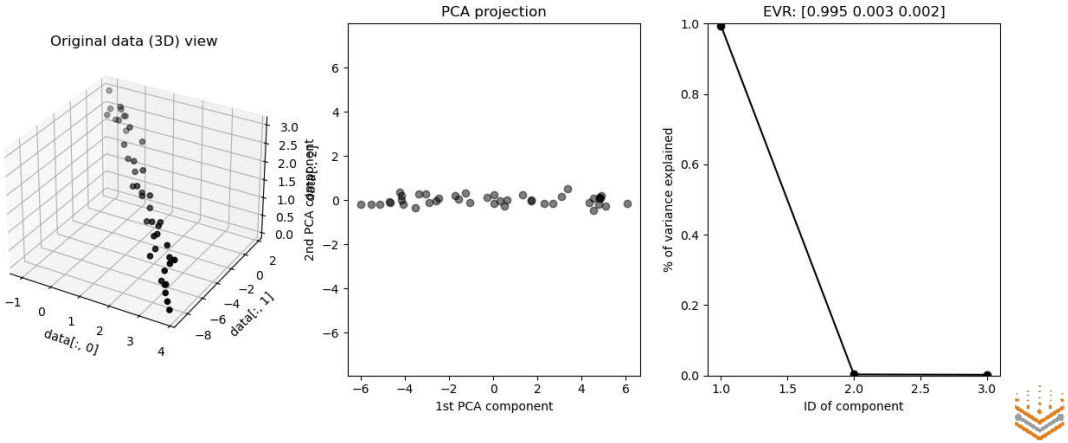
Analiza składowych głównych – wyznaczanie

1. Zbiór przykładów $\mathcal{X}$
   $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$
   $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $x_{ij} \in \mathbb{R}$
2. Zbiór po odjęciu średniej
   $\mathbf{x}'_i = \mathbf{x}_i - \frac{1}{m}\sum_{i=1}^{m} \mathbf{x}_i = \mathbf{x}_i - \bar{\mathbf{x}}$
   zapisujemy w postaci macierzy
   $$X = \begin{bmatrix} x'_{11} & x'_{12} & \ldots \\ x'_{21} & x'_{22} & \ldots \\ \ldots & \ldots & x'_{mn} \end{bmatrix}$$
3. Wyznaczamy macierz kowariancji
   $S = \frac{1}{m-1} X X^\top$
   i jej wektory własne (posortowane po wartościach własnych) układamy w macierz A
4. Dla dowolnego (w ramach rodziny danych) wektora x uzyskujemy składowe z równania
   $\mathbf{c} = A(\mathbf{x} - \bar{\mathbf{x}})$

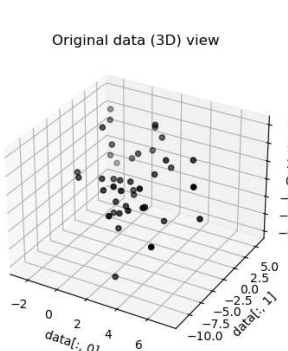
Analiza składowych głównych – przykład/idea działania

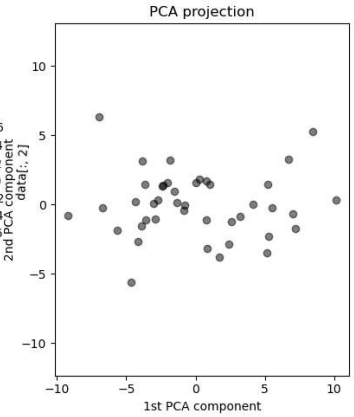


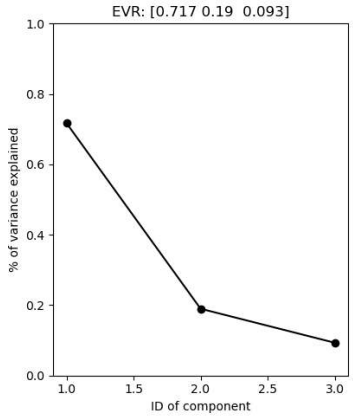
Analiza składowych głównych – przykład/idea działania



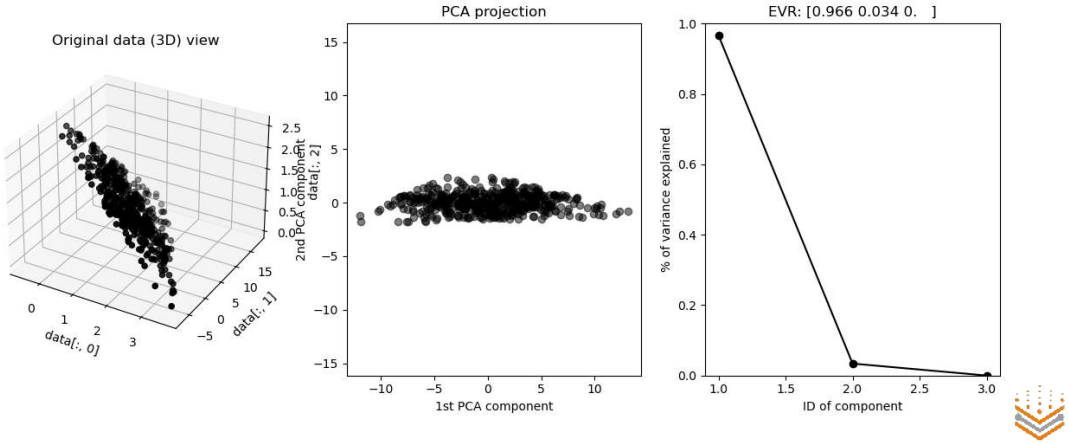
Analiza składowych głównych – przykład/idea działania



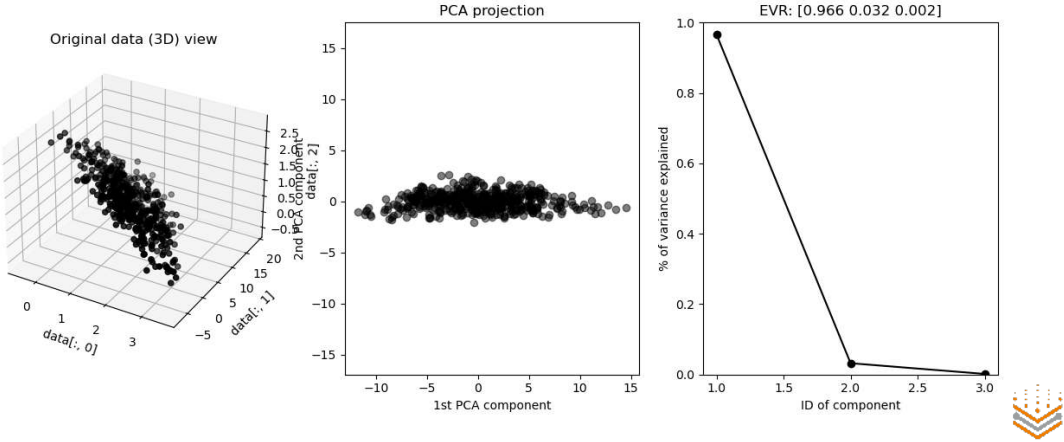




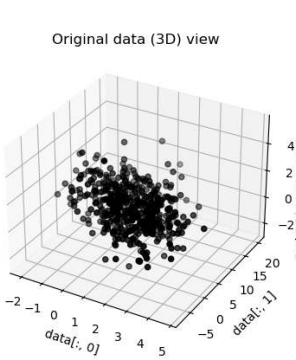
Analiza składowych głównych – przykład/idea działania

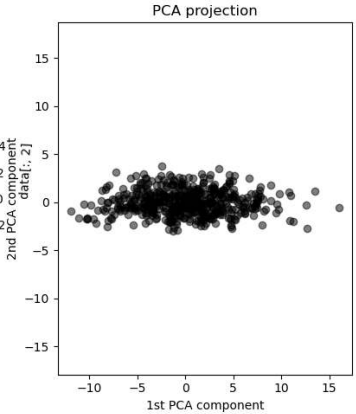


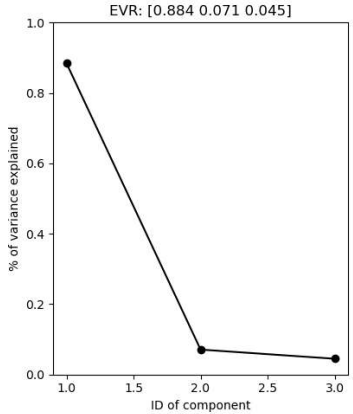
Analiza składowych głównych – przykład/idea działania



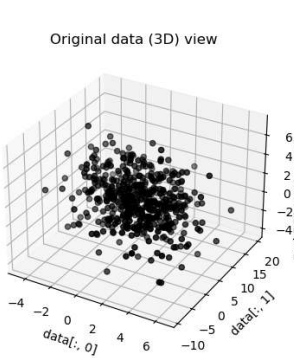
Analiza składowych głównych – przykład/idea działania

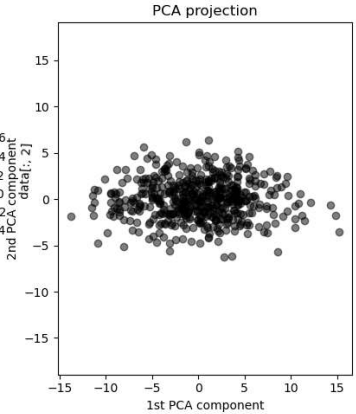


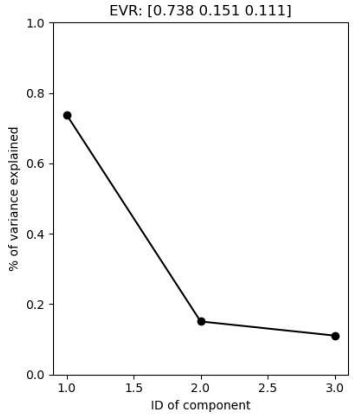




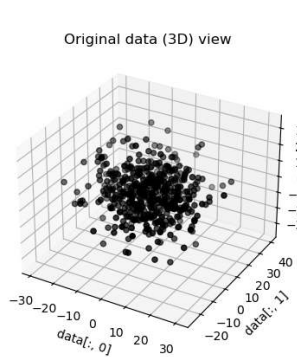
Analiza składowych głównych – przykład/idea działania

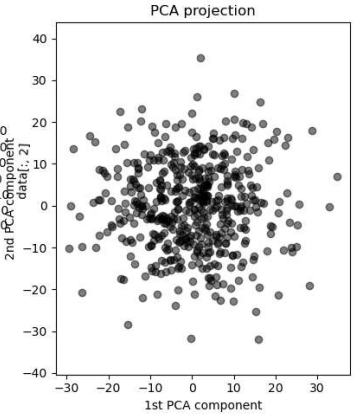


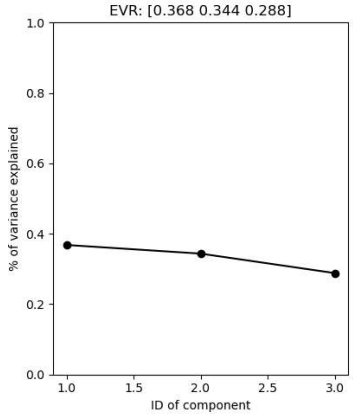




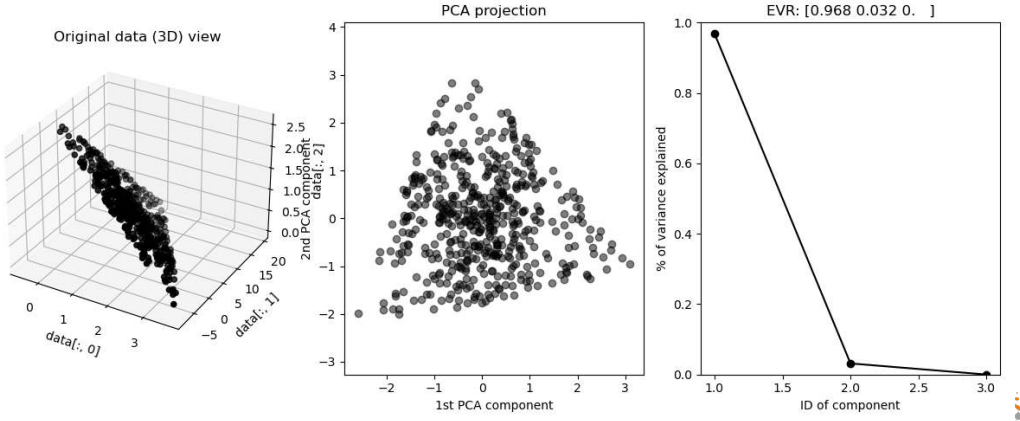
Analiza składowych głównych – przykład/idea działania







Analiza składowych głównych – przykład (wybielenie – whitening)



Analiza składowych głównych – przykład zastosowania (reprezentacja)

- x to zlinearyzowany obraz ($x \in \mathbb{R}^{4096}$):

$\mathbf{x}_1$	$\mathbf{x}_2$
$\mathbf{x}_3$	$\mathbf{x}_4$

- Składowe (wektory z macierzy A) tzw. „eigenfaces”

. . . cechy twarzy

Analiza składowych głównych – przykład zastosowania (reprezentacja)

- x to zlinearyzowany obraz ($x \in \mathbb{R}^{4096}$):

$$\begin{array}{c|c} \mathbf{x}_1 & \mathbf{x}_2 \\ \hline \mathbf{x}_3 & \mathbf{x}_4 \end{array}$$

- Składowe (wektory z macierzy A) tzw. „eigenfaces”

. . . cechy twarzy

Analiza składowych głównych – przykład zastosowania (reprezentacja)

- x to zlinearyzowany obraz ($x \in \mathbb{R}^{4096}$):

$\mathbf{x}_1$	$\mathbf{x}_2$
$\mathbf{x}_3$	$\mathbf{x}_4$

- Składowe (wektory z macierzy A) tzw. „eigenfaces”

. . . cechy twarzy

Analiza składowych głównych – przykład zastosowania (reprezentacja)

- $\mathbf{x} \in \mathbb{R}^{4096} \rightarrow \mathbf{c} \in \mathbb{R}^{4096}$?
  - Macierz kowariancji $S = \frac{1}{m-1} X X^\top$
  - Wektory własne ($\rightarrow A$) oraz wartości własne (wariancja, posortowane)
  - Można wykorzystać pierwszych n wektorów – $\mathbf{x} \in \mathbb{R}^{4096} \rightarrow \mathbf{c} \in \mathbb{R}^{n}$; np. $n = 5, 10, 50$
- Rekonstrukcja z 1, 9, 17, ... (+8 składowych):

Analiza składowych głównych – przykład zastosowania (reprezentacja)

- $\mathbf{x} \in \mathbb{R}^{4096} \rightarrow \mathbf{c} \in \mathbb{R}^{4096}$?
  - Macierz kowariancji $S = \frac{1}{m-1} X X^\top$
  - Wektory własne ($\rightarrow A$) oraz wartości własne (wariancja, posortowane)
  - Można wykorzystać pierwszych n wektorów – $\mathbf{x} \in \mathbb{R}^{4096} \rightarrow \mathbf{c} \in \mathbb{R}^{n}$; np. $n = 5, 10, 50$
- Rekonstrukcja z 1, 9, 17, ... (+8 składowych):

Analiza składowych głównych – przykład zastosowania (reprezentacja)

- $\mathbf{x} \in \mathbb{R}^{4096} \rightarrow \mathbf{c} \in \mathbb{R}^{4096}$?
  - Macierz kowariancji $S = \frac{1}{m-1} X X^\top$
  - Wektory własne ($\rightarrow A$) oraz wartości własne (wariancja, posortowane)
  - Można wykorzystać pierwszych n wektorów – $\mathbf{x} \in \mathbb{R}^{4096} \rightarrow \mathbf{c} \in \mathbb{R}^{n}$; np. $n = 5, 10, 50$
- Rekonstrukcja z 1, 9, 17, ... (+8 składowych):

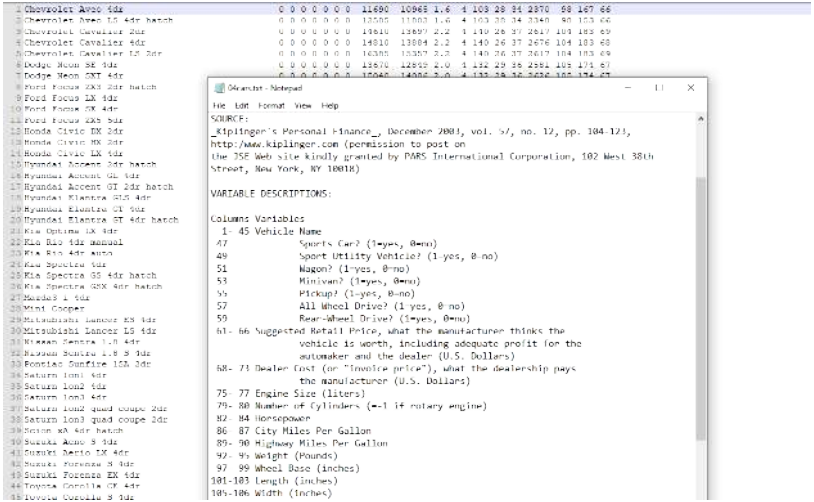
Analiza składowych głównych – przykład zastosowania (reprezentacja)

- $\mathbf{x} \in \mathbb{R}^{4096} \rightarrow \mathbf{c} \in \mathbb{R}^{4096}$?
  - Macierz kowariancji $S = \frac{1}{m-1} X X^\top$
  - Wektory własne ($\rightarrow A$) oraz wartości własne (wariancja, posortowane)
  - Można wykorzystać pierwszych n wektorów – $\mathbf{x} \in \mathbb{R}^{4096} \rightarrow \mathbf{c} \in \mathbb{R}^{n}$; np. $n = 5, 10, 50$
- Rekonstrukcja z 1, 9, 17, ... (+8 składowych):

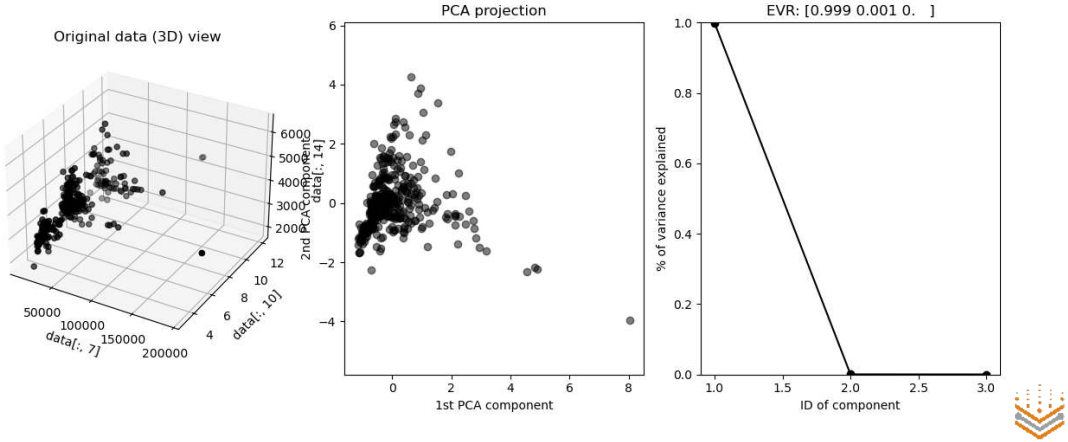
Analiza składowych głównych – przykład zastosowania (reprezentacja)

- $\mathbf{x} \in \mathbb{R}^{4096} \rightarrow \mathbf{c} \in \mathbb{R}^{4096}$?
  - Macierz kowariancji $S = \frac{1}{m-1} X X^\top$
  - Wektory własne ($\rightarrow A$) oraz wartości własne (wariancja, posortowane)
  - Można wykorzystać pierwszych n wektorów – $\mathbf{x} \in \mathbb{R}^{4096} \rightarrow \mathbf{c} \in \mathbb{R}^n$; np. $n = 5, 10, 50$
- Rekonstrukcja z 1, 9, 17, ... (+8 składowych):

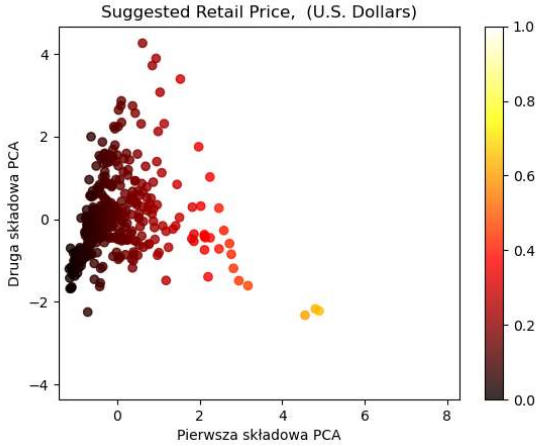
Analiza składowych głównych – przykład zastosowania (analiza)



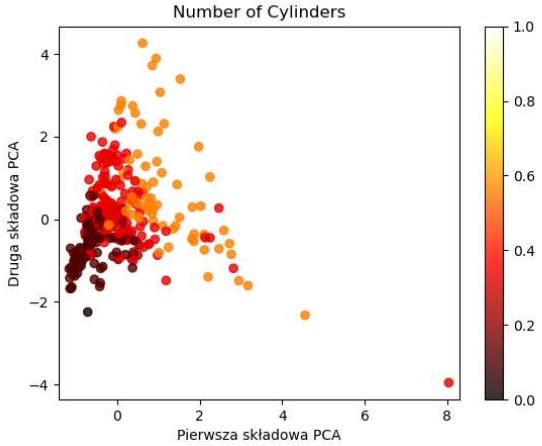
Analiza składowych głównych – przykład zastosowania (analiza)



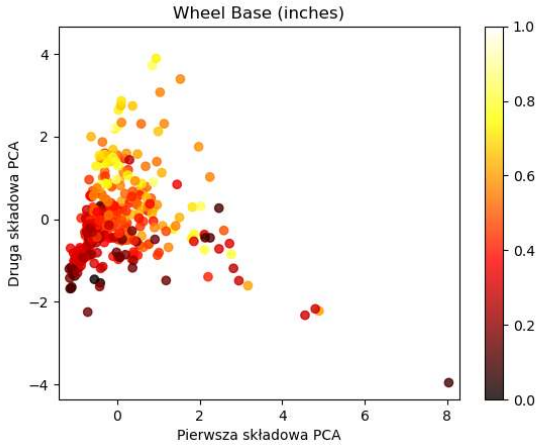
Analiza składowych głównych – przykład zastosowania (analiza)



Analiza składowych głównych – przykład zastosowania (analiza)



Analiza składowych głównych – przykład zastosowania (analiza)



Analiza składowych głównych – przykład zastosowania (analiza)

```
Component 0 ---------------------------

Examples
Lowest
['Kia Rio 4dr manual' 'Hyundai Accent 2dr hatch' 'Toyota Echo 2dr manual']
Highest
['Mercedes-Benz SL600 convertible 2dr' 'Mercedes-Benz CL600 2dr'
'Porsche 911 GT2 2dr']

Column correlations
07 1.00  61- 66 Suggested Retail Price, what the manufacturer thinks the vehicle
is worth, including adequate profit for the automaker and the dealer (U.S. Dollars)
08 1.00  68- 73 Dealer Cost (or "invoice price"), what the dealership pays
```

Analiza składowych głównych – przykład zastosowania (analiza)

```
Component 1 ----------------------------

Examples
Lowest
['Porsche 911 GT2 2dr' 'Mercedes-Benz SL55 AMG 2dr'
'Honda Insight 2dr (gas/electric)']
Highest
['Hummer H2' 'Lincoln Navigator Luxury' 'GMC Yukon XL 2500 SLT']

Column correlations
14 0.83  92- 95 Weight (Pounds)
17 0.74 105-106 Width (inches)
```

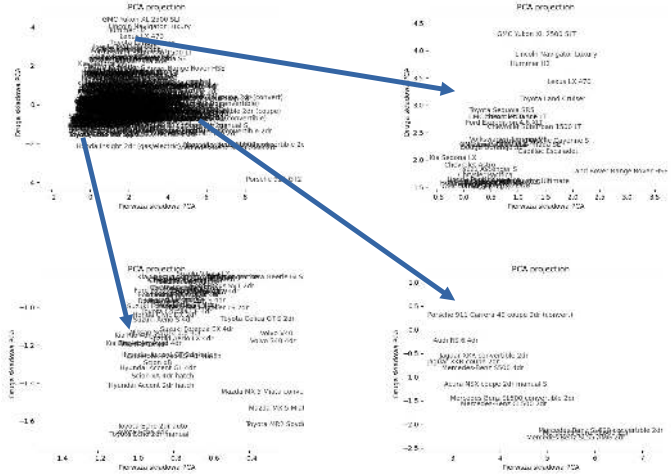
Analiza składowych głównych – przykład zastosowania (analiza)

```
Component 4 ---------------------------

Examples
Lowest
['Lincoln Town Car Ultimate L 4dr' 'Chrysler Concorde LX 4dr'
'Lincoln Town Car Ultimate 4dr']
Highest
['Mini Cooper S' 'Hummer H2' 'Jeep Wrangler Sahara convertible 2dr']

Column correlations
16 0.74 101-103 Length (inches)
15 0.51  97- 99 Wheel Base (inches)
```

Analiza składowych głównych – przykład zastosowania (analiza)



Analiza składowych głównych – podsumowanie

1. Reprezentacja x uwzględniająca charakterystykę źródła danych
   - Mniej współczynników do reprezentacji i klasyfikacji
   - Podzbiór współczynników zachowuje zakres zmienności danych
   - Analiza zbioru danych – data mining – weryfikacja obecności grup, anomalii, outlierów

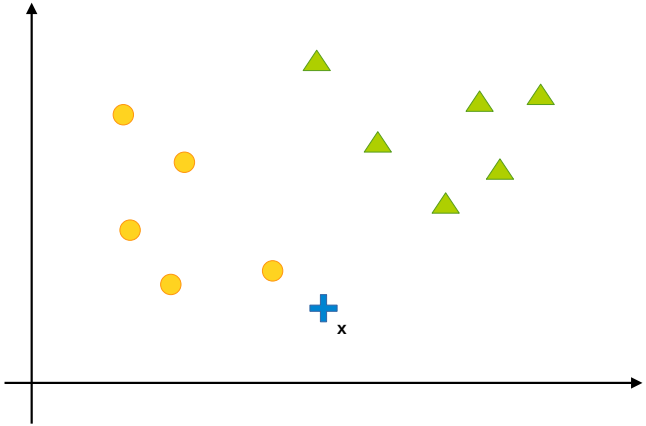
Analiza składowych głównych – podsumowanie

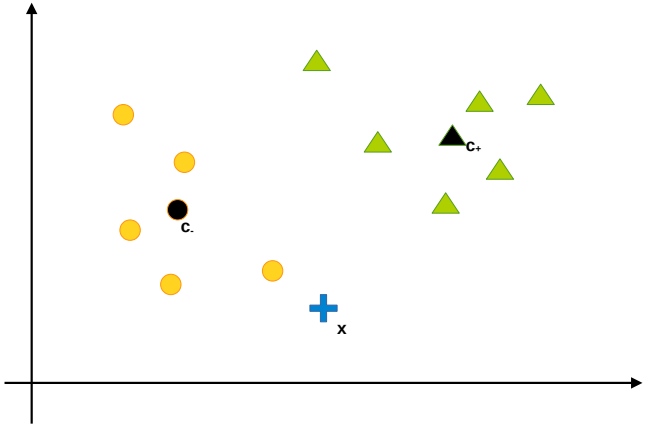
1. Reprezentacja x uwzględniająca charakterystykę źródła danych
   - Mniej współczynników do reprezentacji i klasyfikacji
   - Podzbiór współczynników zachowuje zakres zmienności danych
   - Analiza zbioru danych – data mining – weryfikacja obecności grup, anomalii, outlierów

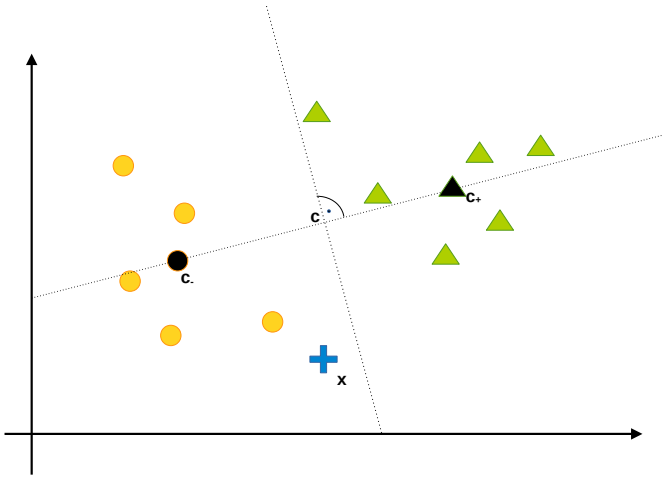
Analiza składowych głównych – podsumowanie

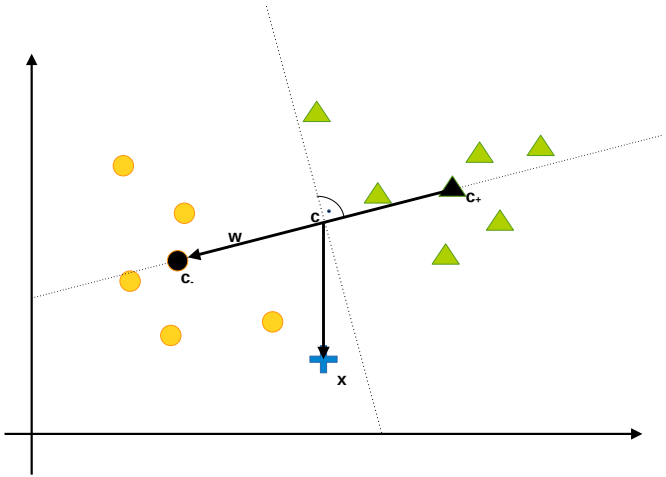
1. Reprezentacja x uwzględniająca charakterystykę źródła danych
   - Mniej współczynników do reprezentacji i klasyfikacji
   - Podzbiór współczynników zachowuje zakres zmienności danych
   - Analiza zbioru danych – data mining – weryfikacja obecności grup, anomalii, outlierów

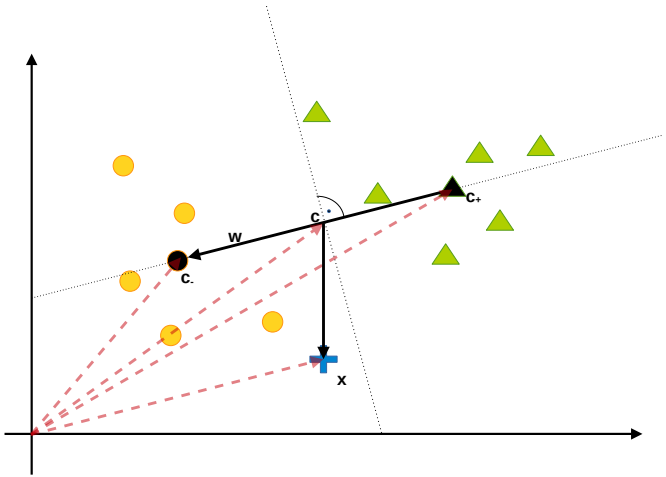
Support Vector Machine

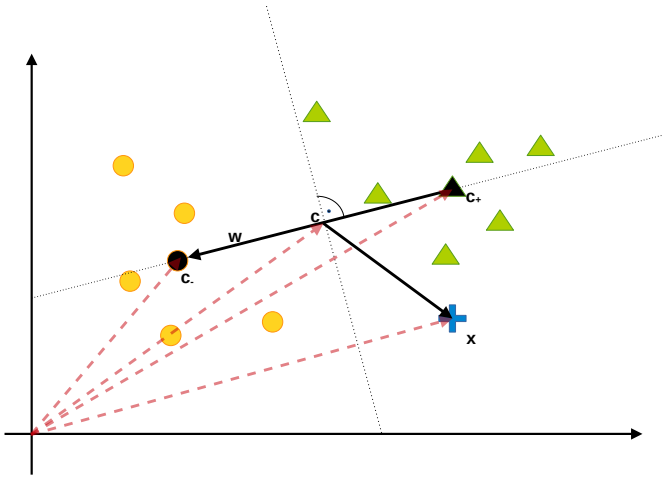

x

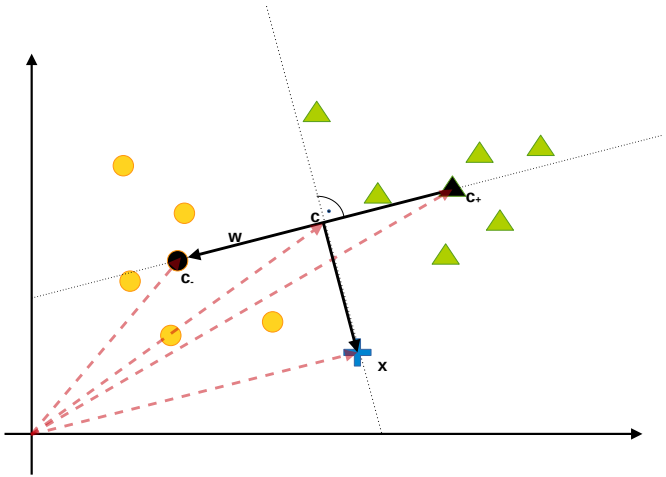

C+
C-
x

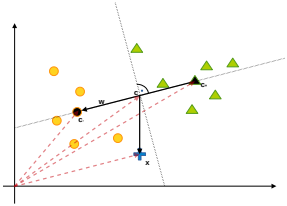

c+
c-
x

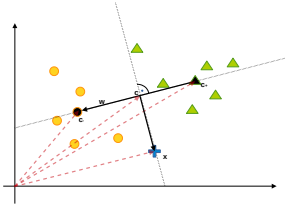

c*
c+
c-
w
x

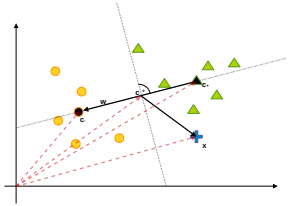

c*
c+
c-
w
x


c+
w
c-
x


c+
w
c-
x





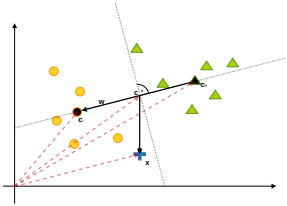


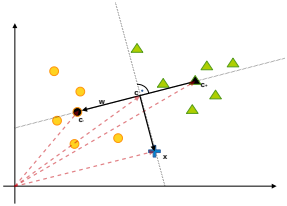
$\mathbf{c}_+ = \frac{1}{n_+}\sum_{i\in\mathcal{I}_+}\mathbf{x}_i \qquad \mathbf{c}_- = \frac{1}{n_-}\sum_{i\in\mathcal{I}_-}\mathbf{x}_i \qquad \mathbf{c} = \frac{\mathbf{c}_- + \mathbf{c}_+}{2} \qquad \mathbf{w} = \mathbf{c}_- - \mathbf{c}_+$

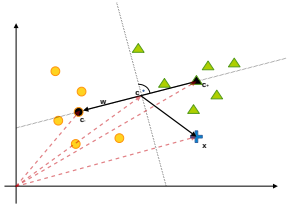
$\mathbf{y} = \operatorname{sgn}\langle(\mathbf{x}-\mathbf{c}),\mathbf{w}\rangle \qquad$ iloczyn skalarny $\langle\mathbf{a},\mathbf{b}\rangle = \|\mathbf{a}\|\|\mathbf{b}\|\cos\theta$

$$\mathbf{y} = \operatorname{sgn}\left\langle(\mathbf{x} - \tfrac{\mathbf{c}_- + \mathbf{c}_+}{2}), \mathbf{c}_- - \mathbf{c}_+\right\rangle =$$

$$\mathbf{y} = \operatorname{sgn}(\langle\mathbf{x},\mathbf{c}_-\rangle - \langle\mathbf{x},\mathbf{c}_+\rangle \underbrace{- \left\langle\frac{\mathbf{c}_- + \mathbf{c}_+}{2}, \mathbf{c}_-\right\rangle + \left\langle\frac{\mathbf{c}_- + \mathbf{c}_+}{2}, \mathbf{c}_+\right\rangle}_{\text{nie zależy od } \mathbf{x}})$$





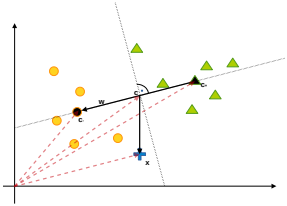


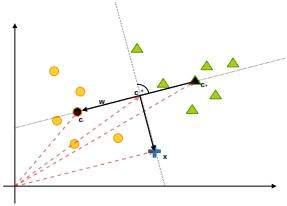
$$\mathbf{c}_+ = \frac{1}{n_+}\sum_{i\in\mathcal{I}_+}\mathbf{x}_i \qquad \mathbf{c}_- = \frac{1}{n_-}\sum_{i\in\mathcal{I}_-}\mathbf{x}_i \qquad \mathbf{c} = \frac{\mathbf{c}_- + \mathbf{c}_+}{2} \qquad \mathbf{w} = \mathbf{c}_- - \mathbf{c}_+$$

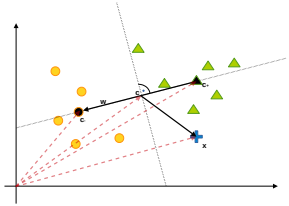
$$\mathbf{y} = \operatorname{sgn}\langle(\mathbf{x}-\mathbf{c}),\mathbf{w}\rangle \qquad \text{iloczyn skalarny } \langle\mathbf{a},\mathbf{b}\rangle = \|\mathbf{a}\|\|\mathbf{b}\|\cos\theta$$

$$\mathbf{y} = \operatorname{sgn}\left\langle(\mathbf{x}-\tfrac{\mathbf{c}_- + \mathbf{c}_+}{2}),\mathbf{c}_- - \mathbf{c}_+\right\rangle =$$

$$\mathbf{y} = \operatorname{sgn}(\langle\mathbf{x},\mathbf{c}_-\rangle - \langle\mathbf{x},\mathbf{c}_+\rangle \underbrace{- \left\langle\frac{\mathbf{c}_- + \mathbf{c}_+}{2},\mathbf{c}_-\right\rangle + \left\langle\frac{\mathbf{c}_- + \mathbf{c}_+}{2},\mathbf{c}_+\right\rangle}_{\text{nie zależy od } \mathbf{x}})$$





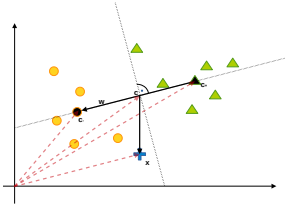


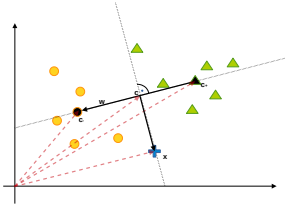
$\mathbf{c}_+ = \frac{1}{n_+}\sum_{i\in\mathcal{I}_+}\mathbf{x}_i \qquad \mathbf{c}_- = \frac{1}{n_-}\sum_{i\in\mathcal{I}_-}\mathbf{x}_i \qquad \mathbf{c} = \frac{\mathbf{c}_- + \mathbf{c}_+}{2} \qquad \mathbf{w} = \mathbf{c}_- - \mathbf{c}_+$

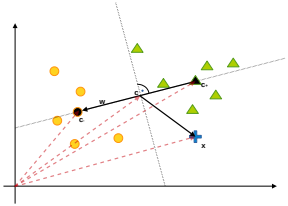
$\mathbf{y} = \operatorname{sgn}\langle(\mathbf{x}-\mathbf{c}),\mathbf{w}\rangle$ iloczyn skalarny $\langle\mathbf{a},\mathbf{b}\rangle = \|\mathbf{a}\|\|\mathbf{b}\|\cos\theta$

$$\mathbf{y} = \operatorname{sgn}\left\langle(\mathbf{x} - \tfrac{\mathbf{c}_- + \mathbf{c}_+}{2}), \mathbf{c}_- - \mathbf{c}_+\right\rangle =$$

$$\mathbf{y} = \operatorname{sgn}(\langle\mathbf{x},\mathbf{c}_-\rangle - \langle\mathbf{x},\mathbf{c}_+\rangle \underbrace{- \left\langle\frac{\mathbf{c}_- + \mathbf{c}_+}{2}, \mathbf{c}_-\right\rangle + \left\langle\frac{\mathbf{c}_- + \mathbf{c}_+}{2}, \mathbf{c}_+\right\rangle}_{\text{nie zależy od } \mathbf{x}})$$





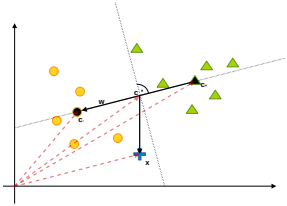


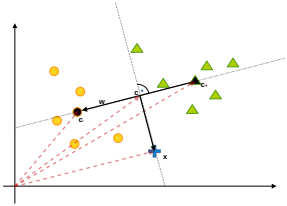
$\mathbf{c}_+ = \frac{1}{n_+}\sum_{i\in\mathcal{I}_+}\mathbf{x}_i \qquad \mathbf{c}_- = \frac{1}{n_-}\sum_{i\in\mathcal{I}_-}\mathbf{x}_i \qquad \mathbf{c} = \frac{\mathbf{c}_- + \mathbf{c}_+}{2} \qquad \mathbf{w} = \mathbf{c}_- - \mathbf{c}_+$

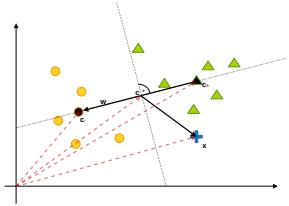
$\mathbf{y} = \operatorname{sgn}\langle(\mathbf{x}-\mathbf{c}),\mathbf{w}\rangle$ iloczyn skalarny $\langle\mathbf{a},\mathbf{b}\rangle = \|\mathbf{a}\|\|\mathbf{b}\|\cos\theta$

$\mathbf{y} = \operatorname{sgn}\left\langle(\mathbf{x} - \frac{\mathbf{c}_- + \mathbf{c}_+}{2}), \mathbf{c}_- - \mathbf{c}_+\right\rangle =$

$\mathbf{y} = \operatorname{sgn}(\langle\mathbf{x},\mathbf{c}_-\rangle - \langle\mathbf{x},\mathbf{c}_+\rangle \underbrace{- \left\langle\frac{\mathbf{c}_- + \mathbf{c}_+}{2}, \mathbf{c}_-\right\rangle + \left\langle\frac{\mathbf{c}_- + \mathbf{c}_+}{2}, \mathbf{c}_+\right\rangle}_{\text{nie zależy od } \mathbf{x}})$





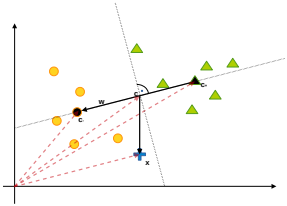


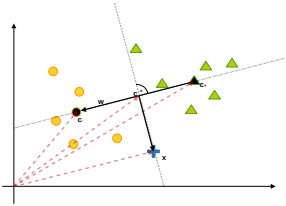
$$\mathbf{c}_+ = \frac{1}{n_+}\sum_{i\in\mathcal{I}_+}\mathbf{x}_i \qquad \mathbf{c}_- = \frac{1}{n_-}\sum_{i\in\mathcal{I}_-}\mathbf{x}_i \qquad \mathbf{c} = \frac{\mathbf{c}_- + \mathbf{c}_+}{2} \qquad \mathbf{w} = \mathbf{c}_- - \mathbf{c}_+$$

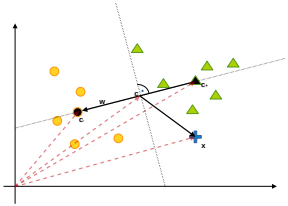
$$\mathbf{y} = \operatorname{sgn}\langle(\mathbf{x}-\mathbf{c}),\mathbf{w}\rangle \qquad \text{iloczyn skalarny } \langle\mathbf{a},\mathbf{b}\rangle = \|\mathbf{a}\|\|\mathbf{b}\|\cos\theta$$

$$\mathbf{y} = \operatorname{sgn}\left\langle(\mathbf{x} - \tfrac{\mathbf{c}_- + \mathbf{c}_+}{2}), \mathbf{c}_- - \mathbf{c}_+\right\rangle =$$

$$\mathbf{y} = \operatorname{sgn}(\langle\mathbf{x},\mathbf{c}_-\rangle - \langle\mathbf{x},\mathbf{c}_+\rangle \underbrace{- \left\langle\frac{\mathbf{c}_- + \mathbf{c}_+}{2}, \mathbf{c}_-\right\rangle + \left\langle\frac{\mathbf{c}_- + \mathbf{c}_+}{2}, \mathbf{c}_+\right\rangle}_{\text{nie zależy od } \mathbf{x}})$$





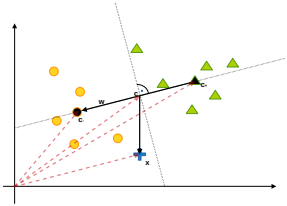


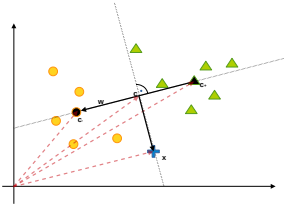
$$\mathbf{y} = \operatorname{sgn}(\langle \mathbf{x}, \mathbf{c}_- \rangle - \langle \mathbf{x}, \mathbf{c}_+ \rangle + b) \qquad \mathbf{c}_- = \frac{1}{n_-} \sum_{i \in \mathcal{I}_-} \mathbf{x}_i$$

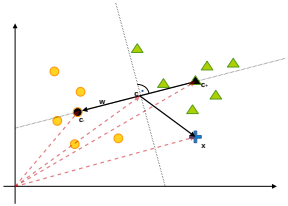
$$\mathbf{y} = \operatorname{sgn}\left(\frac{1}{n_-} \sum_{i \in \mathcal{I}_-} \langle \mathbf{x}_i, \mathbf{x} \rangle - \frac{1}{n_+} \sum_{i \in \mathcal{I}_+} \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$

$$\mathbf{y} = \operatorname{sgn}\left(- \sum_i y_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$

$$\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$





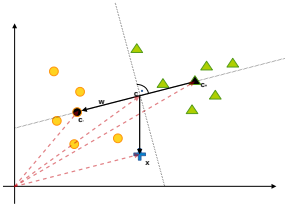


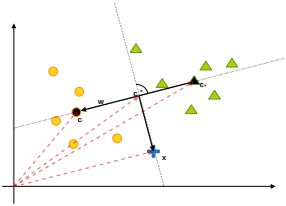
$$\mathbf{y} = \operatorname{sgn}(\langle \mathbf{x}, \mathbf{c}_- \rangle - \langle \mathbf{x}, \mathbf{c}_+ \rangle + b) \qquad \mathbf{c}_- = \frac{1}{n_-} \sum_{i \in \mathcal{I}_-} \mathbf{x}_i$$

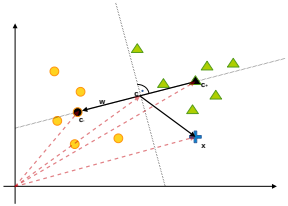
$$\mathbf{y} = \operatorname{sgn}\left(\frac{1}{n_-} \sum_{i \in \mathcal{I}_-} \langle \mathbf{x}_i, \mathbf{x} \rangle - \frac{1}{n_+} \sum_{i \in \mathcal{I}_+} \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$

$$\mathbf{y} = \operatorname{sgn}\left(- \sum_i y_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$

$$\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$





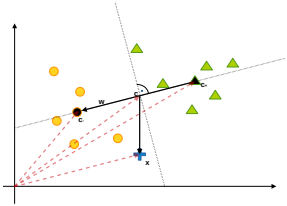


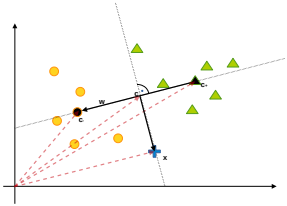
$$\mathbf{y} = \operatorname{sgn}(\langle \mathbf{x}, \mathbf{c}_- \rangle - \langle \mathbf{x}, \mathbf{c}_+ \rangle + b) \qquad \mathbf{c}_- = \frac{1}{n_-} \sum_{i \in \mathcal{I}_-} \mathbf{x}_i$$

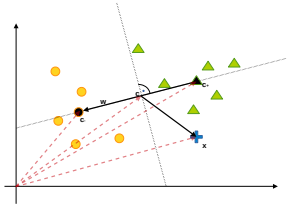
$$\mathbf{y} = \operatorname{sgn}\left(\frac{1}{n_-} \sum_{i \in \mathcal{I}_-} \langle \mathbf{x}_i, \mathbf{x} \rangle - \frac{1}{n_+} \sum_{i \in \mathcal{I}_+} \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$

$$\mathbf{y} = \operatorname{sgn}\left(- \sum_i y_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$

$$\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$







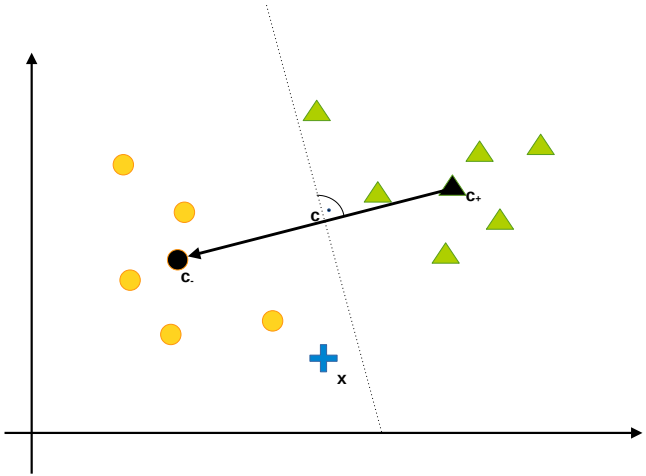
$$\mathbf{y} = \operatorname{sgn}(\langle \mathbf{x}, \mathbf{c}_- \rangle - \langle \mathbf{x}, \mathbf{c}_+ \rangle + b) \qquad \mathbf{c}_- = \frac{1}{n_-} \sum_{i \in \mathcal{I}_-} \mathbf{x}_i$$

$$\mathbf{y} = \operatorname{sgn}\left(\frac{1}{n_-} \sum_{i \in \mathcal{I}_-} \langle \mathbf{x}_i, \mathbf{x} \rangle - \frac{1}{n_+} \sum_{i \in \mathcal{I}_+} \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$

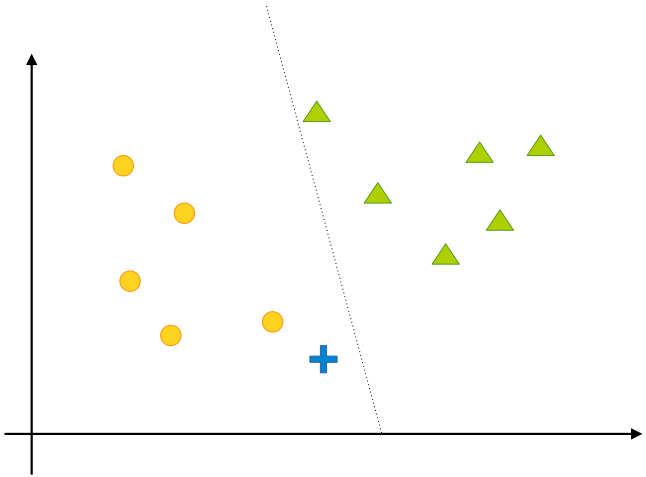
$$\mathbf{y} = \operatorname{sgn}\left(- \sum_i y_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$

$$\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b \right)$$

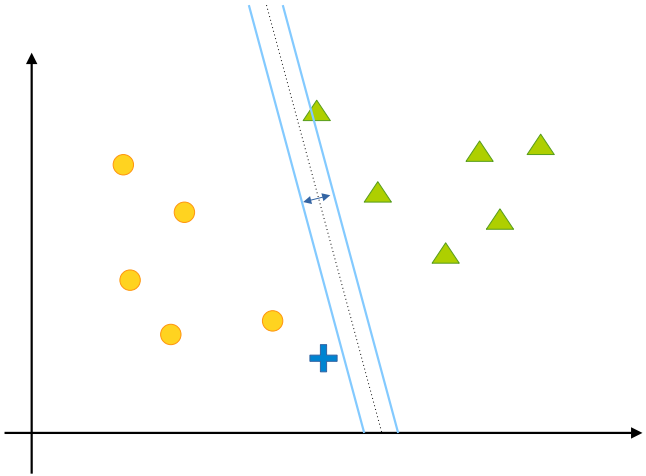
Margines



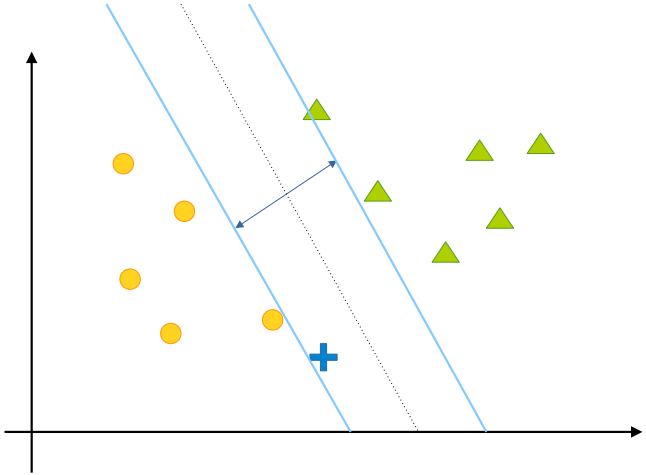
Margines



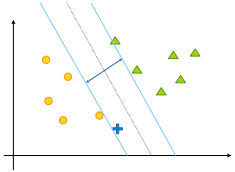
Margines



Margines



SVM

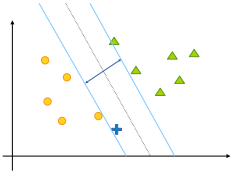


$$\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b\right)$$

$$\underset{\boldsymbol{\alpha} \in \mathbb{R}^n}{\text{maximize}} W(\boldsymbol{\alpha}) = \sum_i \alpha_i - \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle$$

ograniczenia $\alpha_i \geqslant 0, \sum_i \alpha_i y_i = 0$

SVM

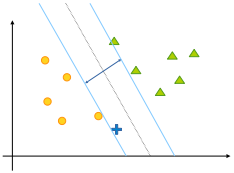


$$\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b\right)$$

zalety Brak [hiper]parametrów!
(Parametry α_i z optymalizacji, ale nie potrzeba np. learning rate, rozmiaru warstw, ...)

wady Tylko dla separowalnych zbiorów danych, których klasy da się rozdzielić hiperpłaszczyzną

SVM

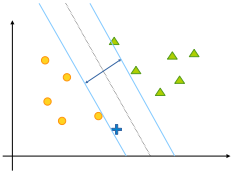


$$\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b\right)$$

zalety Brak [hiper]parametrów!
(Parametry α_i z optymalizacji, ale nie potrzeba np. learning rate, rozmiaru warstw, ...)

wady Tylko dla separowalnych zbiorów danych, których klasy da się rozdzielić hiperpłaszczyzną

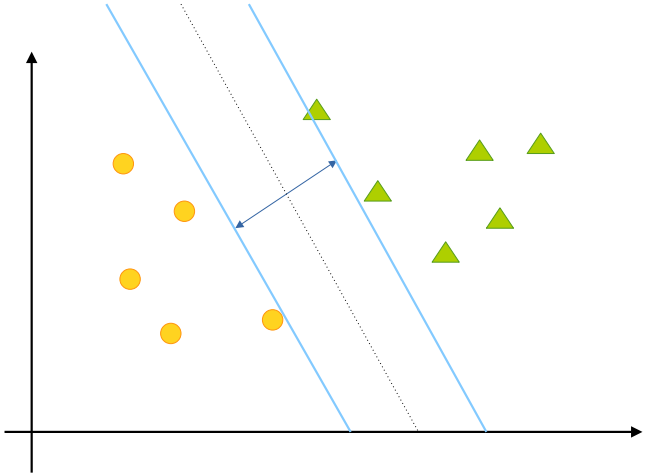
SVM

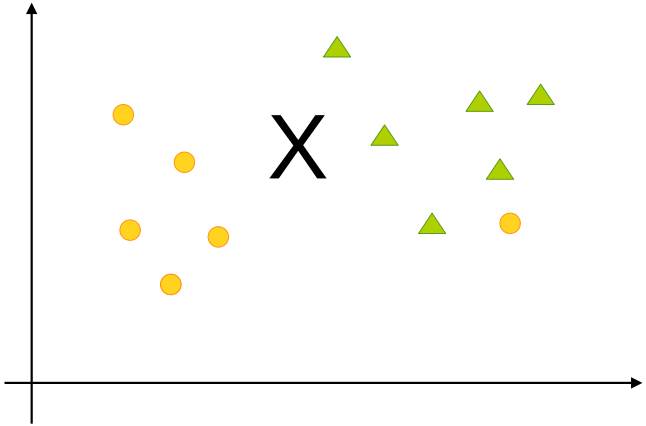


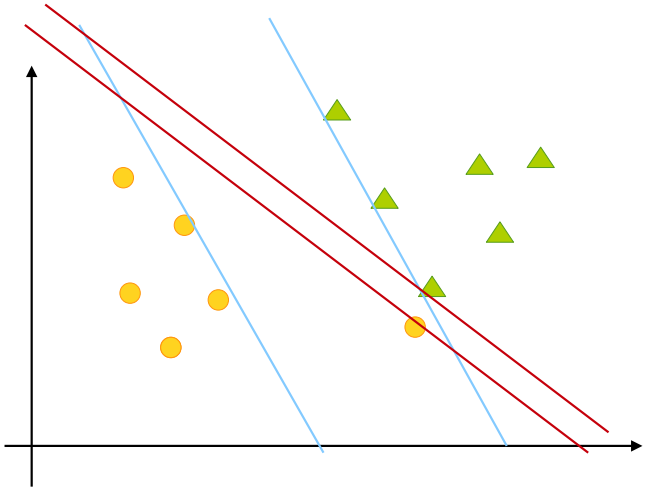
$$\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b\right)$$

zalety Brak [hiper]parametrów!
(Parametry α_i z optymalizacji, ale nie potrzeba np. learning rate, rozmiaru warstw, ...)

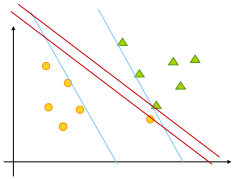
wady Tylko dla separowalnych zbiorów danych, których klasy da się rozdzielić hiperpłaszczyzną







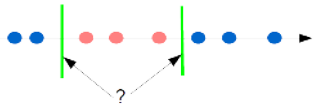
C-SVM



- Wprowadzenie do optymalizacji parametrów ξ_i – błędów klasyfikacji dla konkretnych przykładów ze zbioru treningowego
- Hiperparametr C – sterowanie dopasowania do danych vs wielkość marginesu
- C-SVM może zarówno klasyfikować zbiory niemożliwe do rozdzielenia hiperpłaszczyzną, jak i takie, w których pojedyncze przykłady zmniejszają margines
  - (Wciąż liniowe rozdzielenie → metody nieliniowe)

Metody nieliniowe

Kernel SVM

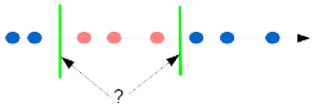


$$x \to x^2$$

Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \to \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \to \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle \dots$ niekoniecznie musimy wyliczać $\Phi(x_i)$!

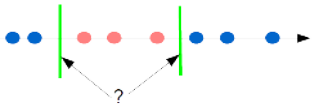


$$x \to x^2$$

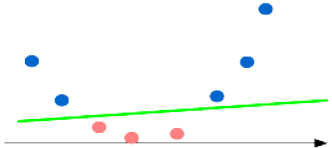
Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \to \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \to \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle \dots$ niekoniecznie musimy wyliczać $\Phi(x_i)$!



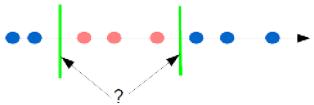
$x \to x^2$



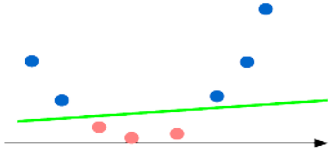
Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \to \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \to \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle \dots$ niekoniecznie musimy wyliczać $\Phi(x_i)$!



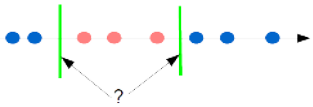
$x \to x^2$



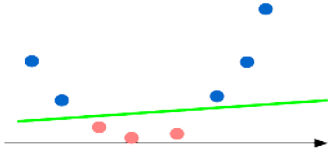
Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \to \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \to \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle \dots$ niekoniecznie musimy wyliczać $\Phi(x_i)$!



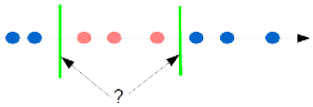
$x \to x^2$



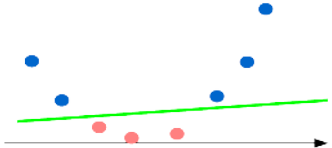
Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \to \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \to \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle \dots$ niekoniecznie musimy wyliczać $\Phi(x_i)$!



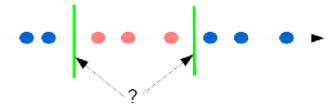
$x \rightarrow x^2$



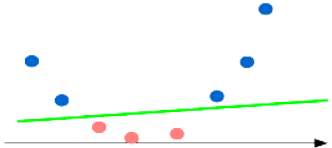
Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \rightarrow \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \rightarrow \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle \dots$ niekoniecznie musimy wyliczać $\Phi(x_i)$!



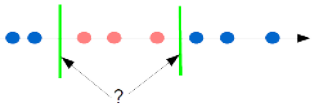
$x \to x^2$



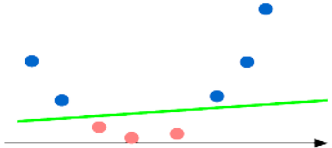
Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \to \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \to \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle \dots$ niekoniecznie musimy wyliczać $\Phi(x_i)$!



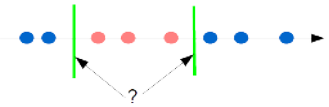
$x \to x^2$



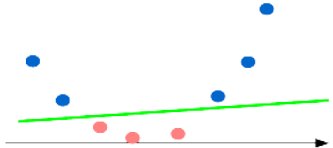
Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \to \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \to \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle \dots$ niekoniecznie musimy wyliczać $\Phi(x_i)$!



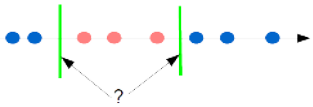
$x \to x^2$



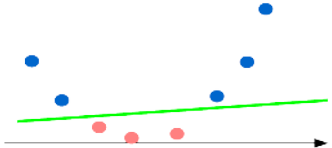
Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \to \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \to \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle \dots$ niekoniecznie musimy wyliczać $\Phi(x_i)$!



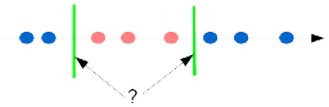
$x \to x^2$



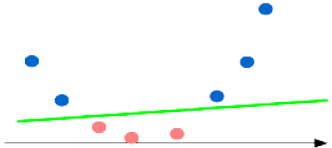
Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \to \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \to \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle$ . . . niekoniecznie musimy wyliczać $\Phi(x_i)$!



$x \to x^2$



Mapujemy dane do innej przestrzeni ($\{x_1, x_2, \dots\} \to \{x_1^2, x_2^2, \dots\}$), mapowanie może być nieliniowe, ale w tej nowej przestrzeni istnieje możliwość liniowego rozdzielenia klas.

W klasyfikatorze SVM, decyzja o klasyfikacji i optymalizacja wykorzystują wyłącznie iloczyn skalarny $\langle \cdot, \cdot \rangle$.

1. Możemy przetwarzać różne dane (nie tylko wektory), o ile dla elementów naszego zbioru danych $x_i, x_j \in \mathcal{X}$ możemy policzyć $\langle x_i, x_j \rangle$
2. Możemy przetwarzać przekształcenia danych $x_i \to \Phi(x_i)$, o ile możemy policzyć $\langle \Phi(x_i), \Phi(x_j) \rangle \dots$ niekoniecznie musimy wyliczać $\Phi(x_i)$!

$\langle x_i, x_j \rangle = x_i x_j$

$\left\langle x_i^2, x_j^2 \right\rangle = x_i^2 x_j^2 = (x_i x_j)^2 = \langle x_i, x_j \rangle^2$

A może przestrzeń funkcji $f, g \in \mathcal{H}, f(x) \in \mathbb{R}, x \in \mathcal{X} \ldots \langle f, g \rangle$?

Rozważmy funkcje w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \qquad (x_i, x_j) \mapsto k(x_i, x_j)$ o cechach:

- symetryczna $k(x_i, x_j) = k(x_j, x_i)$
- dodatnio określona $\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) \geqslant 0 \quad \forall n \in \mathbb{N} \, \forall x_1, \ldots, x_n \in \mathcal{X} \, \forall c_1, \ldots, c_n \in \mathbb{R}$

Na przykład $k(x_i, x_j) = \langle x_i, x_j \rangle^2$:) (a ogólnie $k(x_j, x_i) = (\langle x_i, x_j \rangle + c)^d, c \geqslant 0, d \in \mathbb{N}$)

Dla $x_1 = 42$, możemy zbudować funkcję $k_1(x) = \langle x, 42 \rangle^2 = 1764x^2$

- Dla punktów danych $x_i \in \mathcal{X}$ mamy funkcje $k(x_i, x) \in \mathbb{R}$ – przestrzeń funkcji
- $\langle k(x_i, x), k(x_j, x) \rangle$?

$$\langle x_i, x_j \rangle = x_i x_j$$

$$\left\langle x_i^2, x_j^2 \right\rangle = x_i^2 x_j^2 = (x_i x_j)^2 = \langle x_i, x_j \rangle^2$$

A może przestrzeń funkcji $f, g \in \mathcal{H}, f(x) \in \mathbb{R}, x \in \mathcal{X} \ldots \langle f, g \rangle$?

Rozważmy funkcje w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \qquad (x_i, x_j) \mapsto k(x_i, x_j)$ o cechach:

- symetryczna $k(x_i, x_j) = k(x_j, x_i)$
- dodatnio określona $\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) \geqslant 0 \quad \forall n \in \mathbb{N}\, \forall x_1, \ldots, x_n \in \mathcal{X}\, \forall c_1, \ldots, c_n \in \mathbb{R}$

Na przykład $k(x_i, x_j) = \langle x_i, x_j \rangle^2$:) (a ogólnie $k(x_j, x_i) = (\langle x_i, x_j \rangle + c)^d, c \geqslant 0, d \in \mathbb{N}$)

Dla $x_1 = 42$, możemy zbudować funkcję $k_1(x) = \langle x, 42 \rangle^2 = 1764x^2$

- Dla punktów danych $x_i \in \mathcal{X}$ mamy funkcje $k(x_i, x) \in \mathbb{R}$ – przestrzeń funkcji
- $\langle k(x_i, x), k(x_j, x) \rangle$?

$\langle x_i, x_j \rangle = x_i x_j$

$\left\langle x_i^2, x_j^2 \right\rangle = x_i^2 x_j^2 = (x_i x_j)^2 = \langle x_i, x_j \rangle^2$

A może przestrzeń funkcji $f, g \in \mathcal{H}, f(x) \in \mathbb{R}, x \in \mathcal{X} \ldots \langle f, g \rangle$?

Rozważmy funkcje w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \qquad (x_i, x_j) \mapsto k(x_i, x_j)$ o cechach:

- symetryczna $k(x_i, x_j) = k(x_j, x_i)$
- dodatnio określona $\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) \geqslant 0 \quad \forall n \in \mathbb{N}\, \forall x_1, \ldots, x_n \in \mathcal{X}\, \forall c_1, \ldots, c_n \in \mathbb{R}$

Na przykład $k(x_i, x_j) = \langle x_i, x_j \rangle^2$:) (a ogólnie $k(x_j, x_i) = (\langle x_i, x_j \rangle + c)^d, c \geqslant 0, d \in \mathbb{N}$)

Dla $x_1 = 42$, możemy zbudować funkcję $k_1(x) = \langle x, 42 \rangle^2 = 1764x^2$

- Dla punktów danych $x_i \in \mathcal{X}$ mamy funkcje $k(x_i, x) \in \mathbb{R}$ – przestrzeń funkcji
- $\langle k(x_i, x), k(x_j, x) \rangle$?

$\langle x_i, x_j \rangle = x_i x_j$

$\left\langle x_i^2, x_j^2 \right\rangle = x_i^2 x_j^2 = (x_i x_j)^2 = \langle x_i, x_j \rangle^2$

A może przestrzeń funkcji $f, g \in \mathcal{H}, f(x) \in \mathbb{R}, x \in \mathcal{X} \ldots \langle f, g \rangle$?

Rozważmy funkcje w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \qquad (x_i, x_j) \mapsto k(x_i, x_j)$ o cechach:

- symetryczna $k(x_i, x_j) = k(x_j, x_i)$
- dodatnio określona $\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) \geqslant 0 \quad \forall n \in \mathbb{N}\, \forall x_1, \ldots, x_n \in \mathcal{X}\, \forall c_1, \ldots, c_n \in \mathbb{R}$

Na przykład $k(x_i, x_j) = \langle x_i, x_j \rangle^2$:) (a ogólnie $k(x_j, x_i) = (\langle x_i, x_j \rangle + c)^d, c \geqslant 0, d \in \mathbb{N}$)

Dla $x_1 = 42$, możemy zbudować funkcję $k_1(x) = \langle x, 42 \rangle^2 = 1764x^2$

- Dla punktów danych $x_i \in \mathcal{X}$ mamy funkcje $k(x_i, x) \in \mathbb{R}$ – przestrzeń funkcji
- $\langle k(x_i, x), k(x_j, x) \rangle$?

$\langle x_i, x_j \rangle = x_i x_j$

$\left\langle x_i^2, x_j^2 \right\rangle = x_i^2 x_j^2 = (x_i x_j)^2 = \langle x_i, x_j \rangle^2$

A może przestrzeń funkcji $f, g \in \mathcal{H}, f(x) \in \mathbb{R}, x \in \mathcal{X} \ldots \langle f, g \rangle$?

Rozważmy funkcje w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \qquad (x_i, x_j) \mapsto k(x_i, x_j)$ o cechach:

- symetryczna $k(x_i, x_j) = k(x_j, x_i)$
- dodatnio określona $\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) \geqslant 0 \quad \forall n \in \mathbb{N}\, \forall x_1, \ldots, x_n \in \mathcal{X}\, \forall c_1, \ldots, c_n \in \mathbb{R}$

Na przykład $k(x_i, x_j) = \langle x_i, x_j \rangle^2$:) (a ogólnie $k(x_j, x_i) = (\langle x_i, x_j \rangle + c)^d, c \geqslant 0, d \in \mathbb{N}$)

Dla $x_1 = 42$, możemy zbudować funkcję $k_1(x) = \langle x, 42 \rangle^2 = 1764x^2$

- Dla punktów danych $x_i \in \mathcal{X}$ mamy funkcje $k(x_i, x) \in \mathbb{R}$ – przestrzeń funkcji
- $\langle k(x_i, x), k(x_j, x) \rangle$?

$\langle x_i, x_j \rangle = x_i x_j$

$\left\langle x_i^2, x_j^2 \right\rangle = x_i^2 x_j^2 = (x_i x_j)^2 = \langle x_i, x_j \rangle^2$

A może przestrzeń funkcji $f, g \in \mathcal{H}, f(x) \in \mathbb{R}, x \in \mathcal{X} \ldots \langle f, g \rangle$?

Rozważmy funkcje w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \qquad (x_i, x_j) \mapsto k(x_i, x_j)$ o cechach:

- symetryczna $k(x_i, x_j) = k(x_j, x_i)$
- dodatnio określona $\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) \geqslant 0 \quad \forall n \in \mathbb{N}\, \forall x_1, \ldots, x_n \in \mathcal{X}\, \forall c_1, \ldots, c_n \in \mathbb{R}$

Na przykład $k(x_i, x_j) = \langle x_i, x_j \rangle^2$:) (a ogólnie $k(x_j, x_i) = (\langle x_i, x_j \rangle + c)^d, c \geqslant 0, d \in \mathbb{N}$)

Dla $x_1 = 42$, możemy zbudować funkcję $k_1(x) = \langle x, 42 \rangle^2 = 1764x^2$

- Dla punktów danych $x_i \in \mathcal{X}$ mamy funkcje $k(x_i, x) \in \mathbb{R}$ – przestrzeń funkcji
- $\langle k(x_i, x), k(x_j, x) \rangle$?

$\langle x_i, x_j \rangle = x_i x_j$

$\left\langle x_i^2, x_j^2 \right\rangle = x_i^2 x_j^2 = (x_i x_j)^2 = \langle x_i, x_j \rangle^2$

A może przestrzeń funkcji $f, g \in \mathcal{H}, f(x) \in \mathbb{R}, x \in \mathcal{X} \ldots \langle f, g \rangle$?

Rozważmy funkcje w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \qquad (x_i, x_j) \mapsto k(x_i, x_j)$ o cechach:

- symetryczna $k(x_i, x_j) = k(x_j, x_i)$
- dodatnio określona $\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) \geqslant 0 \quad \forall n \in \mathbb{N}\, \forall x_1, \ldots, x_n \in \mathcal{X}\, \forall c_1, \ldots, c_n \in \mathbb{R}$

Na przykład $k(x_i, x_j) = \langle x_i, x_j \rangle^2$:) (a ogólnie $k(x_j, x_i) = (\langle x_i, x_j \rangle + c)^d, c \geqslant 0, d \in \mathbb{N}$)

Dla $x_1 = 42$, możemy zbudować funkcję $k_1(x) = \langle x, 42 \rangle^2 = 1764x^2$

- Dla punktów danych $x_i \in \mathcal{X}$ mamy funkcje $k(x_i, x) \in \mathbb{R}$ – przestrzeń funkcji
- $\langle k(x_i, x), k(x_j, x) \rangle$?

$\langle k(x_i, x), k(x_j, x) \rangle$?

Jeżeli funkcja w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \quad (x_i, x_j) \mapsto k(x_i, x_j)$ jest symetryczna i dodatnio określona, to (twierdzenie Moore'a–Aronszajna) istnieje przestrzeń Hilberta funkcji określonych na $\mathcal{X}$ w której k jest reprodukującym jądrem (reproducing kernel).

- Przestrzeń Hilberta – przestrzeń liniowa (dodawanie, mnożenie przez skalar) + zdefiniowany iloczyn skalarny który indukuje metrykę (możemy wyznaczyć odległość między elementami zbioru)
- Przestrzeń Hilberta z reprodukującym jądrem (RKHS) – dla każdego $x_i \in \mathcal{X}$ istnieje funkcja $k_{x_i} \in \mathcal{H}$ taka że dla każdej funkcji z tej przestrzeni $f \in \mathcal{H}$ zachodzi $\langle f, k_{x_i} \rangle = f(x_i)$

$\langle k(x_i, x), k(x_j, x) \rangle = k(x_i, x_j)$

$\langle k(x_i, x), k(x_j, x) \rangle$?

Jeżeli funkcja w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \quad (x_i, x_j) \mapsto k(x_i, x_j)$ jest symetryczna i dodatnio określona, to (twierdzenie Moore'a–Aronszajna) istnieje przestrzeń Hilberta funkcji określonych na $\mathcal{X}$ w której k jest reprodukującym jądrem (reproducing kernel).

- Przestrzeń Hilberta – przestrzeń liniowa (dodawanie, mnożenie przez skalar) + zdefiniowany iloczyn skalarny który indukuje metrykę (możemy wyznaczyć odległość między elementami zbioru)
- Przestrzeń Hilberta z reprodukującym jądrem (RKHS) – dla każdego $x_i \in \mathcal{X}$ istnieje funkcja $k_{x_i} \in \mathcal{H}$ taka że dla każdej funkcji z tej przestrzeni $f \in \mathcal{H}$ zachodzi $\langle f, k_{x_i} \rangle = f(x_i)$

$\langle k(x_i, x), k(x_j, x) \rangle = k(x_i, x_j)$

$\langle k(x_i, x), k(x_j, x) \rangle$?

Jeżeli funkcja w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \quad (x_i, x_j) \mapsto k(x_i, x_j)$ jest symetryczna i dodatnio określona, to (twierdzenie Moore'a–Aronszajna) istnieje przestrzeń Hilberta funkcji określonych na $\mathcal{X}$ w której k jest reprodukującym jądrem (reproducing kernel).

- Przestrzeń Hilberta – przestrzeń liniowa (dodawanie, mnożenie przez skalar) + zdefiniowany iloczyn skalarny który indukuje metrykę (możemy wyznaczyć odległość między elementami zbioru)
- Przestrzeń Hilberta z reprodukującym jądrem (RKHS) – dla każdego $x_i \in \mathcal{X}$ istnieje funkcja $k_{x_i} \in \mathcal{H}$ taka że dla każdej funkcji z tej przestrzeni $f \in \mathcal{H}$ zachodzi $\langle f, k_{x_i} \rangle = f(x_i)$

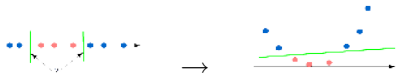
$\langle k(x_i, x), k(x_j, x) \rangle = k(x_i, x_j)$

$\langle k(x_i, x), k(x_j, x) \rangle$?

Jeżeli funkcja w postaci $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \quad (x_i, x_j) \mapsto k(x_i, x_j)$ jest symetryczna i dodatnio określona, to (twierdzenie Moore'a–Aronszajna) istnieje przestrzeń Hilberta funkcji określonych na $\mathcal{X}$ w której k jest reprodukującym jądrem (reproducing kernel).

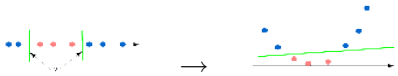
- Przestrzeń Hilberta – przestrzeń liniowa (dodawanie, mnożenie przez skalar) + zdefiniowany iloczyn skalarny który indukuje metrykę (możemy wyznaczyć odległość między elementami zbioru)
- Przestrzeń Hilberta z reprodukującym jądrem (RKHS) – dla każdego $x_i \in \mathcal{X}$ istnieje funkcja $k_{x_i} \in \mathcal{H}$ taka że dla każdej funkcji z tej przestrzeni $f \in \mathcal{H}$ zachodzi $\langle f, k_{x_i} \rangle = f(x_i)$

$\langle k(x_i, x), k(x_j, x) \rangle = k(x_i, x_j)$



Podsumowanie:

1. Mamy zbiór danych $\mathcal{X}$, o złożonej strukturze (nieliniowa granica decyzyjna)
2. Mamy narzędzie liniowej klasyfikacji, oparte o iloczyn skalarny
3. Szukamy nieliniowego przekształcenia danych do innej przestrzeni, gdzie możemy liniowo rozdzielić klasy
4. Jeżeli znajdziemy dobrą funkcję k, to dzięki własnościom RKHS możemy odwzorować dane w innej przestrzeni $\mathcal{H}$, opartej o k, w której wyliczymy iloczyn skalarny jako $k(x_i, x_j)$
5. Ponieważ odwzorowanie jest nieliniowe, liniowe rozdzielenie w $\mathcal{H}$ przekłada się na nieliniową granicę decyzyjną w $\mathcal{X}$

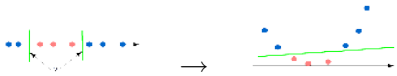


SVM $\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b\right)$ $\quad \rightarrow \quad$ kernel-SVM $\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b\right)$

Popularne funkcje k:

- RBF $k(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$
- Wielomianowa $k(\mathbf{x}_i, \mathbf{x}_j) = \left(\langle \mathbf{x}_i, \mathbf{x}_j \rangle + c\right)^d, c \geqslant 0, d \in \mathbb{N}$
- Sigmoid $k(\mathbf{x}_i, \mathbf{x}_j) = \ldots$

Najczęściej wystarczy jedna dobra funkcja – RBF

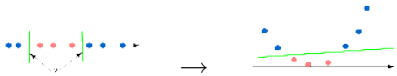


SVM $\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i \langle \mathbf{x}_i, \mathbf{x} \rangle + b\right)$ $\quad \rightarrow \quad$ kernel-SVM $\mathbf{y} = \operatorname{sgn}\left(\sum_i \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b\right)$

Popularne funkcje k:

- RBF $k(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2}$
- Wielomianowa $k(\mathbf{x}_i, \mathbf{x}_j) = \left(\langle \mathbf{x}_i, \mathbf{x}_j \rangle + c\right)^d, c \geqslant 0, d \in \mathbb{N}$
- Sigmoid $k(\mathbf{x}_i, \mathbf{x}_j) = \ldots$

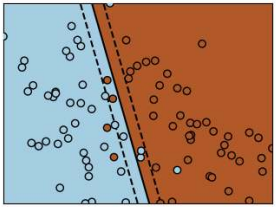
Najczęściej wystarczy jedna dobra funkcja – RBF

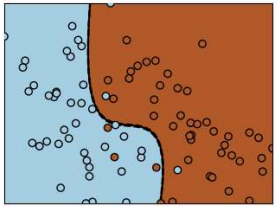


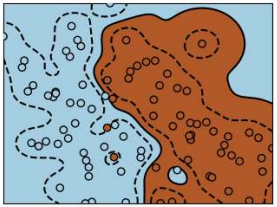
The idea of SVMs: map the training data into a higher-dimensional feature space via Φ, and construct a separating hyperplane with maximum margin there. This yields a nonlinear decision boundary in input space. By the use of a kernel function, it is possible to compute the separating hyperplane without explicitly carrying out the map into the feature space

Learning with Kernels, str. 29

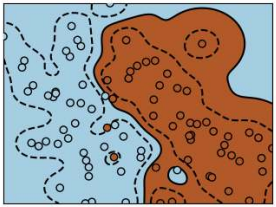
Przykłady funkcji jądrowych: liniowa, wielomianowa, RBF

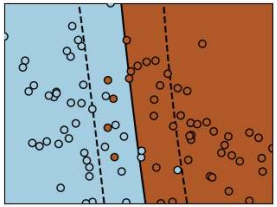


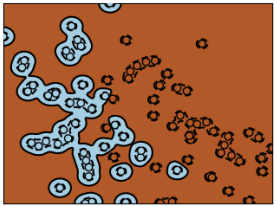




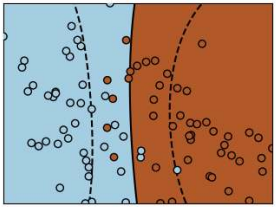
Przykłady funkcji jądrowej RBF dla różnych wartości parametru γ,
kolejno wartości γ: dopasowana do zbioru danych, niska, wysoka

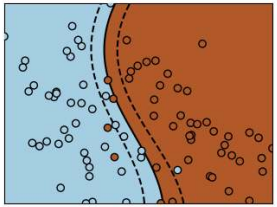


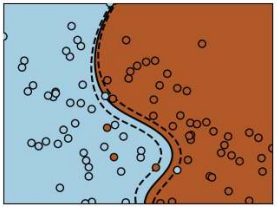




Przykłady funkcji jądrowej RBF dla dopasowanej do danych wartości γ, kolejno wartości C: niska, dopasowana, wysoka

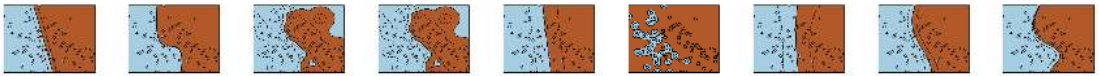






Kernel SVM:

- Skuteczny w praktyce klasyfikator
- Bardzo dobrze opracowany teoretycznie
- Dwa – tylko dwa – hiperparametry, o łatwej interpretacji
- „Kernel trick" – możliwość uogólnienia algorytmów dających się sformułować w oparciu o iloczyn skalarny (np. PCA → kernel PCA)



Pomiar wydajności klasyfikatorów

- Skąd wiemy, jak dobry jest klasyfikator?
- Kiedy jeden klasyfikator jest lepszy od drugiego?
- Jak diagnozować problem z wydajnością/skutecznością?

Miary wydajności

- Skuteczność (accuracy) $\frac{T}{T+F}$
  - Intuicyjna, naturalna
  - Wrażliwa na różną liczebność klas
  - Wrażliwa na rodzaj błędów
- Macierz błędów (confusion matrix)

	Predicted Positive	Predicted Negative
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

- Precison $\frac{TP}{TP+FP}$ oraz recall $\frac{TP}{TP+FN}$
- Sensitivity (czułość) $\frac{TP}{TP+FN}$ oraz specificity (swoistość) $\frac{TN}{TN+FP}$
- True positive ratio $\frac{TP}{TP+FN}$ oraz false positive ratio $\frac{FP}{FP+TN}$
- F1 score $\frac{2}{p^{-1}+r^{-1}} = 2\frac{p \times r}{p+r} = \frac{2TP}{2TP+FP+FN}$

- Skuteczność (accuracy) $\frac{T}{T+F}$
  - Intuicyjna, naturalna
  - Wrażliwa na różną liczebność klas
  - Wrażliwa na rodzaj błędów
- Macierz błędów (confusion matrix)

	Predicted Positive	Predicted Negative
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

- Precison $\frac{TP}{TP+FP}$ oraz recall $\frac{TP}{TP+FN}$
- Sensitivity (czułość) $\frac{TP}{TP+FN}$ oraz specificity (swoistość) $\frac{TN}{TN+FP}$
- True positive ratio $\frac{TP}{TP+FN}$ oraz false positive ratio $\frac{FP}{FP+TN}$
- F1 score $\frac{2}{p^{-1}+r^{-1}} = 2\frac{p \times r}{p+r} = \frac{2TP}{2TP+FP+FN}$

- Skuteczność (accuracy) $\frac{T}{T+F}$
  - Intuicyjna, naturalna
  - Wrażliwa na różną liczebność klas
  - Wrażliwa na rodzaj błędów
- Macierz błędów (confusion matrix)

	Predicted Positive	Predicted Negative
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

- Precison $\frac{TP}{TP+FP}$ oraz recall $\frac{TP}{TP+FN}$
- Sensitivity (czułość) $\frac{TP}{TP+FN}$ oraz specificity (swoistość) $\frac{TN}{TN+FP}$
- True positive ratio $\frac{TP}{TP+FN}$ oraz false positive ratio $\frac{FP}{FP+TN}$
- F1 score $\frac{2}{p^{-1}+r^{-1}} = 2\frac{p \times r}{p+r} = \frac{2TP}{2TP+FP+FN}$

- Skuteczność (accuracy) $\frac{T}{T+F}$
  - Intuicyjna, naturalna
  - Wrażliwa na różną liczebność klas
  - Wrażliwa na rodzaj błędów
- Macierz błędów (confusion matrix)

	Predicted Positive	Predicted Negative
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

- Precison $\frac{TP}{TP+FP}$ oraz recall $\frac{TP}{TP+FN}$
- Sensitivity (czułość) $\frac{TP}{TP+FN}$ oraz specificity (swoistość) $\frac{TN}{TN+FP}$
- True positive ratio $\frac{TP}{TP+FN}$ oraz false positive ratio $\frac{FP}{FP+TN}$
- F1 score $\frac{2}{p^{-1}+r^{-1}} = 2\frac{p \times r}{p+r} = \frac{2TP}{2TP+FP+FN}$

- Skuteczność (accuracy) $\frac{T}{T+F}$
  - Intuicyjna, naturalna
  - Wrażliwa na różną liczebność klas
  - Wrażliwa na rodzaj błędów
- Macierz błędów (confusion matrix)

	Predicted Positive	**Predicted Negative**
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

- Precison $\frac{TP}{TP+FP}$ oraz recall $\frac{TP}{TP+FN}$
- Sensitivity (czułość) $\frac{TP}{TP+FN}$ oraz specificity (swoistość) $\frac{TN}{TN+FP}$
- True positive ratio $\frac{TP}{TP+FN}$ oraz false positive ratio $\frac{FP}{FP+TN}$
- F1 score $\frac{2}{p^{-1}+r^{-1}} = 2\frac{p \times r}{p+r} = \frac{2TP}{2TP+FP+FN}$

- Skuteczność (accuracy) $\frac{T}{T+F}$ $\frac{TP+TN}{TP+FN+FP+TN}$
  - Intuicyjna, naturalna
  - Wrażliwa na różną liczebność klas
  - Wrażliwa na rodzaj błędów
- Macierz błędów (confusion matrix)

	Predicted Positive	**Predicted Negative**
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

- Precison $\frac{TP}{TP+FP}$ oraz recall $\frac{TP}{TP+FN}$
- Sensitivity (czułość) $\frac{TP}{TP+FN}$ oraz specificity (swoistość) $\frac{TN}{TN+FP}$
- True positive ratio $\frac{TP}{TP+FN}$ oraz false positive ratio $\frac{FP}{FP+TN}$
- F1 score $\frac{2}{p^{-1}+r^{-1}} = 2\frac{p \times r}{p+r} = \frac{2TP}{2TP+FP+FN}$

- Skuteczność (accuracy) $\frac{T}{T+F}$ $\frac{TP+TN}{TP+FN+FP+TN}$
  - Intuicyjna, naturalna
  - Wrażliwa na różną liczebność klas
  - Wrażliwa na rodzaj błędów
- Macierz błędów (confusion matrix)

	Predicted Positive	**Predicted Negative**
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

- Precison $\frac{TP}{TP+FP}$ oraz recall $\frac{TP}{TP+FN}$
- Sensitivity (czułość) $\frac{TP}{TP+FN}$ oraz specificity (swoistość) $\frac{TN}{TN+FP}$
- True positive ratio $\frac{TP}{TP+FN}$ oraz false positive ratio $\frac{FP}{FP+TN}$
- F1 score $\frac{2}{p^{-1}+r^{-1}} = 2\frac{p \times r}{p+r} = \frac{2TP}{2TP+FP+FN}$

- Skuteczność (accuracy) $\frac{T}{T+F}$ $\frac{TP+TN}{TP+FN+FP+TN}$
  - Intuicyjna, naturalna
  - Wrażliwa na różną liczebność klas
  - Wrażliwa na rodzaj błędów
- Macierz błędów (confusion matrix)

	Predicted Positive	**Predicted Negative**
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

- Precison $\frac{TP}{TP+FP}$ oraz recall $\frac{TP}{TP+FN}$
- Sensitivity (czułość) $\frac{TP}{TP+FN}$ oraz specificity (swoistość) $\frac{TN}{TN+FP}$
- True positive ratio $\frac{TP}{TP+FN}$ oraz false positive ratio $\frac{FP}{FP+TN}$
- F1 score $\frac{2}{p^{-1}+r^{-1}} = 2\frac{p \times r}{p+r} = \frac{2TP}{2TP+FP+FN}$

- Skuteczność (accuracy) $\frac{T}{T+F}$ $\frac{TP+TN}{TP+FN+FP+TN}$
  - Intuicyjna, naturalna
  - Wrażliwa na różną liczebność klas
  - Wrażliwa na rodzaj błędów
- Macierz błędów (confusion matrix)

	Predicted Positive	**Predicted Negative**
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

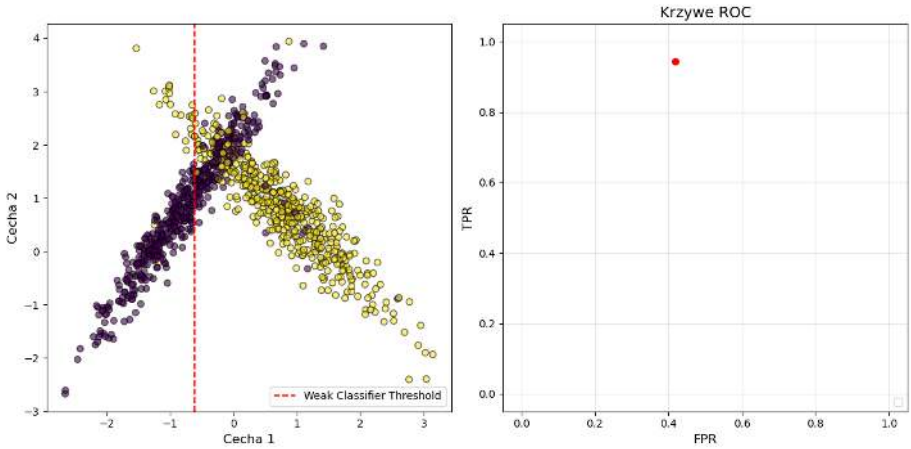
- Precison $\frac{TP}{TP+FP}$ oraz recall $\frac{TP}{TP+FN}$
- Sensitivity (czułość) $\frac{TP}{TP+FN}$ oraz specificity (swoistość) $\frac{TN}{TN+FP}$
- True positive ratio $\frac{TP}{TP+FN}$ oraz false positive ratio $\frac{FP}{FP+TN}$
- F1 score $\frac{2}{p^{-1}+r^{-1}} = 2\frac{p \times r}{p+r} = \frac{2TP}{2TP+FP+FN}$

Jeden klasyfikator, wydajność – niezależnie od hiperparametrów?

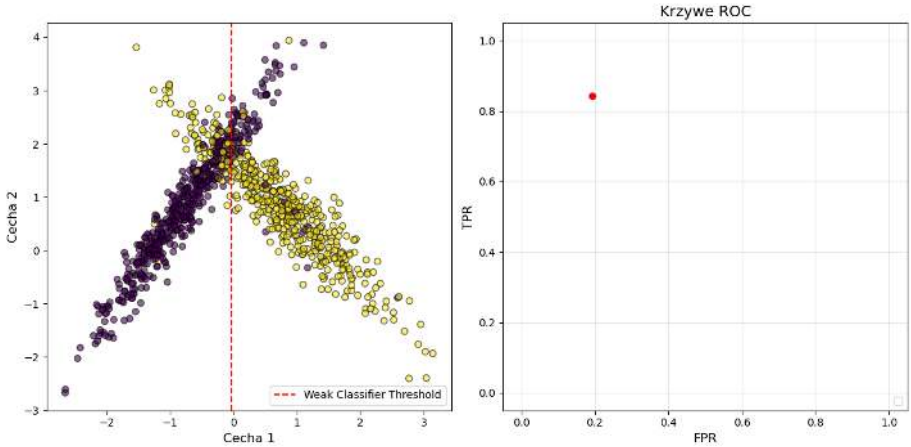
Prosty klasyfikator (średnie klas), progowanie szacowanego prawopodobieństwa $T = 0.1$

Jeden klasyfikator, wydajność – niezależnie od hiperparametrów?

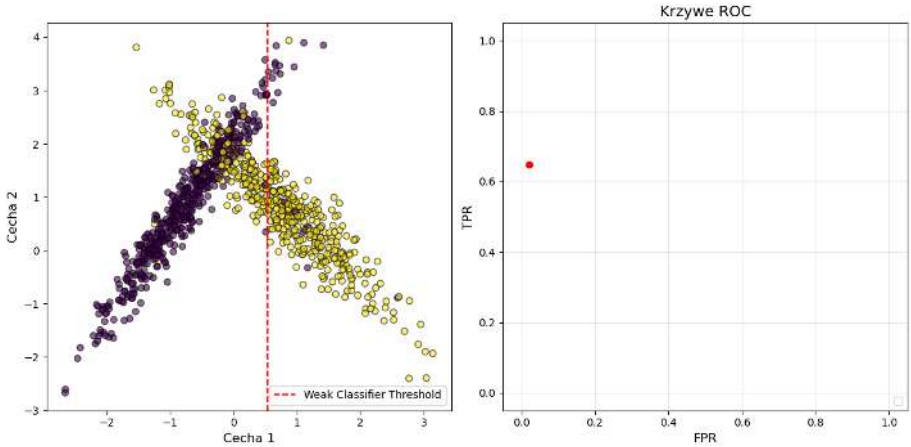
Prosty klasyfikator (średnie klas), progowanie szacowanego prawopodobieństwa $T = 0.1$



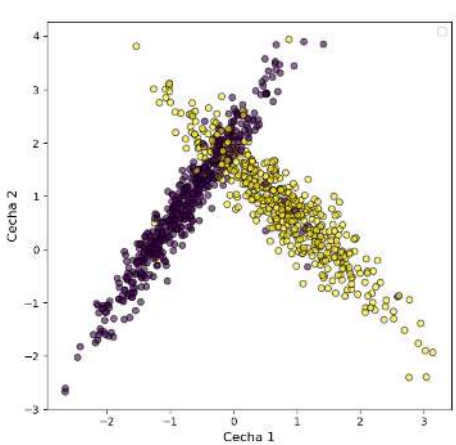
Prosty klasyfikator (średnie klas), progowanie szacowanego prawopodobieństwa $T = 0.5$

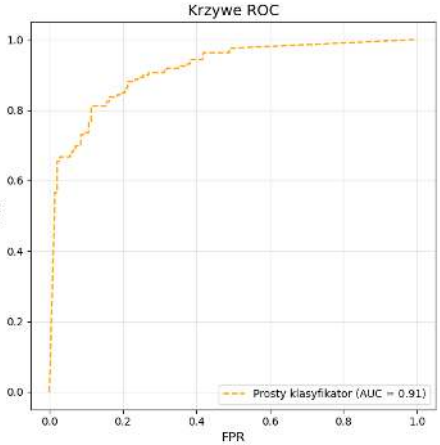


Prosty klasyfikator (średnie klas), progowanie szacowanego prawopodobieństwa $T = 0.9$

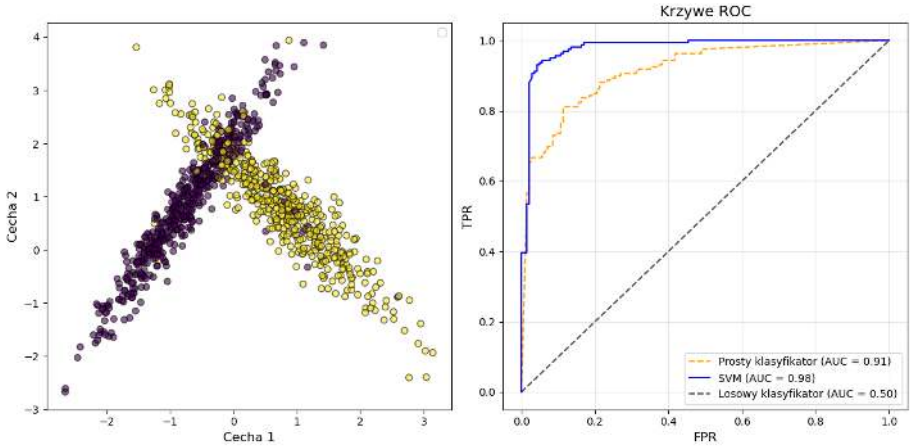


Prosty klasyfikator (średnie klas), progowanie szacowanego prawopodobieństwa – krzywa charakterystyki operatora (receiver operator characteristic, ROC)





Prosty klasyfikator vs SVM – krzywe ROC



Dwa klasyfikatory, wydajność, porównanie?

Np. 88% vs 89% - a co jeżeli to przypadek, i na innej części zbioru danych będzie 89% vs 88%?

(alternatywnie – który zestaw hiperparametrów wybrać?)

1. Testy statystyczne (→ wykłady dr hab. Hanna Wojewódka-Ściążko)
2. Pary wyników (dla konkretnego punktu danych wyniki obu klasyfikatorów) – różnice
3. Czy mają rozkład normalny (test Shapiro-Wilk)?
4. Jeżeli tak (r.n.) – t-test dla par
   - Hipoteza zerowa tego testu mówi o równych średnich, a alternatywna o różnych, zatem odrzucenie hipotezy zerowej świadczy o tym, że wyniki jednej z metod są istotnie statystycznie różne od drugiej
   - Jednostronnie lub dwustronnie
5. Jeżeli nie (r.n.) – Wilcoxon signed-rank test
6. Testy pozwalają określić szansę (przez p-wartość) odpowiedniości lub przewagi klasyfikatorów

Dwa klasyfikatory, wydajność, porównanie?

Np. 88% vs 89% - a co jeżeli to przypadek, i na innej części zbioru danych będzie 89% vs 88%?

(alternatywnie – który zestaw hiperparametrów wybrać?)

1. Testy statystyczne (→ wykłady dr hab. Hanna Wojewódka-Ściążko)
2. Pary wyników (dla konkretnego punktu danych wyniki obu klasyfikatorów) – różnice
3. Czy mają rozkład normalny (test Shapiro-Wilk)?
4. Jeżeli tak (r.n.) – t-test dla par
   - Hipoteza zerowa tego testu mówi o równych średnich, a alternatywna o różnych, zatem odrzucenie hipotezy zerowej świadczy o tym, że wyniki jednej z metod są istotnie statystycznie różne od drugiej
   - Jednostronnie lub dwustronnie
5. Jeżeli nie (r.n.) – Wilcoxon signed-rank test
6. Testy pozwalają określić szansę (przez p-wartość) odpowiedniości lub przewagi klasyfikatorów

Dwa klasyfikatory, wydajność, porównanie?

Np. 88% vs 89% - a co jeżeli to przypadek, i na innej części zbioru danych będzie 89% vs 88%?

(alternatywnie – który zestaw hiperparametrów wybrać?)

1. Testy statystyczne (→ wykłady dr hab. Hanna Wojewódka-Ściążko)
2. Pary wyników (dla konkretnego punktu danych wyniki obu klasyfikatorów) – różnice
3. Czy mają rozkład normalny (test Shapiro-Wilk)?
4. Jeżeli tak (r.n.) – t-test dla par
   - Hipoteza zerowa tego testu mówi o równych średnich, a alternatywna o różnych, zatem odrzucenie hipotezy zerowej świadczy o tym, że wyniki jednej z metod są istotnie statystycznie różne od drugiej
   - Jednostronnie lub dwustronnie
5. Jeżeli nie (r.n.) – Wilcoxon signed-rank test
6. Testy pozwalają określić szansę (przez p-wartość) odpowiedniości lub przewagi klasyfikatorów

Dwa klasyfikatory, wydajność, porównanie?

Np. 88% vs 89% - a co jeżeli to przypadek, i na innej części zbioru danych będzie 89% vs 88%?

(alternatywnie – który zestaw hiperparametrów wybrać?)

1. Testy statystyczne (→ wykłady dr hab. Hanna Wojewódka-Ściążko)
2. Pary wyników (dla konkretnego punktu danych wyniki obu klasyfikatorów) – różnice
3. Czy mają rozkład normalny (test Shapiro-Wilk)?
4. Jeżeli tak (r.n.) – t-test dla par
   - Hipoteza zerowa tego testu mówi o równych średnich, a alternatywna o różnych, zatem odrzucenie hipotezy zerowej świadczy o tym, że wyniki jednej z metod są istotnie statystycznie różne od drugiej
   - Jednostronnie lub dwustronnie
5. Jeżeli nie (r.n.) – Wilcoxon signed-rank test
6. Testy pozwalają określić szansę (przez p-wartość) odpowiedniości lub przewagi klasyfikatorów

Dwa klasyfikatory, wydajność, porównanie?

Np. 88% vs 89% - a co jeżeli to przypadek, i na innej części zbioru danych będzie 89% vs 88%?

(alternatywnie – który zestaw hiperparametrów wybrać?)

1. Testy statystyczne (→ wykłady dr hab. Hanna Wojewódka-Ściążko)
2. Pary wyników (dla konkretnego punktu danych wyniki obu klasyfikatorów) – różnice
3. Czy mają rozkład normalny (test Shapiro-Wilk)?
4. Jeżeli tak (r.n.) – t-test dla par
   - Hipoteza zerowa tego testu mówi o równych średnich, a alternatywna o różnych, zatem odrzucenie hipotezy zerowej świadczy o tym, że wyniki jednej z metod są istotnie statystycznie różne od drugiej
   - Jednostronnie lub dwustronnie
5. Jeżeli nie (r.n.) – Wilcoxon signed-rank test
6. Testy pozwalają określić szansę (przez p-wartość) odpowiedniości lub przewagi klasyfikatorów

Dwa klasyfikatory, wydajność, porównanie?

Np. 88% vs 89% - a co jeżeli to przypadek, i na innej części zbioru danych będzie 89% vs 88%?

(alternatywnie – który zestaw hiperparametrów wybrać?)

1. Testy statystyczne (→ wykłady dr hab. Hanna Wojewódka-Ściążko)
2. Pary wyników (dla konkretnego punktu danych wyniki obu klasyfikatorów) – różnice
3. Czy mają rozkład normalny (test Shapiro-Wilk)?
4. Jeżeli tak (r.n.) – t-test dla par
   - Hipoteza zerowa tego testu mówi o równych średnich, a alternatywna o różnych, zatem odrzucenie hipotezy zerowej świadczy o tym, że wyniki jednej z metod są istotnie statystycznie różne od drugiej
   - Jednostronnie lub dwustronnie
5. Jeżeli nie (r.n.) – Wilcoxon signed-rank test
6. Testy pozwalają określić szansę (przez p-wartość) odpowiedniości lub przewagi klasyfikatorów

Dwa klasyfikatory, wydajność, porównanie?

Np. 88% vs 89% - a co jeżeli to przypadek, i na innej części zbioru danych będzie 89% vs 88%?

(alternatywnie – który zestaw hiperparametrów wybrać?)

1. Testy statystyczne (→ wykłady dr hab. Hanna Wojewódka-Ściążko)
2. Pary wyników (dla konkretnego punktu danych wyniki obu klasyfikatorów) – różnice
3. Czy mają rozkład normalny (test Shapiro-Wilk)?
4. Jeżeli tak (r.n.) – t-test dla par
   - Hipoteza zerowa tego testu mówi o równych średnich, a alternatywna o różnych, zatem odrzucenie hipotezy zerowej świadczy o tym, że wyniki jednej z metod są istotnie statystycznie różne od drugiej
   - Jednostronnie lub dwustronnie
5. Jeżeli nie (r.n.) – Wilcoxon signed-rank test
6. Testy pozwalają określić szansę (przez p-wartość) odpowiedniości lub przewagi klasyfikatorów

Dwa klasyfikatory, wydajność, porównanie?

Np. 88% vs 89% - a co jeżeli to przypadek, i na innej części zbioru danych będzie 89% vs 88%?

(alternatywnie – który zestaw hiperparametrów wybrać?)

1. Testy statystyczne (→ wykłady dr hab. Hanna Wojewódka-Ściążko)
2. Pary wyników (dla konkretnego punktu danych wyniki obu klasyfikatorów) – różnice
3. Czy mają rozkład normalny (test Shapiro-Wilk)?
4. Jeżeli tak (r.n.) – t-test dla par
   - Hipoteza zerowa tego testu mówi o równych średnich, a alternatywna o różnych, zatem odrzucenie hipotezy zerowej świadczy o tym, że wyniki jednej z metod są istotnie statystycznie różne od drugiej
   - Jednostronnie lub dwustronnie
5. Jeżeli nie (r.n.) – Wilcoxon signed-rank test
6. Testy pozwalają określić szansę (przez p-wartość) odpowiedniości lub przewagi klasyfikatorów

Dwa klasyfikatory, wydajność, porównanie?

Np. 88% vs 89% - a co jeżeli to przypadek, i na innej części zbioru danych będzie 89% vs 88%?

(alternatywnie – który zestaw hiperparametrów wybrać?)

1. Testy statystyczne (→ wykłady dr hab. Hanna Wojewódka-Ściążko)
2. Pary wyników (dla konkretnego punktu danych wyniki obu klasyfikatorów) – różnice
3. Czy mają rozkład normalny (test Shapiro-Wilk)?
4. Jeżeli tak (r.n.) – t-test dla par
   - Hipoteza zerowa tego testu mówi o równych średnich, a alternatywna o różnych, zatem odrzucenie hipotezy zerowej świadczy o tym, że wyniki jednej z metod są istotnie statystycznie różne od drugiej
   - Jednostronnie lub dwustronnie
5. Jeżeli nie (r.n.) – Wilcoxon signed-rank test
6. Testy pozwalają określić szansę (przez p-wartość) odpowiedniości lub przewagi klasyfikatorów

Walidacja krzyżowa

Klasyfikator, skuteczność 89% . . . oczekiwania? Spodziewany wynik „w działaniu", najlepsze możliwe oszacowanie, prawdziwa skuteczność klasyfikacji, powtarzalność, uogólnianie. . .

- Zbiór danych – reprezentatywny dla problemu
- Wydzielenie zbioru testowego:
  - Brak – oszacowanie obciążone (skuteczność zawyżona)
  - 50% trening, 50% test – oszacowanie obciążone (skuteczność zaniżona)
  - Wydzielenie pojedynczego przykładu (leave one out) + powtarzanie – duża wariancja, złożone obliczeniowo
    - Ale – leave one patient out!
  - Wydzielenie części (np. 5, 10, 20%) i powtórzenie – walidacja krzyżowa, optymalne
  - Wydzielenie zbioru testowego – niezależna weryfikacja wydajności
- Wydzielenie zbioru walidacyjnego
  - Skąd wiemy jakie hiperparametry ustawić?
  - Wewnętrzna pętla walidacji żeby ustalić wartości hiperparametrów
  - Dwa poziomy walidacji (zewnętrzna – zbiór testowy, wewnętrzna – zbiór walidacyjny) – niezależna weryfikacja wydajności uwzględniająca niepewność doboru hiperparametrów
    - Testowanie metody vs testowanie metody z hiperparametrami dobranymi przez eksperta

Klasyfikator, skuteczność 89% . . . oczekiwania? Spodziewany wynik „w działaniu", najlepsze możliwe oszacowanie, prawdziwa skuteczność klasyfikacji, powtarzalność, uogólnianie. . .

- Zbiór danych – reprezentatywny dla problemu
- Wydzielenie zbioru testowego:
  - Brak – oszacowanie obciążone (skuteczność zawyżona)
  - 50% trening, 50% test – oszacowanie obciążone (skuteczność zaniżona)
  - Wydzielenie pojedynczego przykładu (leave one out) + powtarzanie – duża wariancja, złożone obliczeniowo
    - Ale – leave one patient out!
  - Wydzielenie części (np. 5, 10, 20%) i powtórzenie – walidacja krzyżowa, optymalne
  - Wydzielenie zbioru testowego – niezależna weryfikacja wydajności
- Wydzielenie zbioru walidacyjnego
  - Skąd wiemy jakie hiperparametry ustawić?
  - Wewnętrzna pętla walidacji żeby ustalić wartości hiperparametrów
  - Dwa poziomy walidacji (zewnętrzna – zbiór testowy, wewnętrzna – zbiór walidacyjny) – niezależna weryfikacja wydajności uwzględniająca niepewność doboru hiperparametrów
    - Testowanie metody vs testowanie metody z hiperparametrami dobranymi przez eksperta

Klasyfikator, skuteczność 89% . . . oczekiwania? Spodziewany wynik „w działaniu", najlepsze możliwe oszacowanie, prawdziwa skuteczność klasyfikacji, powtarzalność, uogólnianie. . .

- Zbiór danych – reprezentatywny dla problemu
- Wydzielenie zbioru testowego:
  - Brak – oszacowanie obciążone (skuteczność zawyżona)
  - 50% trening, 50% test – oszacowanie obciążone (skuteczność zaniżona)
  - Wydzielenie pojedynczego przykładu (leave one out) + powtarzanie – duża wariancja, złożone obliczeniowo
    - Ale – leave one patient out!
  - Wydzielenie części (np. 5, 10, 20%) i powtórzenie – walidacja krzyżowa, optymalne
  - Wydzielenie zbioru testowego – niezależna weryfikacja wydajności
- Wydzielenie zbioru walidacyjnego
  - Skąd wiemy jakie hiperparametry ustawić?
  - Wewnętrzna pętla walidacji żeby ustalić wartości hiperparametrów
  - Dwa poziomy walidacji (zewnętrzna – zbiór testowy, wewnętrzna – zbiór walidacyjny) – niezależna weryfikacja wydajności uwzględniająca niepewność doboru hiperparametrów
    - Testowanie metody vs testowanie metody z hiperparametrami dobranymi przez eksperta

Klasyfikator, skuteczność 89% . . . oczekiwania? Spodziewany wynik „w działaniu", najlepsze możliwe oszacowanie, prawdziwa skuteczność klasyfikacji, powtarzalność, uogólnianie. . .

- Zbiór danych – reprezentatywny dla problemu
- Wydzielenie zbioru testowego:
  - Brak – oszacowanie obciążone (skuteczność zawyżona)
  - 50% trening, 50% test – oszacowanie obciążone (skuteczność zaniżona)
  - Wydzielenie pojedynczego przykładu (leave one out) + powtarzanie – duża wariancja, złożone obliczeniowo
    - Ale – leave one patient out!
  - Wydzielenie części (np. 5, 10, 20%) i powtórzenie – walidacja krzyżowa, optymalne
  - Wydzielenie zbioru testowego – niezależna weryfikacja wydajności
- Wydzielenie zbioru walidacyjnego
  - Skąd wiemy jakie hiperparametry ustawić?
  - Wewnętrzna pętla walidacji żeby ustalić wartości hiperparametrów
  - Dwa poziomy walidacji (zewnętrzna – zbiór testowy, wewnętrzna – zbiór walidacyjny) – niezależna weryfikacja wydajności uwzględniająca niepewność doboru hiperparametrów
    - Testowanie metody vs testowanie metody z hiperparametrami dobranymi przez eksperta

Klasyfikator, skuteczność 89% . . . oczekiwania? Spodziewany wynik „w działaniu", najlepsze możliwe oszacowanie, prawdziwa skuteczność klasyfikacji, powtarzalność, uogólnianie. . .

- Zbiór danych – reprezentatywny dla problemu
- Wydzielenie zbioru testowego:
  - Brak – oszacowanie obciążone (skuteczność zawyżona)
  - 50% trening, 50% test – oszacowanie obciążone (skuteczność zaniżona)
  - Wydzielenie pojedynczego przykładu (leave one out) + powtarzanie – duża wariancja, złożone obliczeniowo
    - Ale – leave one patient out!
  - Wydzielenie części (np. 5, 10, 20%) i powtórzenie – walidacja krzyżowa, optymalne
  - Wydzielenie zbioru testowego – niezależna weryfikacja wydajności
- Wydzielenie zbioru walidacyjnego
  - Skąd wiemy jakie hiperparametry ustawić?
  - Wewnętrzna pętla walidacji żeby ustalić wartości hiperparametrów
  - Dwa poziomy walidacji (zewnętrzna – zbiór testowy, wewnętrzna – zbiór walidacyjny) – niezależna weryfikacja wydajności uwzględniająca niepewność doboru hiperparametrów
    - Testowanie metody vs testowanie metody z hiperparametrami dobranymi przez eksperta

Klasyfikator, skuteczność 89% . . . oczekiwania? Spodziewany wynik „w działaniu", najlepsze możliwe oszacowanie, prawdziwa skuteczność klasyfikacji, powtarzalność, uogólnianie. . .

- Zbiór danych – reprezentatywny dla problemu
- Wydzielenie zbioru testowego:
  - Brak – oszacowanie obciążone (skuteczność zawyżona)
  - 50% trening, 50% test – oszacowanie obciążone (skuteczność zaniżona)
  - Wydzielenie pojedynczego przykładu (leave one out) + powtarzanie – duża wariancja, złożone obliczeniowo
    - Ale – leave one patient out!
  - Wydzielenie części (np. 5, 10, 20%) i powtórzenie – walidacja krzyżowa, optymalne
  - Wydzielenie zbioru testowego – niezależna weryfikacja wydajności
- Wydzielenie zbioru walidacyjnego
  - Skąd wiemy jakie hiperparametry ustawić?
  - Wewnętrzna pętla walidacji żeby ustalić wartości hiperparametrów
  - Dwa poziomy walidacji (zewnętrzna – zbiór testowy, wewnętrzna – zbiór walidacyjny) – niezależna weryfikacja wydajności uwzględniająca niepewność doboru hiperparametrów
    - Testowanie metody vs testowanie metody z hiperparametrami dobranymi przez eksperta

Klasyfikator, skuteczność 89% . . . oczekiwania? Spodziewany wynik „w działaniu", najlepsze możliwe oszacowanie, prawdziwa skuteczność klasyfikacji, powtarzalność, uogólnianie. . .

- Zbiór danych – reprezentatywny dla problemu
- Wydzielenie zbioru testowego:
  - Brak – oszacowanie obciążone (skuteczność zawyżona)
  - 50% trening, 50% test – oszacowanie obciążone (skuteczność zaniżona)
  - Wydzielenie pojedynczego przykładu (leave one out) + powtarzanie – duża wariancja, złożone obliczeniowo
    - Ale – leave one patient out!
  - Wydzielenie części (np. 5, 10, 20%) i powtórzenie – walidacja krzyżowa, optymalne
  - Wydzielenie zbioru testowego – niezależna weryfikacja wydajności
- Wydzielenie zbioru walidacyjnego
  - Skąd wiemy jakie hiperparametry ustawić?
  - Wewnętrzna pętla walidacji żeby ustalić wartości hiperparametrów
  - Dwa poziomy walidacji (zewnętrzna – zbiór testowy, wewnętrzna – zbiór walidacyjny) – niezależna weryfikacja wydajności uwzględniająca niepewność doboru hiperparametrów
    - Testowanie metody vs testowanie metody z hiperparametrami dobranymi przez eksperta

Analiza jakościowa (danych i wyników)

Klasyfikator, 89% zweryfikowana walidacją krzyżową, dobór parametrów OK – co dalej?
Analiza ilościowa – miary skuteczności
Analiza jakościowa – przegląd wyników pod kątem outlierów, regularności, wzorców itp.

- Szukaj outlierów i anomalii
- Zobacz jak się grupują przykłady (np. w obrębie klas)
- Stosuj redukcję wymiarowości (dla wizualizacji)
- Próbuj prostych metod
- Wizualizuj co tylko się da

- Powodzenia :)

Klasyfikator, 89% zweryfikowana walidacją krzyżową, dobór parametrów OK – co dalej?

Analiza ilościowa – miary skuteczności

Analiza jakościowa – przegląd wyników pod kątem outlierów, regularności, wzorców itp.

- Szukaj outlierów i anomalii
- Zobacz jak się grupują przykłady (np. w obrębie klas)
- Stosuj redukcję wymiarowości (dla wizualizacji)
- Próbuj prostych metod
- Wizualizuj co tylko się da

- Powodzenia :)

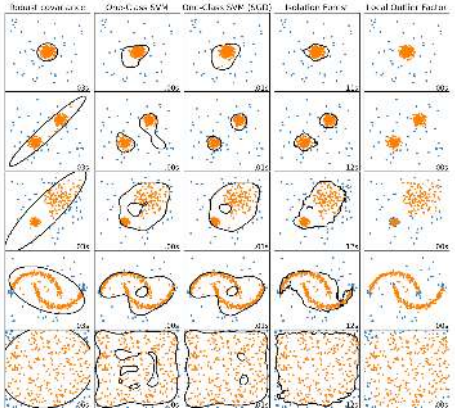
Klasyfikator, 89% zweryfikowana walidacją krzyżową, dobór parametrów OK – co dalej?
Analiza ilościowa – miary skuteczności
Analiza jakościowa – przegląd wyników pod kątem outlierów, regularności, wzorców itp.

- Szukaj outlierów i anomalii
- Zobacz jak się grupują przykłady (np. w obrębie klas)
- Stosuj redukcję wymiarowości (dla wizualizacji)
- Próbuj prostych metod
- Wizualizuj co tylko się da

- Powodzenia :)

Klasyfikator, 89% zweryfikowana walidacją krzyżową, dobór parametrów OK – co dalej?
Analiza ilościowa – miary skuteczności
Analiza jakościowa – przegląd wyników pod kątem outlierów, regularności, wzorców itp.

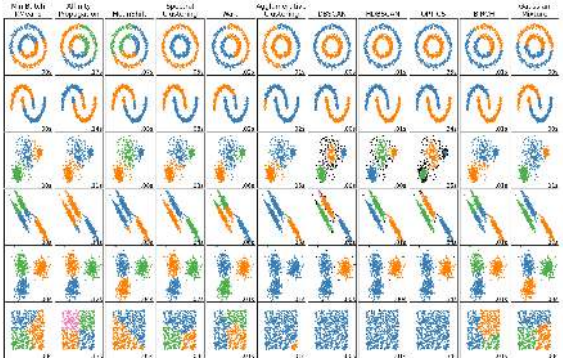
- Szukaj outlierów i anomalii



dokumentacja sklearn, BSD

Klasyfikator, 89% zweryfikowana walidacją krzyżową, dobór parametrów OK – co dalej?
Analiza ilościowa – miary skuteczności
Analiza jakościowa – przegląd wyników pod kątem outlierów, regularności, wzorców itp.

- Szukaj outlierów i anomalii
- Zobacz jak się grupują przykłady (np. w obrębie klas)



dokumentacja sklearn, BSD

- Stosuj redukcje wymiarowości (dla wizualizacji)

Klasyfikator, 89% zweryfikowana walidacją krzyżową, dobór parametrów OK – co dalej?
Analiza ilościowa – miary skuteczności
Analiza jakościowa – przegląd wyników pod kątem outlierów, regularności, wzorców itp.

- Szukaj outlierów i anomalii
- Zobacz jak się grupują przykłady (np. w obrębie klas)
- Stosuj redukcję wymiarowości (dla wizualizacji)
  - PCA
  - tSNE
  - kernel PCA, ICA, NMF, ...
- Próbuj prostych metod
- Wizualizuj co tylko się da
- Powodzenia :)

Klasyfikator, 89% zweryfikowana walidacją krzyżową, dobór parametrów OK – co dalej?
Analiza ilościowa – miary skuteczności
Analiza jakościowa – przegląd wyników pod kątem outlierów, regularności, wzorców itp.

- Szukaj outlierów i anomalii
- Zobacz jak się grupują przykłady (np. w obrębie klas)
- Stosuj redukcję wymiarowości (dla wizualizacji)
- Próbuj prostych metod
  - SVM, RF, MLP
  - kNN, 1-NN
  - Regresja liniowa
- Wizualizuj co tylko się da
- Powodzenia :)

Klasyfikator, 89% zweryfikowana walidacją krzyżową, dobór parametrów OK – co dalej?
Analiza ilościowa – miary skuteczności
Analiza jakościowa – przegląd wyników pod kątem outlierów, regularności, wzorców itp.

- Szukaj outlierów i anomalii
- Zobacz jak się grupują przykłady (np. w obrębie klas)
- Stosuj redukcję wymiarowości (dla wizualizacji)
- Próbuj prostych metod
- Wizualizuj co tylko się da
  - Histogramy/wykresy pudełkowe błędów
  - Przykłady dobrze i źle sklasyfikowane
  - Test wizualnej identyfikacji wzorców
  - Zależność od zewnętrznych parametrów

- Powodzenia :)

Klasyfikator, 89% zweryfikowana walidacją krzyżową, dobór parametrów OK – co dalej?
Analiza ilościowa – miary skuteczności
Analiza jakościowa – przegląd wyników pod kątem outlierów, regularności, wzorców itp.

- Szukaj outlierów i anomalii
- Zobacz jak się grupują przykłady (np. w obrębie klas)
- Stosuj redukcję wymiarowości (dla wizualizacji)
- Próbuj prostych metod
- Wizualizuj co tylko się da

- Powodzenia :)

Akademia Sztuki Kwantowej – podsumowanie bloku ML

Przemysław Głomb

Instytut Informatyki Teoretycznej i Stosowanej Polskiej Akademii Nauk

1 kwietnia 2026 r.

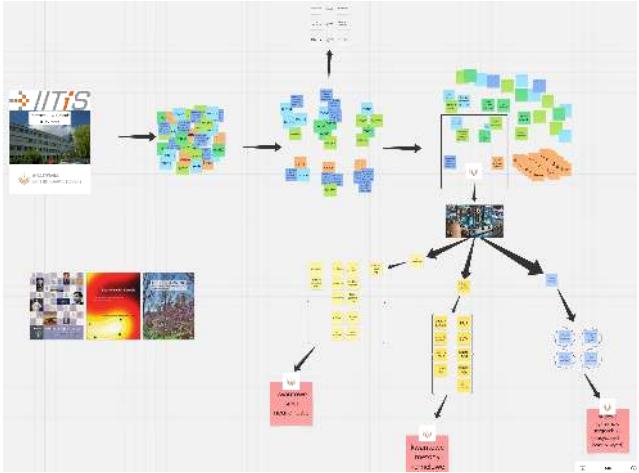
Cel/założenia bloku

- Przygotowanie i realizacja bloku szkoleniowego z zakresu ML
- Zamkniętego („self contained") – wszystkie najważniejsze elementy
- Zaadaptowanego do różnych poziomów doświadczenia uczestników
- Łączącego teorię i praktykę
- Powiązanego z tematami ASK
- Trzymającego się struktury ASK (wykłady/warsztaty)

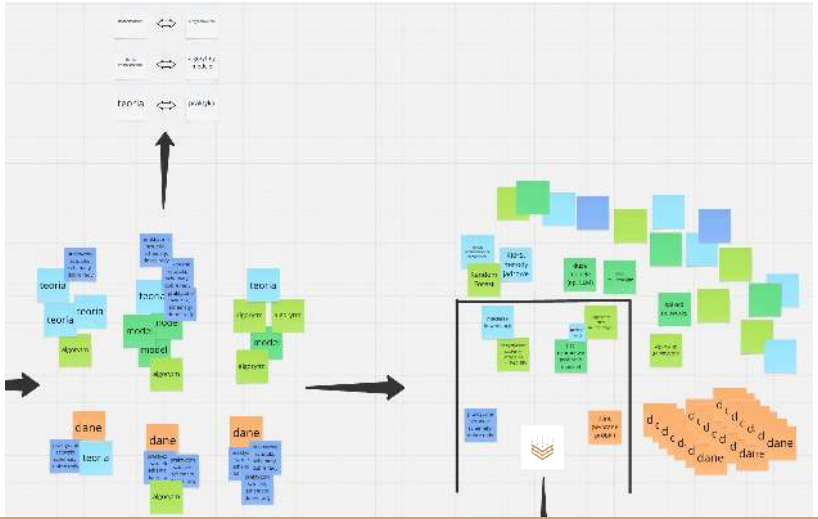
Powiązanie z innymi blokami ASK

- Wykłady wprowadzające (Podstawy matematyczne obliczeń kwantowych i SI, Wprowadzenie do obliczeń kwantowych i algorytmów kwantowych)
- Kwantowy perceptron
- Kwantowe uczenie maszynowe i kwantowe metody jądrowe
- Kwantowe wyżarzanie kombinatorycznych problemów optymalizacyjnych

Droga do celu



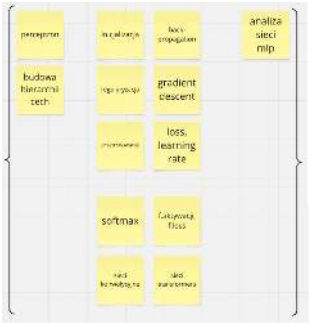
Droga do celu

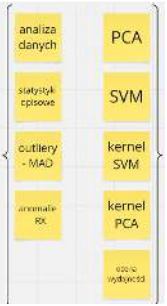


Droga do celu

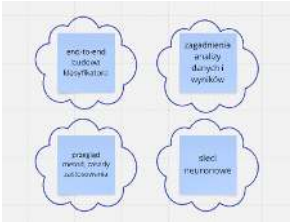
- Modele klasyczne i sieci neuronowe
- Pełen cykl (od analizy danych, przez przygotowanie modelu, do interpretacji wyników)
- Wykłady i warsztaty uzupełniają się
- Wybrane przykłady danych – i modeli
- Podstawy teoretyczne – i schematy użycia
- Matematyka – i programowanie

Droga do celu – wykłady





Droga do celu – warsztaty



- 4 obszary:
  1. Analiza i przygotowanie zbiorów danych
  2. Procedura eksperymentu klasyfikacji
  3. Podstawy sieci neuronowych
  4. Modele głębokiego uczenia
- Miks demonstracji i zadań do wykonania
- Cele: (a) znajomość metod podstawowych, (b) umiejętność ich zakodowania, (c)podstawy teoretyczne i zasady działania metod/procedur

Realizacja – wykłady

- Proporcje zakres/szczegółowość (RL: exploration vs exploitation)
- Płynność przejścia i powiązanie między elementami
- Dobór ilustracji, przykładów, wzorów, wyprowadzeń...
- Ciekawostki
  - 1_perceptron – jak neurony budują hierarchiczną ekstrakcję cech
  - 2_mlp – „pakiet podstawowy" sieci neuronowych
  - 4_covariance – „podróże z macierzą kowariancji"
  - 5_stats – SVM, konstrukcja eksperymentu
- Ciężko się nagrywa, lepiej mówić, a najlepiej mówić na żywo :)
- Refleksje po czasie – podział między filmikami, animacje slajdów, proporcje materiału...

Realizacja – warsztaty

- Różnorodność uczestników – kalibracja poziomu trudności
- Podróż przez wybrane zbiory danych, np. obrazowanie hiperspektralne (wybór metod podyktowany założeniami)
- „Zadanie na 10 minut" – trudne do ukończenia ale wystarczające do ćwiczeń, kompromis ile zadań/jak obszernych
- Zadania z nagrodami :)
- Zadania oraz interaktywne tutoriale (z możliwością samodzielnej realizacji)

Uwagi uczestników (ankiety)

- Łatwiejsze elementy – „Teoria była OK - przyswajalna", „Ćwiczenia praktyczne, ale z pomocą ChatGPT - sam nie byłbym w stanie ich zrobić", „Koncepcje, ponieważ nie wymagały dużego doświadczenia programistycznego"
- Trudniejsze elementy – PyTorch, programowanie
- Znajomość tematu u uczestników – podstawowa ("Ogólne koncepcje", "Głębsze aspekty były dla mnie nowe.")
- Co można było wynieść, co się przyda – praktyczne elementy (używanie bibliotek, implementacja i parametryzacja modeli, ćwiczenia, zebranie zagadnień)
- Uwagi – warsztaty spełniły zadanie, plus uwagi "mniej ale dokładniej"

Kierunki rozwoju, warsztaty z perspektywy czasu

Jak przygotowywałbym warsztaty teraz, w świecie pełnym LLM?

- Bardzo podobnie (+więcej ćwiczeń z modelami w chmurze, -obsługa lokalnego LLM)
- Szczegółowo:
  1. Analiza i przygotowanie danych – proces się nie zmienił; z ChatGPT jest dużo łatwiej... ...ale wciąż inżynier powinien rozumieć mechanikę działania metod i procesu analizy
  2. Klasyfikacja – znajomość i rozumienie konstrukcji eksperymentu jest podstawowe, nawet jeżeli kod pisze nam Copilot
  3. Podstawy sieci neuronowych – warto znać intuicję i podstawy ich działania (i implementacji), nawet jeżeli korzystamy z Claude Code czy Antigravity
  4. Modele głębokiego uczenia – aktualizacja (ale wymagana byłaby i tak)
- Co jest siłą modeli językowych (poza kodowaniem, dokumentacją, znajdowaniem artykułów)
  - Konstrukcja i uzasadnienie ścieżki przetwarzania do konkretnego obszaru problemowego (np. ECG)

Kierunki rozwoju, warsztaty z perspektywy czasu

Jak przygotowywałbym warsztaty teraz, w świecie pełnym LLM?

- Bardzo podobnie (+więcej ćwiczeń z modelami w chmurze, -obsługa lokalnego LLM)
- Szczegółowo:
  1. Analiza i przygotowanie danych – proces się nie zmienił; z ChatGPT jest dużo łatwiej... ...ale wciąż inżynier powinien rozumieć mechanikę działania metod i procesu analizy
  2. Klasyfikacja – znajomość i rozumienie konstrukcji eksperymentu jest podstawowe, nawet jeżeli kod pisze nam Copilot
  3. Podstawy sieci neuronowych – warto znać intuicję i podstawy ich działania (i implementacji), nawet jeżeli korzystamy z Claude Code czy Antigravity
  4. Modele głębokiego uczenia – aktualizacja (ale wymagana byłaby i tak)
- Co jest siłą modeli językowych (poza kodowaniem, dokumentacją, znajdowaniem artykułów)
  - Konstrukcja i uzasadnienie ścieżki przetwarzania do konkretnego obszaru problemowego (np. ECG)

Podsumowanie

Dziękuję za uwagę, udział w warsztatach, wysłuchanie wykładów :)

Wasze pytania? Uwagi? Komentarze?

Ostatnie słowo – ML idzie do przodu (nie biegnie, nie stoi), polecam iść w równym tempie