Sztuczna inteligencja w cyberbezpieczeństwie

dr inż. Łukasz Bartczuk
Katedra Sztucznej Inteligencji

lukasz.Bartczuk@pcz.pl

"There are only two types of companies: those that have been hacked and those that will be."

Robert Mueller,
FBI Director,
2012.

Wprowadzenie

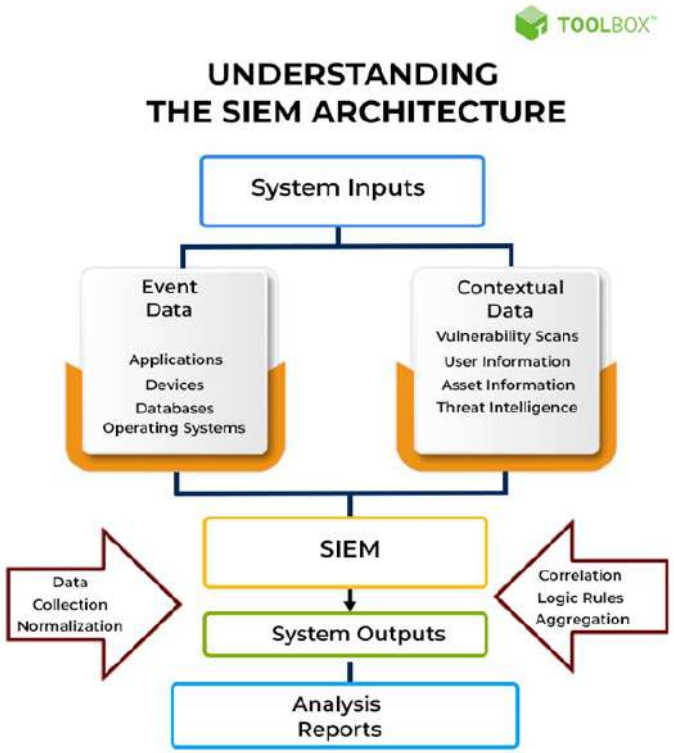
Cyberbezpieczeństwo

Ustawa o krajowym systemie cyberbezpieczeństwa (Dz. U. 2018 poz. 1560)

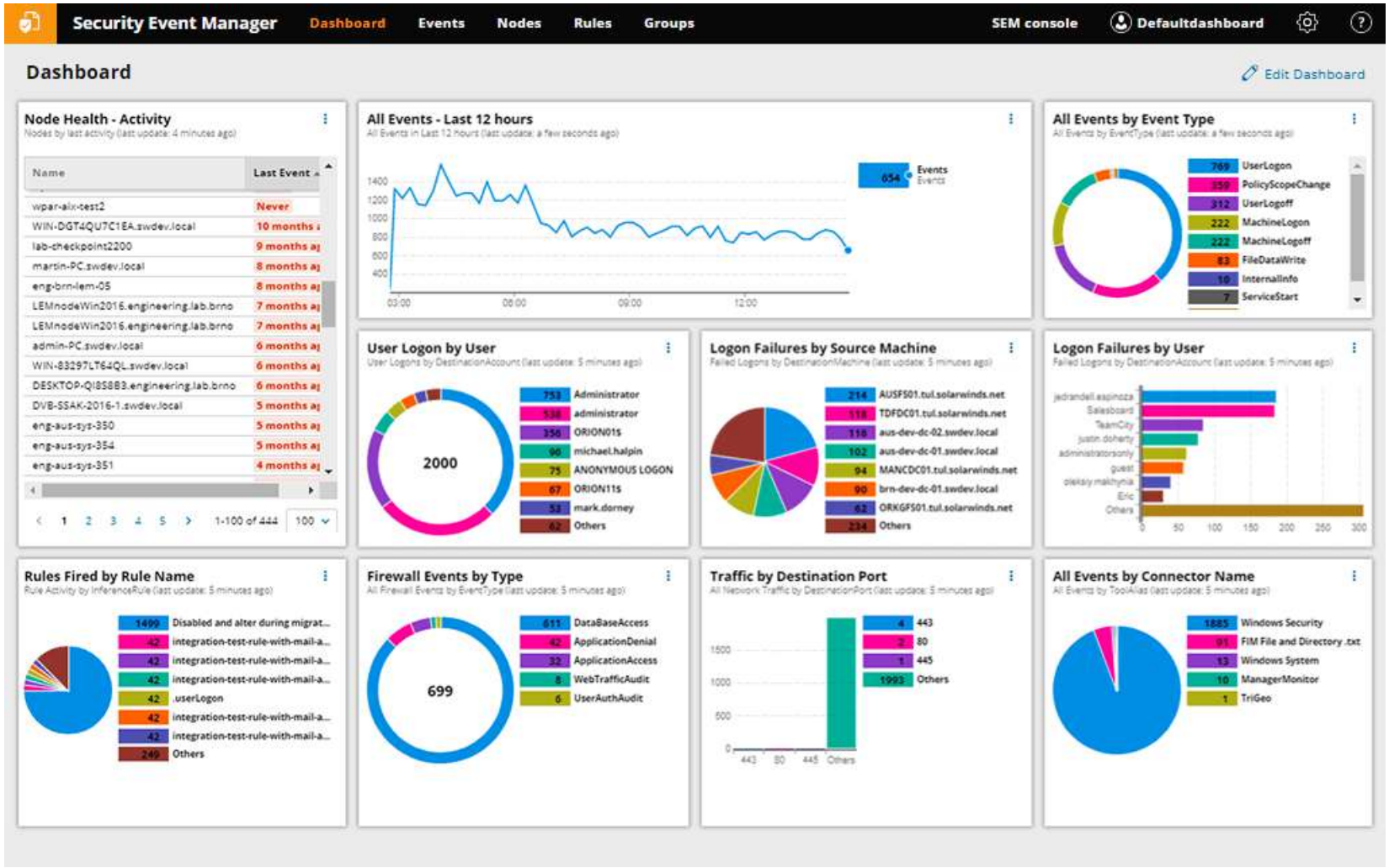
„Cyberbezpieczeństwo – odporność systemów informacyjnych na działania naruszające poufność, integralność, dostępność i autentyczność przetwarzanych danych lub związanych z nimi usług oferowanych przez te systemy"

SIEM

Security information and event management



SIEM


Security Event Manager
Dashboard
Events
Nodes
Rules
Groups
SEM console
Defaultdashboard
Dashboard
Edit Dashboard
Node Health - Activity
Nodes by last activity (last update: 4 minutes ago)
Name
Last Event
wpar-aix-test2
Never
WIN-DGT4QUTC1EA.swdev.local
10 months
lab-checkpoint2200
9 months
martin-PC.swdev.local
8 months
eng-brn-lem-05
8 months
LEMnodeWin2016.engineering.lab.brno
7 months
LEMnodeWin2016.engineering.lab.brno
7 months
admin-PC.swdev.local
6 months
WIN-83297LT64QL.swdev.local
6 months
DESKTOP-QI8S8B3.engineering.lab.brno
6 months
DVB-SSAK-2016-1.swdev.local
5 months
eng-aus-sys-350
5 months
eng-aus-sys-354
5 months
eng-aus-sys-351
4 months
1-100 of 444
100
All Events - Last 12 hours
All Events in Last 12 hours (last update: a few seconds ago)
654
Events
All Events by Event Type
All Events by EventType (last update: a few seconds ago)
769 UserLogon
159 PolicyScopeChange
312 UserLogoff
222 MachineLogon
222 MachineLogoff
83 FileDataWrite
10 InternalInfo
7 ServiceStart
User Logon by User
User Logons by DestinationAccount (last update: 5 minutes ago)
2000
753 Administrator
538 administrator
356 ORION01$
90 michael.halpin
75 ANONYMOUS LOGON
67 ORION11$
53 mark.dorney
62 Others
Logon Failures by Source Machine
Failed Logons by DestinationMachine (last update: 5 minutes ago)
214 AUSFS01.tul.solarwinds.net
118 TDFDC01.tul.solarwinds.net
118 aus-dev-dc-02.swdev.local
102 aus-dev-dc-01.swdev.local
94 MANCDC01.tul.solarwinds.net
90 brn-dev-dc-01.swdev.local
62 ORKGFS01.tul.solarwinds.net
234 Others
Logon Failures by User
Failed Logons by DestinationAccount (last update: 5 minutes ago)
jedrandell espinoza
Salesboard
TeamCity
justin.doherty
administratorsonly
guest
oleksiy.makhynia
Eric
Others
0
50
100
150
200
250
300
Rules Fired by Rule Name
Rule Activity by InferenceRule (last update: 5 minutes ago)
1499 Disabled and alter during migrat...
42 integration-test-rule-with-mail-a...
42 integration-test-rule-with-mail-a...
42 integration-test-rule-with-mail-a...
42 ,userLogon
42 integration-test-rule-with-mail-a...
42 integration-test-rule-with-mail-a...
249 Others
Firewall Events by Type
All Firewall Events by EventType (last update: 5 minutes ago)
699
611 DataBaseAccess
42 ApplicationDenial
32 ApplicationAccess
8 WebTrafficAudit
6 UserAuthAudit
Traffic by Destination Port
All Network Traffic by DestinationPort (last update: 5 minutes ago)
1500
1000
500
0
443
80
445
Others
4 443
2 80
1 445
1993 Others
All Events by Connector Name
All Events by ToolAlias (last update: 5 minutes ago)
1885 Windows Security
91 FIM File and Directory .txt
13 Windows System
10 ManagerMonitor
1 TriGeo

Narzędzia SIEM

Sztuczna inteligencja to system komputerowy, który potrafi wykonywać zadania typowo skojarzone z istotami inteligentnymi

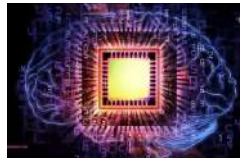
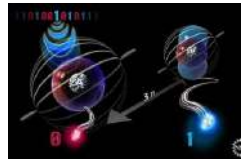
Sztuczna inteligencja dziś i jutro









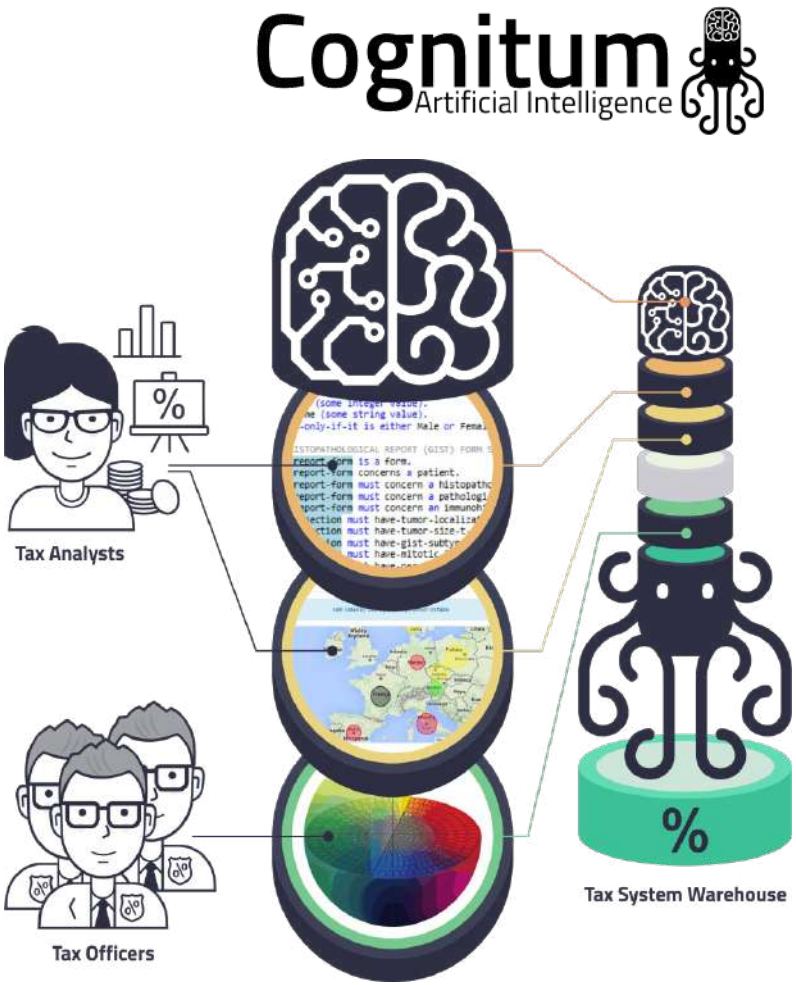




90 XXw | początek XXI w | następne 5-10 lat | następne 20-30 lat | > 30 lat

Na podstawie: "Fourth Industrial Revolution for the Earth", PwC

W roku 2013 superkomputer
Fujitsu K potrzebował
40 minut aby zasymulować
1% pracy naszego mózgu
w ciągu 1 sekundy

Przykładowe obecne Polskie aplikacje wykorzystujące sztuczną inteligencję

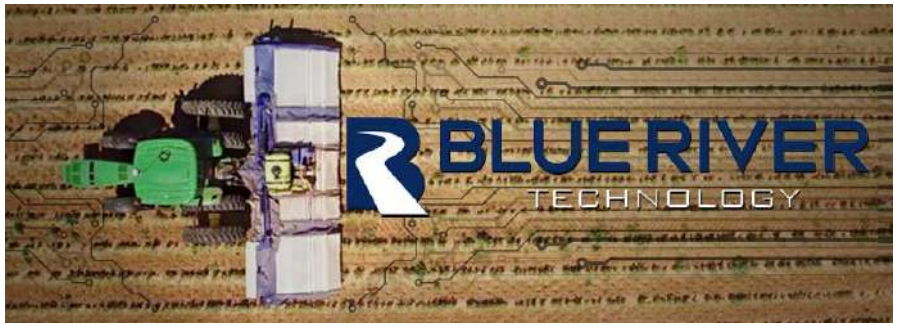


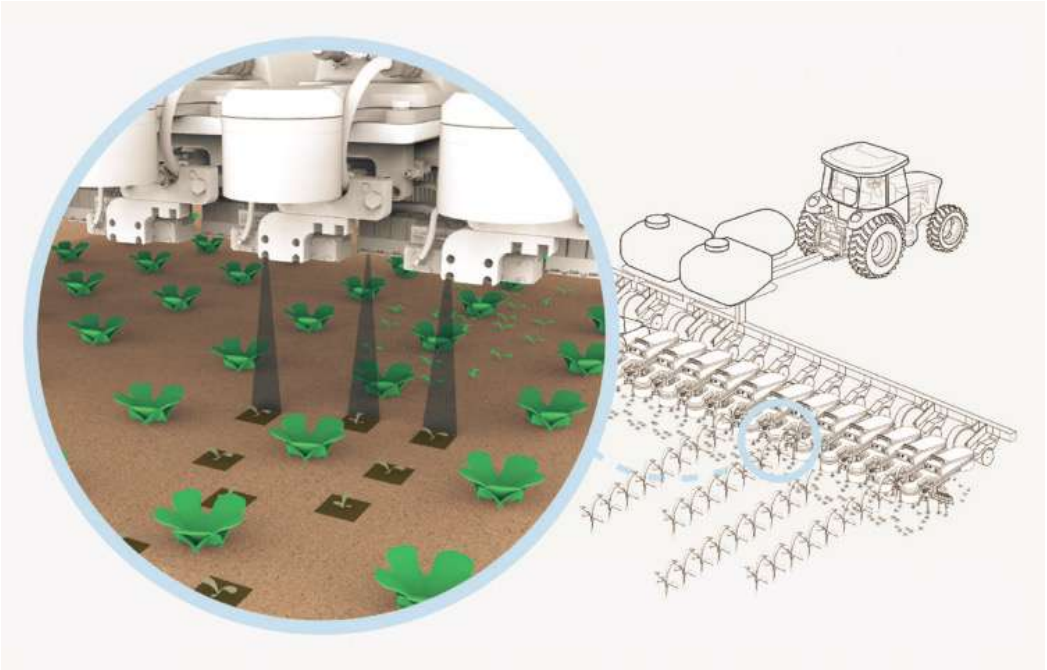
System wykrywania oszustw podatkowych

Korzyści:

- zmniejszenie luki podatkowej
- zminimalizowane koszty niepotrzebnych kontroli

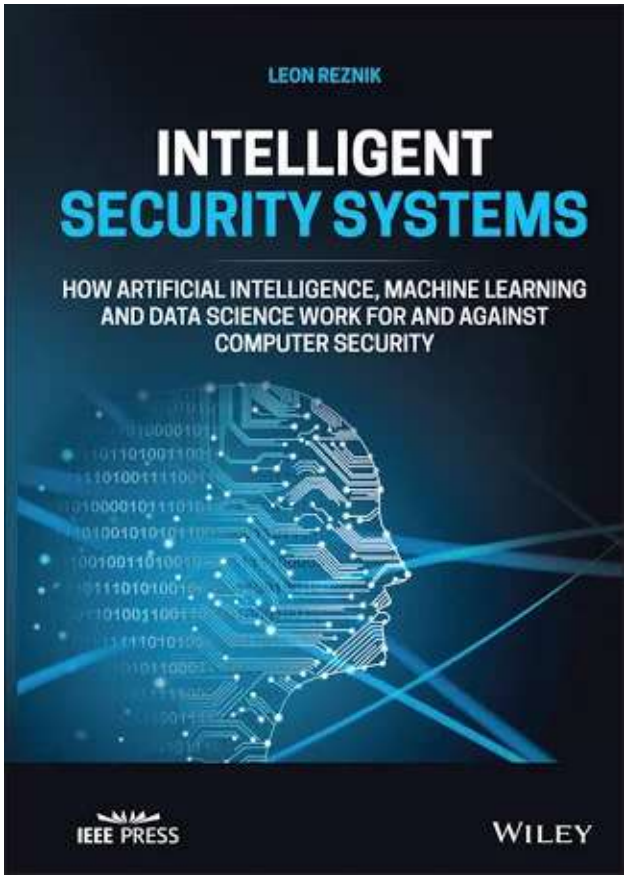
SI w rolnictwie

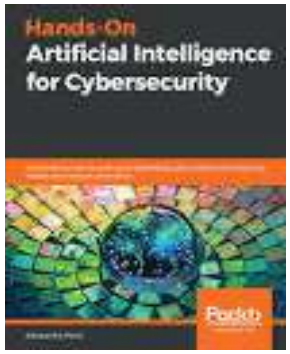


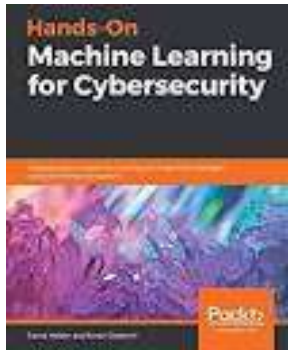


Jak AI może nauczyć się i zrozumieć co jest normalnym działaniem, a co jest działaniem nietypowym

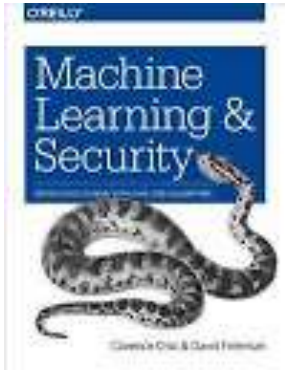
Literatura

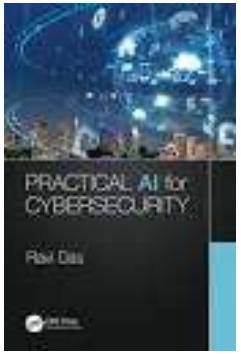

LEON REZNIK
INTELLIGENT
SECURITY SYSTEMS
HOW ARTIFICIAL INTELLIGENCE, MACHINE LEARNING AND DATA SCIENCE WORK FOR AND AGAINST COMPUTER SECURITY
IEEE PRESS
WILEY

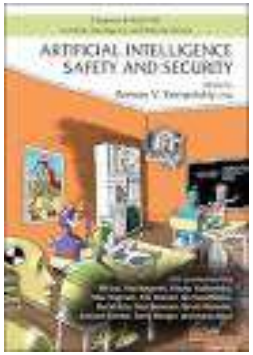

Hands-On
Artificial Intelligence for Cybersecurity
Packt


Hands-On
Machine Learning for Cybersecurity
Packt

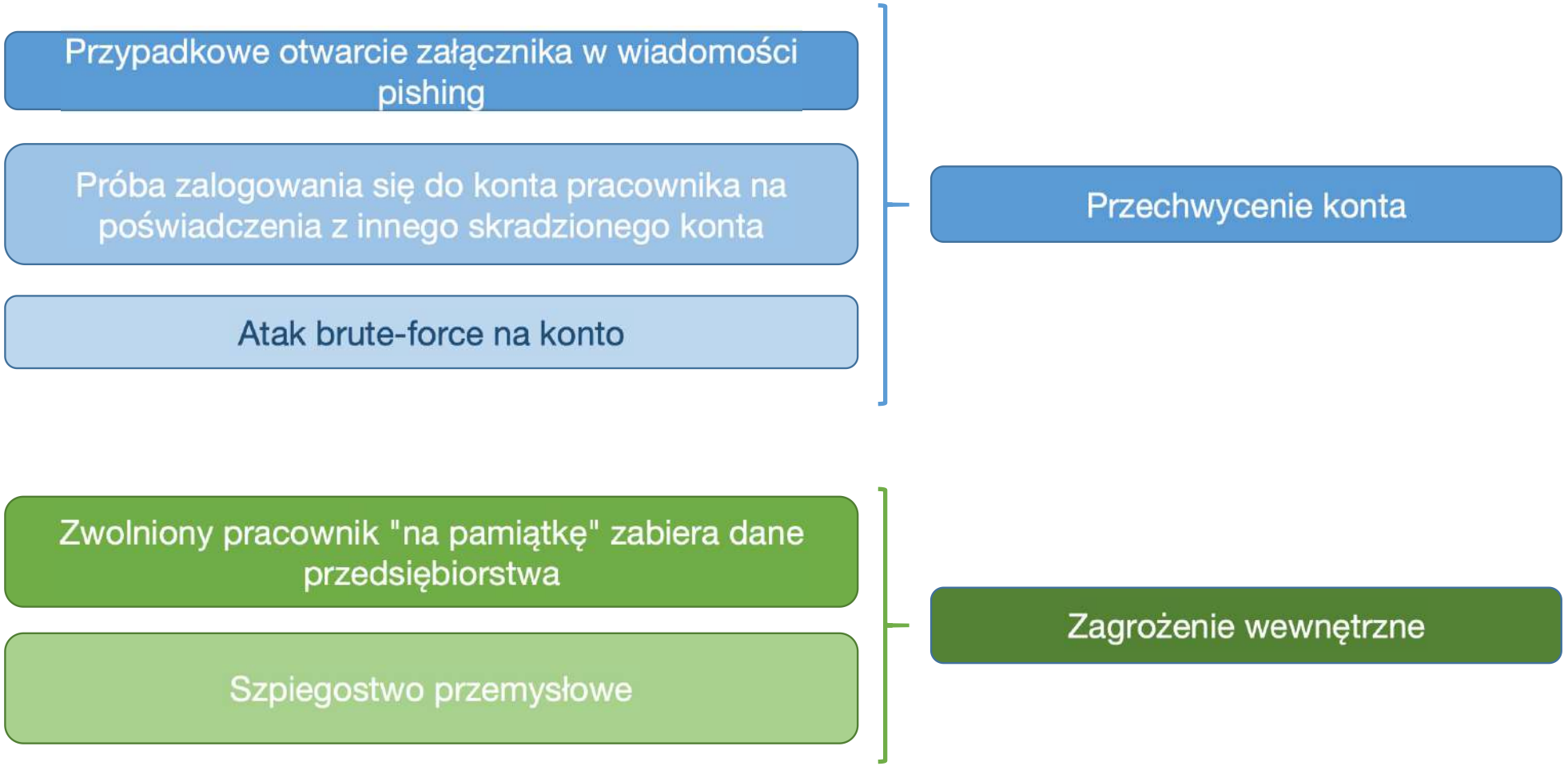

Artificial Intelligence for Cyber Security: Methods, Issues and Possible Horizons or Opportunities


O'REILLY
Machine Learning & Security


PRACTICAL AI for CYBERSECURITY


ARTIFICIAL INTELLIGENCE SAFETY AND SECURITY

Wykrywanie anomalii


Przypadkowe otwarcie załącznika w wiadomości pishing
Próba zalogowania się do konta pracownika na poświadczenia z innego skradzionego konta
Atak brute-force na konto
Przechwycenie konta
Zwolniony pracownik "na pamiątkę" zabiera dane przedsiębiorstwa
Szpiegostwo przemysłowe
Zagrożenie wewnętrzne


Kradzież konta i próba zwiększenia uprawnień
Przypadkowe otwarcie załącznika w wiadomości pishing
Nadużycie przywilejów

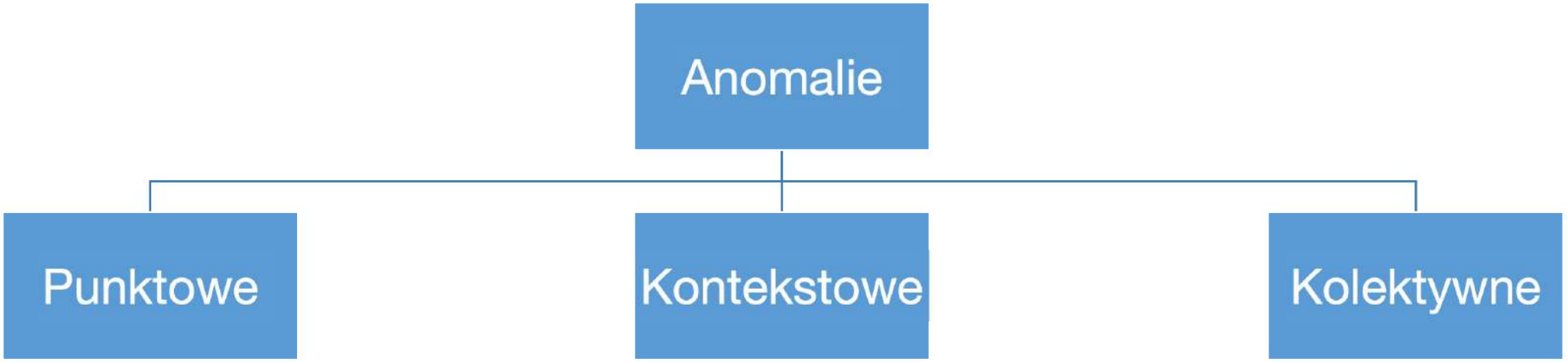
Wykrywanie anomalii

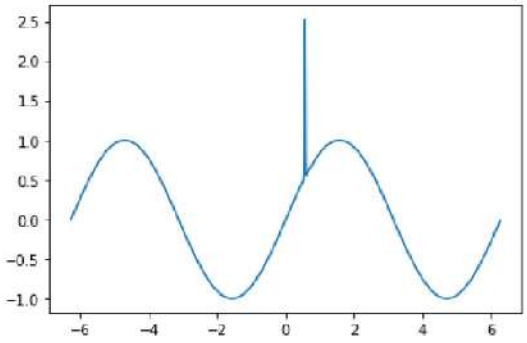
Wykrywanie anomalii jest procesem identyfikowania odstających od normy, nieoczekiwanych elementów lub zdarzeń.

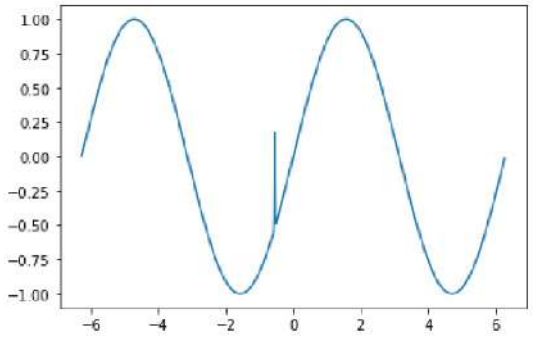
Anomalie występują rzadko

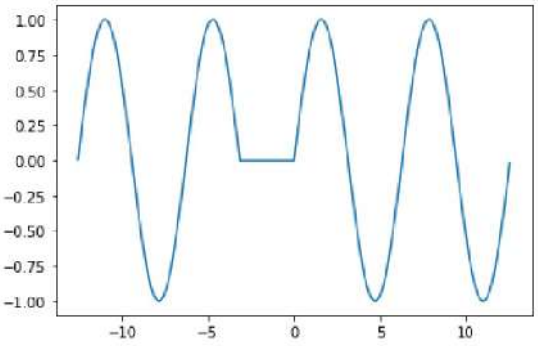
Cechy sytuacji nadzwyczajnych różnią się znacząco od normalnych

Rodzaje anomalii









Przykładowe zastosowania

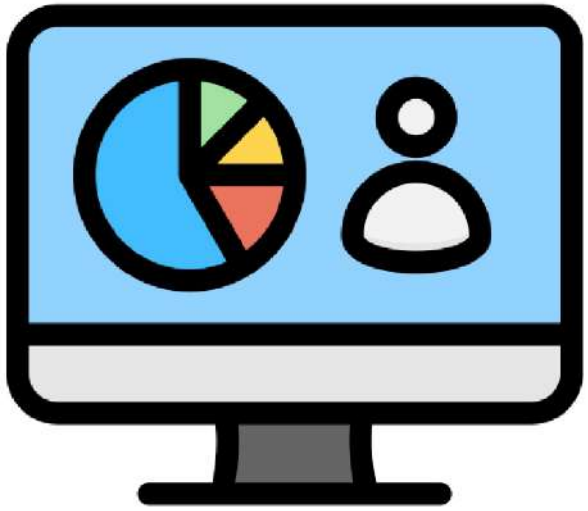
Oszustwa związane z kartami kredytowymi

Cyber atak

Medycyna

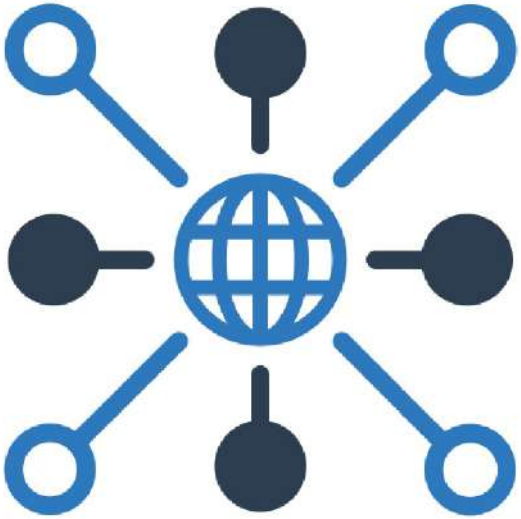
Wykrywanie awarii

Detekcja anomalii, a cyberbezpieczeństwo



User Behavioral Analytics

User / Entity Behavioral Analytics



Network Traffic Analysis

User Behavioral Analytics

Analiza zachowania użytkowników i jednostek polega na identyfikowaniu wzorców aktywności, które są poza normalnymi wzorcami oczekiwanych zachowań w celu wykrycia aktywności podejrzanej lub wręcz szkodliwej

Analiza logów

https://www.xenonstack.com/blog/log-analytics-mining

Logi

Nazwa logów	Źródło
Logi transakcji (transaction logs)	Np. systemy bazodanowe
Log wiadomości (message logs)	Np. programy chat
Syslog	Urządzenia sieciowe: routery, swiche itp.
Plik logów serwera	Web server
Deamon log	Docker
Pods	Kubernetes
Zdarzenia windows	Windows

Kroki w analizie logów

Logi Web Server

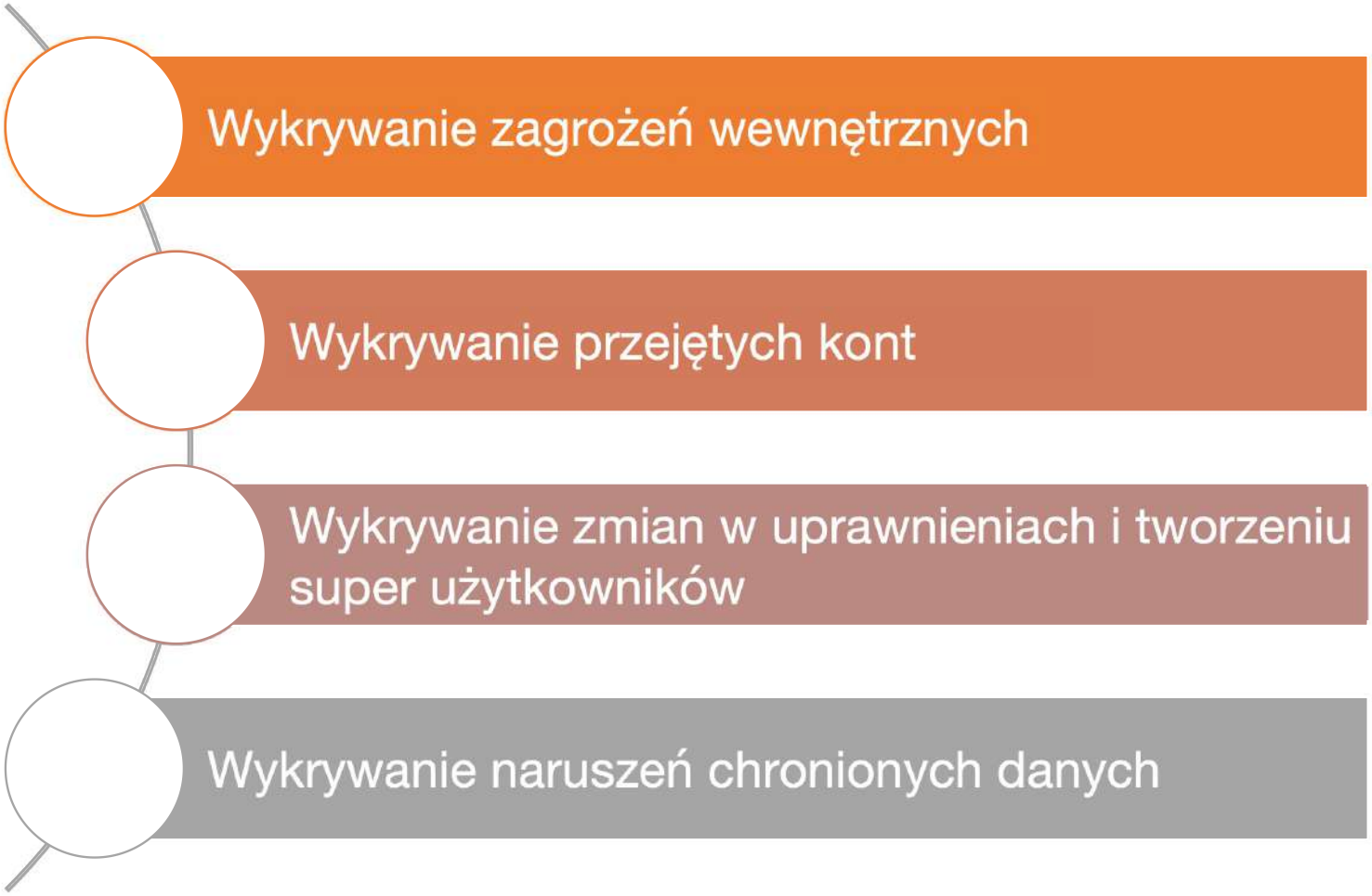
111.222.188.123 HOME - [01/Feb/1998:01:08:39 -0800] "GET /bannerad/ad.htm HTTP/1.0" 200 198 "http://www.referrer.com/bannerad/ba_intro.htm" "Mozilla/4.01 (Macintosh; I; PPC)"

111.222. 188.123 HOME - [01/Feb/1998:01:08:46 -0800] "GET /bannerad/ad.htm HTTP/1.0" 200 28083 "http://www.referrer.com/bannerad/ba_intro.htm" "Mozilla/4.01 (Macintosh; I; PPC)"

111.222. 188.123 AWAY - [01/Feb/1998:01:08:53 -0800] "GET /bannerad/ad7.gif HTTP/1.0" 200 9332 "http://www.referrer.com/bannerad/ba_ad.htm" "Mozilla/4.01 (Macintosh; I; PPC)"

111.222. 188.123 AWAY - [01/Feb/1998:01:09:14 -0800] "GET /bannerad/click.htm HTTP/1.0" 200 207 "http://www.referrer.com/bannerad/menu.htm" "Mozilla/4.01 (Macintosh; I; PPC)"

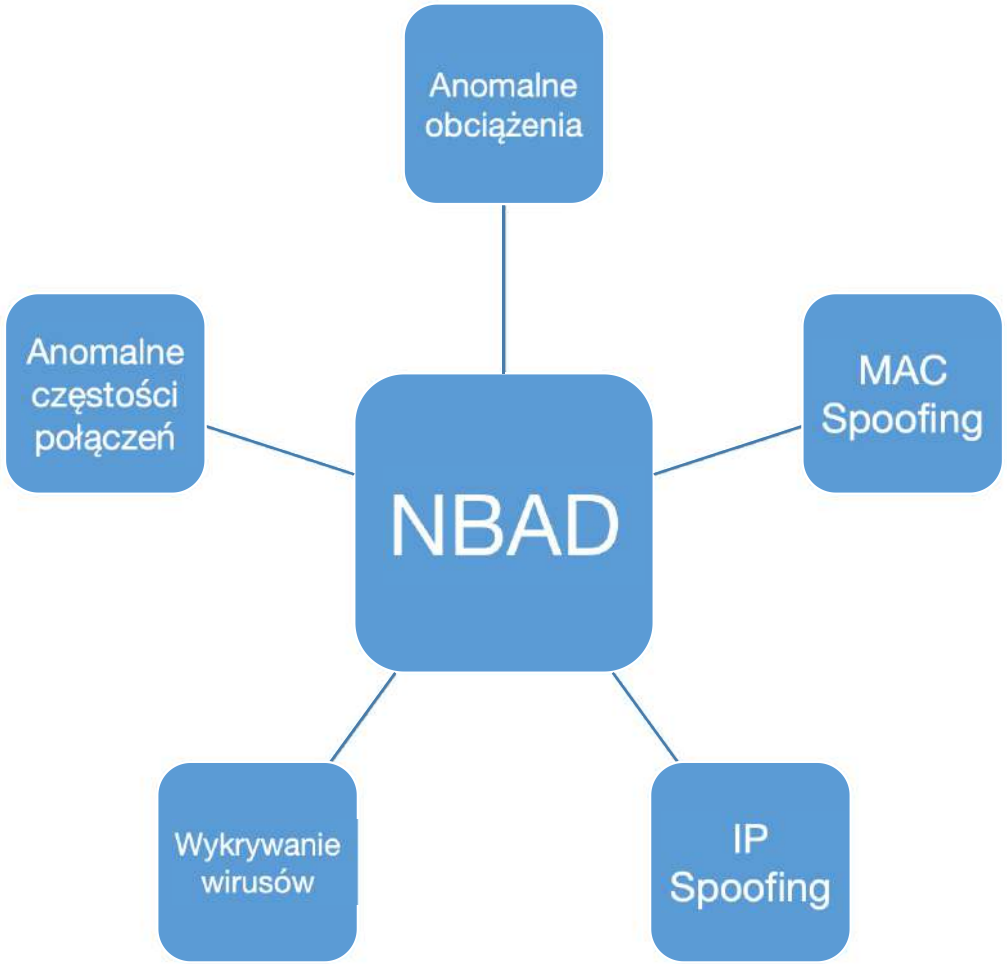
User Behavioral Analytics



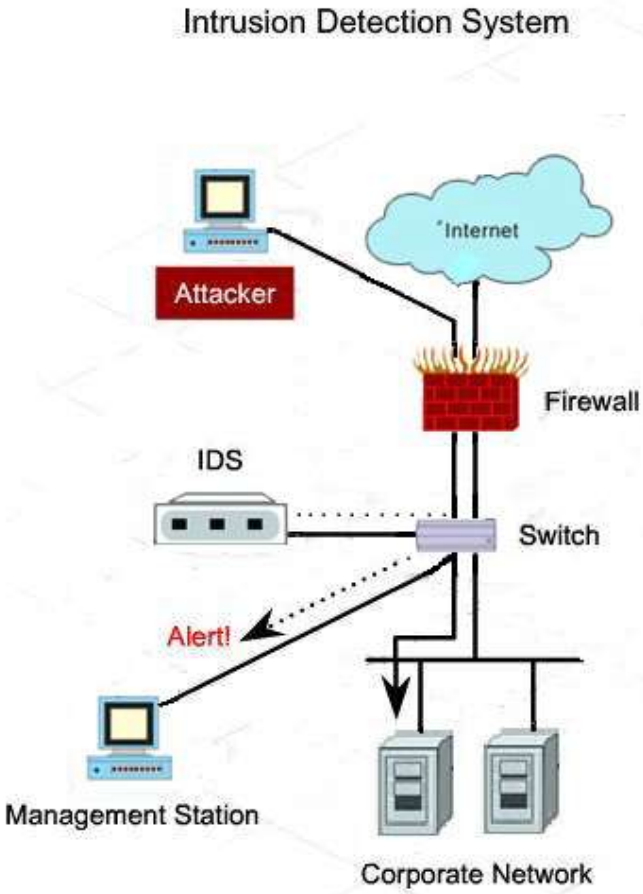
Analiza ruchu sieciowego

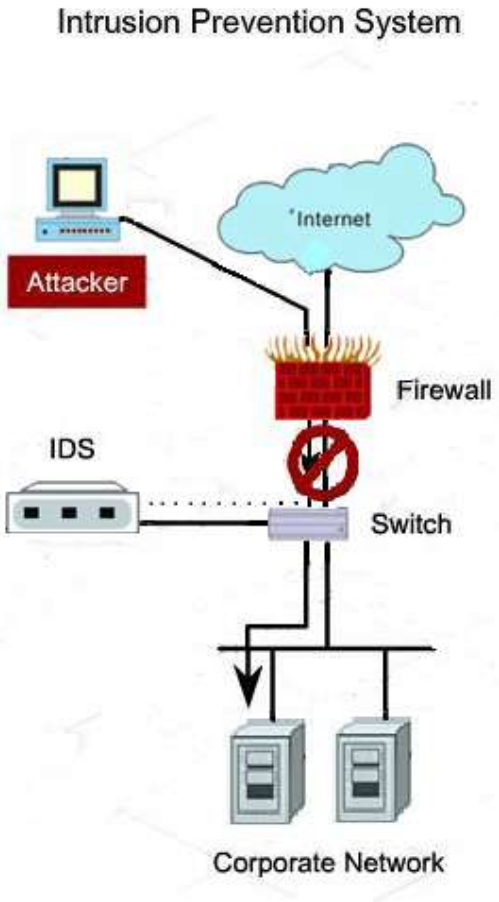
Analiza ruchu sieciowego polega na monitorowaniu dostępności i aktywności sieci komputerowej w celu wykrywania anomalii

Detekcja anomalnych zachowań w sieci



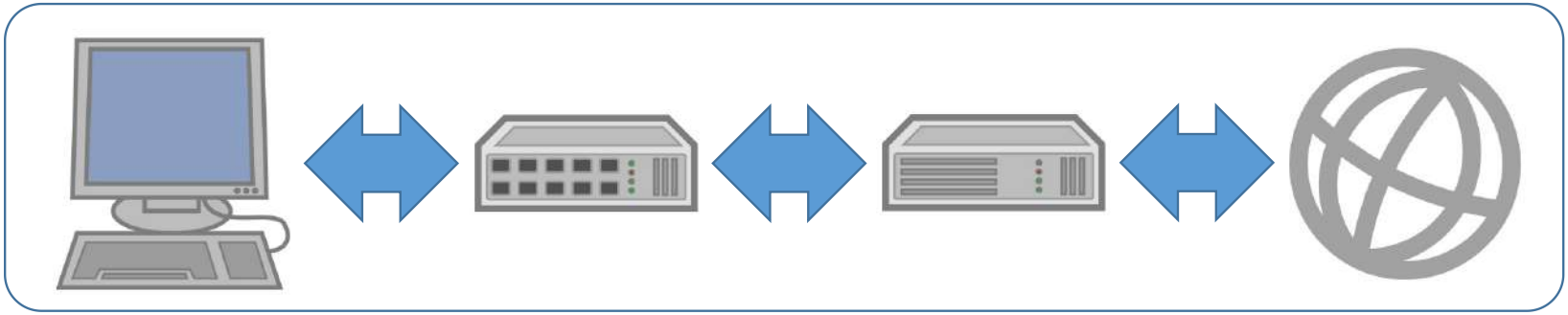
Systemy IDS/IPS

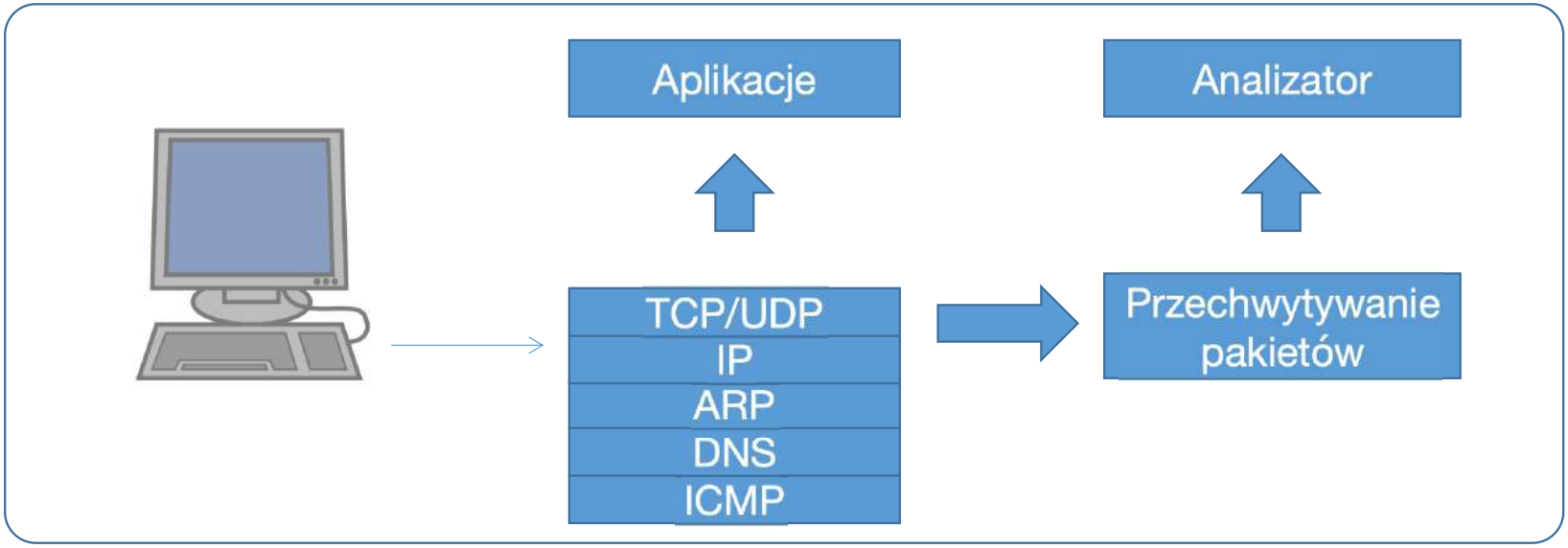




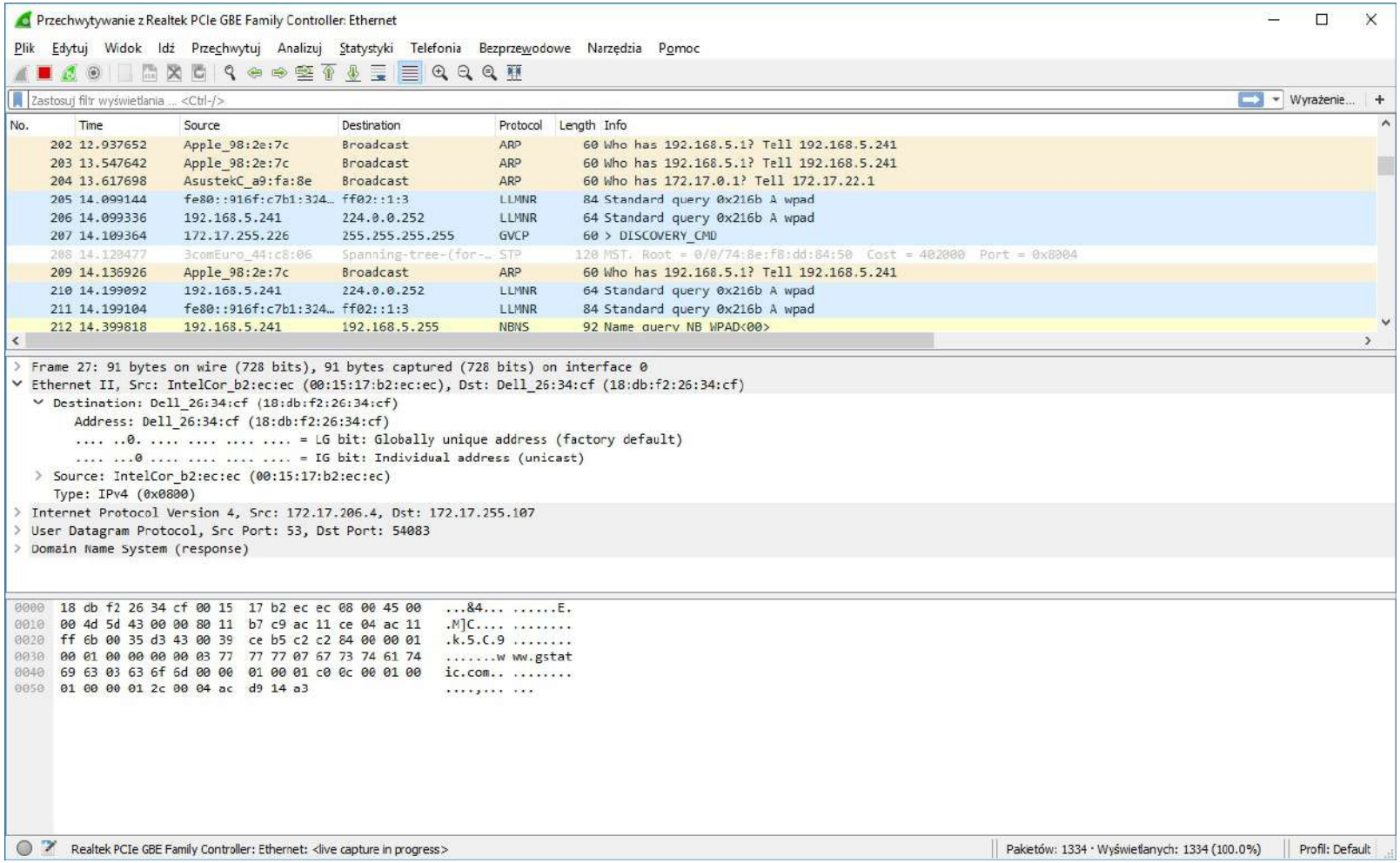
IPS/IDS

Analizator pakietów sieciowych



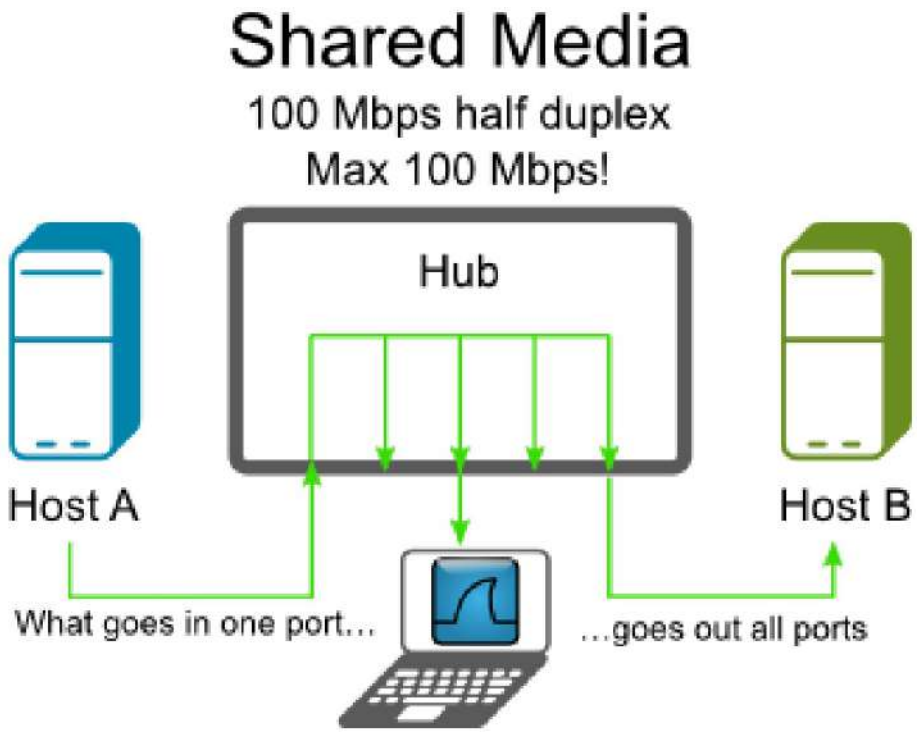


Wireshark


Przechwytywanie z Realtek PCIe GBE Family Controller: Ethernet
Plik Edytuj Widok Idź Przechwytuj Analizuj Statystyki Telefonia Bezprzewodowe Narzędzia Pomoc
Zastosuj filtr wyświetlania ... <Ctrl-/>
Wyrażenie...
No. Time Source Destination Protocol Length Info
202 12.937652 Apple_98:2e:7c Broadcast ARP 60 Who has 192.168.5.1? Tell 192.168.5.241
203 13.547642 Apple_98:2e:7c Broadcast ARP 60 Who has 192.168.5.1? Tell 192.168.5.241
204 13.617698 AsustekC_a9:fa:8e Broadcast ARP 60 Who has 172.17.0.1? Tell 172.17.22.1
205 14.099144 fe80::916f:c7b1:324... ff02::1:3 LLMNR 84 Standard query 0x216b A wpad
206 14.099336 192.168.5.241 224.0.0.252 LLMNR 64 Standard query 0x216b A wpad
207 14.109364 172.17.255.226 255.255.255.255 GVCP 60 > DISCOVERY_CMD
208 14.120477 3comEuro_44:c8:06 Spanning-tree-(for-... STP 120 MST. Root = 0/0/74:8e:f8:dd:84:50 Cost = 402000 Port = 0x8004
209 14.136926 Apple_98:2e:7c Broadcast ARP 60 Who has 192.168.5.1? Tell 192.168.5.241
210 14.199092 192.168.5.241 224.0.0.252 LLMNR 64 Standard query 0x216b A wpad
211 14.199104 fe80::916f:c7b1:324... ff02::1:3 LLMNR 84 Standard query 0x216b A wpad
212 14.399818 192.168.5.241 192.168.5.255 NBNS 92 Name query NB WPAD<00>
Frame 27: 91 bytes on wire (728 bits), 91 bytes captured (728 bits) on interface 0
Ethernet II, Src: IntelCor_b2:ec:ec (00:15:17:b2:ec:ec), Dst: Dell_26:34:cf (18:db:f2:26:34:cf)
Destination: Dell_26:34:cf (18:db:f2:26:34:cf)
Address: Dell_26:34:cf (18:db:f2:26:34:cf)
.... ..0. .... = LG bit: Globally unique address (factory default)
....0 .... = IG bit: Individual address (unicast)
Source: IntelCor_b2:ec:ec (00:15:17:b2:ec:ec)
Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 172.17.206.4, Dst: 172.17.255.107
User Datagram Protocol, Src Port: 53, Dst Port: 54083
Domain Name System (response)
0000 18 db f2 26 34 cf 00 15 17 b2 ec ec 08 00 45 00 ...&4... ......E.
0010 00 4d 5d 43 00 00 80 11 b7 c9 ac 11 ce 04 ac 11 .M]C.... ....
0020 ff 6b 00 35 d3 43 00 39 ce b5 c2 c2 84 00 00 01 .k.5.C.9 ....
0030 00 01 00 00 00 00 03 77 77 77 07 67 73 74 61 74 ....w ww.gstat
0040 69 63 03 63 6f 6d 00 00 01 00 01 c0 0c 00 01 00 ic.com.. ....
0050 01 00 00 01 2c 00 04 ac d9 14 a3 ....,... ...
Realtek PCIe GBE Family Controller: Ethernet: <live capture in progress>
Pakietów: 1334 · Wyświetlanych: 1334 (100.0%)
Profil: Default

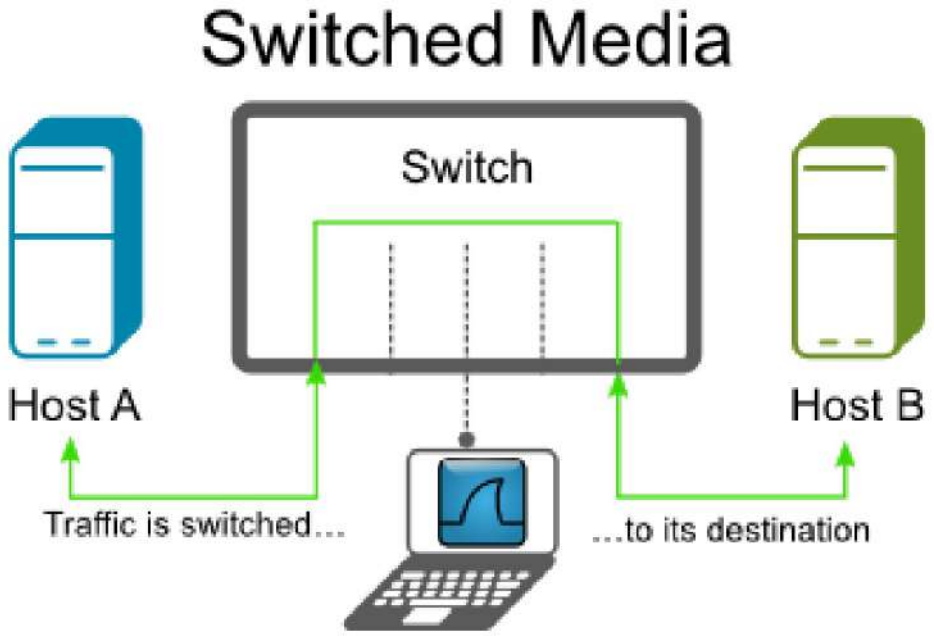
Podłączenie wiresharka do sieci

Podłączenie przez huba



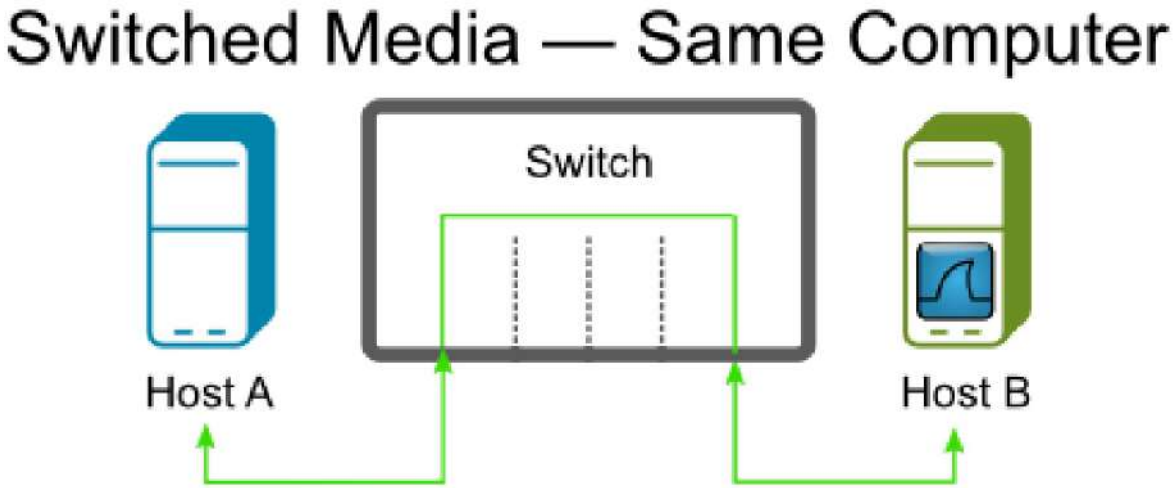
Podłączenie wiresharka do sieci

Podłączenie przez switch



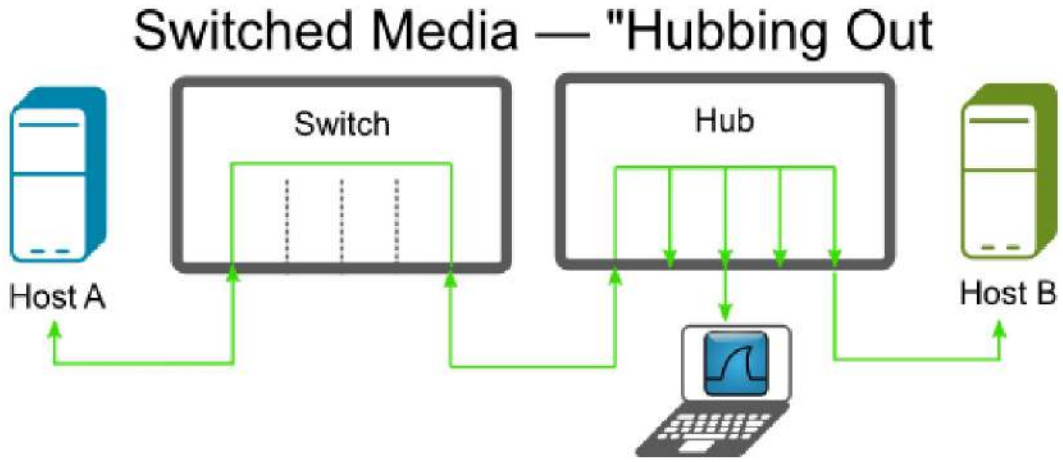
Podłączenie wiresharka do sieci

Podłączenie
na komputerze



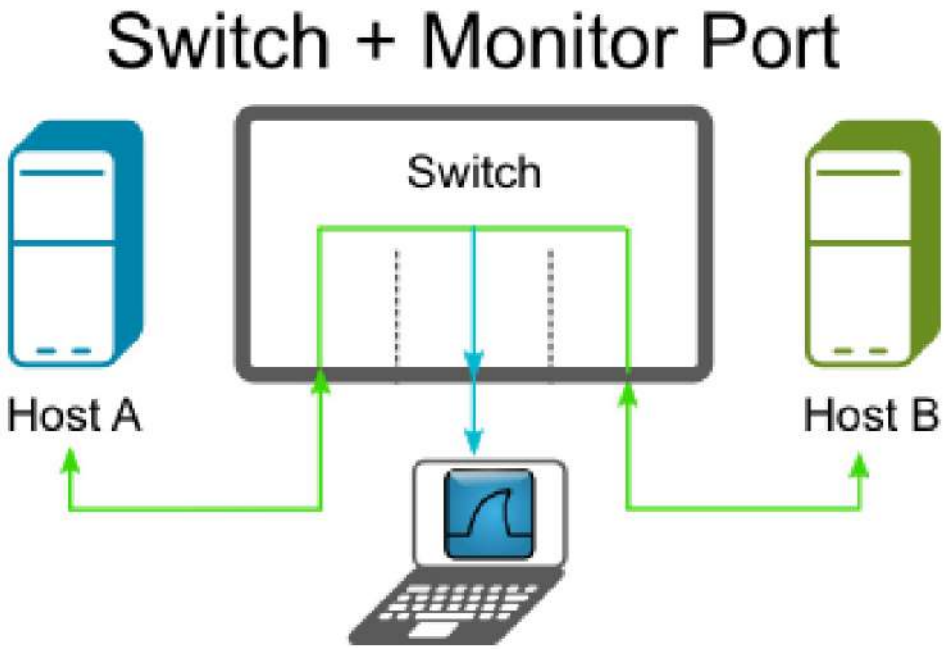
Podłączenie wiresharka do sieci

Podłączenie switch+hub



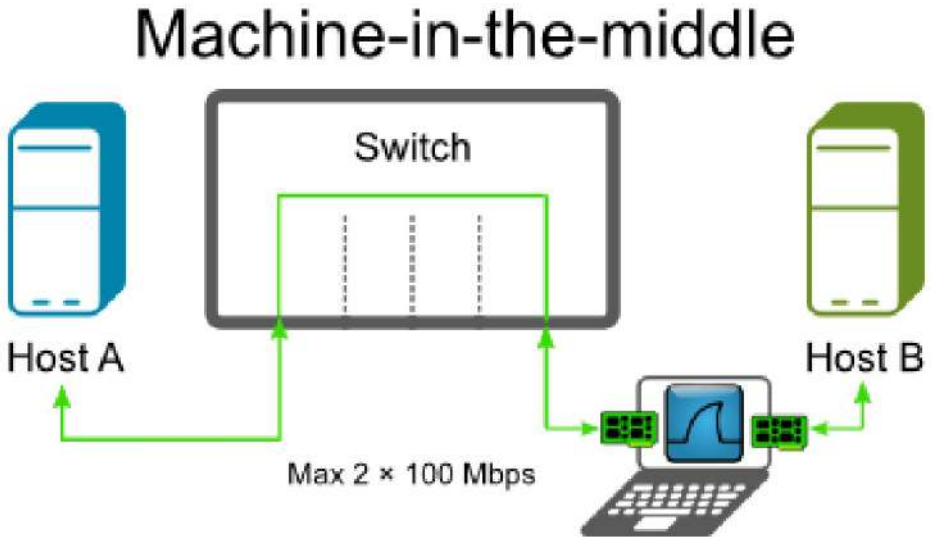
Podłączenie wiresharka do sieci

Podłączenie
switch+mirror port



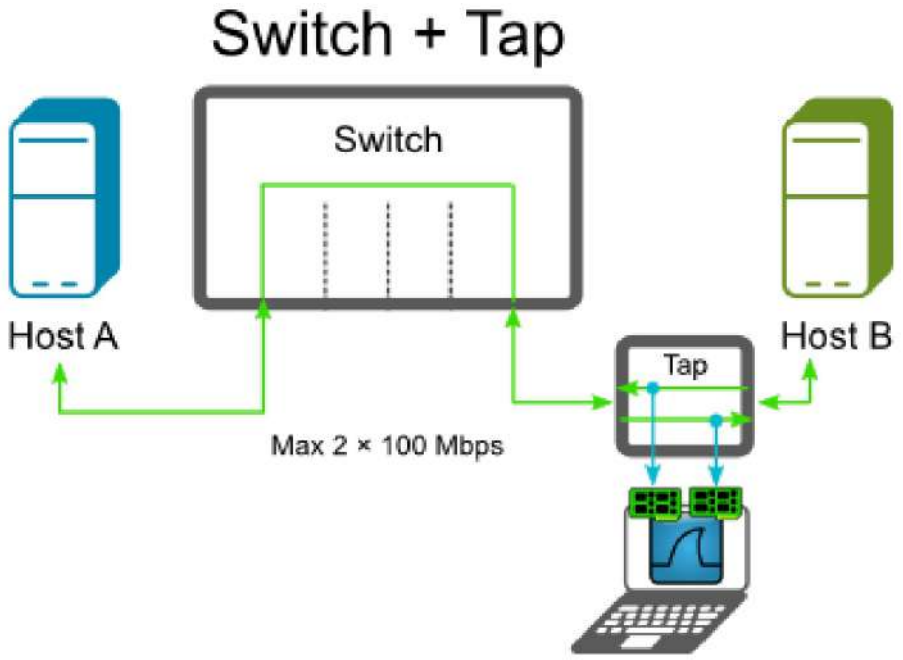
Podłączenie wiresharka do sieci

Podłączenie
„maszyna w środku”



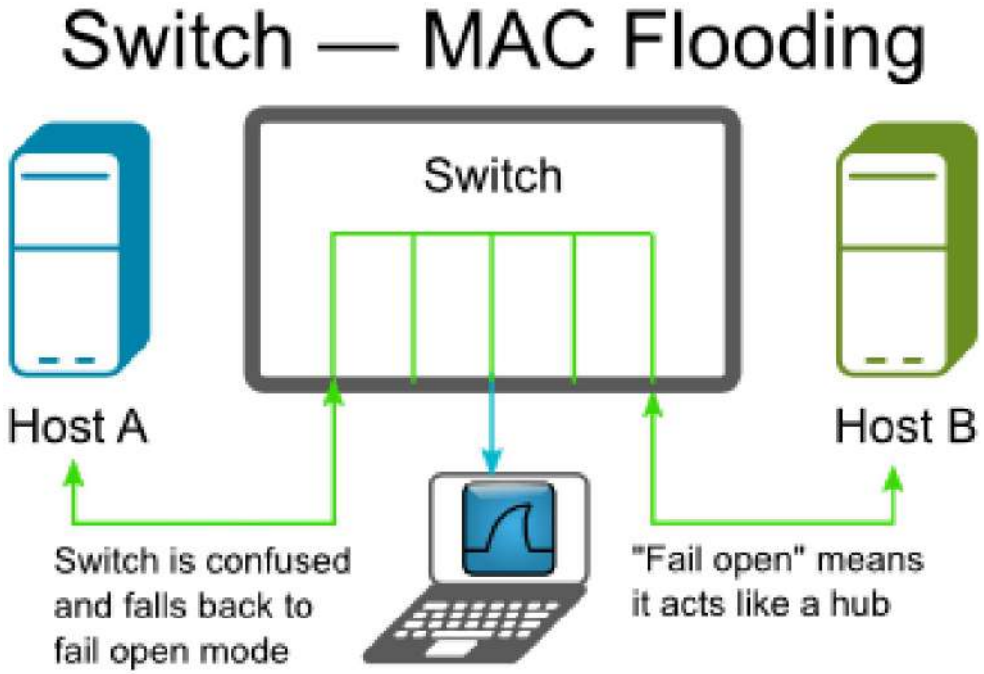
Podłączenie wiresharka do sieci

Podłączenie switch+tap



Podłączenie wiresharka do sieci

Podłączenie
switch+mac flooding



Przykład

Analiza danych ruchu sieciowego

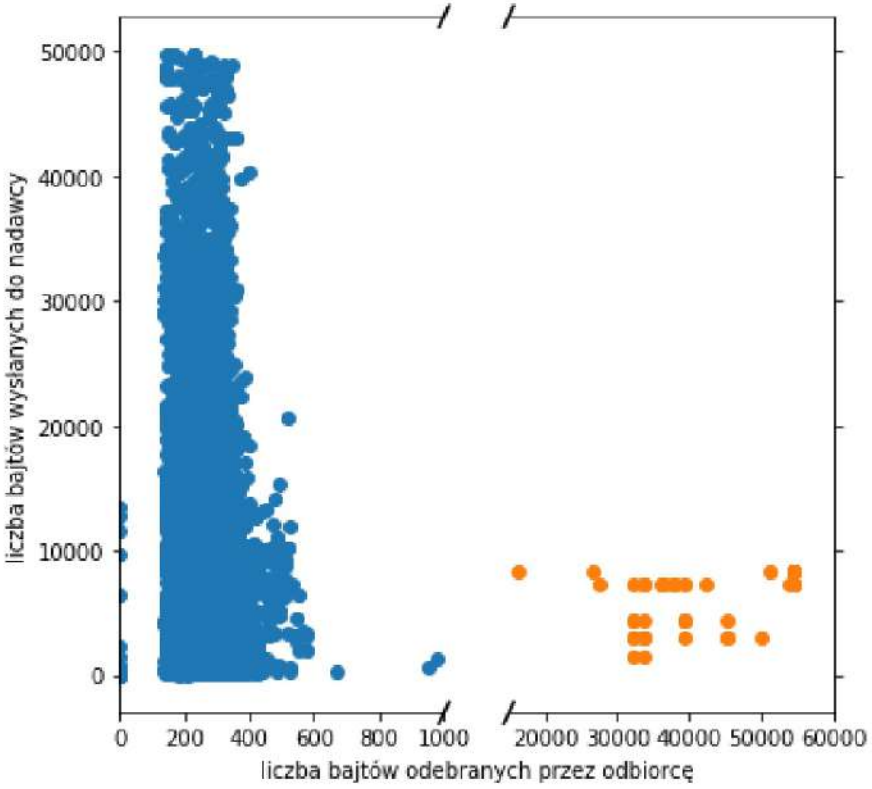
Zbiór danych:

$$\mathbf{X}=[\mathbf{x}^{(1)},\mathbf{x}^{(2)},...,\mathbf{x}^{(R)}]$$

gdzie:

$$\mathbf{x}^{(k)} = [x_1,...,x_i,...,x_j,...,x_n]$$

...
x_i - liczba bajtów odebranych przez odbiorcę
x_j - liczba bajtów wysłanych do nadawcy
...



Przykład

Analiza danych ruchu sieciowego

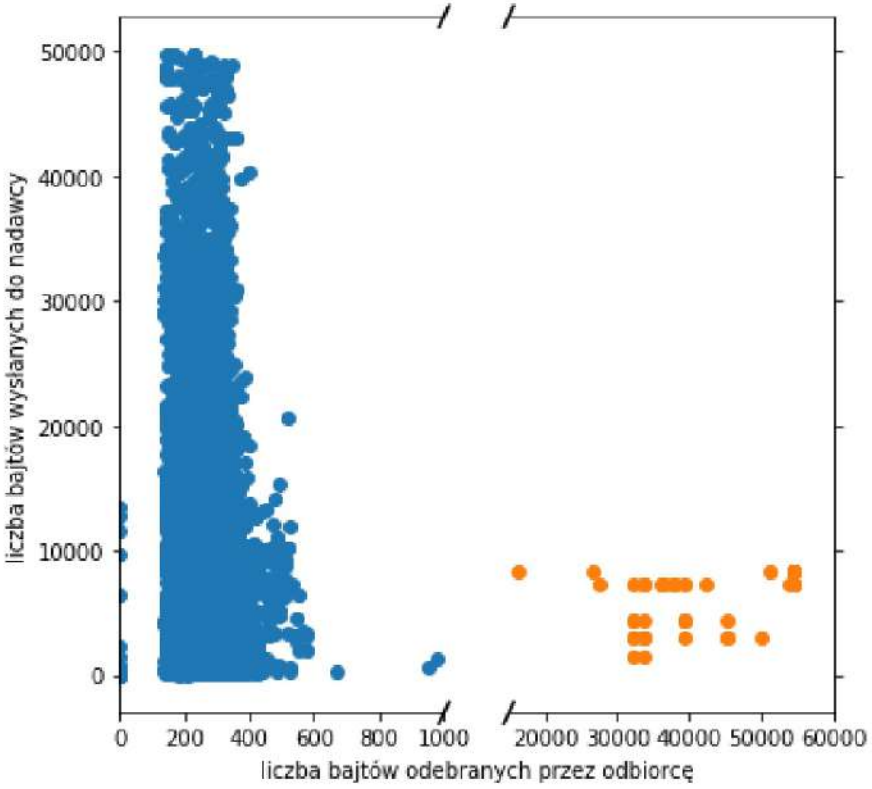
Zbiór danych:

$\mathbf{X}=[\mathbf{x}^{(1)},\mathbf{x}^{(2)},\ldots,\mathbf{x}^{(R)}]$

budujemy model: $P(\mathbf{x}_{test})$

$P(\mathbf{x}_{test}) \geq$ próg OK

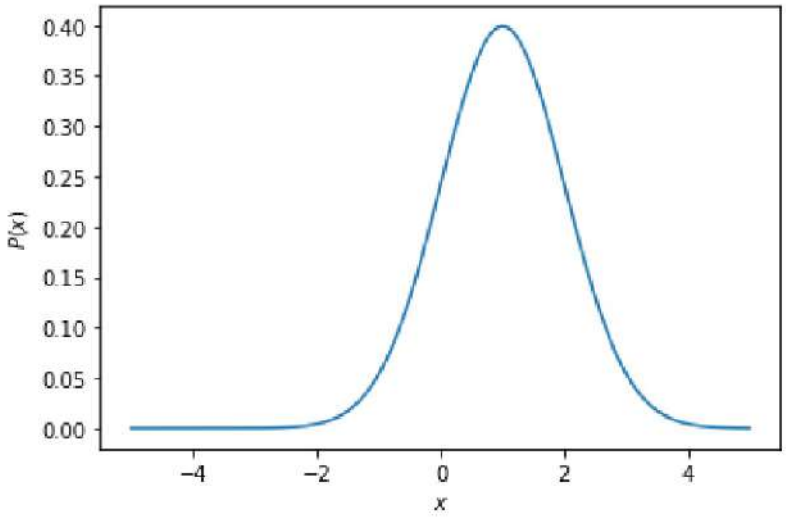
$P(\mathbf{x}_{test}) <$ próg anomalia

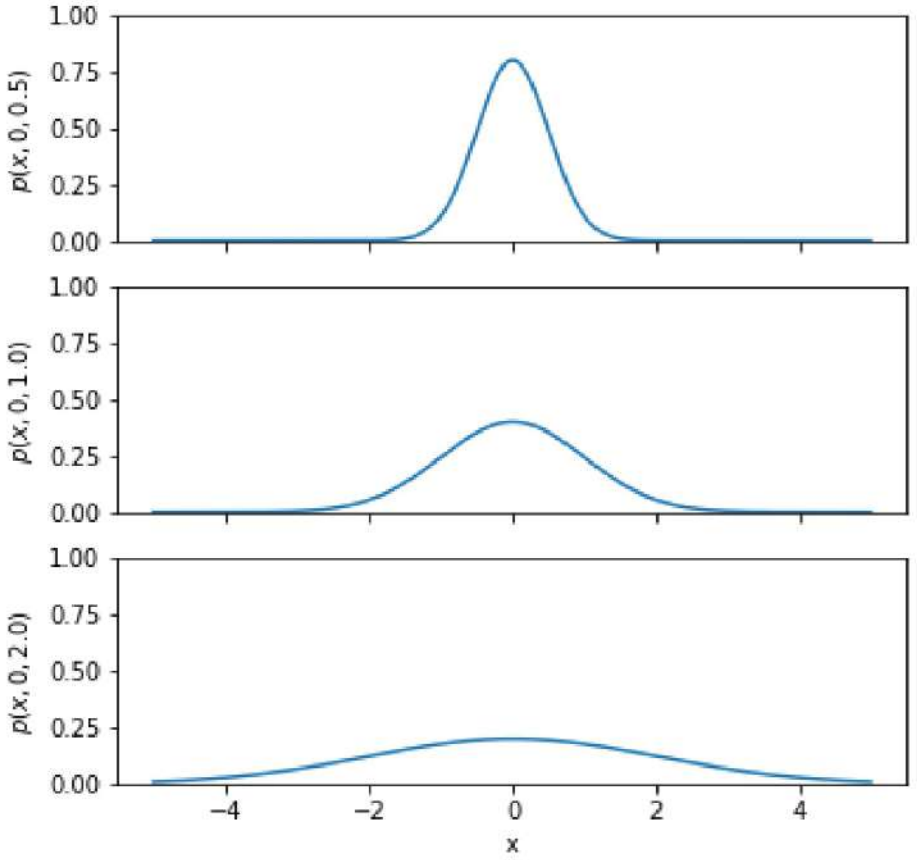


Rozkład Gaussowski x ~ N(μ,σ)

$$p(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

średnia wariancja





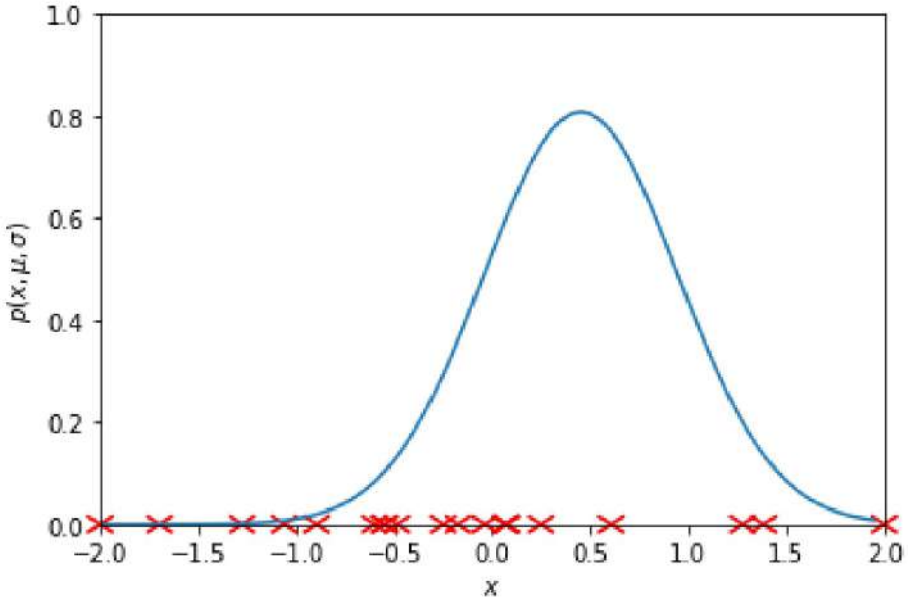
Rozkład Gaussowski - parametry

średnia arytmetyczna

$$\mu = \frac{1}{R}\sum_{i=1}^{R} x_i$$

odchylenie standardowe

$$\sigma^2 = \frac{1}{R}\sum_{i=1}^{R} \left(x_i - \mu\right)^2$$



Najprostszy algorytm wykrywania anomalii

1. Wybierz atrybuty, które mogą mieć znaczenie dla wykrycia anomalii
2. Przyjmując, że są one zgodne z rozkładem normalnym $N(\mu,\sigma)$, wyznacz parametry tego rozkładu dla każdego atrybutu osobno $N(\mu_i,\sigma_j)$.
3. Mając dany nowy przykład $\mathbf{x}^{test}$ oblicz:

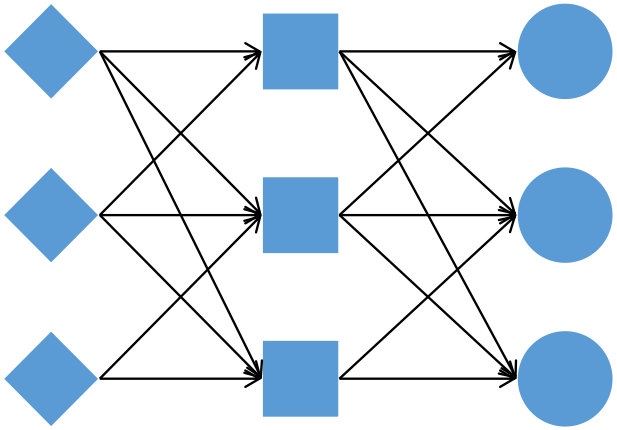
$$p(\mathbf{x}^{test}) = p(x_1^{test})\,p(x_2^{test})\ldots p(x_n^{test}) = \prod_{i=1}^{n} p(x_i^{test})$$

4. Określ przykład $\mathbf{x}^{test}$ jako anomalie jeżeli: $p(\mathbf{x}^{test}) < \text{próg}$

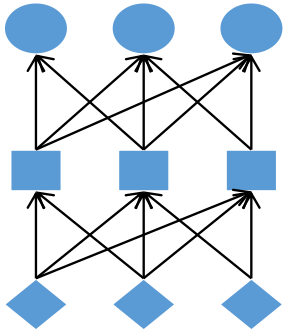
SI w wykrywaniu anomalii

- Grupowanie danych
- K najbliższych sąsiadów
- Isolation Forrest
- One-class SVM
- Autoencodery

Auto-encoder

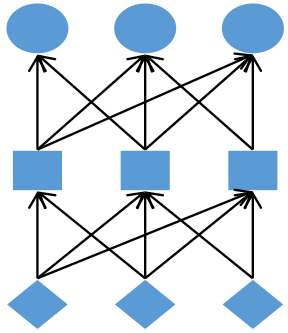


Wykrywanie anomalii - auto-encoder



Jeżeli nauczymy autoencoder na przykładach jednej klasy, to błąd otwarzania przykładów z innej będzie wysoki.

Im większy błąd odtwarzania, tym większe prawdopodobieństwo, że jest to anomalia



Grupowanie danych

Grupowanie danych

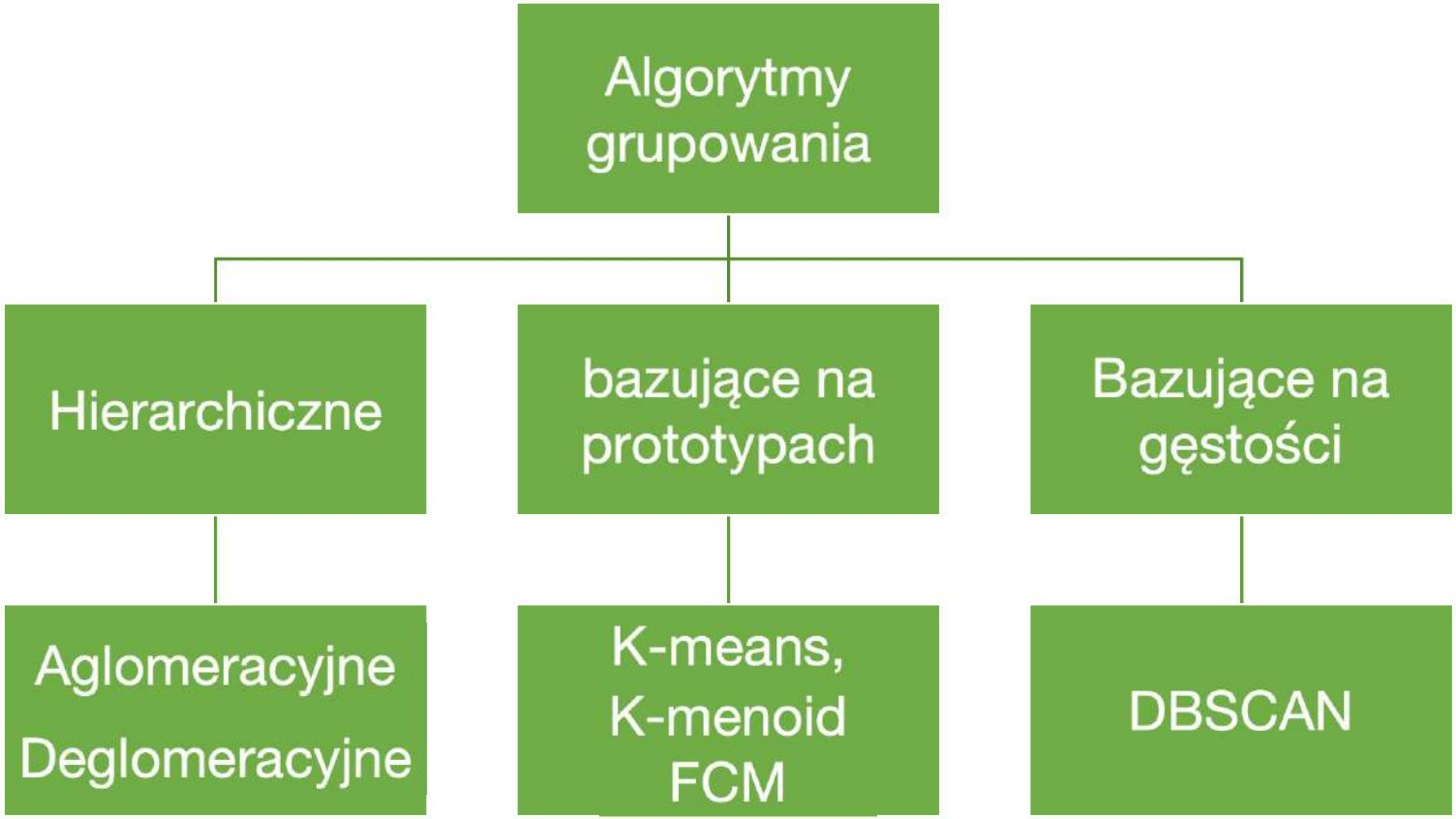
Grupowanie danych polega na podzieleniu obiektów na grupy zawierające obiekty podobne

Podobieństwo obiektów

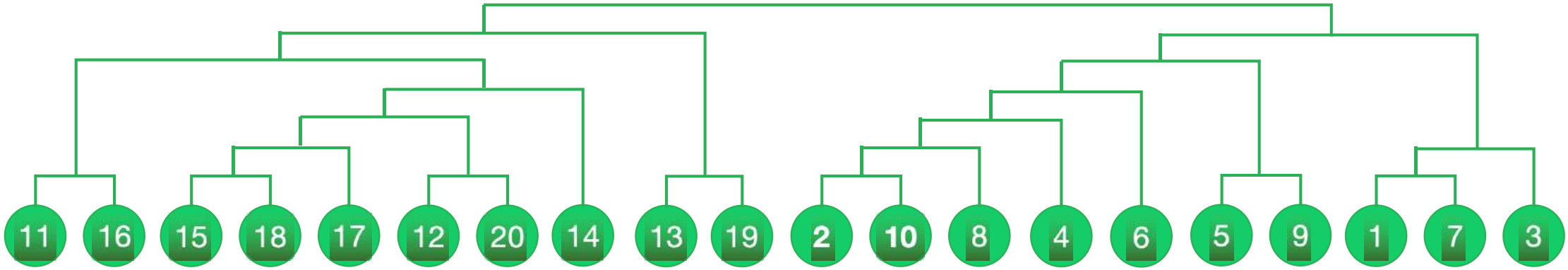
Podobieństwo obiektów

Nazwa	Wzór
Odległość Euklidesowa	$\|\mathbf{x}-\mathbf{y}\| = \sqrt{\sum_i (x_i - y_i)^2}$
Kwadratowa odległość Euklidesowa	$\|\mathbf{x}-\mathbf{y}\| = \sum_i (x_i - y_i)^2$
Odległość Manhattan	$\|\mathbf{x}-\mathbf{y}\| = \sum_i \|x_i - y_i\|$
Maksymalna odległość	$\|\mathbf{x}-\mathbf{y}\| = \max_i \|x_i - y_i\|$

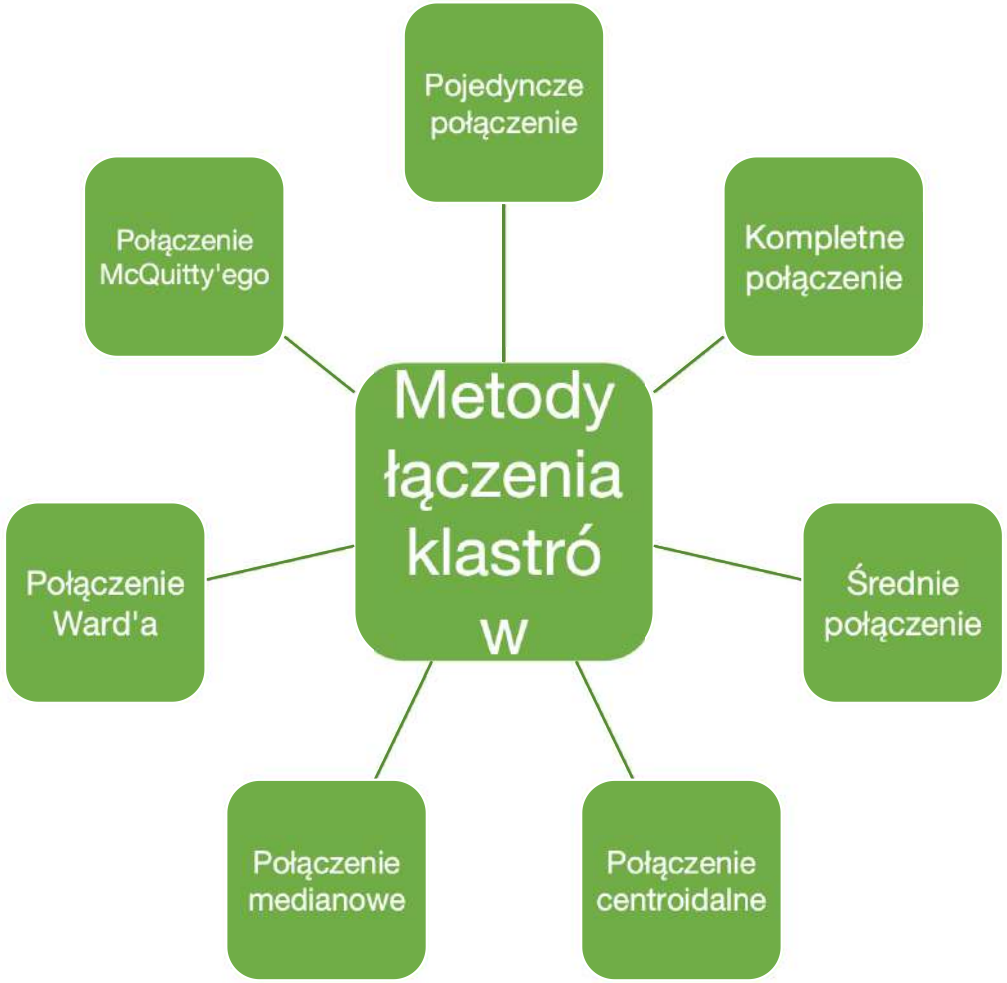
Algorytmy grupowania



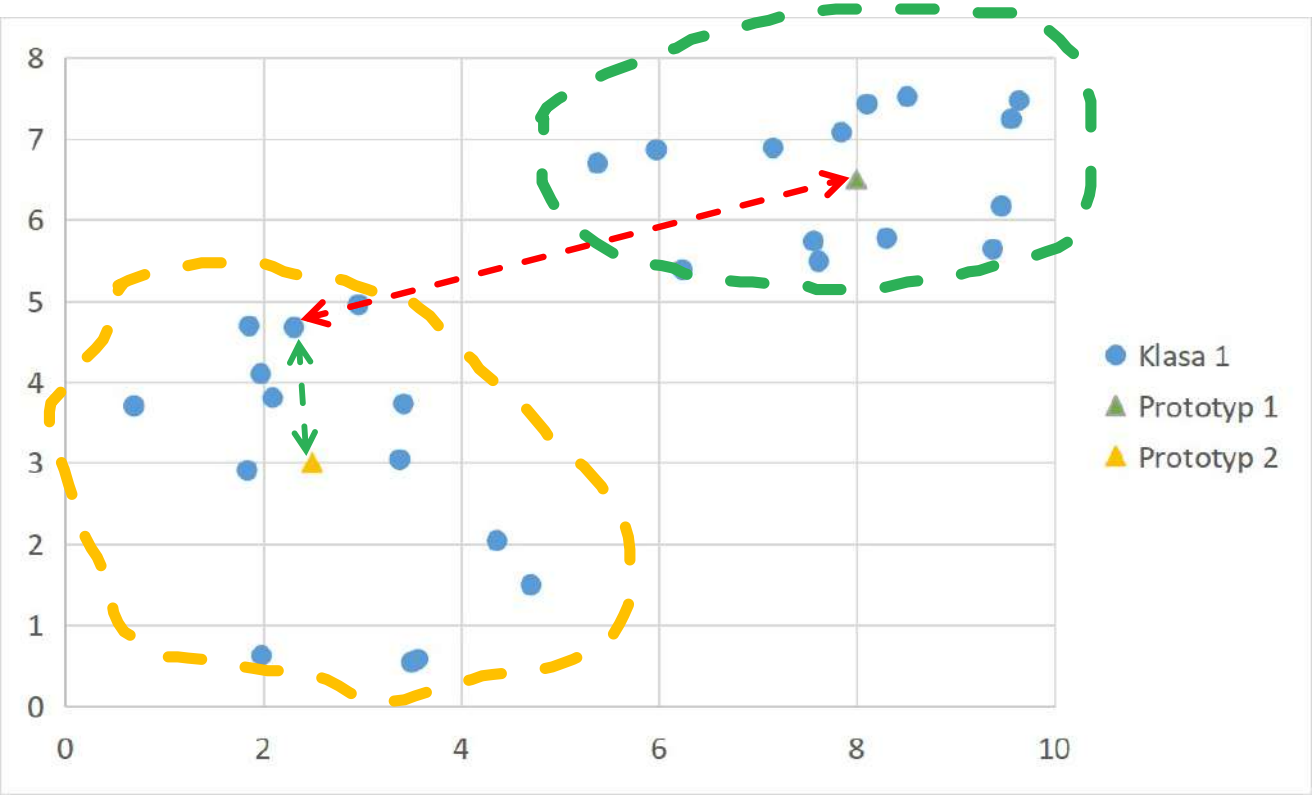
Grupowanie aglomeracyjne



Grupowanie hierarchiczne - łączenie obiektów



Grupowanie bazujące na prototypach



Algorytm HCM (K-means)

podziel $X_1, X_2, \ldots, X_n$ na grupy $C_1, C_2, \ldots, C_K$
while *warunki zatrzymania nie są spełnione* **do**
  for $j \leftarrow 1$ *do* K **do**
    wyznacz centroid c_j jako centrum klastra C_j
  end
  for $i \leftarrow 1$ *do* n **do**
    przypisz X_i do grupy C_j, tak że $d(X_i, c_j) \leq d(X_i, c_l) \quad \forall l \in 1, 2, \ldots, K$
  end
end

Algorytm HCM (K-means)

inicjalizuj $C^{(0)}$;

Powtarzaj

wyznacz $\mu_{ij}^{(t)} = \begin{cases} 1 & \text{jeżeli } d(X_i, c_j) \leq (X_i, c_l), \forall j \neq l \\ 0 & \text{w przeciwnym przypadku} \end{cases}$;

oblicz $c_j^{(t)} = \frac{\sum_{k=1}^{M} \mu_{ik}^{(t)} X_k}{\sum_{k=1}^{M} \mu_{ik}^{(t)}}$;

t=t+1;

dopóki $\|\mathbf{U}^{(t+1)} - \mathbf{U}^{(t)}\| < \epsilon$;

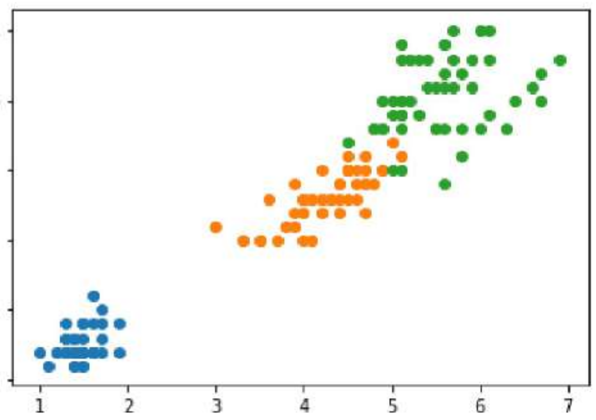
Algorytm K-menoid

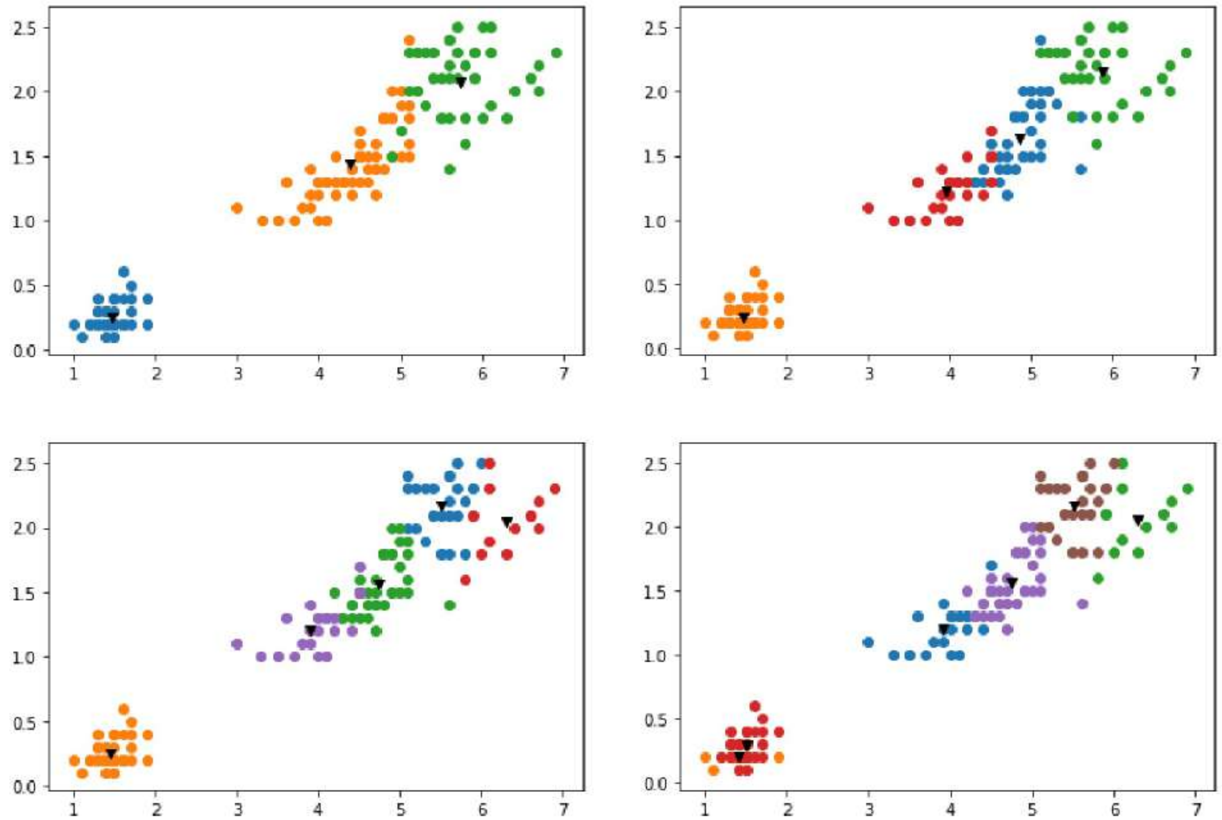
Algorytm K-Menoid zakłada, że centrum klastra jest jednym
z przykładów ze zbioru uczącego

Algorytm K-means++

1. Losowo wybierz pierwszy obiekt prototypowy z przykładów uczących
2. Dla każdego przykładu oblicz ich odległość od najbliższego wcześniej wybranego obiektu prototypowego
3. Wybierz następny obiekt prototypowy z przykładu, tak że prawdopodobieństwo wybrania punktu jako centroid jest wprost proporcjonalne do jego odległości od najbliższego klastra
4. Powtórz kroki 2 i 3 dopóki nie zostanie wybranych k obiektów prototypowych

Grupowanie danych





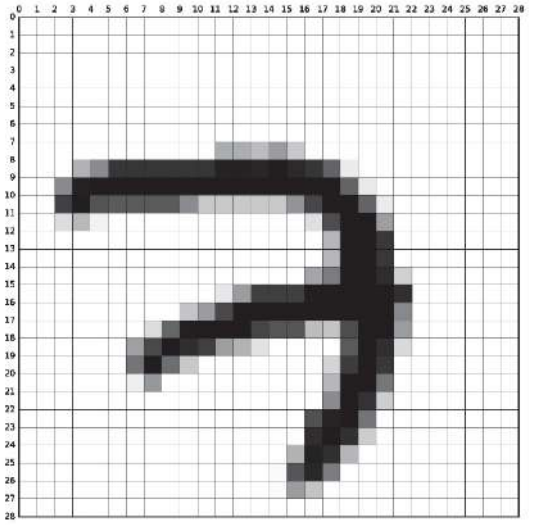
Grupowanie obrazów

Proces podziału zbioru obrazów na rozłączne podzbiory, które są jednorodne pod względem pewnych wybranych własności

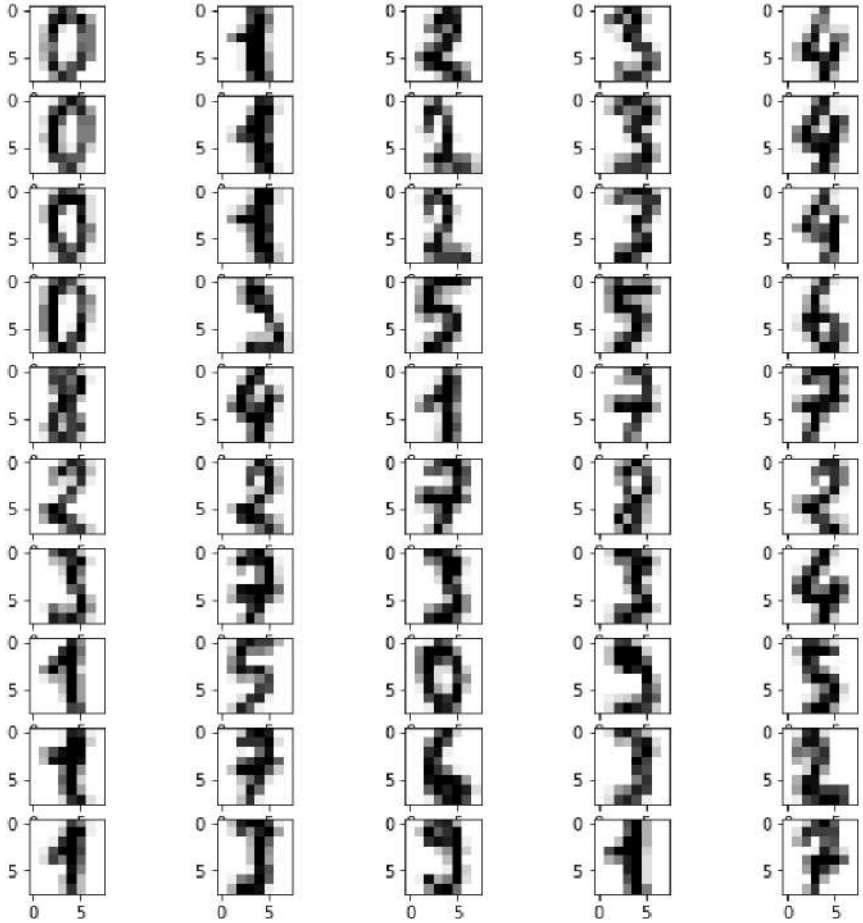
Baza MNIST (Modified National Institute of Standards and Technology database

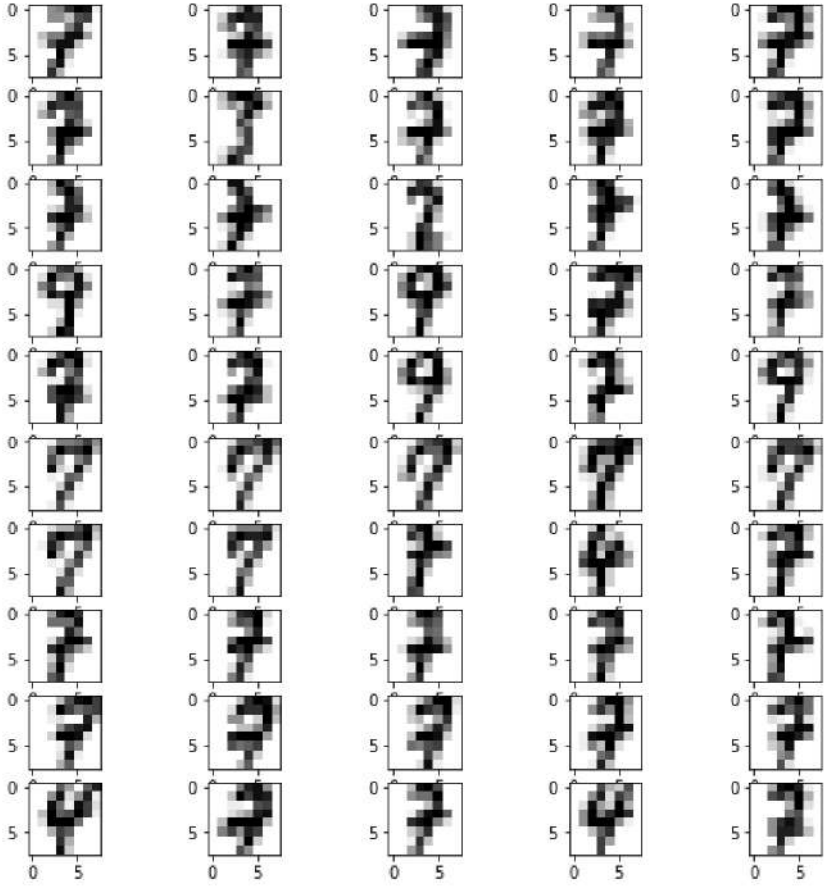
Zbiór danych: 60000 obrazów ręcznie napisanych cyfr
Zbiór znormalizowany:

- w odcieniach szarości
- rozmiar: 28x28

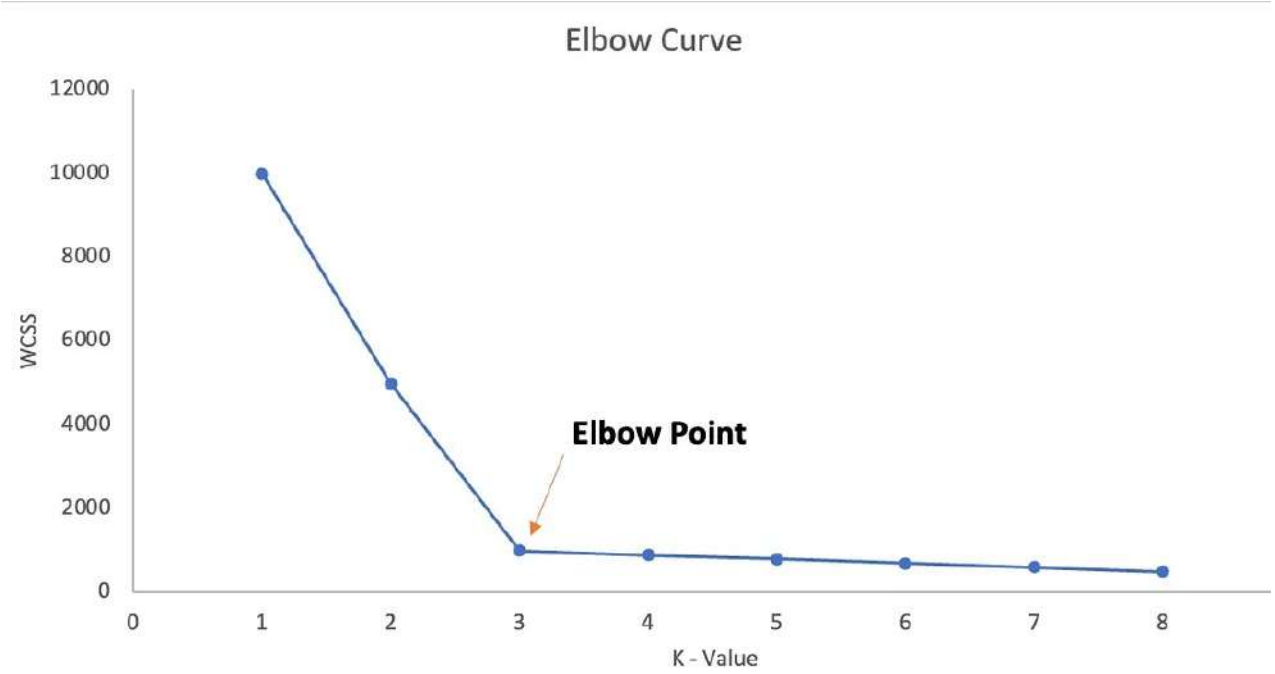


Grupowanie obrazów





Elbow point



WCSS – Within cluster sum of squares

Mądre na lewo, ładne na prawo

no przecież się nie rozerwę

Algorytm Fuzzy C-means

inicjalizuj $C^{(0)}$;

Powtarzaj

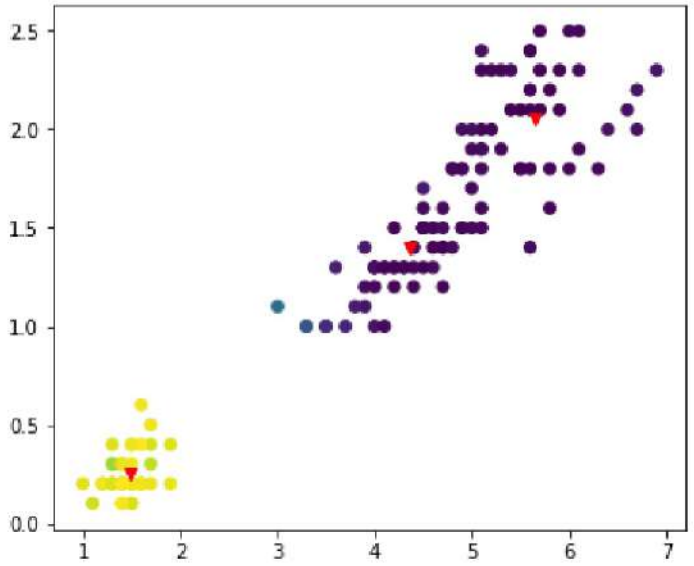
wyznacz $\mu_{ij}^{(t)} = \left(\sum_{j=1}^{C} \left(\frac{\|X_k - v_i^{t-1}\|}{\|X_k - v_j^{t-1}\|} \right)^{\frac{2}{m-1}} \right)^{-1}$;

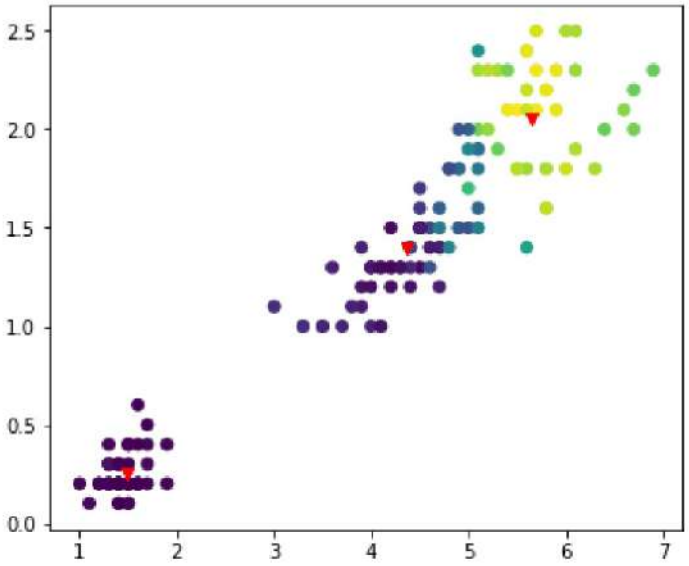
oblicz $c_j^{(t)} = \frac{\sum_{k=1}^{M} \mu_{ik}^{(t)} X_k}{\sum_{k=1}^{M} \mu_{ik}^{(t)}}$;

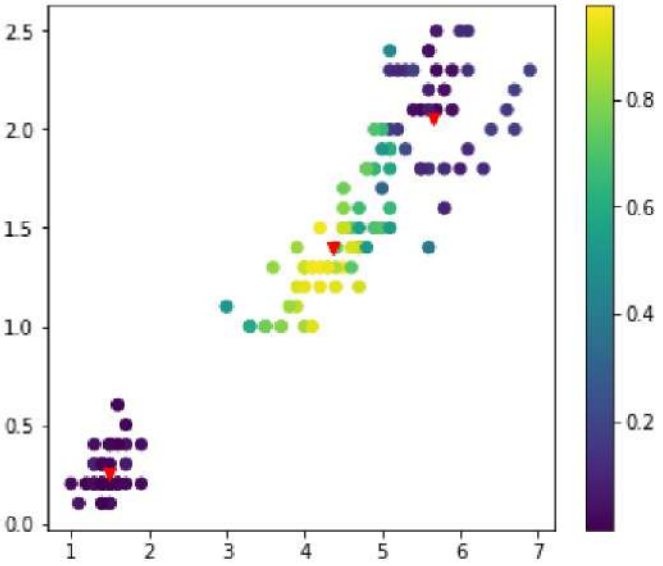
t=t+1;

dopóki $\|\mathbf{U}^{(t+1)} - \mathbf{U}^{(t)}\| < \epsilon$;

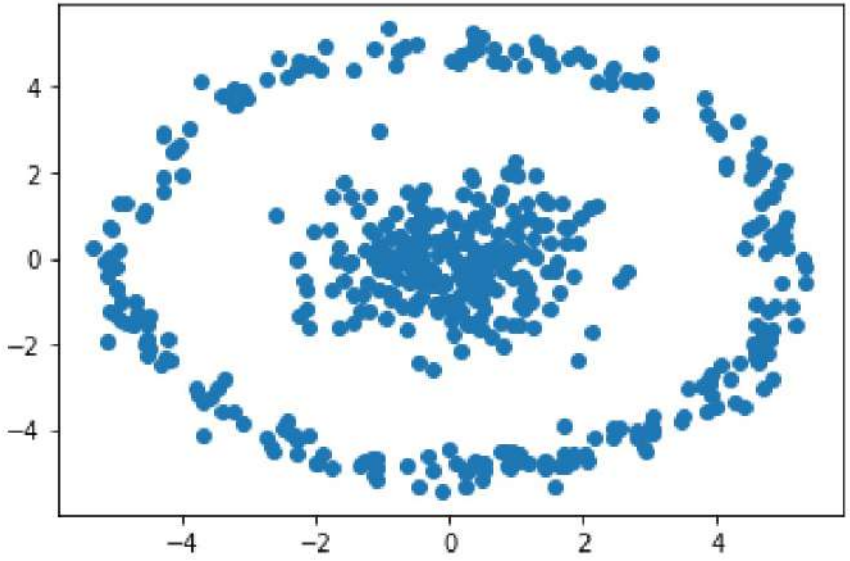
FCM - Results

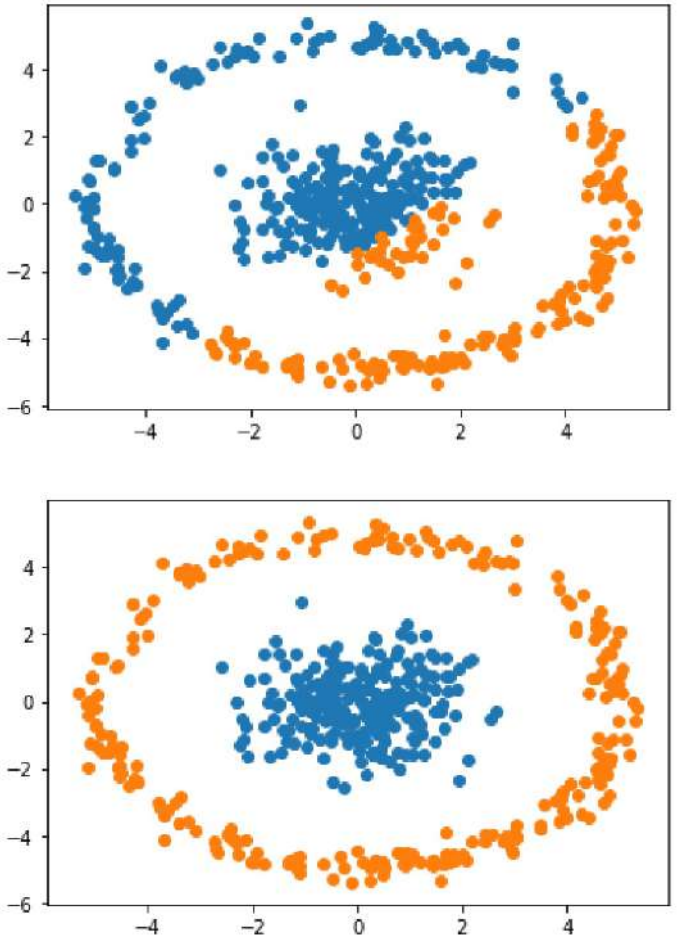

2.5
2.0
1.5
1.0
0.5
0.0
1
2
3
4
5
6
7


2.5
2.0
1.5
1.0
0.5
0.0
1
2
3
4
5
6
7


2.5
2.0
1.5
1.0
0.5
0.0
1
2
3
4
5
6
7
0.8
0.6
0.4
0.2

Grupowanie





DBSCAN

Algorytm DBSCAN (Density Based Spatial Clustering of Applications with Noise) wykorzystuje informacje o gęstości punktów w przestrzeni

- klastry mogą zostać zindetyfikowane jako zbiory obiektów położonych „niedaleko”.
- w przestrzeni pomiędzy klastrami mogą być rzadko położone punkty które traktuje się jako szum

DBSCAN

Parametry

- ε - promień sąsiedztwa obiektów
- minPts - minimalna liczba punktów w sąsiedztwie

Definicje

- Sąsiedztwo $S_{\mathbf{x}} = \{\mathbf{z} \in R^n : d(x, y) < \epsilon\}$
- Punkt rdzenny - punkt, w którego sąsiedztwie znajduje się conajmniej minPts pu $core(\mathbf{x}) = |S_{\mathbf{x}}| \geq minPts$
- Punkt bezpośrednio osiągalny - punkt z jest bezpośrednio osiągalny z x, jeżeli x jest punktem rdzennym i z należy do siąsiedztwa x $\mathbf{x} \rightarrow \mathbf{z} \iff core(\mathbf{x}) \wedge \mathbf{z} \in S_{\mathbf{x}}$

DBSCAN

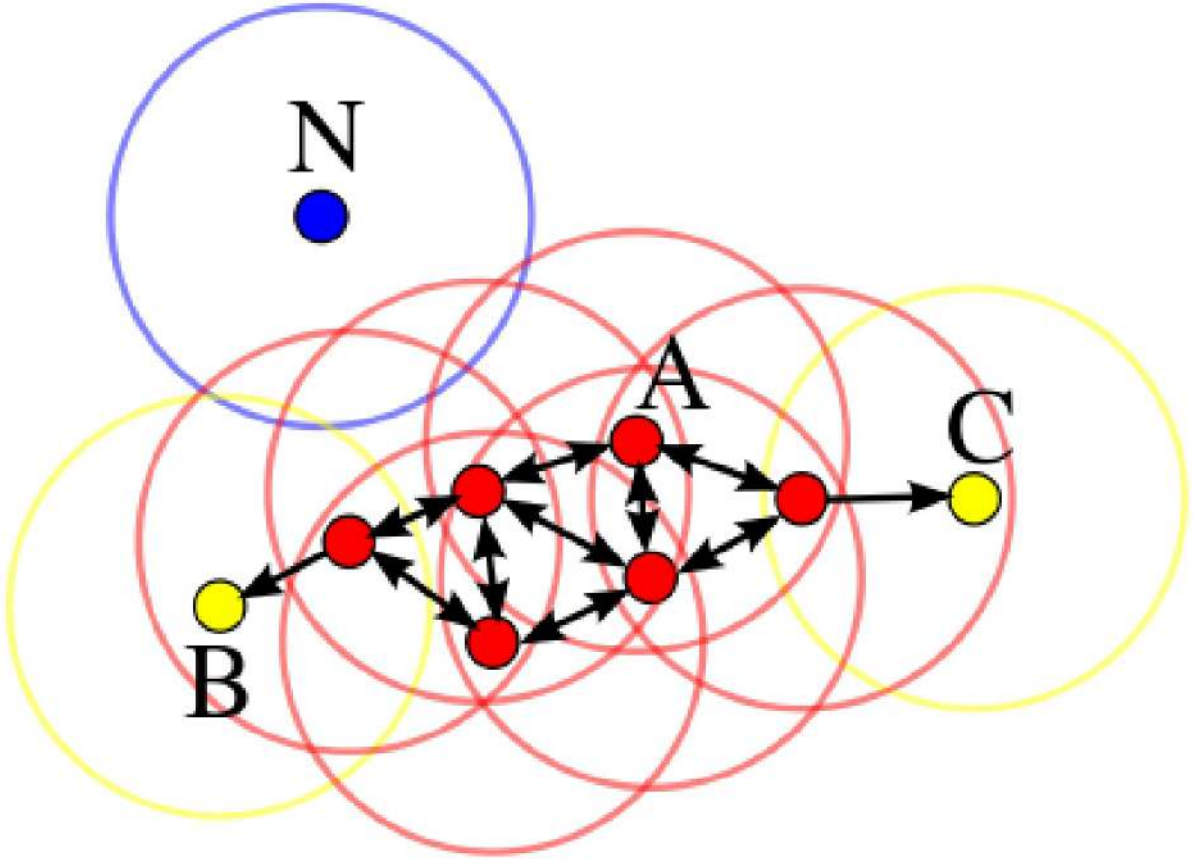
Definicje

- Punkt osiągalny - punkt z jest osiągalny z x, jeżeli istnieje ścieżka bazpośrednio osiągalnych punktów prowadząca od x do z

$$\mathbf{x} \rightarrow \mathbf{v}_1 \rightarrow \cdots \rightarrow \mathbf{v}_k \rightarrow \mathbf{z}$$

- Punkt brzegowy - jeżeli punkt x nie jest rdzennym, to jest brzegowym

DBSCAN

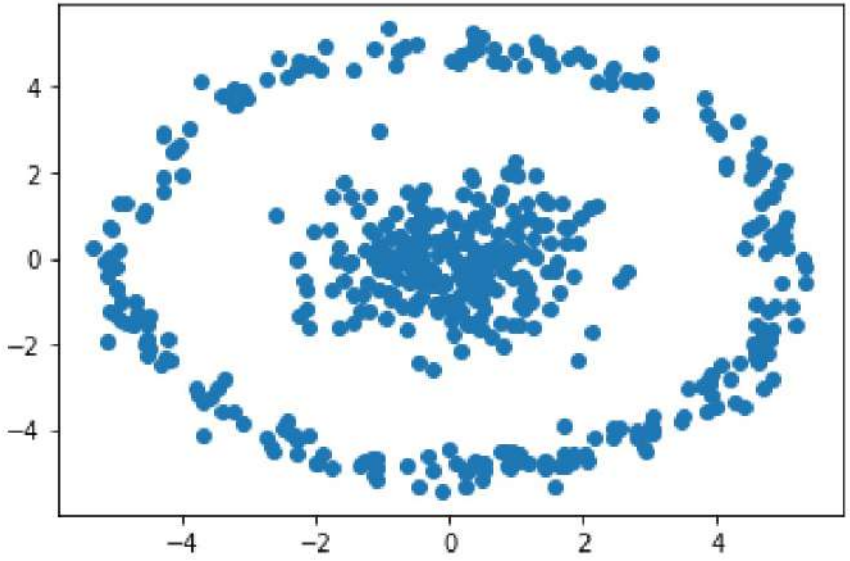


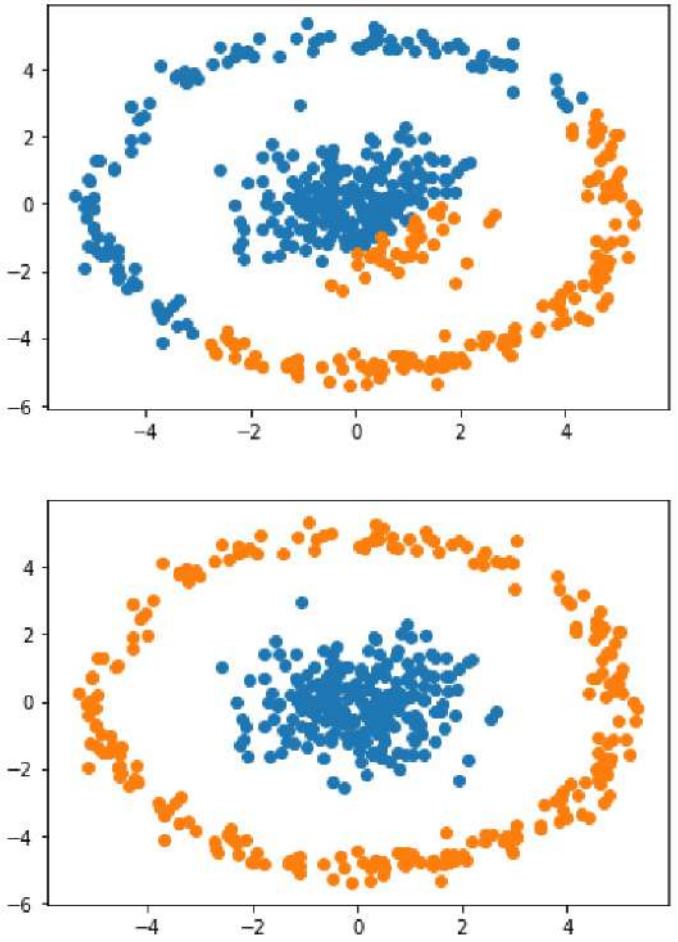
https://en.wikipedia.org/wiki/DBSCAN

DBSCAN

Dla każdego $p \in P$ **wykonaj**
- **Jeżeli** $etykieta(p) \neq nieokreslona$ **to**
  - **kontynuuj**;
- $S_p = OkrelSasiedztwo(P, odleglosc, p, r)$;
- **Jeżeli** $|S_p| < minPts$ **to**
  - $label(p) \leftarrow Szum$;
  - **kontynuuj**;
- $c \leftarrow NowyKlaster$;
- $etykieta(p) \leftarrow c$;
- $SS_p \leftarrow S_p \setminus \{p\}$;
- **Dla każdego** $q \in SS_p$ **wykonaj**
  - **Jeżeli** $etykieta(q) = Szum$ **to**
    - $etykieta(q) \leftarrow c$;
  - **Jeżeli** $etykieta(q) \neq nieokreslona$ **to**
    - **kontynuuj**;
  - $S_q = OkreslSasiedztwo(P, odleglosc, q, r)$;
  - $etykieta(q) \leftarrow c$;
  - **Jeżeli** $|S_q| < minPts$ **to**
    - **kontynuuj**;
  - $SS_p \leftarrow SS_p \cup S_q$;

KMeans, a DBSCAN





Podstawowe metody klasyfikacji

Zagadnienie klasyfikacji

Polega na stworzeniu modelu, który pozwoli na przypianie wartości atrybutu decyzyjnego do nieznanego przykładu

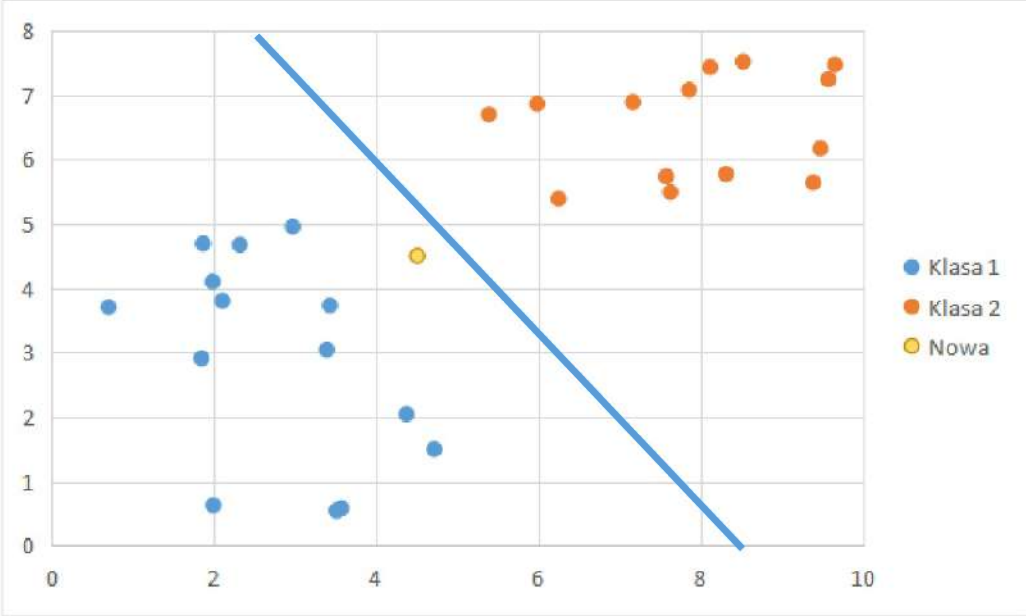
wektor obserwacji

atrybut decyzyjny $y_i \in \{c_1, \ldots, c_C\}$

Dane: $X = \{(\mathbf{x}_i, y_i)\} \qquad i = 1, \ldots, N$

Model: $\mathrm{C}(\mathbf{x}) : \mathbf{x} \rightarrow \{c_1, \ldots, c_C\}$

Zagadnienie klasyfikacji

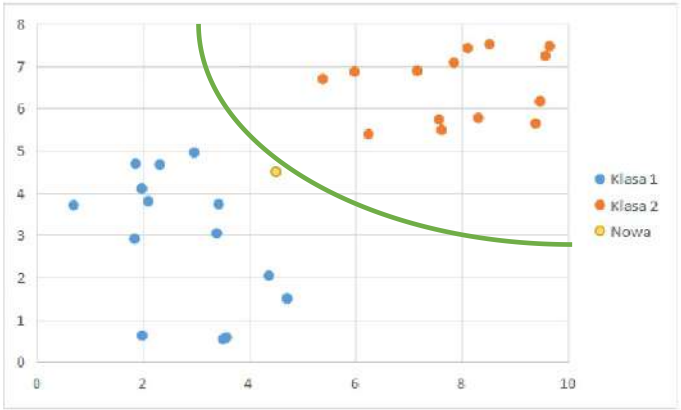


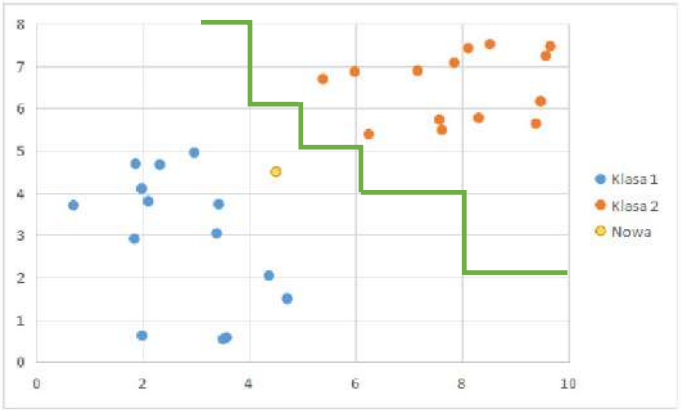
Region decyzyjny klasy c_i to obszar (zbiór obiektów), w którym model przypisze decyzję c_i

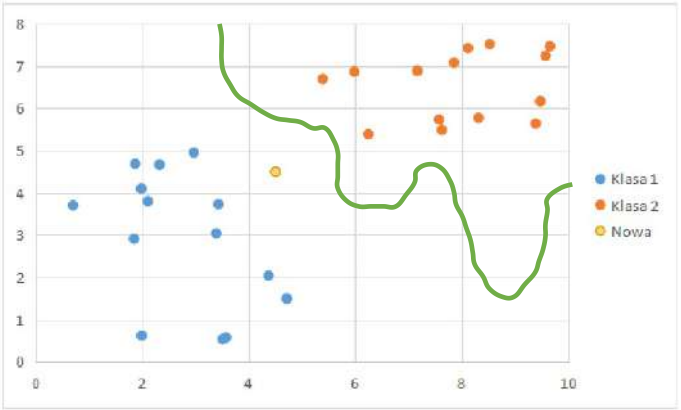
Granica klas to brzeg regionu decyzyjnego

Zagadnienie klasyfikacji

Granice klas mogą mieć różne kształty:
prostych, łamanych, krzywych czy zawierać wyspy



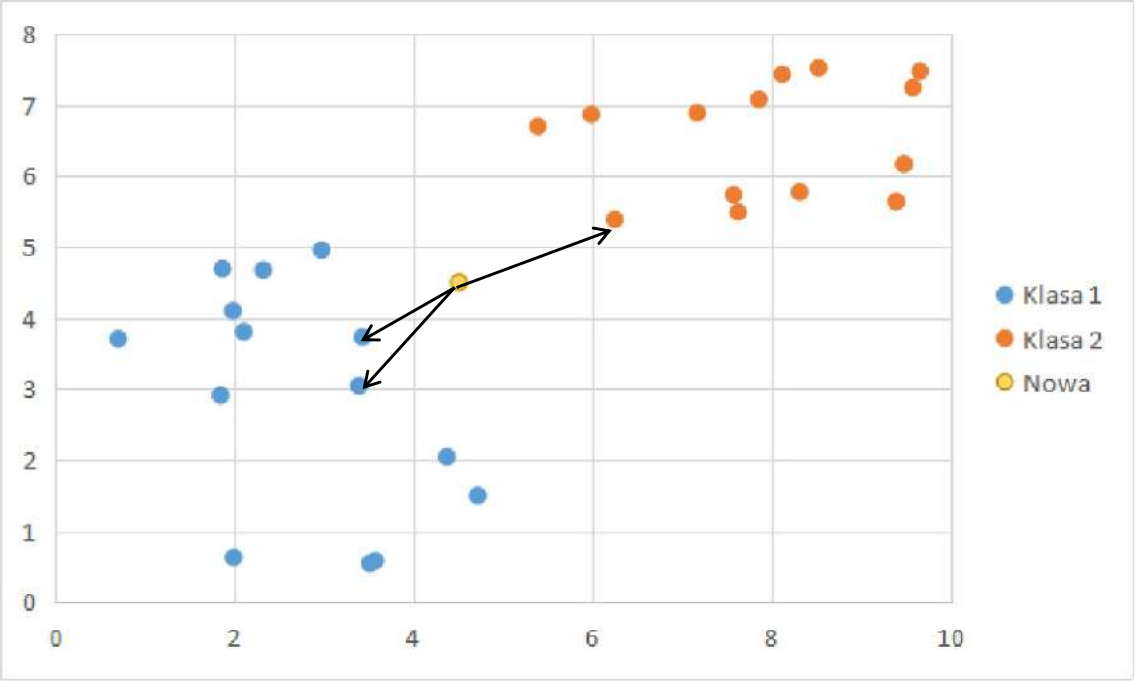




Algorytm k-NN

Algorytm k-NN

Algorytm k-NN (k-najbliższych sąsiadów) jest prostym klasyfikatorem, określającym klasę na podstawie k najbardziej podobnych obiektów



Algorytm k-NN

Problem 1 Jak mierzyć odległość między punktami?

Problem 2 Jakie k wybrać?

Problem 3 Czy wszystkie punkty powinny mieć tę samą wagę?

Jak mierzyć odległość?

Nazwa	Wzór
Odległość Euklidesowa	$\|\mathbf{x}-\mathbf{y}\| = \sqrt{\sum_i (x_i - y_i)^2}$
Kwadratowa odległość Euklidesowa	$\|\mathbf{x}-\mathbf{y}\| = \sum_i (x_i - y_i)^2$
Odległość Manhattan	$\|\mathbf{x}-\mathbf{y}\| = \sum_i \|x_i - y_i\|$
Maksymalna odległość	$\|\mathbf{x}-\mathbf{y}\| = \max_i \|x_i - y_i\|$

Odległość

	Liczba połączeń	Łączny czas [m]
1	51	6570
2	33	6795
3	51	5956
4	36	2575
5	35	3080
6	43	2830
7	57	2527
8	42	5883
9	57	6502
10	42	2478

$$d(X_2, X_4) = 4220{,}001066$$

$$d(X_3, X_5) = 2876{,}044506$$

Normalizacja i standaryzacja

Atrybuty o dużych wartościach mogą dominować nad pozostałymi przy obliczaniu odległości

Normalizacja

$$X^* = \frac{X - \min(X)}{\max(X) - \min(X)}$$

Standaryzacja

$$X^* = \frac{X - \textit{średnia}\,(X)}{\sigma(X)}$$

Normalizacja

	Liczba połączeń	Łączny czas [m]
1	0,75	0,947880473
2	0	1
3	0,75	0,805652073
4	0,125	0,022469307
5	0,083333333	0,139448691
6	0,416666667	0,081538105
7	1	0,011350475
8	0,375	0,788742182
9	1	0,932128793
10	0,375	0

$$X^* = \frac{X - \min(X)}{\max(X) - \min(X)}$$

$$d(X_2, X_4) = 0{,}985490363$$

$$d(X_3, X_5) = 0{,}942481507$$

Standaryzacja

	Liczba połączeń	Łączny czas [m]
1	0,711260513	1,053670052
2	-1,320912381	1,169294204
3	0,711260513	0,738144588
4	-0,982216898	-0,999301006
5	-1,095115392	-0,73978902
6	-0,19192744	-0,8682603
7	1,388651477	-1,023967492
8	-0,304825934	0,700630974
9	1,388651477	1,018725864
10	-0,304825934	-1,049147863

$$X^* = \frac{X - średnia\,(X)}{\sigma(X)}$$

$$d(X_2, X_4) = 2{,}194884921$$

$$d(X_3, X_5) = 2{,}33394122\ 9$$

Jakie k wybrać?

Wybór odpowiedniej wartości parametru k jest bardziej sztuką niż nauką i zależy od rozwiązywanego problemu

Funkcja decyzyjna

Proste głosowanie

Głosowanie ważone

Metody probabilistyczne

Prawdopodobieństwo warunkowe

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Prawdopodobieństwo warunkowe

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

$$P(A|B) \approx P(B|A)P(A)$$

prawdopodobieństwo końcowe (a posteriori)

wiarygodność, prawopodobieństwo wystąpienia

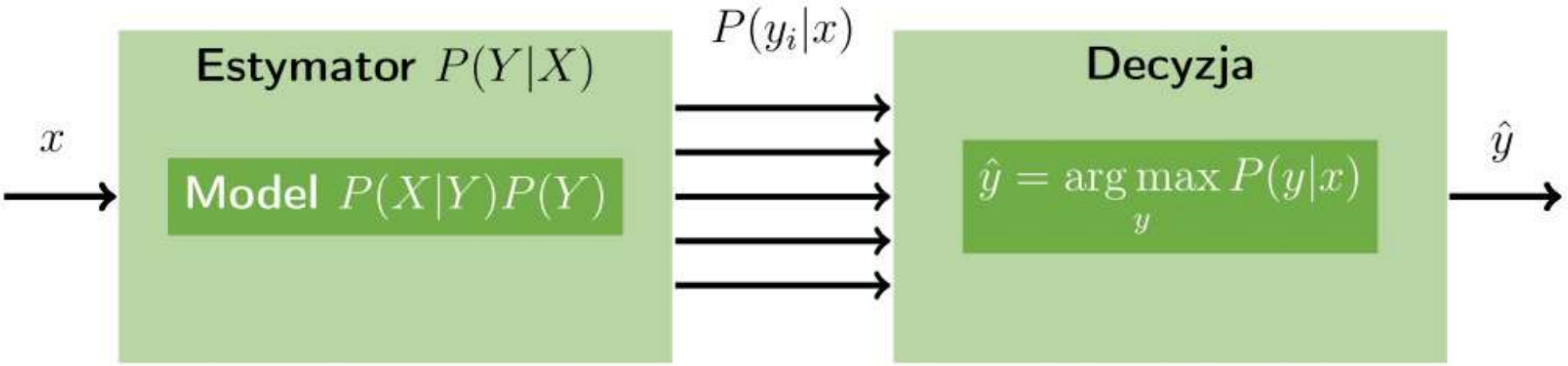
prawdopodobieństwo wstępne (a priori)

Prawdopodobieństwo warunkowe

$$P(Y = y_k|X = x) = \frac{P(X = x|Y = y_k)P(Y = y_k)}{P(X = x)}$$

$$P(Y = y_k|X = x) \approx P(X = x|Y = y_k)P(Y = y_k)$$

Naiwny klasyfikator Bayesa



Naiwny klasyfikator Bayesa

Nazywany jest naiwnym ponieważ, zakłada że dla danej klasy wszystkie cechy są od siebie niezależne

Co w rzeczywistym świecie jest bardzo mocnym uproszczeniem.

$$P(X_1|X_2, Y) = P(X_1, Y)$$

Naiwny klasyfikator Bayesa

$$P(Y \mid X_1, X_2, \ldots, X_n) = \frac{P(X_1, X_2, \ldots, X_n \mid Y) * P(Y)}{P(X_1, X_2, \ldots, X_n)}$$

$$P(X_1, X_2, \ldots, X_n|Y) = \prod_{i=1}^{n} P(X_i|Y)$$

Naiwny klasyfikator Bayesa - przykład

Usługa	Liczba danych odebranych	Liczba danych wysłanych	Konto administ-ratora	decyzja
ftp	duża	duża	fałsz	nie
ftp	duża	duża	prawda	nie
www	duża	duża	fałsz	tak
telnet	średnia	duża	fałsz	tak
telnet	mała	normalna	fałsz	tak
telnet	mała	normalna	prawda	nie
www	mała	normalna	prawda	tak
ftp	mała	duża	fałsz	nie
ftp	mała	normalna	fałsz	tak
telnet	mała	normalna	fałsz	tak
ftp	mała	normalna	prawda	tak
www	mała	duża	prawda	tak
www	duża	normalna	fałsz	tak
telnet	mała	duża	prawda	nie

$$P(TAK) = 9/14 \approx 0{,}64$$

$$P(NIE) = 5/14 \approx 0{,}36$$

Naiwny klasyfikator Bayesa - przykład

	Usługa						Liczba danych odebranych						Liczba danych wysłanych				Konto administratora			
	ftp		www		telnet		duża		średnia		mała		duża		normalna		Tak		Nie	
	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie
#	2	3	4	0	3	2	2	2	4	2	3	1	3	4	6	1	3	3	6	2
	Tak			Nie			Tak			Nie			Tak			Nie	Tak			Nie
#	9			5			9			5			9			5	9			5
PW	0.22	0.6	0.44	0.0	0.33	0.4	0.22	0.4	0.44	0.4	0.33	0.2	0.33	0.8	0.66	0.2	0.33	0.6	0.66	0.4

$$P(TAK|ftp,duża,duża,tak) = 0.22*0.22*0.33*0.33*0.64 = 0,0033$$

$$P(NIE|ftp,duża,duża,tak) = 0.6*0.4*0.8*0.6*0.36 = 0,041$$

Naiwny klasyfikator Bayesa - przykład

	Usługa						Liczba danych odebranych						Liczba danych wysłanych				Konto administratora			
	ftp		www		telnet		duża		średnia		mała		duża		normalna		Tak		Nie	
	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie
#	2	3	4	0	3	2	2	2	4	2	3	1	3	4	6	1	3	3	6	2
	TAK		Nie				Tak		Nie				Tak		Nie		Tak		Nie	
#	9		5				9		5				9		5		9		5	
PW	0.22	0.6	0.44	0.0	0.33	0.4	0.22	0.4	0.44	0.4	0.33	0.2	0.33	0.8	0.66	0.2	0.33	0.6	0.66	0.4

$$P(TAK|ftp,mała,duża,tak) = 0.22*0.33*0.33*0.33*0.64 = 0,005$$

$$P(NIE|ftp,mała,duża,tak) = 0.6*0.2*0.8*0.6*0.36 = 0,020$$

Naiwny klasyfikator Bayesa - przykład

	Usługa						Liczba danych odebranych						Liczba danych wysłanych				Konto administratora			
	ftp		www		telnet		duża		średnia		mała		duża		normalna		Tak		Nie	
	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie	Tak	Nie
#	2	3	4	0	3	2	2	2	4	2	3	1	3	4	6	1	3	3	6	2
	TAK		Nie				Tak		Nie				Tak		Nie		Tak		Nie	
#	9		5				9		5				9		5		9		5	
PW	0.22	0.6	0.44	0.0	0.33	0.4	0.22	0.4	0.44	0.4	0.33	0.2	0.33	0.8	0.66	0.2	0.33	0.6	0.66	0.4

$$P(NIE|www,mała,duża,tak) = 0.0*0.2*0.8*0.6*0.36 = 0$$

Naiwny klasyfikator Bayesa

Jeżeli w zbiorze uczącym nie występuje kombinacja, wartość atrybutu i wartość decyzji, to przypisane prawdopodobieństwo zawsze będzie równe 0.

Rozwiązaniem tego problemu jest zastosowanie wygładzania, czyli zastąpnie 0 w liczniku określoną wartością

Przy czym należy pamiętać, o sumowaniu wartości prawdopodobieństw do 1 dlatego ogólny wzór na wygładzone prawdopodobieństwo warunkowe można przedstawić następująco:

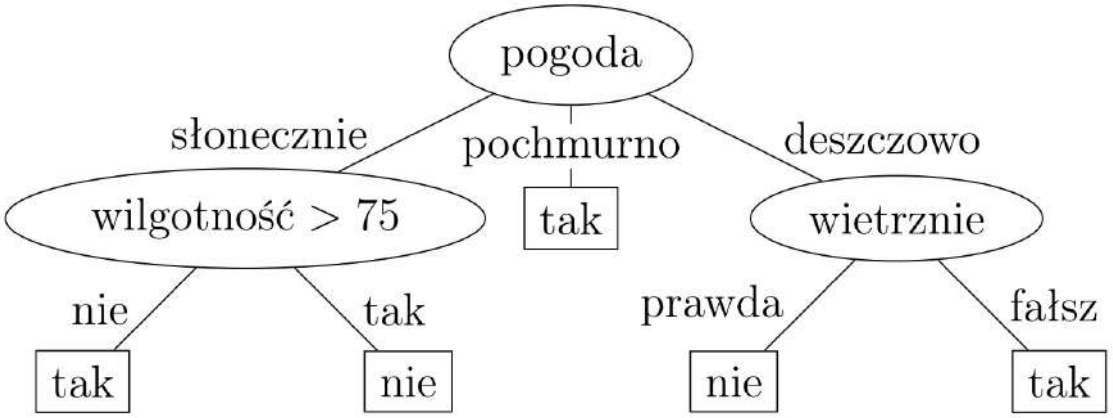
$$P(Y = y_k|X) = \frac{P(X, Y = c_k) + \alpha}{P(Y = c_k) + K\alpha}$$

Drzewa decyzyjne

Drzewa decyzyjne

Graficzna metoda wspomagania procesu

Tworzenie klasyfikatora na podstawie przykładów



Drzewa decyzyjne

Usługa	L. d. o.	L. d. w.	K. admin.	decyzja
ftp	duża	duża	fałsz	nie
ftp	duża	duża	prawda	nie
www	duża	duża	fałsz	tak
telnet	średnia	duża	fałsz	tak
telnet	mała	normalna	fałsz	tak
telnet	mała	normalna	prawda	nie
www	mała	normalna	prawda	tak
ftp	średnia	duża	fałsz	nie
ftp	mała	normalna	fałsz	tak
telnet	średnia	normalna	fałsz	tak
ftp	średnia	normalna	prawda	tak
www	średnia	duża	prawda	tak
www	duża	normalna	fałsz	tak
telnet	średnia	duża	prawda	nie

Łącznie: 14 przykładów

Usługa = {ftp, www, telnet}
L. d. o. = {średnia, duża, mała}
L. d. w. = {duża, normalna}
K. admin. = {fałsz, prawda}
decyzja = {tak (9), nie(5)}

Drzewa decyzyjne

Usługa	L. d. o.	L. d. w.	K. admin.	decyzja
telnet	średnia	duża	fałsz	tak
telnet	mała	normalna	fałsz	tak
telnet	mała	normalna	prawda	nie
telnet	średnia	normalna	fałsz	tak
telnet	średnia	duża	prawda	nie
www	duża	duża	fałsz	tak
www	mała	normalna	prawda	tak
www	średnia	duża	prawda	tak
www	duża	normalna	fałsz	tak
ftp	duża	duża	fałsz	nie
ftp	duża	duża	prawda	nie
ftp	średnia	duża	fałsz	nie
ftp	mała	normalna	fałsz	tak
ftp	średnia	normalna	prawda	tak

decyzja = {tak (3), nie (2)}

decyzja = {tak (4), nie (0)}

decyzja = {tak (2), nie (3)}

Drzewa decyzyjne

Usługa	L. d. o.	L. d. w.	K. admin.	decyzja
telnet	**średnia**	duża	fałsz	tak
telnet	**średnia**	normalna	fałsz	tak
telnet	**średnia**	duża	prawda	nie
www	**średnia**	duża	prawda	tak
ftp	**średnia**	duża	fałsz	nie
ftp	**średnia**	normalna	prawda	tak
www	**duża**	duża	fałsz	tak
www	**duża**	normalna	fałsz	tak
ftp	**duża**	duża	fałsz	nie
ftp	**duża**	duża	prawda	nie
telnet	**mała**	normalna	fałsz	tak
telnet	**mała**	normalna	prawda	nie
www	**mała**	normalna	prawda	tak
ftp	**mała**	normalna	fałsz	tak

decyzja = {tak (4), nie (2)}

decyzja = {tak (2), nie (2)}

decyzja = {tak (3), nie (1)}

Drzewa decyzyjne

Usługa	L. d. o.	L. d. w.	K. admin.	decyzja
telnet	średnia	**normalna**	fałsz	tak
ftp	średnia	**normalna**	prawda	tak
www	duża	**normalna**	fałsz	tak
telnet	mała	**normalna**	fałsz	tak
telnet	mała	**normalna**	prawda	nie
www	mała	**normalna**	prawda	tak
ftp	mała	**normalna**	fałsz	tak
telnet	średnia	**duża**	fałsz	tak
telnet	średnia	**duża**	prawda	nie
www	średnia	**duża**	prawda	tak
ftp	średnia	**duża**	fałsz	nie
www	duża	**duża**	fałsz	tak
ftp	duża	**duża**	fałsz	nie
ftp	duża	**duża**	prawda	nie

decyzja = {tak (6), nie (1)}

decyzja = {tak (3), nie (4)}

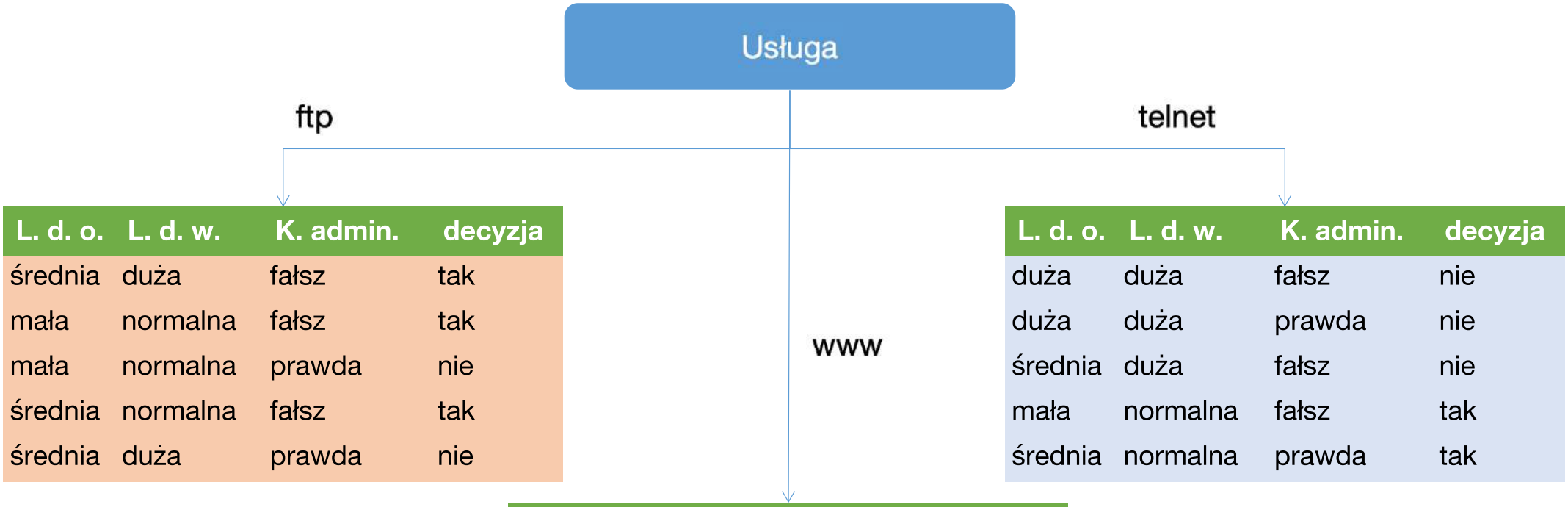
Drzewa decyzyjne

Usługa	L. d. o.	L. d. w.	K. admin.	decyzja
telnet	średnia	normalna	**fałsz**	tak
www	duża	normalna	**fałsz**	tak
telnet	mała	normalna	**fałsz**	tak
ftp	mała	normalna	**fałsz**	tak
telnet	średnia	duża	**fałsz**	tak
ftp	średnia	duża	**fałsz**	nie
www	duża	duża	**fałsz**	tak
ftp	duża	duża	**fałsz**	nie
ftp	średnia	normalna	**prawda**	tak
telnet	mała	normalna	**prawda**	nie
www	mała	normalna	**prawda**	tak
telnet	średnia	duża	**prawda**	nie
www	średnia	duża	**prawda**	tak
ftp	duża	duża	**prawda**	nie

decyzja = {tak (6), nie (2)}

decyzja = {tak (3), nie (3)}

Drzewa decyzyjne

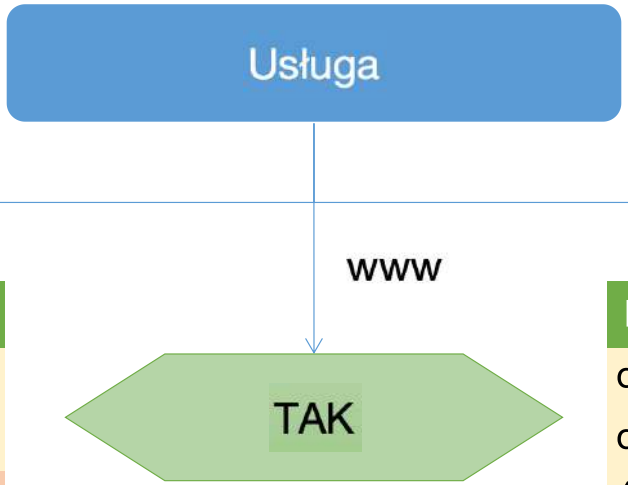


L. d. o.	L. d. w.	K. admin.	decyzja
średnia	duża	fałsz	tak
mała	normalna	fałsz	tak
mała	normalna	prawda	nie
średnia	normalna	fałsz	tak
średnia	duża	prawda	nie

L. d. o.	L. d. w.	K. admin.	decyzja
duża	duża	fałsz	nie
duża	duża	prawda	nie
średnia	duża	fałsz	nie
mała	normalna	fałsz	tak
średnia	normalna	prawda	tak

L. d. o.	L. d. w.	K. admin.	decyzja
duża	duża	fałsz	**tak**
mała	normalna	prawda	**tak**
średnia	duża	prawda	**tak**
duża	normalna	fałsz	**tak**

Drzewa decyzyjne



L. d. o.	L. d. w.	K. admin.	decyzja
średnia	duża	**fałsz**	tak
mała	normalna	**fałsz**	tak
mała	normalna	**prawda**	nie
średnia	normalna	**fałsz**	tak
średnia	duża	**prawda**	nie

L. d. o.	L. d. w.	K. admin.	decyzja
duża	**duża**	fałsz	nie
duża	**duża**	prawda	nie
średnia	**duża**	fałsz	nie
mała	**normalna**	fałsz	tak
średnia	**normalna**	prawda	tak

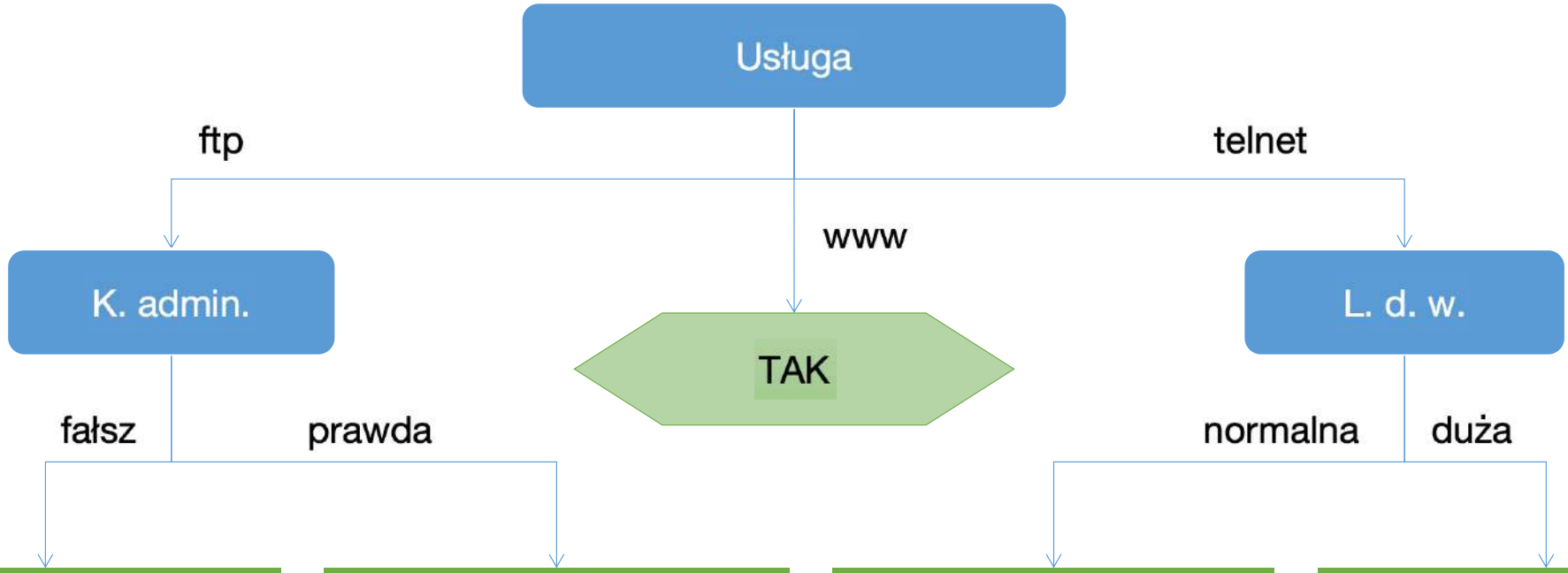
K. admin.:

- fałsz = {tak (3), nie (0)}
- prawda = {tak(0), nie (2)}

L. d. w.:

- duża = {tak (0), nie (3)}
- normalna = {tak(2), nie (0)}

Drzewa decyzyjne



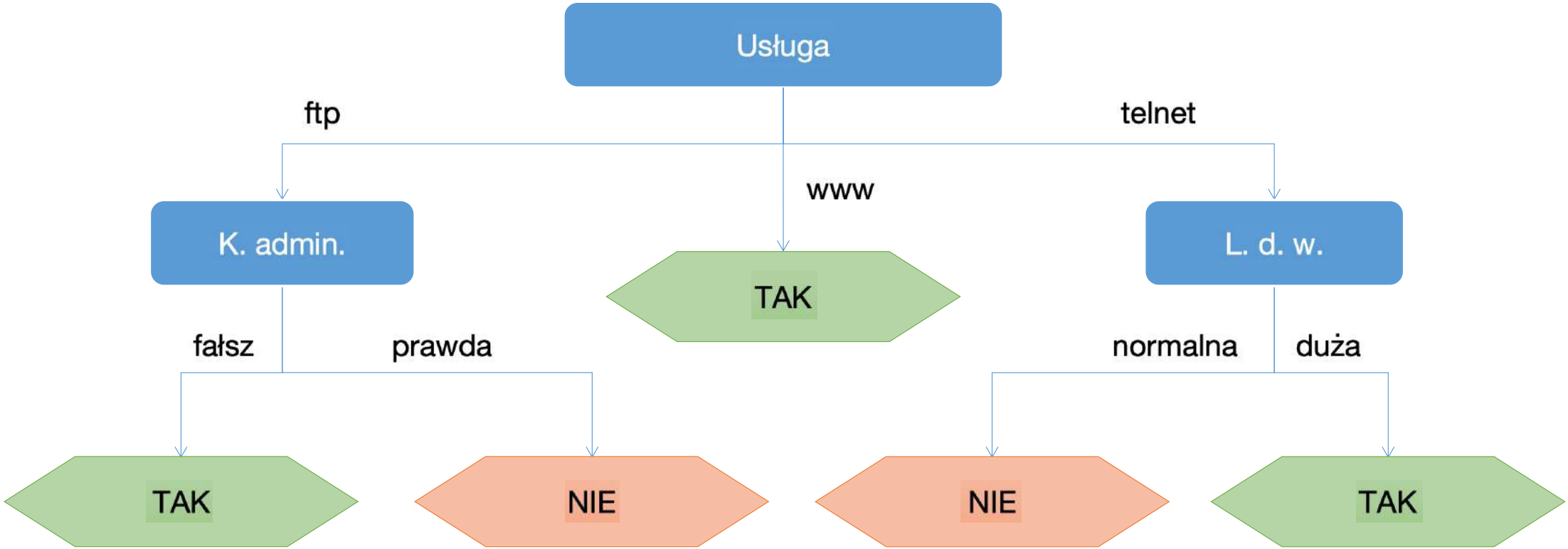
L. d. o.	L. d. w.	decyzja
średnia	duża	**tak**
mała	normalna	**tak**
średnia	normalna	**tak**

L. d. o.	L. d. w.	decyzja
mała	normalna	**nie**
średnia	duża	**nie**

L. d. o.	K. admin.	decyzja
duża	fałsz	**nie**
duża	prawda	**nie**
średnia	fałsz	**nie**

L. d. o.	K. admin.	decyzja
mała	fałsz	**tak**
średnia	prawda	**tak**

Drzewa decyzyjne

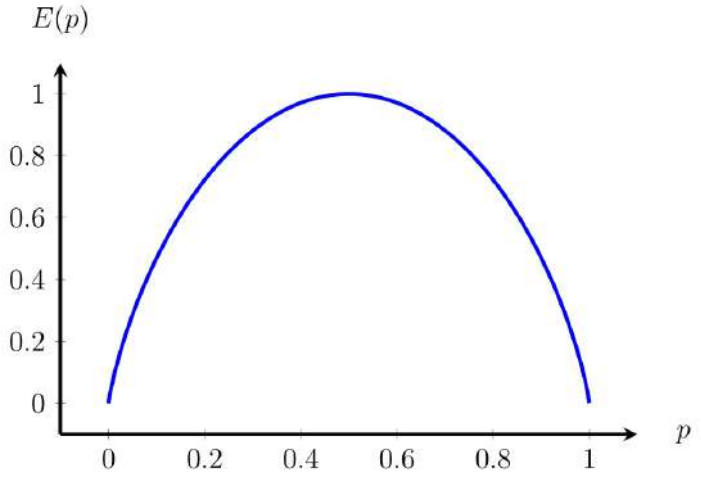


Drzewa decyzyjne - entropia

Rozkład zmiennej losowej: $P = \{p_1, p_2, \ldots, p_n\}$

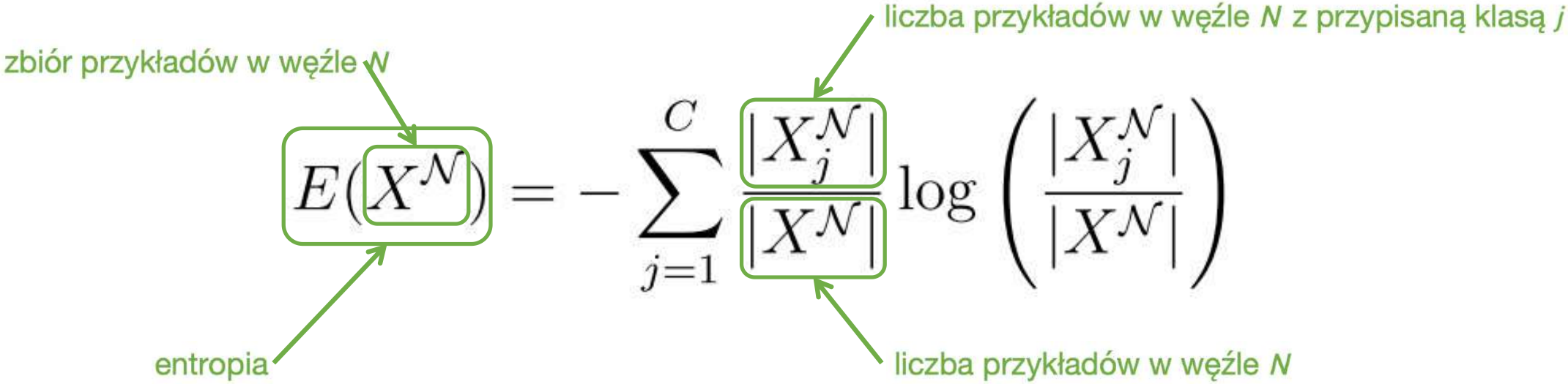
Entropia - miara informacji opisanej danym rozkładem prawdopodobieństwa

$$E(P) = -\sum_{i=1}^{n} p_i \log(p_i)$$



Entropia podziału zbioru na klasy

Przyjmując, że mamy C wartości atrybutu decyzyjnego, entropia zbioru przykładów wynosi



Drzewa decyzyjne

Usługa	L. d. o.	L. d. w.	K. admin.	decyzja
ftp	duża	duża	fałsz	nie
ftp	duża	duża	prawda	nie
www	duża	duża	fałsz	tak
telnet	średnia	duża	fałsz	tak
telnet	mała	normalna	fałsz	tak
telnet	mała	normalna	prawda	nie
www	mała	normalna	prawda	tak
ftp	średnia	duża	fałsz	nie
ftp	mała	normalna	fałsz	tak
telnet	średnia	normalna	fałsz	tak
ftp	średnia	normalna	prawda	tak
www	średnia	duża	prawda	tak
www	duża	normalna	fałsz	tak
telnet	średnia	duża	prawda	nie

$$
\begin{aligned}
E(X) &= -\frac{|X_{\text{nie}}|}{|X|}\log\left(\frac{|X_{\text{nie}}|}{|X|}\right) - \frac{|X_{\text{tak}}|}{|X|}\log\left(\frac{|X_{\text{tak}}|}{|X|}\right) \\
&= -\frac{5}{14}\log\left(\frac{5}{14}\right) - \frac{9}{14}\log\left(\frac{9}{14}\right) \\
&= 0.94
\end{aligned}
$$

Drzewa decyzyjne

Usługa	L. d. o.	L. d. w.	K. admin.	decyzja
telnet	średnia	duża	fałsz	tak
telnet	mała	normalna	fałsz	tak
telnet	mała	normalna	prawda	nie
telnet	średnia	normalna	fałsz	tak
telnet	średnia	duża	prawda	nie
www	duża	duża	fałsz	tak
www	mała	normalna	prawda	tak
www	średnia	duża	prawda	tak
www	duża	normalna	fałsz	tak
ftp	duża	duża	fałsz	nie
ftp	duża	duża	prawda	nie
ftp	średnia	duża	fałsz	nie
ftp	mała	normalna	fałsz	tak
ftp	średnia	normalna	prawda	tak

decyzja = {tak (3), nie (2)}

$$E(X^{A\ \text{usługa}}) = \frac{5}{14}\left(-\frac{3}{5}\log\frac{3}{5} - \frac{2}{5}\log\frac{2}{5}\right) + \frac{4}{14}\left(-\frac{4}{4}\log\frac{4}{4} - \frac{0}{4}\log\frac{0}{4}\right) + \frac{5}{14}\left(-\frac{3}{5}\log\frac{3}{5} - \frac{2}{5}\log\frac{2}{5}\right) = 0,69$$

Entropia podziału zbioru na klasy dla atrybutu A^k

liczba przykładów w węźle $\mathcal{N}$ z m-tą wartością k-tego atrybutu

$$E(X^{\mathcal{N}|A^k}) = \sum_{m=1}^{|A^k|} \frac{|X^{\mathcal{N}|A^k_m}|}{|X^{\mathcal{N}}|} E(X^{\mathcal{N}|A^k})$$

Zysk informacyjny jest to różnica entropii zbioru X i entropii podziału zbioru X zgodnie z wartościami atrybutu A^k

$$G^{A^k} = E(X^{\mathcal{N}}) - E(X^{\mathcal{N}|A^k})$$

$$A^k = \max_k G^{A^k}$$

Drzewa decyzyjne

$$E(X^{A_{\text{usługa}}}) = \frac{5}{14}\left(-\frac{3}{5}\log\frac{3}{5} - \frac{2}{5}\log\frac{2}{5}\right) + \frac{4}{14}\left(-\frac{4}{4}\log\frac{4}{4} - \frac{0}{4}\log\frac{0}{4}\right) + \frac{5}{14}\left(-\frac{3}{5}\log\frac{3}{5} - \frac{2}{5}\log\frac{2}{5}\right) = 0,69$$

$$E(X^{A^{\text{LDO}}}) = \frac{11}{14}\left(-\frac{4}{11}\log\frac{4}{11} - \frac{7}{11}\log\frac{7}{11}\right) + \frac{3}{14}\left(-\frac{1}{3}\log\frac{1}{3} - \frac{2}{3}\log\frac{2}{3}\right) = 0,92$$

$$E(X^{A^{\text{LDW}}}) = \frac{5}{14}\left(-\frac{1}{5}\log\frac{1}{5} - \frac{4}{5}\log\frac{4}{5}\right) + \frac{9}{14}\left(-\frac{4}{9}\log\frac{4}{9} - \frac{5}{9}\log\frac{5}{9}\right) = 0,89$$

$$E(X^{A^{\text{K. admin.}}}) = \frac{8}{14}\left(-\frac{2}{8}\log\frac{2}{8} - \frac{6}{8}\log\frac{6}{8}\right) + \frac{6}{14}\left(-\frac{3}{6}\log\frac{3}{6} - \frac{3}{6}\log\frac{3}{6}\right) = 0,89$$

$$G^{A^{\text{usługa}}} = 0,94 - 0,69 = 0,25$$

$$G^{A^{\text{LDO}}} = 0,94 - 0.92 = 0,02$$

$$G^{A^{\text{LDW}}} = 0,94 - 0,89 = 0,05$$

$$G^{A^{\text{K. admin.}}} = 0,94 - 0,89 = 0,05$$

Algorytm ID3

1 **if** *warunek zakończenia jest spełniony* **then**
2 ⌊ powróć
3 Oblicz entropię w aktualnym węźle $E(X^{\mathcal{N}})$ Oblicz entropię dla każdego dostępnego atrybutu $E(X^{\mathcal{N}|\mathcal{A}^{\|}})$;
4 Oblicz zysk informacyjny dla każdego dostępnego atrybutu G^{A^k};
5 Wybierz atrybut A^k dla którego zysk jest największy;
6 Podziel zbiór przykładów uczących ze względu na wartość atrybutu $A^{\max}$ na rozłączne podzbiory;
7 Dodaj do drzewa krawędzie opisane wartościami atrybutu $A^{\max}$;
8 Usuń atrybut $A^{\max}$ Dla każdego poddrzewa powtórz kroki od 1 do 8

Inne miary wyboru podziału - Gini Index

$$Gini(X^{\mathcal{N}}) = 1 - \sum_{j=1}^{C} \left(\frac{|X_j^{\mathcal{N}}|}{|X^{\mathcal{N}}|} \right)^2$$

$$GiniGain(X^{\mathcal{N}|A^k}) = \sum_{m=1}^{|A^k|} \frac{|X^{\mathcal{N}|A_m^k}|}{|X^{\mathcal{N}}|} Gini(X^{\mathcal{N}|A^k})$$

Zalety i wady algorytmu ID3

- prostota
- jeżeli dane nie są zaszumione to algorytm będzie dawał poprawne wyniki

- działa jedynie dla danych dyskretnych
- nie działa jeżeli są braki w danych
- może dawać bardzo duże drzewa
- może dawać drzewa „przeuczone”

Drzewa decyzyjne

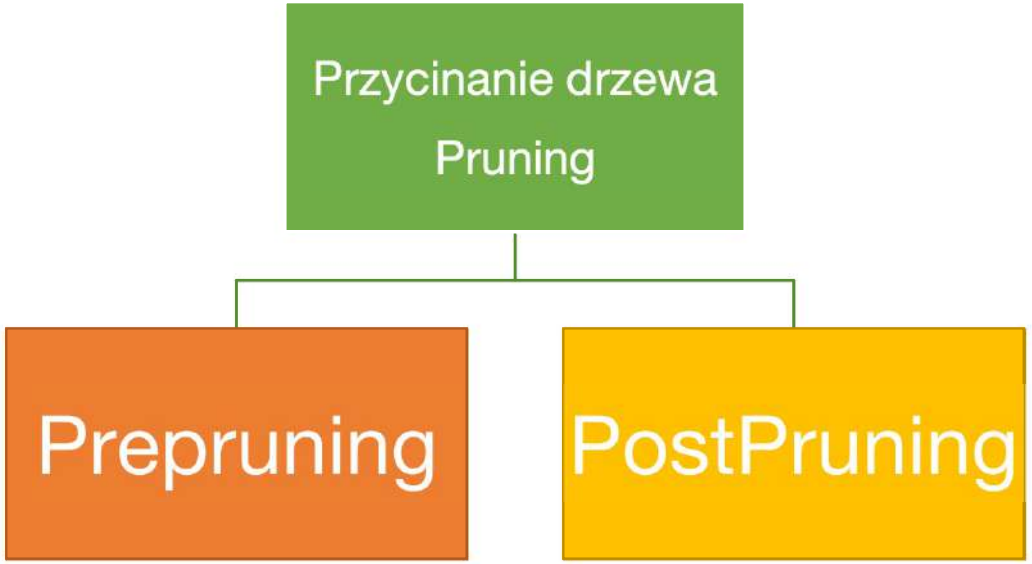
Usługa	L. d. o.	L. d. w.	K. admin.	decyzja
ftp	85	85	fałsz	nie
ftp	80	90	prawda	nie
www	83	86	fałsz	tak
telnet	70	96	fałsz	tak
telnet	68	80	fałsz	tak
telnet	65	70	prawda	nie
www	64	65	prawda	tak
ftp	72	95	fałsz	nie
ftp	69	70	fałsz	tak
telnet	75	80	fałsz	tak
ftp	75	70	prawda	tak
www	72	90	prawda	tak
www	81	75	fałsz	tak
telnet	71	91	prawda	nie

Algorytm C4.5

- Zysk informacyjny jest obliczany na podstawie tych przykładów, dla których wartość jest zdefiniowana
- Atrybuty mogą mieć ciągłe wartości:
  - Wartości dla danego atrybutu ustawiamy w kolejności rosnącej
  - Dla każdej z tych wartości dzielimy przykłady na dwa podzbiory $\leq a^k_m$ i $a^k_m <$
  - Obliczamy zysk, dla każdego takiego podziału
  - Wybieramy taki podział, dla którego zysk jest największy

Przycinanie drzewa

Przycinanie drzewa polega na zamianie, wybranych węzłów wewnętrznych na liście



Przycinanie drzewa

Przycinanie drzewa polega na zamianie, wybranych węzłów wewnętrznych na liście

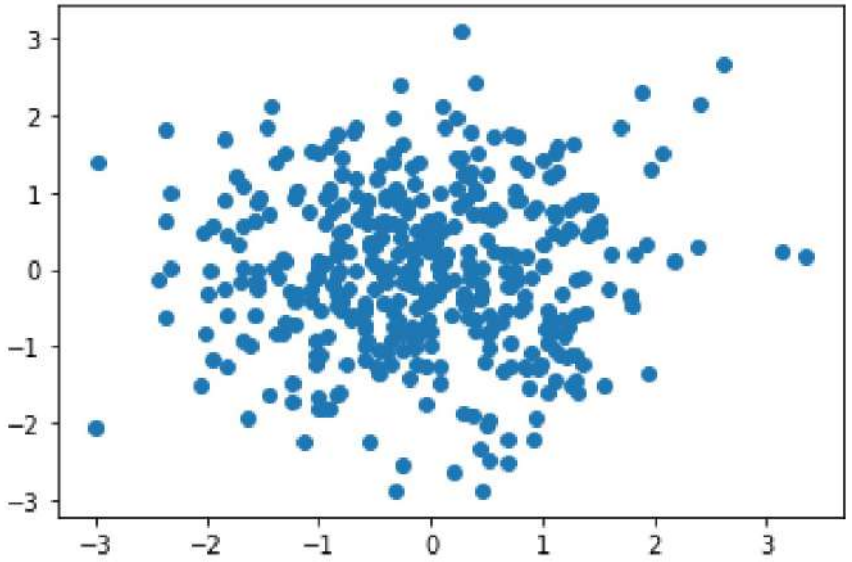
- Obcinanie względem błędu
- Obcinanie względem kosztu

$$\frac{\varepsilon(T_O) - \varepsilon(T)}{|liscie(T)| - |liscie(T_O)|}$$

Isolation Forrest

Wykorzystanie lasu drzew regresji do wykrywania anomalii

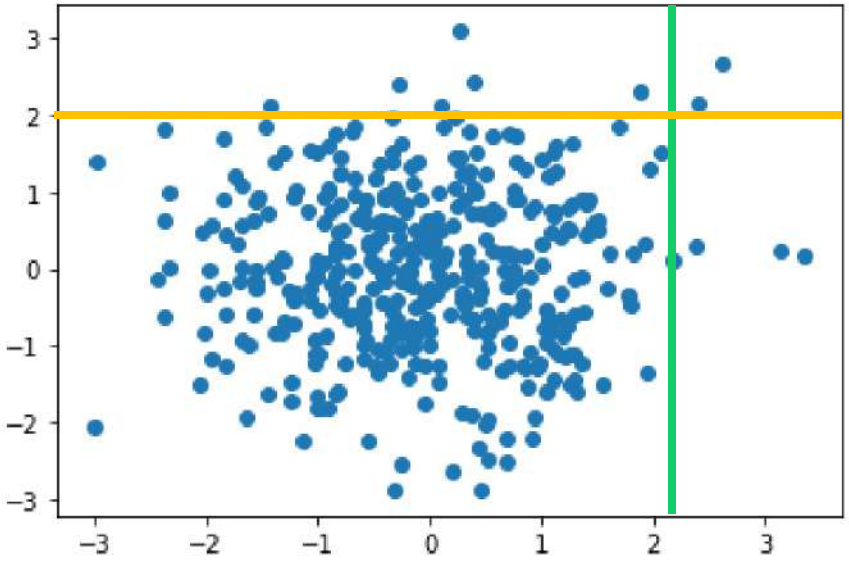
Przykłady anomalne występują rzadko i są odmienne od przykładów normalnych, dlatego są podatne na izolację



Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation forest." 2008 Eighth IEEE International Conference on Data Mining. IEEE, 2008.

Isolation Tree

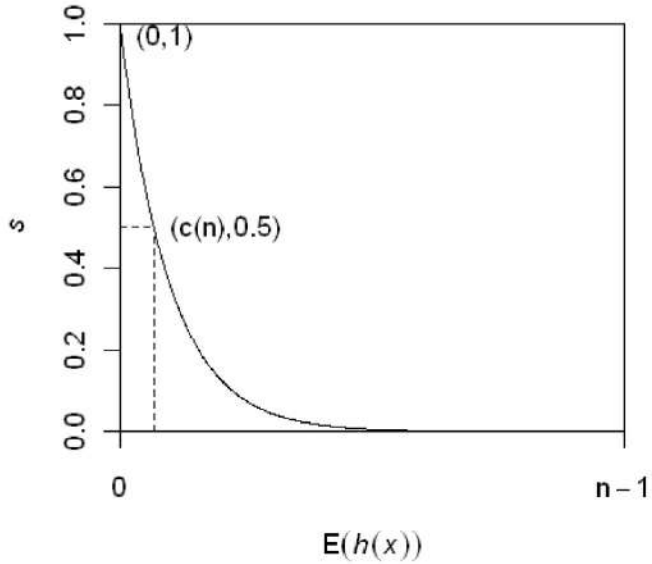
1. Losowo wybierz atrybut do podziału danych
2. Losowo wybierz punkt podziału na dwie części
3. Powtarzaj kroki 1-2 aż do:
    1. osiągnięcia maksymalnej głębokości drzewa
    2. osiągnięcia w liściu tylko jednego przykładu
    3. wszystkie przykłady w liściu mają te same wartości



Isolation Forrest - wykrywanie anomalii

Wskaźnik anomalii (Anomaly score)

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}}$$



h(x) - długość ścieżki w drzewie do osiągnięcia przykładu x
c(*n*) - średnia długość h(x) dla zbioru danych o n przykładach

$$c(n) = 2*(\ln(n-1)+0.5772156649)+(2(n-1)/n)$$

E(h(x)) - średnia wartość h(x) dla kolekcji drzew

Isolation forrest - wykrywanie anomalii

Wskaźnik anomalii (Anomaly score)

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}}$$

Jeżeli $s(x,n) \approx 1$ x jest **zdecydowanie anomalią**

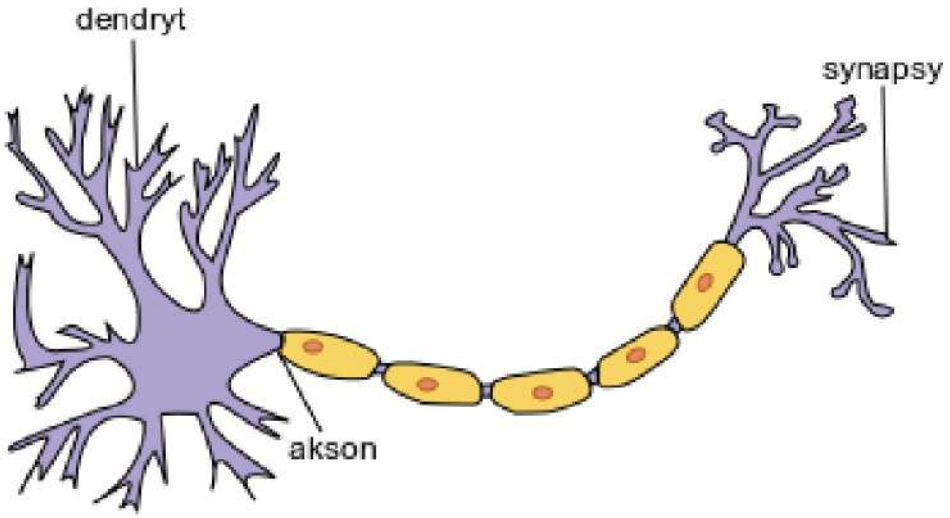
Jeżeli $s(x,n) << 0.5$ to x jest **przykładem „normalnym”**

Jeżeli $s(x,n) \approx 0.5$ dla wszystkich przykładów, to w zbiorze nie ma wyraźnych anomalii

Sieci neuronowe

Neurony

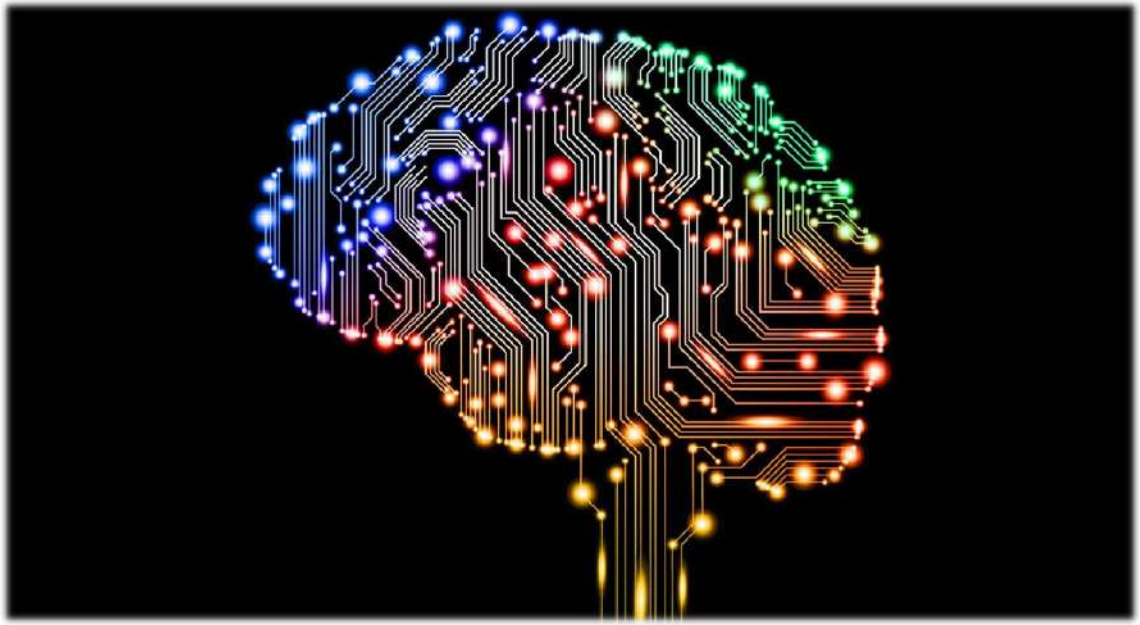
Neuron - podstawowy element układu nerwowego



Ludzki mózg zawiera ok. 100 miliardów neuronów

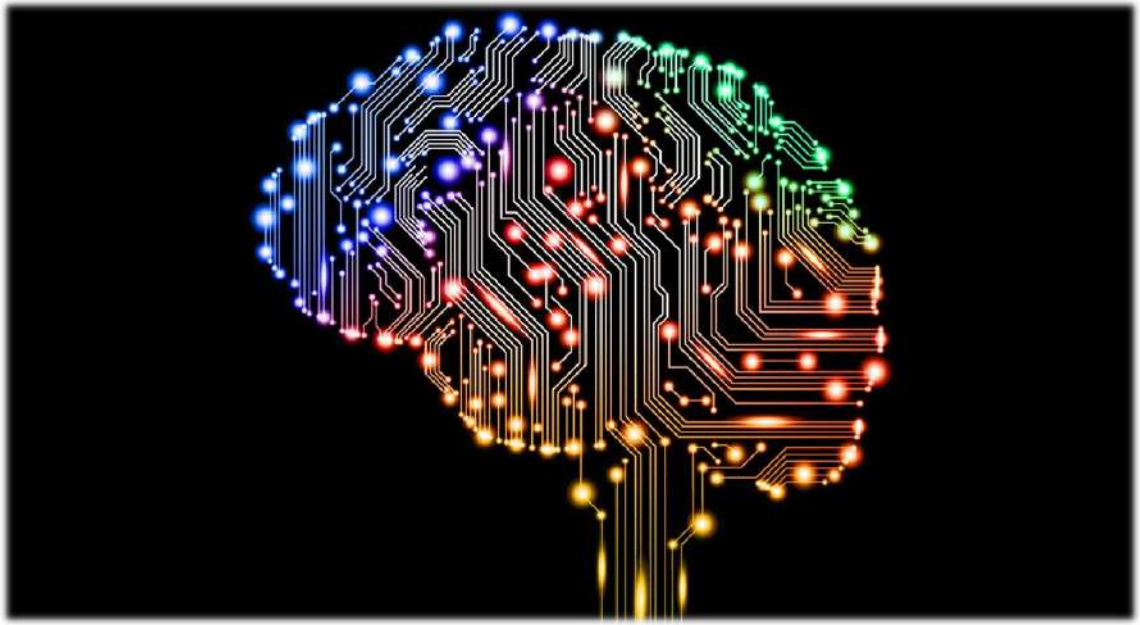
Sztuczne sieci neuronowe

Sztuczne sieci neuronowe są próbą zamodelowania naszego mózgu i sposobu jego pracy

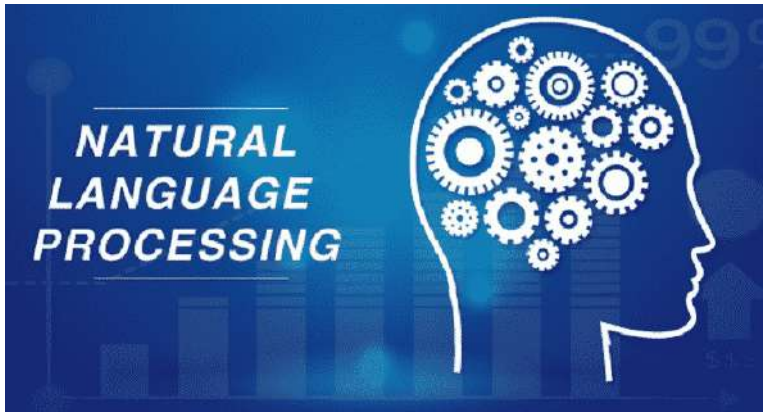


Sztuczne sieci neuronowe

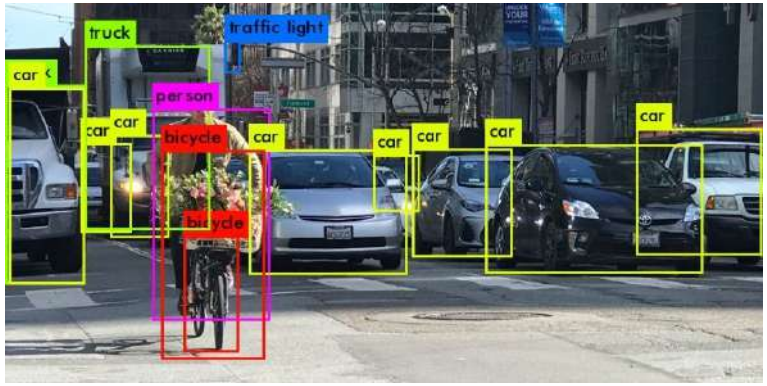
Sztuczne sieci neuronowe to uniwersalne aproksymatory



Przykładowe zastosowania SSN



- Klasyfikacja tekstu
- Klasyfikacja nazwanych jednostek
- Etykietowanie POS
- Detakcja parafraz
- Tłumaczenie maszynowe
- Rozpoznawanie mowy
- ...

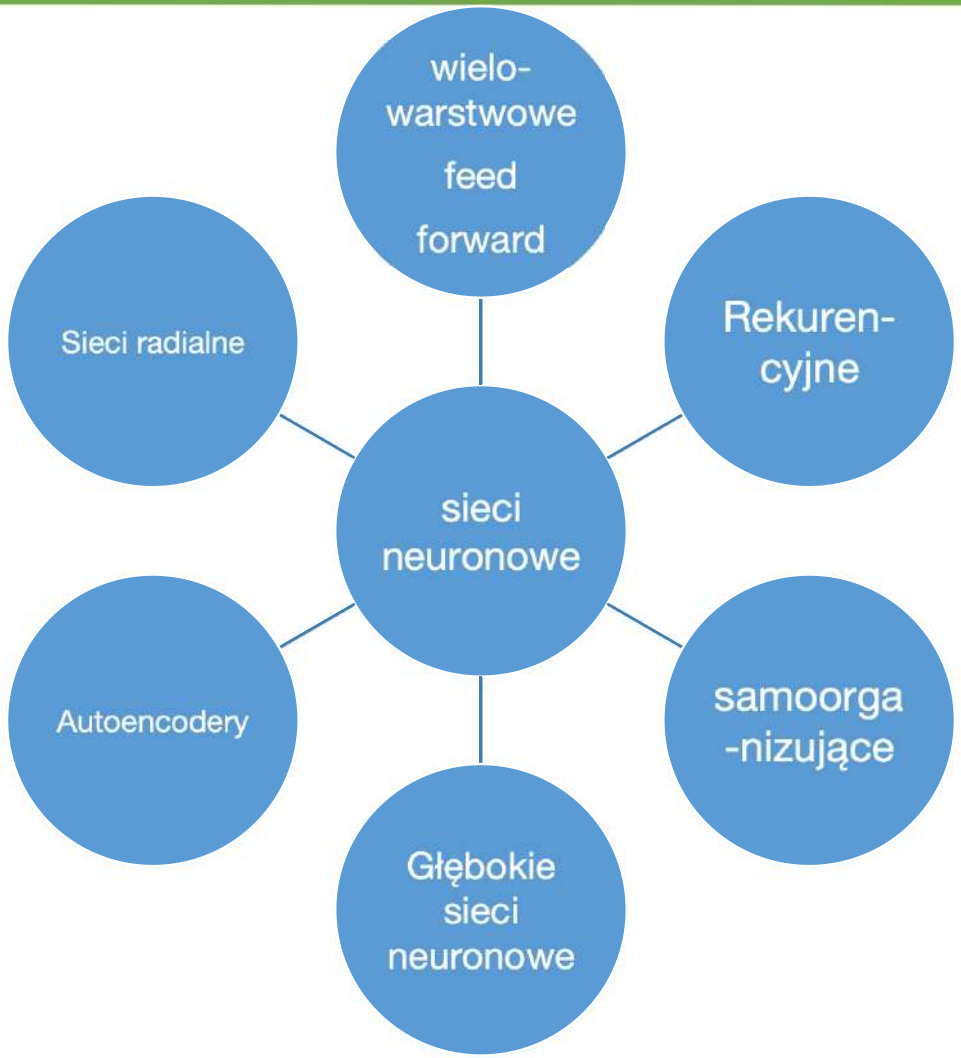


- Rozpoznawanie obrazów
- OCR
- Automatyczne pojazdy
- Klasyfikacja i tagowanie obrazów
- Rozszerzona rzeczywistość
- Automatyczna identyfikacja wad
- Automatyczne dodawanie dźwięku do niemych filmów
- ...


ID: 110222198001011010
危险
90.91%
ID: 110222198001011010
57.92%
ID: 110222198001011010
53.49%


POLICE

Sieci neuronowe



Proste modele neuronu

Sieci neuronowe

Model neuronu można wyrazić za pomocą zależności

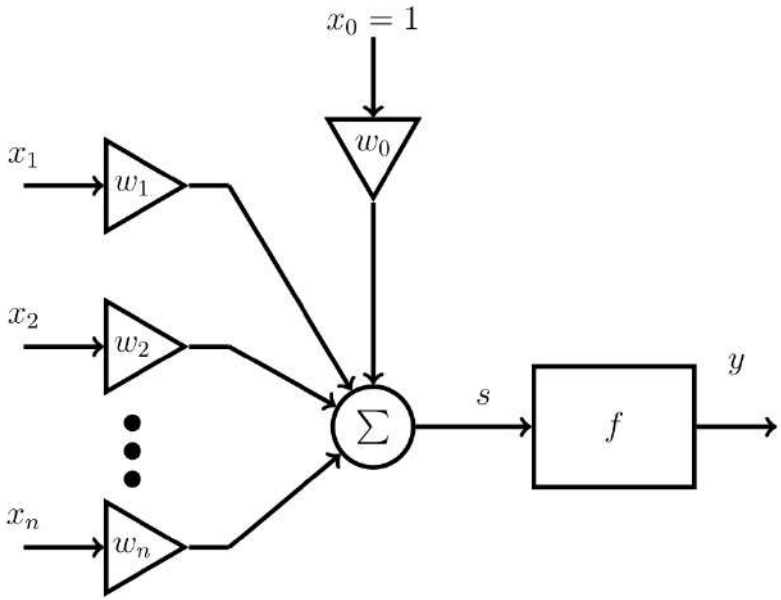
funkcja aktywacji neuronu

$$y = f(s)$$

$$s = \sum_{i=0}^{n} x_i w_i$$

wejście neuronu

waga neuronu

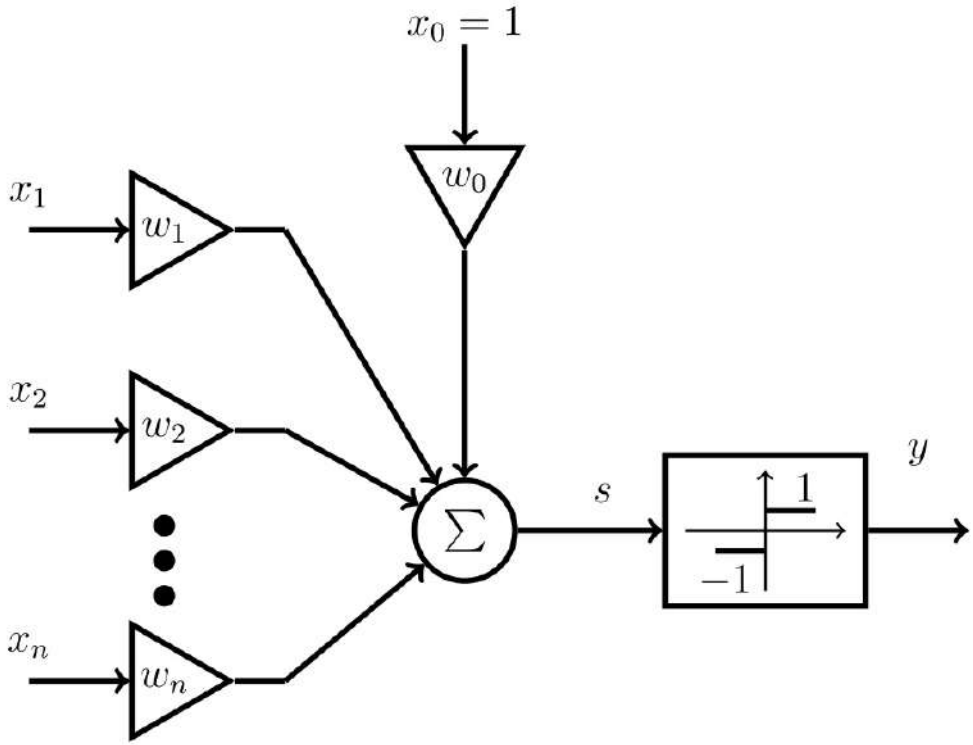


Perceptron

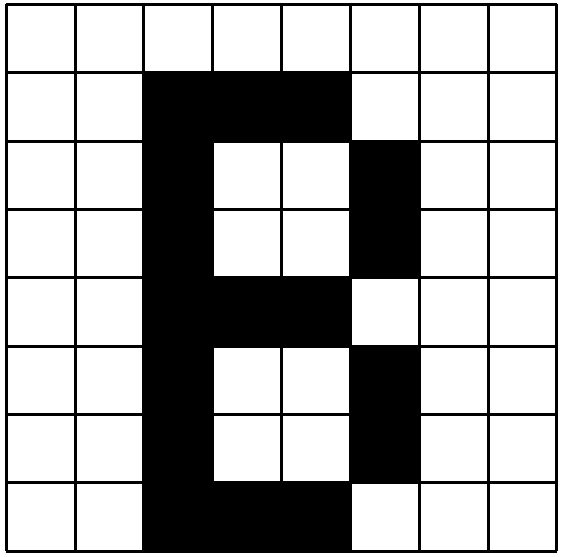
Najprostszym sztucznym neuronem jest perceptron

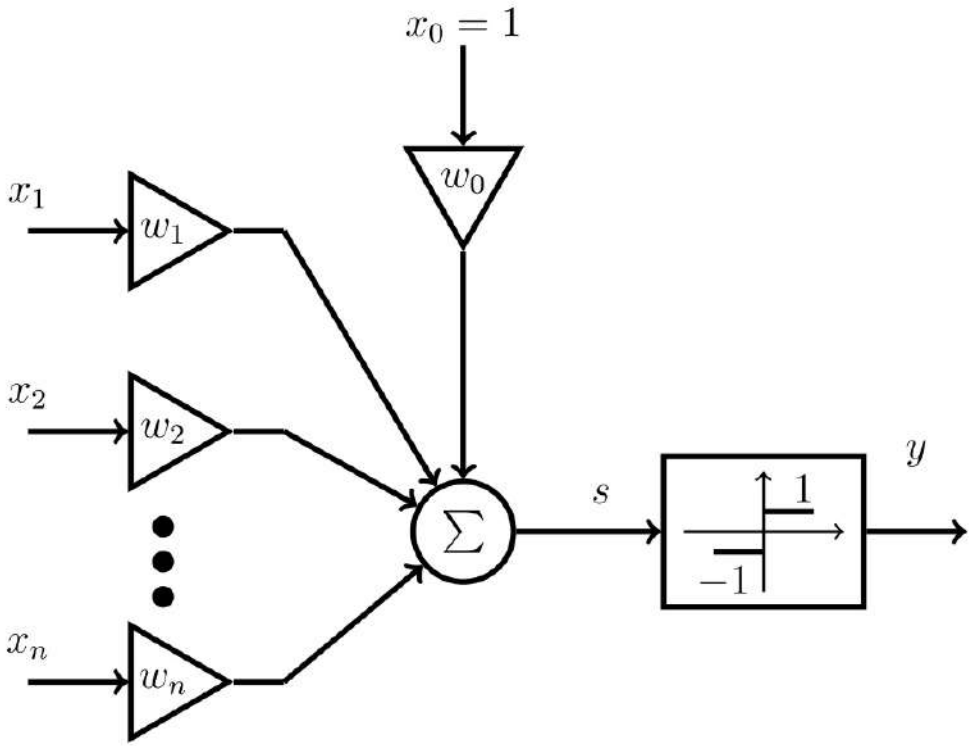
$$y = f\left(\sum_{i=1}^{n} x_i w_i + w_0 \right)$$

$$f(s) = \begin{cases} 1 & \text{gdy } s > 0 \\ -1 & \text{gdy } s \leq 0 \end{cases}$$



Perceptron





$n = 8 \times 8 = 64$

Algorytm uczenia perceptronu (1)

Podczas uczenia perceptronu modyfikowane są jego wagi

1. Wybieramy losowo wagi początkowe
2. Na wejścia neuronu podajemy wektor uczący $\mathbf{x}(t) = [x_1(t), \ldots, x_n(t)]$
3. Obliczamy wartość wejściową perceptronu $y(\mathbf{x}(t))$
4. Porównujemy wyjście neuronu z etykietą $\hat{y}(t)$ przypisaną do obiektu $\mathbf{x}(t)$
5. Modyfikujemy wagi

współczynnik uczenia

$$w_i(t+1) = \begin{cases} w_i(t) + \eta[\hat{y}(t) - y(\mathbf{x}(t))]x_i(t) & \text{gdy } y(\mathbf{x}(t)) \neq \hat{y}(t) \\ w_i(t) & \text{gdy } y(\mathbf{x}(t)) = \hat{y}(t) \end{cases}$$

6. Wracamy do punktu 2

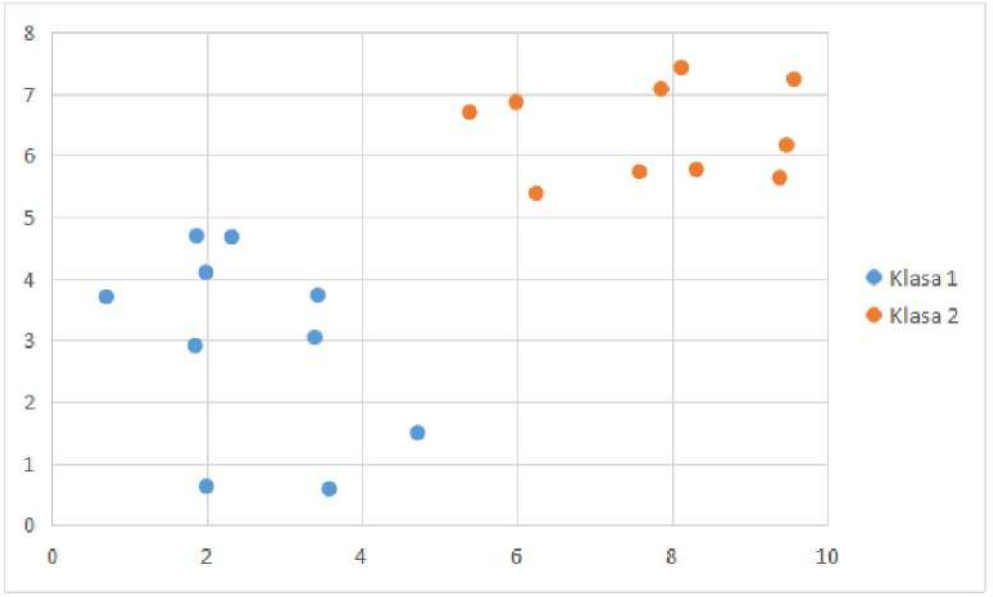
Algorytm uczenia perceptronu (2)

Podczas uczenia perceptronu modyfikowane są jego wagi

Po podaniu na wejście perceptronu wszystkich obiektów zbioru uczącego obliczamy błąd np. średniokwadratowy i sprawdzamy czy jest on mniejszy niż założony z góry próg. Jeżeli nie to powtarzamy uczenie

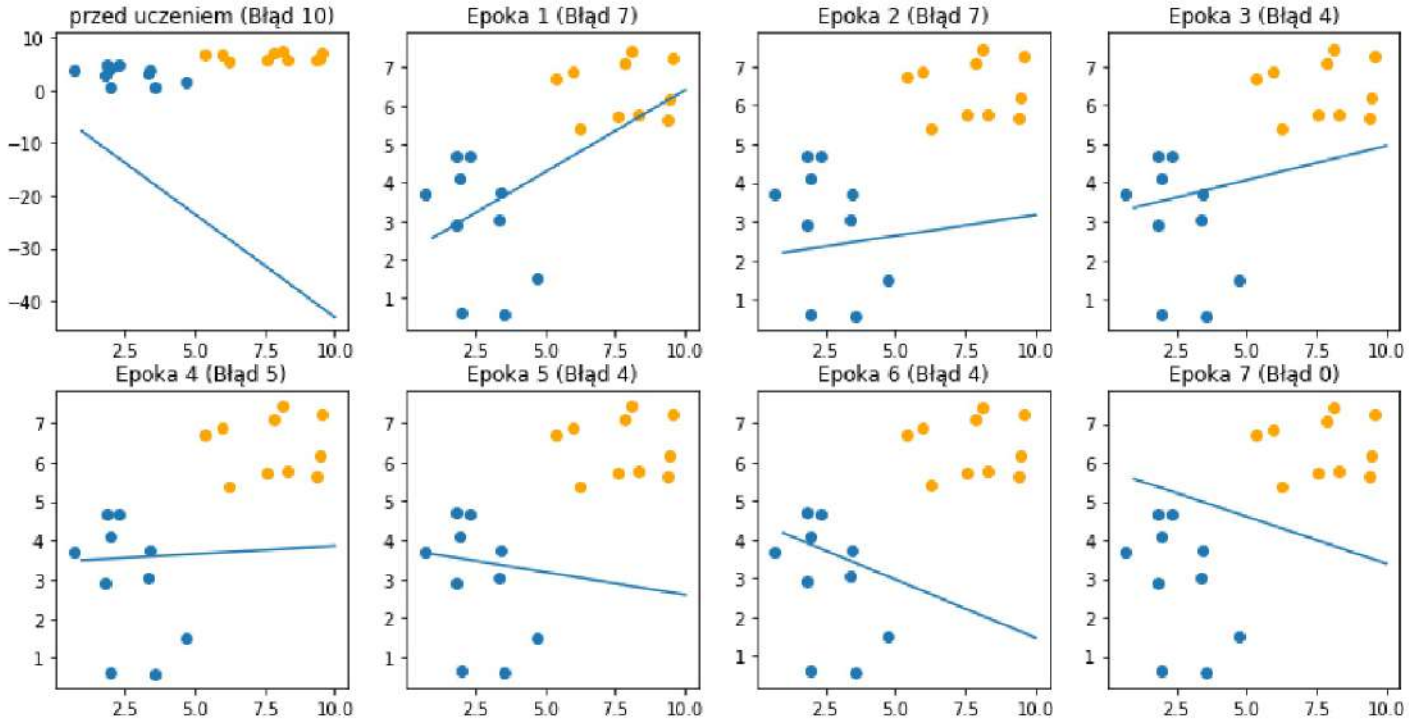
Perceptron uczenie - przykład

Klasa 1		Klasa 2	
X	Y	X	Y
4,71	1,50	8,11	7,43
2,32	4,67	7,57	5,73
1,99	0,63	5,38	6,70
1,84	2,91	6,24	5,39
3,38	3,04	9,46	6,17
0,70	3,70	7,85	7,08
3,57	0,58	9,56	7,24
1,98	4,10	9,38	5,64
3,42	3,73	5,98	6,86
1,86	4,69	8,30	5,77



Perceptron - uczenie

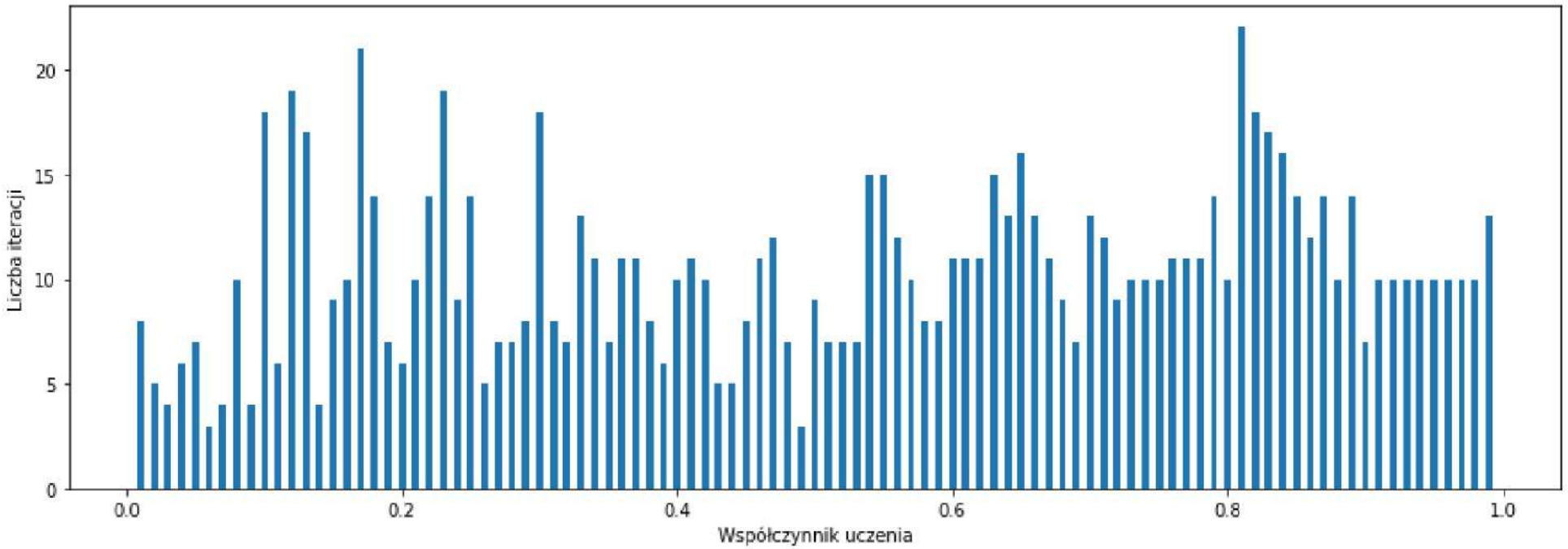
Wagi na początku uczenia: [1.89676844, 1.89640504, 0.48462088]



Wagi na końcu uczenia: [1.87676844, -0.07839496, -0.32217912]

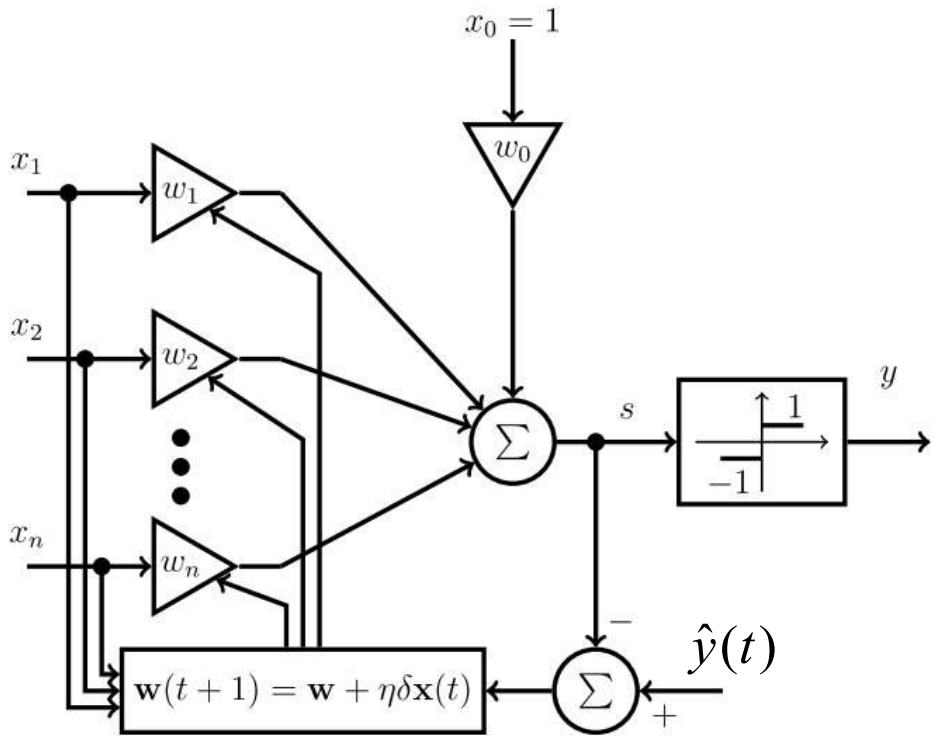
Współczynnik uczenia

Współczynnik uczenia, określa wpływ błędu na modyfikację wag.



Model Adaline (Adaptive Linear Neuron)

Model Adaline jest identyczny jak perceptron.
Różnią się one algorytmem uczenia

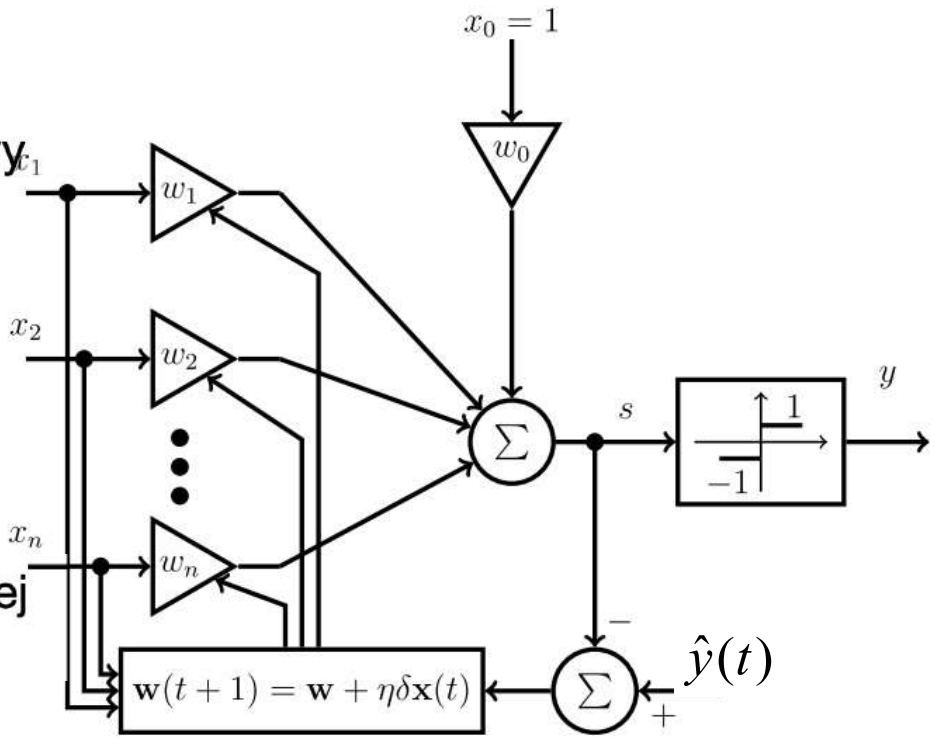


Model Adaline (Adaptive Linear Neuron)

Do uczenia wykorzystuje się błąd średniokwadratowy

$$Q(\mathbf{w}) = \frac{1}{2}\left[\hat{y}(t) - \sum_{i=0}^{n} w_i x_i\right]^2$$

Porównujemy wyjście sieci z wyjściem części liniowej neuronu



Model Adaline (Adaptive Linear Neuron)

Wagi są modyfikowane wg wzoru

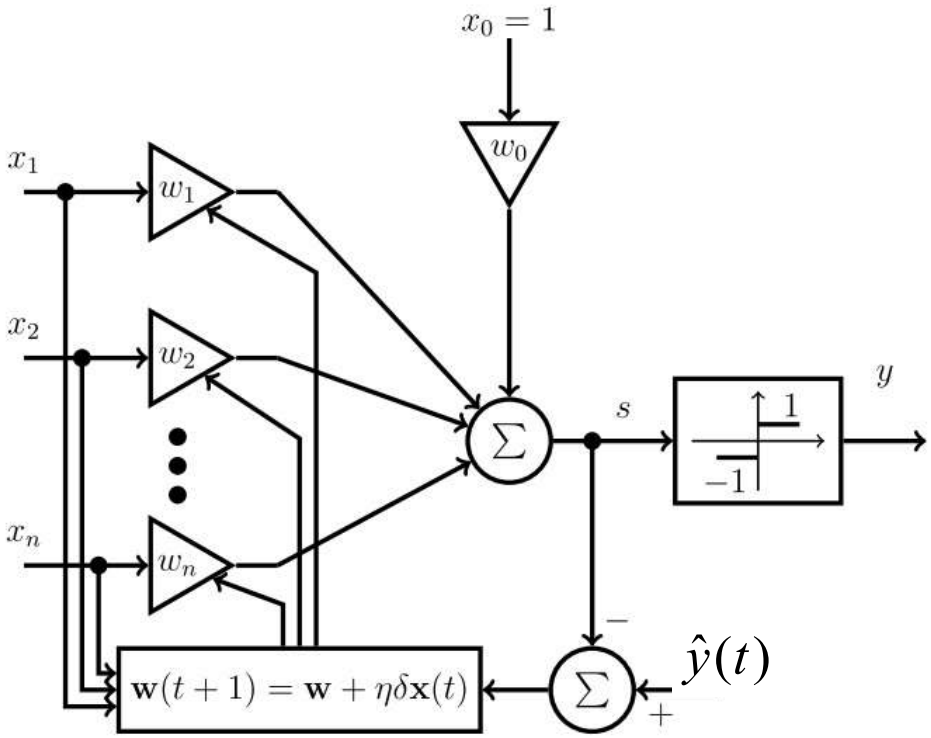
$$w_i(t+1) = w_i(t) - \eta \frac{\partial Q(w_i)}{\partial w_i}$$

współczynnik uczenia

co po przekształceniach doprowadzi do wzoru:

$$\begin{aligned} w_i(t+1) &= w_i(t) - \eta \delta x_i(t) \\ &= w_i(t) - \eta \left(\hat{y}(t) - \sum_{i=0}^{n} w_i x_i \right) x_i(t) \end{aligned}$$

Jest to tzw. reguła delta



Neuron sigmoidalny

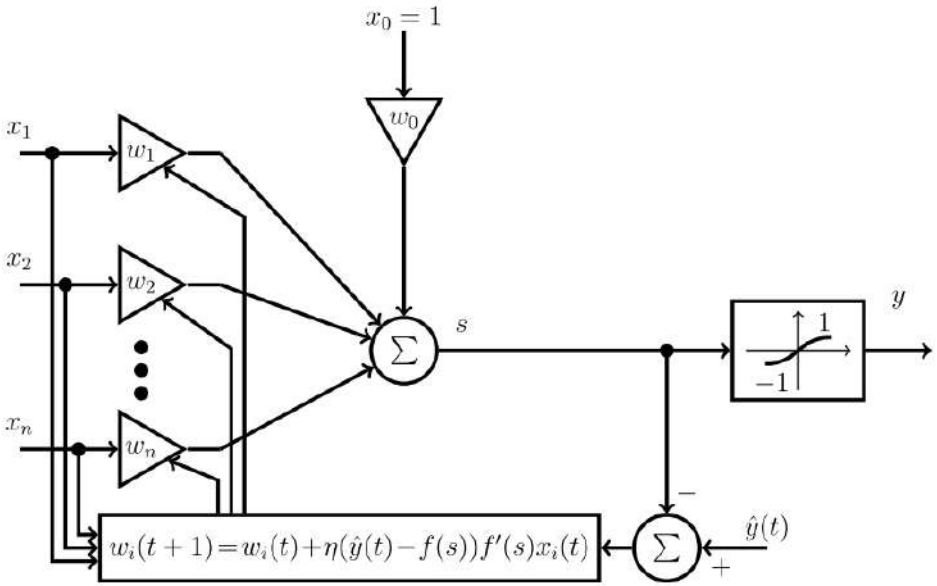
Model sigmoidalny jest bardo podobny do perceptrona.
Różnią się one funkcją aktywacji.

funkcja unipolarna

$$f(x) = \frac{1}{1 + e^{-\beta x}}$$

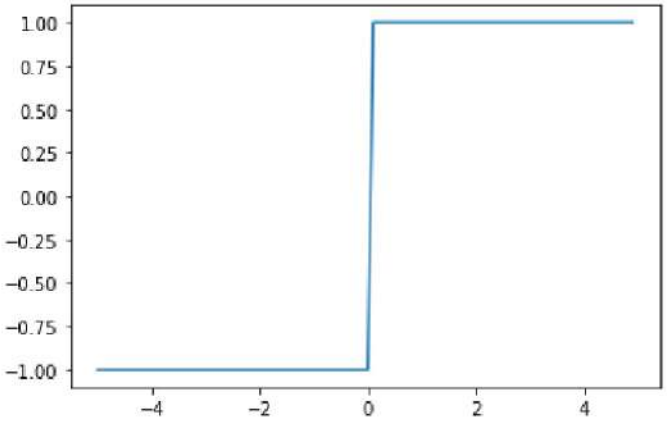
funkcja bipolarna

$$f(x) = \tanh(\beta x) = \frac{1 - e^{\beta x}}{1 + e^{-\beta x}}$$

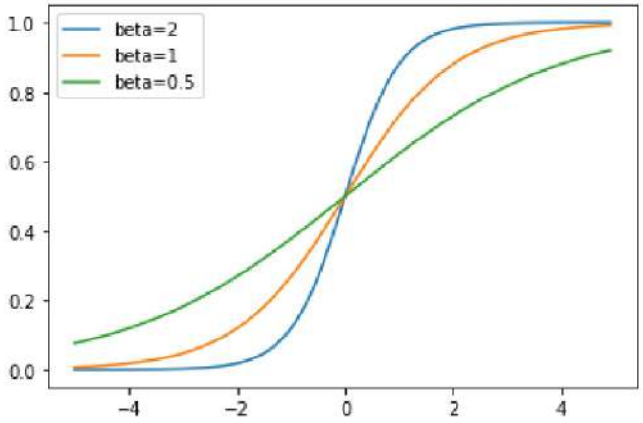


Neuron sigmoidalny

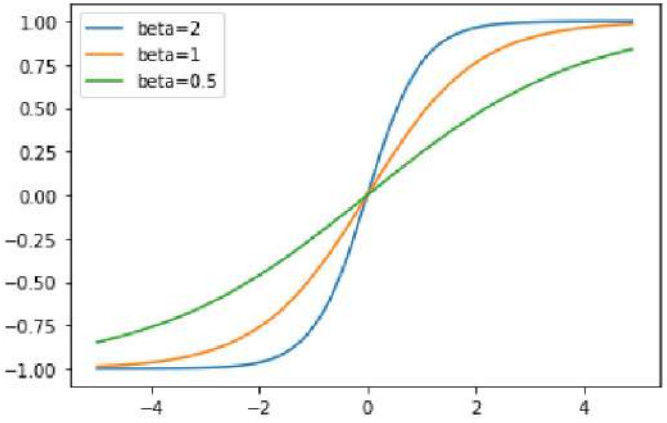
perceptron



unipolarna



bipolarna



Neuron sigmoidalny - uczenie

Ogólna idea uczenia jest taka sama jak przy modelu Adaline,
ale ponieważ funkcja aktywacji jest bardziej rozbudowana, to wzory również są bardziej rozbudowane

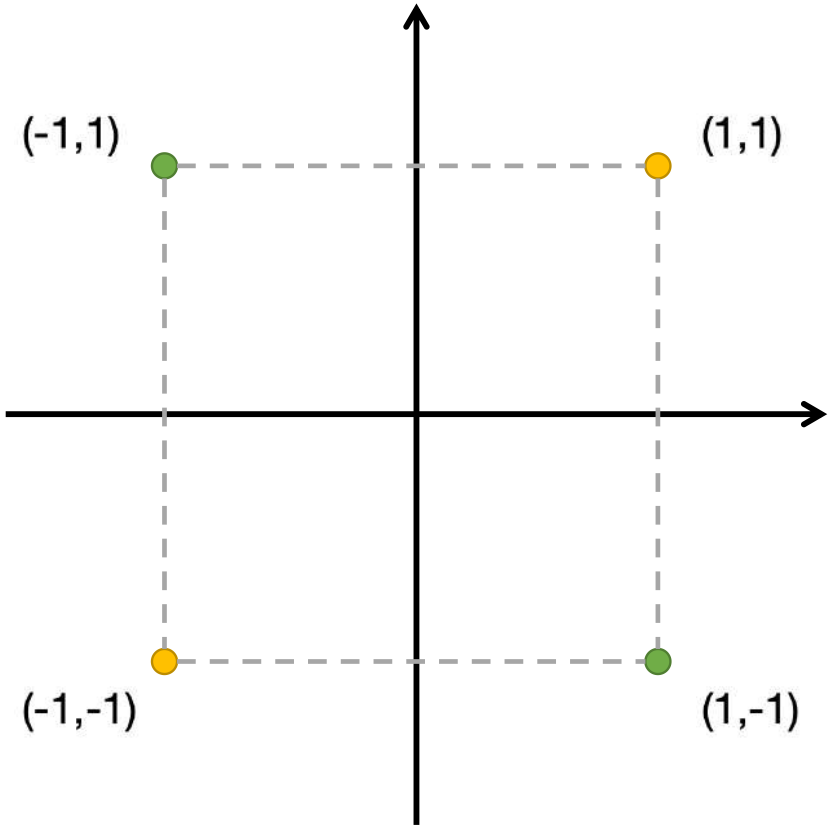
$$w_i(t+1) = w_i(t) - \eta \frac{\partial Q(w_i)}{\partial w_i}$$

$$w_i(t+1) = w_i(t) - \eta \left(\hat{y}(t) - f\left(\sum_{i=0}^{n} w_i x_i \right) \right) f'\left(\sum_{i=0}^{n} w_i x_i \right) x_i(t)$$

$f'(x) = \beta f(x)(1 - f(x))$ dla funkcji unipolarnej

$f'(x) = \beta (1 - f^2(x))$ dla funkcji bipolarnej

Problem XOR



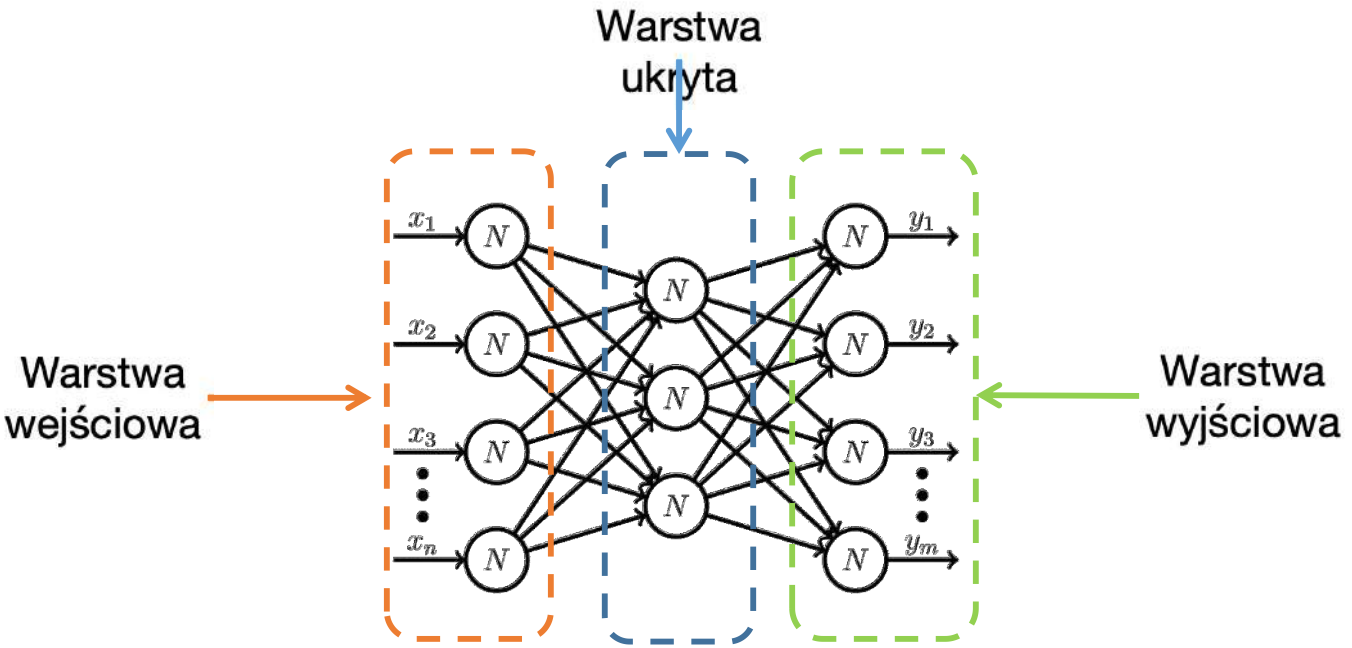
x_1	x_2	XOR(x_1,x_2)
1	1	-1
1	-1	1
-1	1	1
-1	-1	-1

Tego problemu nie da się rozwiązać za pomocą pojedyńczego neuronu

Sieci neuronowe typu feed-forward

Sieci neuronowe wielowarstwowe

Sieć neuronowa wielowarstwowa, jest to zestaw połączonych ze sobą neuronoów ułożonych w dwie lub więcej warstw

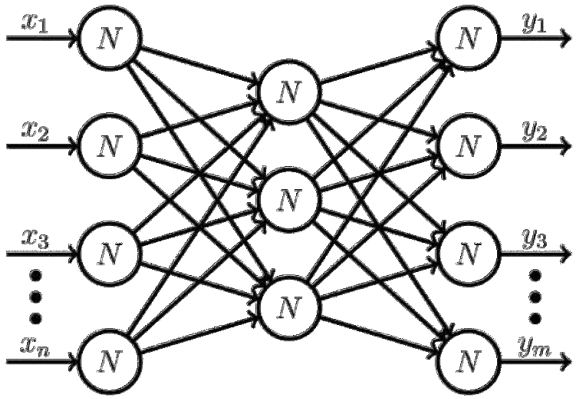


Pojedynczy neuron ma taką samą budowę jak przedstawiono wcześniej

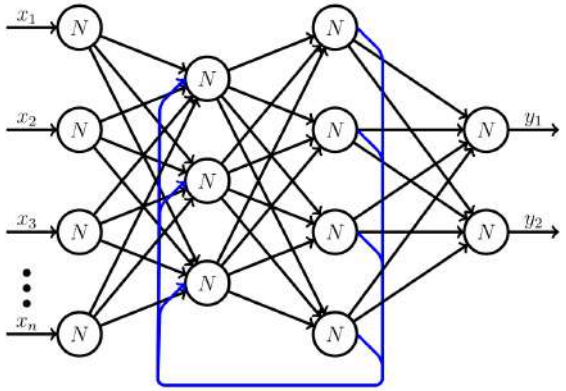
Sieci neuronowe wielowarstwowe

W zależności od rodzaju warstw i sposobu ich połączenia możemy określć różne typy sieci wielowarstwowych

Sieci feed-forward

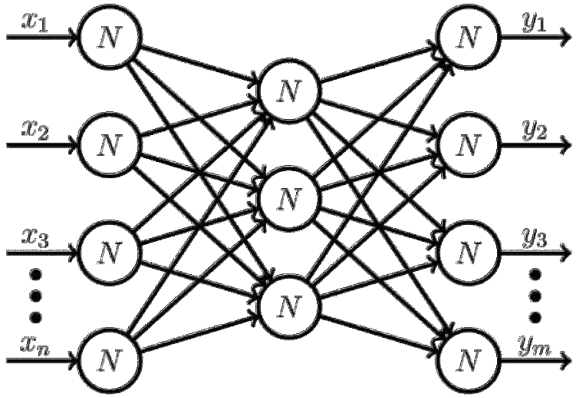


Sieci rekurencyjne



Sieci neuronowe wielowarstwowe

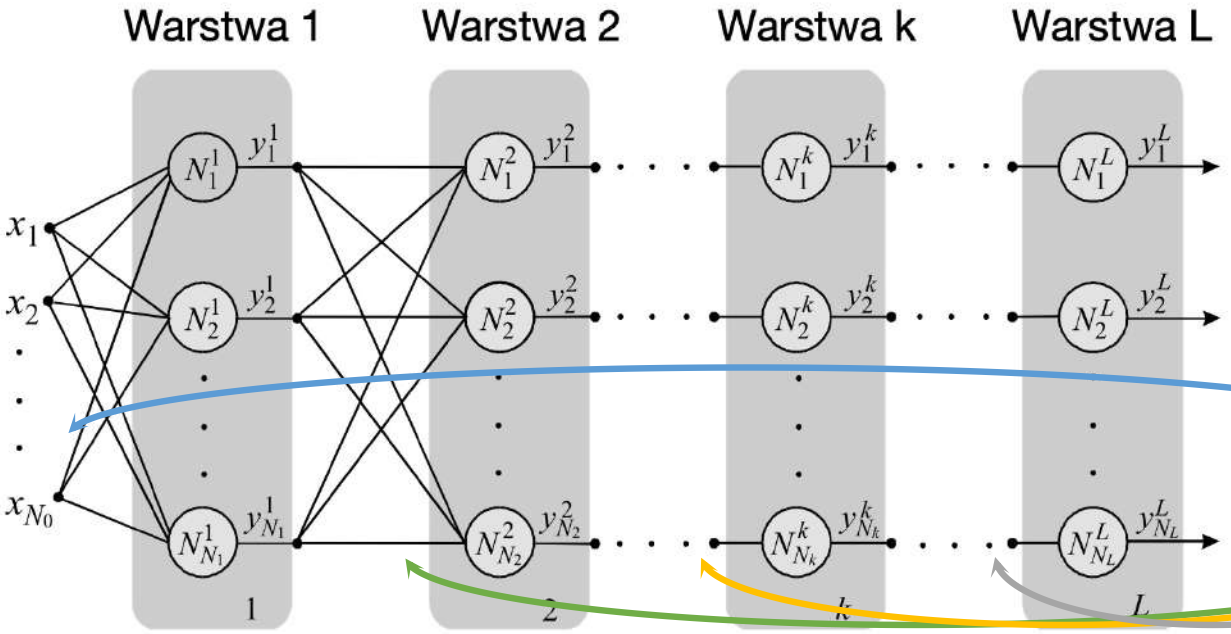
Istnieje wiele metod uczenia sieci neuronowych



- Algorytm wstecznej propagacji błędu
- Algorytm wstecznej propagacji błędu z członem momentum
- Algorytm zmiennej metryki
- Algorytm Levenberga-Marquardta
- Algorytm RLS

Algorytm wstecznej propagacji błędu

Algorytm wstecznej propagacji błędu jest to podstawowa metoda uczenia sieci neuronowych.



Wejście sieci

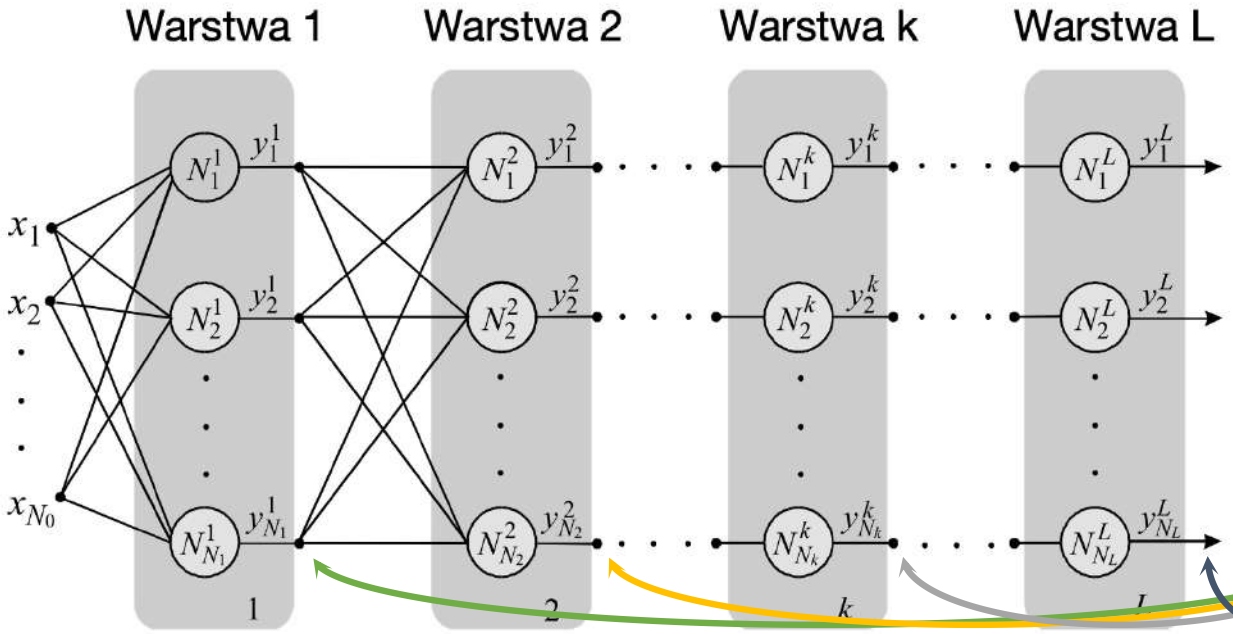
$$\mathbf{x}(t) = [x_1(t), \ldots, x_{N_0}(t)]$$

Wejście warstwy

$$x_i^k(t) = \begin{cases} x_i(t) & \text{dla } k = 1 \\ y_i^{k-1}(t) & \text{dla } k = 2, \ldots, L \\ 1 & \text{dla } i = 0 \quad k = 1, \ldots, L \end{cases}$$

Algorytm wstecznej propagacji błędu

Algorytm wstecznej propagacji błędu jest to podstawowa metoda uczenia sieci neuronowych.



Wyjście sieci

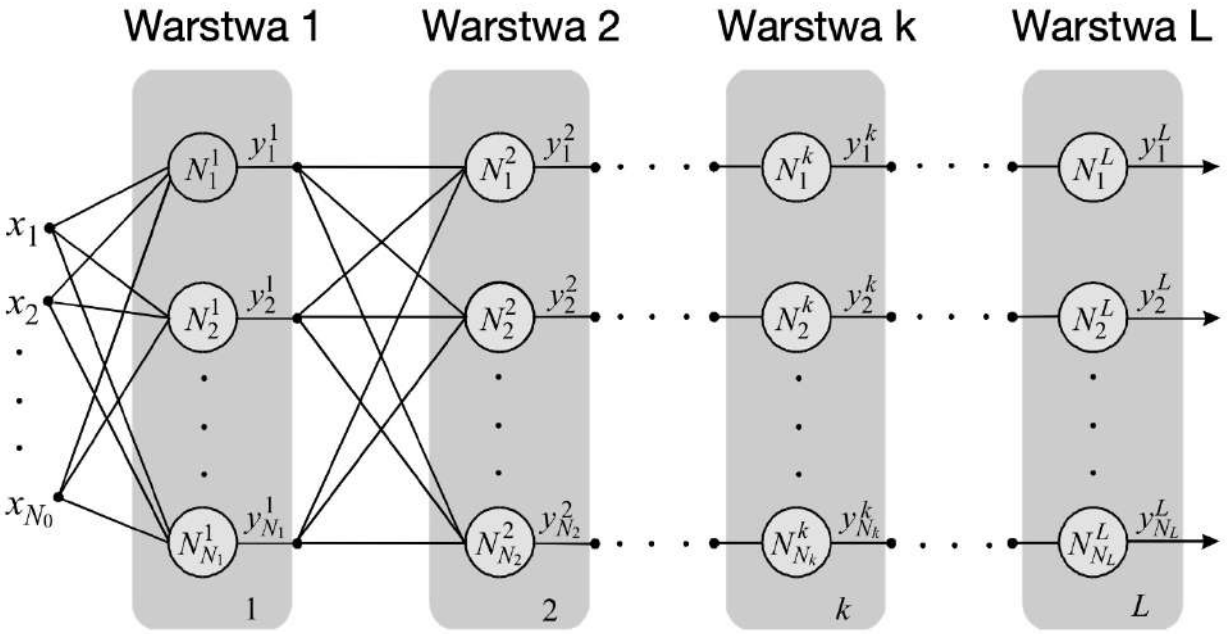
$$\mathbf{y}(t) = [y_1^L(t), \ldots, y_{N_L}^L(t)]$$

Wyjście k-tej warstwy

$$y_i^k(t) = f\left(s_i^k(t)\right) = f\left(\sum_{j=0}^{N_{k-1}} w_{ij}^k(t) x_j^k(t)\right)$$

Algorytm wstecznej propagacji błędu

Algorytm wstecznej propagacji błędu jest to podstawowa metoda uczenia sieci neuronowych.



Wyjścia sieci są porównywane z sygnałami wzorcowymi

$$\mathbf{d}(t) = [d_1^L(t), \ldots, d_{N_L}^L(t)]$$

Błąd na wyjściu sieci

$$Q(t) = \sum_{i=1}^{N_L} \left(\varepsilon_i^L\right)^2 = \sum_{i=1}^{N_L} \left(d_i^L(t) - y_i^L(t)\right)^2$$

Algorytm wstecznej propagacji błędu

Podobnie jak w przypadku uczenia pojedyńczego neuronu, wagi sieci będziemy modyfikować zgodnie z zależnością

$$w_{ij}^{k}(t+1) = w_{ij}^{k}(t) - \eta \frac{\partial Q(t)}{\partial w_{ij}^{k}(t)}$$

współczynnik uczenia

Posługując się prostą sztuczką matematyczną, możemy zapisać, że:

$$\frac{\partial Q(t)}{\partial w_{ij}^{k}(t)} = \frac{\partial Q(t)}{\partial s_{i}^{k}(t)} \frac{\partial s_{i}^{k}(t)}{\partial w_{ij}^{k}(t)} = \frac{\partial Q(t)}{\partial s_{i}^{k}(t)} x_{j}^{k}(t)$$

Algorytm wstecznej propagacji błędu

Jeżeli przyjmiemy, że:

$$\delta_i^k(t) = -\frac{1}{2}\frac{\partial Q(t)}{\partial s_i^k(t)}$$

to wzór:

$$\frac{\partial Q(t)}{\partial w_{ij}^k(t)} = \frac{\partial Q(t)}{\partial s_i^k(t)} x_j^k(t)$$

będziemy mogli zapisać w formie:

$$\frac{\partial Q(t)}{\partial w_{ij}^k(t)} = -2\delta_i^k(t) x_j^k(t)$$

a sposów modyfikacji wag, jako:

$$w_{ij}^k(t+1) = w_{ij}^k(t) - \eta \frac{\partial Q(t)}{\partial w_{ij}^k(t)}$$

$$= w_{ij}^k(t) + 2\eta\delta_i^k(t) x_j^k(t)$$

Algorytm wstecznej propagacji błędu

Sposób wyznaczania wartości $\delta_i^L(t)$ zależy od warstwy sieci

Dla warstwy ostatniej

$$\delta_i^L(t) = -\frac{1}{2}\frac{\partial Q(t)}{\partial s_i^L(t)} = -\frac{1}{2}\frac{\partial\left(d_i^L(t) - y_i^L(t)\right)^2}{\partial s_i^L(t)} = Q_i^L(t) f'\left(s_i^L(t)\right)$$

Dla warstw wcześniejszych

$$\delta_i^L(t) = -\frac{1}{2}\frac{\partial Q(t)}{s_i^k(t)} = \sum_{m=1}^{N_{k+1}} \delta_m^{k+1}(t) w_{mi}^{k+1}(t) f'\left(s_i^k(t)\right) = f'\left(s_i^k(t)\right)\sum_{m=1}^{N_{k+1}} \delta_m^{k+1}(t) w_{mi}^{k+1}(t) = f'\left(s_i^k(t)\right) Q_i^k(t)$$

Algorytm wstecznej propagacji błędu

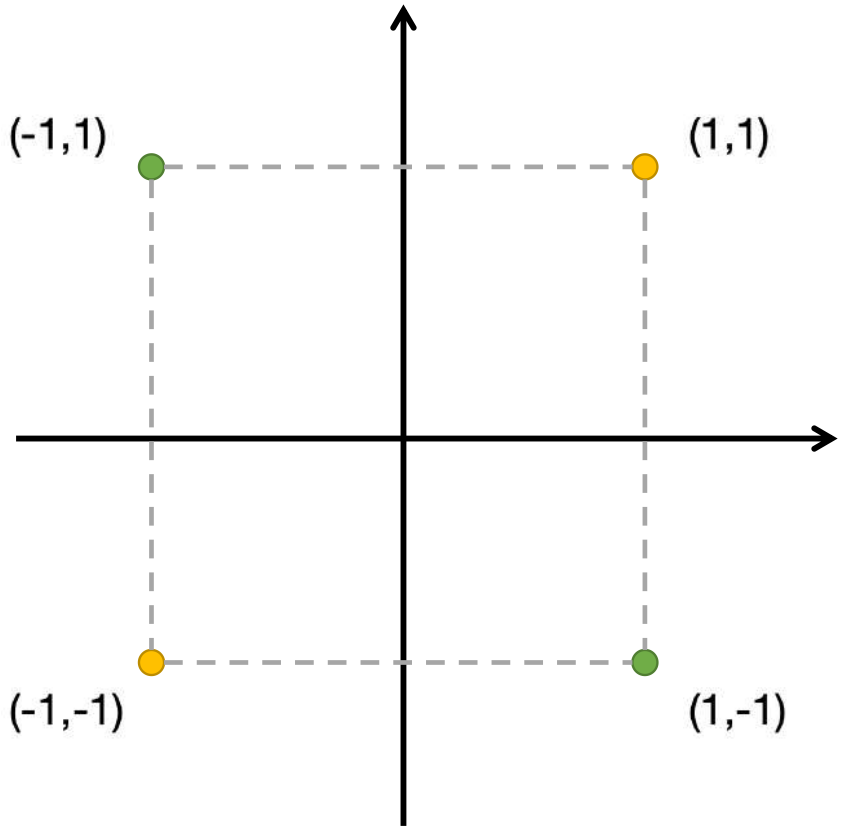
1. Wyznaczamy wartość na wyjściach sieci $$y_i^k(t) = f\left(s_i^k(t)\right) = f\left(\sum_{j=0}^{N_{k-1}} w_{ij}^k(t) x_j^k(t)\right)$$

2. Wyznaczamy błąd zaczynając od ostatniej warstwy $$Q_i^L(t) = \sum_{i=1}^{N_L} \left(\varepsilon_i^L\right)^2 = \sum_{i=1}^{N_L} \left(d_i^L(t) - y_i^L(t)\right)^2$$

   i w warstwach wcześniejszych $$Q_i^k(t) = \sum_{m=1}^{N_{k+1}} \delta_m^{k+1}(t) w_{mi}^{k+1}(t)$$

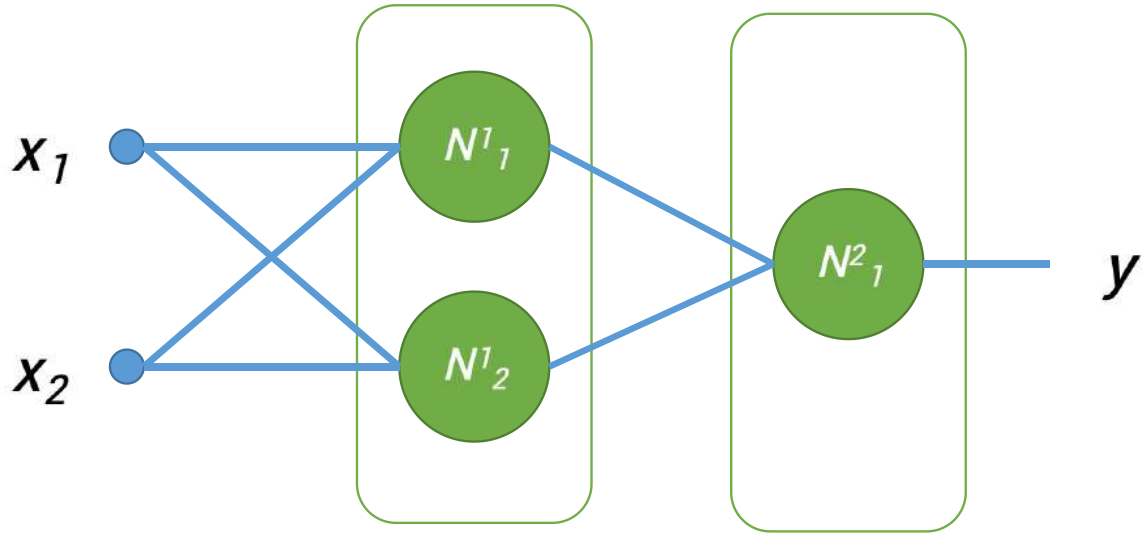
3. Modyfikujemy wagi wg wzoru $$w_{ij}^k(t+1) = w_{ij}^k(t) + 2\eta\delta_i^k(t) x_j^k(t)$$

Sieci neuronowe - przykład

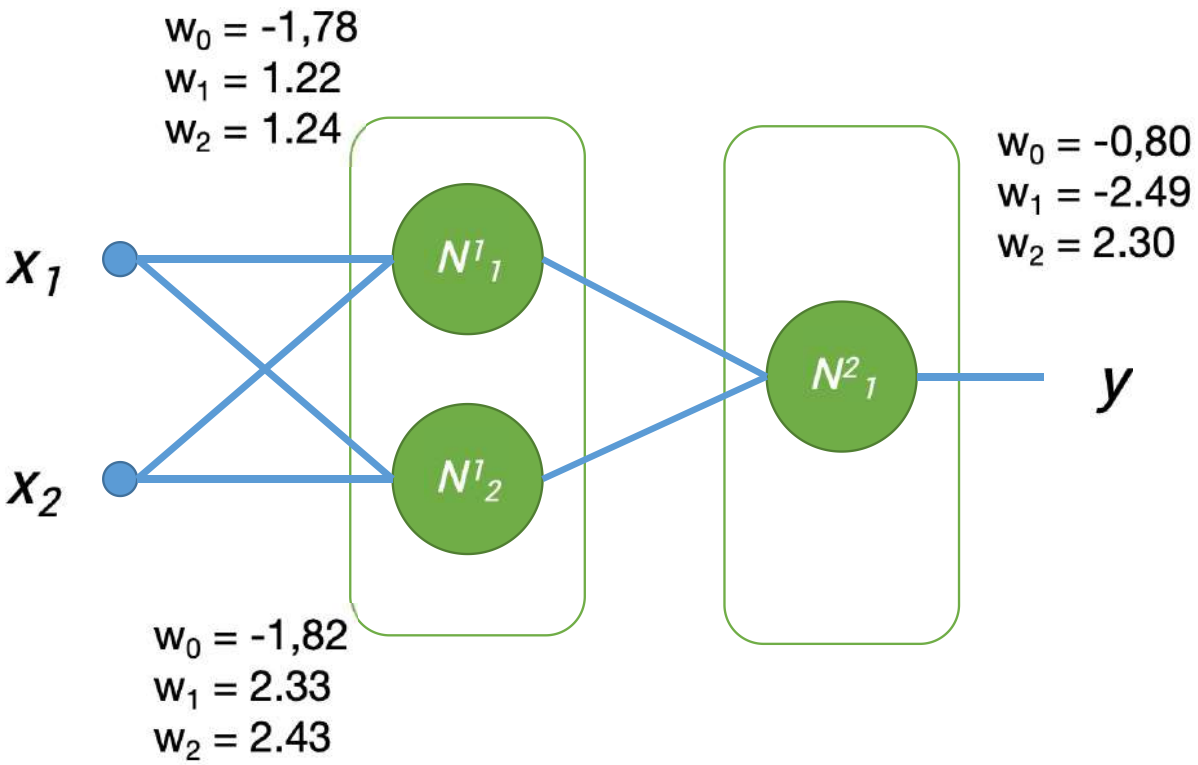


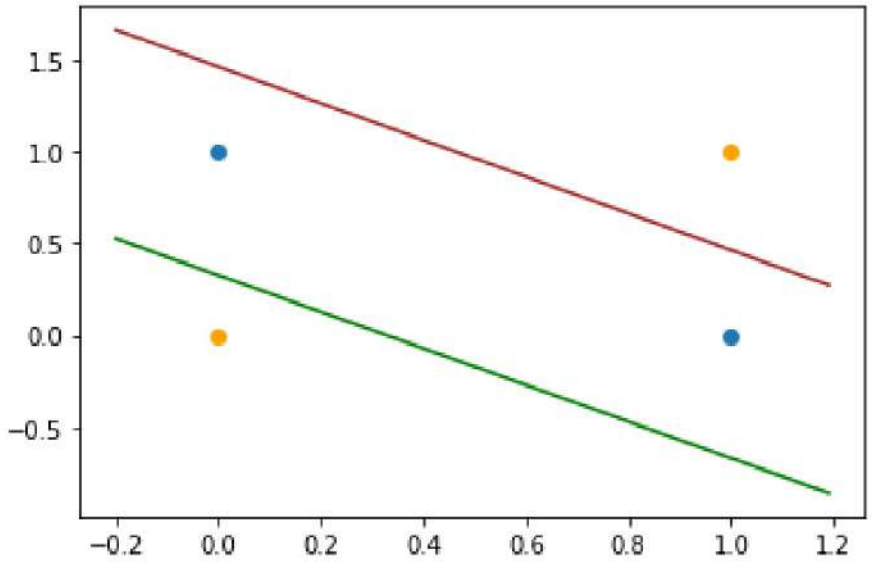
x_1	x_2	XOR(x_1,x_2)
1	1	-1
1	-1	1
-1	1	1
-1	-1	-1

Sieci neuronowe - przykład



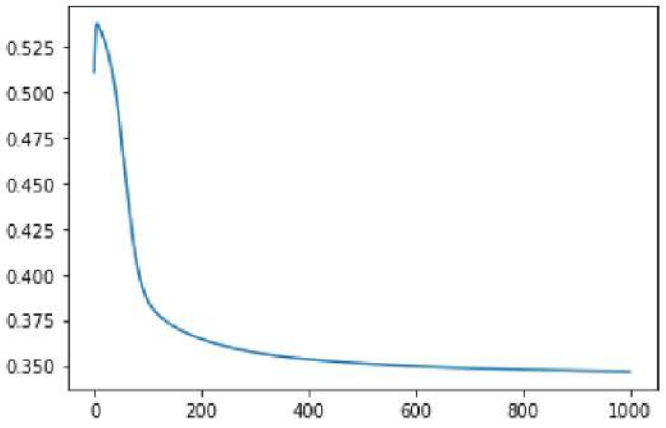
Sieci neuronowe - przykład



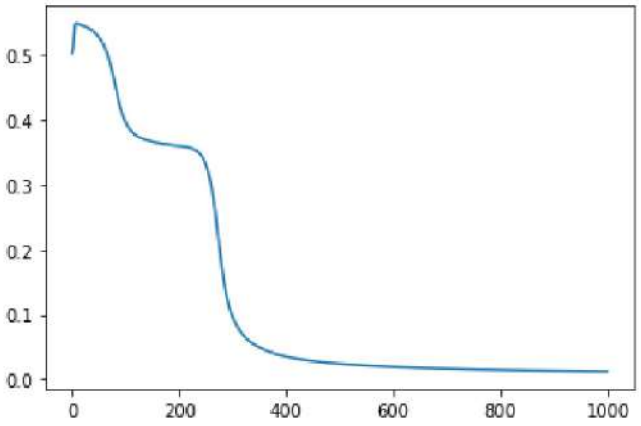


Sieci neuronowe - przykład

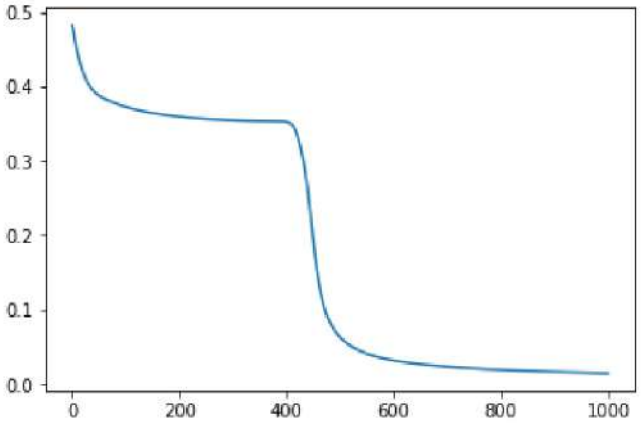
Uczenie 1



Uczenie 2



Uczenie 3



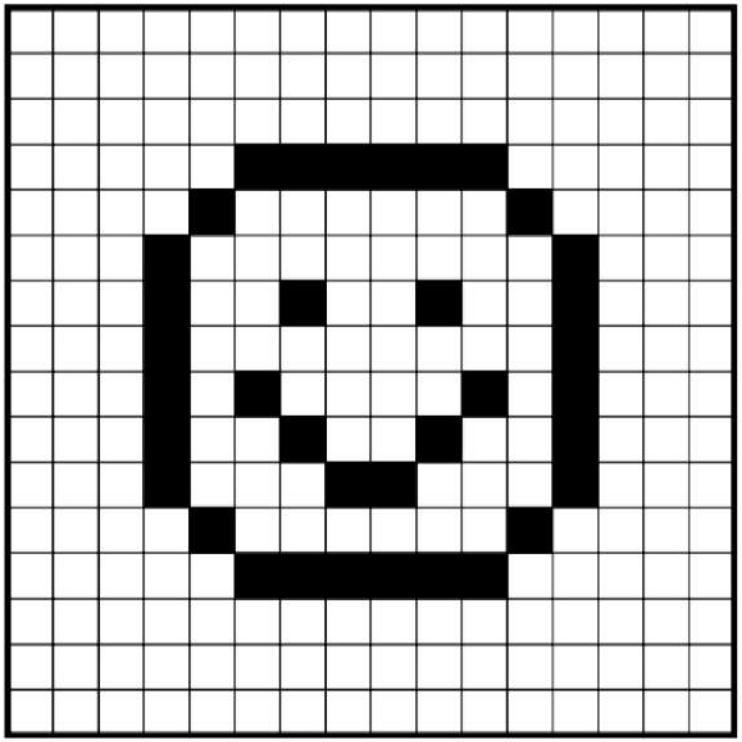
Konwolucyjne sieci neuronowe

Reprezentacja obrazów

- Obraz w odcieniach szarości jest macierzą
  wartości o 0 do 255
  reporezentujące intensywność
- 0 - czarny; 255 - biały
- Obrazy kolorowe wymagają 3 kanałów intensywności: czerwony, zielony, niebieski
- Wartości można znormalizować:

$$\hat{x} = \frac{x}{255} - 0.5$$

Reprezentacja obrazów

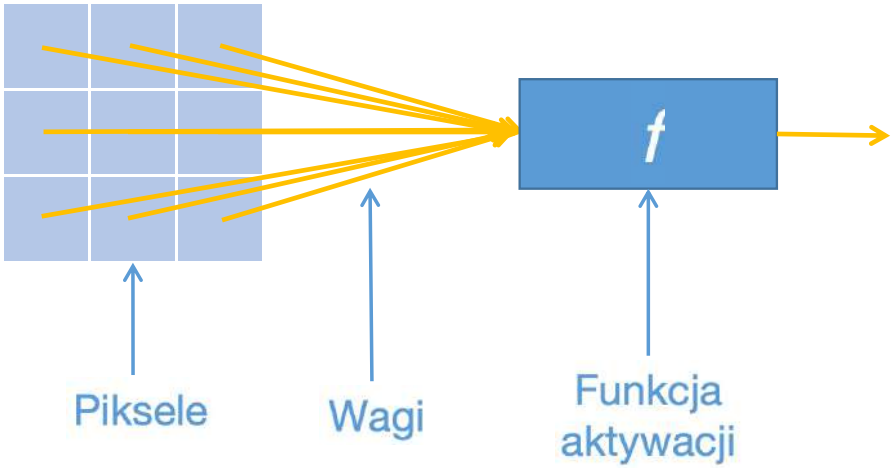


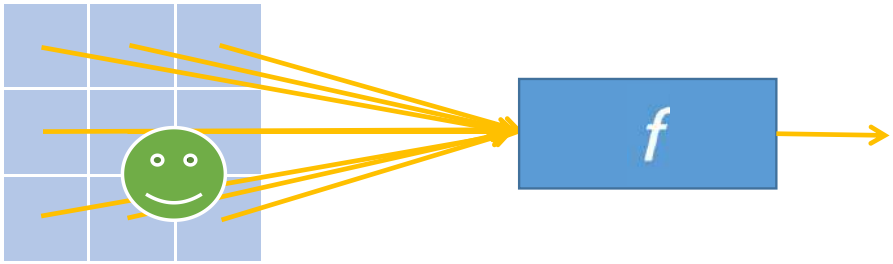
(a) Pixels

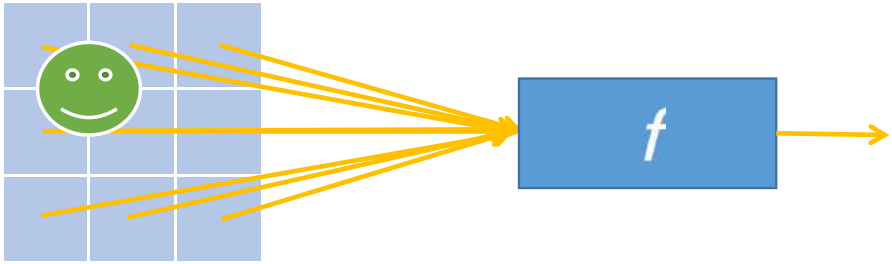
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	1	0	1	0	0	0	0	1	0	1	0	0	0
0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0
0	0	0	1	0	0	0	1	1	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(b) Numeric pixel values

Czy MLP będzie działał?







Konwolucje

Konwolucje są specjalnym rodzajem złożenia funkcji

wektor wejściowy $\mathbf{x}=(x_1, x_2, \ldots, x_n)^T$

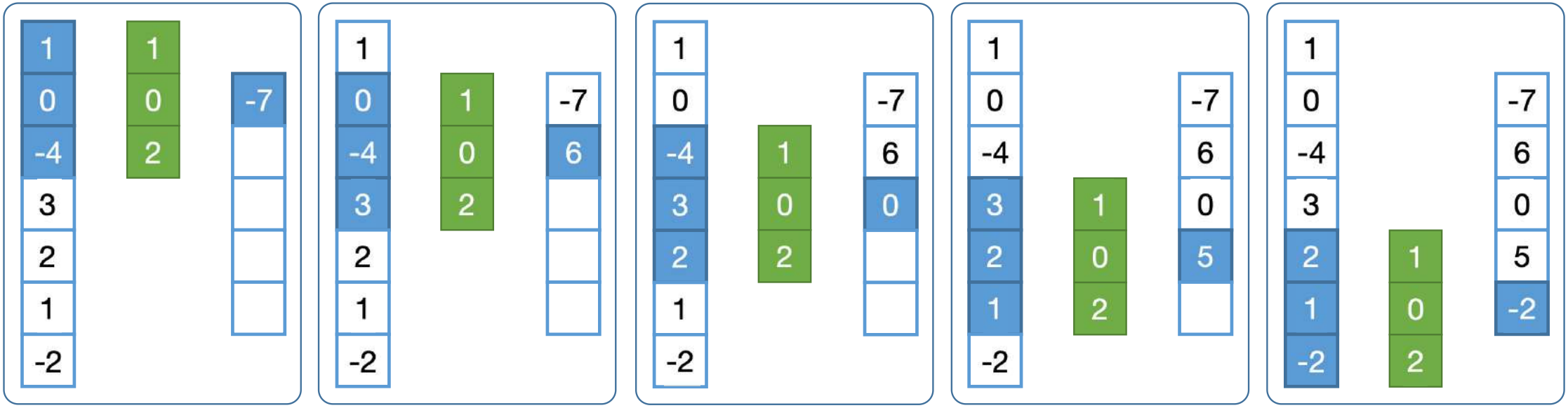
filtr 1D $\mathbf{w}=(w_1, w_2, \ldots, w_k)^T$

okno $\mathbf{x}_k(\mathrm{i})=(x_\mathrm{i}, x_{i+1}, x_{i+2}, \ldots, x_{i+k-1})^T$

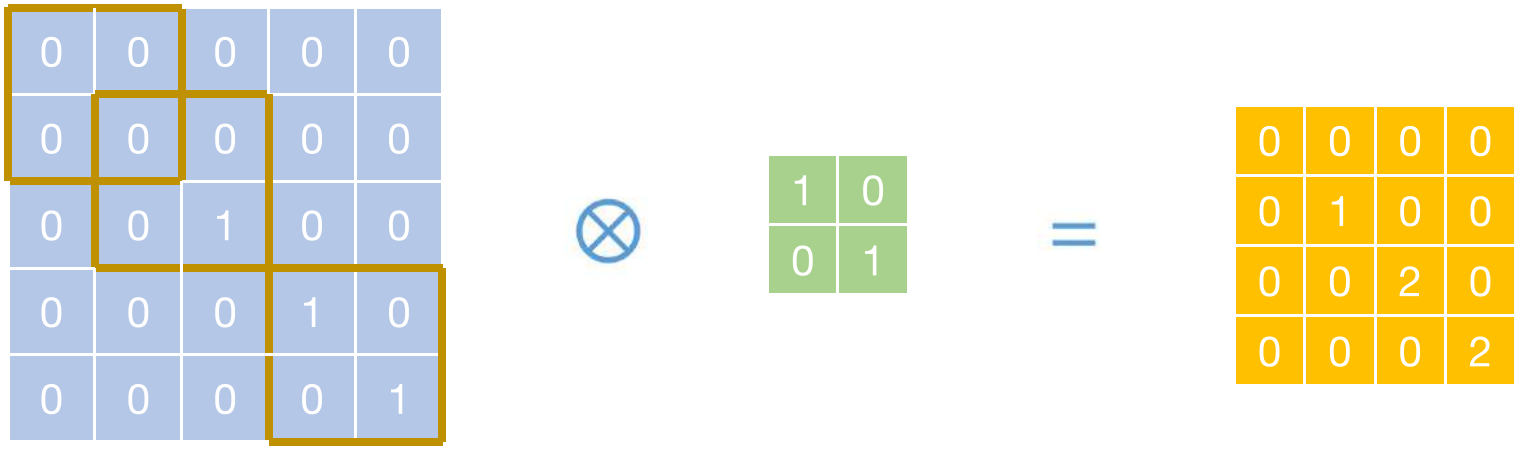
konwolcja 1D $$\mathbf{x} * \mathbf{w} = \left(\sum_{i=1}^{k} \left(\mathbf{x}_k(1) \otimes \mathbf{w}\right)(i), \ldots, \sum_{i=1}^{k} \left(\mathbf{x}_k(n-k+1) \otimes \mathbf{w}\right)(i) \right)^T$$

Konwolucje

Konwolucje są specjalnym rodzajem złożenia funkcji



Konwolucje 2D



Rezultat konwolucji jest zawsze mniejszy od obrazu wejściowego

Przykłady filtrów

2	−1	−1
−1	2	−1
−1	−1	2

(a) Diagonal

−1	2	−1
−1	2	−1
−1	2	−1

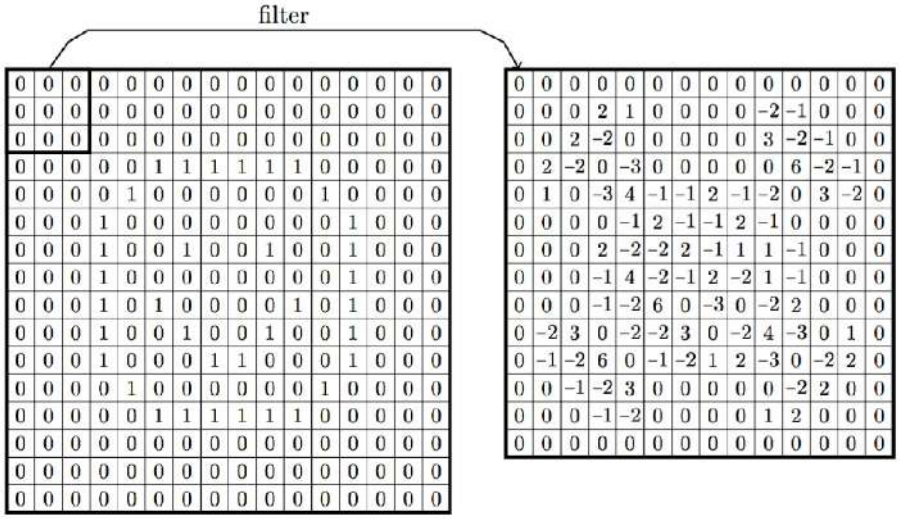
(b) Vertical

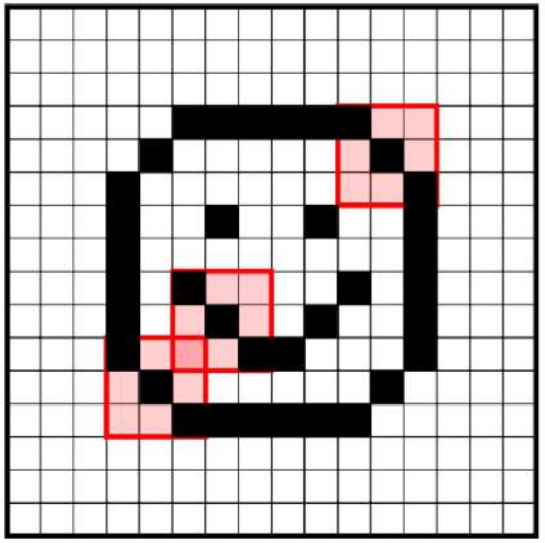
−1	−1	−1
2	2	2
−1	−1	−1

(c) Horizontal

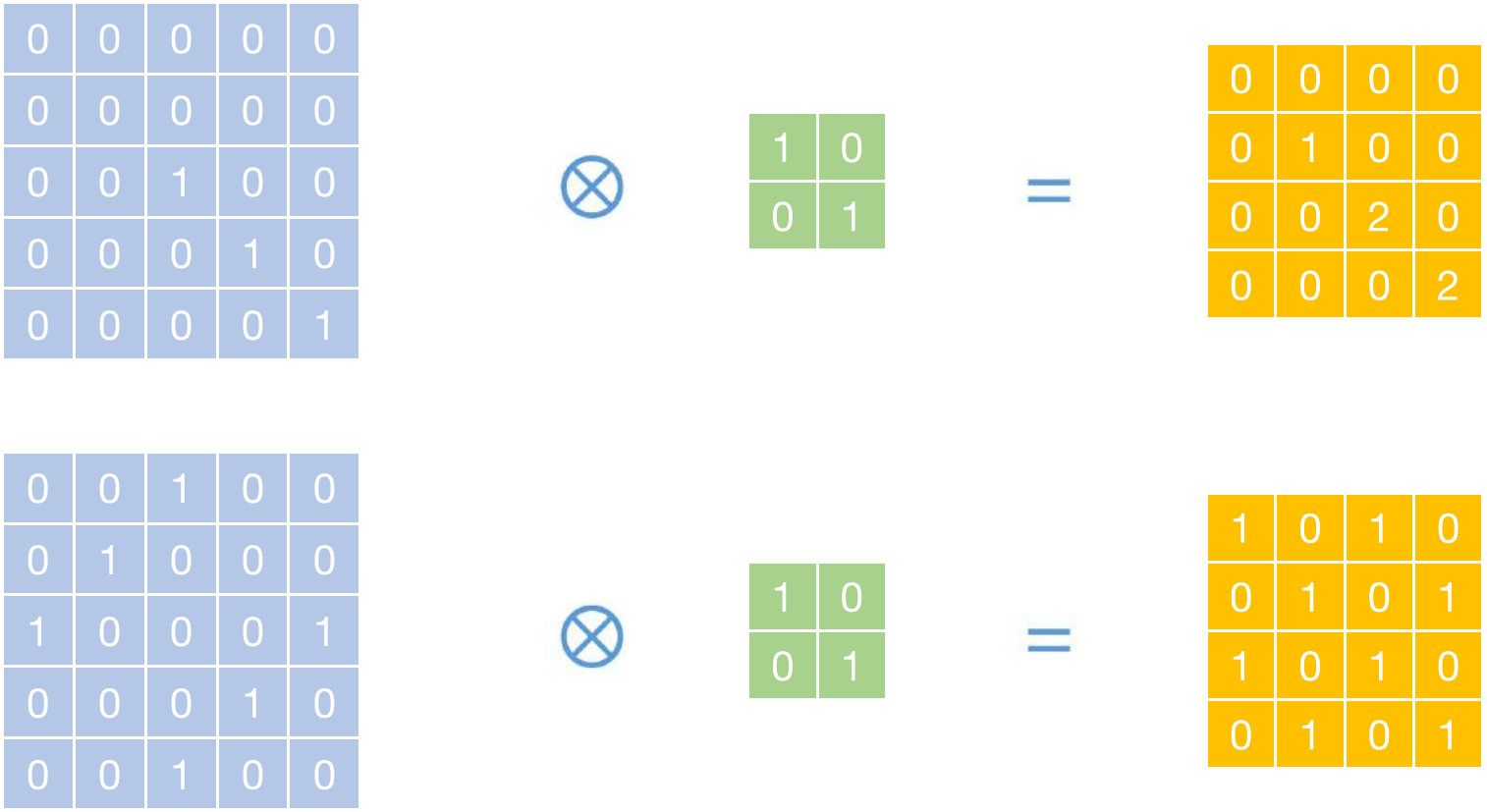
−1	−1	2
−1	2	−1
2	−1	−1

(d) Anti-diagonal

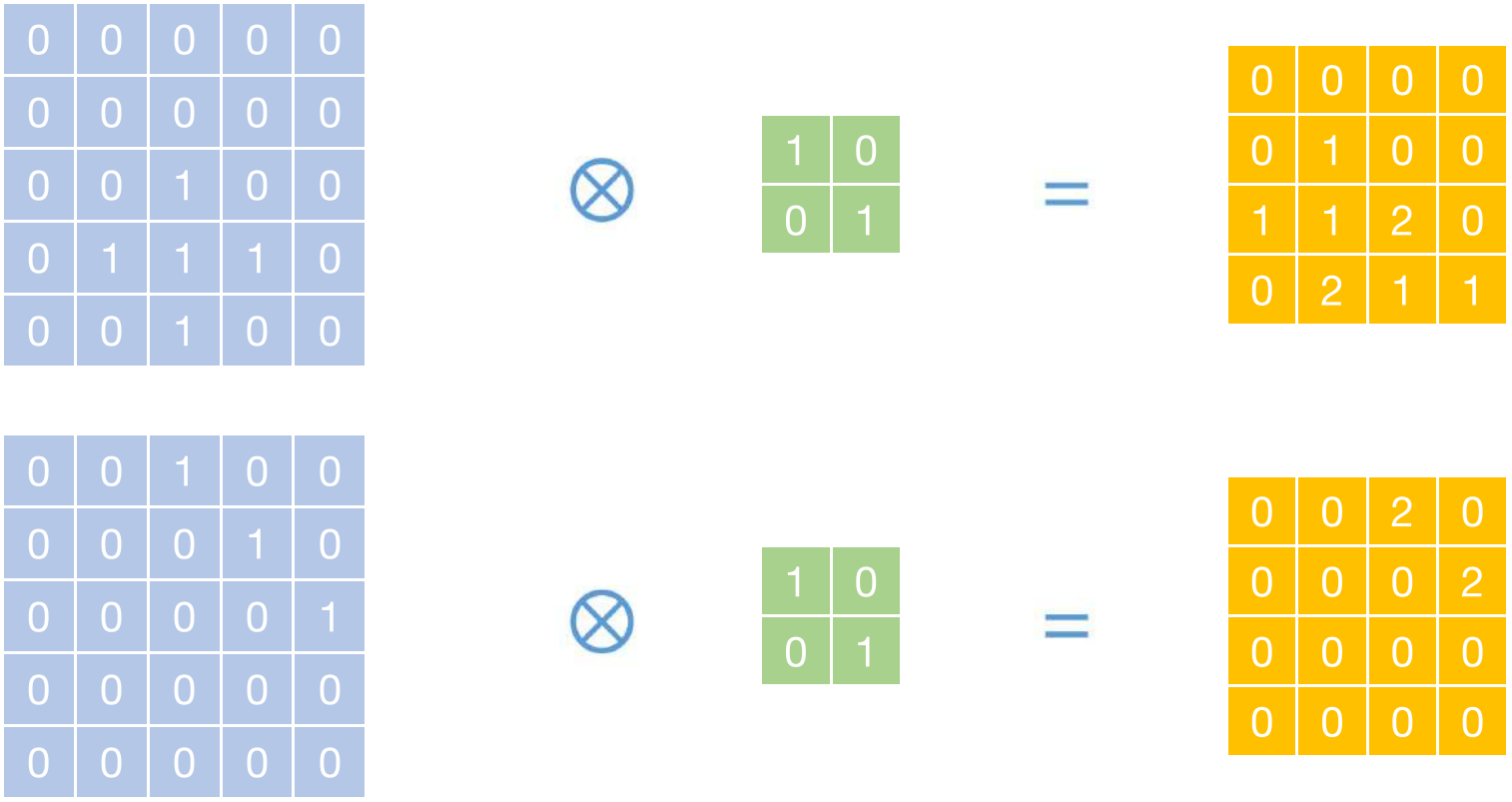




Konwolucje 2D



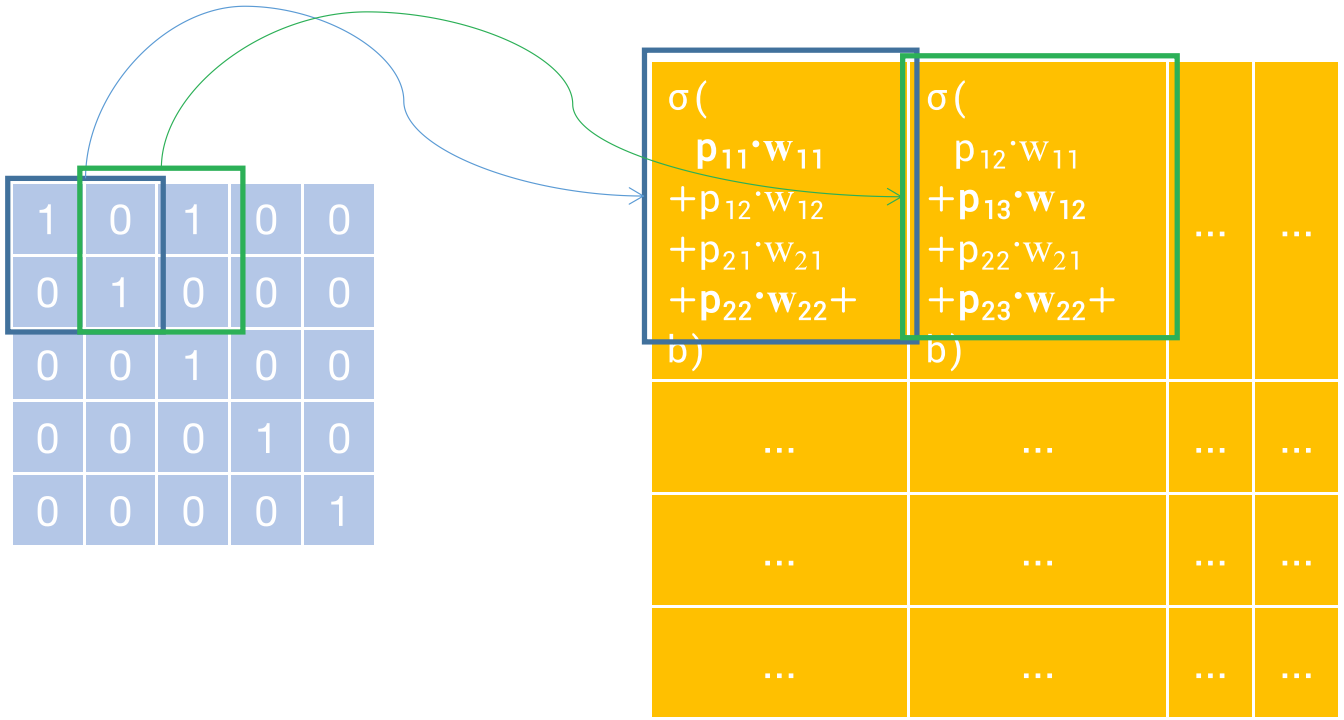
Konwolucje 2D



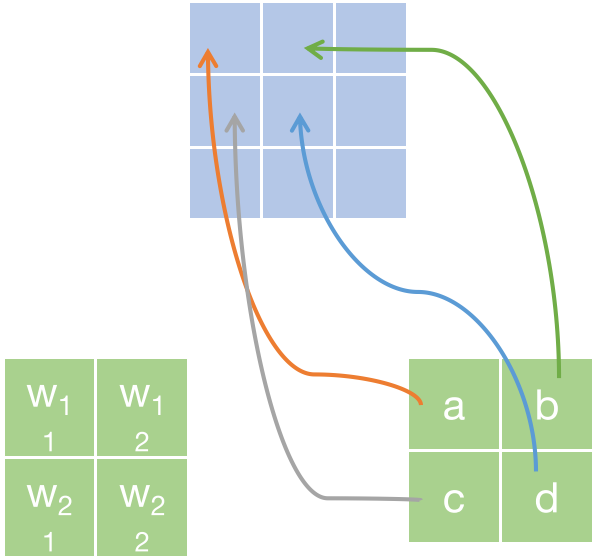
Konwolucje w sieciach neuronowych

Elementy wspólne dla wszystkich neuronów: wagi i bias

$w_{1\,1}$	$w_{1\,2}$
$w_{2\,1}$	$w_{2\,2}$



Uczenie sieci konwolucyjnych

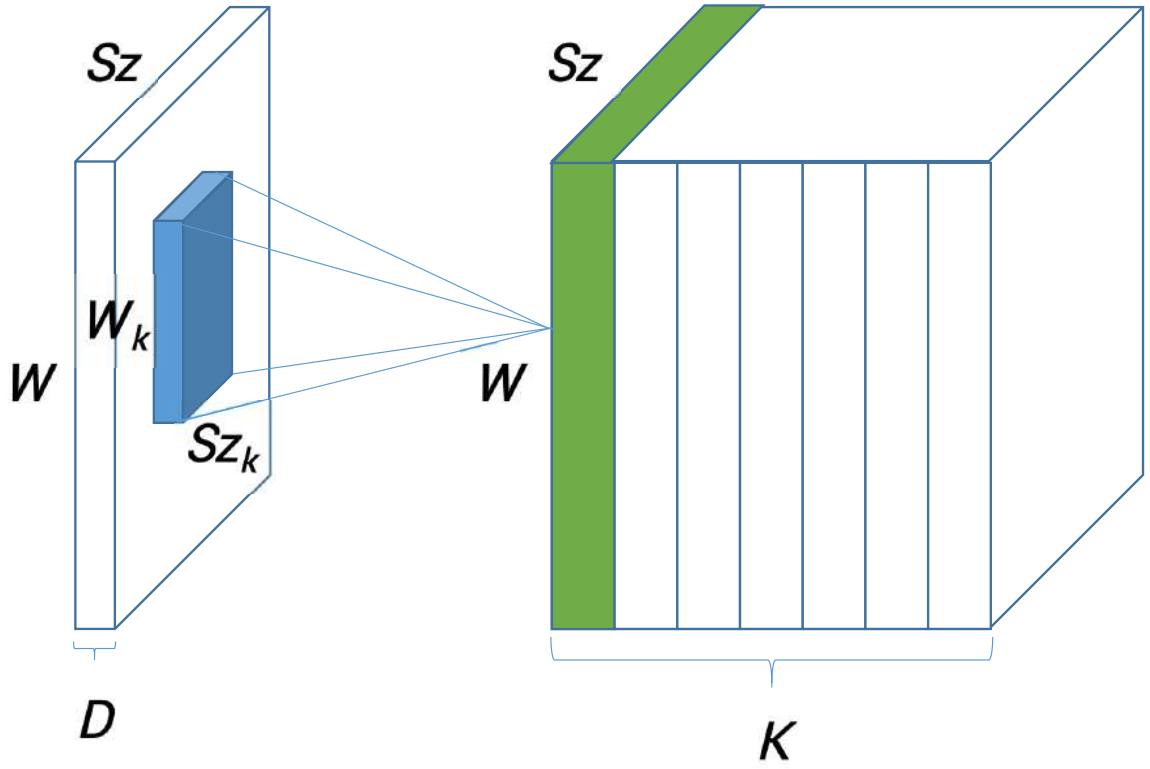


$$a(t+1) = a(t) - \gamma \frac{\partial L}{\partial a} \qquad b(t+1) = b(t) - \gamma \frac{\partial L}{\partial b}$$

$$c(t+1) = c(t) - \gamma \frac{\partial L}{\partial c} \qquad d(t+1) = d(t) - \gamma \frac{\partial L}{\partial d}$$

$$w_{11}(t+1) = w_{11}(t) - \gamma \left(\frac{\partial L}{\partial a} + \frac{\partial L}{\partial b} + \frac{\partial L}{\partial c} + \frac{\partial L}{\partial d} \right)$$

Jedno jądro to za mało



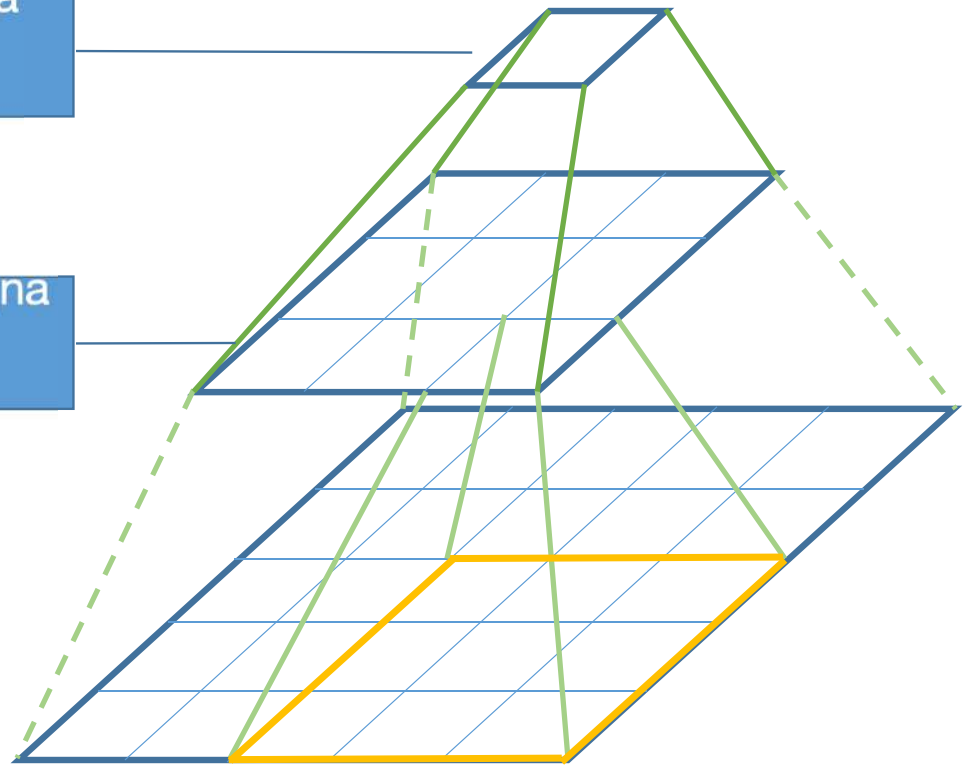
Obraz o rozmiarach: $Sz \cdot W$
Liczba wymiarów: D
Rozmiar jądra: $Sz_k \cdot W_k$
Liczba jąder: K

Łączna liczba parametrów do uczenia
$(Sz_k \cdot W_k \cdot D+1) \cdot K$

Jedna wartstwa to też za mało

Druga warstwa konwolucyjna
3x3
pole postrzegania 5x5

Pierwsza warstwa konwolucyjna
3x3
pole postrzegania 3x3



Im większy obraz tym więcej warstw konwolucyjnych będziemy potrzebować

Filtry

Możemy definiować filtry (konwolucje) i stosować je do obrazu

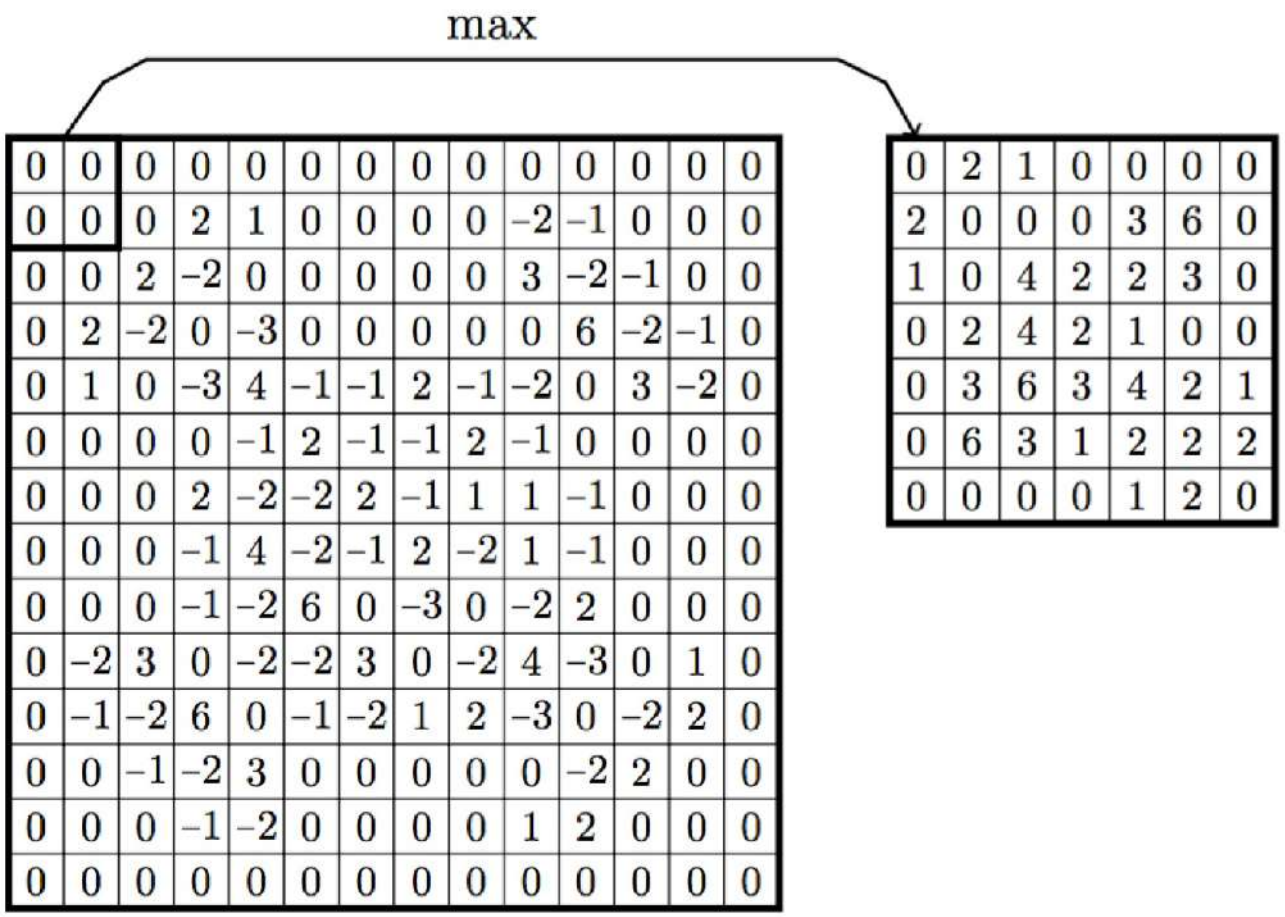
- Filtry w sieciach konwolucyjnych są uczone
- W pierwszej warstwie konwolucyjnej filtry są stosowane bezpośrednio do obrazu
- W drugiej warstwie konwolucyjnej filtry są stosowane do wyjścia pierwszej warstwy itd.
- Ostatania warstwa jest w pełni połączona

Warstwy typu pooling

Są to specjalne warstwy, które stosują stałe, proste filtry, nie podlegające uczeniu

- służą przede wszystkim do redukcji rozmiaru
- najpopularniejszy - max pooling - wybiera maksymalną wartość z danego okna

Max pooling - przykład



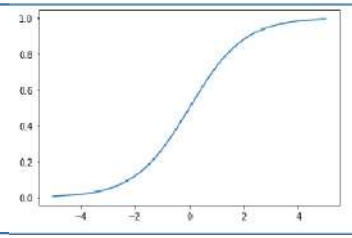
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	2	1	0	0	0	0	−2	−1	0	0	0
0	0	2	−2	0	0	0	0	0	3	−2	−1	0	0
0	2	−2	0	−3	0	0	0	0	0	6	−2	−1	0
0	1	0	−3	4	−1	−1	2	−1	−2	0	3	−2	0
0	0	0	0	−1	2	−1	−1	2	−1	0	0	0	0
0	0	0	2	−2	−2	2	−1	1	1	−1	0	0	0
0	0	0	−1	4	−2	−1	2	−2	1	−1	0	0	0
0	0	0	−1	−2	6	0	−3	0	−2	2	0	0	0
0	−2	3	0	−2	−2	3	0	−2	4	−3	0	1	0
0	−1	−2	6	0	−1	−2	1	2	−3	0	−2	2	0
0	0	−1	−2	3	0	0	0	0	0	−2	2	0	0
0	0	0	−1	−2	0	0	0	0	1	2	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0

0	2	1	0	0	0	0
2	0	0	0	3	6	0
1	0	4	2	2	3	0
0	2	4	2	1	0	0
0	3	6	3	4	2	1
0	6	3	1	2	2	2
0	0	0	0	1	2	0

Funkcje aktywacji

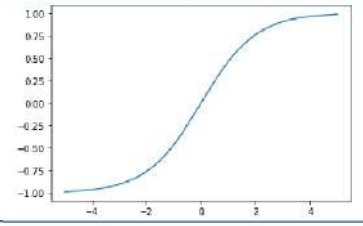
Funkcja sigmoidalna

$$f(x) = \frac{1}{1+\exp(-x)}$$



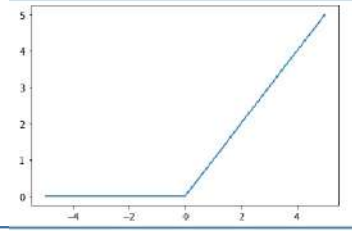
Funkcja tanh

$$f(x) = \frac{1-\exp(-x)}{1+\exp(-x)}$$



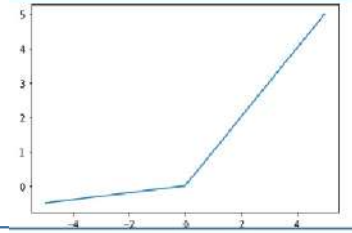
Funkcja ReLU

$$f(x) = \max(0, x)$$

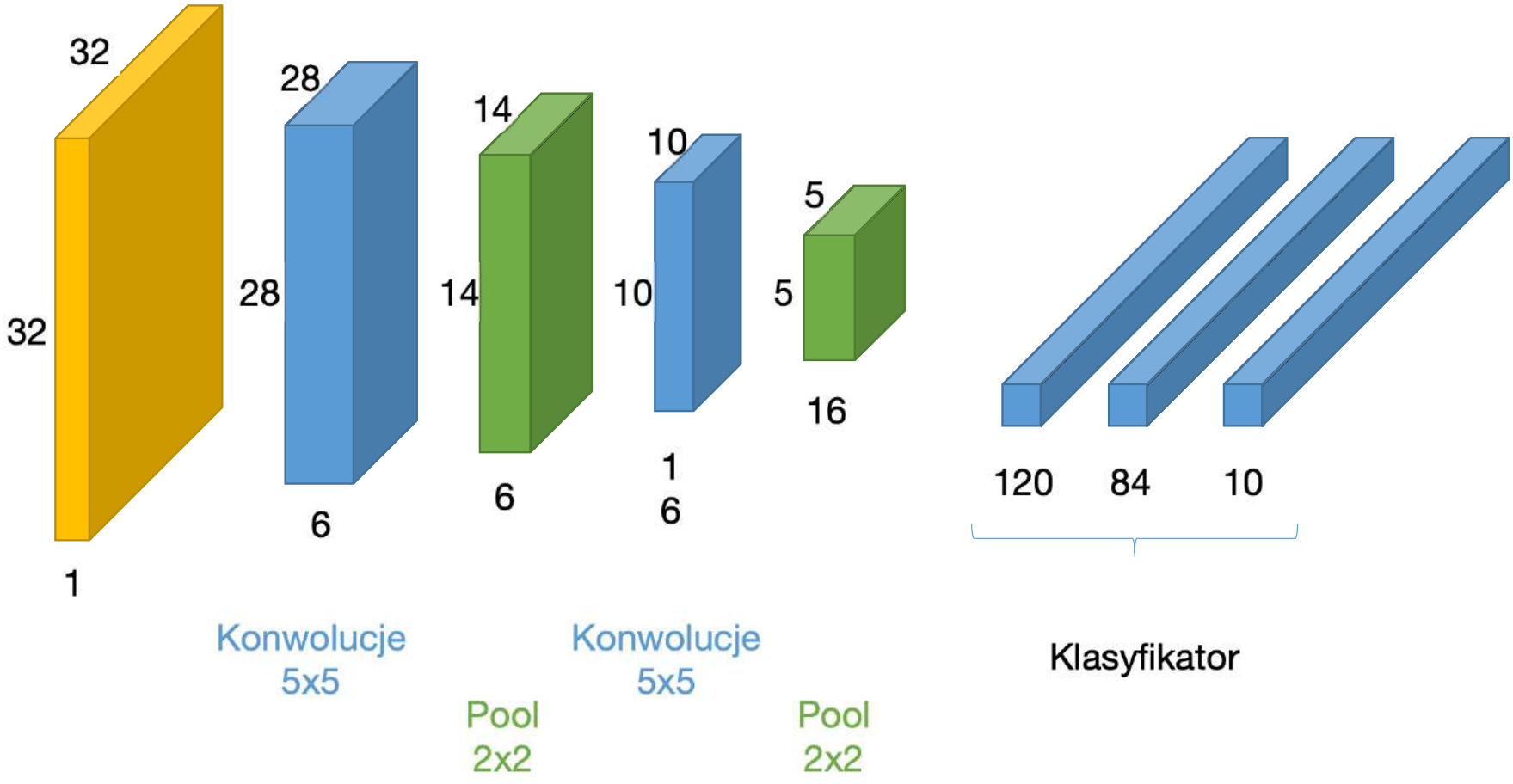


Funkcja Leaky ReLU

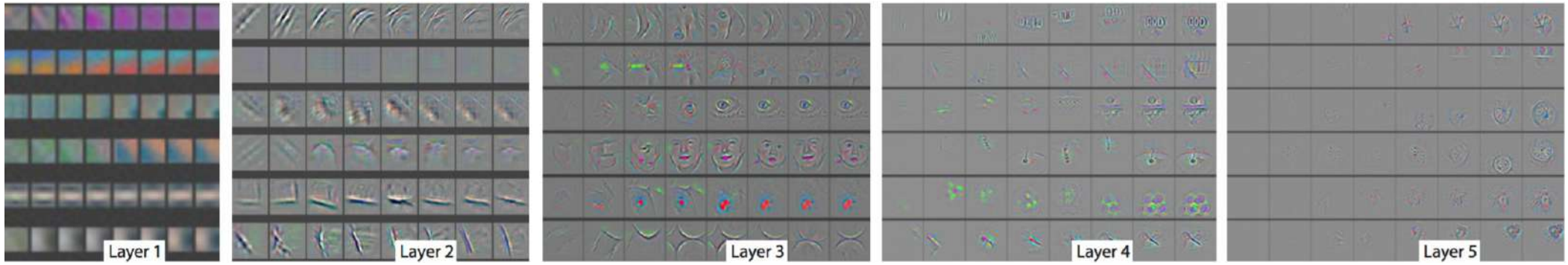
$$f(x) = \max(ax, x)$$



LeNet

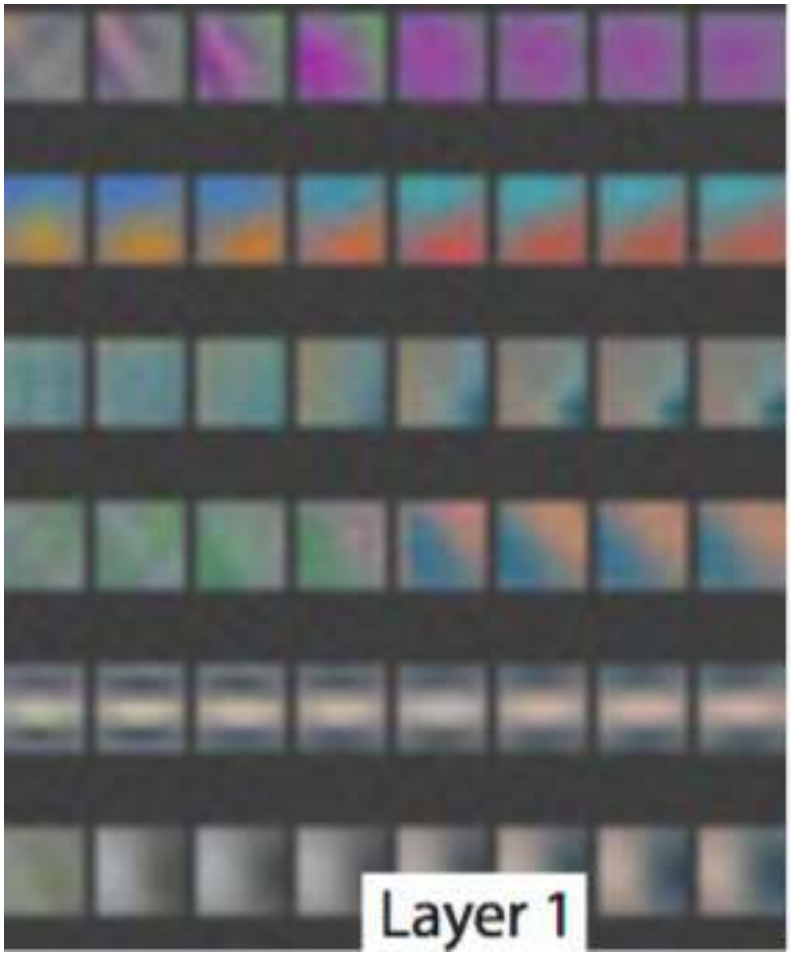

32
32
1
28
28
6
14
14
6
10
10
1
6
5
5
16
120
84
10
Konwolucje
5x5
Pool
2x2
Konwolucje
5x5
Pool
2x2
Klasyfikator

Wizualizacja CNN

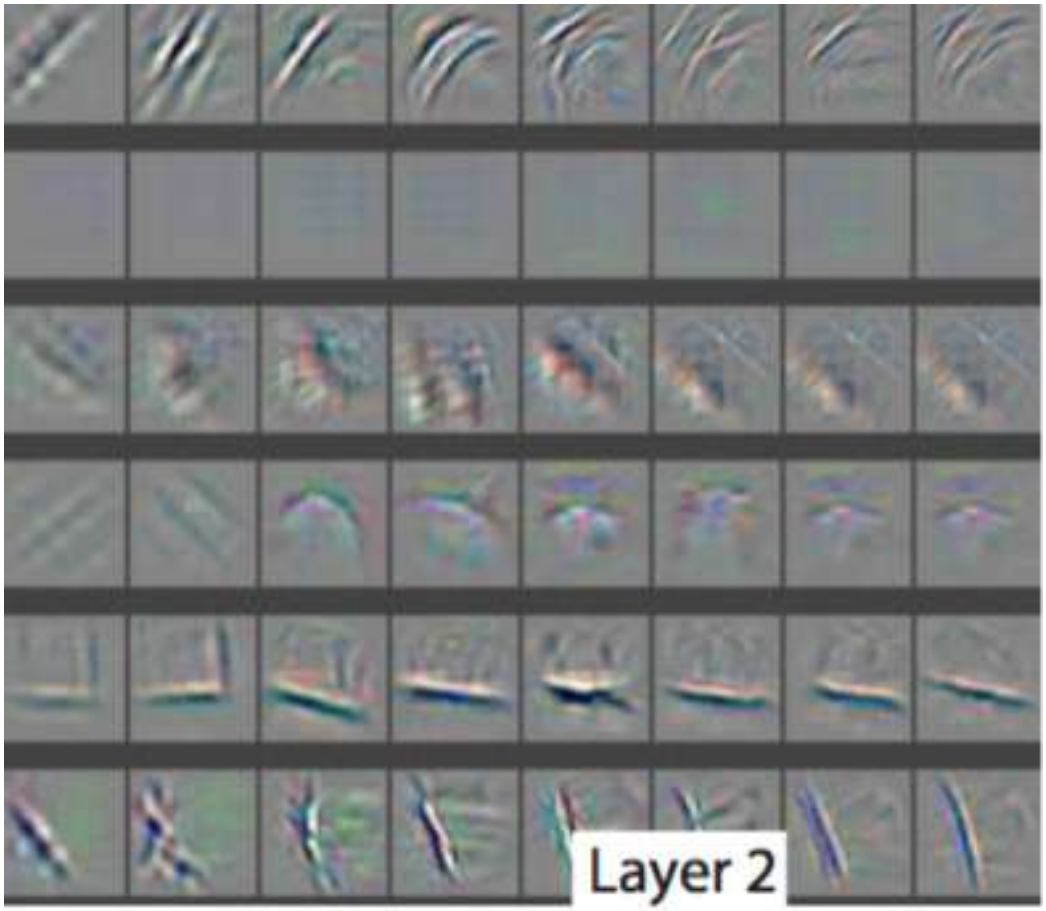


- Warstwy konwolucyjne 1:5
- Każdy wiersz zawiera losowo wybrane filtry
- Każda kolumna odpowiada epoce uczenia 1,2,5,10,20,30,40,64

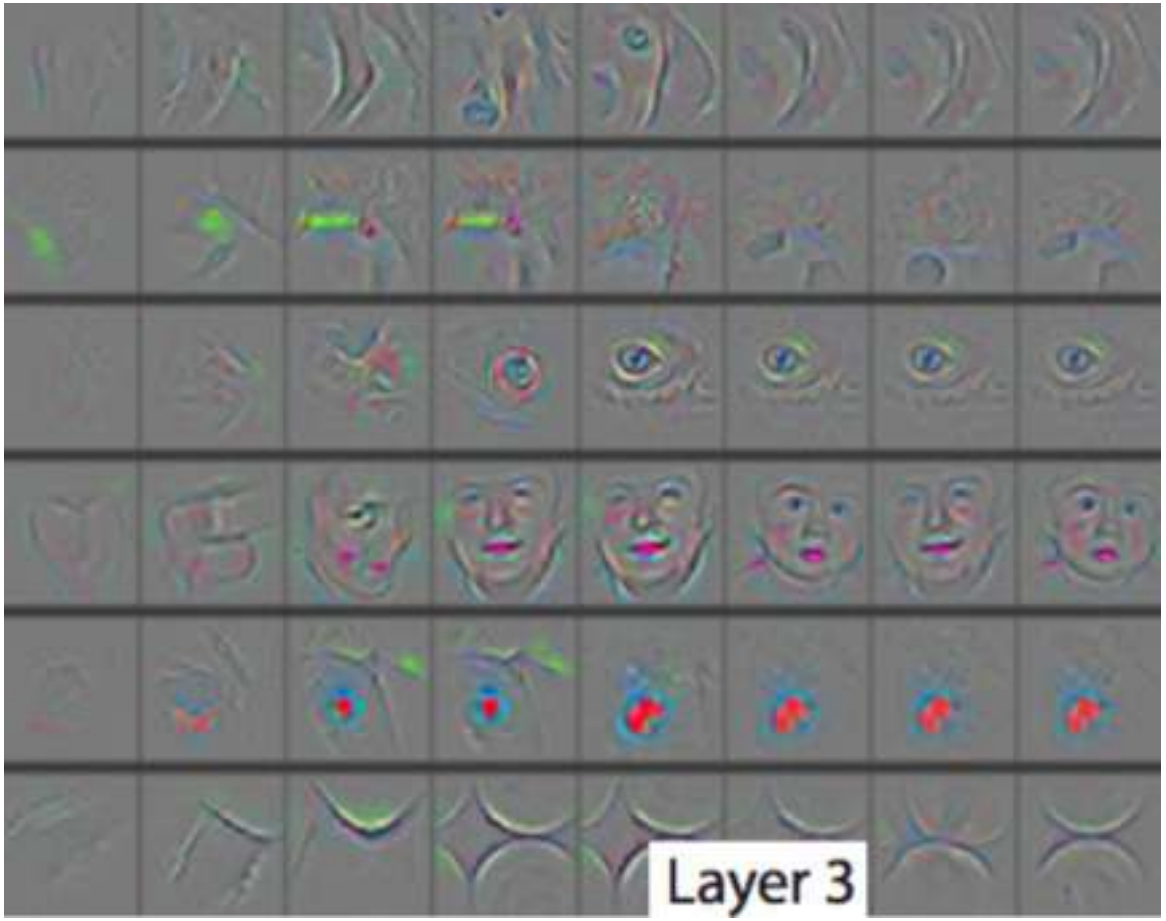
Wizualizacja CNN - 1 warstwa



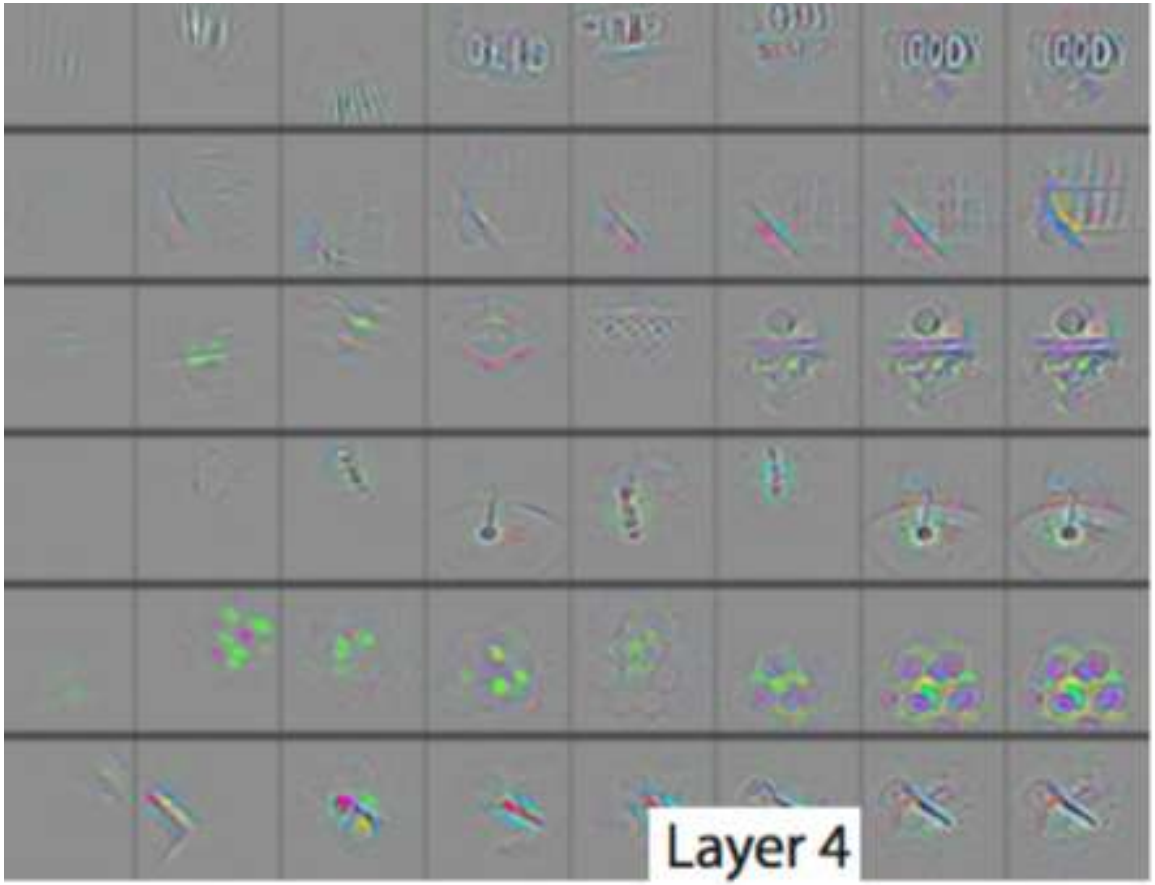
Wizualizacja CNN - 2 warstwa



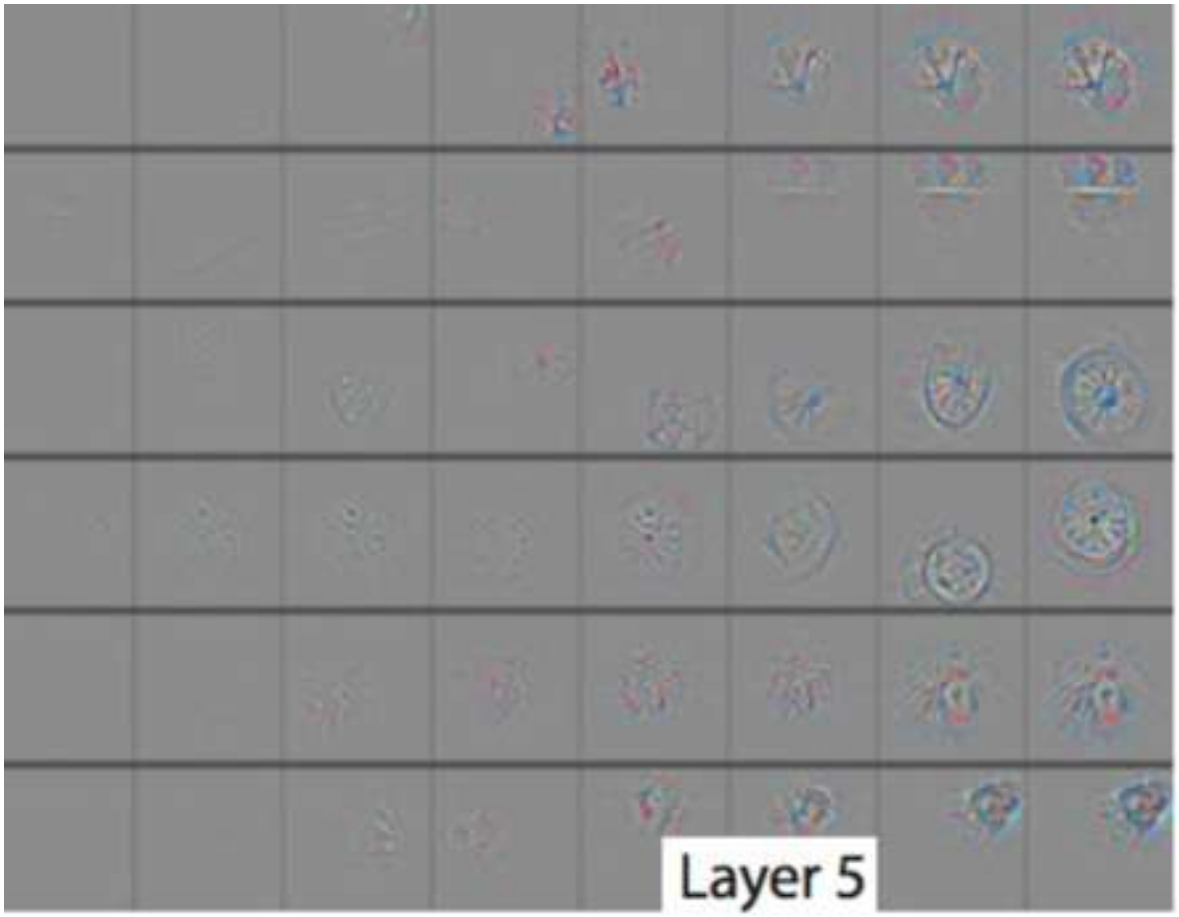
Wizualizacja CNN - 3 warstwa



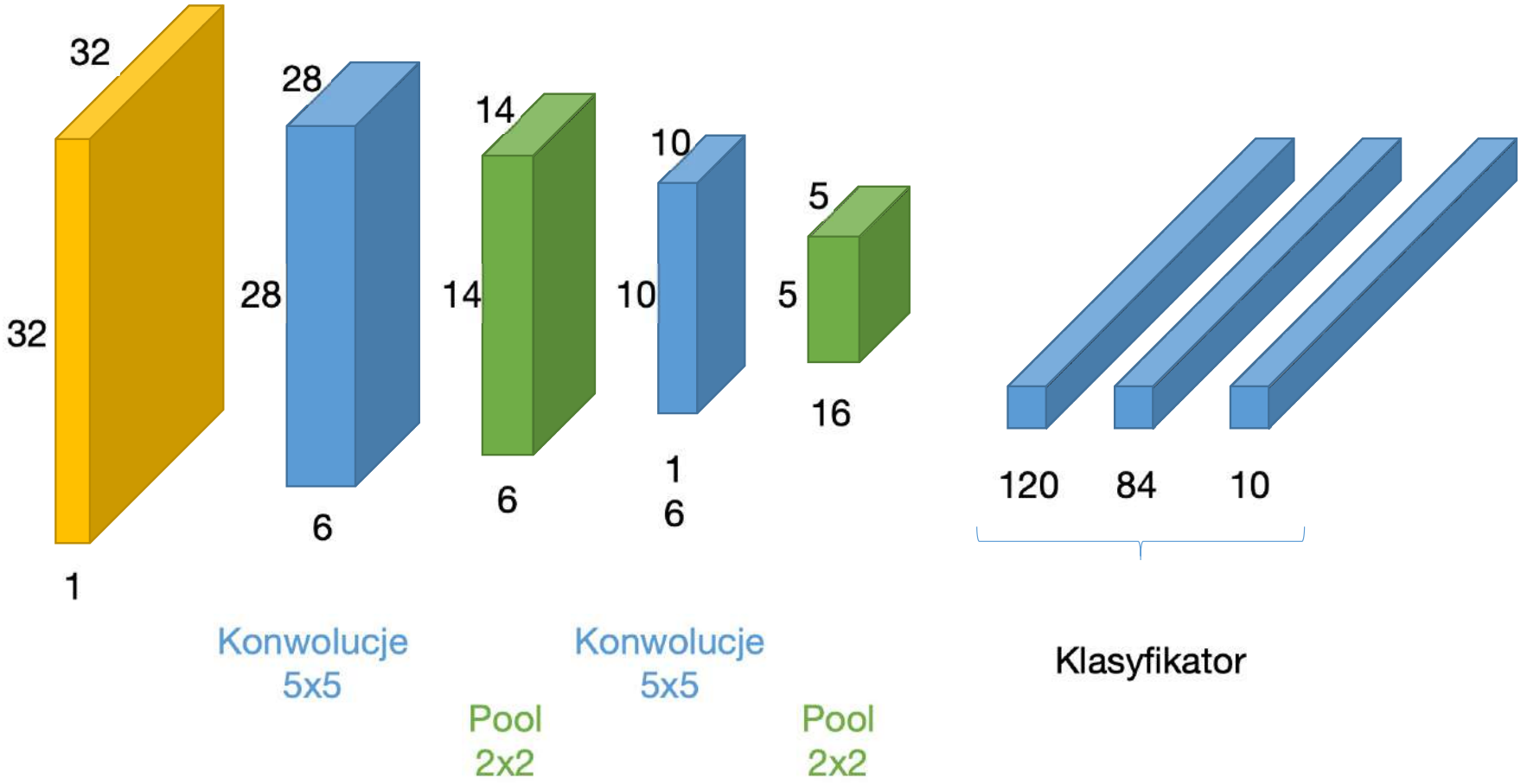
Wizualizacja CNN - 4 warstwa



Wizualizacja CNN - 5 warstwa



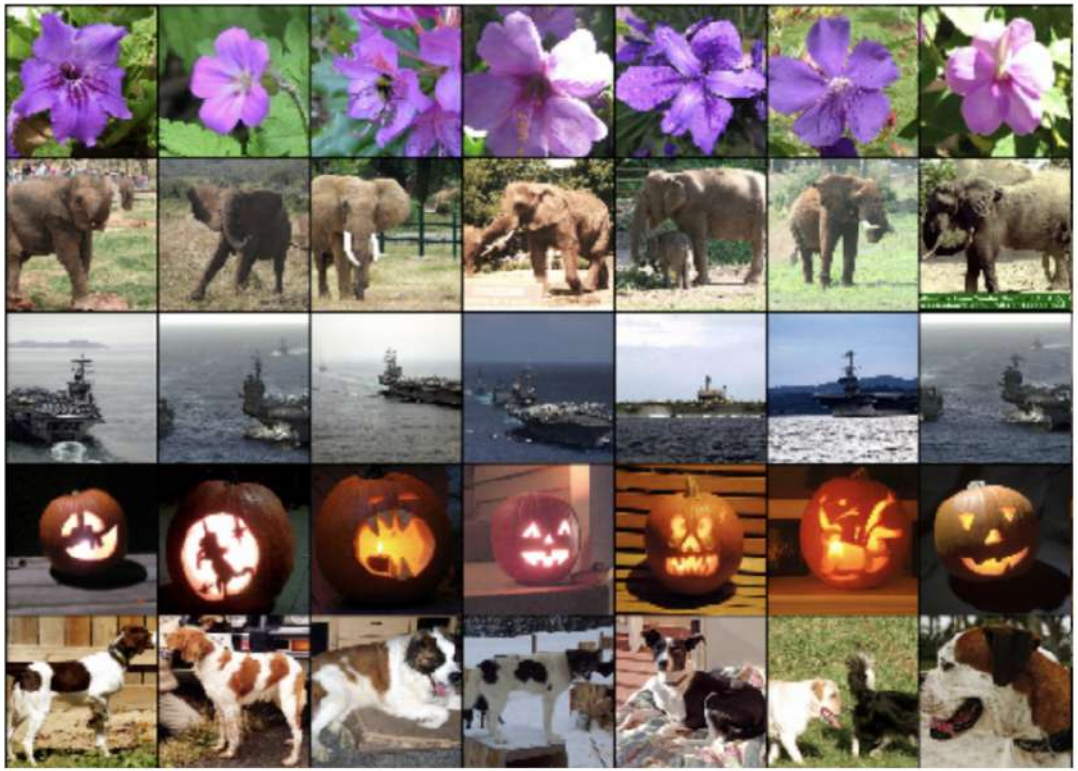
LeNet



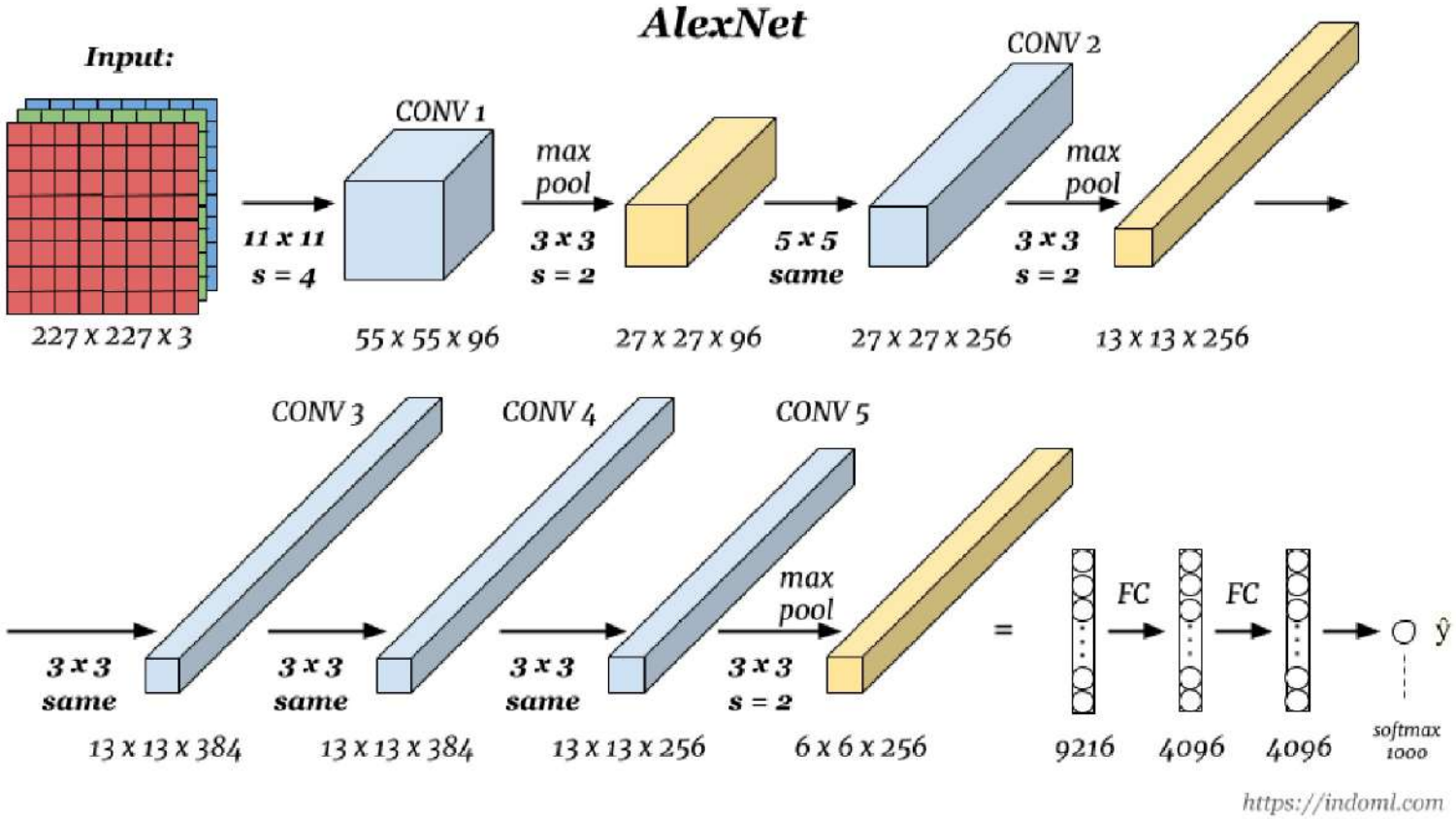
Liczba parametrów do uczenia: ok. 60000

Zbiór danych Image Net

Ponad 1000 klas, ponad 14000000 opisanych obrazów

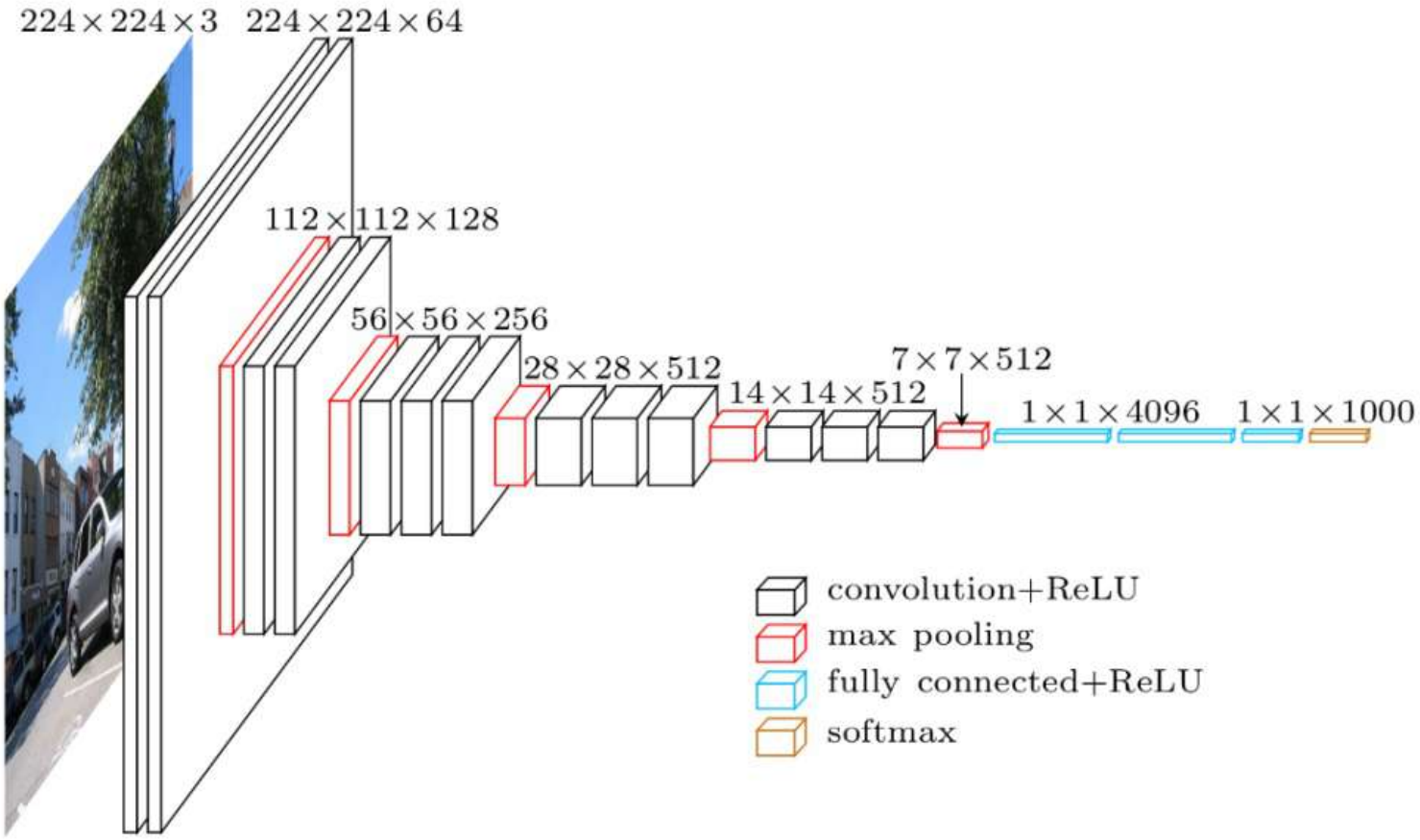


AlexNet



Liczba parametrów do uczenia: ok. 60 000 000

VGG-16



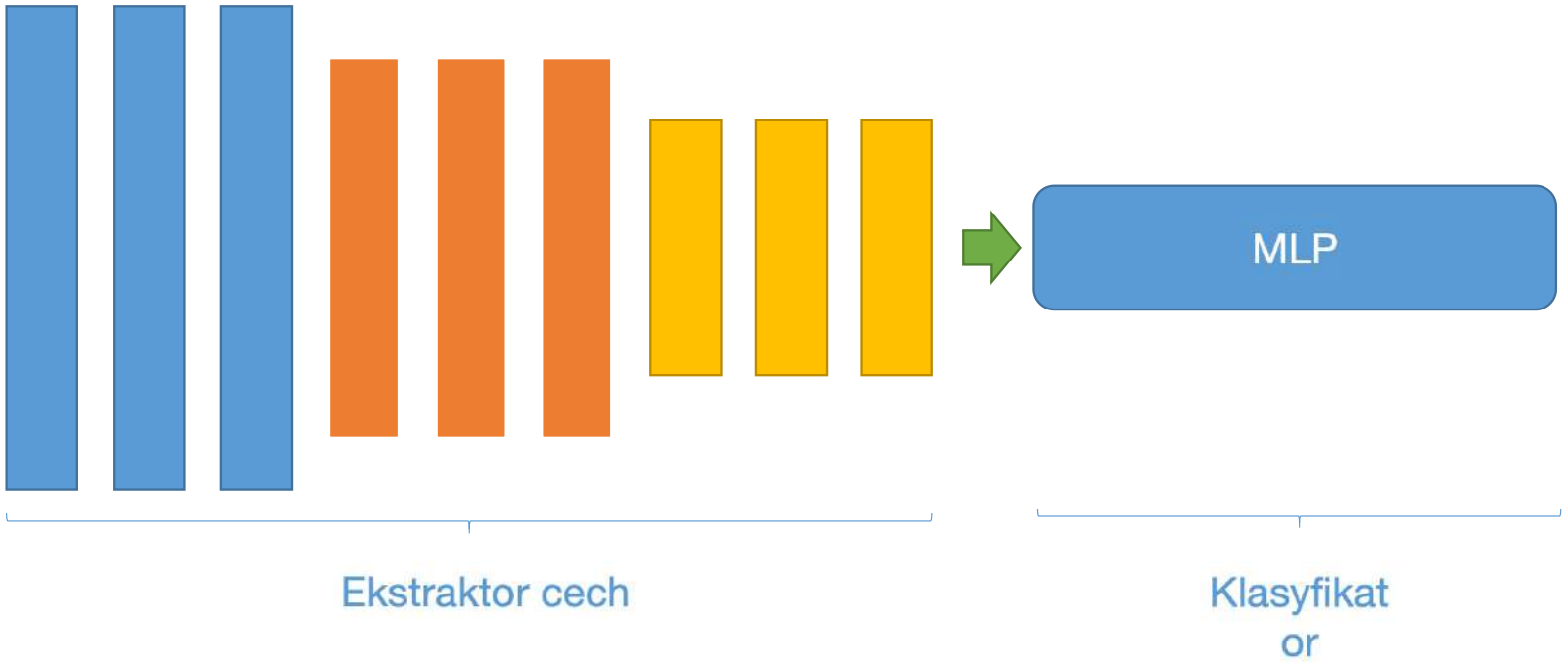
Liczba parametrów do uczenia: ok. 138 000 000

Inne popularne architektury

VGG	Pobodna do AlexNet, 138 milionów parametrów. Błąd 5 najlepszych: 8% (jeden model)
Inception V3	25 milionów parametrów. Błąd 5 najlepszych: 5.6% (jeden model), 3.6% (zesp
ResNet	60 milionów parametrów. 152 warstwy Błąd 5 najlepszych: 4.5% (jeden model), 3.5% (zesp

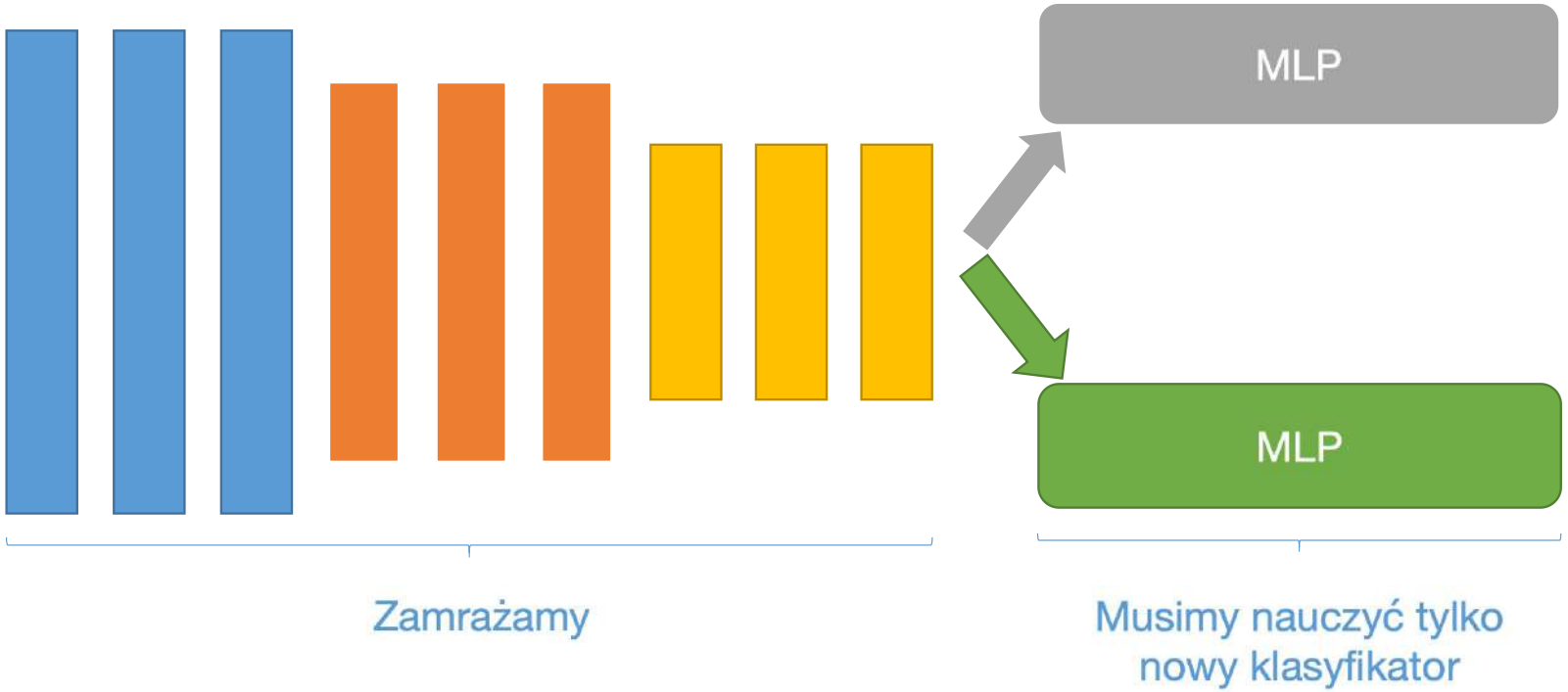
Uczenie transferowe

Uczenie głębokiej sieci neuronowej od podstaw wymaga wiele danych i czasu.



Uczenie transferowe

Uczenie transferowe polega na wykorzystaniu istniejącej sieci do realizacji nowych zadań



Uczenie transferowe

Uczenie transferowe polega na wykorzystaniu istniejącej sieci do realizacji nowych zadań

Wymaga zdecydowanie mniejszej liczby przykadów uczących

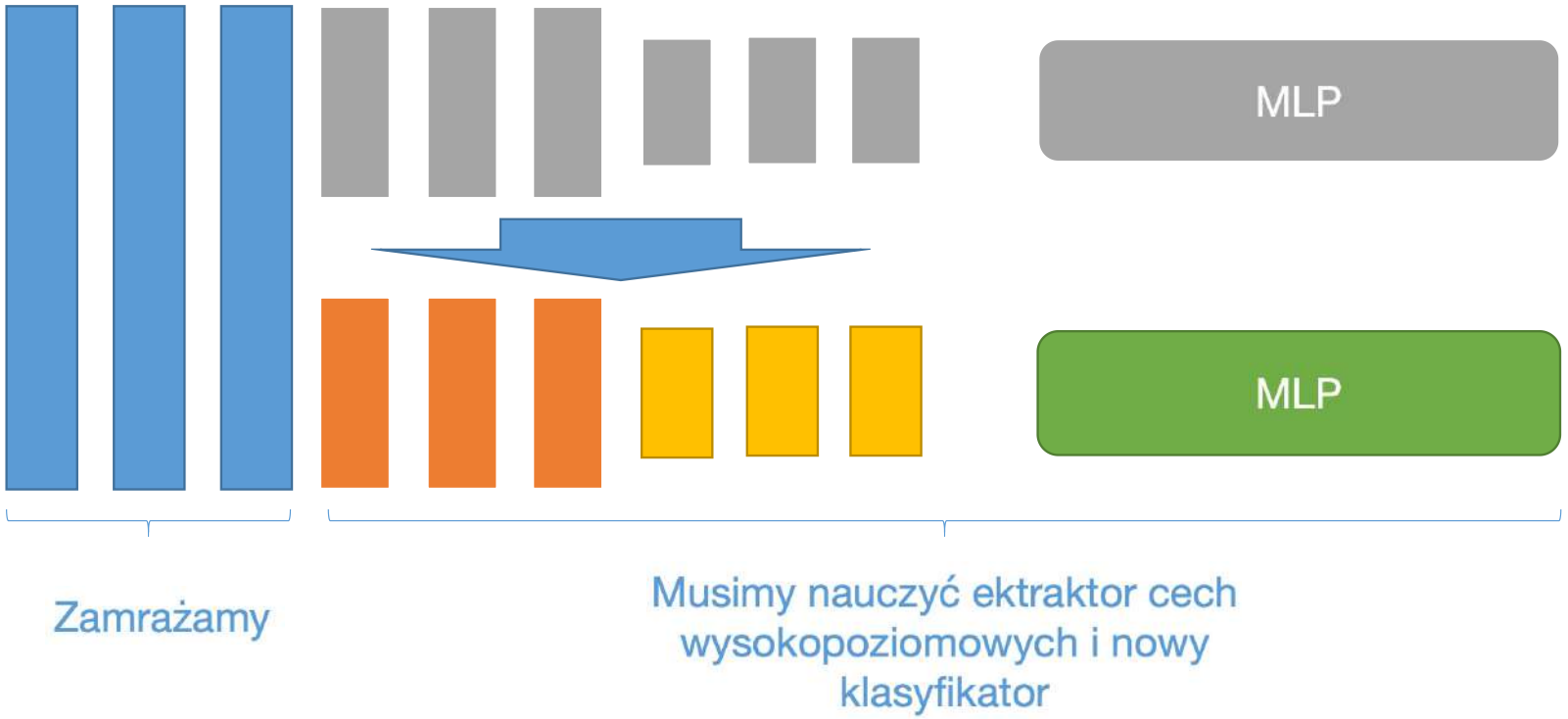
Będzie działać dobrze, jeżeli nowe zadanie jest podobne do zadania pierwotnego

Uczenie transferowe

Uczenie transferowe polega na wykorzystaniu istniejącej sieci do realizacji nowych zadań

Uczenie transferowe

Możemy zainicjalizować wagi istnijącymi wartościami



Analiza danych tekstowych - podstawy

Analiza danych tekstowych

- analiza sentymentu
- przydzielanie treści do kategorii
- rozpoznawanie nazwanych jednostek
- łączenie jednostek,
- tłumaczenie tekstu
- ekstrakcja relacji
- generowanie zapytań SQL
- skracanie tekstu
- ...

Zastosowanie w cyberbezpieczeństwie

Wykrywanie cybernękania
i zachowań seksualnych

Systemy wczesnego wykrywania
samobójstw i prób samookaleczeń

Wykrywanie obraźliwych treści

Wykrywanie nielegalnych działań

systemy interwencyjne do
wykrywania radykalizacji
i kryminalizacji nastolatków

narzędzia profilowania do
wykrywania oszustw tożsamości,
„catfishing”, ...

Przetwarzanie wstępne tekstu (preprocessing)

Wyrażenia regularne

Formalny język do opisu łańcuchów znaków

Jak opisać takie termy jak np.:

- Dom
- dom
- domek
- domeczek
- 12.12.2020
- 12-12-2020

Wyrażenia regularne

Litery umieszcza się w nawiasach kwadratowych

Wzorzec	Przykładowe dopasowanie
[Dd]om	Dom, dom
[1234567890]	dowolna cyfra

Zakresy

Wzorzec	Przykładowe dopasowanie
[a-z]	dowolna mała litera
[A-Z]	dowolna duża litera
[0-9]	dowolna cyfra

Wyrażenia regularne

Negacje

Wzorzec	Przykładowe dopasowanie
[^A-Z]	każdy znak tylko nie duża litera
[^Dd]	każdy znak tylko nie D (duże i małe)

Uwaga! negacja działa tylko gdy operator ^ występuje na początku wzorca

Wzorzec	Przykładowe dopasowanie
[a^b]	dopasuje się do łańcucha znaków “a^b”
[A-Z]	dowolna duża litera
[0-9]	dowolna cyfra

Wyrażenia regularne

Alternatywy - operator |

Wzorzec	Przykładowe dopasowanie
Dom|Mieszkanie	wyrazy: Dom, Mieszkanie
[Dd]om|[Mm]ieszkanie	wyrazy: Dom, dom, Mieszkanie, mieszkanie

Operatory: * ? + .

Wzorzec	Przykładowe dopasowanie
[0-9]*	0 lub więcej cyfr
[0-9]+	1 lub więcej cyfr
[-]?[0-9]+	dowolna liczba całkowita z opcjonalnym znakiem
.*	dowolny łańcuch znaków

Wyrażenia regularne

Kotwice: ^ $

Wzorzec	Przykładowe dopasowanie
^[A-Z]	dowolna litera na początku łańcucha znaków
^[^a-zA-Z]	dowolny znak (poza literą) na początku łańcucha znaków
\.$	kropka na końcu łańcucha znaków
.$	dowolny znak na końcu łańcucha znaków

Tokenizacja

Tokenizacja polega na zamianie tekstu na poszczególne jednostki (tokeny)

"Cyberbezpieczeństwo: źródła zagrożeń i konsekwencje narażenia bezpieczeństwa danych"

Wraz z rozwojem Internetu pojawiła się konieczność dbania o bezpieczeństwo tele-informatyczne. Celem ataku może bowiem stać się każda firma i organizacja.

Szacuje się, że na całym świecie straty poniesione wskutek przestępstw teleinformatycznych osiągną w roku 2019 kwotę 2 trylionów dolarów amerykańskich. Zdaniem dyrektor generalnej giganta komputerowego IBM, Ginni Rometty, przestępstwa tego typu stanowią zagrożenie nr 1 dla wszystkich istniejących firm. Niemniej jednak pomimo organizowanych międzynarodowych kampanii zwiększających świadomość tego typu zagrożeń, wiele organizacji nadal posiada niedostateczną wiedzę na ten temat i nie stosuje odpowiednich zabezpieczeń.

https://www.regus.pl/work-poland/cybersecurity-threats-where-do-they-come-from-and-whats-at-risk/

Tokenizacja

{“Wraz”, “z”, “rozwojem”, “Internetu”, “pojawiła”, “się”, “konieczność”, “dbania”, “o”, “bezpieczeństwo”, “tele”, “-”, “informatyczne”}

{„Wraz z”, „z rozwojem”, „rozwojem Internetu”, „Internetu pojawiła”, „pojawiła się”, „się konieczność”, „konieczność dbania”, „dbania o”, „o bezpieczeństwo”, ...}

{„Wraz z rozwojem”, „z rozwojem Internetu”, „rozwojem Internetu pojawiła”, „Internetu pojawiła się”, „pojawiła się konieczność”, ...}

Stemming

Stemming - doprowadzenie połączonych lub podobnych wariantów słowa do wersji bazowej

Zagadnienie stosunkowo „proste” dla języka angielskiego, zdecydowanie „trudne” dla języka polskiego

Stemming - przykład

Weźmy pod uwagę różne formy wyrazu “dbać”

dbać, dba, dbacie, dbaj, dbają, dbając, dbająca, dbającą, dbające, dbającego, dbającej, dbającemu, dbający, dbających, dbającym, dbającymi, dbajcie, dbajcież, dbajmy, dbajmyż, dbajże, dbali, dbaliby, dbalibyście, dbalibyśmy, dbaliście, dbaliśmy, dbał, dbała, dbałaby, dbałabym, dbałabyś, dbałam, dbałaś, dbałby, dbałbym, dbałbyś, dbałem, dbałeś, dbało, dbałoby, dbały, dbałyby, dbałybyście, dbałybyśmy, dbałyście, dbałyśmy, dbam, dbamy, dbania, dbaniach, dbaniami, dbanie, dbaniem, dbaniom, dbaniu, dbano, dbań, dbasz, niedbająca, niedbającą, niedbające, niedbającego, niedbającej, niedbającemu, niedbający, niedbających, niedbającym, niedbającymi, niedbania, niedbaniach, niedbaniami, niedbanie, niedbaniem, niedbaniom, niedbaniu, niedbań

Lista słów z odmianami: https://sjp.pl/slownik/odmiany/

Usuwanie stop words

W językach naturalnych występują słowa, które pojawiają się często, a nie mają dużego znaczenia dla zrozumienia sensu wiadomości.

```
(',', 48),   ('w', 29),
('i', 25),   ('z', 21),
('na', 17),  ('się', 14),
('do', 14),  ('danych', 13),
('lub', 12), ('jak', 10),
('typu', 9), ('przez', 9),
('ze', 9),   ('bezpieczeństwa', 8),
('', 8),     ('może', 8),
('to', 8),   ('firmy', 8),
('(', 8),    (')', 8),
```

the, is, he, a, an, her, ...

który, jaki, z, a, i, w, ...

Kodowanie tokenów

One hot encoding

Bags of words

Words embedding

Słownik

Zbiór termów w dokumencie

Wraz z rozwojem Internetu pojawiła się konieczność dbania o bezpieczeństwo teleinformatyczne.

“Wraz” “z” “rozwojem” “Internetu” “pojawiła” “się” “konieczność” “dbania” “o” “bezpieczeństwo” “teleinformatyczne”

One hot encoding

„bezpieczeństwo” „dbania” „Internetu” „konieczność” „o” „pojawiła” „rozwojem” „się” „teleinformatyczne” „Wraz” „z”

```
[[0,0,0,0,0,0,0,0,0,1,0  #Wraz
 [0,0,0,0,0,0,0,0,0,0,1  #z
 [0,0,0,0,0,0,1,0,0,0,0  #rozwojem
 [0,0,1,0,0,0,0,0,0,0,0  #Internetu
 [0,0,0,0,0,1,0,0,0,0,0  #pojawiła
 [0,0,0,0,0,0,0,1,0,0,0  #się
 [0,0,0,1,0,0,0,0,0,0,0  #konieczność
 [0,1,0,0,0,0,0,0,0,0,0  #dbania
 ...,
]
```

Bags of words

W modelu tym zapisujemy liczbę wystąpień danego słowa w dokumencie

„bezpieczeństwo” „dbania” „Internetu” „konieczność” „o” „pojawiła” „rozwojem” „się” „teleinformatyczne” „Wraz” „z”

```
[[0,1,1,1,0,1,1,1,0,1,1]]
```

Jak określić o czym jest dokument?

Spójrzmy na najczęściej występujące słowa w tym dokumencie

```
('danych', 13),          ('typu', 9),
('bezpieczeństwa', 8),   ('firmy', 8),
('celu', 6),             ('zagrożeń', 5),
('bezpieczeństwo', 4),   ('cyberbezpieczeństwa', 4),
('zewnątrz', 4),         ('dostęp', 4),
('informacji', 4),       ('poufnych', 4),
('względu', 4),          ('oprogramowanie', 4),
('danych.', 4),          ('ataków', 4),
```

Jak określić znaczenie danego słowa dla dokumentu?

Czy liczba wystąpień jest dobrą miarą?

Wyraz “dragon”:

W opsie https://pl.wikipedia.org/wiki/Dragoni: pojawia się 35 razy

W “Potopie” Henryka Sienkiewicza pojawia się 119 razy

Częstotliwość termów

Określa znormalizowaną liczbę wystąpień danego termu
w dokumencie

$$tf_{n,d} = \frac{C(W_n, d)}{\sum_k C(W_k, d)}$$

Wyraz “dragon”:

W opsie https://pl.wikipedia.org/wiki/Dragoni: 35/790 = 0,044

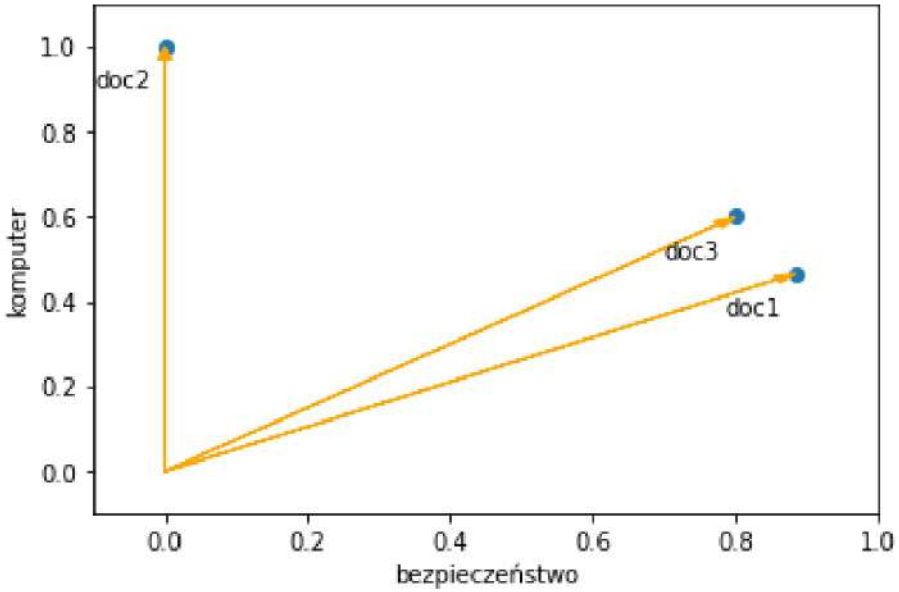
W “Potopie” Henryka Sienkiewicza pojawia się 119 razy = 0,000289

Kolekcja dokumentów

	Słowo 1	Słowo 2	Słowo 3	Słowo 4	Słowo 5	
DOC 1	X	0	X	X	0	…
DOC 2	0	X	X	0	0	…
DOC 3	X	0	0	0	X	…
	…	…	…	…	…	

	dane	bezpie-czeństwo	komputer	truskawka	winogrona	
DOC 1	100 0,46	76 0,35	40 0,18	0	0	…
DOC 2	10 0,13	0	4 0,05	40 0,54	20 0,27	…
DOC 3	50 0,41	40 0,33	30 0,25	0	0	…
	…	…	…	…	…	

Podobieństwo dokumentów



$$\text{cosine}(\mathbf{v}, \mathbf{w}) = \frac{\mathbf{v} \cdot \mathbf{w}}{|\mathbf{v}||\mathbf{w}|} = \frac{\sum_{i=1}^{N} v_i w_i}{\sqrt{\sum_{i=1}^{N} v_i^2} \sqrt{\sum_{i=1}^{N} w_i^2}}$$

Odwrotna częstotliwość dokumentu

Częstotliwość danego słowa w dokumencie nie mówi nam nic na temat znaczenia tego słowa w stosunku do innych dokumentów w korpusie

$$IDF(t, D) = \frac{|D|}{|D_t|}$$

Odwrotna częstotliwość dokumentu określa ważność termu w korpusie.
Im mniej razy dany term występuje w korpusie, tym jest bardziej istotny.

Odwrotna częstotliwość dokumentu

Rozmiar korpusu	1000000	IDF1	IDF
Phishing	300	3333	3,52
Stalking	100	10000	4

$$IDF(t, D) = \log_{10}\left(\frac{|D|}{|D_t|}\right)$$

Kolekcja dokumentów

	dane	bezpie-czeństwo	komputer	truskawka	winogrona
DOC 1	100 0,46	76 0,35	40 0,18	0	0
DOC 2	10 0,13	0	4 0,05	40 0,54	20 0,27
DOC 3	50 0,41	40 0,33	30 0,25	0	0
	…	…	…	…	…