Wprowadzenie do uczenia maszynowego

Agnieszka Ławrynowicz

2019

Wprowadzenie: co to jest uczenia maszynowe?

Co to jest uczenie maszynowe?

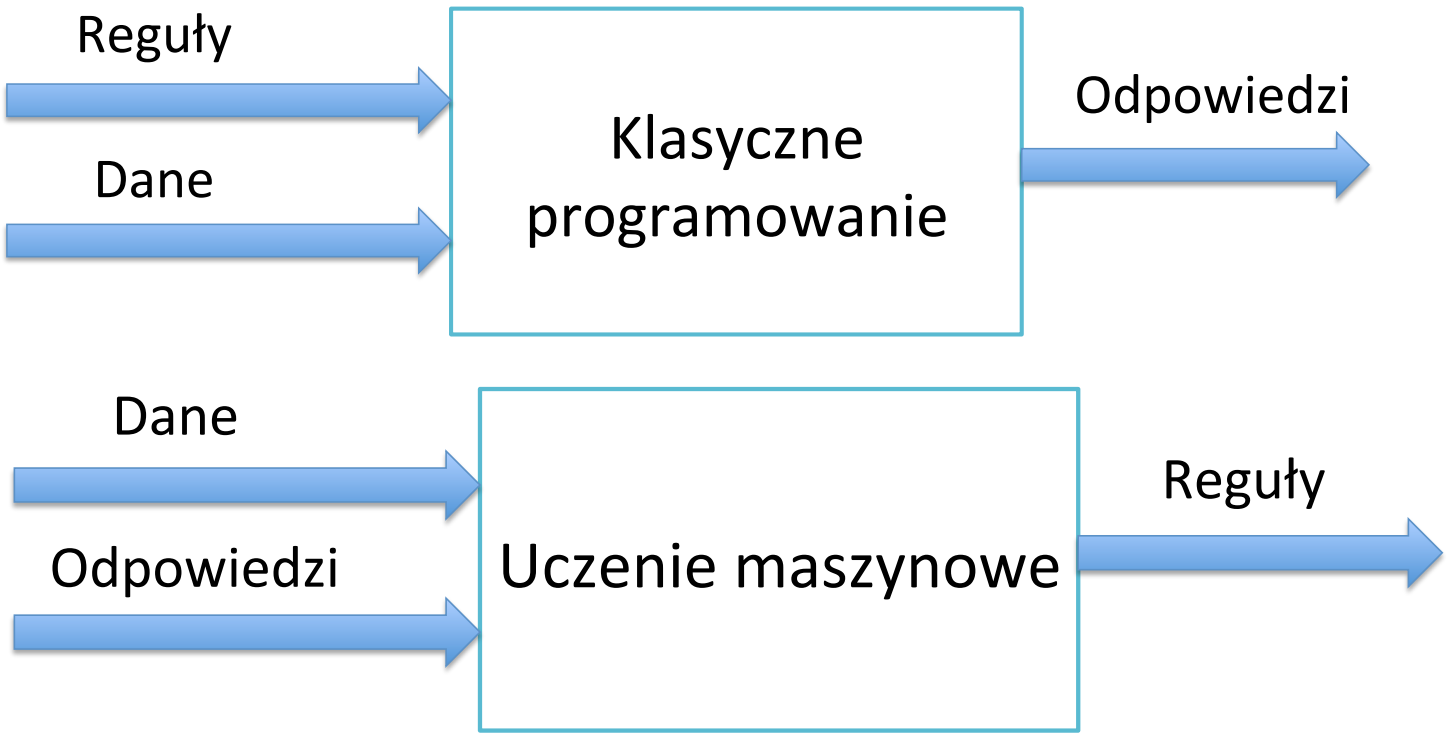
Co to dokładnie znaczy, że maszyna *nauczyła się* czegoś?
Czy jeśli pobiorę kopię Wikipedii to mój komputer "nauczył" się zawartej tam wiedzy?
Czym różni się uczenie maszynowe od sztucznej inteligencji?
Czy uczenie maszynowe i *data science* (nauka o danych) to jedno i to samo?

Co to jest uczenie maszynowe?

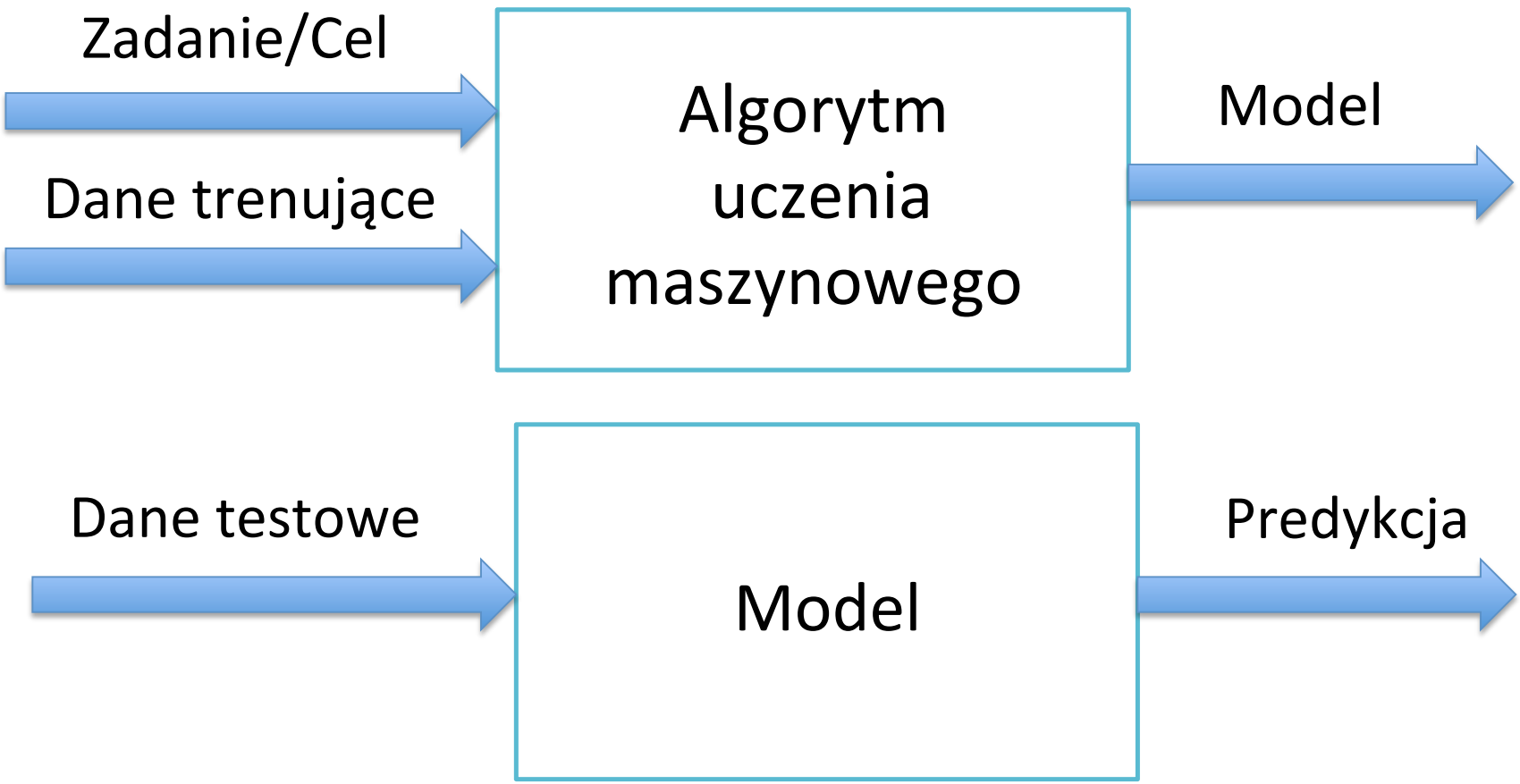
"dziedzina nauki, która zajmuje się sprawianiem aby komputery mogły uczyć się bez ich zaprogramowania wprost" (Arthur Samuel)

"program komputerowy uczy się na podstawie doświadczenia (E) w odniesieniu do pewnej klasy zadań (T) i miary efektywności (P), jeśli jego efektywność wykonywania zadania T (mierzona za pomocą P) poprawia się wraz z doświadczeniem E" (Tom Mitchell)

Nowy paradygmat programowania

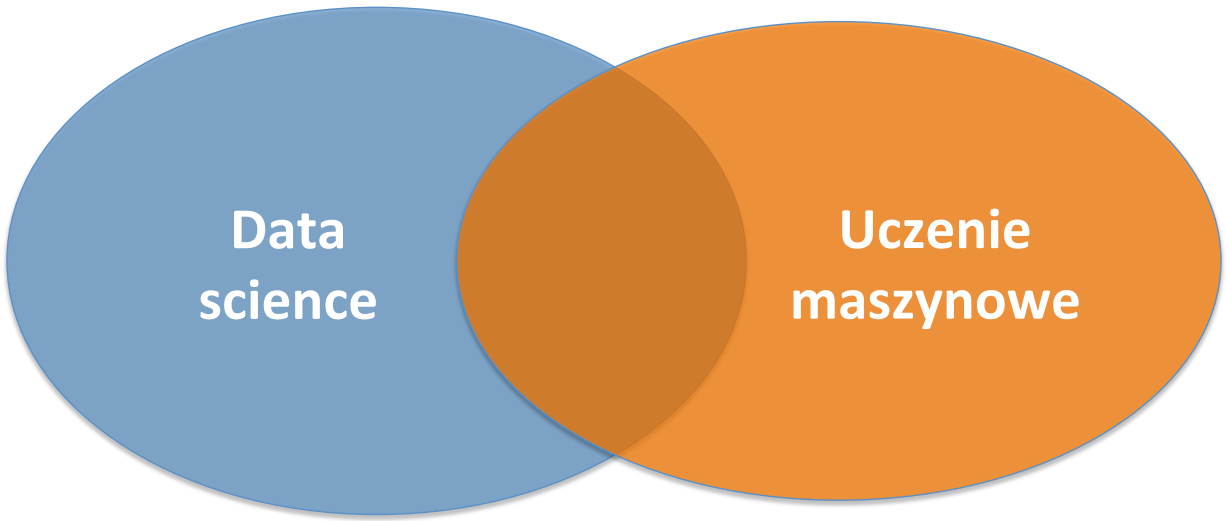


Uproszczony schemat uczenia maszynowego



Sztuczna inteligencja a uczenie maszynowe

Uczenie maszynowe a *data science* (nauka o danych)



Typowe zadania: uczenie nadzorowane i uczenie nienadzorowane

Uczenie nadzorowane

Każdy przykład ze zbioru trenującego składa się ze:

- zbioru cech/atrybutów wejściowych (typowo wektora $\vec{x}$),
- wartości wyjściowej $f(\vec{x})$, tzw. atrybutu decyzyjnego.

Mając na wejściu poprawnie zaetykietowany (wartościami wyjściowymi) zbiór przykładów o postaci $(\vec{x}, f(\vec{x}))$, system uczy się modelu (funkcji h), która ma jak najlepiej aproksymować funkcję f w celu poprawnej predykcji etykiet nowych (nie widzianych wcześniej) przykładów.

Uczenie nadzorowane: regresja i klasyfikacja

Najpopularniejsze zadania uczenia nadzorowanego to regresja i klasyfikacja. Ich celem jest predykcja wartości atrybutu decyzyjnego na podstawie wartości pozostałych atrybutów.

Regresja: atrybut decyzyjny jest liczbą rzeczywistą.

Klasyfikacja: atrybut decyzyjny ma wartość dyskretną (binarną, nominalną). Jest to tzw. klasa.

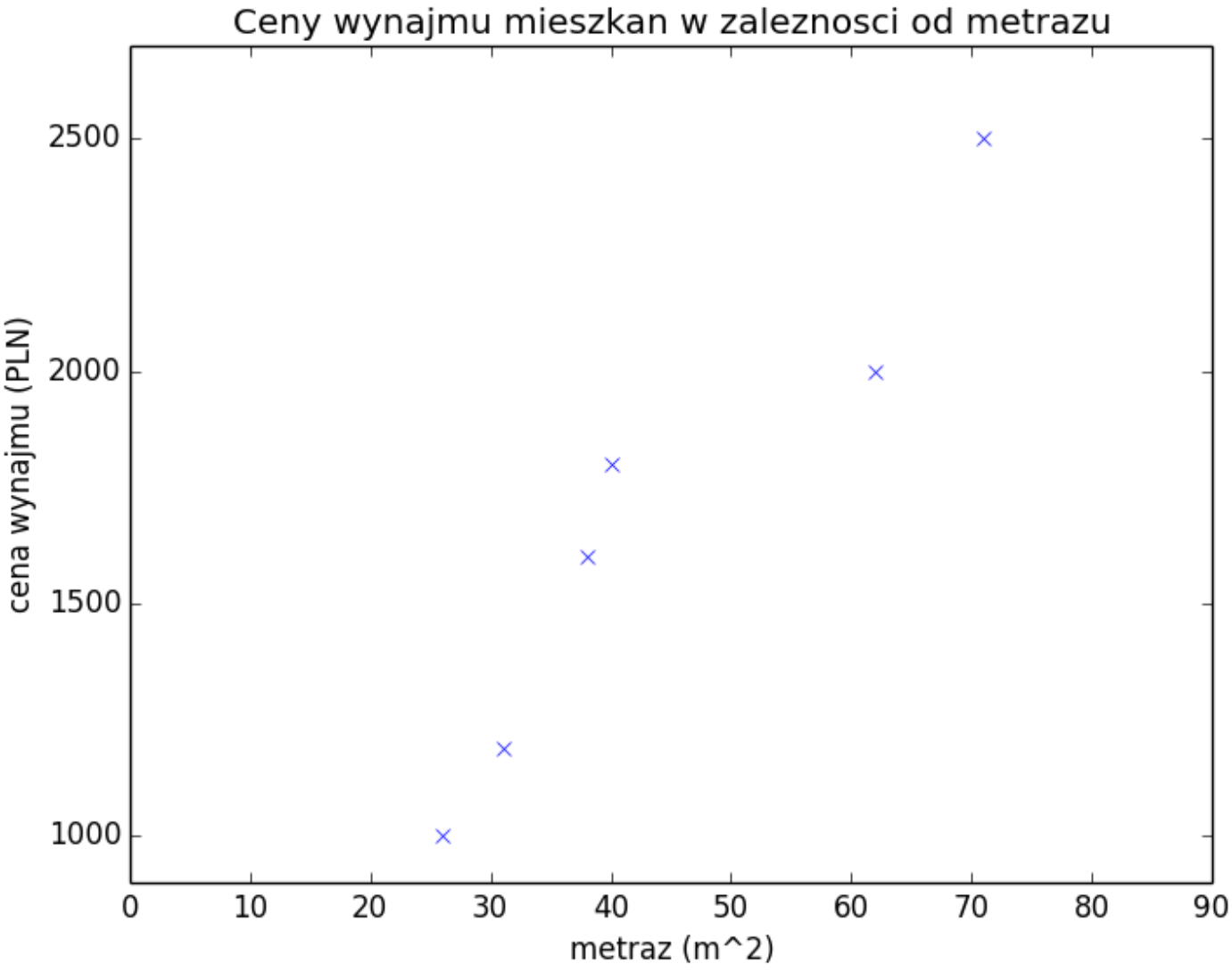
Przykład

Example

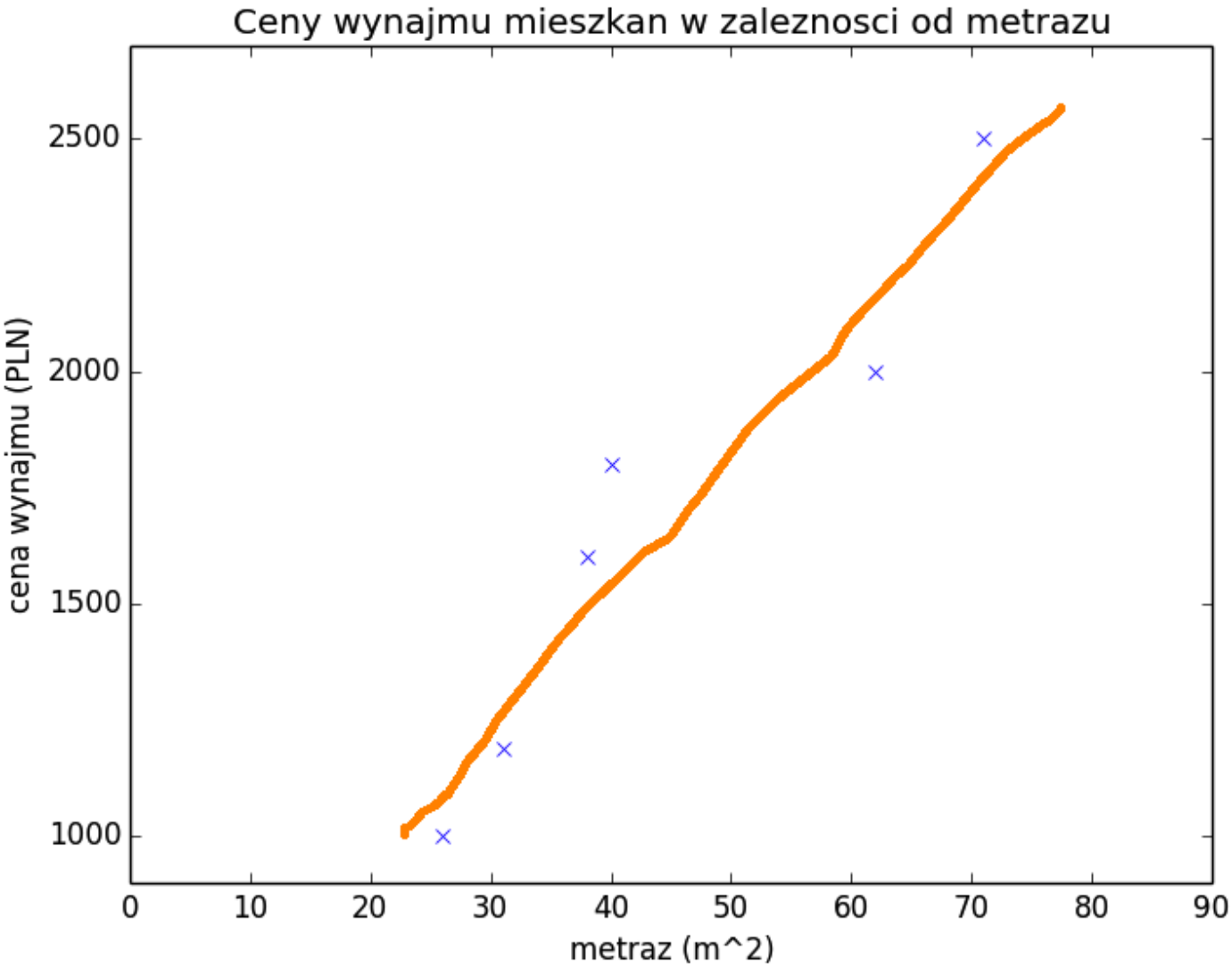
Rodzeństwo, trojaczki Ania, Michał i Paweł od nowego roku akademickiego rozpoczynają studia w Poznaniu. Szukają studenckiego mieszkania i muszą uwzględnić koszt wynajmu w swoim budżecie.
Doszli do wniosku, że kluczową cechą jaka wpływa na koszt wynajmu jest metraż mieszkania.
Przyjrzeli się ofertom wynajmu zamieszczonym na portalu internetowym i zanotowali sobie ich dane w tabeli:

Metraż (m^2)	Cena (PLN)
71	2 500
38	1 600
31	1 190
40	1 800
26	1 000
62	2 000

Przykład c.d.


Ceny wynajmu mieszkan w zaleznosci od metrazu
cena wynajmu (PLN)
2500
2000
1500
1000
0
10
20
30
40
50
60
70
80
90
metraz (m^2)

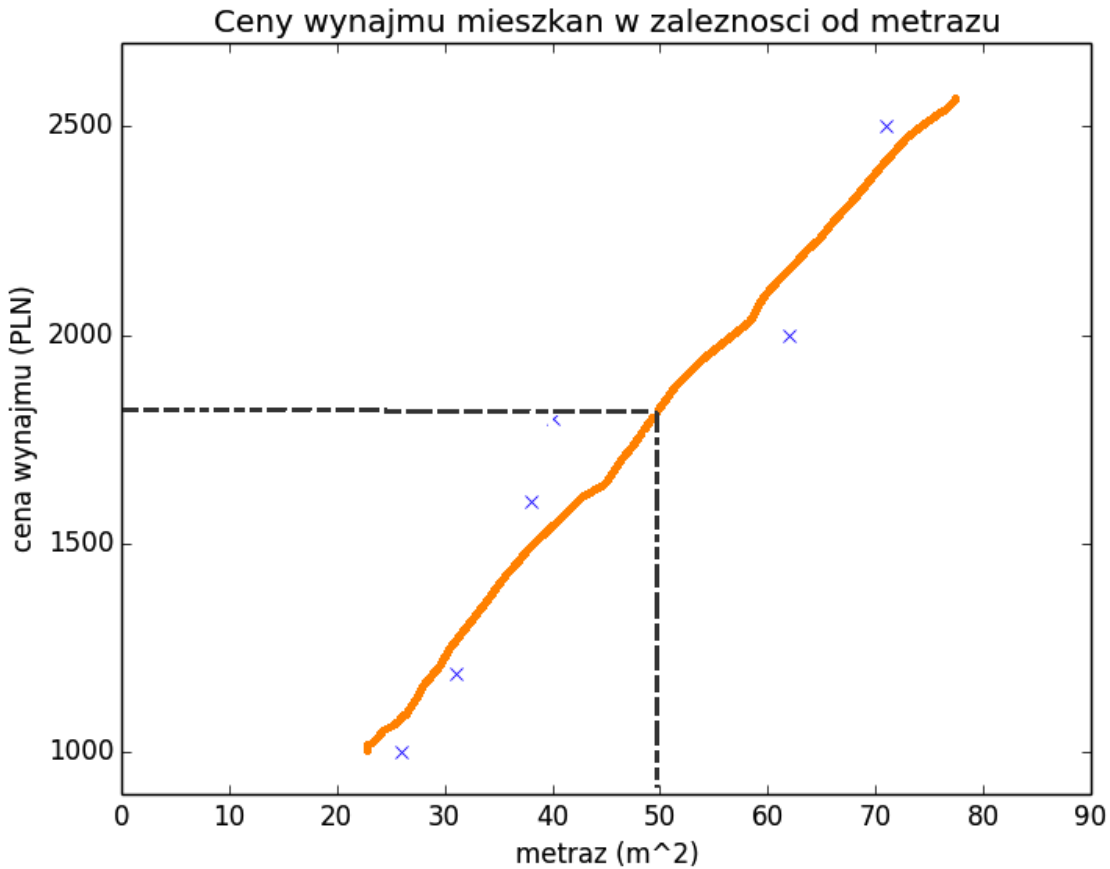
Przykład c.d.


Ceny wynajmu mieszkan w zaleznosci od metrazu
cena wynajmu (PLN)
metraz (m^2)
1000
1500
2000
2500
0
10
20
30
40
50
60
70
80
90

Przykład c.d.

Example

Ania, Michał i Paweł przypuszczają, że odpowiednie dla nich mogłoby być mieszkanie o metrażu około $50m^2$.

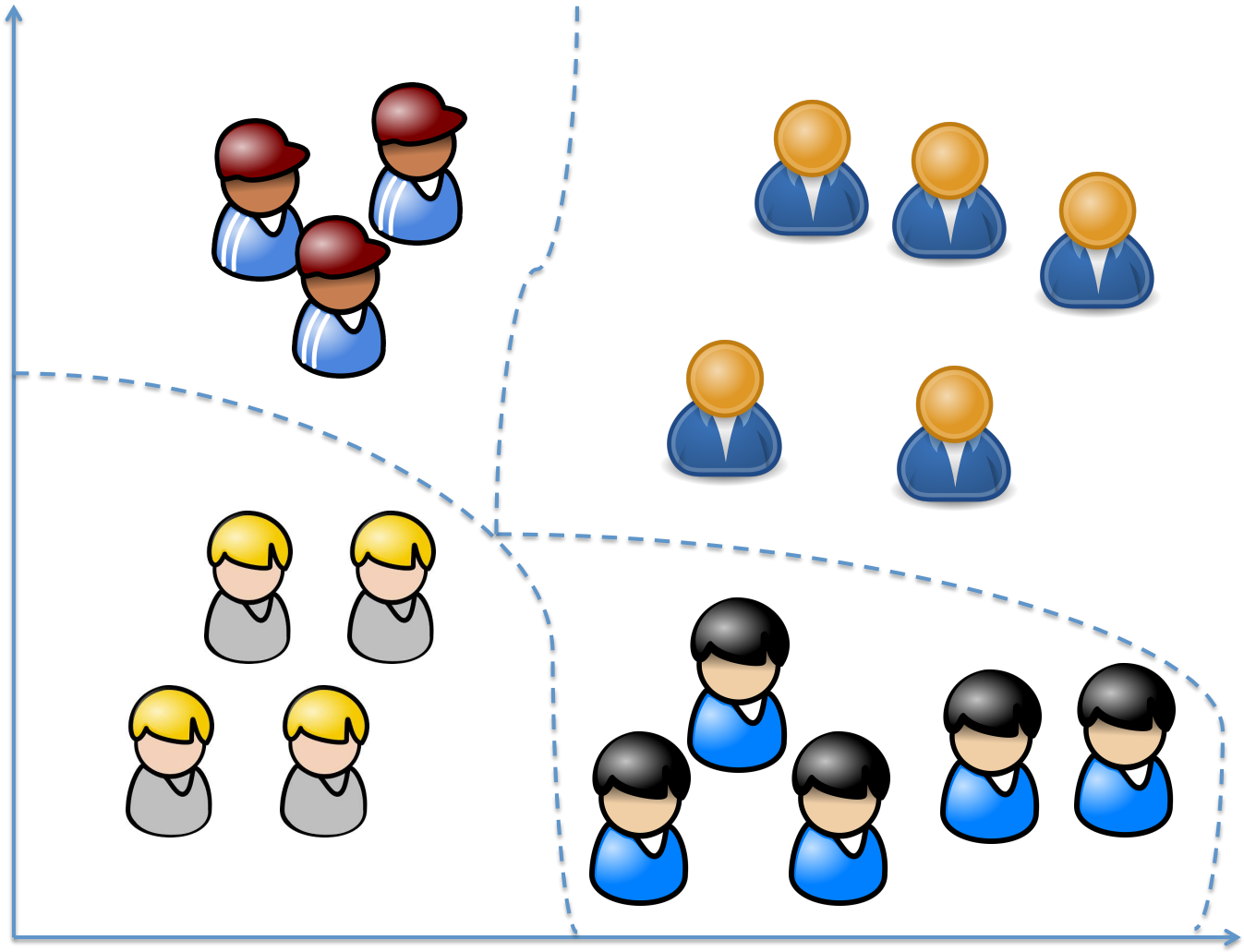


Uczenie nienadzorowane

Każdy przykład ze zbioru trenującego składa się ze zbioru cech wejściowych (typowo wektora $\vec{x}$).

Mając na wejściu zbiór przykładów o postaci ($\vec{x}$), system uczy się modelu (funkcji h), która opisuje dane wejściowe (np. generuje wzorce pojawiające się w danych lub analizuje skupienia danych).

Przykład – segmentacja klientów



Wybrane modele uczenia nadzorowanego i ich trenowanie

Jak reprezentujemy model h?

Regresja liniowa:

$$h(x) = a_1 x + a_0 \text{ (z jedną zmienną)}$$
$$h(\vec{x}) = \vec{a_n}^T \vec{x} + a_0 \text{ (z } n \text{ zmiennych)}$$

$\vec{a_i}$ to parametry modelu a zadanie regresji liniowej polega na ich wyznaczeniu tak aby $h_{\vec{a}}(x_i)$ była jak najbliższa $f(\vec{x_i})$ dla naszych przykładów trenujących $(\vec{x_i}, f(\vec{x_i}))$

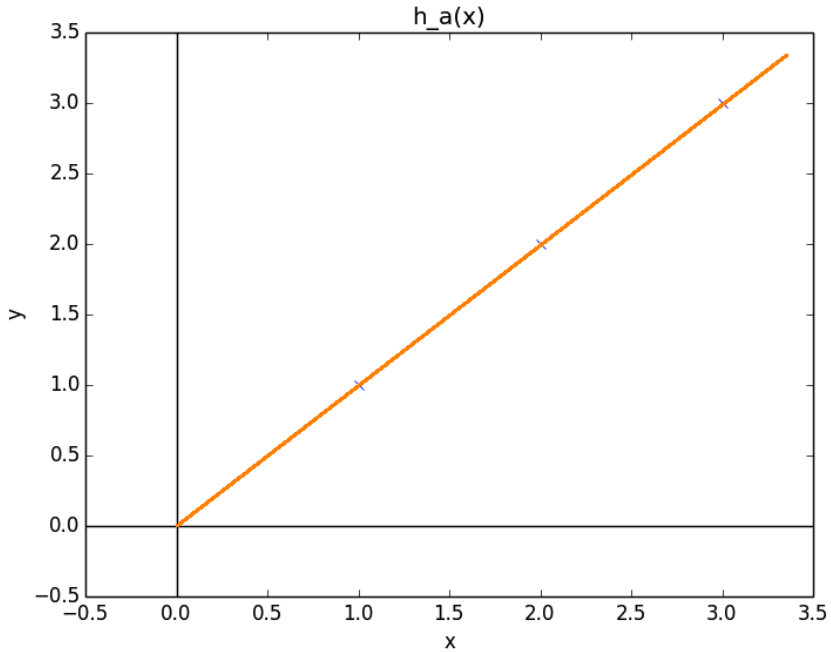
Funkcja kosztu

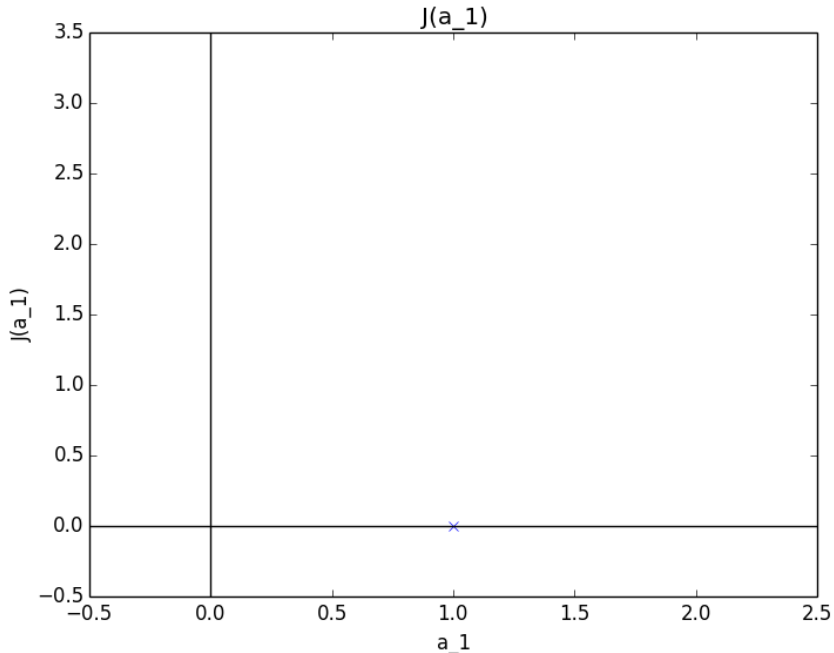
Na ile nasz model się myli?

Błąd średniokwadratowy:

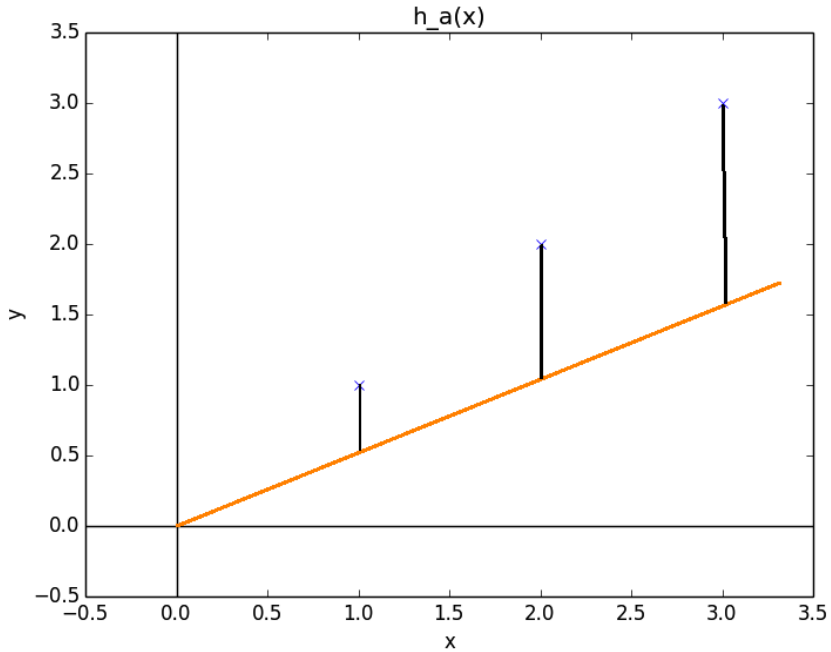
$$J(\vec{a}, a_0) = \frac{1}{m} \sum_{i=1}^{m} (\vec{a}^T x_i + a_0 - y_i)^2$$

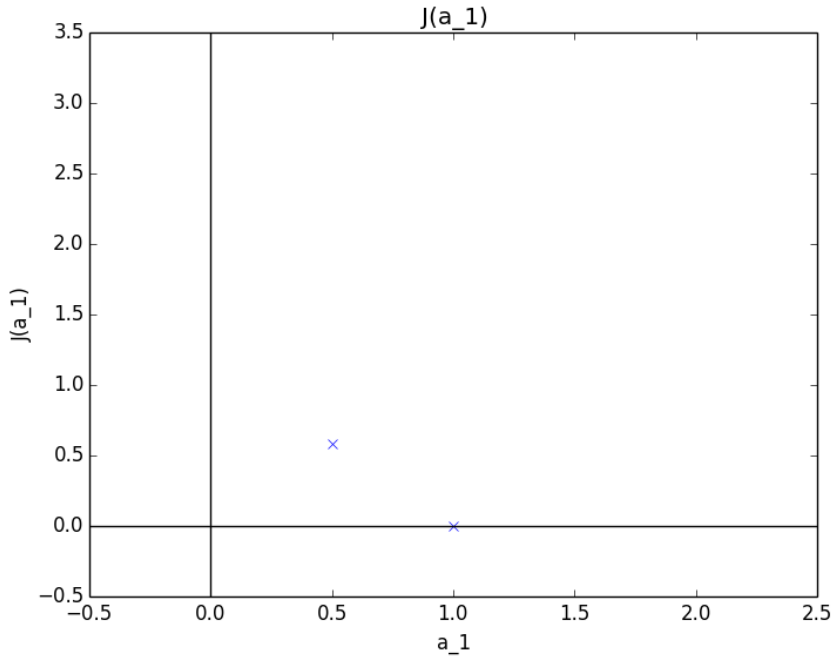
Funkcja kosztu



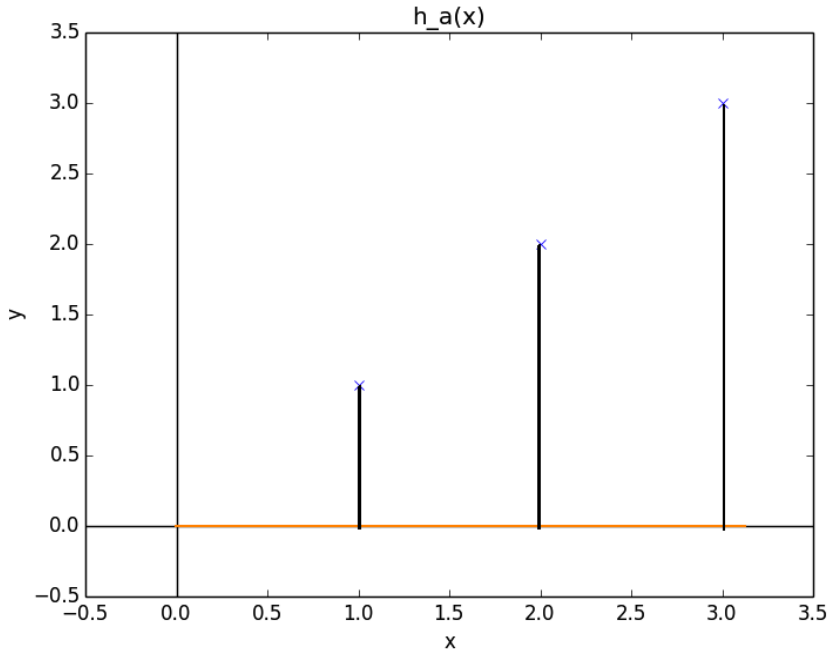


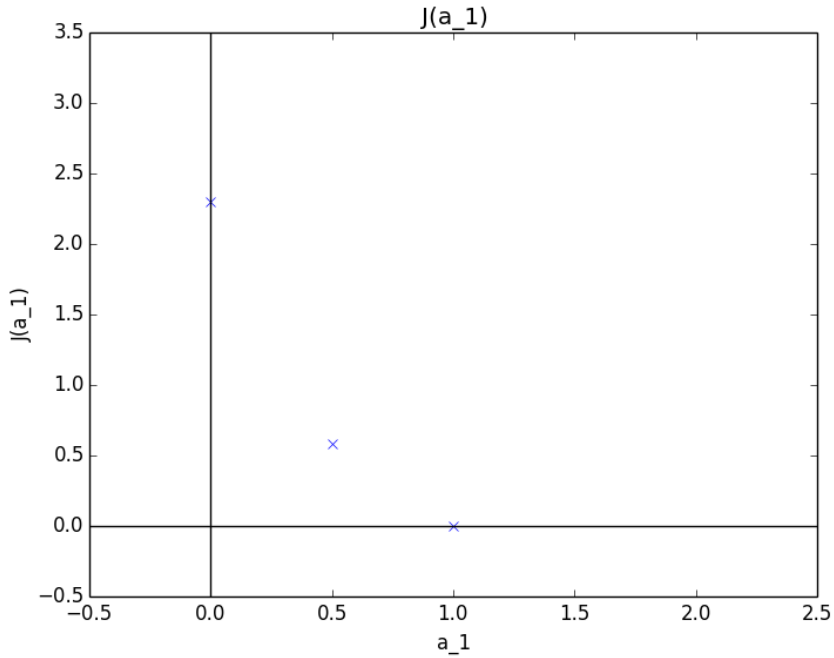
Funkcja kosztu



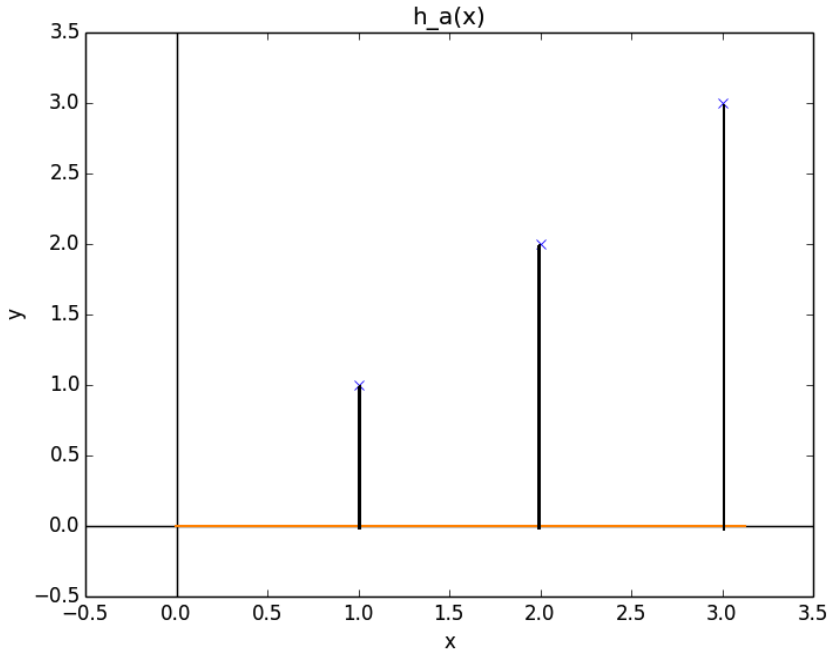


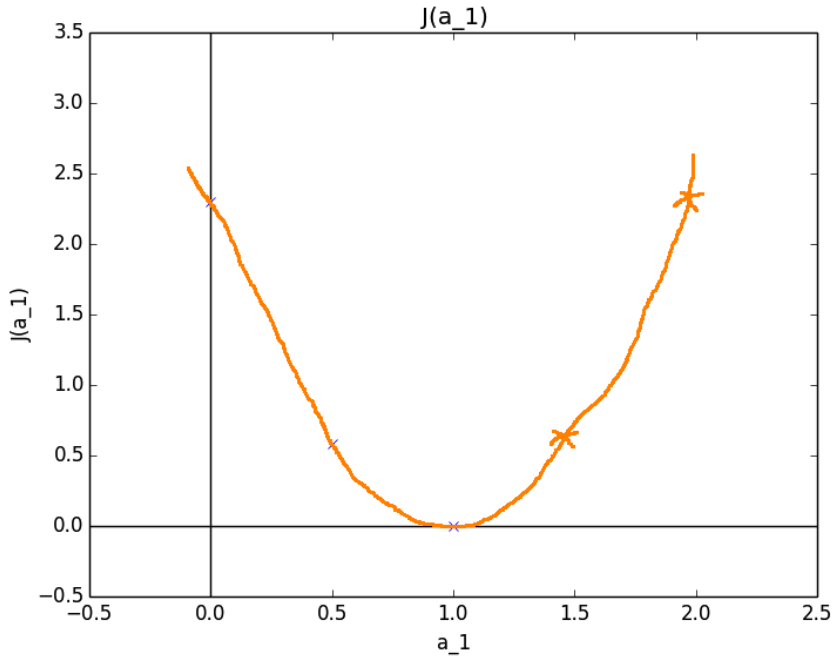
Funkcja kosztu





Funkcja kosztu





Regresja liniowa

Model (hipoteza):

$h(\vec{x}) = \vec{a_n}^T \vec{x} + a_0$

Parametry:

$\vec{a_i}, a_0$

Funkcja kosztu:

$J(\vec{a}, a_0) = \frac{1}{m} \sum_{i=1}^{m} (\vec{a}^T x_i + a_0 - y_i)^2$

Cel:

minimalizacja funkcji kosztu $J(\vec{a}, a_0)$

Algorytm spadku gradientowego

Minimalizacja wybranej funkcji kosztu $J(\vec{a}, a_0)$

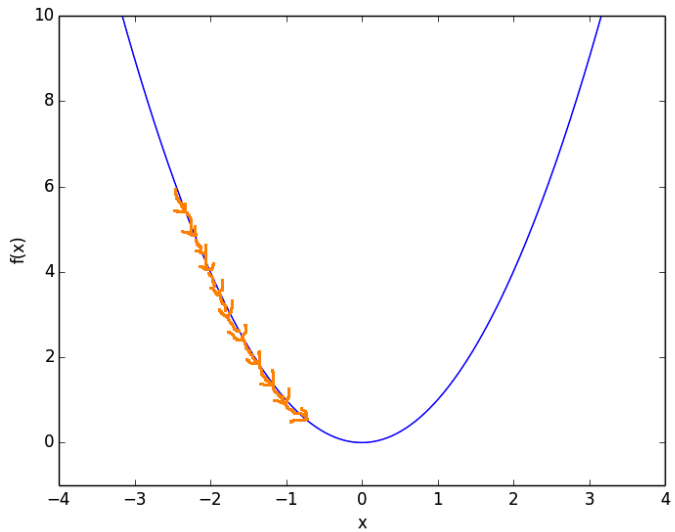
- Przypisz początkowe wagi $\vec{a}, a_0$ (np. $\vec{a}0, a_0 = 0$)
- iteracyjnie podążaj w kierunku minimum funkcji, zmieniając wagi aż osiągniesz minimum

Dla regresji liniowej z jednym atrybutem, wagi aktualizowane są według wzorów (obie naraz w każdym kroku):

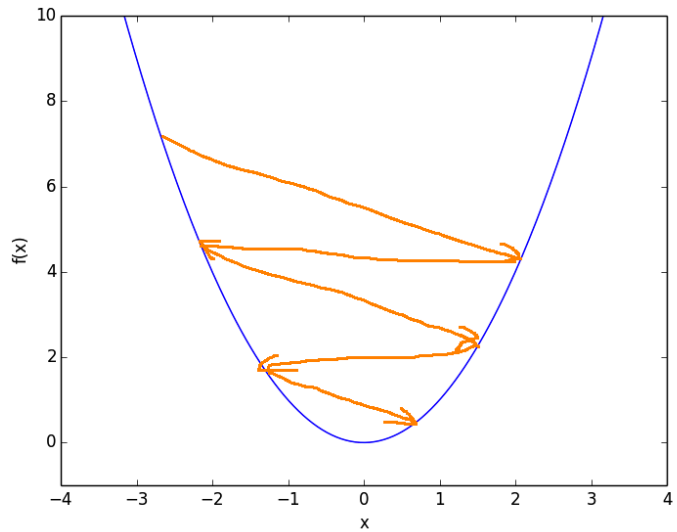
$$a_1 = a_1 - \eta \frac{1}{m} \sum_{i=1}^{m} (a_1 x_i + a_0 - y_i) x_i$$

$$a_0 = a_0 - \eta \frac{1}{m} \sum_{i=1}^{m} (a_1 x_i + a_0 - y_i)$$

η - parametr szybkości uczenia

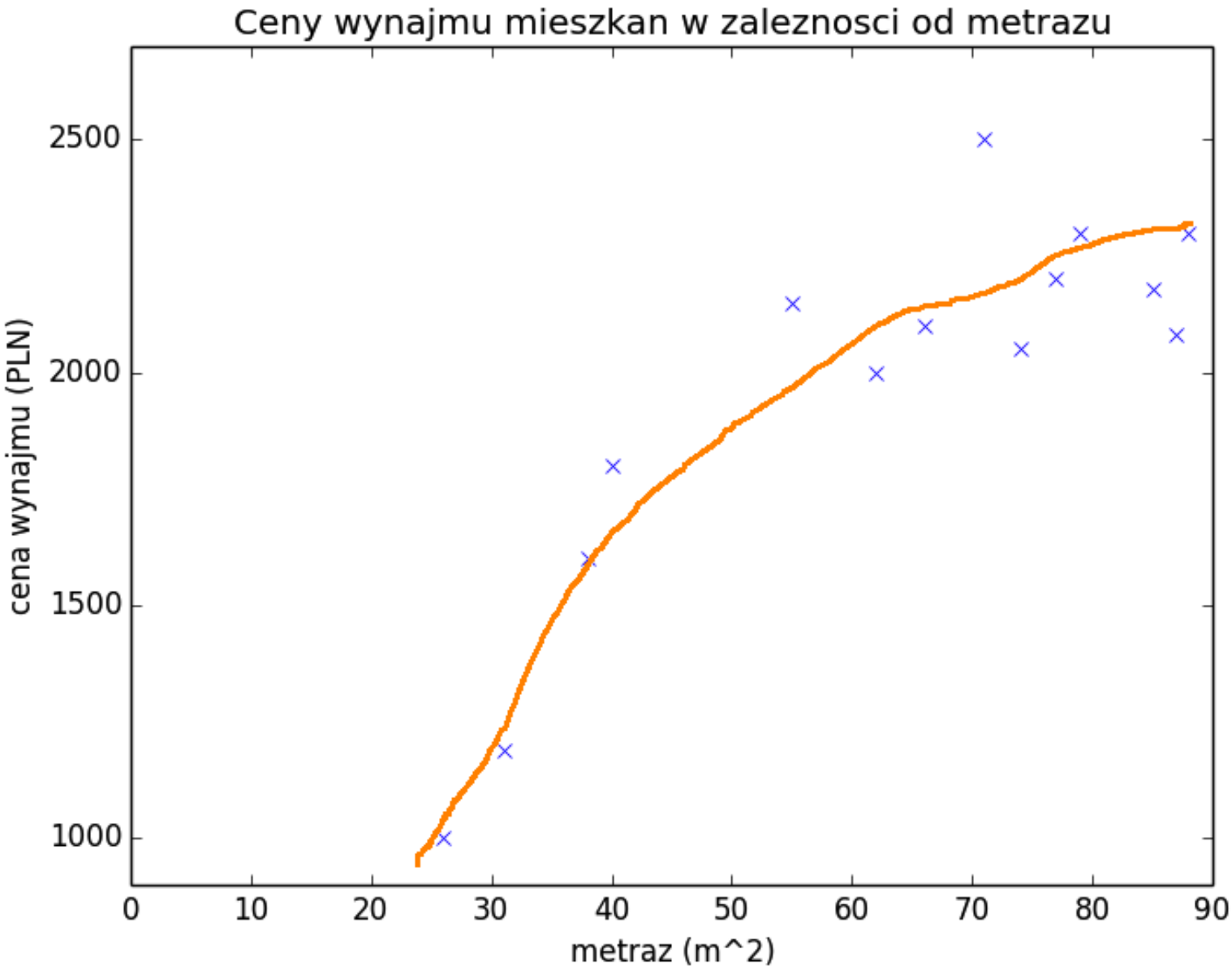


za mały:
uczenie jest wolne



za duży:
algorytm nie trafia w minimum,
a nawet nie zbiega się

Regresja wielomianowa



Regresja wielomianowa

Model liniowy nie zawsze dobrze aproksymuje dane.
W takim przypadku można posłużyć się funkcją wielomianową:

$$y = a_1 x^1 + a_2 x^2 + ... + a_k x^k + a_0$$

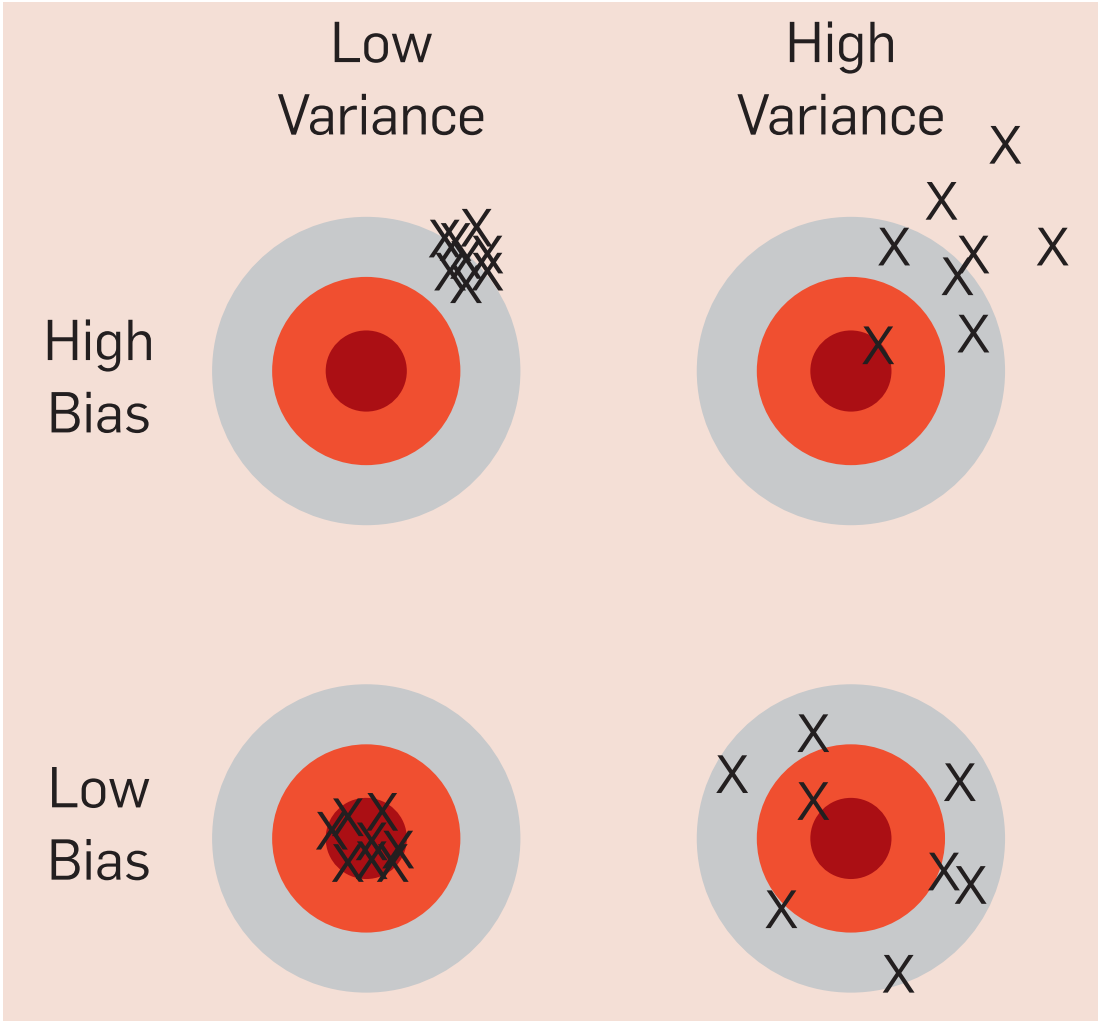
k-stopień wielomianu

Błąd uogólnienia modelu

Suma trzech różnych błędów:

- Błąd obciążenia (*bias*): tendencja do systematycznego uczenia się z tych samych błędnych rzeczy. Wynika z błędnych założeń, takich jak założenie, że dane są 'liniowe', gdy w rzeczywistości są 'kwadratowe'.
- Błąd wariancji (*variance*): tendencja do uczenia się losowych rzeczy bez względu na prawdziwy sygnał. Wynika z nadmiernej wrażliwości modelu na małe różnice w danych trenujących.
- Błąd nieredukowalny: wynika z zaszumienia samych danych. Jedynym sposobem na jego zmniejszenie jest oczyszczenie danych (np. naprawa źródeł danych, jak popsuty czujnik, który generuje pomiary, wykrycie lub usunięcie wartości odstających i różnego rodzaju przetwarzanie wstępne danych).

Obciążenie (*bias*) i wariancja (*variance*) na przykładzie gry w rzutki

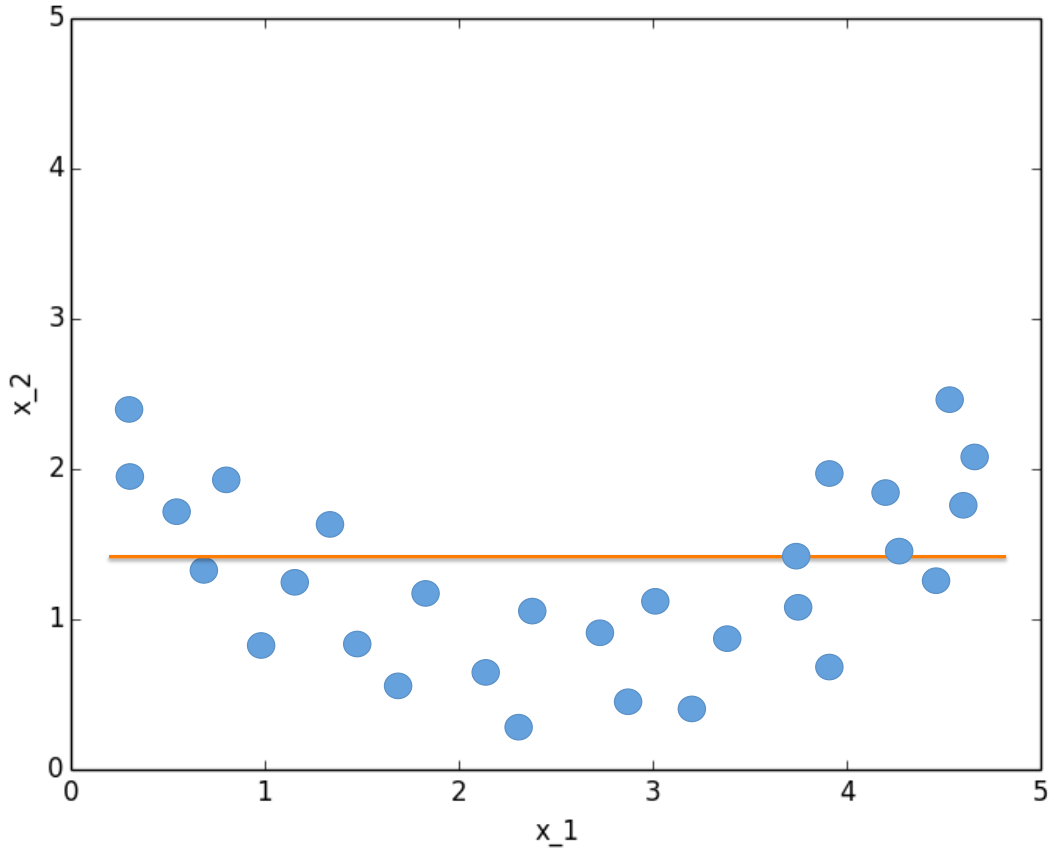


Źródło: Pedro M. Domingos: A few useful things to know about machine learning. *Commun. ACM* 55(10): 78-87 (2012)

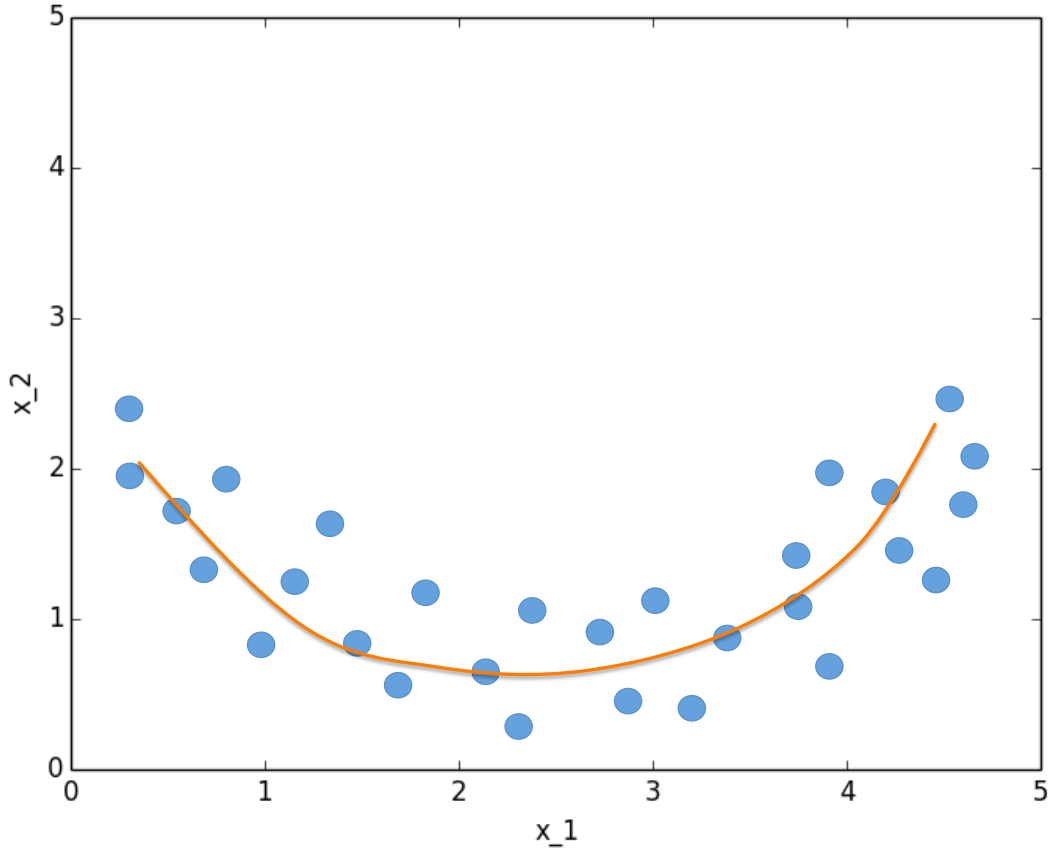
Kompromis między obciążeniem a wariancją (*bias-variance tradeoff*)

- Zwiększenie złożoności modelu zazwyczaj zwiększa jego wariancję i zmniejsza jego obciążenie.
- Zmniejszenie złożoności modelu zwiększa jego obciążenie i zmniejsza jego wariancję.
- Kiedy model staje się zbyt skomplikowany może dojść do przeuczenia – stanu, w którym model uczy się też odchyleń nie mających wpływu na realny trend.

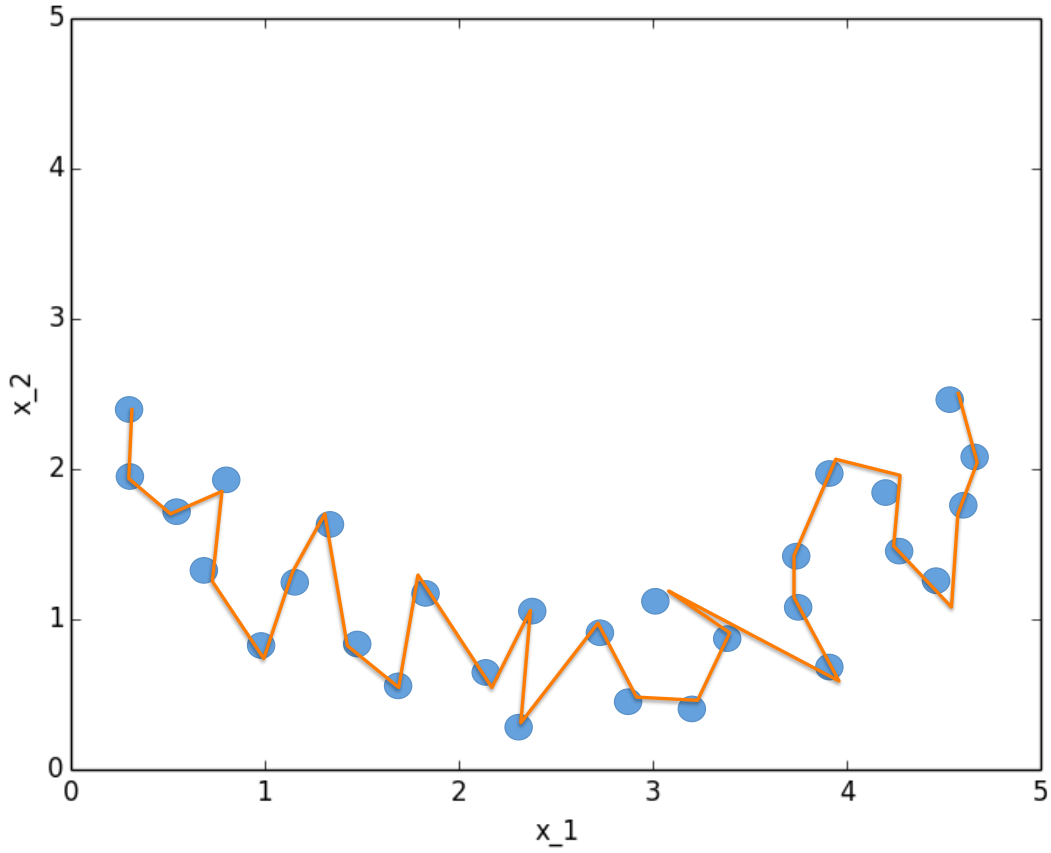
Underfitting



Dobry model: Niskie obciążenie, niska wariancja



Overfitting (przeuczenie)



Regularyzacja

- Model przeuczony bardzo kiepsko sprawdzi się na nieobserwowanych dotąd danych.
- Potrzeba mechanizmu, który potrafi zapobiegać przeuczeniu, 'sterować' tym na ile model ma generalizować dane trenujące.
- Do funkcji kosztu (na etapie trenowania modelu) dodajemy komponent regularyzacyjny.
- Regularyzacja Ridge:
$J(\vec{a}, a_0) = \frac{1}{m} \sum_{i=1}^{m} (\vec{a}^T x_i + a_0 - y_i)^2 + \alpha \frac{1}{2} \sum_{i=1}^{k} (\vec{a_i}^T)^2$

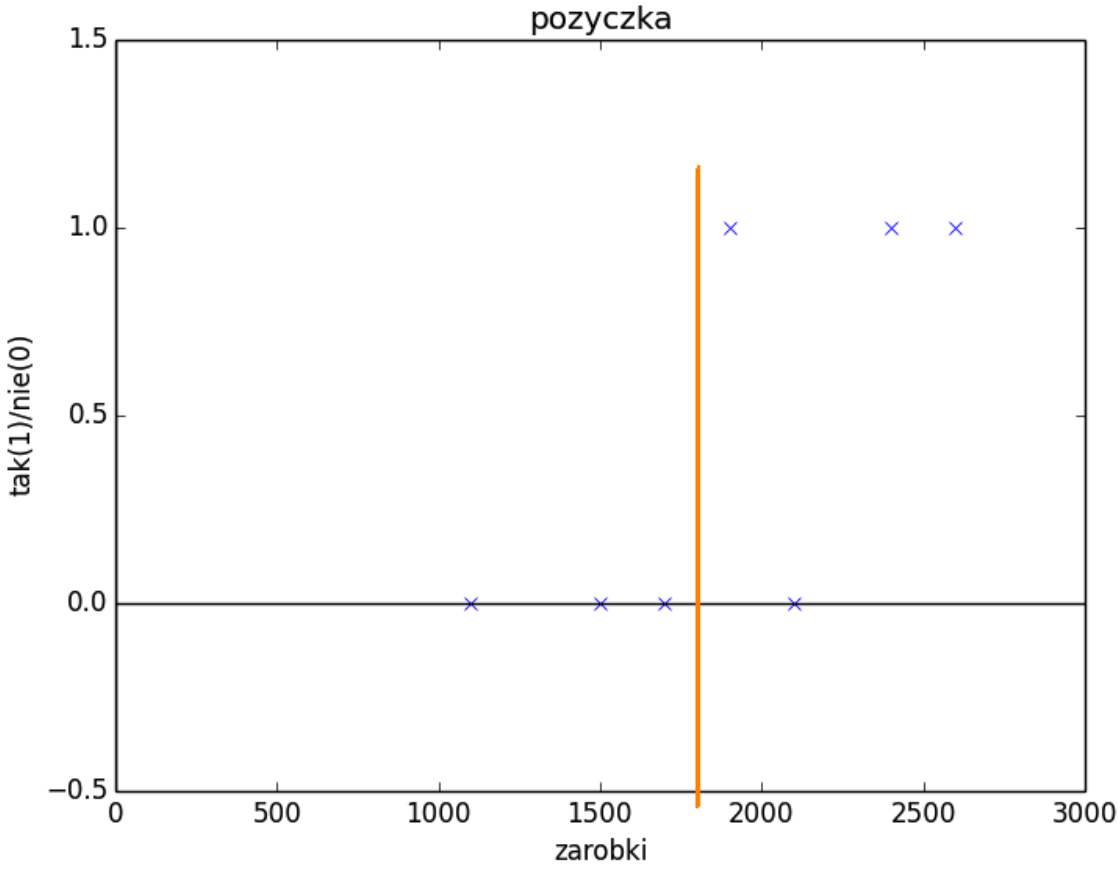
Klasyfikacja

Example

Ania, Michał i Paweł od pewnego czasu już studiują.
Ania pracuje dorywczo jako programistka gier. Jej średnie miesięczne zarobki kształtują się na poziomie 2 200PLN. Postanowiła wziąć pożyczkę na wyjazd na narty. Zastanawia się czy na podstawie jej zarobków zostanie jej udzielona pożyczka. Analiza sytuacji na rynku pokazuje aktualny trend.

Miesięczne zarobki (PLN)	Pożyczka
1 500	nie
2 100	nie
2 600	tak
1 900	tak
2 400	tak
1 100	nie
1 700	nie

Klasyfikacja



Przykładowy klasyfikator:

jeśli $h_a(x) \geq 1800$ odpowiedź=1 (tak)
jeśli $h_a(x) < 1800$ odpowiedź=0 (nie)

Regresja logistyczna - model

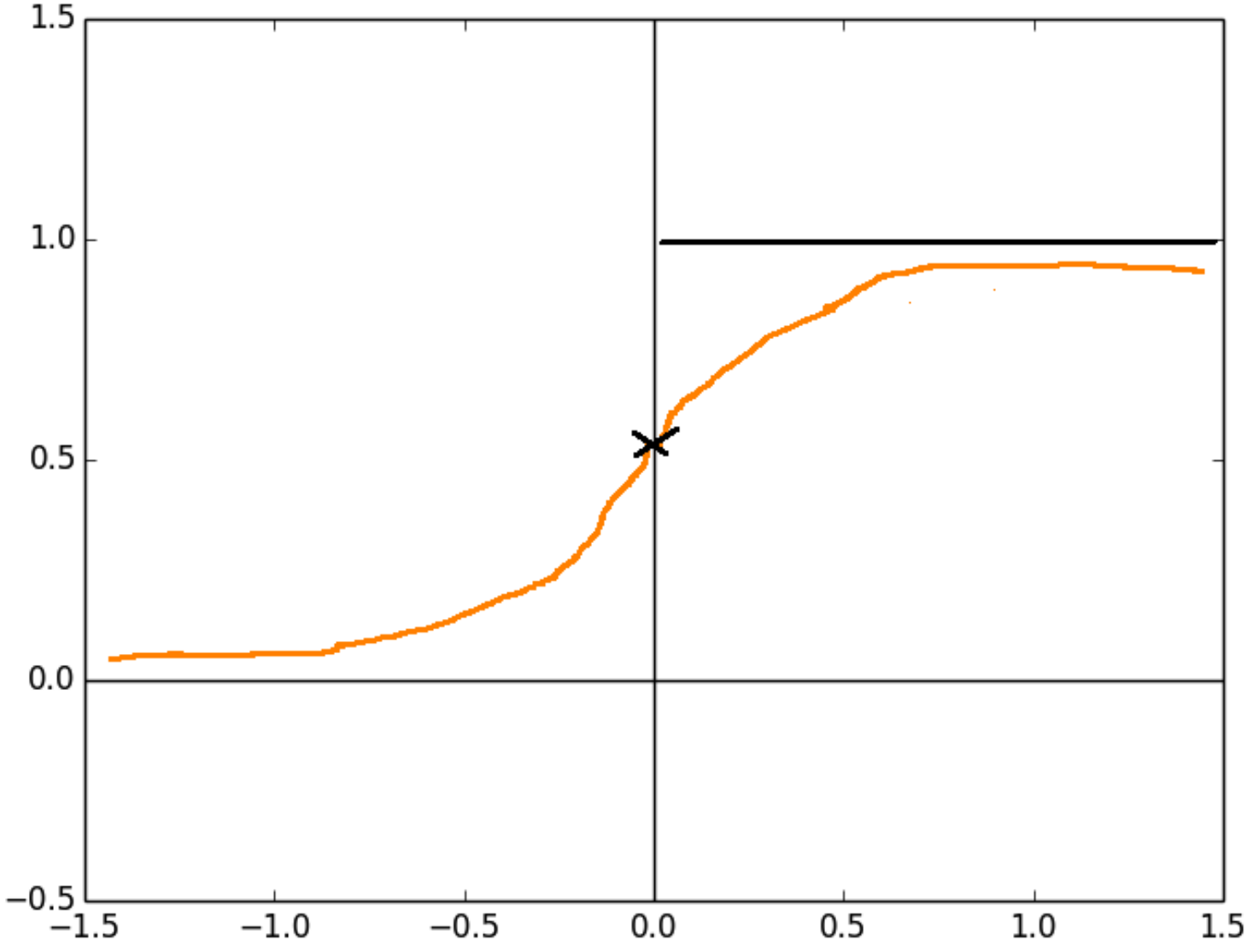
Chcielibyśmy mieć: $0 \leq h_{\vec{a}}(\vec{x}) \leq 1$

$$h_{\vec{a}}(\vec{x}) = g(\vec{a}^T \vec{x})$$

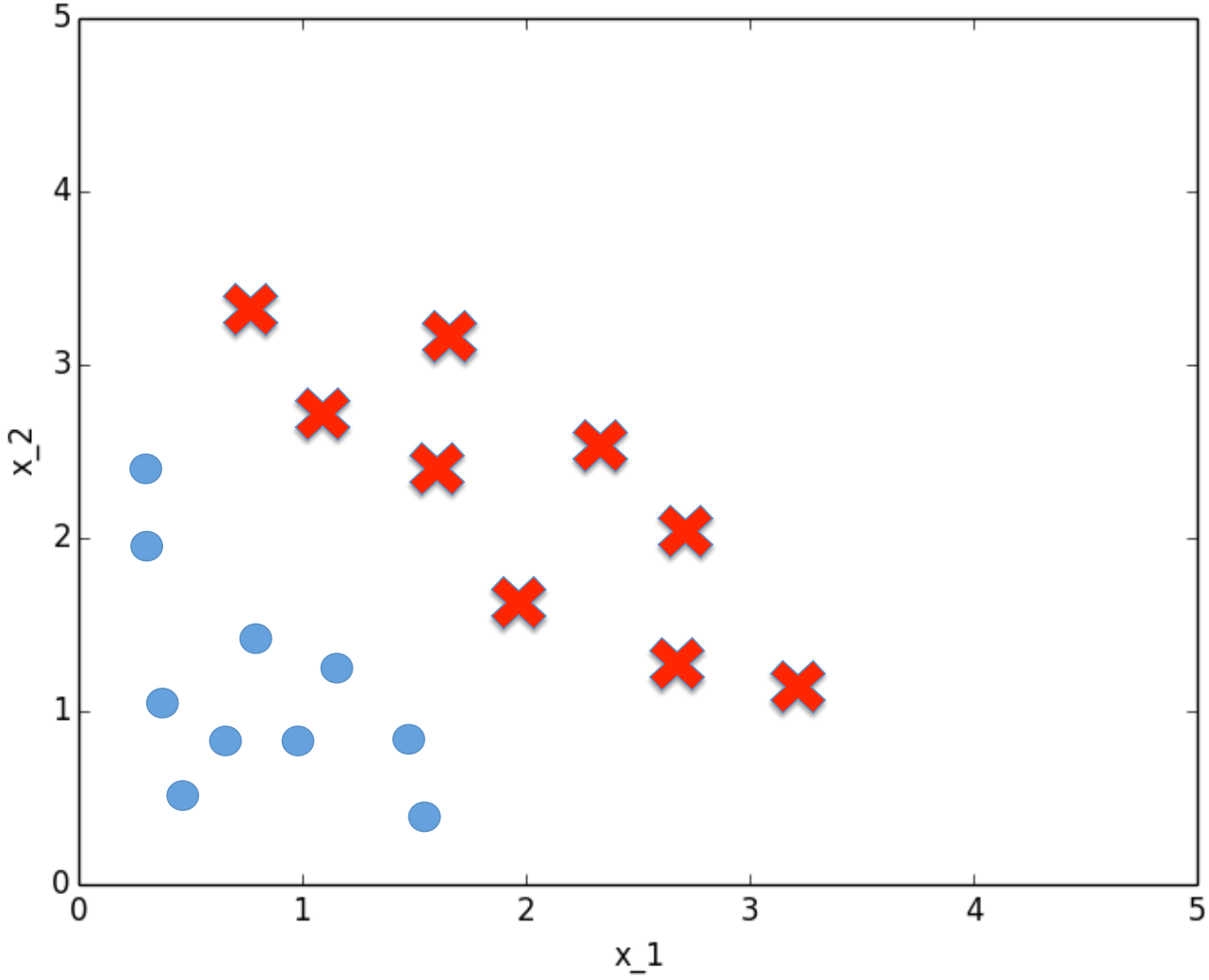
$$g(\vec{a}) = \frac{1}{1 + e^{-\vec{a}^T \vec{x}}}$$

funkcja sigmoidalna

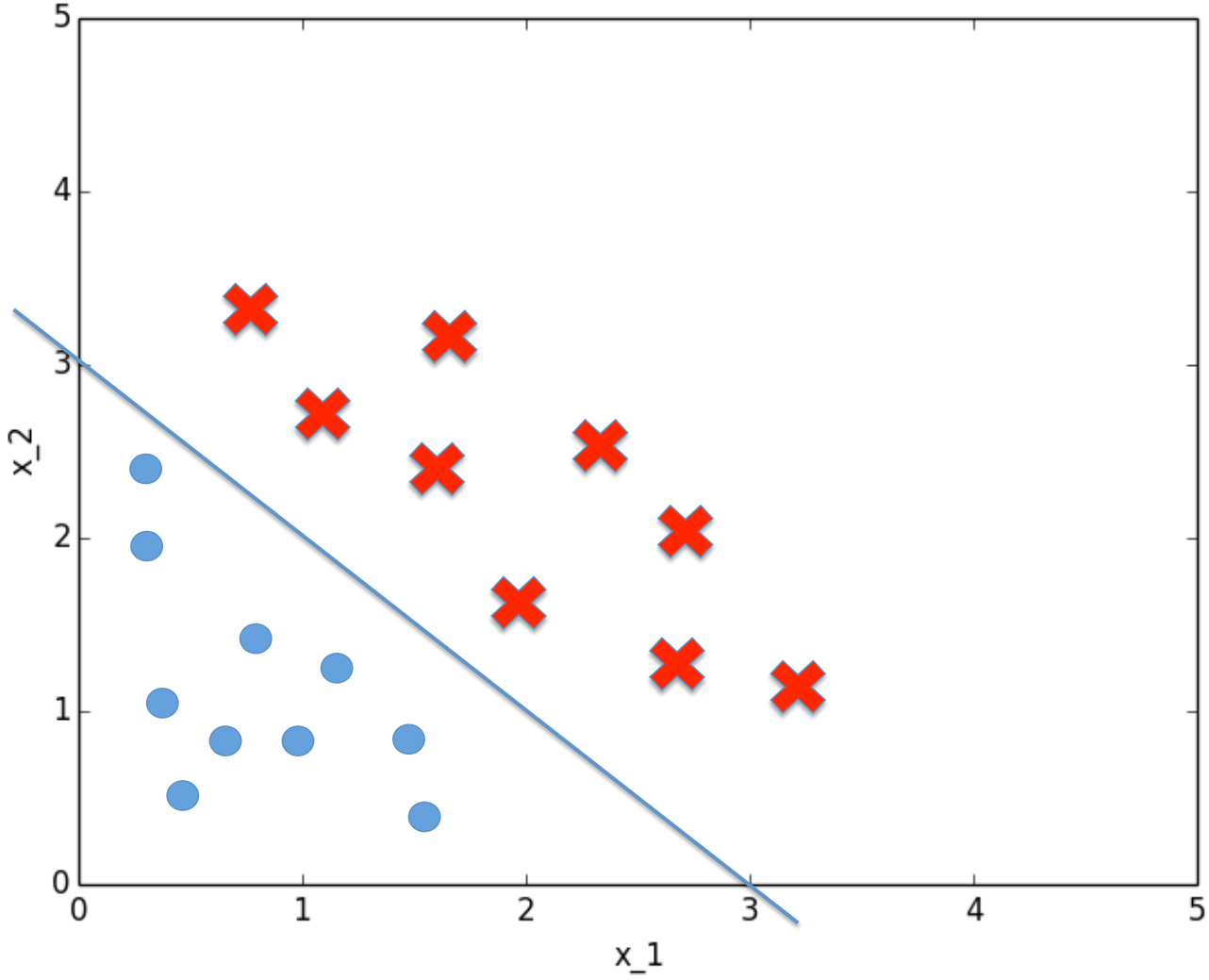
Regresja logistyczna - model



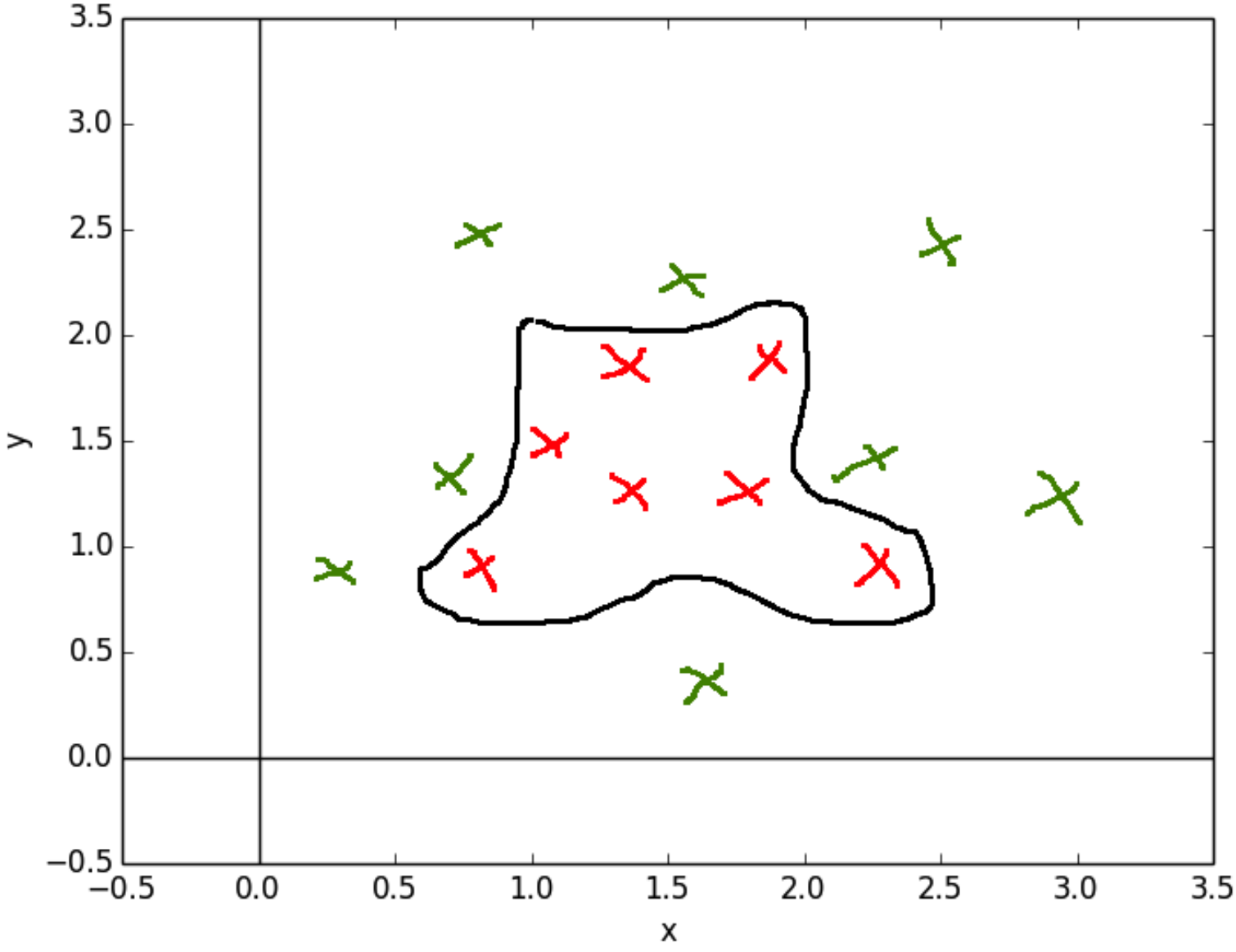
Granica klas (*decision boundary*)



Granica klas (*decision boundary*)



Granica klas (*decision boundary*)



Regresja logistyczna - funkcja kosztu

$$\text{koszt}(h_{\vec{a}}, y) = \begin{cases} -log(h_{\vec{a}}(\vec{x})) \text{ gdy } & y = 1 \\ -log(1 - h_{\vec{a}}(\vec{x})) \text{ gdy } & y = 0 \end{cases}$$

$$h_0(\vec{x}) \to 0, \text{koszt} \to \infty$$

Ewaluacja modeli uczenia maszynowego

Ewaluacja modeli

- funkcja oceny
- procedura oceny

Funkcje oceny klasyfikatora

dokładność klasyfikacji $L(i,j)$-koszt błędnej klasyfikacji

$$L(i,j) = \begin{cases} 1 \text{ gdy } & i \neq j \\ 0 \text{ gdy } & i = j \end{cases}$$

dokładność$= \frac{1}{m} \sum_{i=1}^{m} L(y_i, f(x_i))$

Precision/Recall

Macierz pomyłek:

		Rzeczywista wartość	
		T	F
Przewidywana wartość	T	True positive (TP)	False positive (FP)
	F	False negative (FN)	True negative (TN)

Precision (precyzja):

$$\frac{TP}{TP + FP}$$

Recall (czułość):

$$\frac{TP}{TP + FN}$$

Zadanie

Treść (1/2):

Ania i Michał, studenci informatyki, postanowili wybrać się w październikowy weekend na grzyby. Ponieważ większość swojego czasu spędzają przed komputerem nie czuli się kompetentnymi grzybiarzami, jednak postanowili wykorzystać swoje mocne strony podczas tej wyprawy.

W Internecie odnaleźli zbiór danych opisujący cechy grzybów jadalnych i niejadalnych i postanowili wytrenować klasyfikator, którego mogliby użyć podczas grzybobrania. Obliczyli, że ich klasyfikator ma trafność klasyfikacji 92% (na zbiorze trenującym).

Michał był dumny z tak wysokiego wyniku i chciał już przystąpić do przygotowywania zapiekanki z grzybami. Jednak Ania zawahała się (1 grzyb wydawał się jej nieco podejrzany) i postanowiła najpierw zanieść zebrane grzyby do konsultacji do babci, wytrawnej grzybiarki.

Okazało się, że spośród zebranych przez Anię i Michała 18 grzybów, babcia oceniła 16 jako jadalne a 2 grzyby okazały się po analizie babci niejadalne (o dziwo, grzyb, który wydawał się Ani podejrzany, babcia uznała za jadalny).

Mimo stosunkowo wysokiej wartości trafności klasyfikacji, Ania i Michał mogli zatruć się grzybami!

Zadanie

Treść (2/2):

a) Twoim zadaniem jest obliczenie miar precyzji (*precision*) i czułości (*recall*) oraz dokładności (*accuracy*) dla podanej poniżej macierzy pomyłek (dla zbioru testowego, zaetykietowanego przez babcię)

True positive (TP): 15	False positive (FP): 2
False negative (FN): 1	True negative (TN): 0

b) Ile grzybów zostało niepoprawnie sklasyfikowanych przez Anię i Michała?

Zadanie

Rozwiązanie:

a)

Precyzja (precision):

$$\frac{TP}{TP+FP} = \frac{15}{15+2} \approx 0,88$$

Czułość (recall):

$$\frac{TP}{TP+FN} = \frac{15}{15+1} \approx 0,94$$

Dokładność (accuracy):

$$\frac{TP+TN}{TP+TN+FP+FN} = \frac{15}{15+0+2+1} \approx 0,83$$

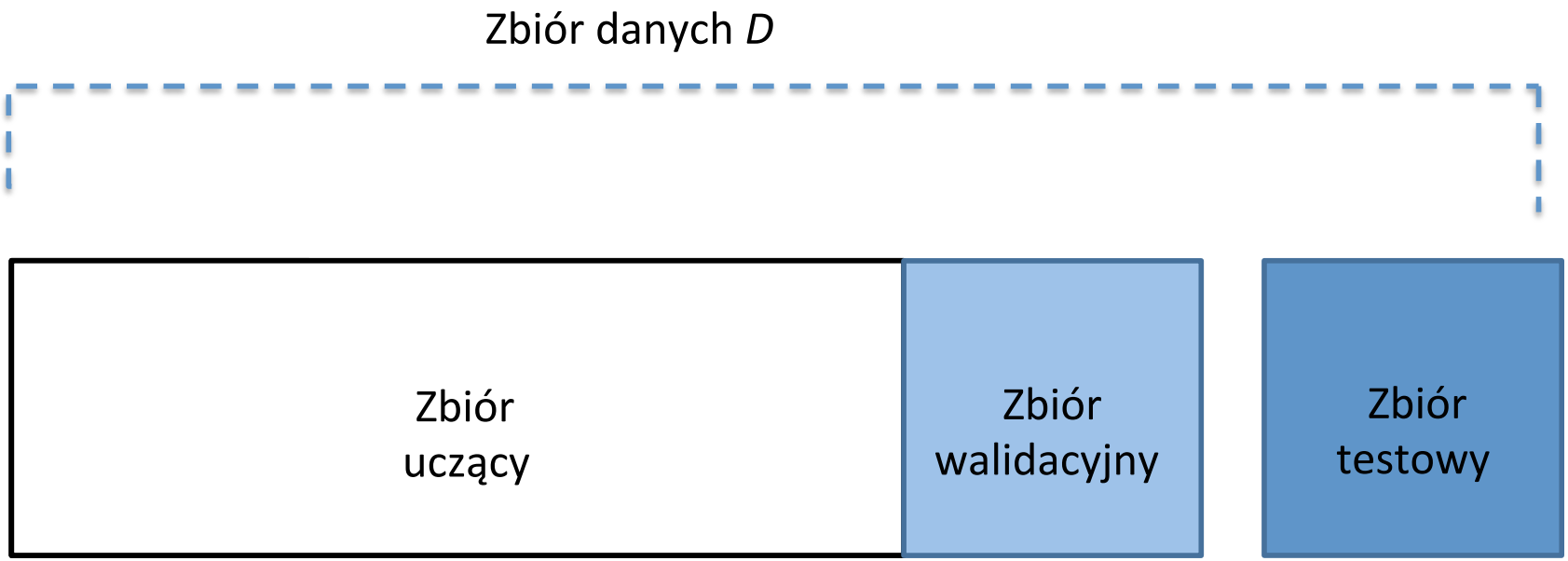
b) Trzy grzyby zostały niepoprawnie sklasyfikowane.

Która miara najlepiej nadaje się do oceny klasyfikatora w tym zadaniu?

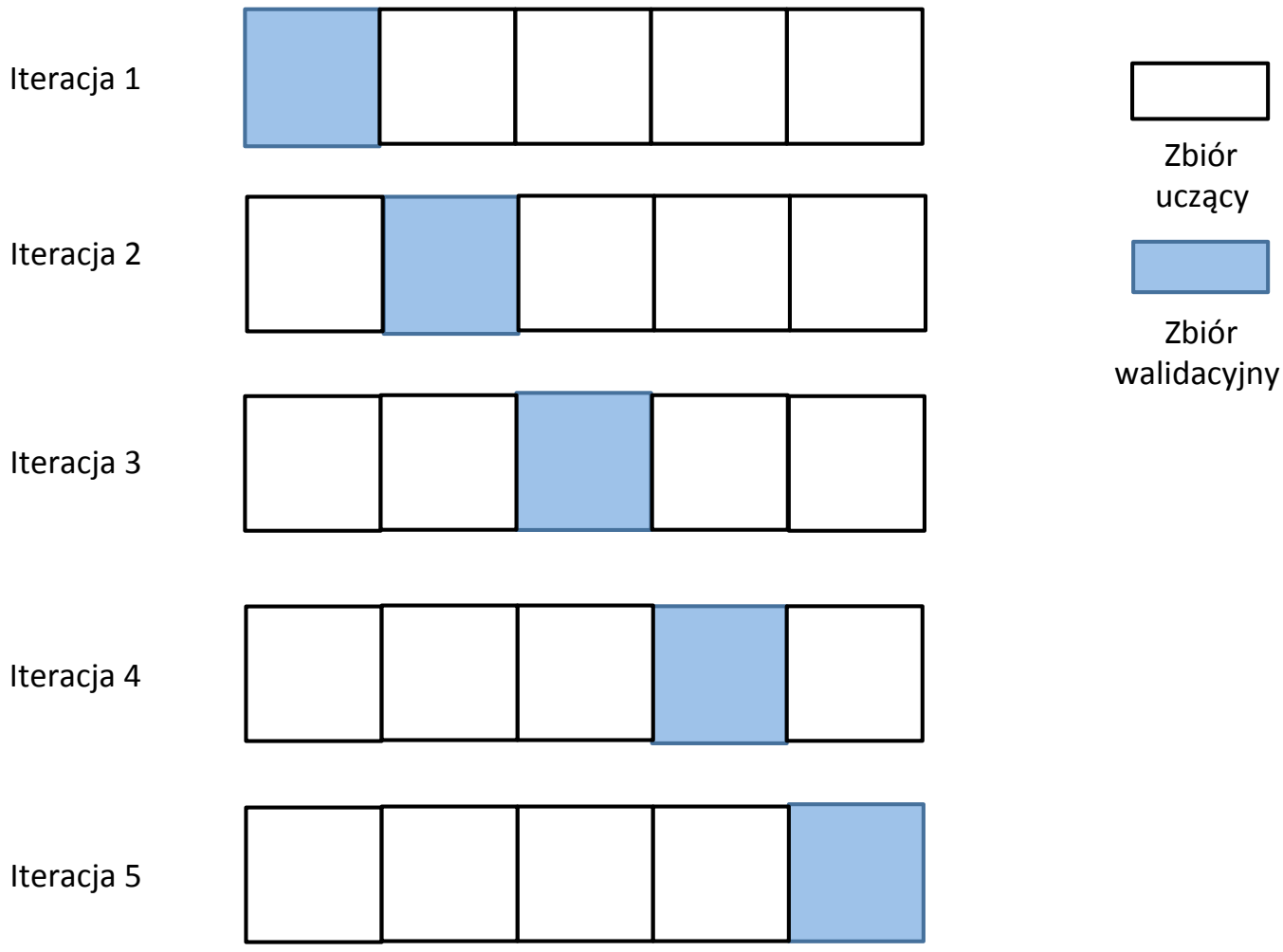
Która miara najlepiej nadaje się do oceny klasyfikatora w tym zadaniu?

Miara precyzji, jeśli chcemy zminimalizować prawdopodobieństwo zatrucia się grzybem, który został błędnie sklasyfikowany jako jadalny. Wyrzucenie jadalnego grzyba może być z naszego punktu widzenia mniej kosztowne niż utrata zdrowia.

Prawidłowy podział zbioru danych



Procedura oceny: walidacja krzyżowa (k-krotna)



Dziękuję za uwagę!