Support Vector Machines

Accelerated Network Technologies
Research Group

Václav Bartoš

Brno University of Technology, Faculty of Information Technology
Bozetechova 2, 612 00 Brno, CZ
http://merlin.fit.vutbr.cz/ant/

BRNO UNIVERSITY OF TECHNOLOGY
FIT
FACULTY OF INFORMATION TECHNOLOGY

evropský sociální fond v ČR
EVROPSKÁ UNIE
MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY
OP Vzdělávání pro konkurenceschopnost
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Support Vector Machines

- SVM - Binární lineární klasifikátor
  - Rozděluje data (reprezentovaná jako body v n-rozměrném prostoru) do dvou tříd.
  - Rozděluje nadrovinou
  - V základní variantě tedy musí být data lineárně separovatelná
  - Úkolem je najít takovou nadrovinu, aby vzdálenost k nejbližším bodům v obou skupinách byla maximální.

SVM - Výpočet dělící nadroviny

- Definujme trénovací množinu dat jako:

  $\{\boldsymbol{x}_i, c_i\} \quad \boldsymbol{x}_i \in \mathbb{R}^D, c_i \in \{-1, 1\}$

- Nadrovina je dána svým normálovým vektorem **w** a posunutím *b*.

SVM - Výpočet dělící nadroviny

- Vzdálenost bodu od nadroviny je $1/\|\boldsymbol{w}\|$
- Snažíme se najít **w** a *b* tak, aby platilo:

$$\boldsymbol{x}_i \cdot \boldsymbol{w} + b \geq +1 \quad \text{když} \quad c_i = +1$$
$$\boldsymbol{x}_i \cdot \boldsymbol{w} + b \leq -1 \quad \text{když} \quad c_i = -1$$

- Což lze spojit do:

$$c_i(\boldsymbol{x}_i \cdot \boldsymbol{w} + b) - 1 \geq 0$$

- Snažíme se maximalizovat $1/\|\boldsymbol{w}\|$, tedy minimalizovat $\|\boldsymbol{w}\|$.

SVM - Výpočet dělící nadroviny

- Minimalizaci $\|\boldsymbol{w}\|$ převedeme na minimalizaci $\frac{1}{2}\|\boldsymbol{w}\|^2$.
- Dostaneme optimalizační problém kvadratického programování
  - Minimalizuj (ve **w**, *b*)

    $$\frac{1}{2}\|\boldsymbol{w}\|^2$$

    tak, aby platilo

    $$c_i(\boldsymbol{w}\cdot\boldsymbol{x}_i - b) - 1 \geq 0 \quad \forall i$$
- Řeší se pomocí Lagrangeových multiplikátorů:

$$L_P \equiv \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=-1}^{n} \alpha_i [c_i(\boldsymbol{w}\cdot\boldsymbol{x}_i - b) - 1]$$

minimalizujeme **w**,*b;* maximalizujeme *α*

SVM - Výpočet dělící nadroviny

$$L_P \equiv \frac{1}{2} \| \boldsymbol{w} \|^2 - \sum_{i=-1}^{n} \alpha_i [c_i (\boldsymbol{w} \cdot \boldsymbol{x}_i - b) - 1]$$

- Lze převést na duální formu:

$$L_D \equiv \sum_{i=1}^{L} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j c_i c_j \boldsymbol{x}_i \cdot \boldsymbol{x}_j \quad \text{tak aby} \quad a_i \geq 0, \quad \sum_{i=1}^{L} \alpha_i c_i =$$

- Z toho pomocí QP solveru vypočítáme *α* a dopočítáme **w** a *b*.

- Podstatné je, že duální forma potřebuje jen skalární součiny vstupních vektorů $\mathbf{x}_i$.

SVM - Použití

- Jakmile známe **w** a *b* definující dělící nadrovinu, třída nové instance dat se vypočítá jednoduše:

$$c' = \mathrm{sgn}\left(\boldsymbol{w} \cdot \boldsymbol{x}' + b\right)$$

Transductive SVM

- K trénovacím datům navíc množina neoznačených dat.
- SVM se snaží najít nadrovinu tak, aby byla optimální i vzhledem k těmto datům.
- Původní optimalizační problém se změní na:
  - Minimalizuj (ve **w**, *b*, *c**)

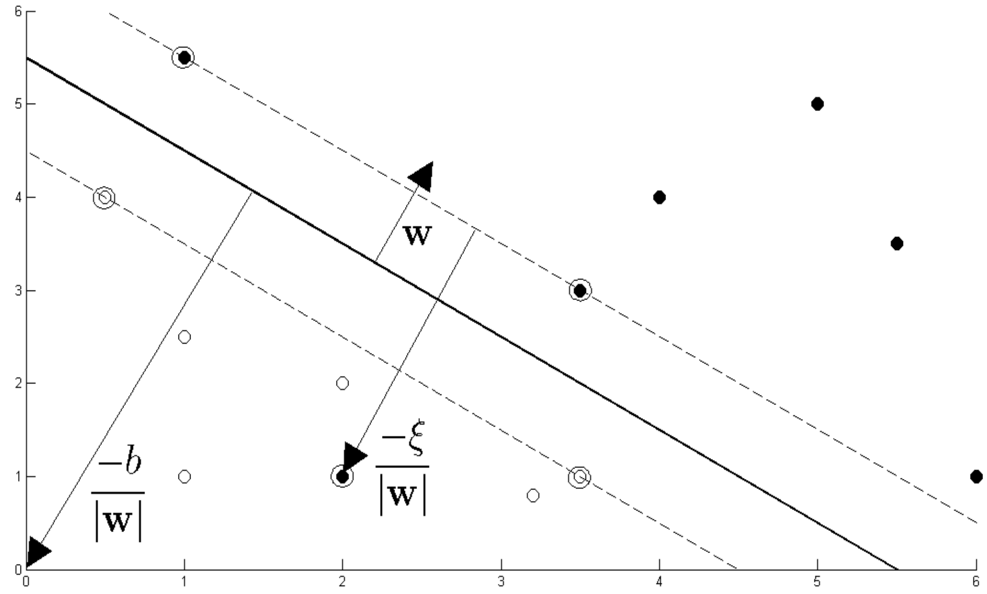
$$\frac{1}{2}\|\boldsymbol{w}\|^2$$

tak, aby platilo

$$c_i(\boldsymbol{w}\cdot\boldsymbol{x_i}+b) \geq 1,$$
$$c_j^*(\boldsymbol{w}\cdot\boldsymbol{x_j^*}+b) \geq 1,$$
$$c_j^* \in \{-1, 1\}$$

Soft margin SVM

- Pokud data nejsou úplně lineárně separabilní z důvodu několika špatně označených instancí dat.



- Zavedeme proměnnou ξ vyjadřující "rezervu":

$$\boldsymbol{x}_i \cdot \boldsymbol{w} + b \geq +1 - \xi_i \quad \text{když} \quad c_i = +1$$

$$\boldsymbol{x}_i \cdot \boldsymbol{w} + b \leq -1 + \xi_i \quad \text{když} \quad c_i = -1$$

$$\xi_i \geq 0$$

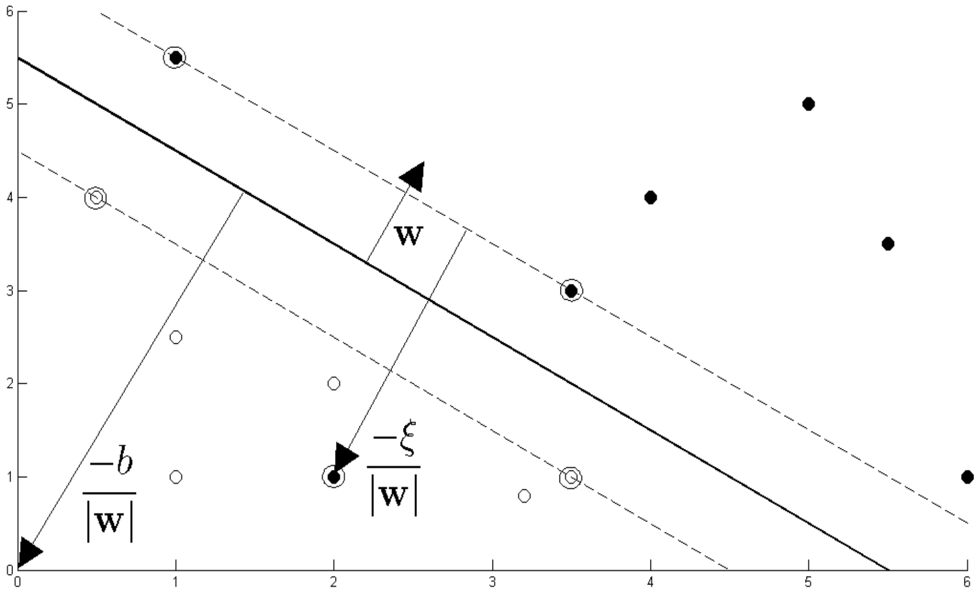
Soft margin SVM

- Optimalizační problém se změní na:
  - Minimalizuj (ve **w**, *b*)

$$\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{L}\xi_i$$

tak, aby platilo

$$c_i(\boldsymbol{w}\cdot\boldsymbol{x}_i - b) - 1 + \xi_i \geq 0 \quad \forall i$$



- Pomocí parametru C můžeme měnit trade-off mezi šířkou "prázdného prostoru" a množstvím špatně klasifikovaných dat.

Nonlinear classification

- Do teď musela být data lineárně separovatelná.
- Pomocí tzv. *kernel trick* lze s SVM udělat nelineární klasifikátor.
  - Problém v původních N dimenzích se pomocí nelineární transformace převede do vícedimenzionálního prostoru, ve kterém již data jsou lineárně separovatelná.
    - Dimenzí může být i nekonečno
  - Algoritmus tedy zůstává stejný, transformují se vstupní data.

Nonlinear classification

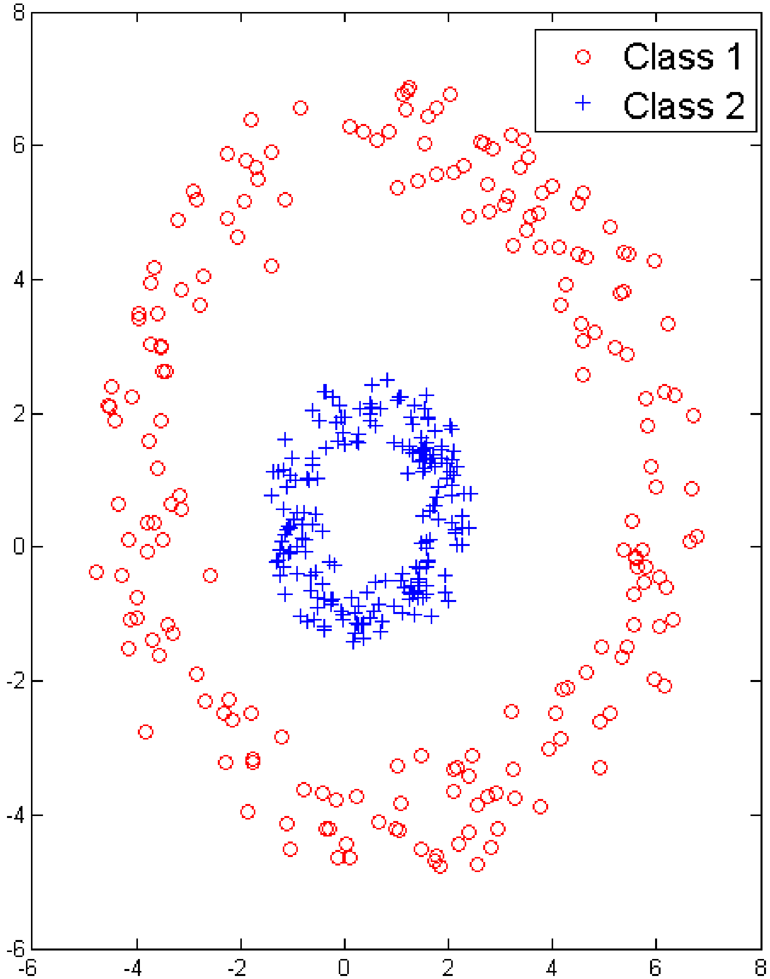
- Vztah kernelu *k* a transformace *φ* je:

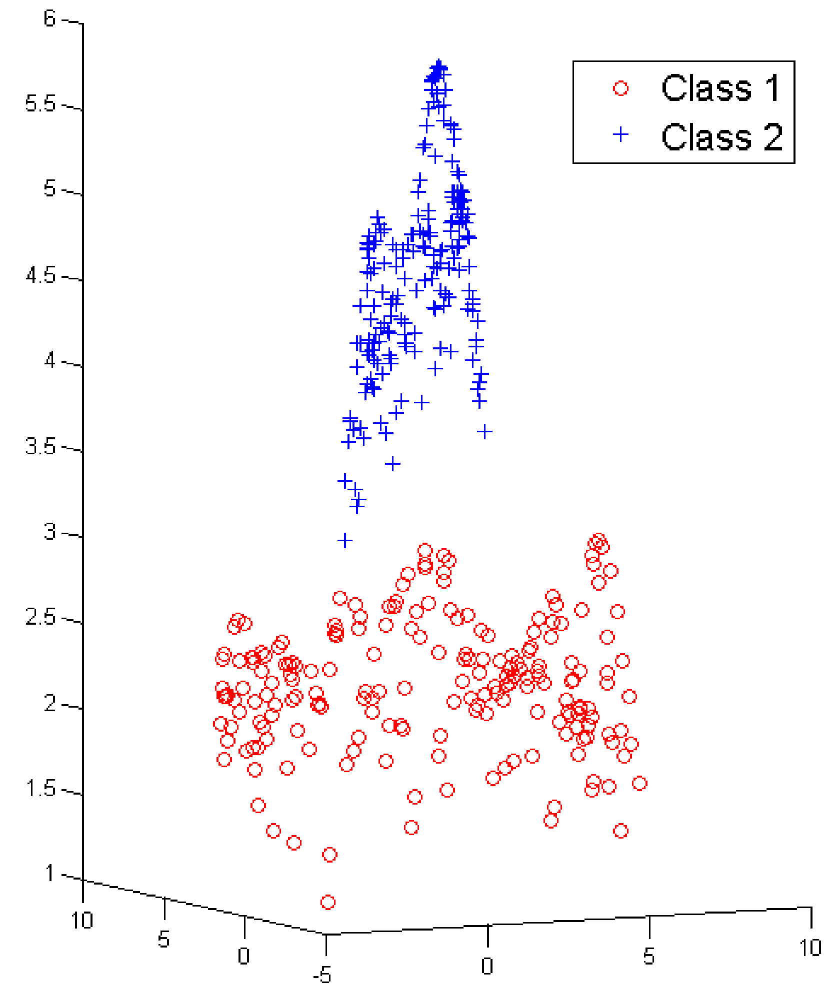
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \varphi(\boldsymbol{x}_i) \cdot \varphi(\boldsymbol{x}_j)$$

- Výpočty s vektory v SVM obsahují pouze skalární součiny (nebo je na ně lze převést).
- Proto ve skutečnosti nemusíme data transformovat, stačí ve výpočtech každý skalární součin nahradit nějakou kernel funkcí.
- Často používaným kernelem je *Gaussian radial basis function*

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\gamma \left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|^2\right) \quad \text{pro} \quad \gamma > 0$$

Nonlinear classification





Nonlinear classification

- Používané kernely:
  - *Radial basis function*

    $$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\gamma \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2) \text{ pro } \gamma > 0$$

  - *Polynomial kernel*

    $$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i \cdot \boldsymbol{x}_j + a)^b$$

  - *Sigmoidal kernel*

    $$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \tanh(a\, \boldsymbol{x}_i \cdot \boldsymbol{x}_j - b)$$

- Efektivita SVM záleží na zvoleném kernelu a jeho parametrech (v případě soft margin ještě na konst. C).

Konec

Děkuji za pozornost.

Hlavní zdroje:

http://en.wikipedia.org/wiki/Support_vector_machine
http://www.tristanfletcher.co.uk/SVM%20Explained.pdf