

Uczenie maszynowe i deep learning.

By Mateusz Macias

ML a programowanie

- Programowanie:

- Umiemy zdefiniować procedurę prowadzącą do celu.
- Programujemy podając ciąg instrukcji.
- Umiemy zdefiniować czy nam się udało czy nie.

- Uczenie maszynowe:

- Nie umiemy zdefiniować procedury prowadzącej do celu.
- Czasem nie umiemy zdefiniować celu.
- Ale wszystko jest w danych.
- Tworzymy algorytm który na bazie danych stworzy algorytm który rozwiąże zadanie.

Przykłady.

- Samojeżdżące samochody.
- Systemy rekomendacyjne (Netflix, Youtube etc.)
- Przewidywanie cen nieruchomości.
- Diagnozowanie raka.
- Przewidywanie pogody.
- Projektowanie leków.
- Granie w gry (atari, go).
- Rozpoznawanie mowy.
- Przewidywanie zachowań klientów.
- Debluring.
- High Frequency Trading
- Generowanie zdjęć śmiesznych kotów.

Supervised learning



Mamy:
Zbiór próbek
(X, Y)

Chcemy:
Program który na bazie
 X przewidzi Y

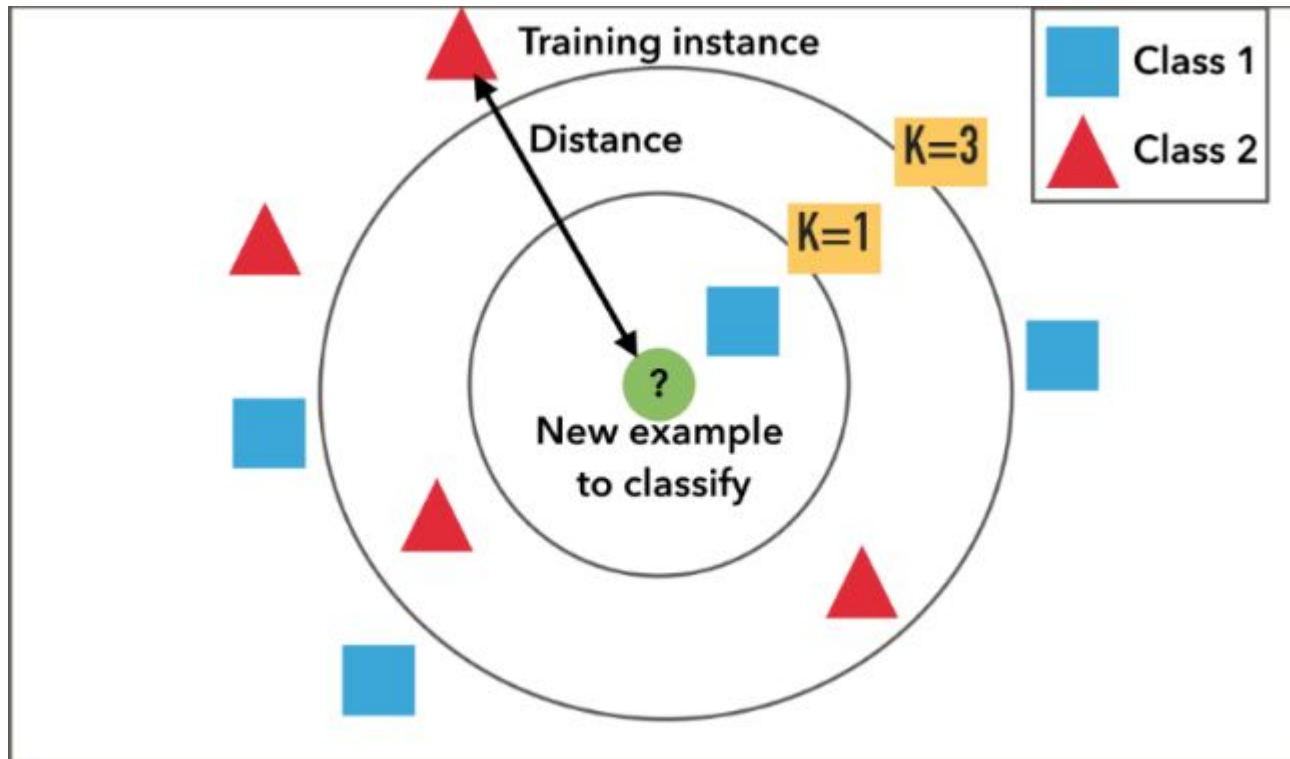
Przykłady.

| X | Y |
|--|--|
| Informacje o mieszkaniu: rozmiar, dzielnica, rynek wtórny pierwotny, rok budowy itp. | Za ile sprzedane? |
| Zdjęcia (kotów i psów) | Czy na zdjęciu jest kot czy pies? |
| Zdjęcie mammograficzne. | Czy jest rak piersi? |
| Historia kursu akcji do teraz. | Kurs akcji za minutę? |
| Obraz z kamerki na samochodzie. | O ile stopni przechylić kierownicę aby utrzymać pas? |
| Dane o kliencie (wiek, zawód, zarobki, historia transakcji itp.). | Czy kupi gdy do niego zadzwonimy? |
| Nagranie mowy. | Transkrypcja na tekst. |

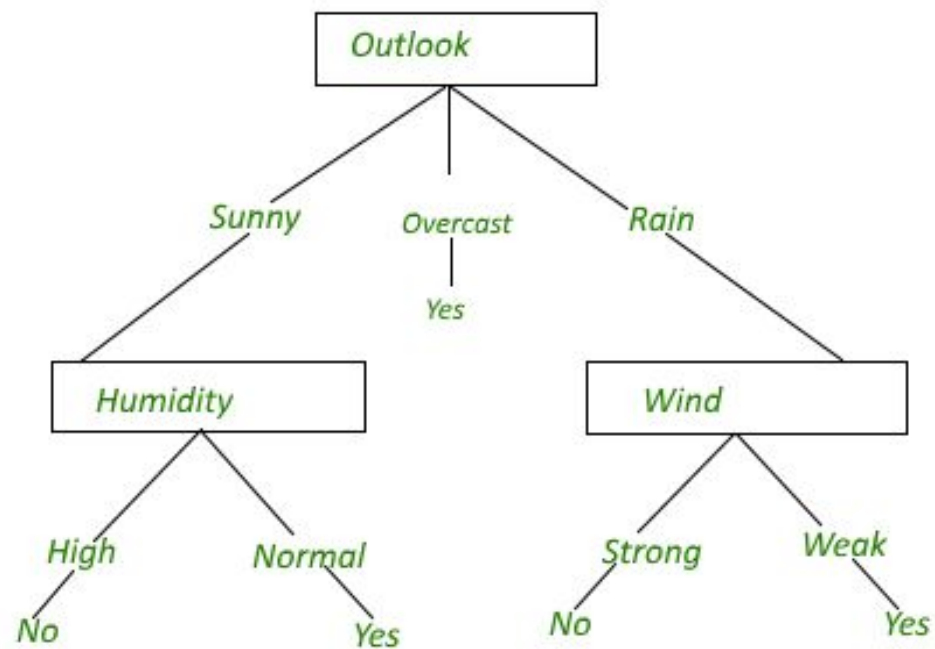
Konfundujące słowo: model

- Model – to co bierze X i wypływa y (funkcja mapująca x na y).
- Model – pewien zbiór funkcji mapujących x na y (zbiór modeli w rozumieniu definicji 1).
- Model – pewien parametryzowalny zbiór funkcji mapujących x na y (zbiór modeli w rozumieniu definicji 1).
- Model - pewien zbiór funkcji mapujących x na y (zbiór modeli w rozumieniu definicji 1) wraz z algorytmem wyboru najlepszego z nich (z punktu wyboru naszych danych).

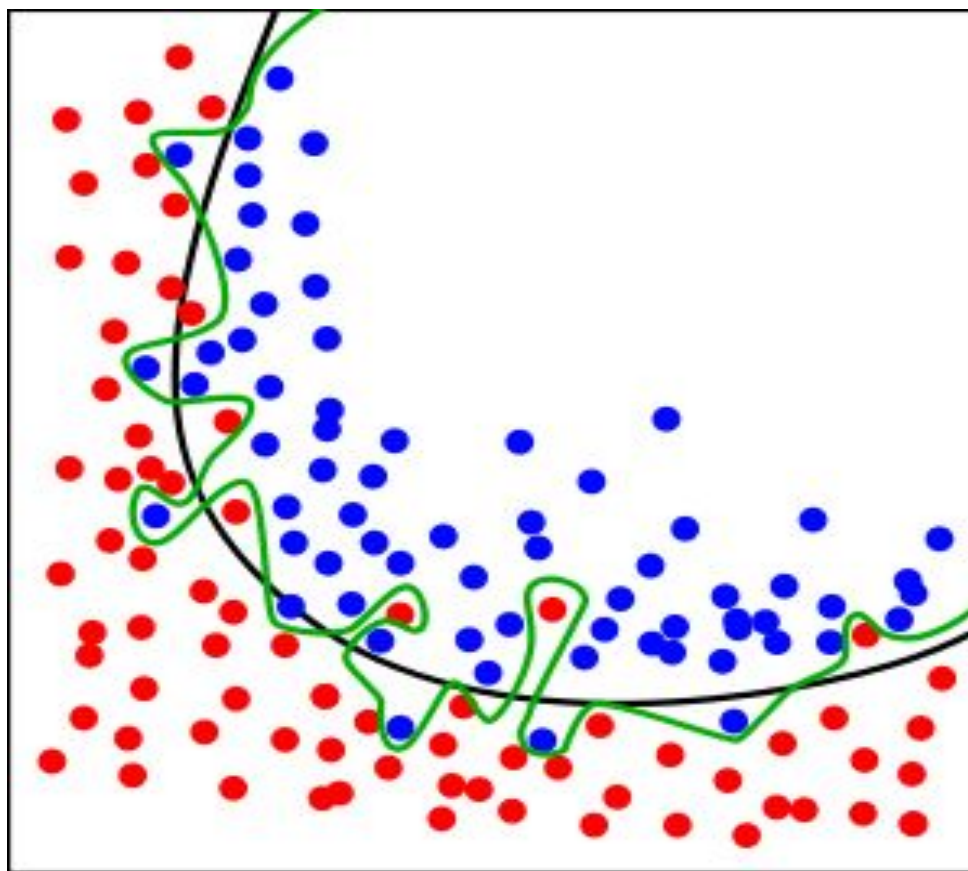
Proste Modele: KNN



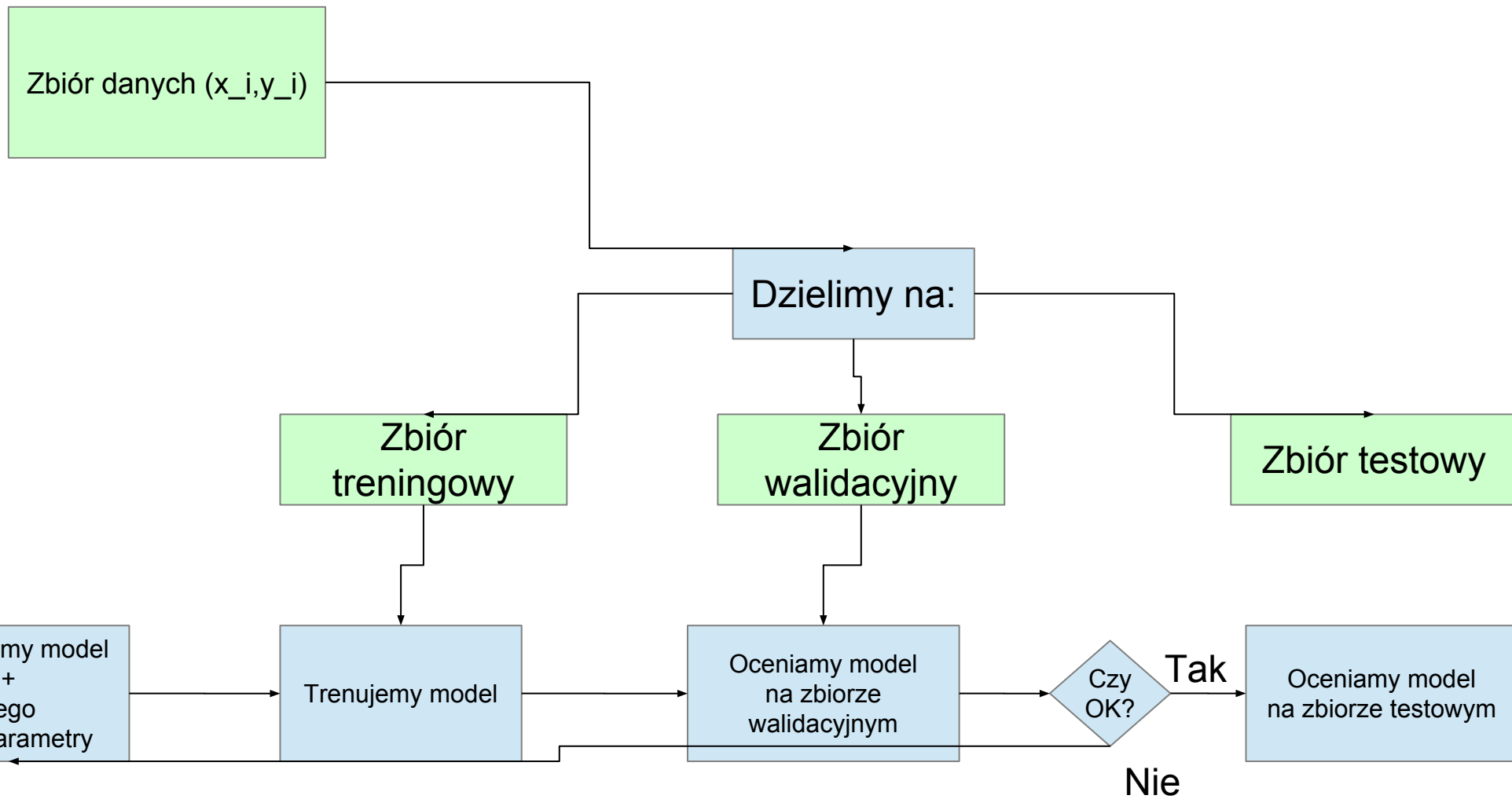
Decision Tree for *PlayTennis*



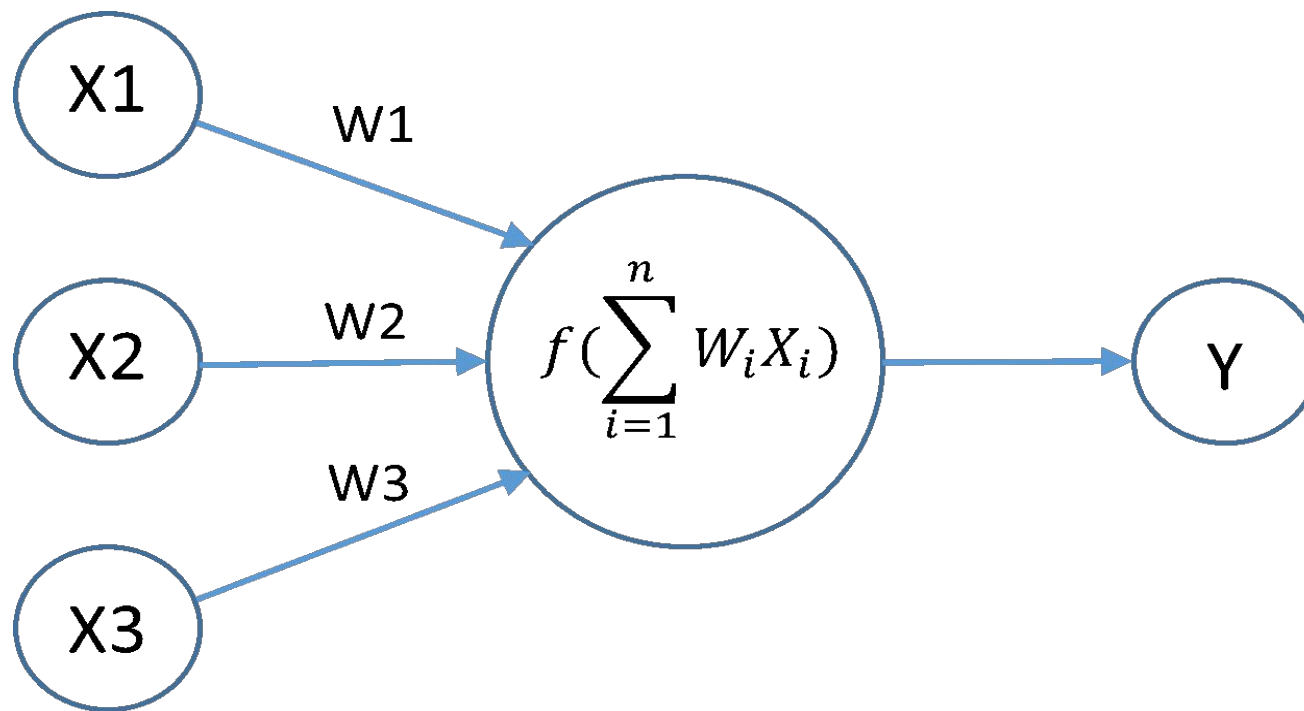
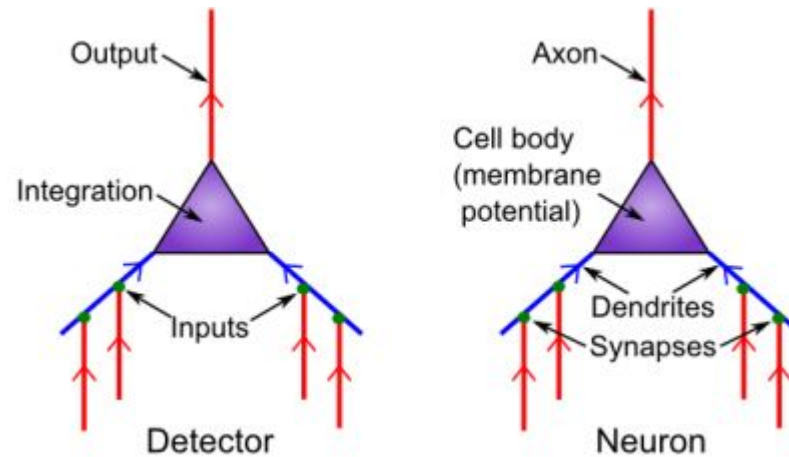
Overfitting



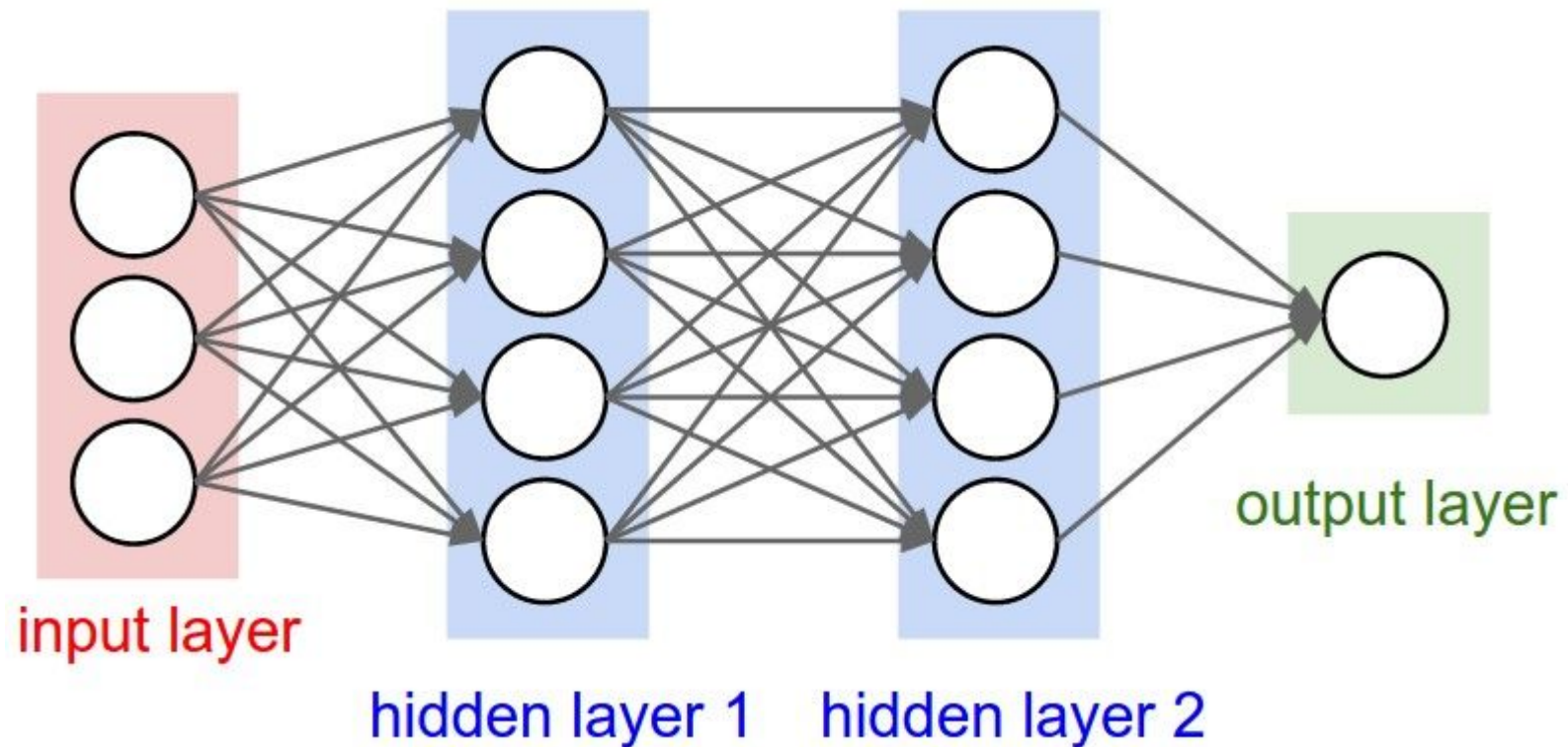
Jak to robimy – czyli o metodologii i modelach



Neuron



Sieć neuronowa

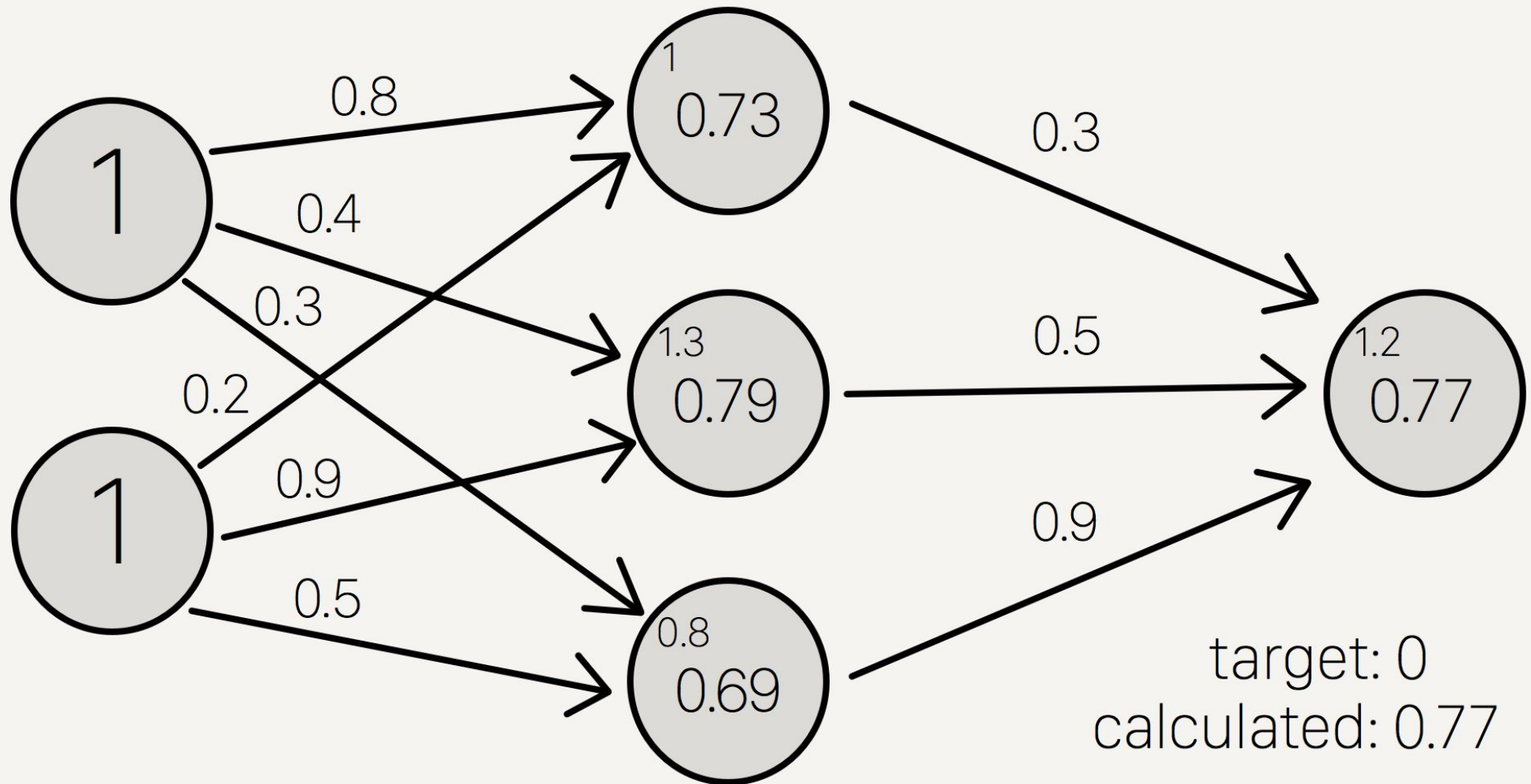


$$Y = f(W1 * f(W2 * f(W3 * X)))$$

INPUT

HIDDEN

OUTPUT



$$f = 1/(1+e^{(-x)})$$

Proces uczenia

Znaleźć takie wagi aby sieć dobrze przewidywała Y na podstawie X .

Musimy mierzyć jak dobrze radzi sobie sieć. Np. tak:

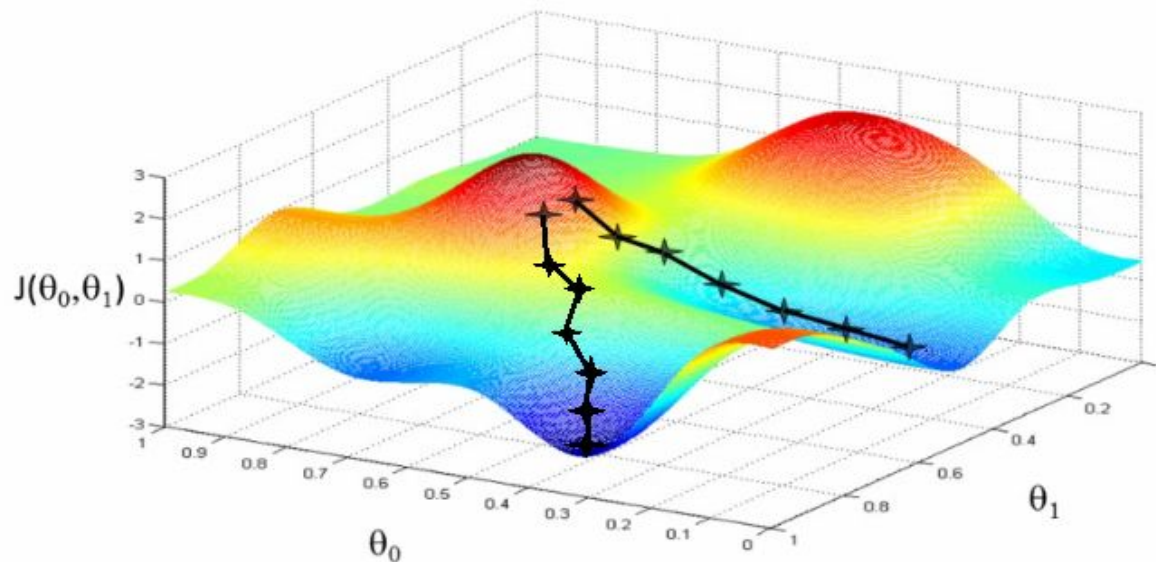
$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \tilde{y}_i)^2$$

Proces uczenia cz2

Repeat until convergence {

$$\theta_j \leftarrow \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

}



Obrazy

<https://www.geeksforgeeks.org/decision-tree/>

/

https://grey.colorado.edu/mediawiki/sites/CompCogNeuro/images/thumb/5/53/fig_neuron_as_detector.png/400px-fig_neuron_as_detect.png

<https://s3-ap-south-1.amazonaws.com/av-blog-media/wp-content/uploads/2017/03/06100746/grad.png>

http://www.conowego.pl/uploads/pics/Miniaturka_01.jp

[g
https://i.stack.imgur.com/gzrsx.png](https://i.stack.imgur.com/gzrsx.png)

[g
https://i.stack.imgur.com/19Cmk.gif](https://i.stack.imgur.com/19Cmk.gif)

[f
https://www.codeproject.com/KB/recipes/879043/GradientDescent.jpg](https://www.codeproject.com/KB/recipes/879043/GradientDescent.jpg)

[g
https://upload.wikimedia.org/wikipedia/commons/thumb/1/19/Overfitting.svg/300px-Overfitting.svg.png](https://upload.wikimedia.org/wikipedia/commons/thumb/1/19/Overfitting.svg/300px-Overfitting.svg.png)

https://upload.wikimedia.org/wikipedia/commons/thumb/6/60/ArtificialNeuronModel_english.png/600px-ArtificialNeuronModel_english.png

<https://medium.com/@adi.bronshtein/a-quick-introduction-to-k-nearest-neighbors-algorithm-62214cea29c7>