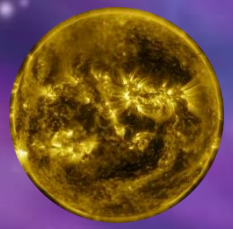
METODY INŻYNIERII WIEDZY

KNOWLEDGE ENGINEERING AND DATA MINING



Maszyna Wektorów Nośnych

Support Vector Machine
SVM

Adrian Horzyk

Akademia Górniczo-Hutnicza

Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej

Katedra Automatyki i Inżynierii Biomedycznej, Laboratorium Biocybernetyki

30-059 Kraków, al. Mickiewicza 30, paw. C3/205

horzyk@agh.edu.pl, Google: Adrian Horzyk

IDEA I NARODZINY SVM

Prof. V. Vapnik w 1998 r. stworzył nowe podejście do kształtowania struktury sieci neuronowej oraz definiowania problemu uczenia próbując wyeliminować znane wady sieci neuronowych typu MLP i RBF stosujące minimalizację nieliniowych funkcji błędu, tj.:

- **Minimalizowana funkcja jest zwykle wielomodalna względem optymalizowanych parametrów i posiada wiele minimów lokalnych, w których proces uczenia może utknąć w zależności od punktu startowego, których zwykle istnieje nieskończona ilość.**
- **Algorytm uczący zwykle nie jest w stanie skutecznie kontrolować złożoności struktury sieci neuronowej, co w istotny sposób wpływa na zdolności uogólniające sieci.**

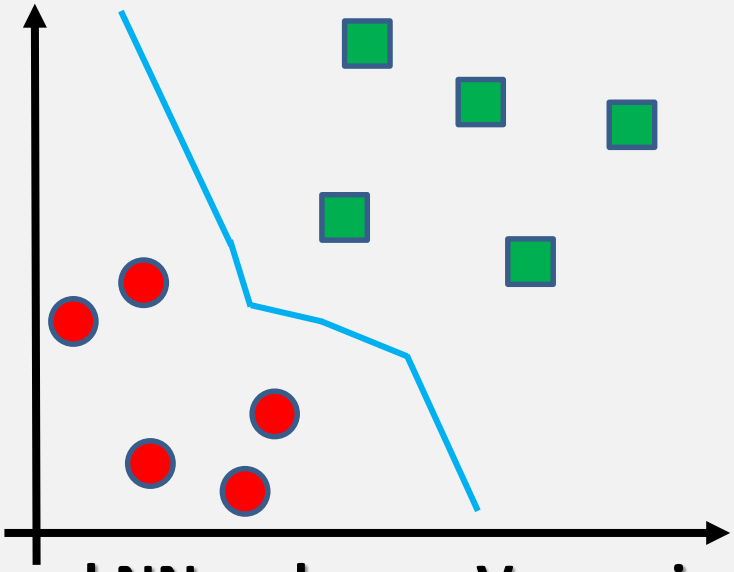
Istotą zmiany jest przedstawienie procesu uczenia jako procesu dobierania wag, w którym maksymalizowany jest margines separacji oddzielający skrajne (najbliższe) punkty w przestrzeni danych definiujących różne klasy.

Bierzemy więc pod uwagę tylko te najtrudniej separowalne punkty przestrzeni przy budowie modelu, które określają tzw. wektory nośne (wspierające).

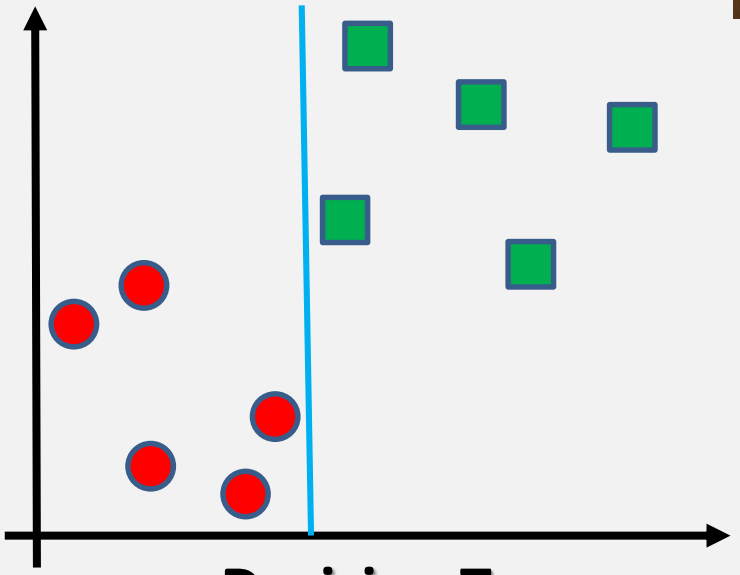
Sieci SVM tworzą specyficzną dwuwarstwową strukturę neuropodobną stosującą różne rodzaje funkcji aktywacji (liniowe, wielomianowe, radialne, sigmoidalne) oraz specyficzny sposób uczenia oparty na programowaniu kwadratowym, które charakteryzuje się istnieniem tylko jednego minimum globalnego.

Sieci SVM dedykowane są głównie do zagadnień klasyfikacji, w których jedną klasę separujemy możliwie dużym marginesem od pozostałych klas.

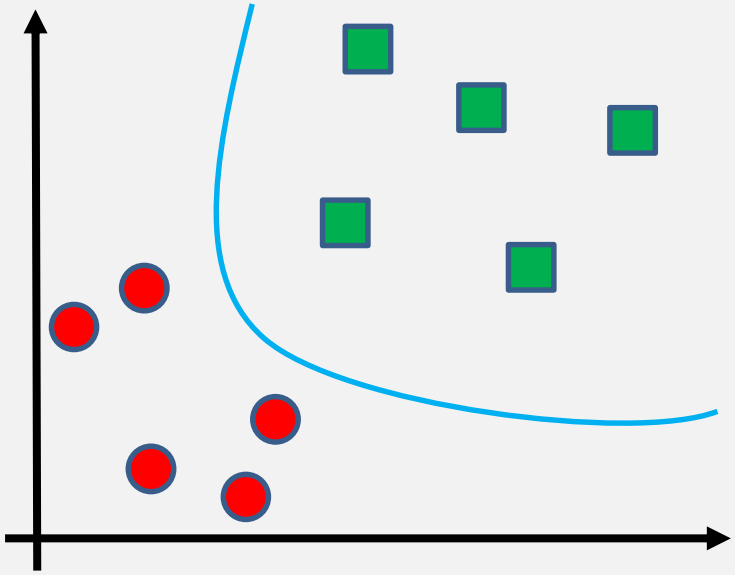
DYSKRYMINACJA i KLASYFIKACJA



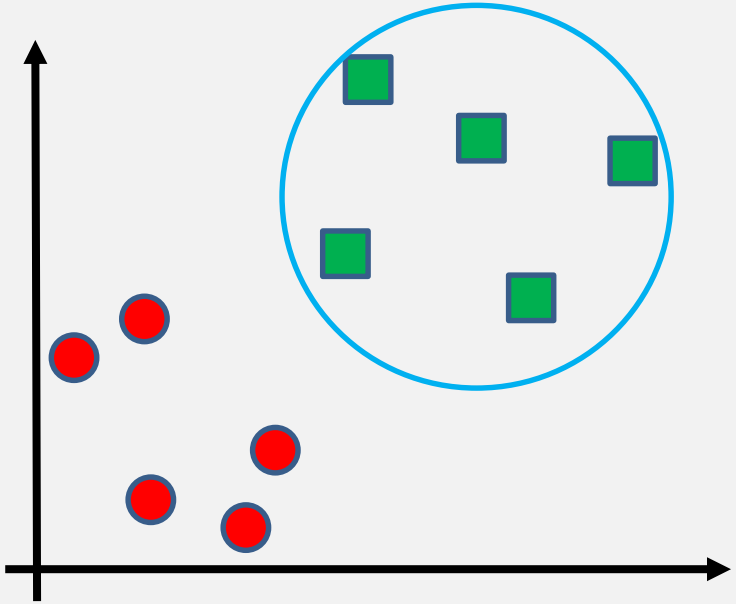
kNN – obszary Voronoi



Decision Tree



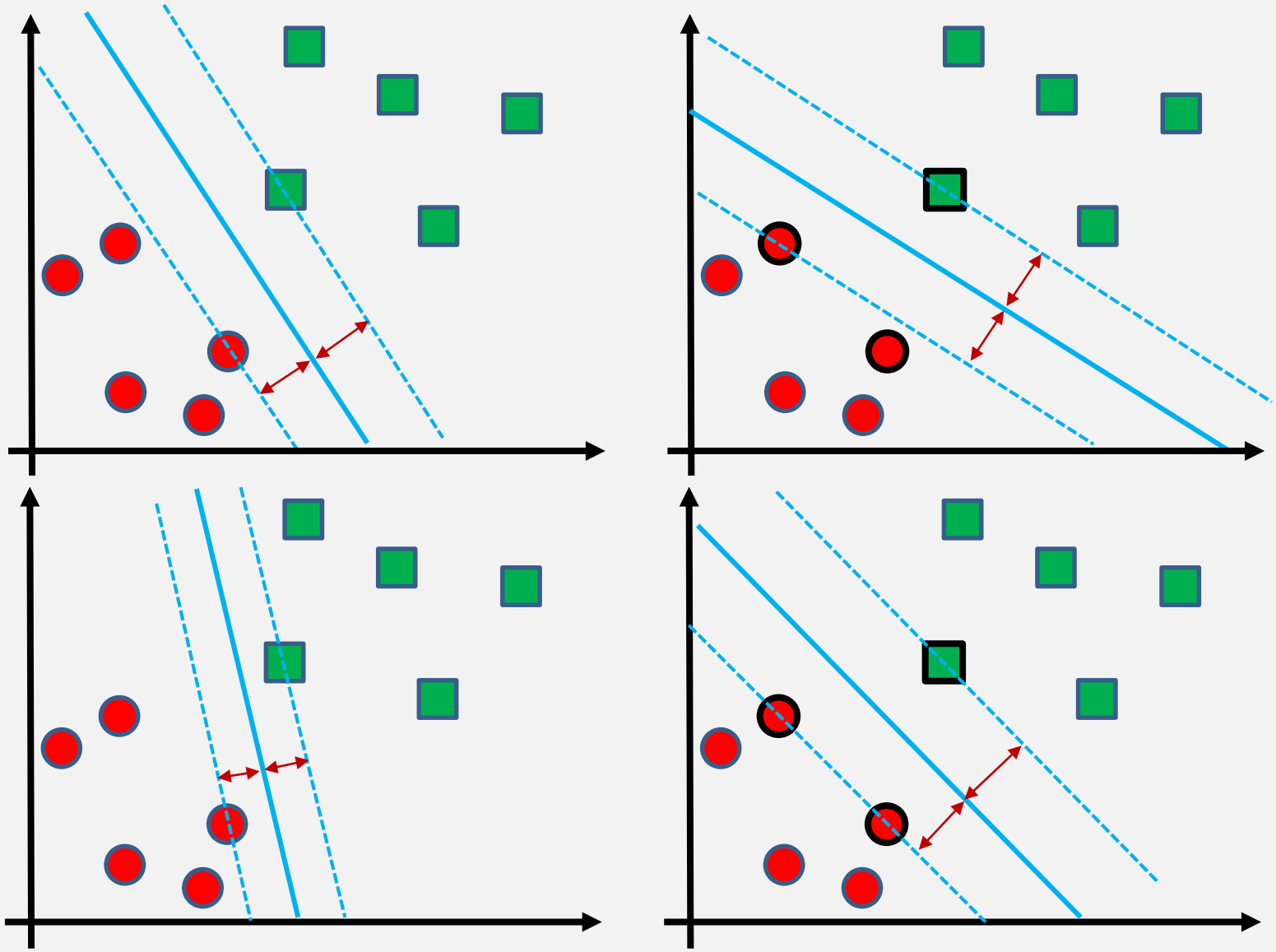
MLP



RBF

Najszersza granica dyskryminacji

Metoda SVM ma na celu wyznaczyć najszerszą granicę dyskryminacji spośród możliwych, których zwykle istnieje nieskończona ilość:



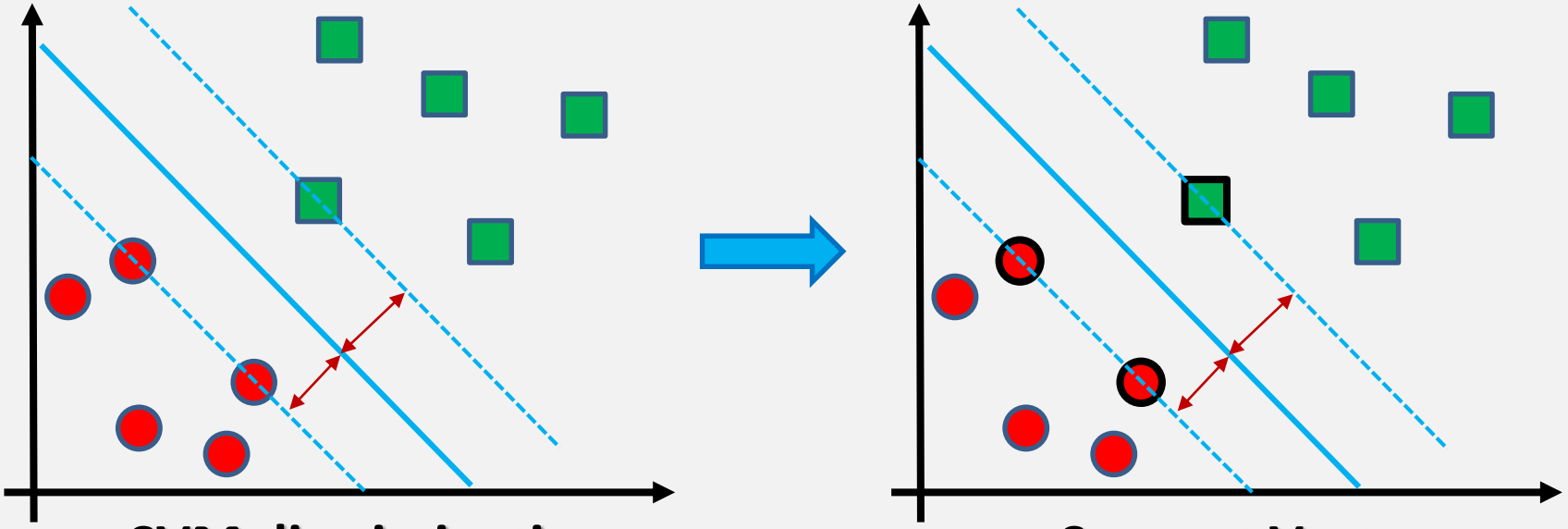
Jak dyskryminować i separować najlepiej?

Próbując osiągnąć jak najlepszą dyskryminację wzorców poszczególnych klas warto zmaksymalizować margines oddzielający wzorce poszczególnych klas.

Ponadto mając do czynienia z wieloma danymi, można ograniczyć analizę tylko do tych najtrudniejszych punktów przestrzeni, czyli wzorców różnych klas, które leżą najbliżej, gdyż je najtrudniej zdyskryminować (odseparować).

Model uwzględniający najtrudniejsze wzorce powinien charakteryzować się dobrą jakością oraz prostotą reprezentacji.

Spróbujemy więc wyznaczyć optymalną hiperpłaszczyznę dyskryminującą wzorce jednej klasy (kwadratów) od pozostałych (tutaj kółeczek).



Maszyna Wektorów Nośnych

Załóżmy, że mamy zbiór p par uczących:

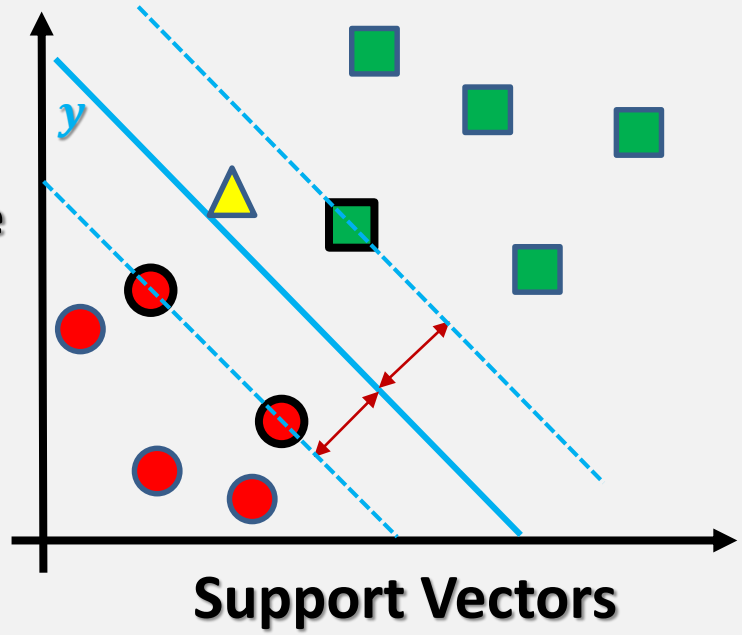
(x_i, d_i) dla $i = 1, 2, \ldots, p$

gdzie x_i – wektor danych wejściowych

$d_i \in \{-1; +1\}$ – reprezentuje dyskryminowane klasy: $d_i = +1$ oznacza klasę dyskryminowaną, zaś $d_i = -1$ oznacza pozostałe klasy.

Przy założeniu liniowej separowalności obu klas możliwe jest określenie równania hiperpłaszczyzny separującej te wzorce:

$y(x) = w^T x + b = 0$



gdzie w – wektor wag, a x – wektor danych wejściowych, b – polaryzacja

Możemy więc zdefiniować równania decyzyjne:

Jeżeli $w^T x + b \geq 0$ wtedy $d_i = +1$

Jeżeli $w^T x + b \leq 0$ wtedy $d_i = -1$

Co możemy zapisać w postaci nierówności: $d_i(w^T x + b) \geq 1$, której spełnienie przez pary punktów (x_i, d_i) definiuje wektory nośne (*support vectors*), które decydują o położeniu hiperpłaszczyzny i szerokości marginesu separacji.

Potrzebne jest więc wyznaczenie b oraz w, żeby określić decyzję.

Przekroczenie granic separacji

Czasami jednak występuje konieczność zmniejszenia marginesu separacji dla problemów niecałkowicie separowalnych liniowo oraz pewnych punktów (x_i, d_i) leżących wewnątrz strefy marginesu separacji, co możemy zapisać za pomocą nierówności:

$$d_i\left(w^T x_i + b\right) \geq 1 - \delta_i$$

gdzie $\delta_i \geq 0$ i zmniejsza margines separacji, przy czym jeśli:

$0 \leq \delta_i < 1$ – wtedy (x_i, d_i) leży po właściwej stronie hiperpłaszczyzny, więc decyzja o przynależności do klasy będzie poprawna,

$\delta_i = 1$ – wtedy (x_i, d_i) leży na hiperpłaszczyźnie, więc decyzja o przynależności do klasy będzie nieokreślona,

$1 < \delta_i$ – wtedy (x_i, d_i) leży po niewłaściwej stronie hiperpłaszczyzny, więc decyzja o przynależności do klasy będzie błędna.

Określając granicę decyzyjną należy więc możliwie zminimalizować wartość δ_i.

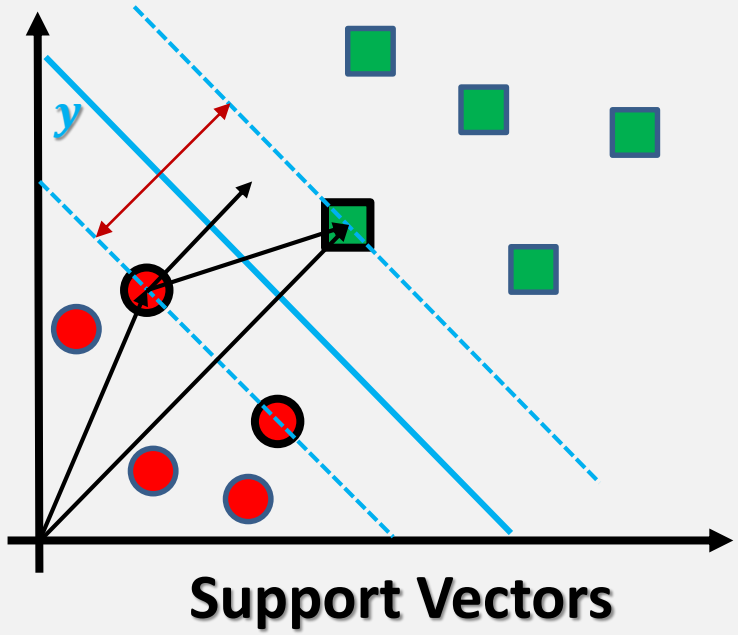
Szerokość marginesu separacji

Szerokość marginesu separacji możemy wyznaczyć jako iloczyn kartezjański wektora wag oraz różnicy odległości dwóch wektorów nośnych należących do przeciwnych klas:

$$\rho = (x^+ - x^-) \cdot \frac{w}{\|w\|} = \frac{2}{\|w\|} = 2 \cdot r(x_{SV})$$

gdyż odległość wektorów nośnych od hiperpłaszczyzny określona jest następująco:

$$r(x_{SV}) = \frac{y(x_{SV})}{\|w\|} = \begin{cases} \frac{1}{\|w\|} & dla\ \ y(x_{SV}) = 1 \\ \frac{-1}{\|w\|} & dla\ \ y(x_{SV}) = -1 \end{cases}$$



Chcąc więc zmaksymalizować margines separacji pomiędzy wektorami nośnymi różnych klas $\rho = \frac{2}{\|w\|}$ trzeba zminimalizować $\|w\|$, co jest równoważne minimalizacji wyrażenia $\frac{1}{2}\|w\|^2$ przy pewnych ograniczeniach liniowych wynikających ze zdefiniowanej nierówności decyzyjnej.

W takich przypadkach stosujemy mnożniki Lagrange'a i minimalizujemy funkcję Lagrange'a.

Minimalizacja funkcji Lagrange'a

Możemy więc teraz określić funkcję Lagrange'a dla problemu maksymalizacji marginesu separacji:

$$\min_{w} \frac{1}{2}\|w\|^2 + \vartheta \sum_{i=1}^{p} \delta_i$$

przy zdefiniowanych ograniczeniach:

$$d_i\left(w^T x_i + b\right) \geq 1 - \delta_i$$
$$\delta_i \geq 0$$

gdzie ϑ – to waga, z jaką traktowane są błędy testowania w stosunku do marginesu separacji, decydującą o złożoności sieci neuronowej, dobieraną przez użytkownika w sposób eksperymentalny, np. metodą walidacji krzyżowej.

Otrzymujemy więc następującą funkcję Lagrange'a:

$$L(w, b, \alpha, \delta, \mu) = \frac{1}{2} w^T w + \vartheta \sum_{i=1}^{p} \delta_i - \sum_{i=1}^{p} \alpha_i \left[d_i\left(w^T x_i + b\right) - (1 - \delta_i) \right] - \sum_{i=1}^{p} \mu_i \delta_i$$

gdzie α_i jest wektorem mnożników Lagrange'a o wartościach nieujemnych odpowiadającym poszczególnym ograniczeniom funkcyjnym, a μ_i ograniczeniom nierównościowym nakładanym na zmienne δ_i.

Rozwiązanie minimalizacji funkcji Lagrange'a polega na określeniu punktu siodłowego, czyli wyznaczenia pochodnych cząstkowych względem mnożników.

Minimalizacja funkcji Lagrange'a

Warunki optymalnego rozwiązania wyznaczone są zależnościami:

$$\frac{\partial L(w,b,\alpha,\delta,\mu)}{\partial w} = 0 \quad \rightarrow \quad w = \sum_{i=1}^{p} \alpha_i d_i x_i$$

$$\frac{\partial L(w,b,\alpha,\delta,\mu)}{\partial b} = 0 \quad \rightarrow \quad \sum_{i=1}^{p} \alpha_i d_i = 0$$

$$\frac{\partial L(w,b,\alpha,\delta,\mu)}{\partial w} = 0 \quad \rightarrow \quad \mu_i = \vartheta - \alpha_i$$

które podstawimy teraz do funkcji Lagrange'a:

$$L(w,b,\alpha,\delta,\mu) = \frac{1}{2} w^T w + \vartheta \sum_{i=1}^{p} \delta_i - \sum_{i=1}^{p} \alpha_i \left[d_i \left(w^T x_i + b \right) - (1 - \delta_i) \right] - \sum_{i=1}^{p} \mu_i \delta_i$$

$$= \frac{1}{2} \sum_{i=1}^{p} \alpha_i d_i x_i \sum_{j=1}^{p} \alpha_j d_j x_j + \vartheta \sum_{i=1}^{p} \delta_i - \sum_{i=1}^{p} \alpha_i \left[d_i \left(\sum_{j=1}^{p} \alpha_j d_j x_j \, x_i + b \right) - (1 - \delta_i) \right] - \sum_{i=1}^{p} \mu_i \delta_i$$

$$= \frac{1}{2} \sum_{i=1}^{p} \alpha_i d_i x_i \sum_{j=1}^{p} \alpha_j d_j x_j + \vartheta \sum_{i=1}^{p} \delta_i - \sum_{i=1}^{p} \alpha_i d_i x_i \sum_{j=1}^{p} \alpha_j d_j x_j + b \sum_{i=1}^{p} \alpha_i d_i + \sum_{i=1}^{p} \alpha_i (1 - \delta_i) - \sum_{i=1}^{p} (\vartheta - \alpha_i) \delta_i$$

$$= \frac{1}{2} \sum_{i=1}^{p} \alpha_i d_i x_i \sum_{j=1}^{p} \alpha_j d_j x_j + \vartheta \sum_{i=1}^{p} \delta_i - \sum_{i=1}^{p} \alpha_i d_i x_i \sum_{j=1}^{p} \alpha_j d_j x_j + b \sum_{i=1}^{p} \alpha_i d_i + \sum_{i=1}^{p} \alpha_i - \sum_{i=1}^{p} \alpha_i \delta_i - \vartheta \sum_{i=1}^{p} \delta_i$$

$$+ \sum_{i=1}^{p} \alpha_i \delta_i = \sum_{i=1}^{p} \alpha_i - \frac{1}{2} \sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_j d_i d_j x_i x_j$$

Problem dualny

W punkcie siodłowym iloraz mnożnika Lagrange'a d_{SV} i odpowiedniego ograniczenia związanego δ_{SV} z wektorem nośnym x_{SV} jest równy zeru ($d_{SV}\delta_{SV} = 0$), gdyż δ_{SV}=0, więc zależność:

$$d_i\left(w^T x_i + b\right) \geq 1 - \delta_i$$

w punkcie wektora nośnego sprowadza się do:

$$w^T x_i + b = \pm 1$$

co pozwala wyznaczyć wartość b :

$$b = \pm 1 - w^T x_i$$

Otrzymaliśmy więc problem dualny

$$\max_{\alpha} Q(\alpha) = \sum_{i=1}^{p} \alpha_i - \frac{1}{2}\sum_{i=1}^{p}\sum_{j=1}^{p} \alpha_i \alpha_j d_i d_j x_i x_j$$

przy ograniczeniach dla $i = 1, 2, \ldots, p$ zdefiniowanych następująco:

$$0 \leq \alpha_i \leq \vartheta \qquad\qquad \sum_{i=1}^{p} \alpha_i d_i = 0$$

Rozwiązanie problemu dualnego pozwala znaleźć poszukiwaną hiperpłaszczyznę:

$$y(x) = \sum_{i=1}^{p} \alpha_i d_i x_i^T x_j + b$$

Wnioski

Zmienna dopełniająca δ_i ani mnożniki Lagrange'a nią związane nie pojawiają się w sformułowaniu problemu dualnego.

Mnożniki muszą spełniać jedynie podstawowy warunek mówiący, iż iloczyn mnożników i wartości funkcji ograniczenia dla każdej pary danych uczących jest równy zeru. Jeśli więc ograniczenie spełnione jest z nadmiarem dla wektorów nienośnych, wtedy mnożniki te muszą być równe zeru. Niezerowe wartości mnożników występują zaś dla wektorów nośnych.

Niezerowe wartości mnożników określają wektory nośne, których ilość oznaczymy $N_{SV} \leq p$, a więc równanie sieci liniowej SVM o wagach optymalnych wyznacza hiperpłaszczyznę zależne jest tylko od wektorów nośnych:

$$y(x) = \sum_{i=1}^{N_{SV}} \alpha_i d_i x_i^T x_j + b$$

Większość problemów klasyfikacji nie posiada jednak właściwości liniowej separowalności. Potrzebne jest więc nieliniowe rzutowanie danych oryginalnych w inną przestrzeń funkcyjną, gdzie wzorce staną się separowalne liniowo i będzie można zastosować hiperpłaszczyznę separującą SVM.

Warunkiem jest zastosowanie transformacji nieliniowej o odpowiednio wysokim wymiarze K przestrzeni cech $K \geq N$.

Nieliniowa sieć SVM

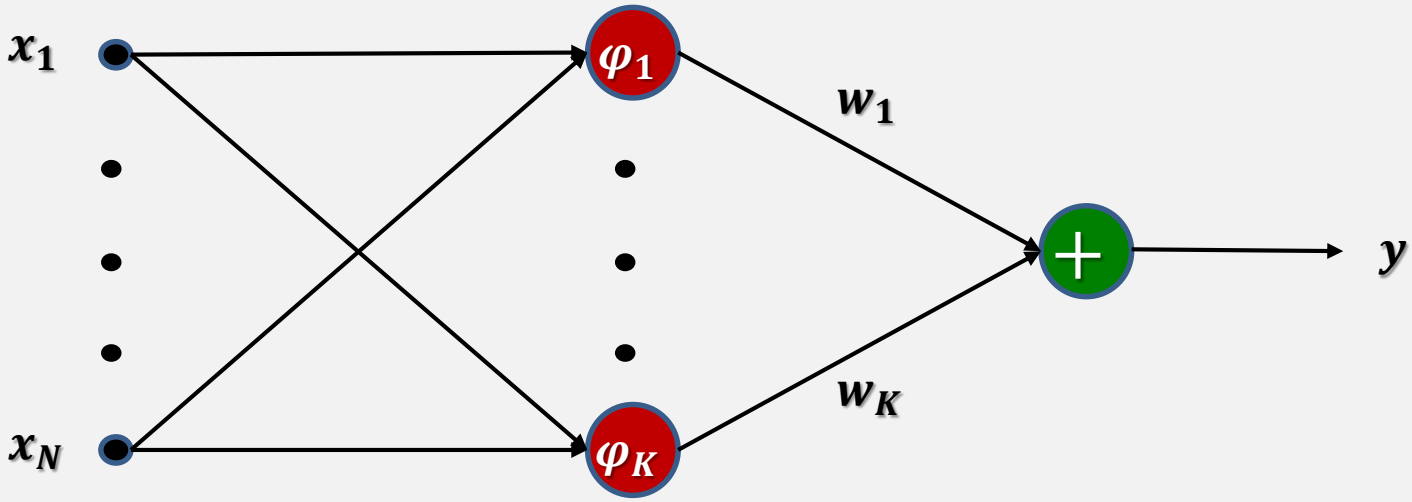
Dla zadań nieseparowalnych liniowo rzutujemy każdy wzorzec z jego N wymiarowej przestrzeni cech do K wymiarowej przestrzeni cech $\varphi_j(x), j = 1, 2, \dots, K.$

W efekcie tego nieliniowego przekształcenia równanie hiperpłaszczyzny określone będzie wzorem:

$$y(x) = w^T \varphi(x) + b = \sum_{j=1}^{K} w_i \varphi_j(x) + b = 0$$

gdzie w_i oznaczają wagi prowadzące od neuronu o nieliniowej funkcji aktywacji φ_j na wektorze danych wejściowych x do wyjściowego neuronu liniowego

Otrzymujemy więc dwuwarstwową strukturę sieci neuronowej zawierającą jedną warstwę ukrytą:



Nieliniowa sieć SVM

Rozwiązanie problemu pierwotnego uzyskujemy więc poprzez zastąpienie zmiennej x_i przez $\varphi_i(x)$. Otrzymujemy więc:

$$\max_{\alpha} Q(\alpha) = \sum_{i=1}^{p} \alpha_i - \frac{1}{2} \sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_j d_i d_j K(x_i, x_j)$$

gdzie K nazywamy funkcją jądra (*kernel function*), zdefiniowaną następująco:

$$K(x_i, x_j) = \varphi^T(x_i)\varphi(x_j)$$

Rozwiązanie problemu sprowadza się do wyznaczenia wartości wag sieci:

$$w = \sum_{i=1}^{p} \alpha_i d_i \varphi(x_i)$$

$$b = \pm 1 - w^T \varphi(x_i)$$

Otrzymując ostatecznie sygnał wyjściowy dla nieliniowej sieci SVN w postaci:

$$y(x) = w^T \varphi(x) + b = \sum_{i=1}^{N_{SV}} \alpha_i d_i \, K(x_i, x) + b = 0$$

Na kandydatów na funkcje jądra K możemy wybrać funkcje spełniające warunek twierdzenia Mercera, np. funkcje gaussowskie, wielomianowe, sklejane, a nawet sigmoidalne przy pewnych ograniczeniach.

Nieliniowe funkcje jądra sieci SVM

Do najczęściej stosowanych funkcji jądra należą:

- **Funkcje liniowe:**

$$K(x_i, x) = x^T x_i + \gamma$$

- **Funkcje wielomianowe:**

$$K(x_i, x) = \left(x^T x_i + \gamma\right)^p$$

- **Funkcje gaussowskie:**

$$K(x_i, x) = exp\left(-\gamma \, \|x - x_i\|^2\right)$$

- **Funkcje sigmoidalne:**

$$K(x_i, x) = tgh\left(\beta x^T x_i + \gamma\right)$$

Gdzie β, γ to stałe współczynniki liczbowe, a p to stopień wielomianu.

Sieć SVM o radialnych funkcjach bazowych jest bardzo podobna do sieci radialnej RBF, aczkolwiek sposób jej tworzenia i wyznaczania wag różni się.

Podobnie stosując funkcje sigmoidalne otrzymujemy dwuwarstwową sieć MLP.

Chcąc zastosować sieci SVM do większej ilości klas niż dwie trzeba zbudować kilka sieci SVM, które dyskryminują wzorce każdej z klas od pozostałych lub pomiędzy parą każdych dwóch klas, a następnie wyniki są sumowane.

Dążenie do poprawności SVM

Stosuje się często współczynnik kary za niespełnienie któregoś z ograniczeń, co wymusza dążenie sieci do optymalności dla przyjętych stałych.

Z warunków optymalności Kuhna-Tuckera problemu optymalizacyjnego sformułowanego dla SVM wynikają następujące zależności:

$$\alpha_i\left[d_i\left(w^T\varphi(x_i)+b\right)-(1-\delta_i)\right]=0$$

$$0\le\alpha_i\le\vartheta$$

$$\mu_i\delta_i=0$$

$$\alpha_i+\mu_i=\vartheta$$

$$\delta_i\ge 0$$

W zależności od wyznaczonych współczynników Lagrange'a mamy więc do czynienia z trzema przypadkami:

- $\alpha_i=0$ – co oznacza, że jeśli $\alpha_i+\mu_i=\vartheta$, to $\mu_i=\vartheta$, a więc z zależności $\mu_i\delta_i=0$ wynika iż $\delta_i=0$, stąd para ucząca (x_i,d_i) spełnia ograniczenie z nadmiarem, a więc bez zmniejszania szerokości marginesu separacji
- $0<\alpha_i<\vartheta$ – co oznacza, iż $\mu_i=\vartheta-\alpha_i$, a więc również $\delta_i=0$, stąd para ucząca(x_i,d_i)definiuje wektor nośny, który jest położony dokładnie na marginesie separacji.
- $\alpha_i=\vartheta$ – oznacza, iż $\mu_i=\vartheta-\alpha_i=0$, a więc $\delta_i\ge 0$, co oznacza, iż wzorzec uczący jest wewnątrz marginesu separacji powodując zwężenie marginesu separacji albo nawet po niewłaściwej stronie, jeśli $\delta_i>1$.

ALGORYTMY ROZWIĄZANIA ZADANIA DUALNEGO dla dużych zbiorów danych

- Niezależnie od zastosowanego jądra i rodzaju zadania główny problem obliczeniowy w sieciach SVM sprowadza się do rozwiązania zadania programowania kwadratowego z ograniczeniami liniowymi.
- Problemem staje się duża ilość danych uczących, co związane jest z nieraz ogromną ilością optymalizowanych zmiennych – tutaj mnożników Lagrange'a. Pojawiają się problemy z pamięcią i złożonością obliczeniową, co eliminuje możliwość zastosowania klasycznych metod programowania kwadratowego, np. MINOS, OSL, LOQO czy Matlab.
- Stosuje się dekompozycję zbioru uczącego na szereg podzbiorów oraz strategię aktywnych ograniczeń wynikających z równości, zaniedbując te nieaktywne ze znakiem silniej nierówności. Dzięki temu w kolejnych iteracjach następuje przemieszczanie części wzorców ze zbioru ograniczeń aktywnych do nieaktywnych.
- Wykorzystuje się również różne wersje algorytmu programowania sekwencyjnego SMO lub BSVM Platta oraz suboptymalną metodę SVM_{Light} Joachimsa.

LITERATURA I BIBLIOGRAFIA

- Stanisław Osowski, Metody i narzędzia eksploracji danych, BTC, Legionowo, 2013.
- T. Joachims, Making large scale SVM learning practical, in Advances in kernel methods – suport vector learning, B. Scholkopf, C. Burges, A. Smola eds., MIT Press, pp 41-56, Cambridge 1998.
- Lin C.J., Chang C.C, LIBSVM: a library for support vector machines: http://www.csie.ntu.edu.tw/cjlin/libsvm
- Open MIT Lectures about SVM : https://www.youtube.com/watch?v=_PwhiWxHK8o
- Caltech Lectures about SVM: https://www.youtube.com/watch?v=eHsErlPJWUU