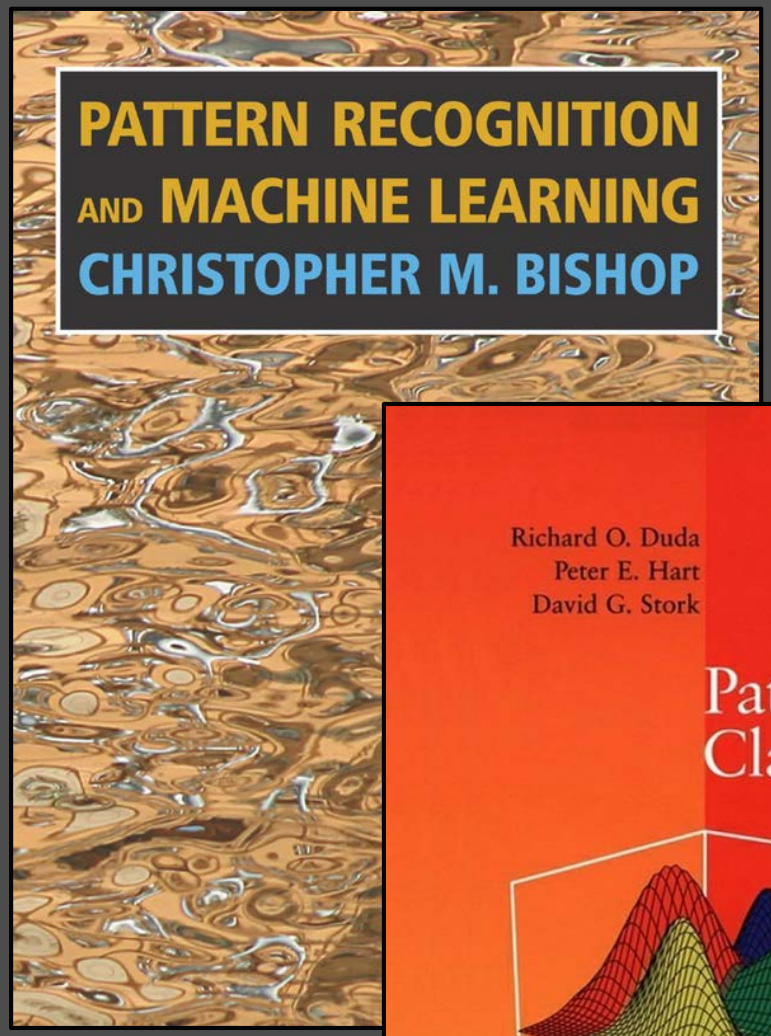
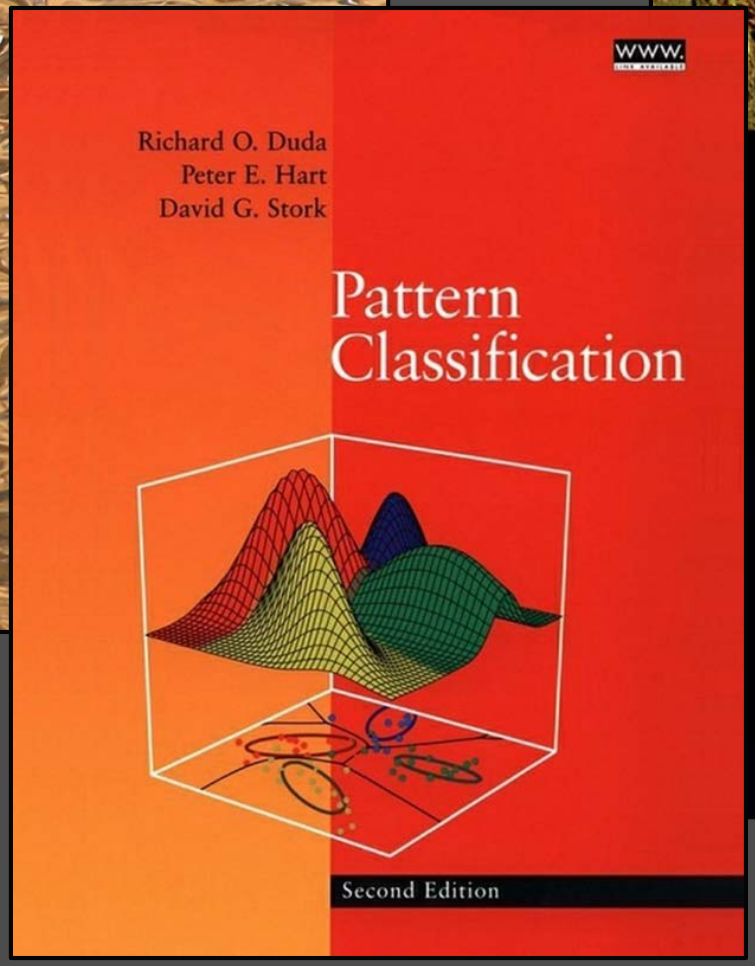
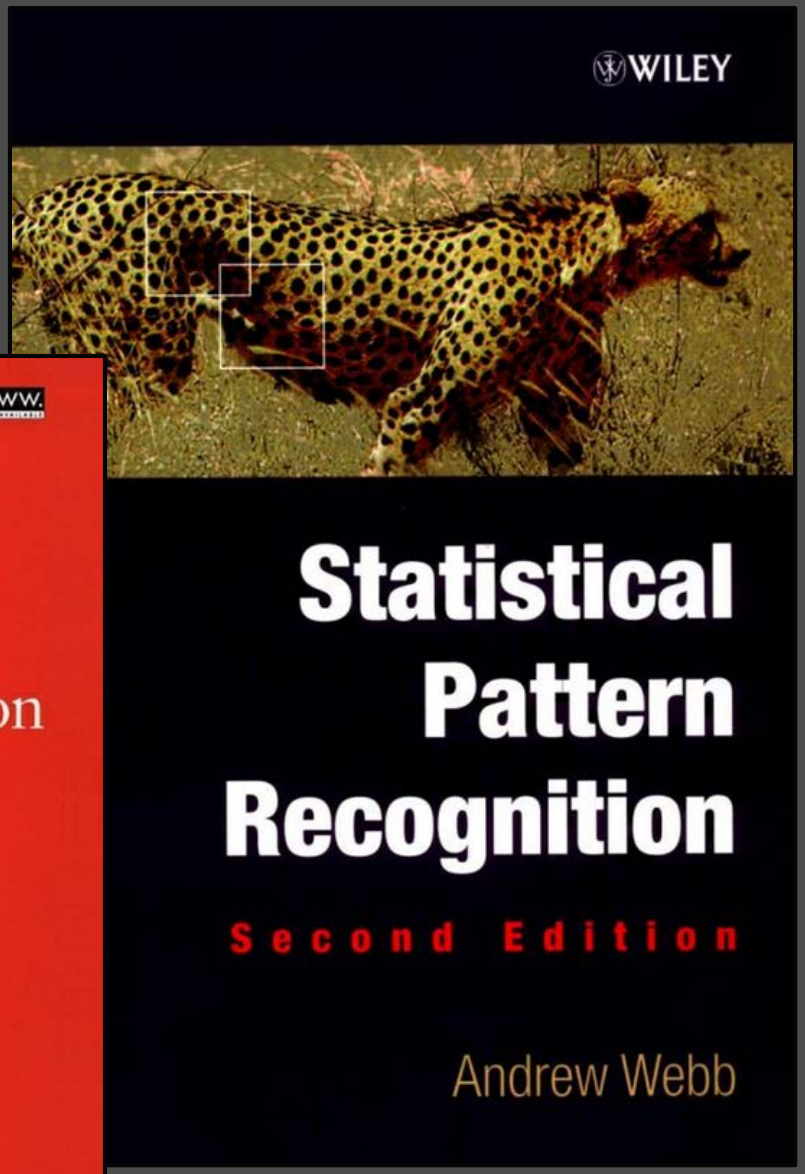

PATTERN RECOGNITION
AND MACHINE LEARNING
CHRISTOPHER M. BISHOP


WWW.
Richard O. Duda
Peter E. Hart
David G. Stork
Pattern
Classification
Second Edition


WILEY
Statistical
Pattern
Recognition
Second Edition
Andrew Webb

Popularne klasyfikatory w pakietach komputerowych

- Klasyfikator liniowy
- Uogólniony klasyfikator liniowy
- SVM
- Naiwny klasyfikator bayesowski
- Ocena klasyfikatora – ROC
- Lista popularnych pakietów

Klasyfikator liniowy

Działanie binarnego klasyfikatora liniowego:

$$g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$$

Jeżeli $g(\mathbf{x}) > 0$ to klasa ω_1

Jeżeli $g(\mathbf{x}) < 0$ to klasa ω_2

- Równanie $g(\mathbf{x}) = 0$ definiuje hiperpowierzchnię rozdzielającą

- Graficzna interpretacja

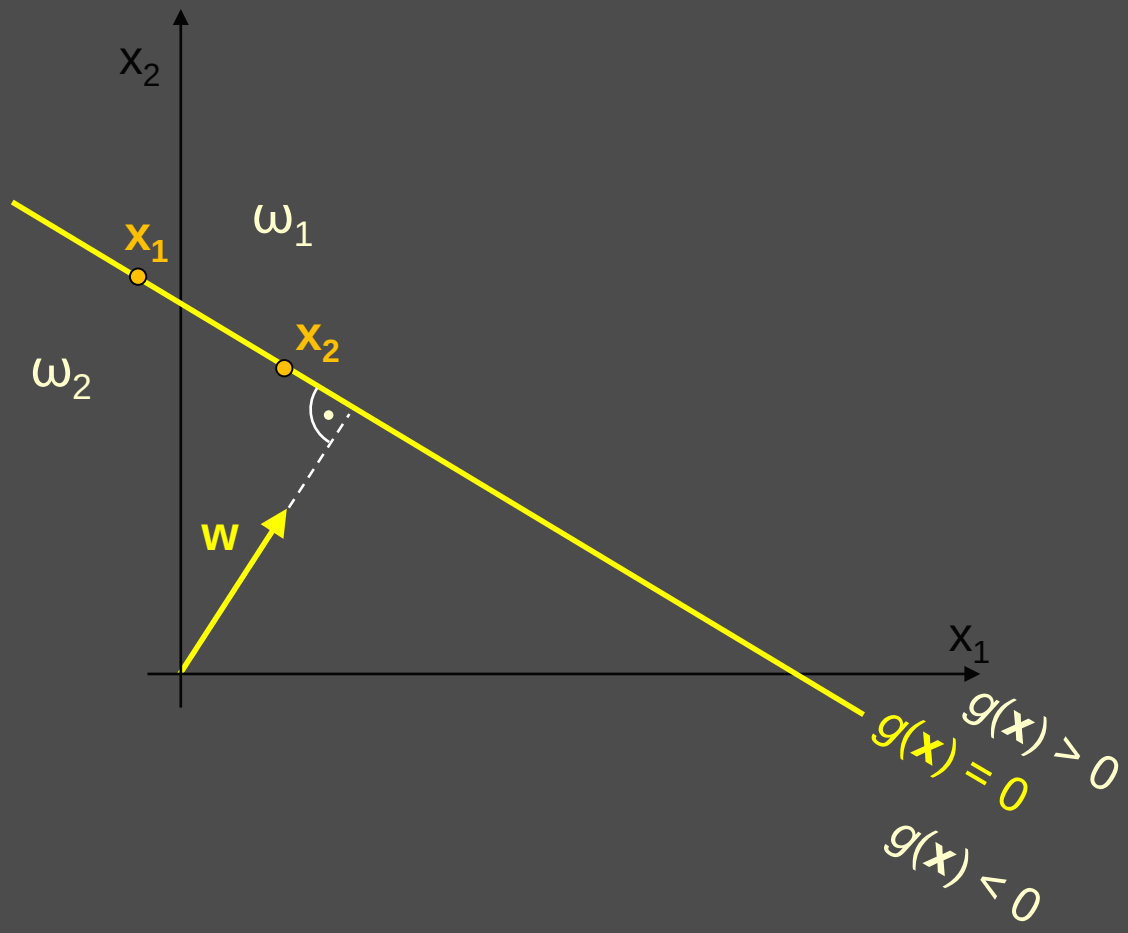
Jeżeli $\mathbf{x}_1$ i $\mathbf{x}_2$ leżą na powierzchni rozdzielającej, to

$$g(\mathbf{x}_1) = g(\mathbf{x}_2)$$

$$\mathbf{w}^T \mathbf{x}_1 + w_0 = \mathbf{w}^T \mathbf{x}_2 + w_0$$

$$\mathbf{w}^T(\mathbf{x}_1 - \mathbf{x}_2) = 0$$

Wektor **w** jest ortogonalny do powierzchni rozdzielającej

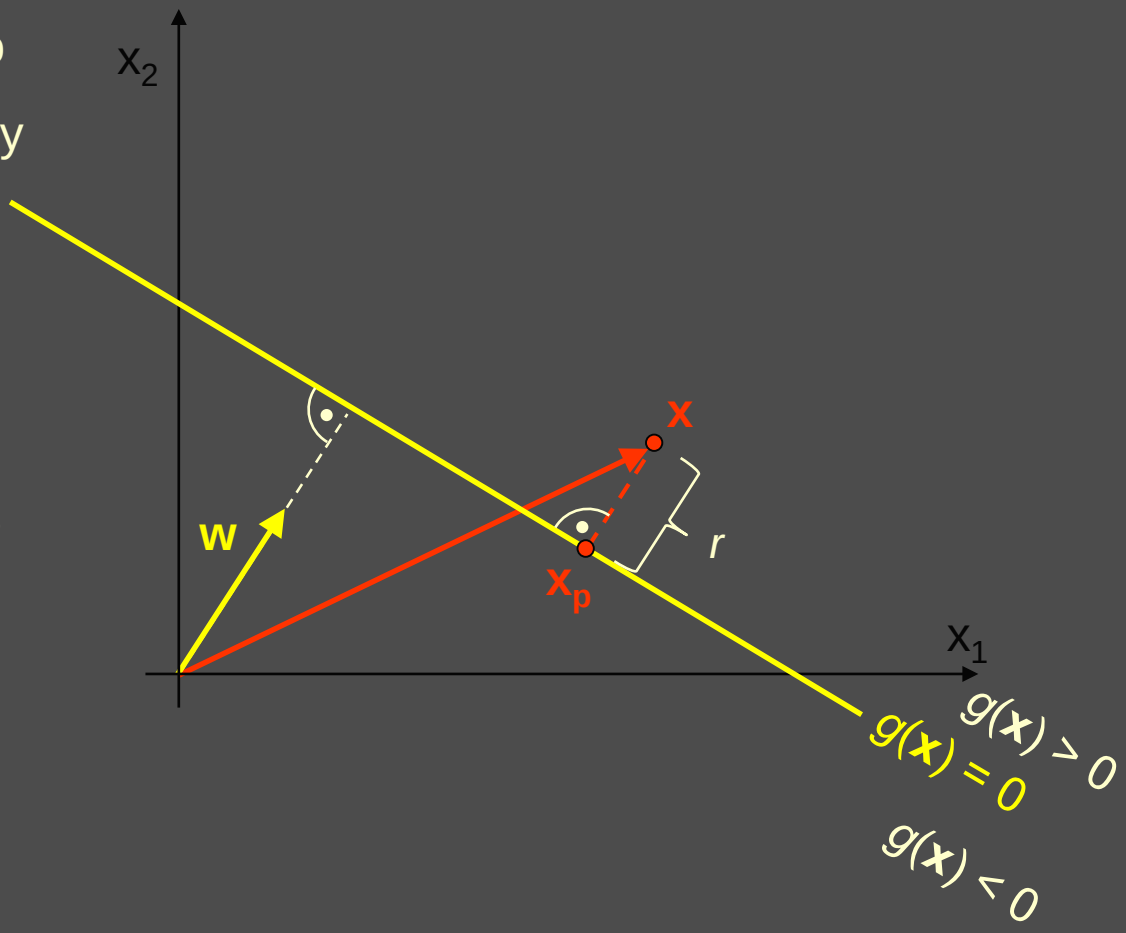


- Graficzna interpretacja

Wektor $\frac{\mathbf{w}}{\|\mathbf{w}\|}$ jest unormowany do 1 a kierunek i zwrot ma identyczny jak **w**. Można zatem wyrazić dowolny wektor **x** w postaci

$$\mathbf{x} = \mathbf{x}_p + r \frac{\mathbf{w}}{\|\mathbf{w}\|},$$

gdzie $\mathbf{x}_p$ jest rzutem wektora **x** na powierzchnię rozdzielającą.



- Graficzna interpretacja

Mnożąc lewostronnie przez $\mathbf{w}^T$ i

dodając obustronnie w_0 dostajemy:

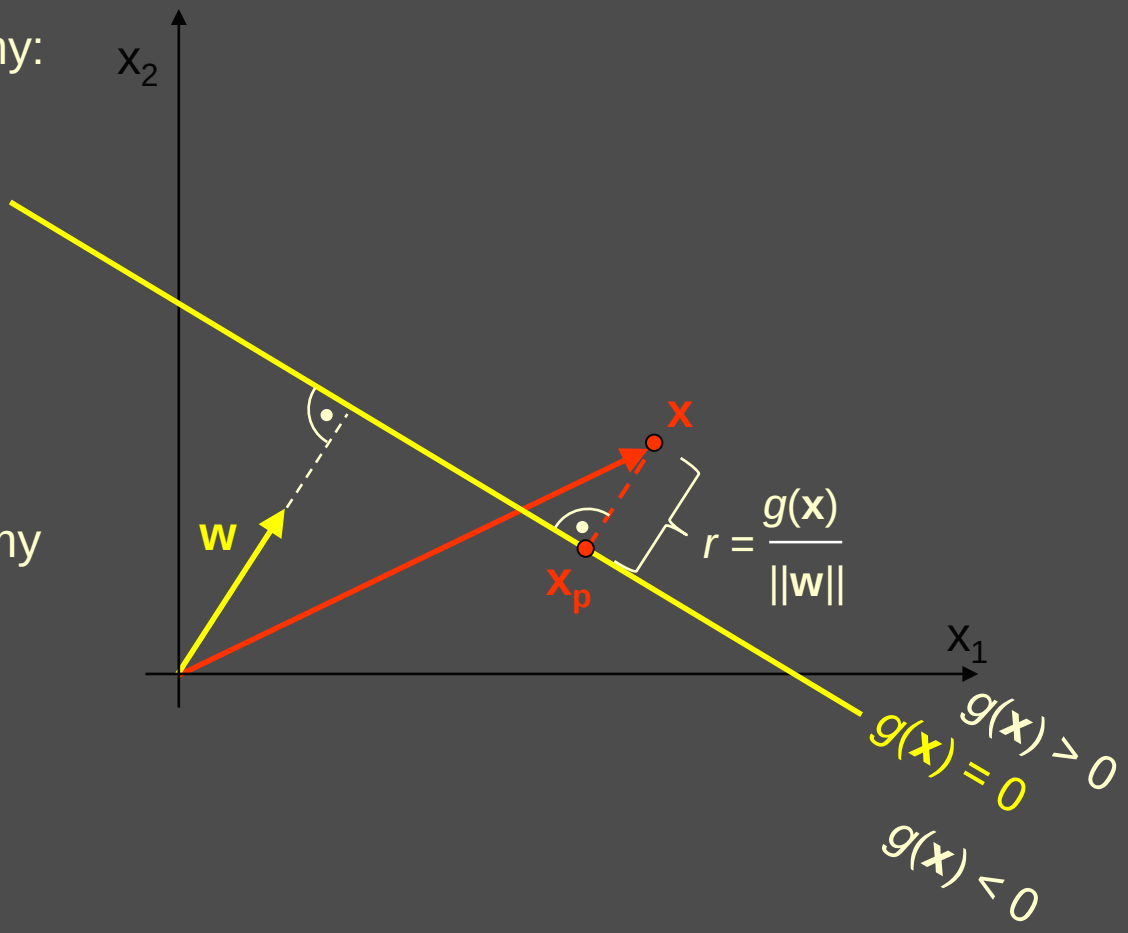
$$\mathbf{w}^T\mathbf{x} = \mathbf{w}^T\mathbf{x}_p + r\frac{\mathbf{w}^T\mathbf{w}}{\|\mathbf{w}\|}$$

$$\mathbf{w}^T\mathbf{x} + w_0 = \mathbf{w}^T\mathbf{x}_p + w_0 + r\frac{\|\mathbf{w}\|^2}{\|\mathbf{w}\|}$$

$$g(\mathbf{x}) = g(\mathbf{x}_p) + r\,\|\mathbf{w}\|,$$

a ponieważ $g(\mathbf{x}_p) = 0$, to dostajemy

$$r = \frac{g(\mathbf{x})}{\|\mathbf{w}\|}$$

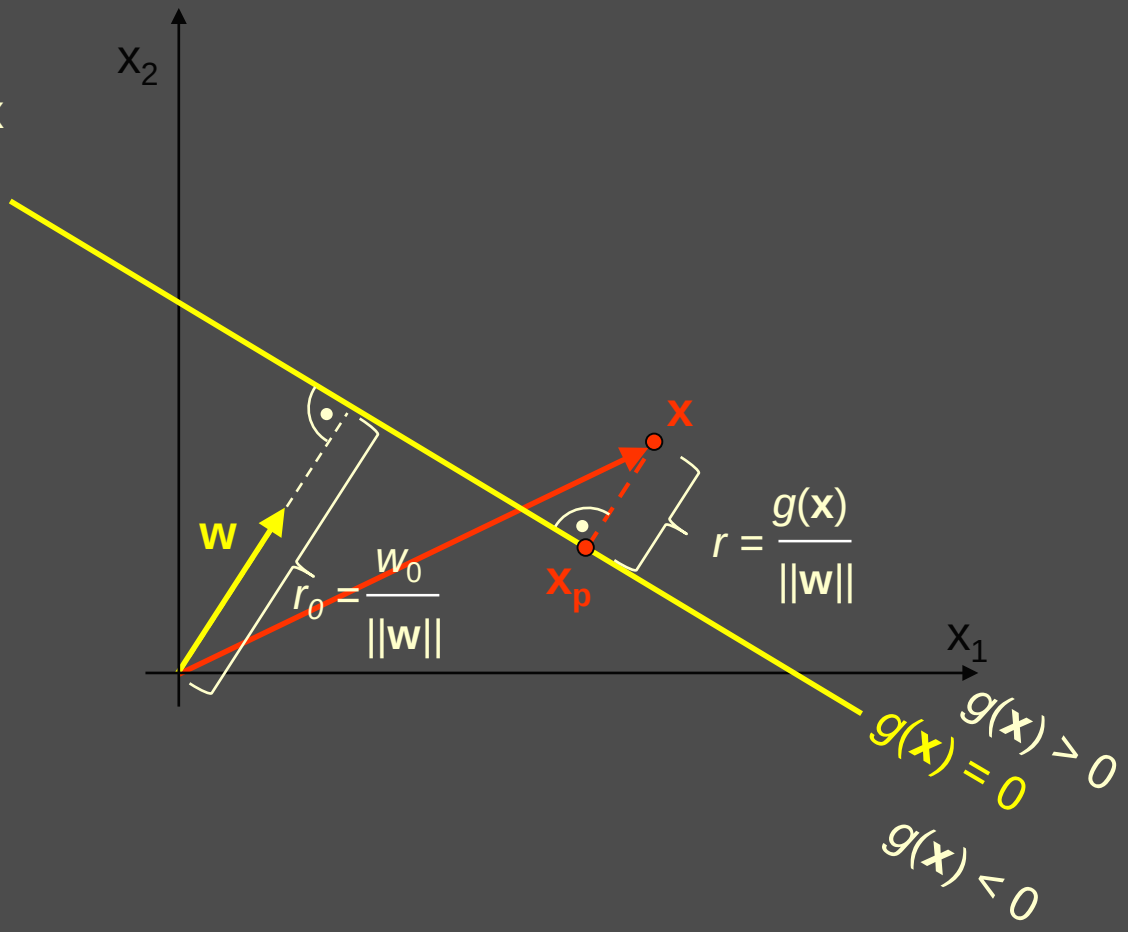


- Graficzna interpretacja

W szczególnym przypadku, dla **x** leżącego w początku układu współrzędnych (**x** = **0**) zachodzi

$$g(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0 = w_0 ,$$

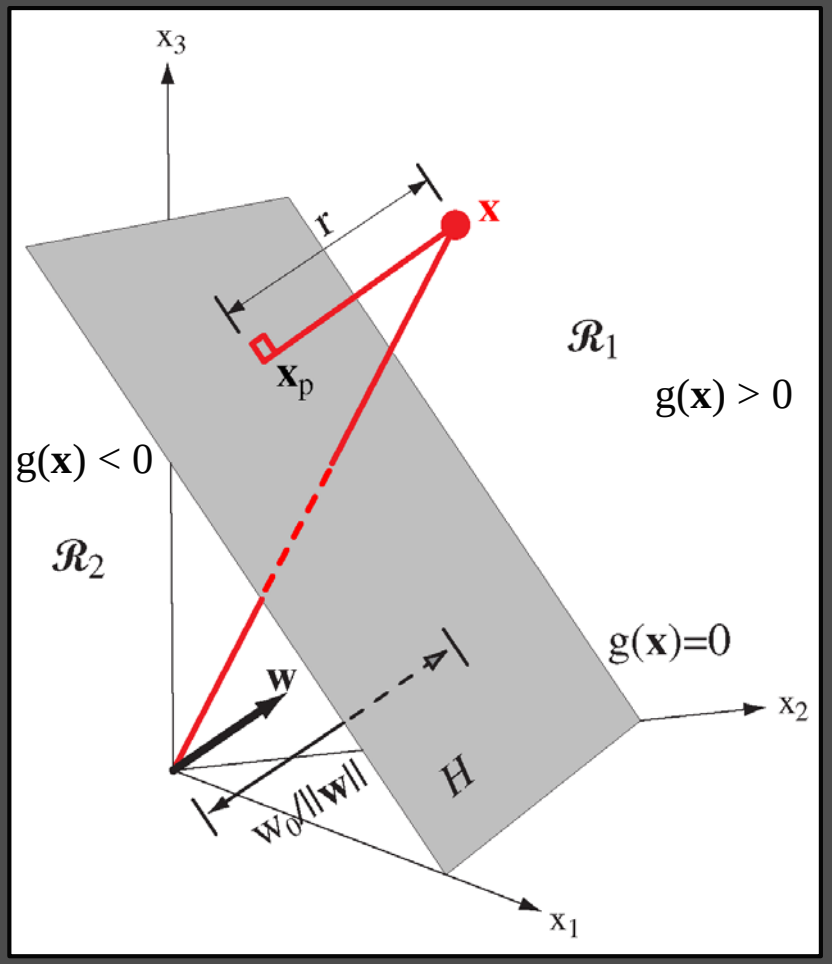
i wówczas $r_0 = \frac{w_0}{\|\mathbf{w}\|}$



- Graficzna interpretacja w trzech wymiarach

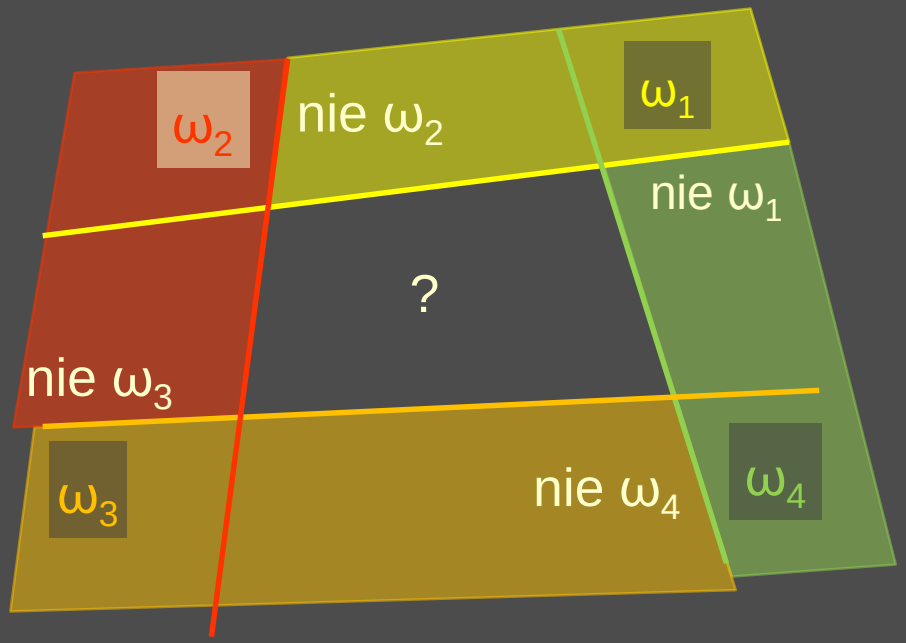
Zależność $r = \frac{g(\mathbf{x})}{\|\mathbf{w}\|}$ ($\|\mathbf{w}\| = const.$)

oznacza, że funkcję $g(\mathbf{x})$ można traktować jako miarę odległości punktu $\mathbf{x}$ od powierzchni $g(\mathbf{x}) = 0$.



Klasyfikator dla wielu klas

- *c* klas obiektów
- Można zastosować *c* – 1 klasyfikatorów binarnych (wada: istnienie obszarów o niejednoznacznej klasyfikacji)



- Zdefiniowanie *c* dyskryminatorów liniowych:

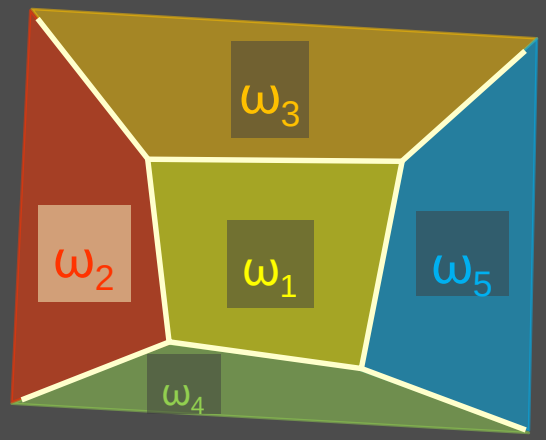
$$g_i(\mathbf{x}) = \mathbf{w}_i^T\mathbf{x} + w_{i0} \quad (i = 1,\ldots,c)$$

Jeżeli dla wszystkich $j \neq i$ $g_i(\mathbf{x}) > g_j(\mathbf{x})$, to klasa ω_i

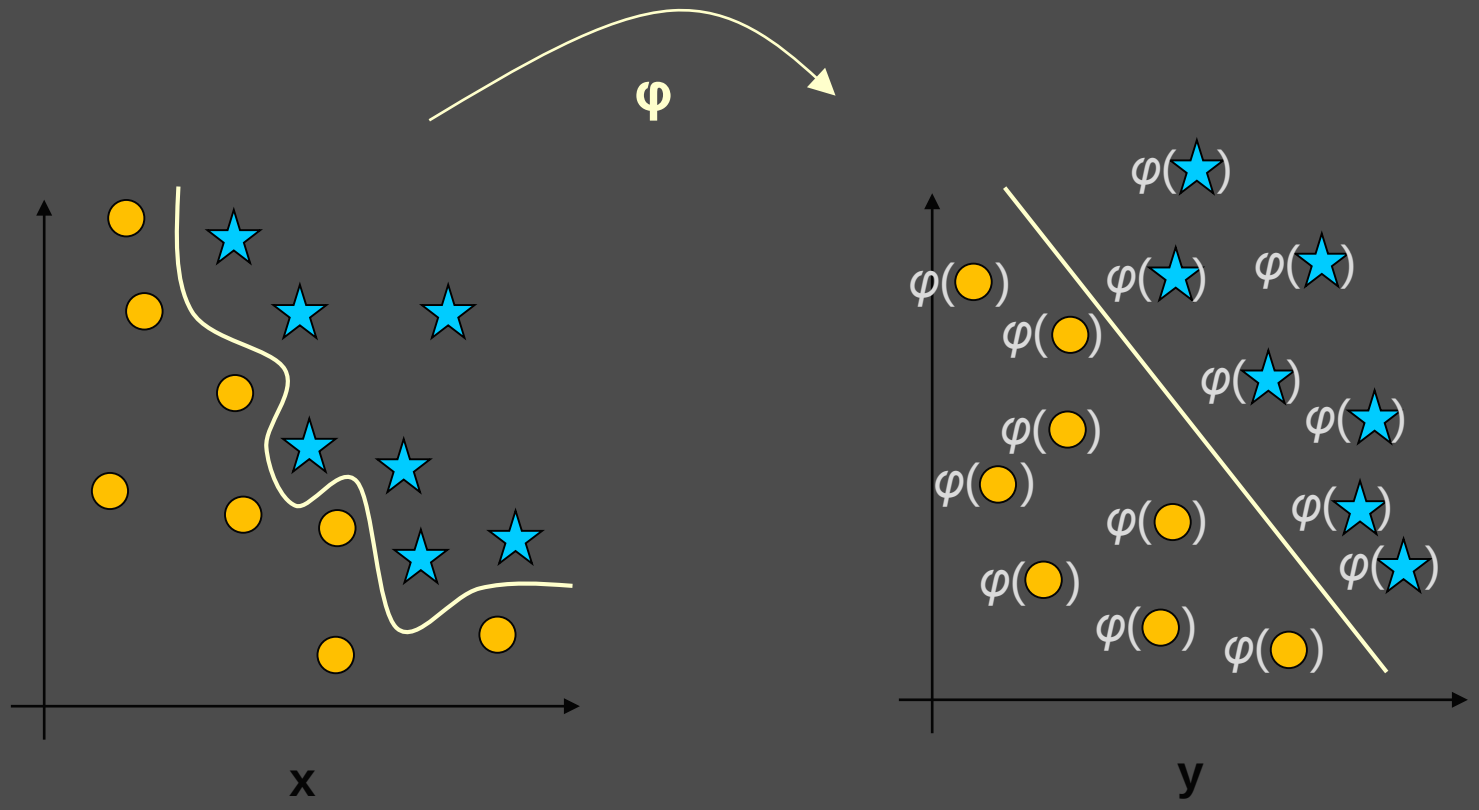
- Powierzchnie rozdzielające są zdefiniowane jako $g_i(\mathbf{x}) = g_j(\mathbf{x})$ a zatem

$$(\mathbf{w}_i - \mathbf{w}_j)^T\mathbf{x} + (w_{i0} - w_{j0}) = 0$$

- Obszary decyzyjne klasyfikatora liniowego są wypukłe



- Nieliniowe przekształcenie **φ** – interpretacja geometryczna



Uogólniony klasyfikator liniowy

- Zdefiniować nieliniową transformację do nowej przestrzeni, w której określony będzie klasyfikator liniowy
- Funkcja *g* ma uogólnioną postać

$$g(\mathbf{x}) = \mathbf{w}^T\boldsymbol{\varphi}(\mathbf{x})$$

lub dokładniej: $g(\mathbf{x}) = \mathbf{w}^T\mathbf{y}$, $\mathbf{y} = \boldsymbol{\varphi}(\mathbf{x})$,

$$\boldsymbol{\varphi}(\mathbf{x}) = [\varphi_1(\mathbf{x}) \ \ldots \ \varphi_m(\mathbf{x})]^T$$

- Geneza: *metoda funkcji potencjałowych* (1964)
- Przekształcenie **φ** odwzorowuje *d*-wymiarowe punkty **x** w *m*-wymiarowe punkty **y**

- Typowe postacie funkcji φ_i:
  - kwadratowa, wielomian, funkcja Gaussa, logarytmiczna
- Przekształcenie $\boldsymbol{\varphi}(\mathbf{x})$ jest nieliniowe, ale klasyfikator g jest liniowy ze względu na $\mathbf{y}$
- Można zatem stosować algorytmy uczenia dla klasyfikatorów liniowych
- Model z parametrem swobodnym w_0 uzyskuje się przez rozszerzenie wektora $\mathbf{y}$ o składową równą 1:

$$\mathbf{y} = [1\ x_1\ \ldots\ x_d]^T = [1\ \mathbf{x}^T]^T$$

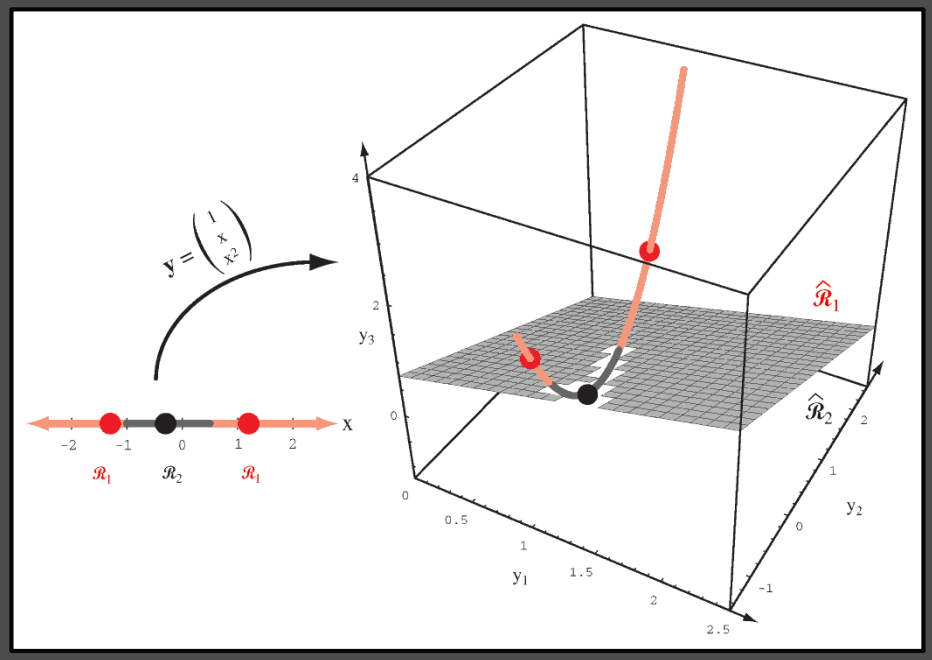
oraz odpowiednio $\mathbf{w} = [w_0\ w_1\ \ldots\ w_d]^T$

Przykład 1

$$g(x) = w_1 + w_2 x + w_3 x^2 = \mathbf{w}^T\mathbf{y},$$

$$\mathbf{w} = [w_1\ w_2\ w_3\ ]^T\ , \qquad \mathbf{y} = [1\ x\ \ x^2\ ]^T$$

Nieliniowe przekształcenie **φ**
z przestrzeni 1-wymiarowej do
3-wymiarowej pozwala
klasyfikatorowi liniowemu
(w rozszerzonej przestrzeni)
odseparować klasy
nieseparowalne liniowo
(w oryginalnej przestrzeni).

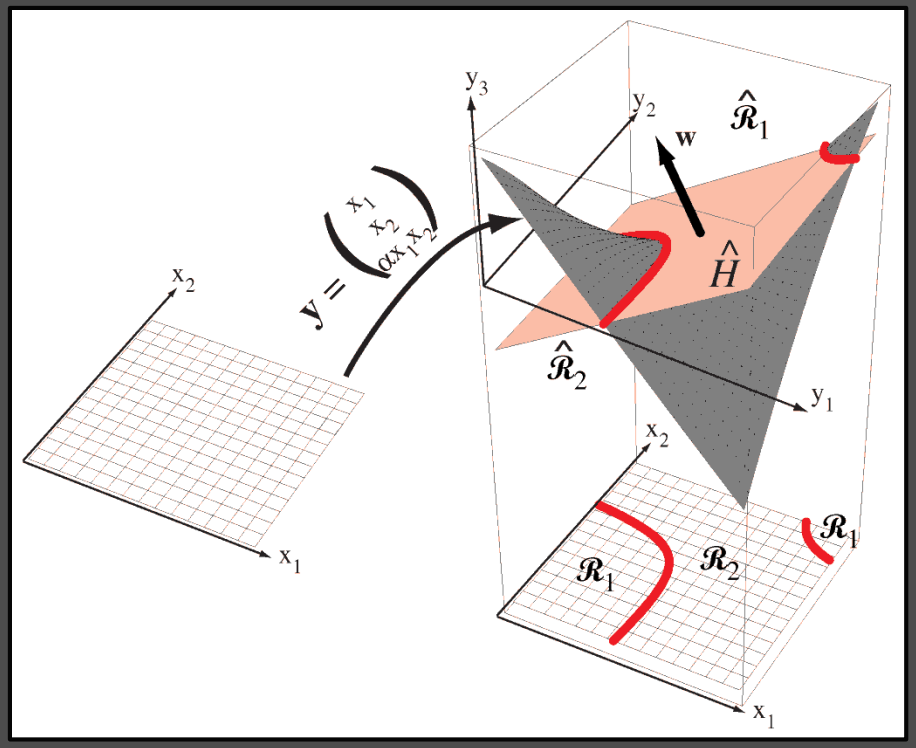


Przykład 2

$$g(x) = w_1 x_1 + w_2 x_2 + w_3 x_1 \cdot x_2 = \mathbf{w}^T \mathbf{y},$$

$$\mathbf{w} = [w_1 \; w_2 \; w_3]^T, \qquad \mathbf{y} = [x_1 \;\; x_2 \;\; x_1 \cdot x_2]^T$$

Powierzchnie rozdzielające w 3-wymiarowej przestrzeni zmiennych **y** są liniowe, jednak w 2-wymiarowej przestrzeni zmiennych **x** przyjmują bardziej skomplikowany kształt.

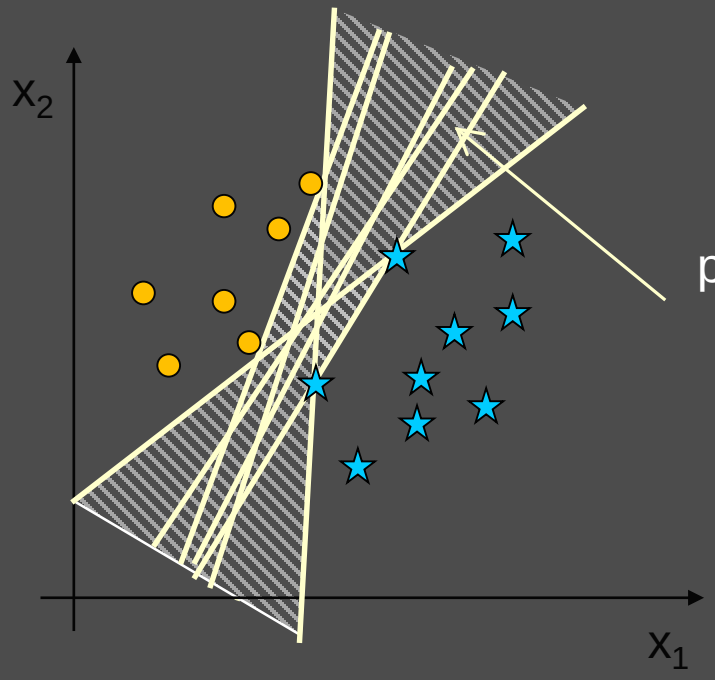


Popularne przykłady uogólnionych klasyfikatorów liniowych

- Dwuwarstwowa sieć neuronowa (perceptron) z liniową warstwą wyjściową
- Sieci neuronowe o radialnych funkcjach bazowych (RBF – *Radial Basis Function*)
- Maszyny wektorów wspierających (SVM – *Support Vector Machines*)

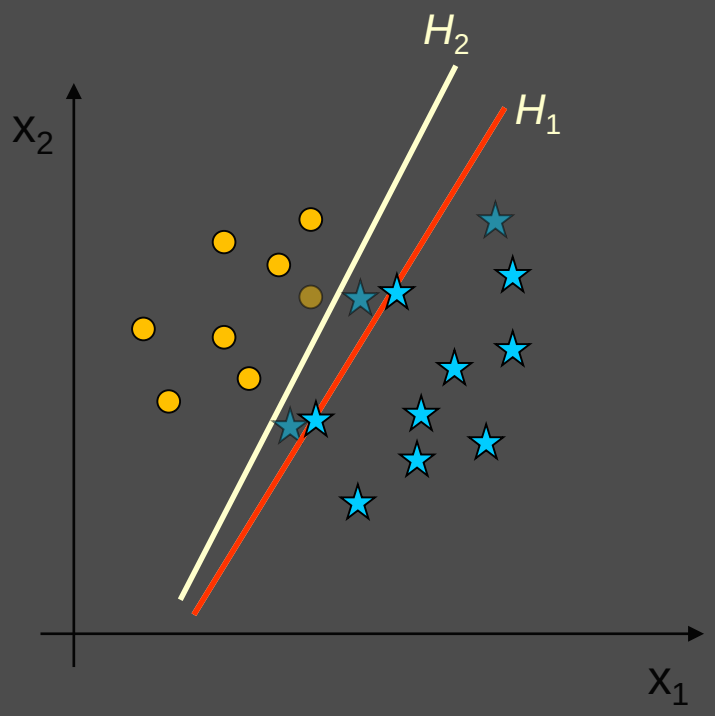
SVM – *Support Vector Machine*

- Początki: Vapnik 1992
- Problem: położenie powierzchni rozdzielającej nie jest jednoznacznie określone (zakładamy liniową separowalność klas)



poprawne powierzchnie rozdzielające

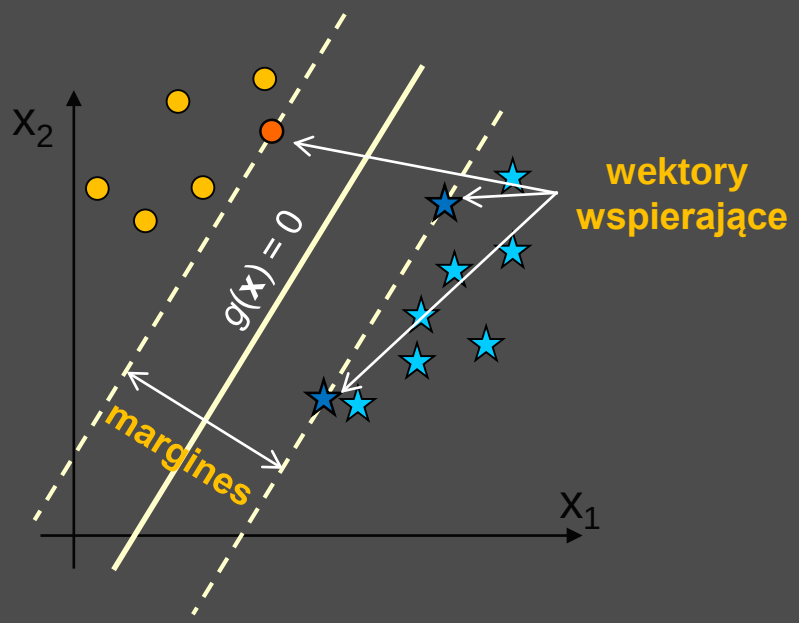
- Którą z nich wybrać?
- Co się stanie, jeżeli przeprowadzimy klasyfikację nowych obiektów?



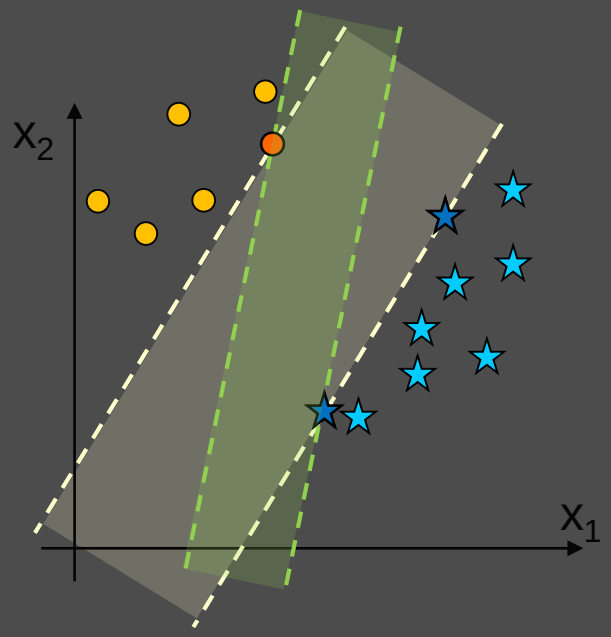
Wybór powierzchni rozdzielającej H_1 położonej w pobliżu obiektów ze zbioru uczącego jest ryzykowny :

istnieje duża szansa, że nowy obiekt znajdzie się po jej niewłaściwej stronie.

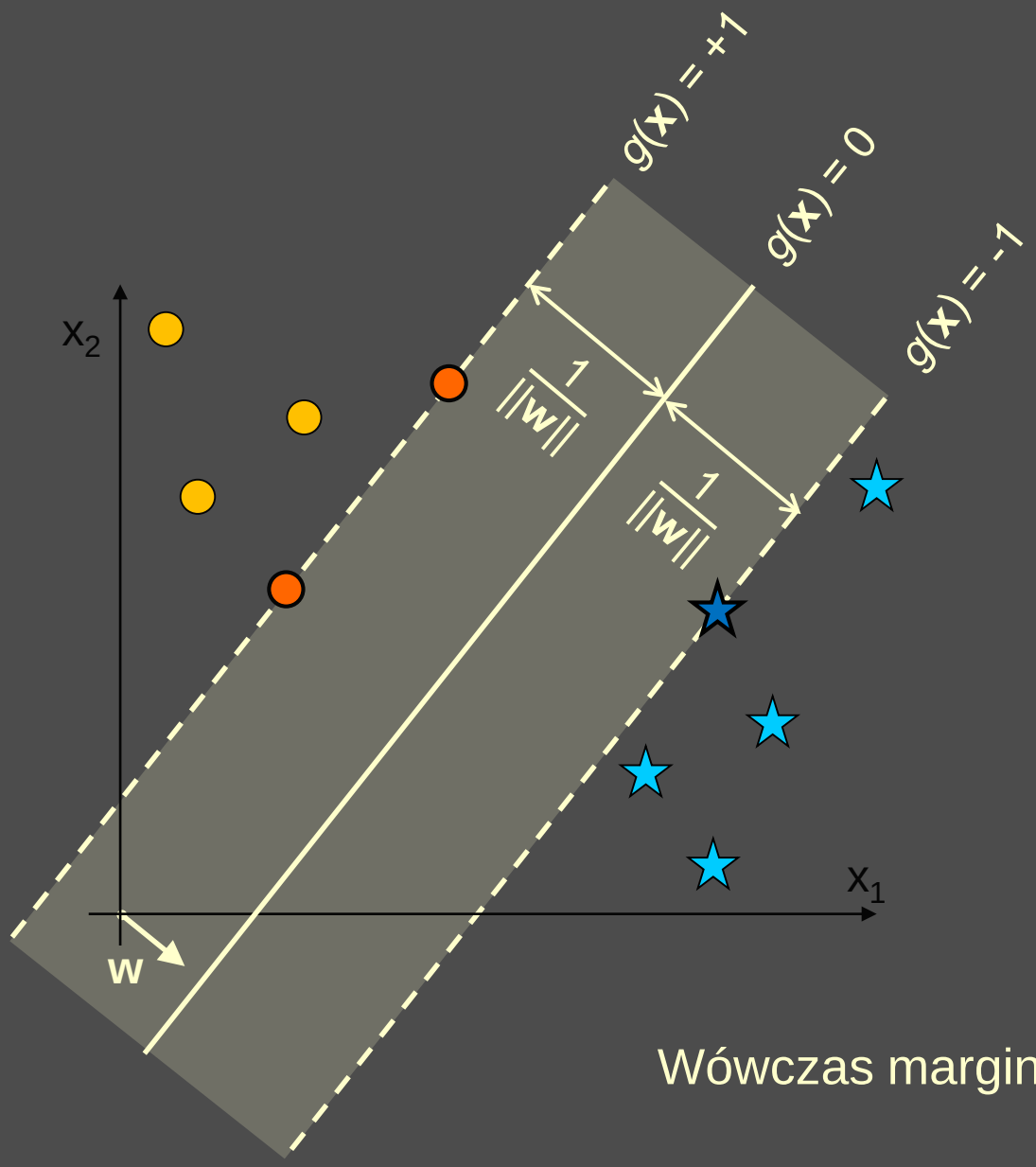
- Wniosek: im dalej obiekty ze zbioru uczącego znajdują się od powierzchni rozdzielającej, tym lepiej z punktu widzenia zdolności klasyfikatora do uogólniania
- Należy zatem wybierać taką powierzchnię rozdzielającą, która leży najdalej od najbliższych obiektów ze zbioru uczącego



- Dokładniej, szukamy powierzchni rozdzielającej, która maksymalizuje tzw. margines między klasami



- Optymalna powierzchnia rozdzielająca prowadzi do modelu o najmniejszym wymiarze VC
- Do jej określenia wystarczą wektory wspierające, pozostała część zbioru uczącego nie wpływa na wynik



Chcemy tak dobrać parametry klasyfikatora, aby dla wektorów wspierających zachodziło $g(\mathbf{x}) = const.$ dla jednej klasy i $g(\mathbf{x}) = -const.$ dla drugiej klasy. Najprościej przyjąć stałą równą 1.

Wówczas margines jest równy $\frac{2}{\|\mathbf{w}\|}$.

- Maksymalizacja marginesu $\frac{2}{\|\mathbf{w}\|}$ sprowadza się do minimalizacji $\|\mathbf{w}\| = \mathbf{w}^T\mathbf{w}$
- Zbiór uczący: $T = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, gdzie $y_n \in \{-1, +1\}$
- Ograniczenie: w obrębie marginesu nie może się znaleźć żaden obiekt ze zbioru uczącego:

$$g(\mathbf{x}_n) \geq +1 \text{ dla } y_n = +1$$

$$g(\mathbf{x}_n) \leq -1 \text{ dla } y_n = -1$$

co można zapisać zwięźle jako $y_n \cdot g(\mathbf{x}_n) \geq 1$

(równość zachodzi tylko dla wektorów wspierających)

Zadanie optymalizacji

- Zmienne decyzyjne: $\mathbf{w}$, w_0
- Funkcja celu: $\frac{1}{2}\,\mathbf{w}^T\mathbf{w}$
- Ograniczenia: $y_n \cdot (\mathbf{w}^T\mathbf{x}_n + w_0) - 1 \geq 0\ ,\ \ n = 1,2,\ldots,N$
- Szukane: $\min\limits_{\mathbf{w},w_0} \frac{1}{2}\,\mathbf{w}^T\mathbf{w}$

Rozwiązanie

- Funkcja Lagrange'a

$$L(\mathbf{w}, w_0, \boldsymbol{\lambda}) = \frac{1}{2}\,\mathbf{w}^T\mathbf{w} - \sum_{n=1}^{N} \lambda_n \left[\, y_n \cdot (\mathbf{w}^T\mathbf{x}_n + w_0) - 1 \,\right]$$

Warunki Kuhna-Tuckera

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{n=1}^{N} \lambda_n y_n \mathbf{x}_n = \mathbf{0}_d$$

$$\frac{\partial L}{\partial w_0} = -\sum_{n=1}^{N} \lambda_n y_n = 0$$

$$\Longrightarrow$$

$$\mathbf{w} = \sum_{n=1}^{N} \lambda_n y_n \mathbf{x}_n$$

$$\sum_{n=1}^{N} \lambda_n y_n = 0$$

(*)

$$y_n \cdot (\mathbf{w}^T \mathbf{x}_n + w_0) - 1 \geq 0$$

$$\lambda_n [\, y_n \cdot (\mathbf{w}^T \mathbf{x}_n + w_0) - 1 \,] = 0$$

$$\lambda_n \geq 0$$

$n = 1,2,\ldots,N$

Dla ograniczeń nieaktywnych $\lambda_n = 0$. Dla ograniczeń aktywnych $\lambda_n > 0$.

Ograniczenia są aktywne tylko dla wektorów wspierających.

$n \in SV$, gdzie SV oznacza zbiór indeksów wektorów wspierających

Gdybyśmy znali wartości mnożników Lagrange'a λ_n dla wektorów wspierających, można uzyskać rozwiązanie $\mathbf{w}$ z wzoru:

$$\mathbf{w} = \sum_{n \in SV} \lambda_n y_n \mathbf{x}_n ,$$

gdzie *SV* oznacza zbiór indeksów wektorów wspierających.

Rozwiązanie w_0 można uzyskać z dowolnego wektora wspierającego:

$$\lambda_n [y_n \cdot (\mathbf{w}^T \mathbf{x}_n + w_0) - 1] = 0$$

$$y_n \cdot (\mathbf{w}^T \mathbf{x}_n + w_0) = 1 \quad \text{(zauważmy, że } y_n = 1 / y_n \text{)}$$

$$w_0 = y_n - \mathbf{w}^T \mathbf{x}_n$$

W celu uzyskania większej stabilności numerycznej, można uśrednić w_0 po wszystkich wektorach wspierających

- Jak najprościej wyznaczyć wartości mnożników Lagrange'a?
- Problem dualny
  - Wstawiamy równania (*) do funkcji Lagrange'a, otrzymując:

$$L_D(\mathbf{w}, w_0, \boldsymbol{\lambda}) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{n=1}^{N} \lambda_n \left[y_n \cdot (\mathbf{w}^T\mathbf{x}_n + w_0) - 1 \right] =$$

$$= \frac{1}{2}\mathbf{w}^T \sum_{n=1}^{N} \lambda_n y_n \mathbf{x}_n - \sum_{n=1}^{N} \lambda_n y_n \mathbf{w}^T\mathbf{x}_n - \sum_{n=1}^{N} \lambda_n y_n w_0 + \sum_{n=1}^{N} \lambda_n =$$

$$= \frac{1}{2}\sum_{n=1}^{N} \lambda_n y_n \mathbf{w}^T\mathbf{x}_n - \sum_{n=1}^{N} \lambda_n y_n \mathbf{w}^T\mathbf{x}_n - w_0 \sum_{n=1}^{N} \lambda_n y_n + \sum_{n=1}^{N} \lambda_n =$$

$$= -\frac{1}{2}\sum_{n=1}^{N} \lambda_n y_n \mathbf{w}^T\mathbf{x}_n - 0 + \sum_{n=1}^{N} \lambda_n = \sum_{n=1}^{N} \lambda_n - \sum_{n=1}^{N} \lambda_n y_n \left(\sum_{m=1}^{N} \lambda_m y_m \mathbf{x}_m\right)^T \mathbf{x}_n$$

$$\boxed{L_D(\boldsymbol{\lambda}) = \sum_{n=1}^{N} \lambda_n - \sum_{n=1}^{N}\sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m \mathbf{x}_n{}^T \mathbf{x}_m}$$

- Problem dualny ma postać
  - Zmienne decyzyjne: $\boldsymbol{\lambda}$
  - Funkcja celu: $L_D(\boldsymbol{\lambda}) = \sum_{n=1}^{N} \lambda_n - \sum_{n=1}^{N}\sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m \mathbf{x}_n^T \mathbf{x}_m$
  - Ograniczenia:

$$\sum_{n=1}^{N} \lambda_n y_n = 0 \, , \, \lambda_n \geq 0$$

  - Wyznaczanie mnożników Lagrange'a λ_n : pakiety obliczeń numerycznych dostarczają wielu szybkich metod optymalizacji funkcji kwadratowej

Klasy nie są liniowo separowalne

Przeformułowanie ograniczeń:

$g(\mathbf{x}_n) \geq +1 - \xi_n$ dla $y_n = +1$

$g(\mathbf{x}_n) \leq -1 + \xi_n$ dla $y_n = -1$

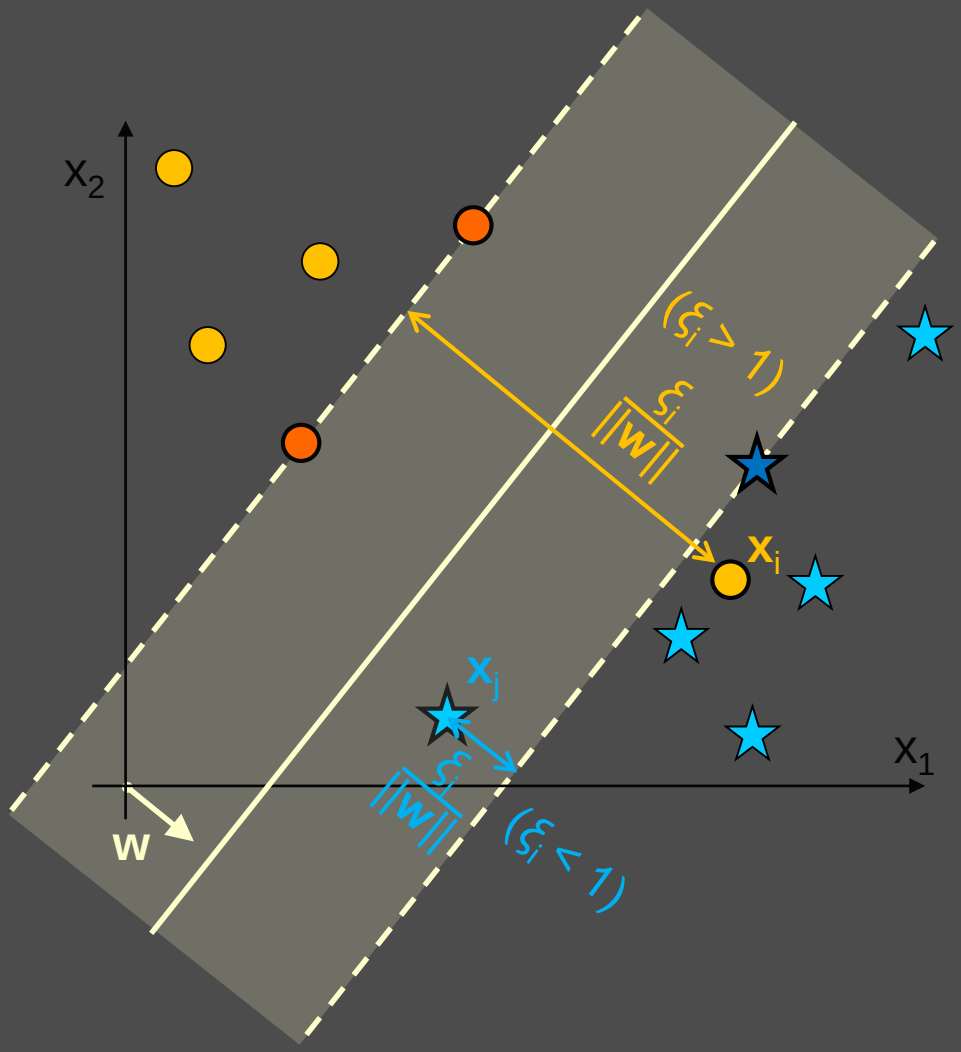
co można zapisać zwięźle jako

$$y_n \cdot g(\mathbf{x}_n) \geq 1 - \xi_n$$

Minimalizowana jest funkcja

$$\frac{1}{2}\,\mathbf{w}^T\mathbf{w} + C\sum_{n=1}^{N}\xi_n$$

gdzie C jest tzw. parametrem regularyzacji



- Wyrażenia (*) nie ulegają zmianie
- Prymalna postać funkcji Lagrange'a jest inna, ale dualna postać jest identyczna, przy czym pojawia się dodatkowe ograniczenie

$$0 \leq \lambda_n \leq C$$

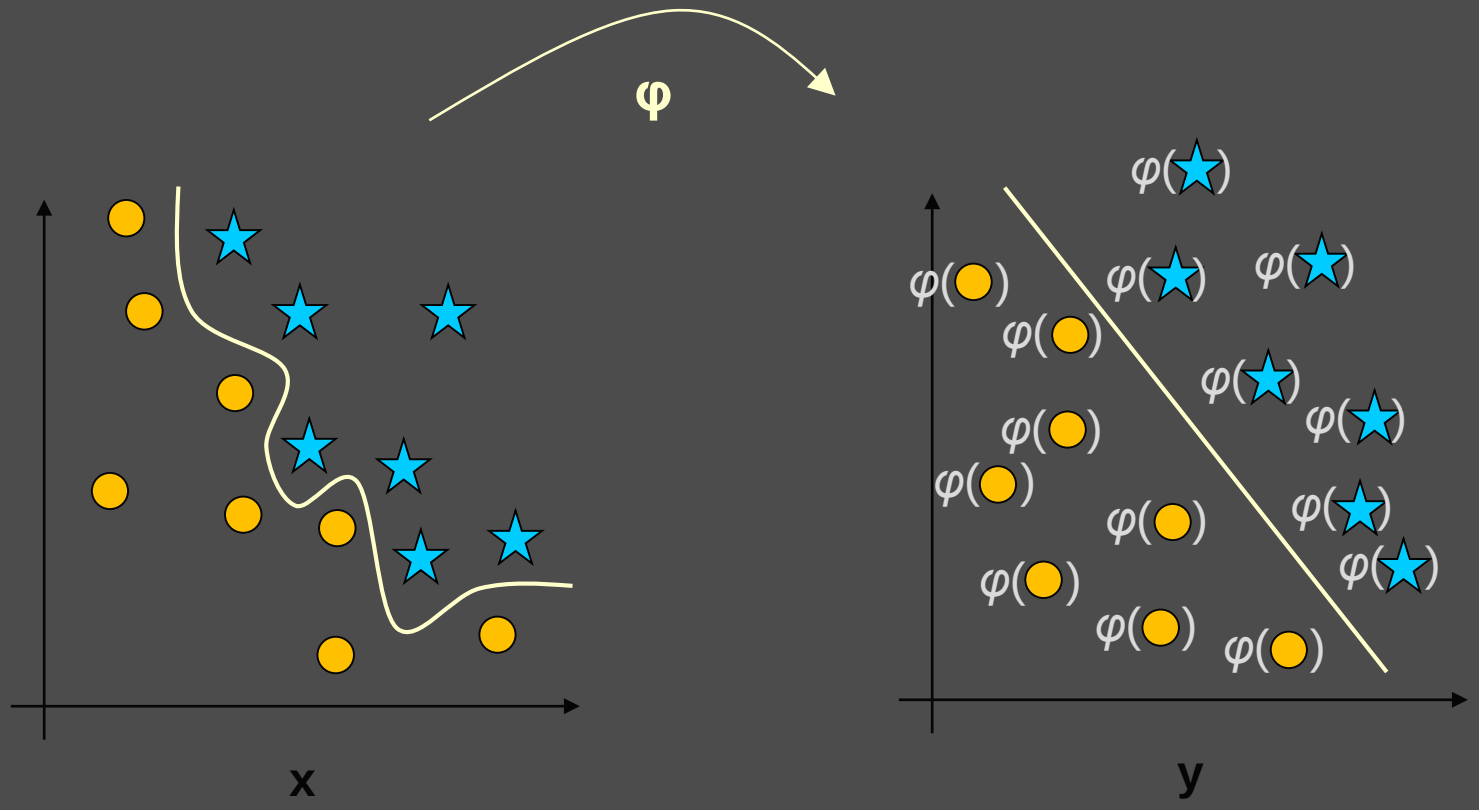
- Rozwiązanie **w** nie ulega zmianie
- Rozwiązanie w_0 wyznaczamy – podobnie jak poprzednio – z dowolnego wektora wspierającego lub uśredniając po wszystkich wektorach wspierających

Nieliniowe SVM

- Klasy nie są liniowo separowalne
- Można utworzyć SVM w transformowanej przestrzeni, w której klasy dadzą się liniowo odseparować
- Punkty **x** oryginalnej przestrzeni przekształcamy nieliniową funkcją $\varphi(\mathbf{x})$
- Dualna postać funkcji Lagrange'a:

$$L_D(\boldsymbol{\lambda}) = \sum_{n=1}^{N} \lambda_n - \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n \lambda_m y_n y_m \varphi^T(\mathbf{x}_n) \varphi(\mathbf{x}_m)$$

- Nieliniowe przekształcenie **φ** – interpretacja geometryczna



- Funkcja decyzyjna ma postać:

$$g(\mathbf{x}) = \sum_{n \in SV} \lambda_n y_n \varphi^T(\mathbf{x}_n) \varphi(\mathbf{x}) + w_0$$

- Wygodnie jest wprowadzić funkcję jądrową (*kernel function*), która zastąpi operację iloczynu skalarnego $\varphi^T(\mathbf{x}_n)\, \varphi(\mathbf{x}_m)$:

$$K(\mathbf{x}_n, \mathbf{x}_m) = \varphi^T(\mathbf{x}_n)\, \varphi(\mathbf{x}_m)$$

- Funkcja jądrowa musi spełniać *kryterium Mercera*

- Typowe funkcje jądrowe stosowane w klasyfikatorach SVM

  - wielomianowa: $(1+ \mathbf{x}_n^T \mathbf{x}_m)^d$

  - gaussa: $\exp\left[- |\mathbf{x}_n - \mathbf{x}_m|^2 / \sigma^2 \right]$

  - sigmoida: $\tanh(k\mathbf{x}_n^T \mathbf{x}_m - \delta)$

Naiwny klasyfikator bayesowski

- Zakłada niezależność warunkowych rozkładów klas
- Probabilistyczny model zadania rozpoznawania

$\mathbf{x} = [\, x_1\, x_2\, \ldots\, x_d\,]^T$ – wynik pomiaru d cech obiektu

$\{\omega_1, \omega_2, \ldots, \omega_c\}$ – zbiór klas, do których może należeć $\mathbf{x}$

$T = \{\, (\mathbf{x}_n, \omega_j);\quad n = 1,\ldots,N,\quad j \in \{1,..,c\}\,\}$ – zbiór uczący

$P(\omega_i|\mathbf{x})$ – prawdopodobieństwo, że obiekt o cechach **x** należy do klasy ω_i

$P(\omega_i)$ – prawdopodobieństwo a'priori klasy ω_i

$P(\mathbf{x})$ – prawdopodobieństwo a'priori cech **x**

$P(\mathbf{x}|\omega_i)$ – prawdopodobieństwo, że obiekt z klasy ω_i będzie mieć cechy **x**

- rozkłady $P(\omega_i)$, $P(\mathbf{x})$ i $P(\mathbf{x}|\omega_i)$ można estymować ze zbioru uczącego
- rozkład a'posteriori $P(\omega_i|\mathbf{x})$ należy wyznaczyć w oparciu o $P(\omega_i)$, $P(\mathbf{x})$ i $P(\mathbf{x}|\omega_i)$

Wzór Bayesa: $$P(\omega_i|\mathbf{x}) = \frac{P(\mathbf{x}|\omega_i) \cdot P(\omega_i)}{P(\mathbf{x})}$$

Jak działa **klasyfikator**?

Reguła decyzyjna:

Obiekt o cechach **x** jest przyporządkowywany do klasy ω_j wtedy i tylko wtedy, gdy

$$P(\omega_j|\mathbf{x}) > P(\omega_i|\mathbf{x}) \quad \text{dla } i = 1,\ldots,c;\ i \neq j$$

a zatem poszukujemy klasy ω_j, która maksymalizuje $P(\omega_j|\mathbf{x})$.

Wstawiając wzór Bayesa otrzymujemy regułę:

$$\frac{P(\mathbf{x}|\omega_j) \cdot P(\omega_j)}{P(\mathbf{x})} > \frac{P(\mathbf{x}|\omega_i) \cdot P(\omega_i)}{P(\mathbf{x})}$$

Uwzględniając, że $P(\mathbf{x})$ nie zależy od klasy, reguła upraszcza się do:

$$P(\mathbf{x}|\omega_j) \cdot P(\omega_j) > P(\mathbf{x}|\omega_i) \cdot P(\omega_i)$$

a zatem poszukujemy klasy ω_j , która maksymalizuje iloczyn

$$P(\mathbf{x}|\omega_j) \cdot P(\omega_j) .$$

- Jak **estymować rozkłady** a'priori i warunkowe?
  - Rozkład a'priori $P(\omega_i)$:

$$P(\omega_i) \leftarrow \frac{k_i}{N}$$

gdzie k_i jest liczbą obiektów klasy ω_i w zbiorze uczącym T

  - Rozkłady warunkowe $P(\mathbf{x}|\omega_i)$:

Wektory **x** mają wymiar d, więc należałoby estymować c d-wymiarowych rozkładów. Aby zmniejszyć złożoność obliczeniową, przyjmuje się „naiwne" założenie o warunkowej niezależności klas:

$$P(\mathbf{x}|\omega_i) \approx \prod_{s=1}^{d} P(x_s|\omega_i)$$

- Zamiast wyznaczać *c* *d*-wymiarowych rozkładów $P(\mathbf{x}|\omega_i)$ wyznaczać będziemy c *1*-wymiarowych rozkładów $P(x_1|\omega_i)$, $P(x_2|\omega_i)$, … , $P(x_d|\omega_i)$, $(i = 1,\ldots,c)$

- Jeżeli cechy $x_s = \gamma$ przyjmują wartości ze skończonego zbioru $\gamma \in \Gamma$

$$P(x_s|\omega_i) \leftarrow \frac{q_i(x_s,\gamma)}{k_i}$$

gdzie $q_i(\gamma)$ jest liczbą obiektów klasy ω_i przyjmujących wartość

$x_s = \gamma$ $(\gamma \in \Gamma)$

- Jeżeli cechy x_s są ciągłe, przyjmuje się rozkłady o parametrach $\boldsymbol{\theta}$:

$$P(x_s|\omega_i) = f_i\,(x_s, \boldsymbol{\theta}),$$

których parametry estymuje się z danych (w pakietach domyślnie przyjmuje się rozkład Gaussa).

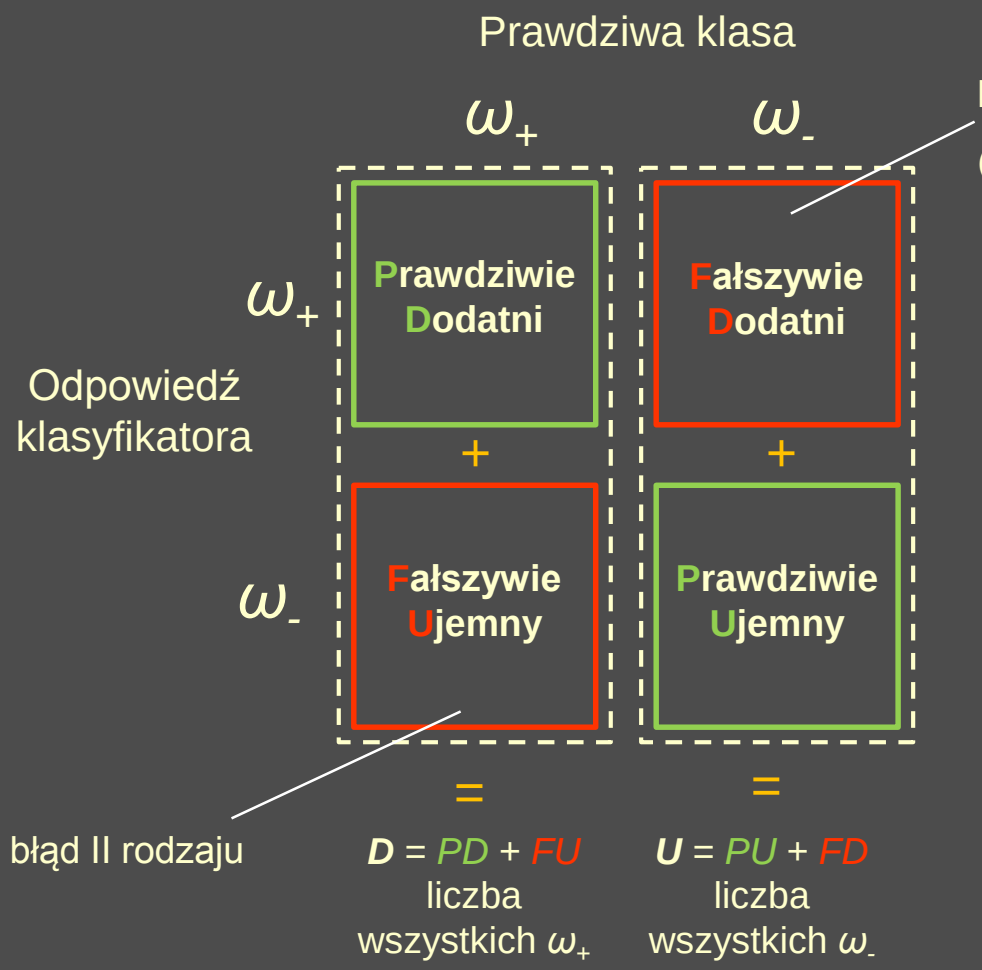
Korekta Laplace'a

- Użycie przybliżenia $P(\mathbf{x}|\omega_i) \approx \prod_{s=1}^{d} P(x_s|\omega_i)$ sprawia problem wtedy, gdy istnieje taka wartość $\gamma_0 \in \Gamma$, że w zbiorze uczącym T, wśród obiektów klasy ω_i, żadna cecha x_s nie przyjmuje wartości γ_0.
- W takim przypadku pewne $P(x_s|\omega_i) = 0$, co skutkuje $P(\mathbf{x}|\omega_i) = 0$ nawet, gdy $P(x_s|\omega_i)$ dla innych cech przyjmują duże wartości
- Rozwiązanie: skorygować estymaty prawdopodobieństw, dodając 1 do każdej $q_i(\gamma)$ (i zwiększając k_i o liczbę dodanych jedynek)
- Jeżeli zbiór uczący jest duży, ta korekta nie zaburzy w istotny sposób pozostałych estymat

Uwagi

- Pomimo założenia o niezależności warunkowych rozkładów klas, naiwny klasyfikator bayesowski w praktycznych zastosowaniach daje zaskakująco dobre rezultaty (niewiele gorsze od sieci neuronowych)
- Działa szybko i dokładnie dla dużych zbiorów danych
- Stanowi dobry punkt odniesienia do oceny innych klasyfikatorów

Jakość klasyfikatora



Liczność PD (*czułość*): *LPD* = *PD* / ***D***

True positive rate (sensitivity)

False positive rate: LFD = *FD* / ***U***

Liczność PU (*swoistość*): *LPU* = *PU* / ***U***

True negative rate (specificity)

LPU = 1 - *LFD*

Dokładność (*accuracy*):

ACC = (*PD* + *PU*) / (***D*** + ***U***)

Przestrzeń ROC

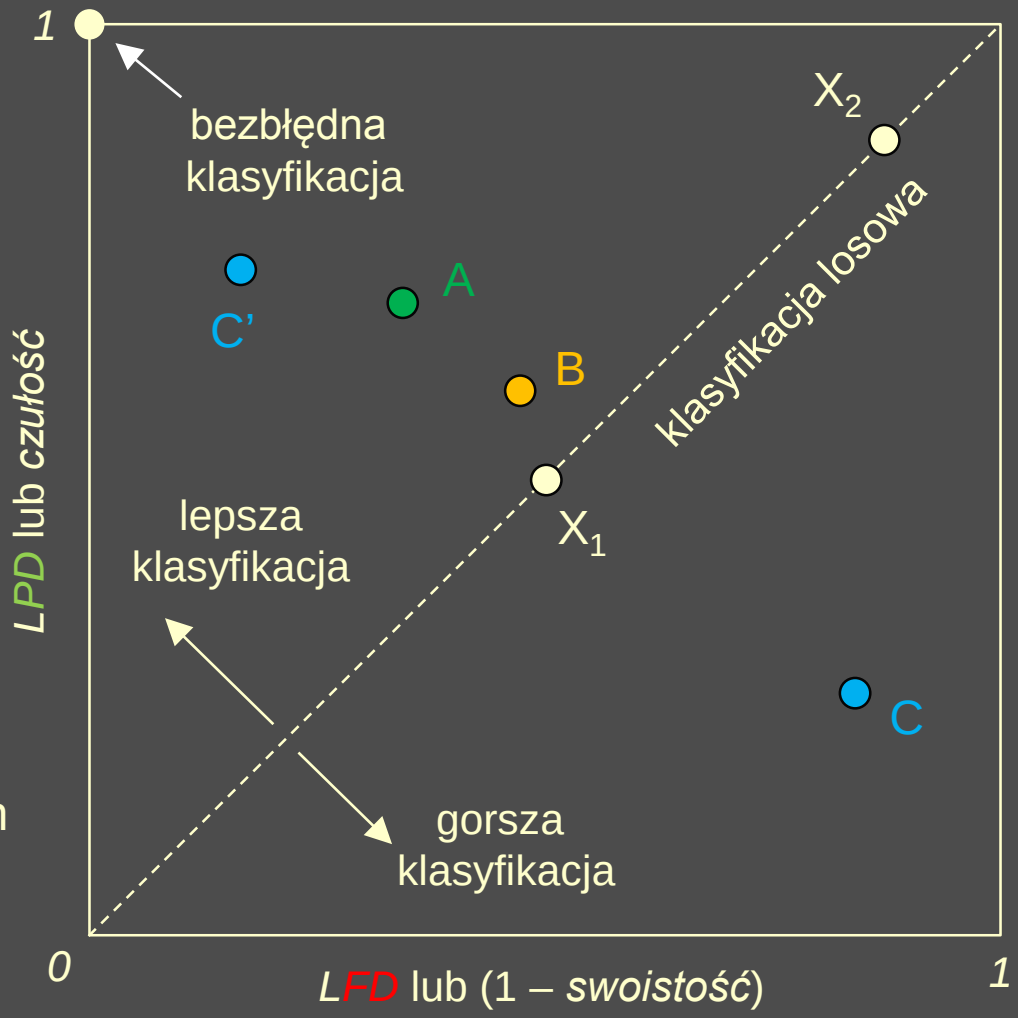
(*Receiver Operating Characteristic*)

Klasyfikator A jest lepszy od B

Klasyfikator C jest najgorszy, ale klasyfikator C' – który daje odpowiedzi przeciwne do C – jest lepszy od A

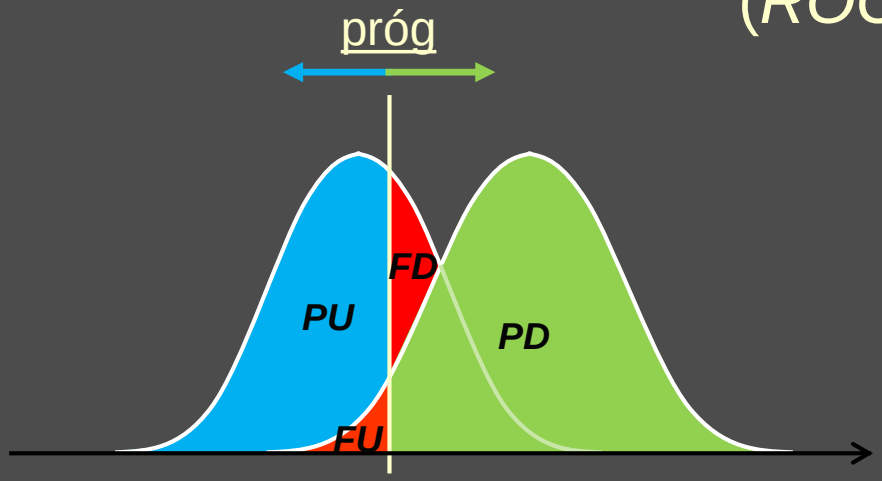
Klasyfikator X_1 losuje obie klasy z jednakowym prawdopodobieństwem

Klasyfikator X_2 losuje jedną z klas z prawdopodobieństwem 0.9

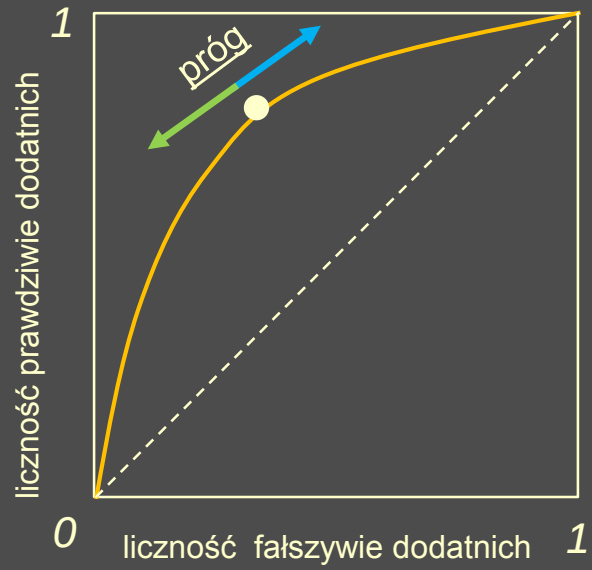


Krzywa ROC

(*ROC curve*)



Metoda rozwijana od II wojny światowej (po ataku na Pearl Harbor w 1941 r.): detekcja japońskich samolotów na podstawie sygnałów z radaru



Prawdziwie Dodatni	**Fałszywie Dodatni**
Fałszywie Ujemny	**Prawdziwie Ujemny**

- Inny przykład wielkości progowej:
  - wyjście klasyfikatora neuronowego jest wielkością ciągłą, z przedziału [0,1], decyzja o klasie zależy od przyjętej wartości progu
- Krzywa ROC jest estymowana z niezależnego zbioru testującego
- Do wyznaczenia krzywej ROC można użyć walidacji krzyżowej lub metody bootstrap
- Krzywa ROC nie zależy od rozkładu a’priori klas

Wybór najlepszego klasyfikatora

$p(\omega_+)$, $p(\omega_-)$ – prawdopodobieństwa a'priori klas

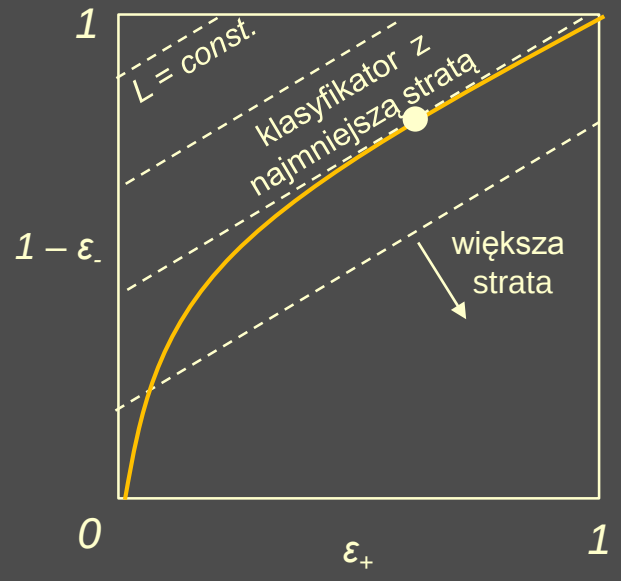
ε_+ (ε_-) – prawdopodobieństwo błędnej klasyfikacji obiektu z klasy ω_+ (ω_-)

$\lambda_{10} = \lambda(\alpha_+|\omega_-)$ $\lambda_{01} = \lambda(\alpha_-|\omega_+)$ – straty wynikające z błędnej klasyfikacji

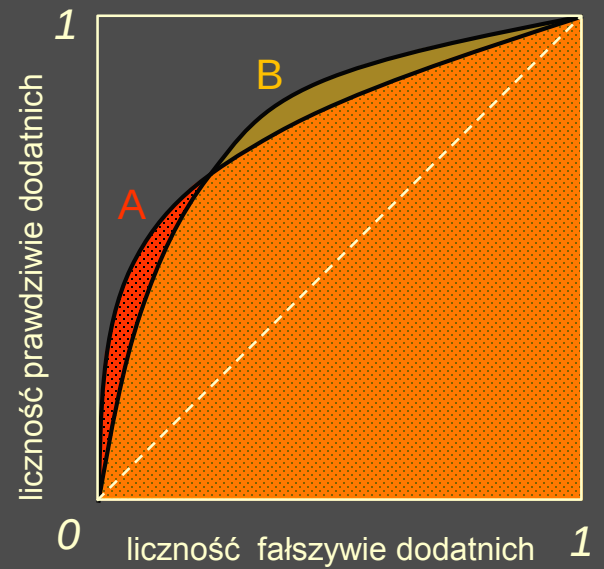
$\lambda_{11} = \lambda_{00} = 0$

Średnia strata

$L = \lambda_{01}\, p(\omega_+)\, \varepsilon_+ + \lambda_{10}\, p(\omega_-)\, \varepsilon_-$



Pole pod krzywą ROC (AUC – *Area Under Curve*)



Teoretycznie $0 \leq AUC \leq 1$

Praktycznie AUC – 0,5

Ważna własność statystyczna:

AUC jest równe prawdopodobieństwu, że klasyfikator przyporządkuje wyższą rangę (np. aktywację neuronu) losowo wybranemu obiektowi klasy ω_+ niż losowo wybranemu obiektowi klasy ω_-

AUC jest miarą jakości klasyfikatora niezależną od funkcji strat

Test McNemara

- Porównujemy dwa klasyfikatory (A i B) w oparciu o niezależny zbiór testujący
- Czy wyznaczona różnica jakości jest dziełem przypadku, czy jest istotna statystycznie?

n_{00} – liczba obiektów nieprawidłowo klasyfikowanych przez A i B

n_{01} – liczba obiektów nieprawidłowo klasyfikowanych tylko przez A

n_{10} – liczba obiektów nieprawidłowo klasyfikowanych tylko przez B

n_{11} – liczba obiektów prawidłowo klasyfikowanych przez A i B

$$z = \frac{|n_{01} - n_{10}| - 1}{\sqrt{n_{01} + n_{10}}}$$

Wielkość z^2 ma w przybliżeniu rozkład χ^2 z jednym stopniem swobody. Hipoteza zerowa (że klasyfikatory A i B mają tę samą jakość) może być odrzucona na poziomie istotności 0.05 jeżeli |z| > 1.96

Popularne pakiety

- WEKA
- Matlab (*Statistical Pattern Recognition Toolbox*)
- Statistica
- R