Pokročilé metody analýzy dat v neurovědách

RNDr. Eva Koriťáková, Ph.D.
doc. RNDr. Ladislav Dušek, Dr.

Blok 8

Klasifikace dat II

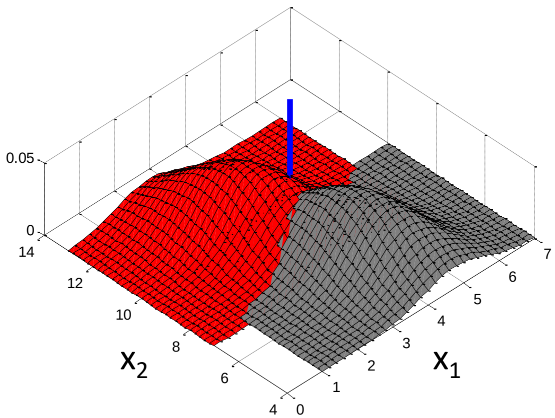
Osnova

1. Klasifikace pomocí hranic – metoda podpůrných vektorů (SVM)
2. Další metody klasifikace
3. Hodnocení úspěšnosti klasifikace a srovnání klasifikátorů

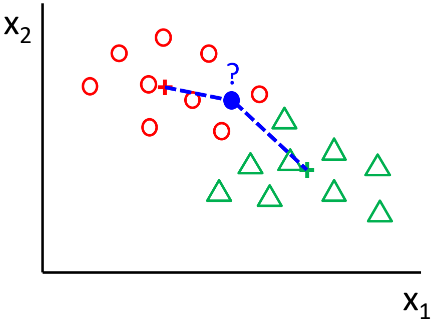
Klasifikace pomocí hranic – metoda podpůrných vektorů (SVM)

Typy klasifikátorů – podle principu klasifikace

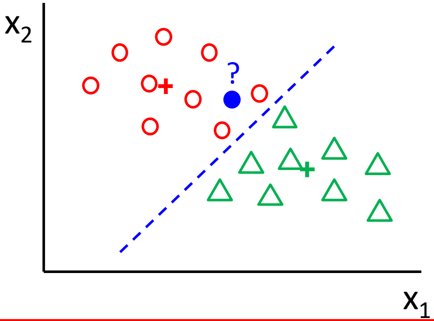
- **klasifikace pomocí diskriminačních funkcí:**
  - diskriminační funkce určují míru příslušnosti k dané klasifikační třídě
  - pro danou třídu má daná diskriminační funkce nejvyšší hodnotu



- **klasifikace pomocí min. vzdálenosti od etalonů klasif. tříd:**
  - etalon = reprezentativní objekt(y) klasifikační třídy
  - počet etalonů klasif. třídy různý – od jednoho vzorku (např. centroidu) po úplný výčet všech objektů dané třídy (např. u klasif. pomocí metody průměrné vazby)

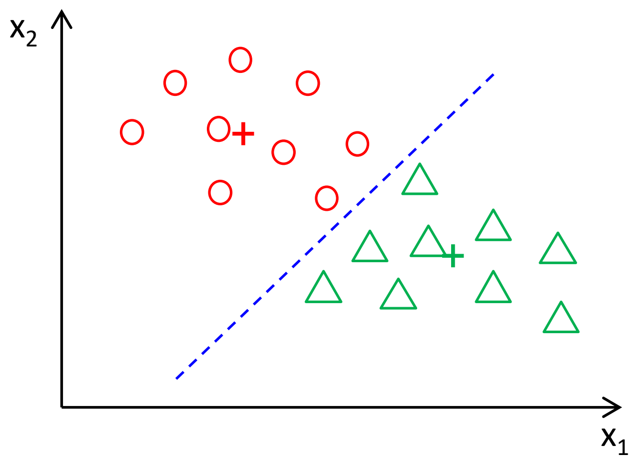


- **klasifikace pomocí hranic v obrazovém prostoru:**
  - stanovení hranic (hraničních ploch) oddělujících klasifikační třídy

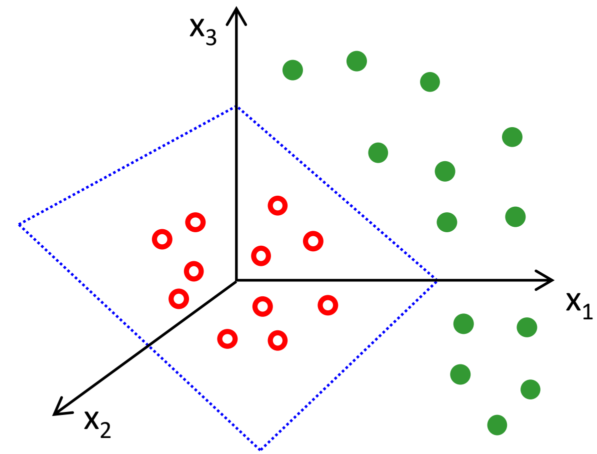


Motivace

2-rozměrný prostor



3-rozměrný prostor



Hranice je nadplocha o rozměru o jedna menší než je rozměr prostoru

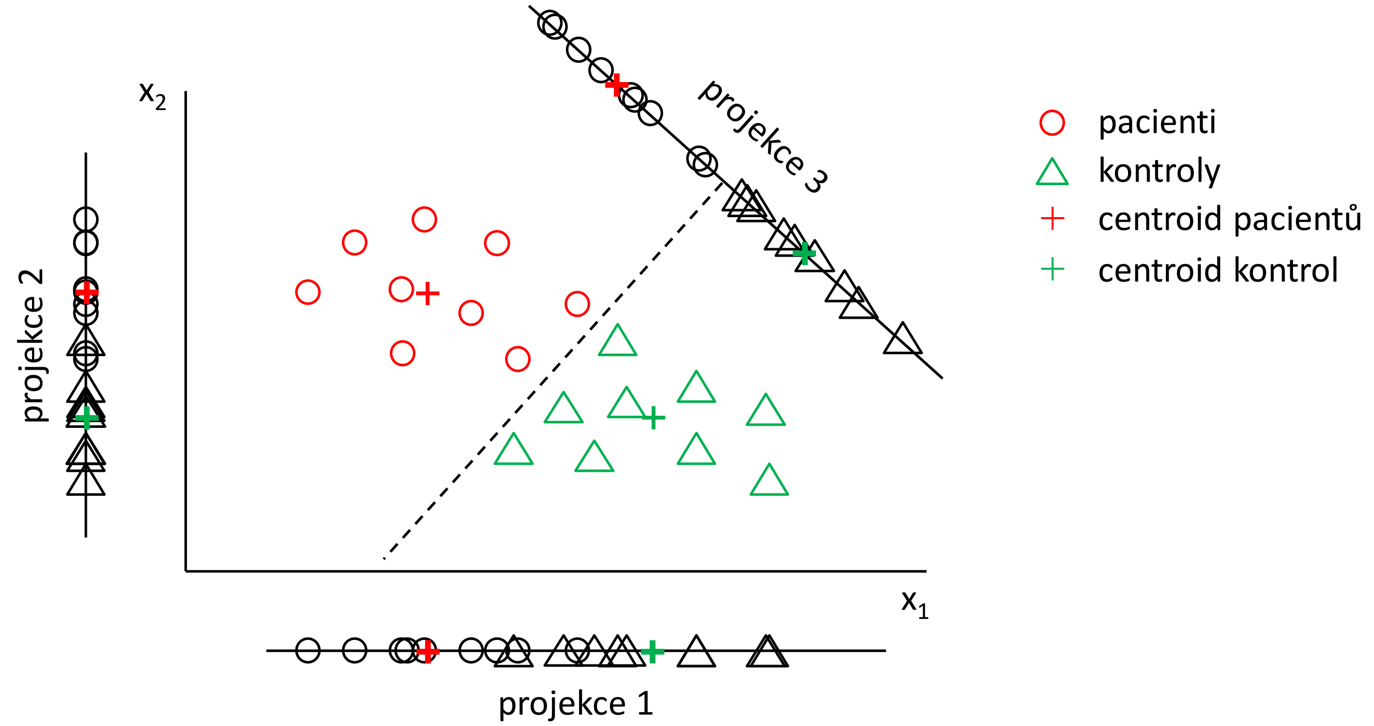
- ve 2-rozměrném prostoru je hranicí křivka (v lineárním případě přímka)
- v 3-rozměrném prostoru plocha (v lineárním případě rovina)

Hranice je tedy dána rovnicí: $\mathrm{h}(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + \mathrm{w}_0 = 0$

Výpočet hranice různými metodami (např. Fisherova LDA, SVM apod. – viz dále)

Fisherova lineární diskriminace (FLDA)

- použití pro lineární klasifikaci
- princip: transformace do jednorozměrného prostoru tak, aby se třídy od sebe maximálně oddělily (maximalizace vzdálenosti skupin a minimalizace variability uvnitř skupin

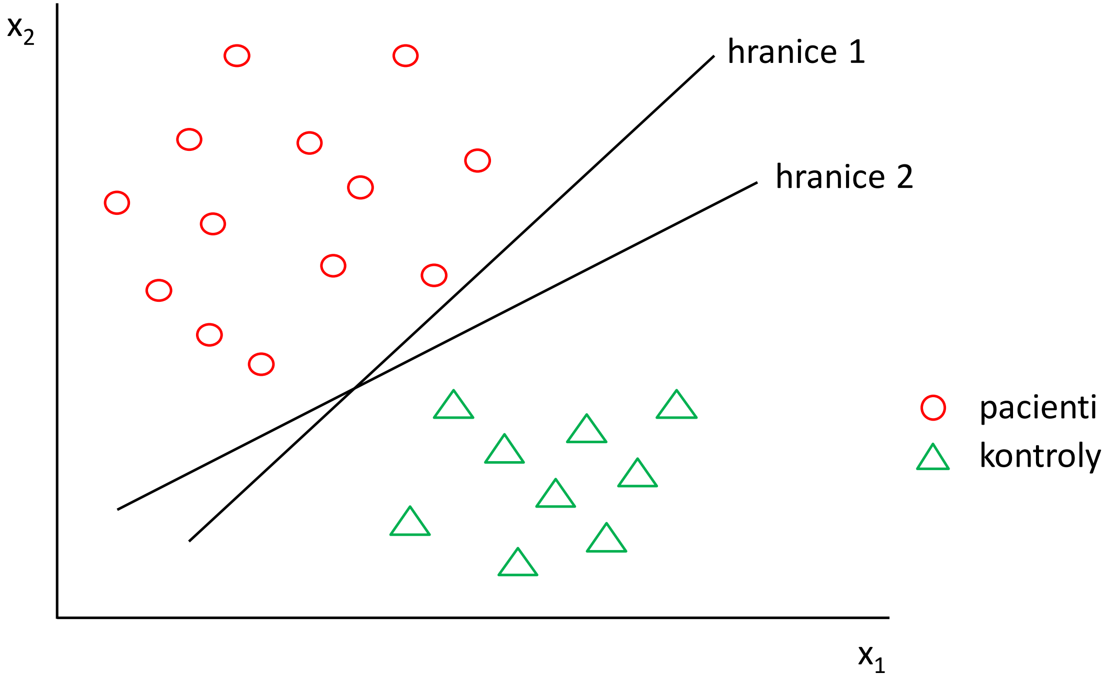


- předpoklad: vícerozměrné normální rozdělení u jednotlivých skupin

Metoda podpůrných vektorů (SVM)

Anglicky: Support Vector Machines

Princip: Proložení klasifikační hranice tak, aby byla v co největší vzdálenosti od subjektů z obou tříd.



Výhody:

+ nemá předpoklady o normálním rozdělení dat
+ lze využít pro lineární i pro nelineární klasifikaci

Nevýhody:

- vyžaduje stanovení parametrů (např. C) a případně i typu jádra

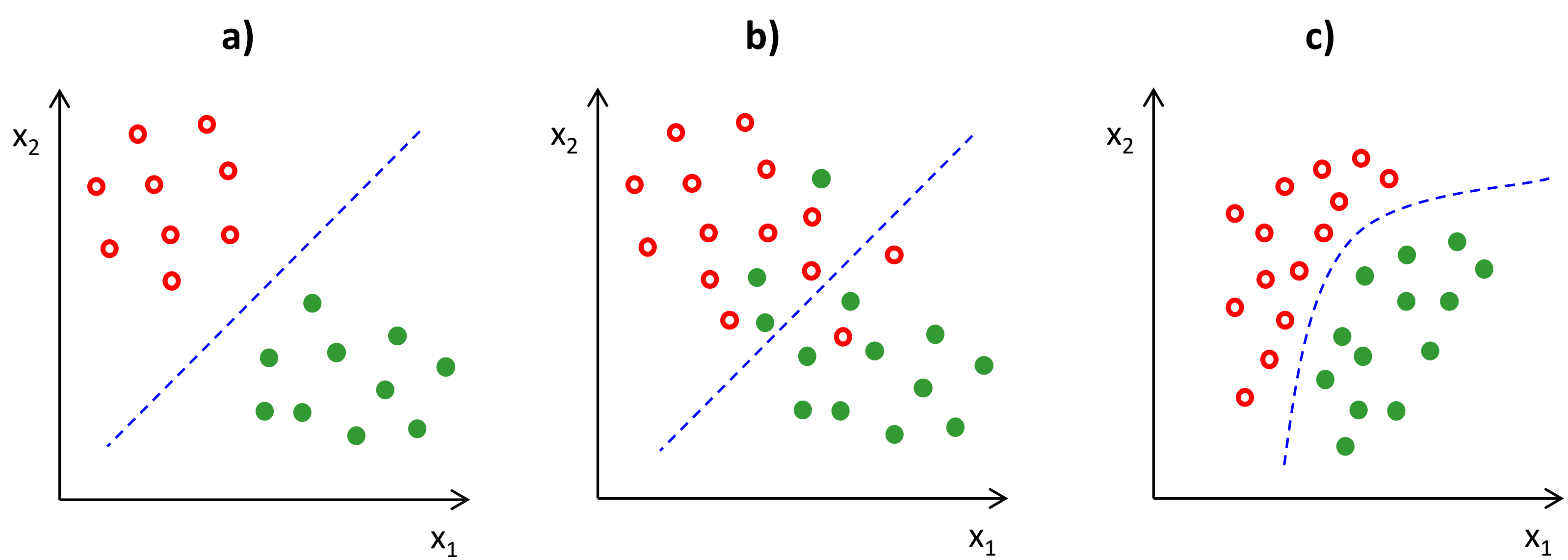
Metoda podpůrných vektorů (SVM) – varianty

Varianty SVM dle typu vstupních dat:

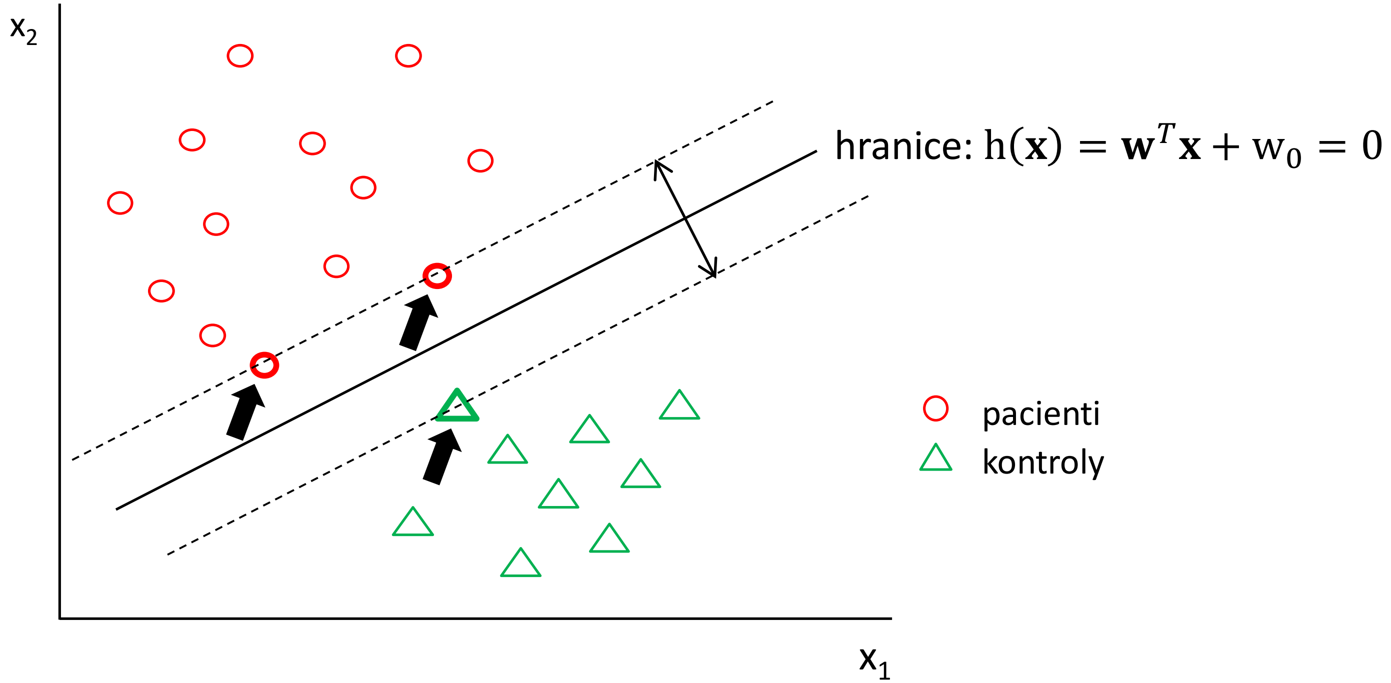
a) lineární verze metody podpůrných vektorů pro lineárně separabilní třídy (anglicky *maximal margin classifier*)

b) lineární verze metody podpůrných vektorů pro lineárně neseparabilní třídy (anglicky *support vector classifier*)

c) nelineární verze metody podpůrných vektorů (anglicky *support vector machine*)



Metoda podpůrných vektorů (SVM) – princip



- proložení klasifikační hranice tak, aby byla v co největší vzdálenosti od subjektů z obou tříd → tzn. aby byl okolo hranice co nejširší pruh bez bodů (tzv. toleranční pásmo = margin)
- na popis hranice stačí pouze nejbližší body, kterých je obvykle málo a nazývají se **podpůrné vektory** (support vectors)

Lineární SVM – lineárně separabilní třídy

- Pro všechny body z trénovací množiny platí:

$$\mathbf{w}^{\mathrm{T}}\mathbf{x} + w_0 \geq 1 \quad \text{pro všechna } \mathbf{x} \text{ z } \omega_D,$$
$$\mathbf{w}^{\mathrm{T}}\mathbf{x} + w_0 \leq -1 \quad \text{pro všechna } \mathbf{x} \text{ z } \omega_H,$$

- což můžeme stručněji zapsat jako

$$\boxed{\delta_{x_k}\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_k + w_0\right) \geq 1, \text{pro k=1, ..., N,}}$$

- kde δ_{x_k} = 1 pro $\mathbf{x}_k$ ze třídy ω_D a δ_{x_k} = -1 pro $\mathbf{x}_k$ ze třídy ω_H

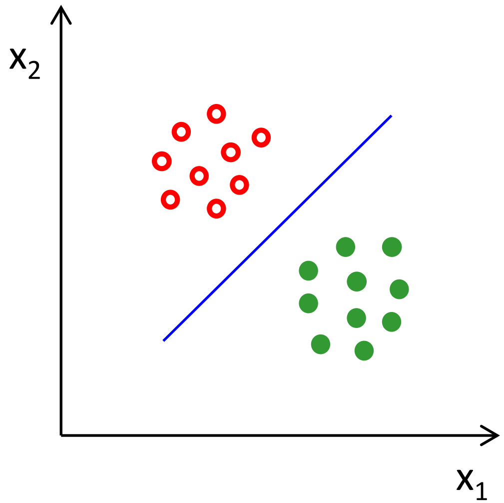
- hledáme takové hodnoty $\mathbf{w}$ a w_0, aby byla celková šířka tolerančního pásma $\frac{1}{\|\mathbf{w}\|} + \frac{1}{\|\mathbf{w}\|} = \frac{2}{\|\mathbf{w}\|}$ co největší

- hledat maximum funkce $\frac{2}{\|\mathbf{w}\|}$ je to stejné, jako hledat minimum funkce $\frac{\|\mathbf{w}\|}{2}$ a toto minimum se nezmění, když kladnou hodnotu v čitateli umocníme na druhou (což nám zjednoduší výpočty), takže dostáváme následující kriteriální funkci, jejíž hodnotu se snažíme minimalizovat:

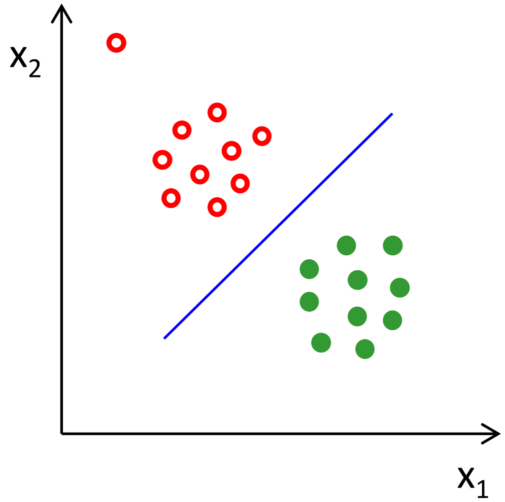
$$\boxed{\mathrm{J}(\mathbf{w}, w_0) = \frac{\|\mathbf{w}\|^2}{2}}$$

→ řešení pomocí metody Lagrangeova součinitele

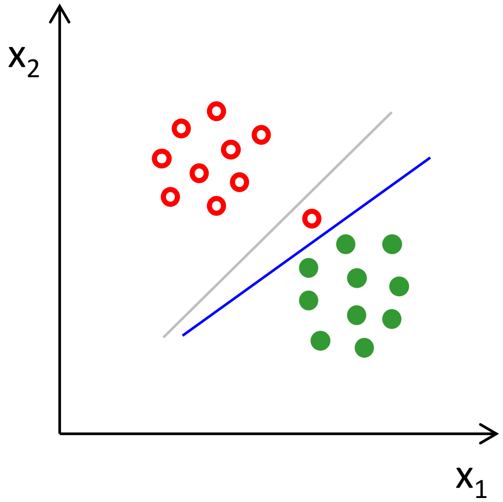
Lineární SVM – vliv odlehlých hodnot



klasifikace v případě dat neobsahujících odlehlé hodnoty



klasifikace v případě odlehlé hodnoty, která není podpůrným vektorem (poloha klasifikační hranice se nezmění)



klasifikace v případě odlehlé hodnoty, která je podpůrným vektorem (poloha hranice se změní)

→ lepší použít lineární SVM pro lineárně neseparabilní třídy, kterou tato odlehlá hodnota téměř neovlivní

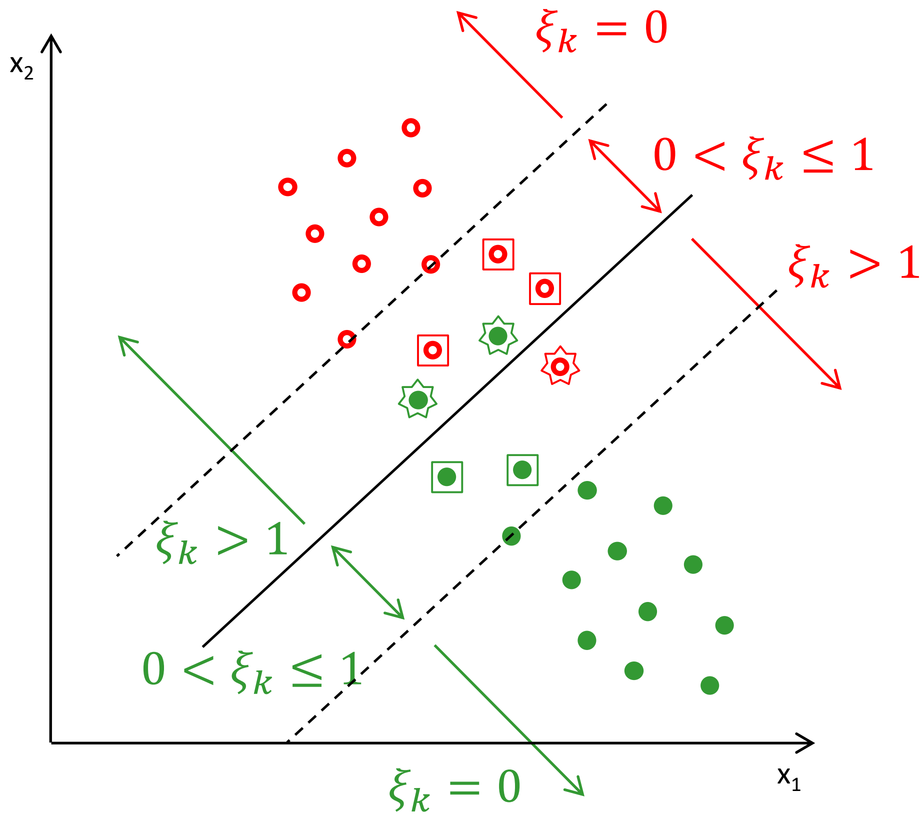
Lineární SVM – lineárně neseparabilní třídy

- zavedeme relaxační proměnné $\xi_k \geq 0$ vyjadřující, jak moc každý bod porušuje podmínku $\delta_{x_k}\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_k + w_0\right) \geq 1$

- 3 situace:
  1. objekt leží **vně** pásma a je **správně** klasifikován: $\xi_k = 0$
  2. objekt leží **uvnitř** pásma a je **správně** klasifikován (body s čtverečky): $0 < \xi_k \leq 1$
  3. objekt leží na **opačné straně** hranice a je **chybně** klasifikován (body s hvězdičkami): $\xi_k > 1$

- podmínky jsou pak ve tvaru:

$$\boxed{\delta_{x_k}\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_k + w_0\right) \geq 1 - \xi_k}$$



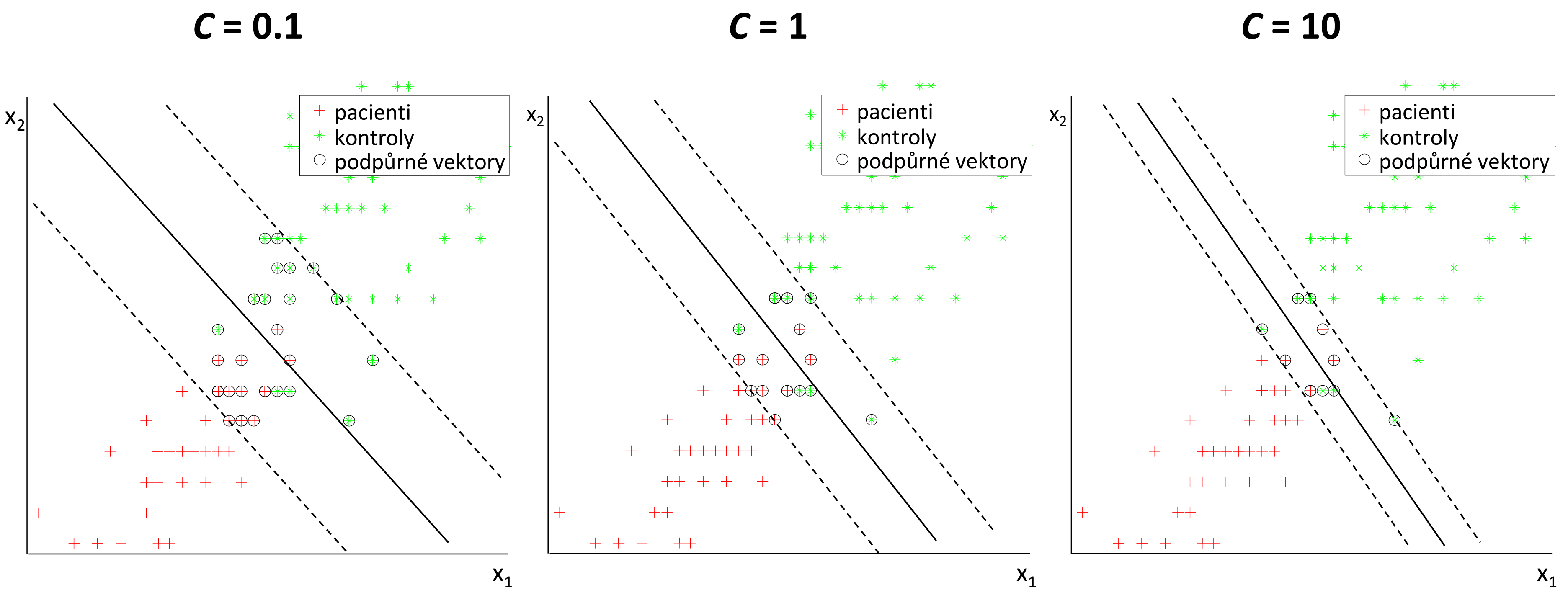
Lineární SVM – lineárně neseparabilní třídy

- když chceme najít hranici poskytující co nejrobustnější klasifikaci, musíme se snažit:
  - maximalizovat šířku tolerančního pásma
  - minimalizovat počet subjektů z trénovací množiny, které leží v tolerančním pásmu nebo jsou dokonce špatně klasifikovány (tj. těch, pro které $\xi_k > 0$)
- to můžeme vyjádřit jako minimalizaci kriteriální funkce:

$$\boxed{J(\mathbf{w}, w_0, \boldsymbol{\xi}) = \frac{\|\mathbf{w}\|^2}{2} + C\sum_{k=1}^{N} \xi_k}$$

- kde C vyjadřuje poměr vlivu obou členů kriteriální funkce:
  - **pro nízké hodnoty C** bude toleranční pásmo širší a počet trénovaních subjektů v tolerančním pásmu a počet chybně klasifikovaných trénovacích subjektů bude vyšší
  - **pro vysoké hodnoty C** bude toleranční pásmo užší, ale počet trénovaních subjektů v tolerančním pásmu a počet chybně klasifikovaných trénovacích subjektů bude nižší
- řešíme opět pomocí metody Lagrangeova součinitele

SVM – vliv parametru C („box constraint“)

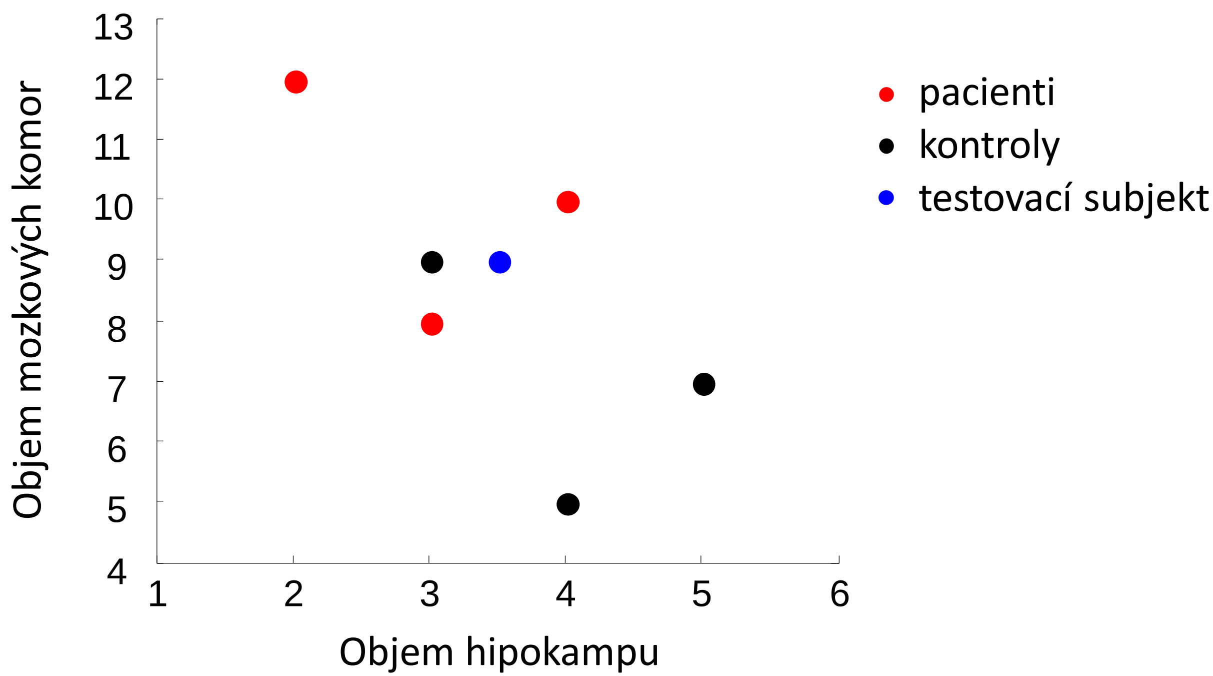


- **pro nízké hodnoty C –** toleranční pásmo širší, ale počet subjektů v tolerančním pásmu a počet chybně klasifikovaných trénovacích subjektů vyšší
- **pro vysoké hodnoty C –** toleranční pásmo užší, ale počet subjektů v tolerančním pásmu a počet chybně klasifikovaných trénovacích subjektů nižší
- zpravidla nevíme, jaká hodnota parametru C pro data nejvhodnější → klasifikace s několika hodnotami C a výběr toho výsledku, který je nejlepší (křížová validace)

Příklad

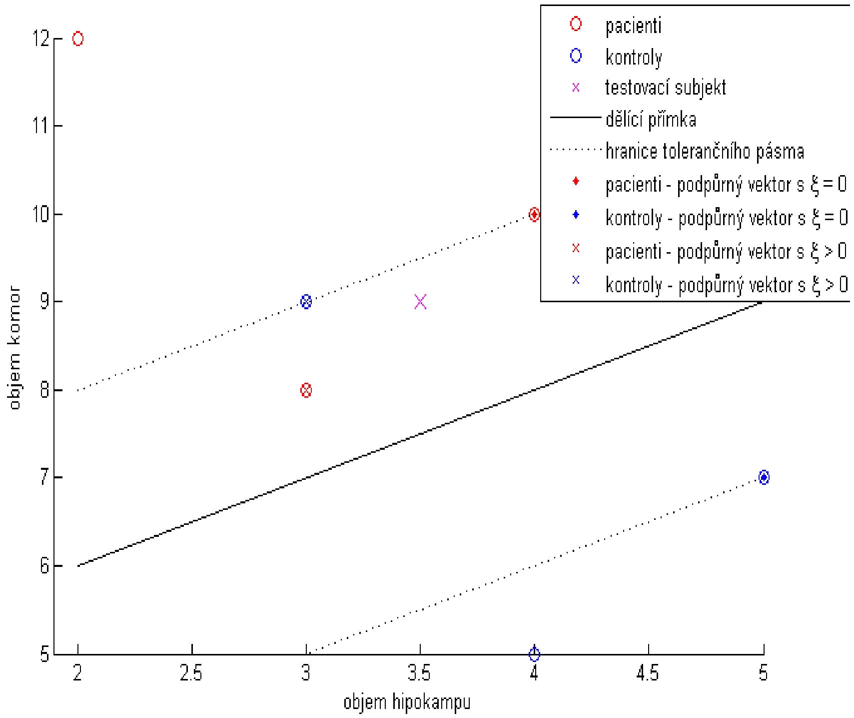
Příklad: Bylo provedeno měření objemu hipokampu a mozkových komor (v cm^3) u 3 pacientů se schizofrenií a 3 kontrol: $\mathbf{X}_D = \begin{bmatrix} 2 & 12 \\ 4 & 10 \\ 3 & 8 \end{bmatrix}$, $\mathbf{X}_H = \begin{bmatrix} 5 & 7 \\ 3 & 9 \\ 4 & 5 \end{bmatrix}$.

Určete, zda testovací subjekt $\mathbf{x}_0 = [3{,}5 \quad 9]$ patří do skupiny pacientů či kontrolních subjektů pomocí metody podpůrných vektorů.

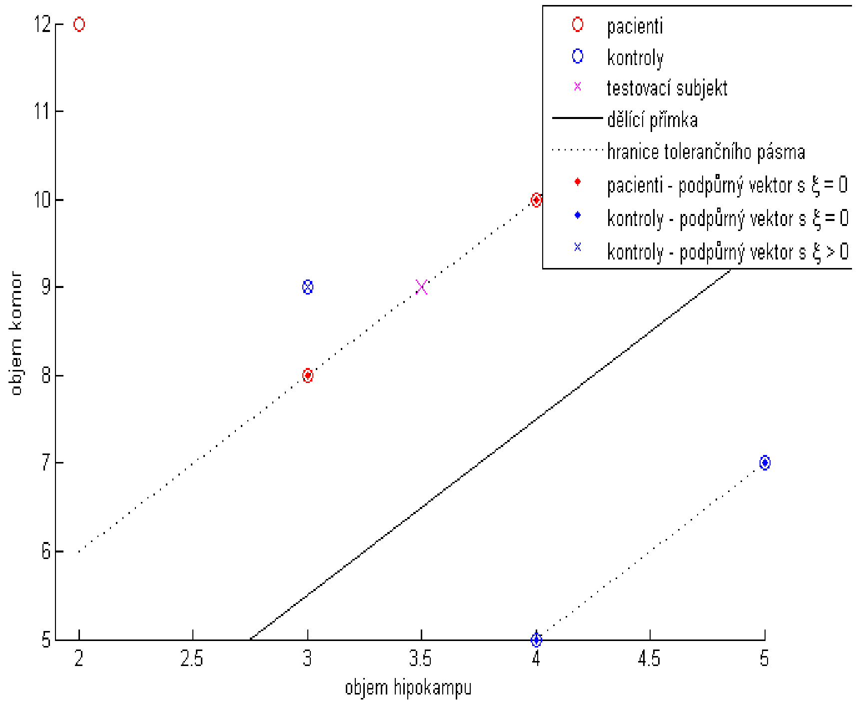


Příklad – srovnání výsledků pro $C = 1$ a $C = 10$

$C = 1$:

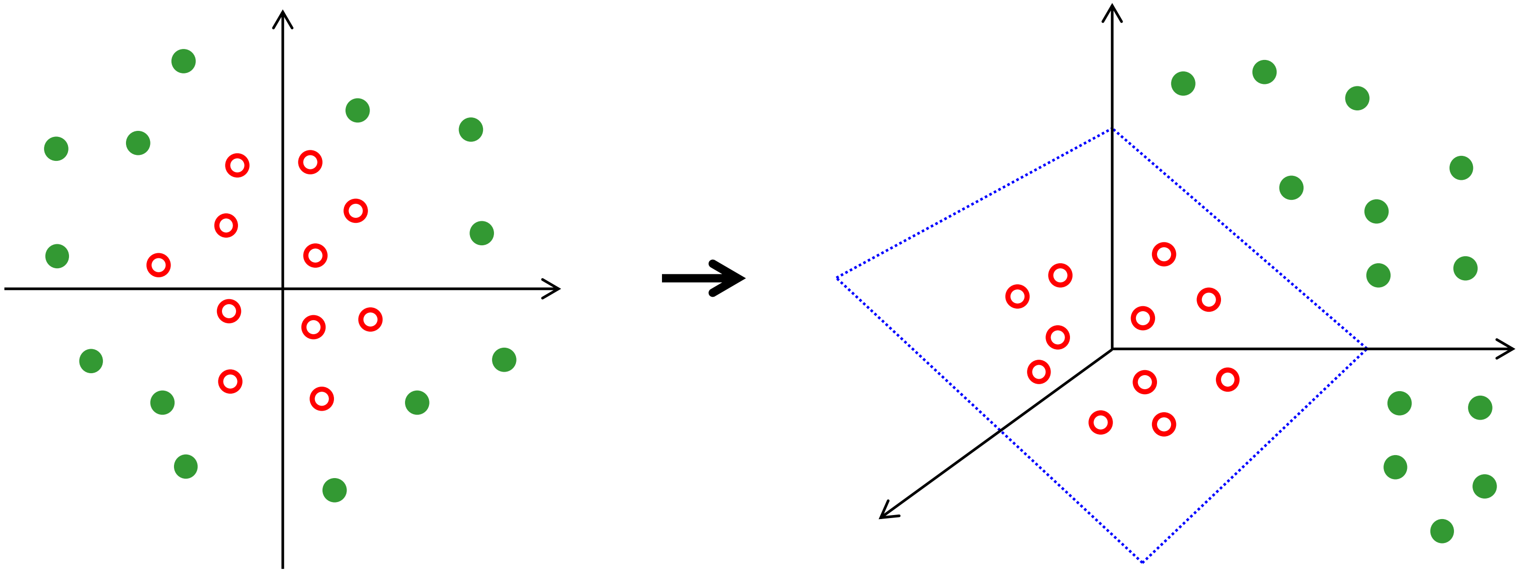


$C = 10$:

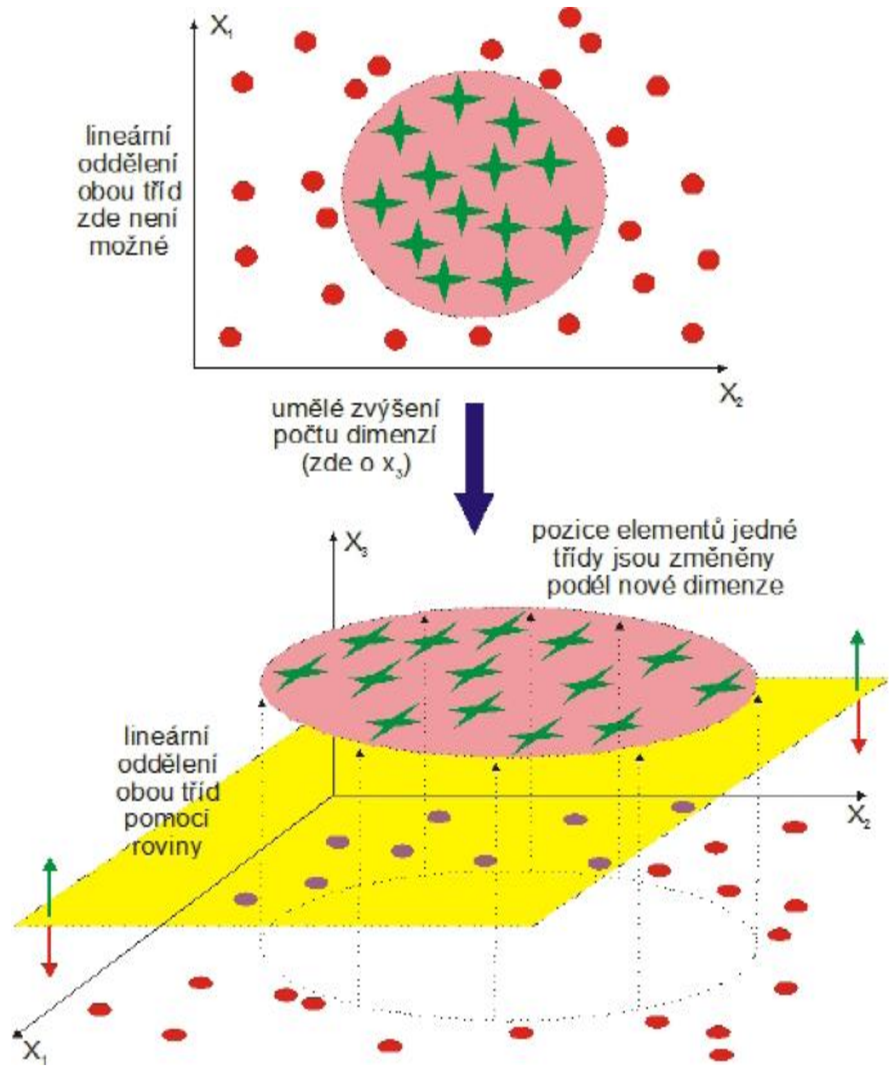


Nelineární SVM

- **princip:** zobrazíme původní *p*-rozměrný obrazový prostor nelineární transformací pomocí **jader** (např. polynomiální nebo radiální bázová funkce) do nového *m*-rozměrného prostoru tak, aby v novém prostoru byly klasifikační třídy lineárně separabilní

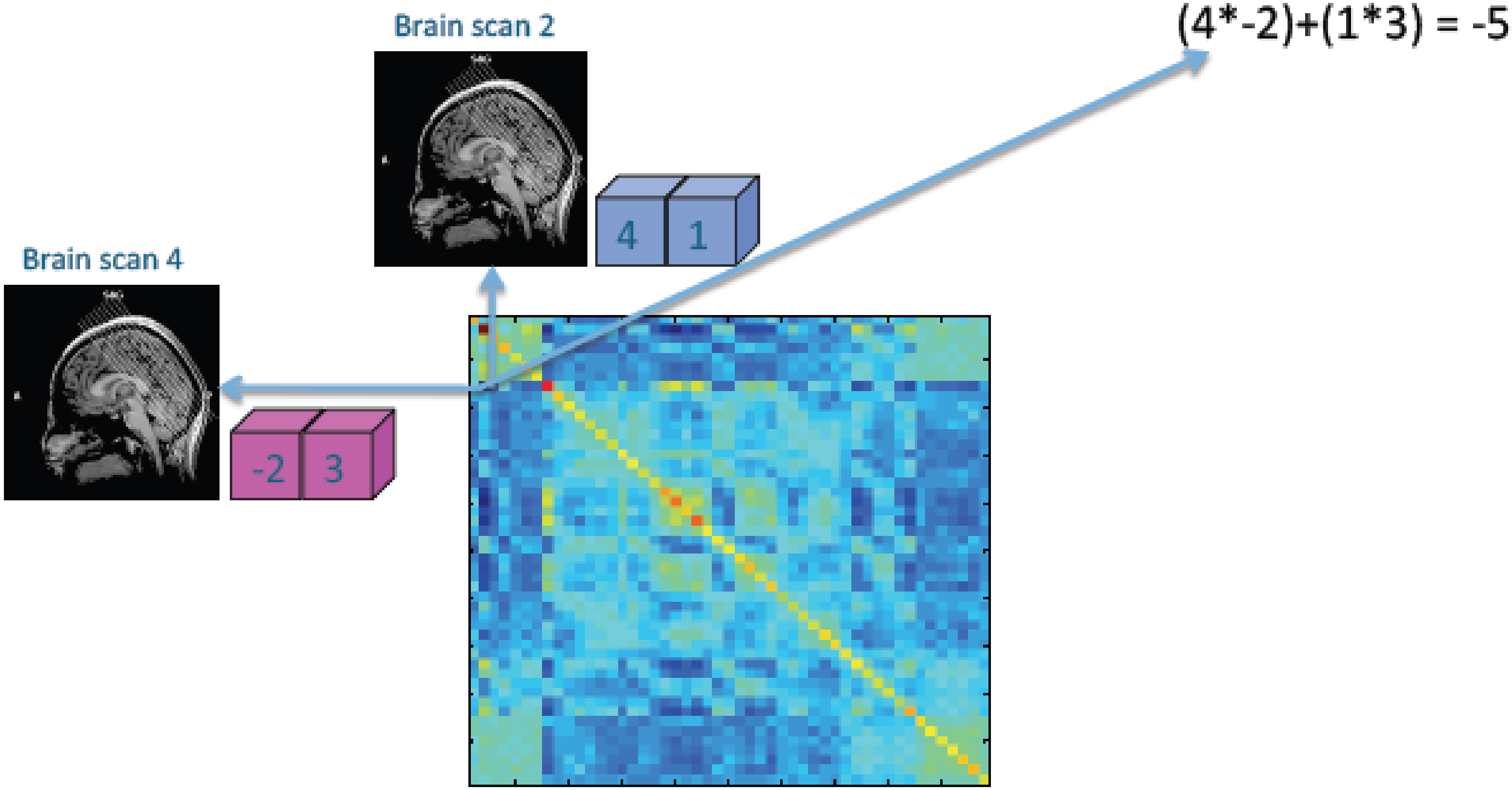


Nelineární SVM – ukázka



Nelineární SVM – jádro

Anglicky: kernel



Nelineární SVM

- transformace do nového prostoru může proběhnout navýšením počtu proměnných (např. přidáním kvadratických forem původních proměnných, tedy soubor pak bude obsahovat proměnné $\mathbf{x}_1, \mathbf{x}_1^2, \mathbf{x}_2, \mathbf{x}_2^2, \dots, \mathbf{x}_p, \mathbf{x}_p^2$)
- tento přístup však výpočetně náročný → použití jader (*kernels*)
- u lineárního SVM pro lineárně separabilní i neseparabilní třídy lze Lagrangeovu funkci přepsat do podoby

$$L(\mathbf{w}, w_0, \boldsymbol{\lambda}) = \sum_{k=1}^{N} \lambda_k - \frac{1}{2}\sum_{i,j} \lambda_i \lambda_j \delta_{x_i} \delta_{x_j} \mathbf{x}_i^T \mathbf{x}_j$$

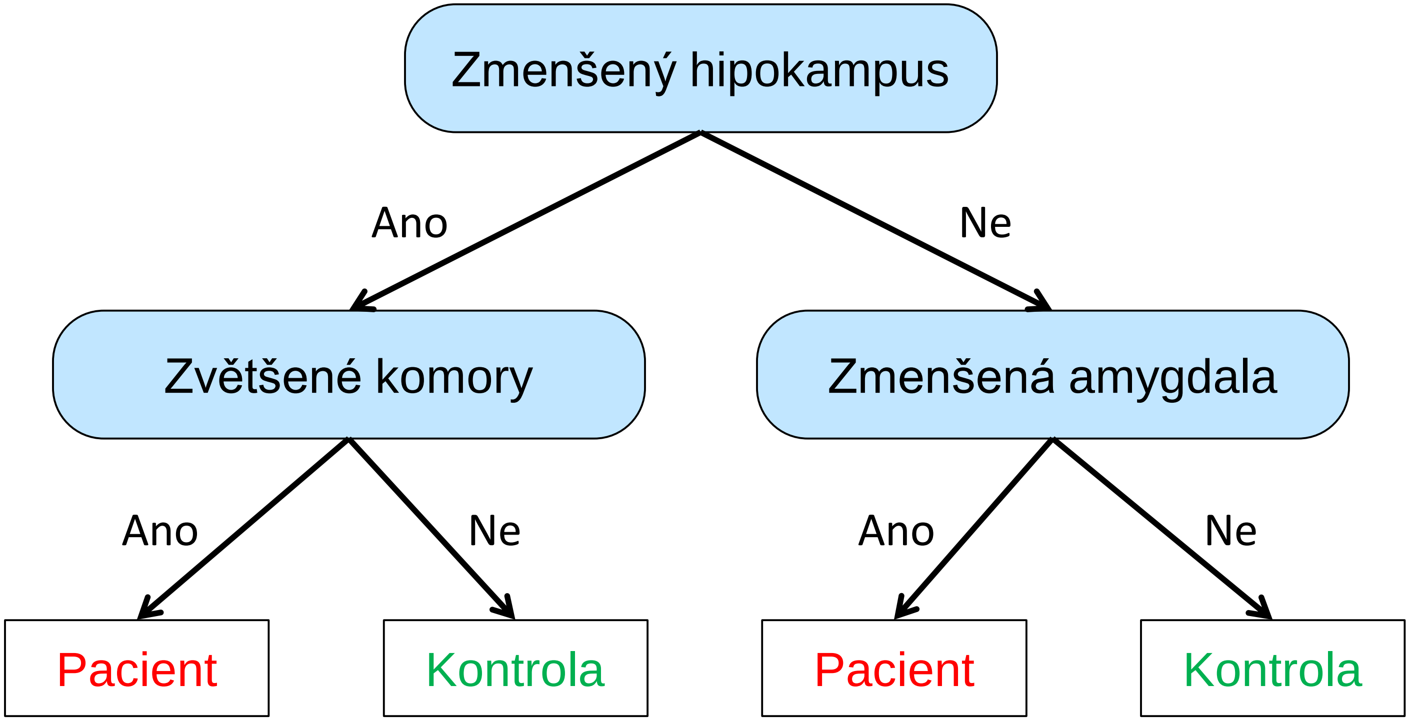
- kde si skalární součin $\mathbf{x}_i^T \mathbf{x}_j$ můžeme zapsat obecně jako $K(\mathbf{x}_i, \mathbf{x}_j)$, kde K je nějaká funkce, kterou nazveme jádro
- typy jader:
  - lineární jádro: $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$ → lineární SVM
  - polynomiální jádro stupně *d*: $K(\mathbf{x}_i, \mathbf{x}_j) = \left(1 + \mathbf{x}_i^T \mathbf{x}_j\right)^d = \left(1 + \sum_{l=1}^{p} \mathrm{x}_{il} \mathrm{x}_{jl}\right)^d$
  - radiální bázové jádro: $K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \sum_{l=1}^{p} (\mathrm{x}_{il} - \mathrm{x}_{jl})^2\right)$
  - atd.

nelineární SVM

Další metody klasifikace

Klasifikační (rozhodovací) stromy a lesy

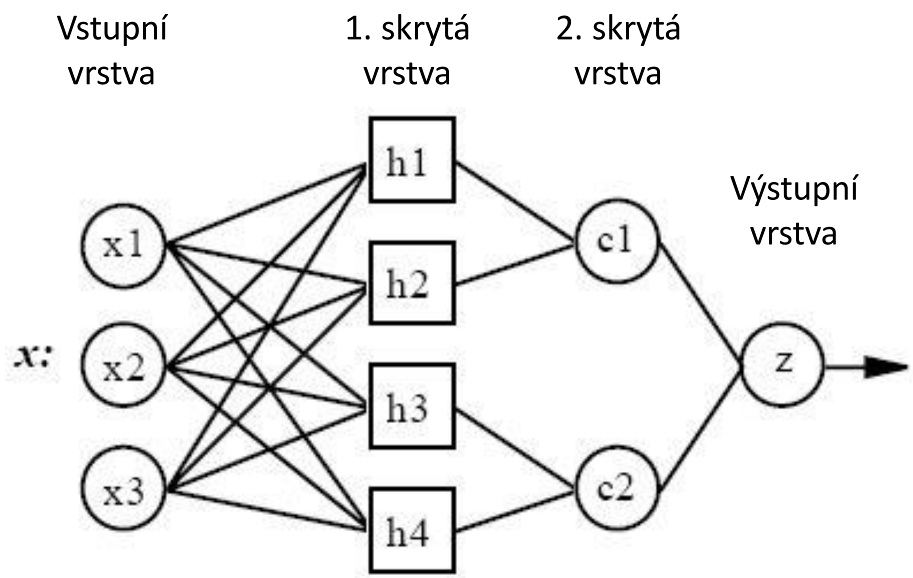
Princip: Postupné rozdělování datasetu do skupin podle hodnot jednotlivých proměnných.

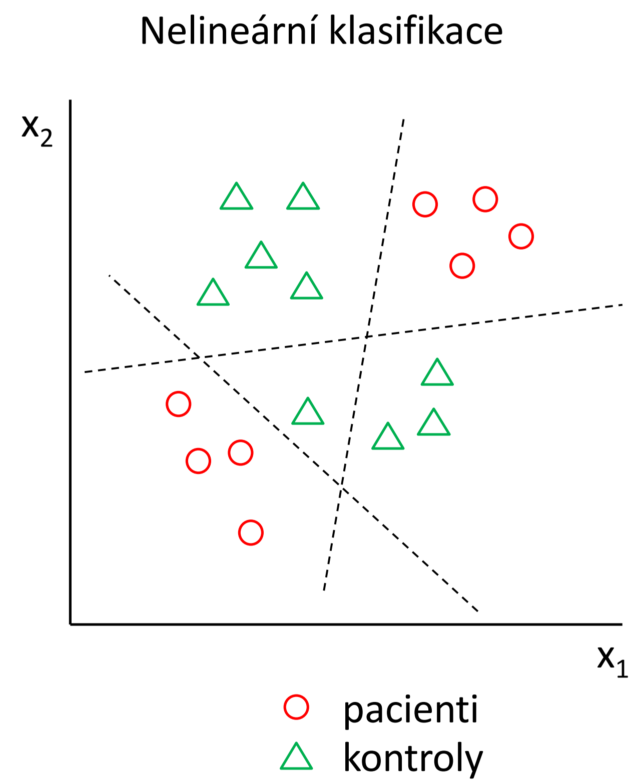


Klasifikační lesy – použití více klasifikačních stromů ke klasifikaci, každý strom zpravidla používá jen část původních dat (část subjektů nebo část proměnných).

Neuronové sítě

Princip: Postupné učení neuronové sítě (tzn. postupné nastavování vah u jednotlivých neuronů), aby byla chyba klasifikace trénovací množiny minimální. Umožňuje nelineární klasifikaci.



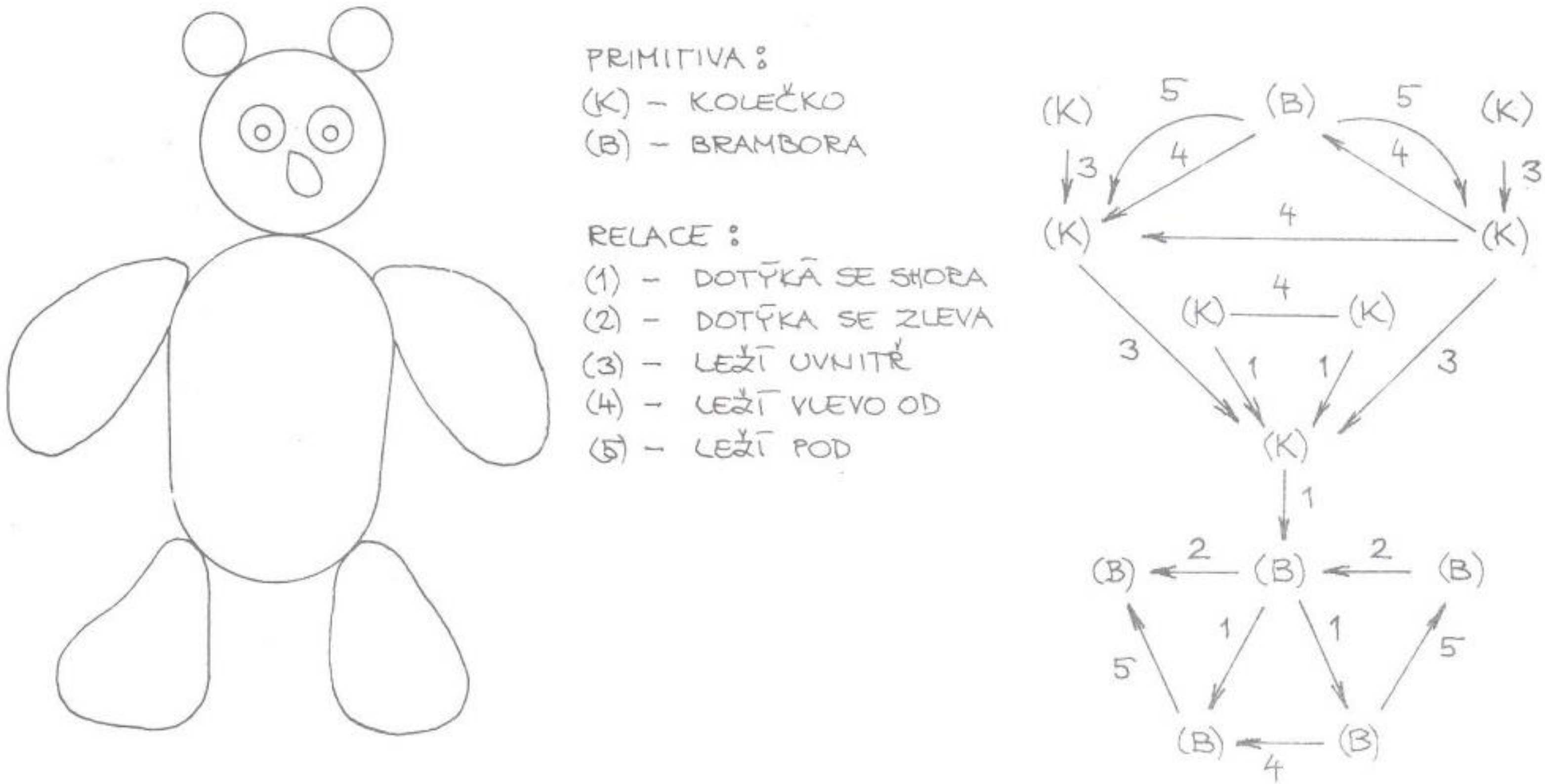


Více typů neuronových sítí – např.:

- Vícevrstvé neuronové sítě typu perceptron
- RBF (Radial Basis Function) sítě
- LVQ (Learing Vector Quantization) sítě

Strukturální (syntaktické) klasifikátory

Princip: Vstupní data popsána relačními strukturami.



Lze vytvořit i **kombinované klasifikátory** – jednotlivá primitiva doplněna příznakovým popisem.

Poznámka

- Nelze dopředu říci, která klasifikační metoda bude pro daná data fungovat nejlépe → potřebné vyzkoušet více klasifikačních metod a zvolit nejvhodnější pro daná data.
- U velkých datových souborů je obtížné dopředu určit, zda je možné data oddělit lineárně nebo ne → potřebné vyzkoušet lineární i nelineární klasifikační metody.

Hodnocení úspěšnosti klasifikace a srovnání klasifikátorů

Hodnocení úspěšnosti klasifikace - úvod

Vstupní data

Subjekt	voxel 1	voxel 2	voxel 3	...	Skutečnost (správná třída)
1					pacient
2					pacient
3					pacient
4					kontrola
5					kontrola
6					kontrola

Výsledek klasifikace

Výsledek klasifikace
pacient
pacient
kontrola
kontrola
pacient
kontrola

Jak dobrá je klasifikační metoda, kterou jsme použili?

Hodnocení úspěšnosti klasifikace

Matice záměn (konfusní matice, confusion matrix):

		Skutečnost (správná třída)	
		Pacienti (+)	Kontroly (-)
Výsledek klasifikace	Pacienti (+)	**TP**	**FP**
	Kontroly (-)	**FN**	**TN**

TP („true positive“) – kolik výsledků bylo skutečně pozitivních
(tzn. kolik pacientů bylo správně diagnostikováno jako pacienti).

FP („false positive“) – kolik výsledků bylo falešně pozitivních
(tzn. kolik zdravých lidí bylo chybně diagnostikováno jako pacienti).

FN („false negative“) – kolik výsledků bylo falešně negativních
(tzn. kolik pacientů bylo chybně diagnostikováno jako zdraví).

TN („true negative“) – kolik výsledků bylo skutečně negativních
(tzn. kolik zdravých lidí bylo správně diagnostikováno jako zdraví).

Hodnocení úspěšnosti klasifikace

		Skutečnost (správná třída)	
		Pacienti (+)	Kontroly (-)
Výsledek klasifikace	Pacienti (+)	TP	FP
	Kontroly (-)	FN	TN
		TP+FN	FP+TN

Senzitivita (sensitivity)

TP / (TP+FN)

Specificita (specificity)

TN / (FP+TN)

Celková správnost (accuracy): (TP+TN)/(TP+FP+FN+TN)

Chyba (error): (FP+FN)/(TP+FP+FN+TN)

Příklad – klasifikace pomocí FLDA

Subjekt	Skuteč-nost	Výsledek LDA
1	P	P
2	P	P
3	P	K
4	K	K
5	K	P
6	K	K

Výsledek klasifikace	Skutečnost (správná třída)	
	Pacienti (+)	Kontroly (-)
Pacienti (+)	TP=2	FP=1
Kontroly (-)	FN=1	TN=2

Senzitivita: TP/(TP+FN)=2/(2+1)=0,67

Specificita: TN/(FP+TN)=2/(1+2)=0,67

Správnost: (TP+TN)/(TP+FP+FN+TN)=(2+2)/(2+1+1+2)=0,67

Chyba: (FP+FN)/(TP+FP+FN+TN)=(1+1)/(2+1+1+2)=0,33

Intervaly spolehlivosti pro celkovou správnost

- celková správnost: $\frac{TP+TN}{TP+FP+FN+TN} = 1 - \frac{N_{error}}{N}$

- z toho plyne: $\hat{P}_A = 1 - \hat{P}_E = \frac{N_{cor}}{N}$ (tedy $N_{cor} \sim Bi(N, P_A)$)

- za splnění předpokladů, že $\hat{P}_A \cdot N > 5$, $\left(1 - \hat{P}_A\right) \cdot N > 5$ a $N > 30$, lze spočítat 95% interval spolehlivosti pro správnost pomocí aproximace na normální rozdělení:

$$\left[\hat{P}_A - 1{,}96 \cdot \sqrt{\frac{\hat{P}_A\left(1-\hat{P}_A\right)}{N}}; \hat{P}_A + 1{,}96 \cdot \sqrt{\frac{\hat{P}_A\left(1-\hat{P}_A\right)}{N}}\right]$$

Příklad – pokračování

Správnost: (TP+TN)/(TP+FP+FN+TN) = 0,67

IS pro správnost:
$$\left[\hat{P}_A - 1{,}96 \cdot \sqrt{\frac{\hat{P}_A\left(1-\hat{P}_A\right)}{N}};\ \hat{P}_A + 1{,}96 \cdot \sqrt{\frac{\hat{P}_A\left(1-\hat{P}_A\right)}{N}}\right]$$

$$\left[0{,}66 - 1{,}96 \cdot \sqrt{\frac{0{,}66\left(1-0{,}66\right)}{6}};0{,}66 + 1{,}96 \cdot \sqrt{\frac{0{,}66\left(1-0{,}66\right)}{6}}\right]$$

$$\left[0{,}29;1{,}00\right]$$

Trénovací a testovací data

1. resubstituce

2. náhodný výběr s opakováním (bootstrap)

3. predikční testování externí validací (hold-out)

4. křížová validace (cross validation)
   - *k*-násobná (*k*-fold)
   - „odlož-jeden-mimo“ (leave-one-out, jackknife)

1. resubstituce

- stejná trénovací a testovací množina

- **výhody:**
  + jednoduché
  + rychlé

- **nevýhody:**
  - příliš optimistické výsledky!!!

2. náhodný výběr s opakováním (bootstrap)

- náhodně vybereme N subjektů s opakováním jako trénovací data (tzn. subjekty se v trénovací sadě mohou opakovat) a zbylé subjekty (ani jednou nevybrané) použijeme jako testovací data
- pro rozumně velká data se vybere zhruba 63,2% subjektů pro učení a 36,8% subjektů pro testování
- trénování a testování se provede jen jednou

- **výhody:**
  \+ velká trénovací sada
  \+ rychlé

- **nevýhody:**
  \- data se v trénovací sadě opakují
  \- výsledek vcelku závislý na výběru trénovacích dat

3. predikční testování externí validací (hold-out)

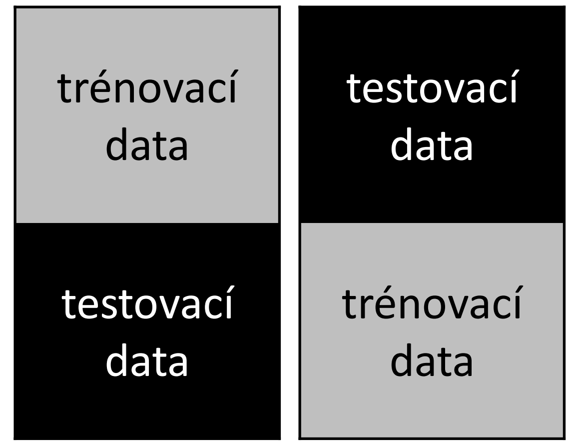
- použití části dat (většinou dvou třetin) na trénování a zbytku dat (třetiny) na testování

- **výhody:**
  + nezávislá trénovací a testovací sada

- **nevýhody:**
  - méně dat pro trénování i testování
  - výsledek velmi závislý na výběru trénovacích dat

3. predikční testování externí validací (hold-out) – modifikace 1

- použití části dat (obvykle poloviny) pro trénování a zbytku (poloviny) pro testování a následné přehození testovací a trénovací sady → zprů-měrování 2 výsledků klasifikace

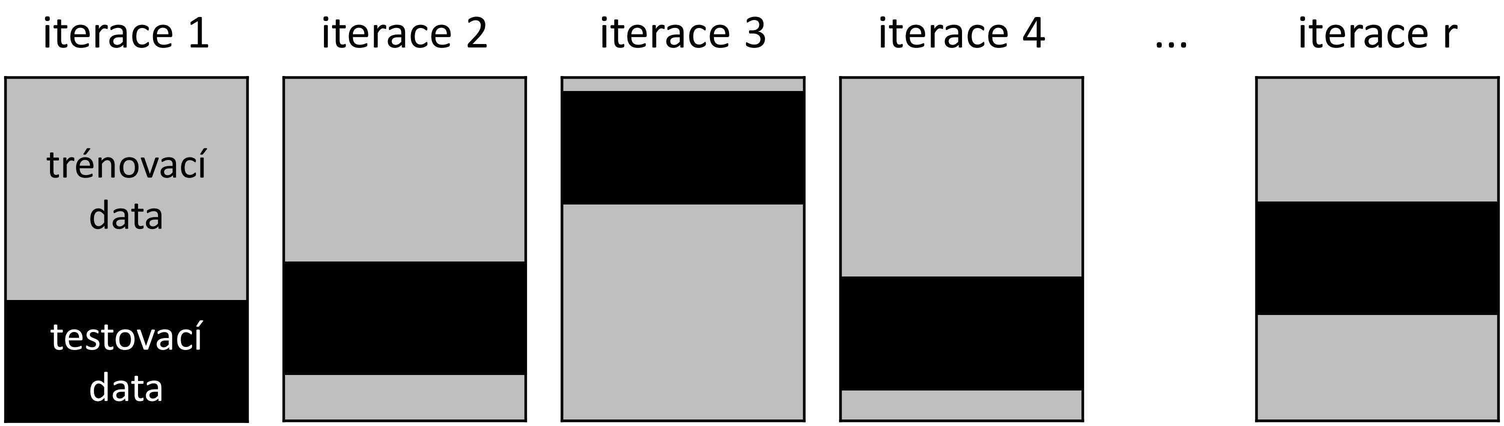


- **výhody:**
  \+ nezávislá trénovací a testovací sada

- **nevýhody:**
  \- při malých souborech může být polovina dat pro trénování příliš málo
  \- výsledek velmi závislý na výběru trénovacích dat (i když trochu méně než předtím)

3. predikční testování externí validací (hold-out) – modifikace 2

- *r*-krát náhodně rozdělíme soubor na trénovací a testovací data (většinou dvě třetiny pro trénování a třetinu pro testování) a *r* výsledků zprůměrujeme



- **výhody:**
  + poměrně přesný odhad úspěšnosti klasifikace
- **nevýhody:**
  - trénovací i testovací sady se překrývají
  - časově náročné

4. *k*-násobná křížová validace (*k*-fold cross validation)

- používán též název příčná validace
- rozdělení souboru na *k* částí, 1 část použita na testování a zbylých *k*-1 částí na trénování → postup se opakuje (všechny části 1x použity pro testování)
- speciálním případem je „odlož-jeden-mimo“ (leave-one-out) CV (pro *k*=N)

např. pro *k*=5:

iterace 1	iterace 2	iterace 3	iterace 4	iterace 5
testování	trénování	trénování	trénování	trénování
trénování	testování	trénování	trénování	trénování
trénování	trénování	testování	trénování	trénování
trénování	trénování	trénování	testování	trénování
trénování	trénování	trénování	trénování	testování

- **výhody:**
  + testovací sady se nepřekrývají
  + poměrně přesný odhad úspěšnosti klasifikace

- **nevýhody:**
  - časově náročné

„odlož-jeden-mimo“ křížová validace

- anglický překlad: leave-one-out (nebo jackknife)
- pro k=N (tzn. v každé z N iterací je jeden subjekt použit na testování a zbylých N-1 subjektů na trénování)
- platí výhody a nevýhody zmíněné u *k*-násobné křížové validace se čtyřmi komentáři:
    - časově nejnáročnější ze všech možných k
    - velmi vhodná pro malé soubory dat
    - na rozdíl od jakékoliv *k*-fold CV dostaneme vždy pouze jeden výsledek úspěšnosti (tzn. výsledek úspěšnosti nezávisí na tom, jak se jednotlivé subjekty „namíchají“ do jednotlivých skupin)
    - v některých článcích se uvádí, že lehce nadhodnocuje úspěšnost → doporučuje se 10-násobná křížová validace

Příklad - „odlož-jeden-mimo“ křížová validace

Iterace:	iter. 1	iter. 2	iter. 3	iter. 4	iter. 5	iter. 6
	1	1	1	1	1	1
	2	**2**	2	2	2	2
	3	3	**3**	3	3	3
	4	4	4	**4**	4	4
	5	5	5	5	**5**	5
	6	6	6	6	6	**6**
Skutečnost:	pacient	pacient	pacient	kontrola	kontrola	kontrola
Výsledek klasifikace:	pacient	kontrola	kontrola	kontrola	pacient	kontrola

Výsledek klasifikace	Skutečnost pac.	Skutečnost kont.
pacient	TP=1	FP=1
kontrola	FN=2	TN=2

Senzitivita: 1/(1+2)=0,33

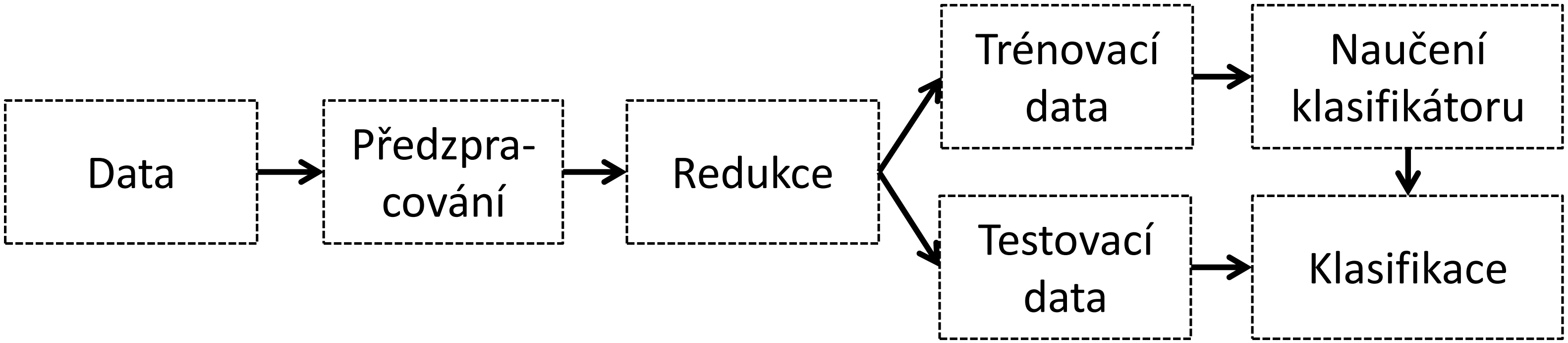
Specificita: 2/(1+2)=0,67

Správnost: (1+2)/(1+1+2+2)=0,50

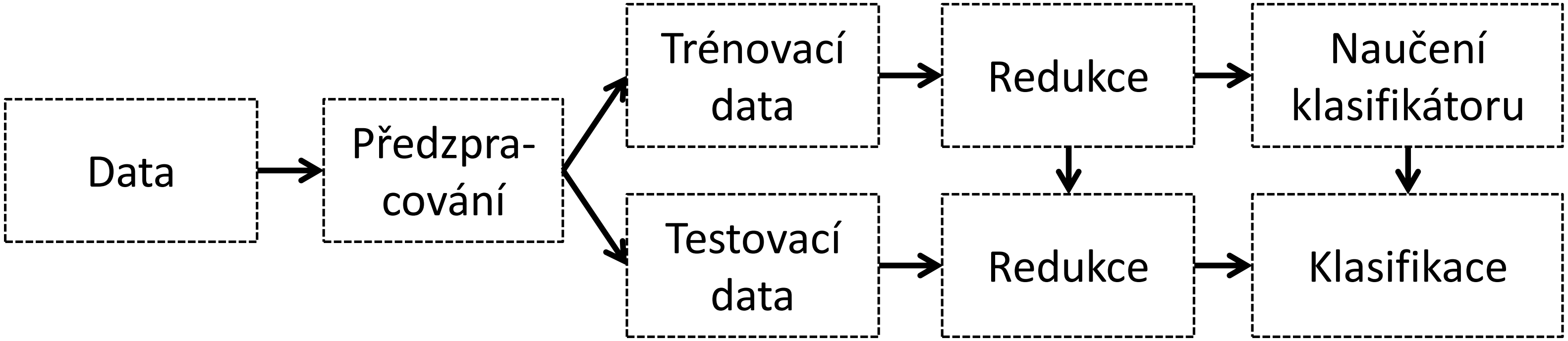
Chyba: (1+2)/(1+1+2+2)=0,50

Upozornění !!!

Postup 1:



Postup 2:



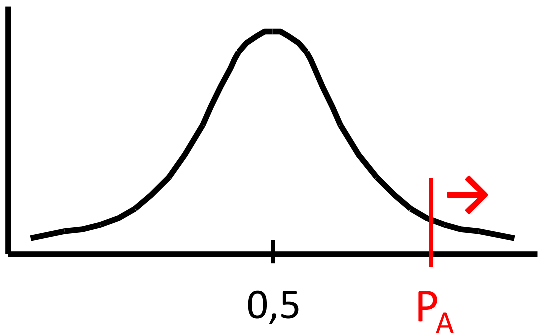
Postup 1 je nesprávný, je potřebné rozdělit soubor na trénovací a testovací ještě před redukcí dat, jinak dostaneme nadhodnocené výsledky!!!

Je klasifikace lepší než náhodná klasifikace?

- permutační testování
- jednovýběrový binomický test

Permutační testování

- r-krát náhodně přeházíme identifikátory příslušnosti do skupin u subjektů a provedeme klasifikaci (se stejným nastavením jako při použití originálních dat)
- p-hodnota se vypočte jako: $^{n}/_{r}$, kde *n* je počet iterací, v nichž byla úspěšnost klasifikace (např. celková správnost) vyšší nebo rovna úspěšnosti klasifikace originálních dat (P_A)
- pozn. pokud histogram z r celkových správností získaných permutacemi neleží kolem 0,5, máme v algoritmu zřejmě někde chybu!



Jednovýběrový binomický test

- testujeme, zda se liší celková správnost (což je podíl správně zařazených subjektů) od správnosti získané náhodnou klasifikací
- správnost u náhodné klasifikace: $P_{A_0} = {}^{N_i}/_{N}$, kde N_i je počet subjektů nejpočetnější skupiny
- $z = \frac{P_A - P_{A_0}}{\sqrt{\left(P_{A_0}(1-P_{A_0})\right)/N}}$
- Pokud $|z| > 1{,}96$, zamítáme nulovou hypotézu o shodnosti správnosti naší klasifikace a správnosti náhodné klasifikace

Příklad – jednovýběrový binomický test

- uvažujme např. výsledek klasifikace pacientů a kontrol pomocí LDA (pomocí resubstituce): $P_A = 0{,}67$, $N = 6$, $P_{A_0} = {}^{N_i}/_{N} = 0{,}5$

- $z = \frac{P_A - P_{A_0}}{\sqrt{\left(P_{A_0}(1-P_{A_0})\right)/N}} = \frac{0{,}67-0{,}5}{\sqrt{\left(0{,}5(1-0{,}5)\right)/6}} = 0{,}83$

- protože $|z|$ <1,96, nezamítáme nulovou hypotézu o shodnosti správnosti naší klasifikace a správnosti náhodné klasifikace (tzn. neprokázali jsme, že by naše klasifikace byla lepší než náhodná klasifikace)

- nezamítnutí nulové hypotézy vyplývá už i z vypočteného intervalu spolehlivosti (0,29 – 1,00), protože tento interval spolehlivosti obsahuje hodnotu 0,5

Srovnání úspěšnosti klasifikace

- Srovnání 2 klasifikátorů
- Srovnání 3 a více klasifikátorů

Srovnání 2 klasifikátorů

Klasifikátor 1	Klasifikátor 2 Správně (1)	Klasifikátor 2 Chybně (0)
Správně (1)	N_{11}	N_{10}
Chybně (0)	N_{01}	N_{00}

Celkem:

$$N_{11} + N_{10} + N_{01} + N_{00} = N_{ts}$$

McNemarův test: $$\chi^2 = \frac{(|N_{01} - N_{10}| - 1)^2}{N_{01} + N_{10}}$$

Pokud $\chi^2 > 3{,}841$, zamítáme nulovou hypotézu H_0 o shodnosti celkové správnosti klasifikace pomocí dvou klasifikátorů

Dvouvýběrový binomický test:

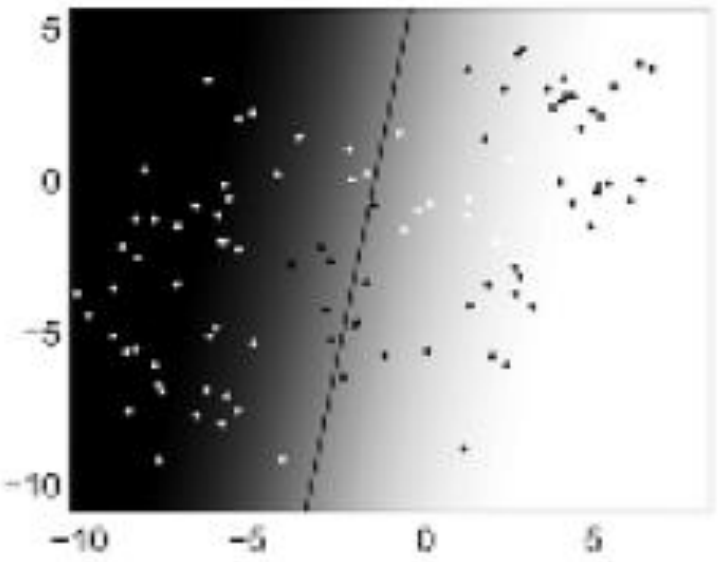
$$z = \frac{p_1 - p_2}{\sqrt{(2p(1-p))/(N_{ts})}} \qquad p_1 = \frac{N_{11} + N_{10}}{N_{ts}}; \qquad p_2 = \frac{N_{11} + N_{01}}{N_{ts}} \qquad p = \frac{1}{2}(p_1 + p_2)$$

Pokud $|z| > 1{,}96$, zamítáme nulovou hypotézu H_0 o shodnosti podílu správně klasifikovaných subjektů dvou klasifikátorů

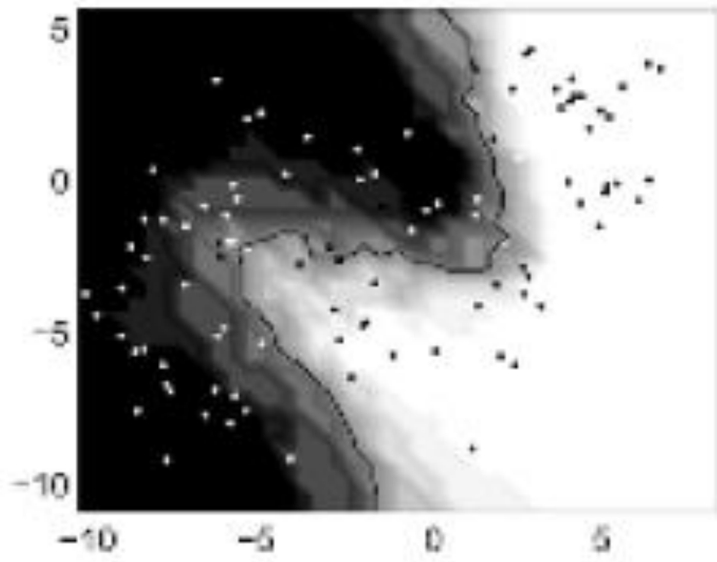
Dvouvýb. binomický test předpokládá nezávislost (tzn. že každý klasifikátor byl testován na jiném testovacím souboru) → raději používat McNemarův test

Příklad – srovnání 2 klasifikátorů

Lineární diskriminační analýza (LDA)



Metoda 9 nejbližších sousedů (9-nn)



Příklad – srovnání 2 klasifikátorů

Matice záměn:

LDA		9-nn	
42	8	44	6
8	42	2	48
84% správnost		92% správnost	

Shody u klasifikátorů:

Klasifikátor 1: LDA	Klasifikátor 2: 9-nn Správně (1)	Klasifikátor 2: 9-nn Chybně (0)
Správně (1)	$N_{11} = 82$	$N_{10} = 2$
Chybně (0)	$N_{01} = 10$	$N_{00} = 6$

McNemarův test:

$$x^2 = \frac{(|10-2|-1)^2}{10+2} = \frac{49}{12} \approx \mathbf{4.0833}$$

Protože $\chi^2 > 3{,}841$, zamítáme H_0.

Dvouvýb. binomický test:

$$z = \frac{0.84 - 0.92}{\sqrt{(2 \times 0.88 \times 0.12)/(100)}} \approx \mathbf{-1.7408}$$

Protože $|z| < 1{,}96$, nezamítáme H_0.

Srovnání 3 a více klasifikátorů

Testuje se, zda jsou statisticky významně odlišné správnosti klasifikátorů měřené na stejných testovacích datech – tzn. H_0: $p_1 = p_2 = \cdots = p_L$, kde p_L je správnost L-tého klasifikátoru. Poté je možno srovnávat správnosti klasifikátorů vždy po dvou, aby se zjistilo, které klasifikátory se od sebe liší.

Cochranův Q test:

$$Q_C = (L-1)\frac{L\sum_{i=1}^{L} G_i^2 - T^2}{LT - \sum_{j=1}^{N_{ts}} (L_j)^2}$$

Pokud $Q_C > \chi^2\,(L-1)$, zamítáme H_0.

F-test:

$$F_{\mathrm{cal}} = \frac{MSA}{MSAB}$$

Pokud $F_{cal} > F(L-1, (L-1)\times(N_{ts}-1))$, zamítáme H_0.

Looney doporučuje F-test, protože je méně konzervativní.

S. W. Looney. A statistical technique for comparing the accuracies of several classifiers. *Pattern Recognition Letters*, 8:5–9, 1988.

Příklad – srovnání 3 a více klasifikátorů

Matice záměn:

LDA		9-nn		Parzen	
42	8	44	6	47	3
8	42	2	48	5	45
84% správnost		92% správnost		92% správnost	

Cochranův Q test:

$$Q_C = 2 \times \frac{3 \times (84^2 + 92^2 + 92^2) - 268^2}{3 \times 268 - (80 \times 9 + 11 \times 4 + 6 \times 1)} \approx \mathbf{3.7647}$$

Protože $Q_C < \chi^2 (L-1) = 5{,}991$, nezamítáme H_0.

F-test:

$$F_{\text{cal}} = \frac{0.2223}{0.0549} \approx \mathbf{4.0492}$$

Protože $F_{cal} > F(2; 198) = 3{,}09$, zamítáme H_0.

Hodnocení úspěšnosti klasifikace a srovnání klasifikátorů - shrnutí

- výpočet úspěšnosti klasifikace (správnosti, chyby, senzitivity, specificity a přesnosti) pomocí matice záměn
- výpočet intervalu spolehlivosti pro správnost a chybu
- volba trénovacího a testovacího souboru:
  - resubstituce
  - náhodný výběr s opakováním (bootstrap)
  - predikční testování externí validací (hold-out)
  - křížová validace (cross validation): k-násobná, „odlož-jeden-mimo“
- srovnání úspěšnosti klasifikace s náhodnou klasifikací
  - permutační testování
  - jednovýběrový binomický test
- srovnání úspěšnosti klasifikace 2 klasifikátorů:
  - McNemarův test
  - dvouvýběrový binomický test
- srovnání úspěšnosti klasifikace 3 a více klasifikátorů:
  - Cochranův Q test
  - F-test

Poděkování

Příprava výukových materiálů předmětu
„DSAN02 Pokročilé metody analýzy dat v neurovědách“
byla finančně podporována prostředky projektu FRMU
č. MUNI/FR/0260/2014 „Pokročilé metody analýzy dat
v neurovědách jako nový předmět na LF MU“