Internet a klasifikační metody, 3. přednáška

Hlavní typy klasifikačních metod

volitelný předmět pro magisterské studium

Martin Holeňa

O čem to bude?

- Základní rozdělení klasifikačních metod do skupin
- Klasifikační metody nehledající hranici mezi třídami
  - klasifikace na základě k nejbližších sousedů
  - odhadování pravděpodobnosti tříd: bayesovská klasifikace
  - prokládání normálních rozdělení: diskriminační analýza
- Příklad metody hledající hranice: neuronové sítě

Rozdělení klasifikačních metod

- Kriterium: hledá metoda hranici mezi třídami?

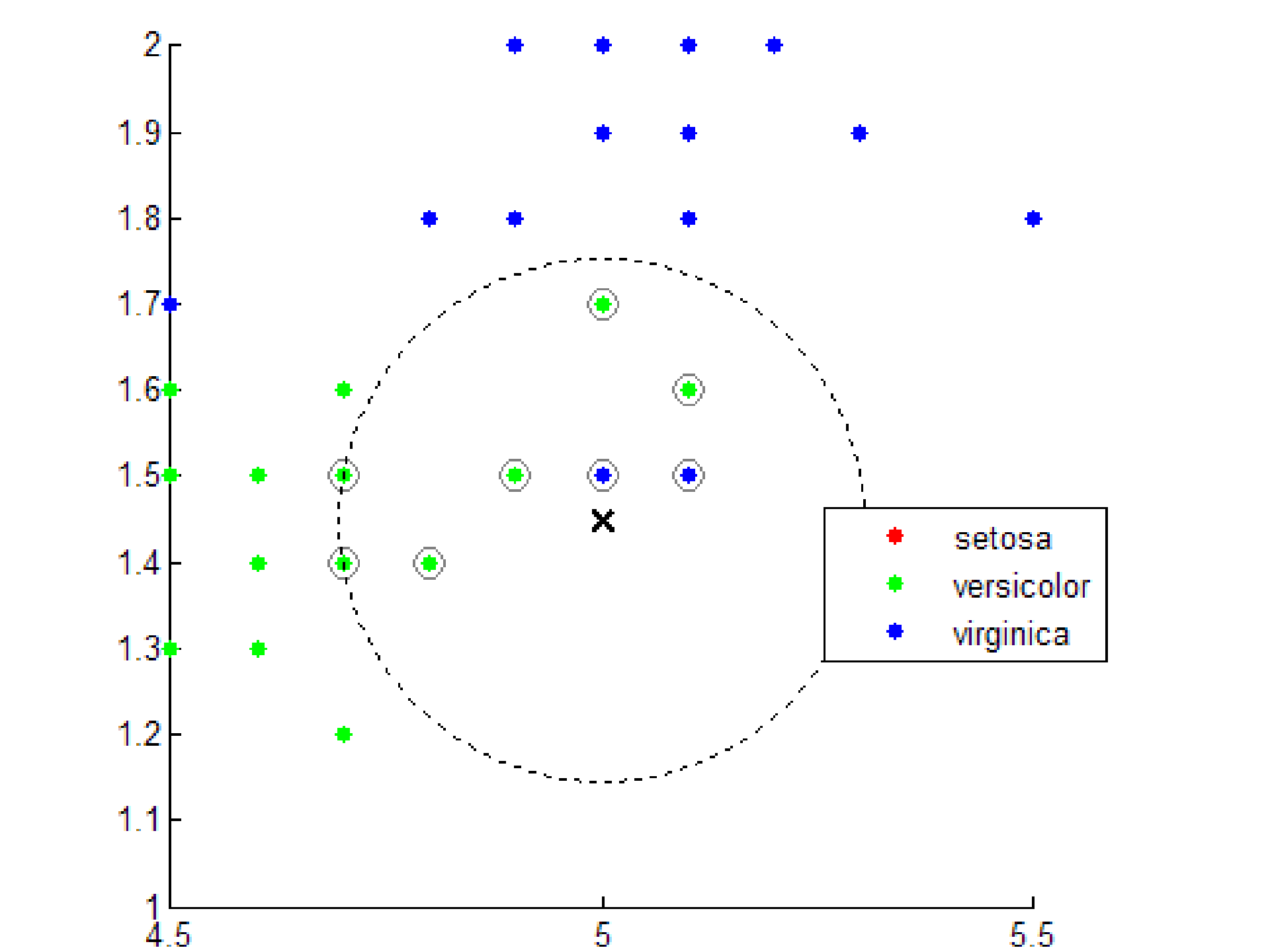
1. *Primárně nehledají hranici*, ale zkušenost podobných případů | pravděpodobnost tříd (bodové odhady, rozdělení)
   - tradiční metody (předpočítačové), výpočetně nepříliš náročné
2. *Primárně hledají hranici*: moderní, výpočetně náročné
   - např. neuronové sítě, rozhodovací stromy, SVM, ...

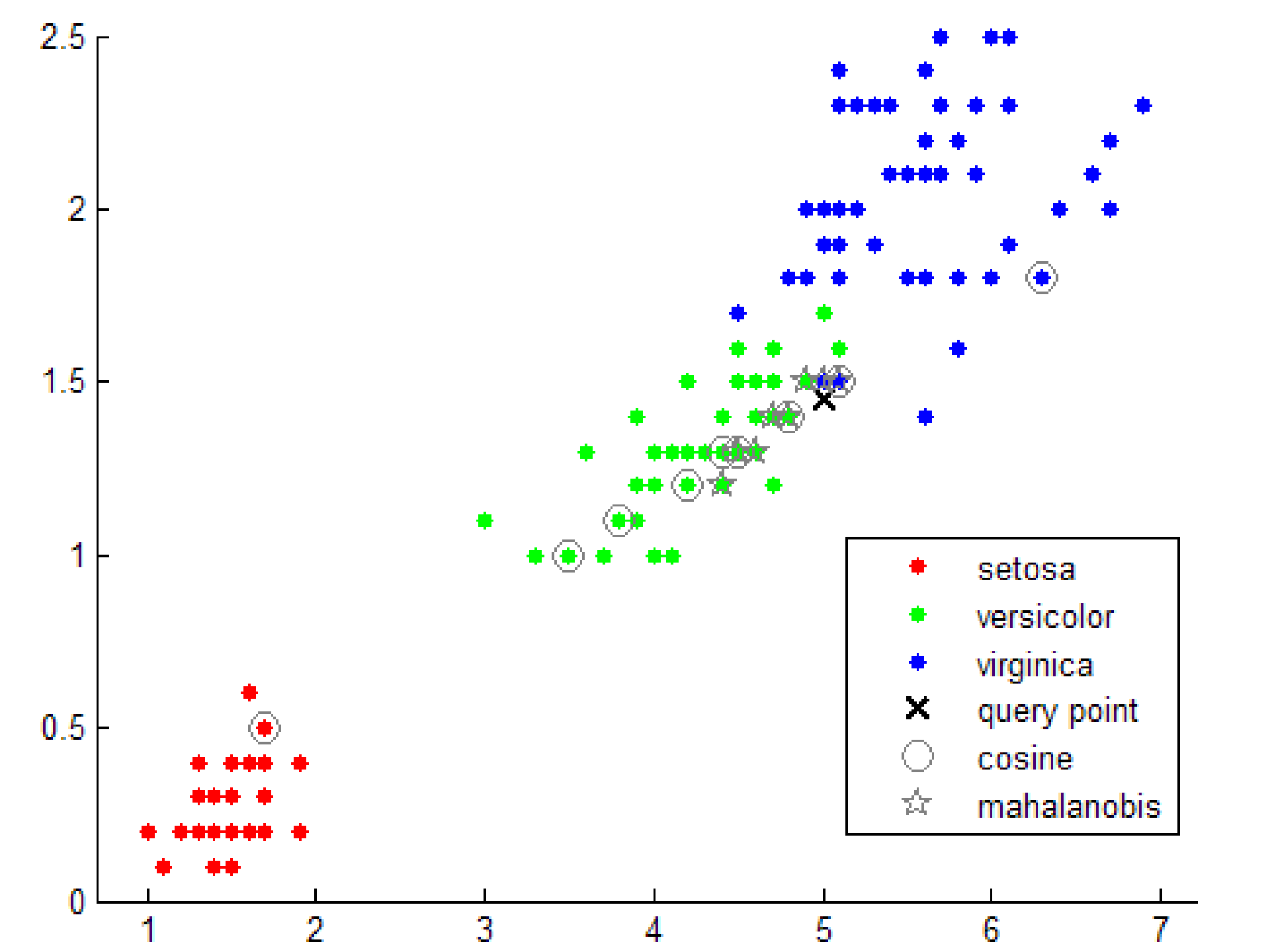
Klasifikace dle podobných případů

- Nový objekt klasifikován jako *třída nejčastější*

  u k objektů nejpodobnějších podle zvolené podobnosti

  - podmnožina objektů použitých pro učení (= zapamatování)
  - $\Rightarrow$ metoda k nejbližších sousedů (k nearest neigbours, k-NN)


2
1.9
1.8
1.7
1.6
1.5
1.4
1.3
1.2
1.1
1
4.5
5
5.5
setosa
versicolor
virginica


2.5
2
1.5
1
0.5
0
1
2
3
4
5
6
7
setosa
versicolor
virginica
query point
cosine
mahalanobis

Klasifikace dle podobných případů

- Nový objekt klasifikován jako *třída nejčastější*

  u k objektů nejpodobnějších podle zvolené podobnosti

  - podmnožina objektů použitých pro učení (= zapamatování)
  - $\Rightarrow$ metoda k nejbližších sousedů (k nearest neigbours, k-NN)

- Často silná závislost na volbě k

  - *volba k* z konečného počtu možností: krosvalidací

Podobnosti používané u k-NN

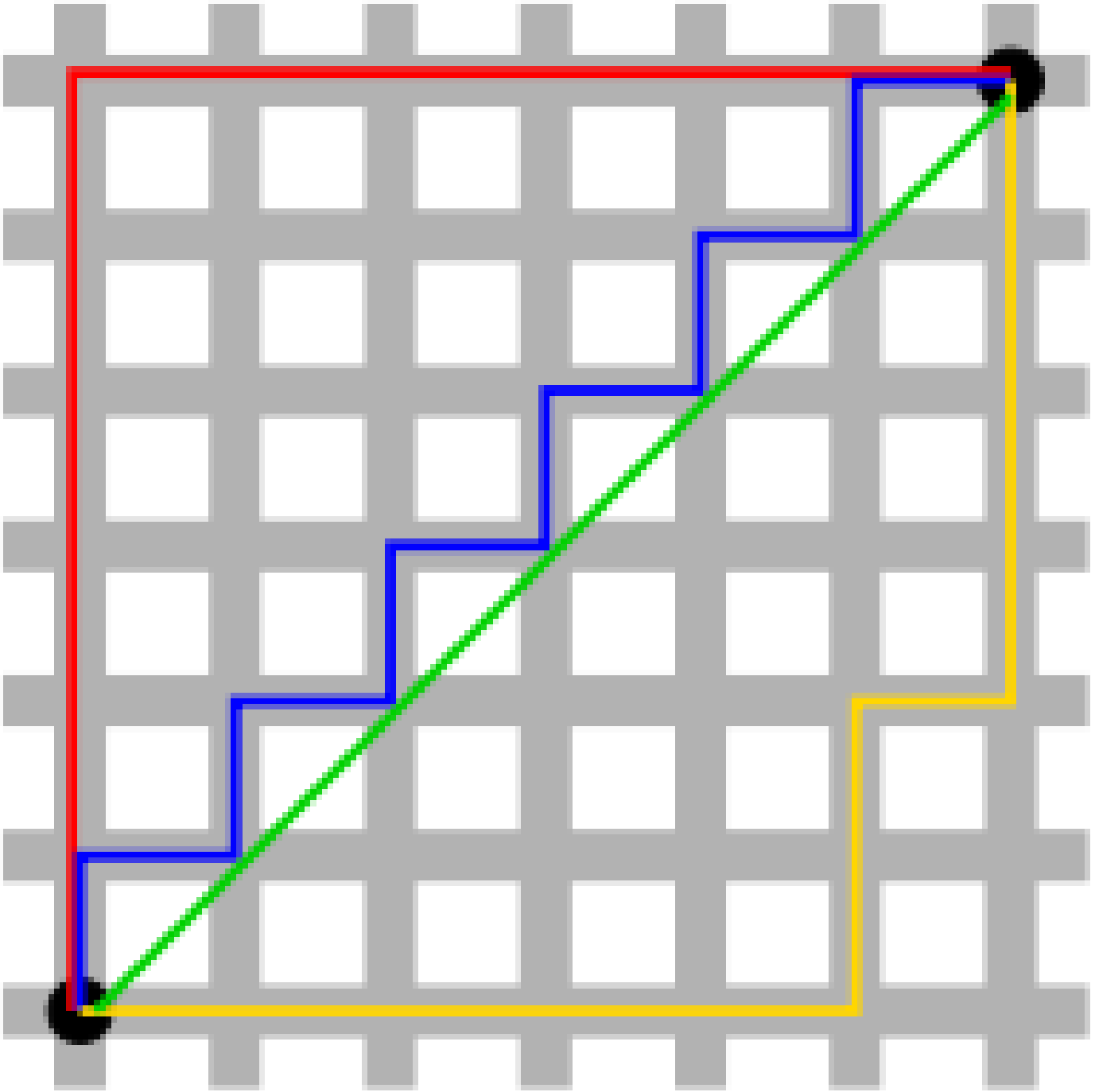
- Často podobnost x a $y \sim \frac{1}{d(x,y)}$ *vzdálenost*, obvykle $d(x,y) = \|x - y\|$

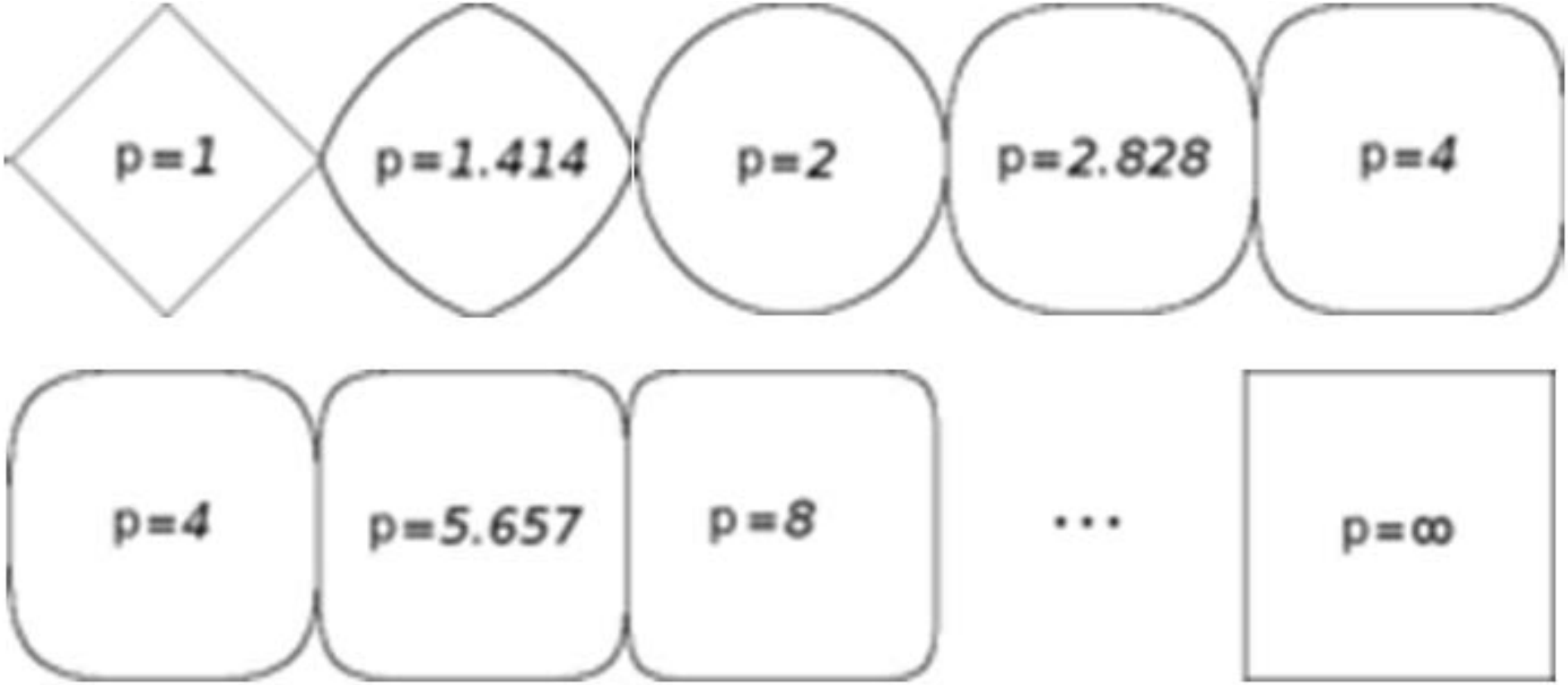
1. $\|x\|_p = \|(x_1,\ldots,x_n)\|_p = \sqrt[p]{\sum_{i=1}^{n} |x_i|^p}$, $p \in [1,+\infty]$ $(p = +\infty: \max_i |x_i|)$

p=1:

citi-block

vzdálenost




p=1
p=1.414
p=2
p=2.828
p=4
p=4
p=5.657
p=8
...
p=∞

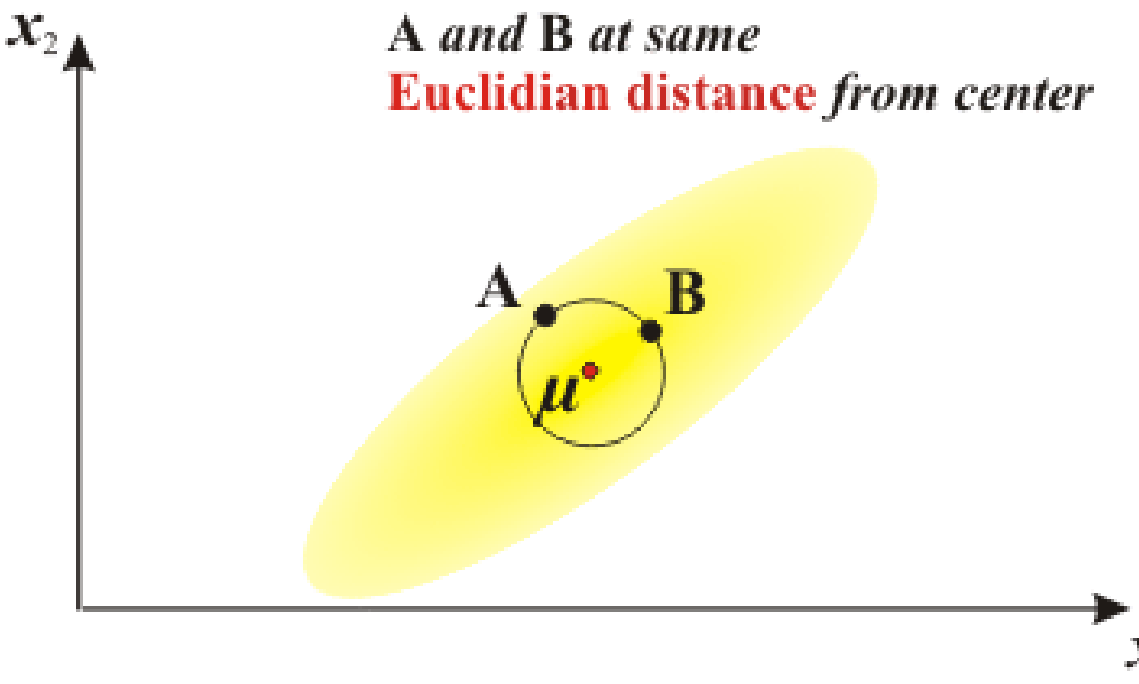
Podobnosti používané u k-NN

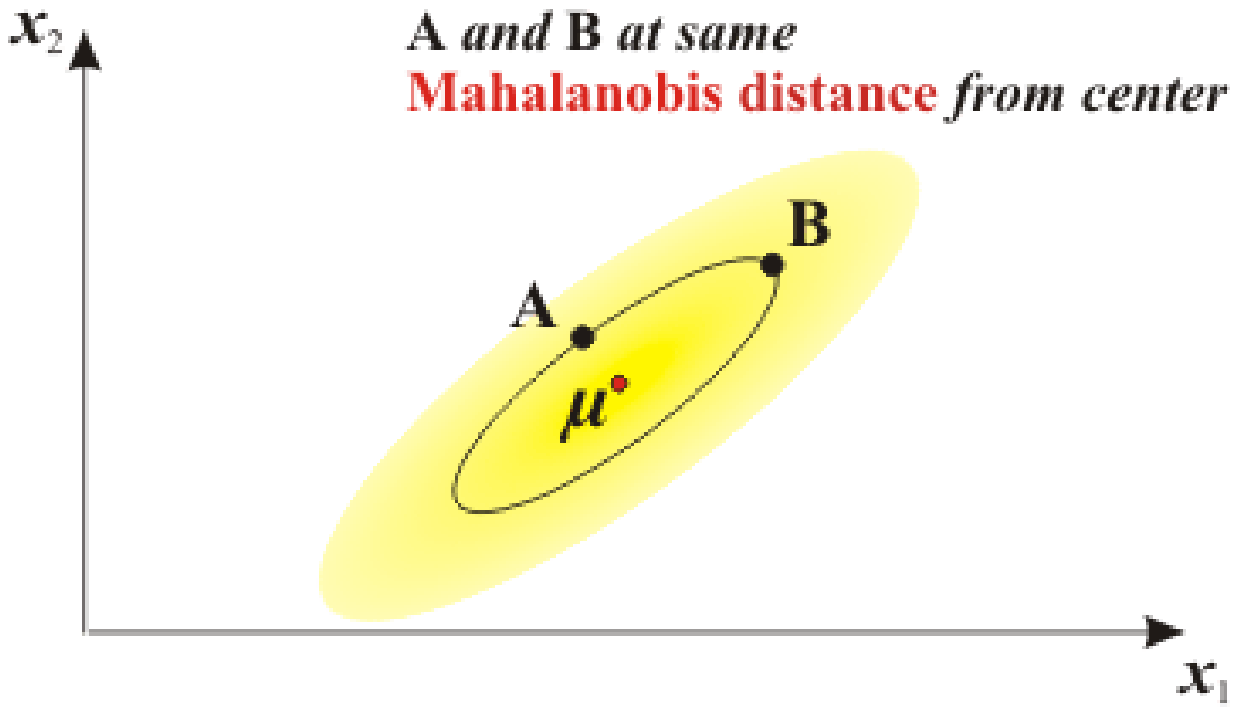
- Často podobnost x a $y \sim \frac{1}{d(x,y)}$ *vzdálenost*, obvykle $d(x,y) = \|x-y\|$

1. $\|x\|_p = \|(x_1,\ldots,x_n)\|_p = \sqrt[p]{\sum_{i=1}^n |x_i|^p}$, $p \in [1,+\infty]$ $(p = +\infty: \max_i |x_i|)$
2. *Mahalanobisova*: náhodné vektory $X, Y, \mathrm{Var}X = \mathrm{Var}Y = \Sigma$ pozitivně definitní

$$d(x,y) = \left\| \sqrt{\Sigma^{-1}}(X-\mu) - \sqrt{\Sigma^{-1}}(Y-\mu) \right\|_2 = \sqrt{(X-Y)^\top \Sigma^{-1}(X-Y)}$$

- *Hammingova* $d(x,y) = \#\{i : x_i \neq y_i\}$, *Jaccardova* podobnost $\frac{\#\{i: x_i \neq y_i\}}{\#\{i: \max(x_i, y_i) = 1\}}$
- Korelační koeficient x a y, obvykle Pearsonův: $\frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}}$


x_2
A and B at same
Euclidian distance from center
A
B
μ
x_1


x_2
A and B at same
Mahalanobis distance from center
B
A
μ
x_1

k-NN při kolaborativním filtrování

- Typicky používané *podobnosti:* spojité atributy – *Pearsonův* korelační koeficient, nominální atributy – *Jaccardova* podobnost
- Problém: *vychýlenost* (bias) doporučovaného | ostatních uživatelů
  - někteří *uživatelé* hodnotí systematicky přísněji | mírnější
  - některé *produkty* jsou obvykle nadhodnocené | podhodnocené
  - před měřením podobnosti musíme vychýlenost korigovat

k-NN při odhalování malware

- Používají se *statické vlastnosti* software (= kód),

  ne dynamické (= průběh interakcí s OS)

- Kódování vlastností: *n-gramy bytů* (bigramy, trigramy, …)
  - vhodně vybraná množina n-gramů: *profil software*

- K profilu neznámého software hledáme k nejpodobnějších

  v databázi známého software (malware + neškodného)

Odhadování pravděpodobné třídy

Odhad pravděpodobnosti náležení bodu x jednotlivým třídám

1. Pomocí *Bayesovy věty:* $P(\text{třída } C \mid \text{atributy } A) = \frac{P(A|C)P(C)}{P(A)}$ ↘apriorní
   1. naivní klasifikátor: předpokládá nezávislost $P(A_1,\ldots,A_n|C) = P(A_1|C)\cdots P(A_n|C)$
   2. ne naivní: bayesovské sítě (výpočetně náročné)
2. Pomocí *logitového modelu*: jen náležení | nenáležení $(C\ |\neg C)$

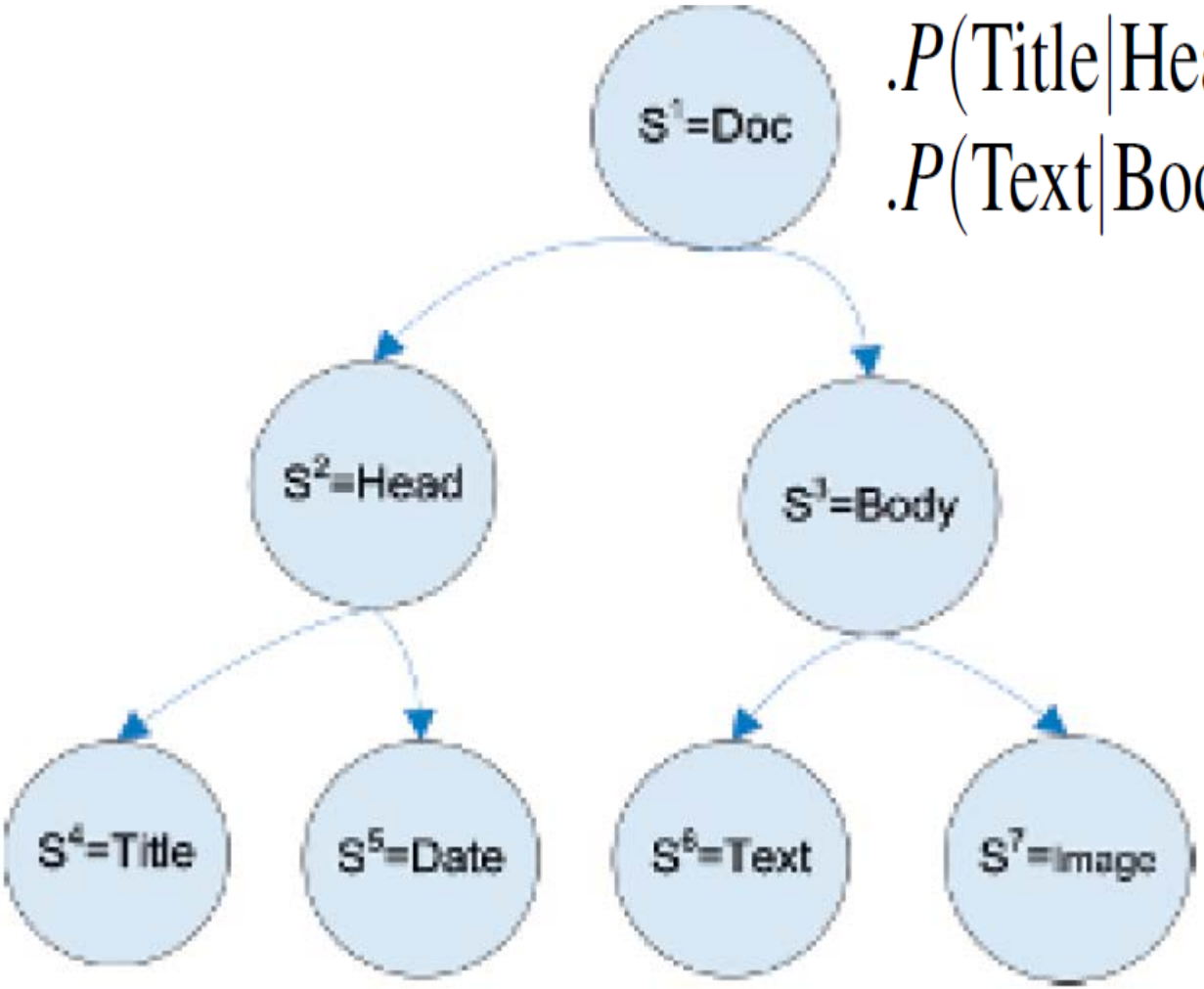
$$P(C|A_1,\ldots,A_n) = \frac{1}{1+e^{-\alpha_1 A_1 \ldots - \alpha_n A_n}}, P(\neg C|A_1,\ldots,A_n) = \frac{e^{-\alpha_1 A_1 \ldots - \alpha_n A_n}}{1+e^{-\alpha_1 A_1 \ldots - \alpha_n A_n}}$$

Bayesovské spamové filtry

- *Většinou naivní* Bayesův klasifikátor, A_i: slova, metainformace
  - ošálitelné dodatečnými slovy A_i s $P(A_i|\text{ham}) \gg P(A_i|\text{spam})$

    $\Rightarrow$ ve výsledku $P(\text{ham}|A_1, \ldots, A_n) > P(\text{spam}|A_1, \ldots, A_n)$ (*Bayesian poisoning*)
- Proto alternativně používány jednoduché bayesovské sítě.

$$P(d) = P(S^1|\theta)P(S^2|S^1,\theta)P(S^3|S^1,\theta)P(S^4|S^2,\theta)$$
$$.P(S^5|S^2,\theta)P(S^6|S^3,\theta)P(S^7|S^3,\theta)$$
$$= P(\text{Doc}|\theta)P(\text{Head}|\text{Doc},\theta)P(\text{Body}|\text{Doc},\theta)$$
$$.P(\text{Title}|\text{Head},\theta)P(\text{Date}|\text{Head},\theta)$$
$$.P(\text{Text}|\text{Body},\theta)P(\text{Image}|\text{Body},\theta)$$



Bayesovské spamové filtry

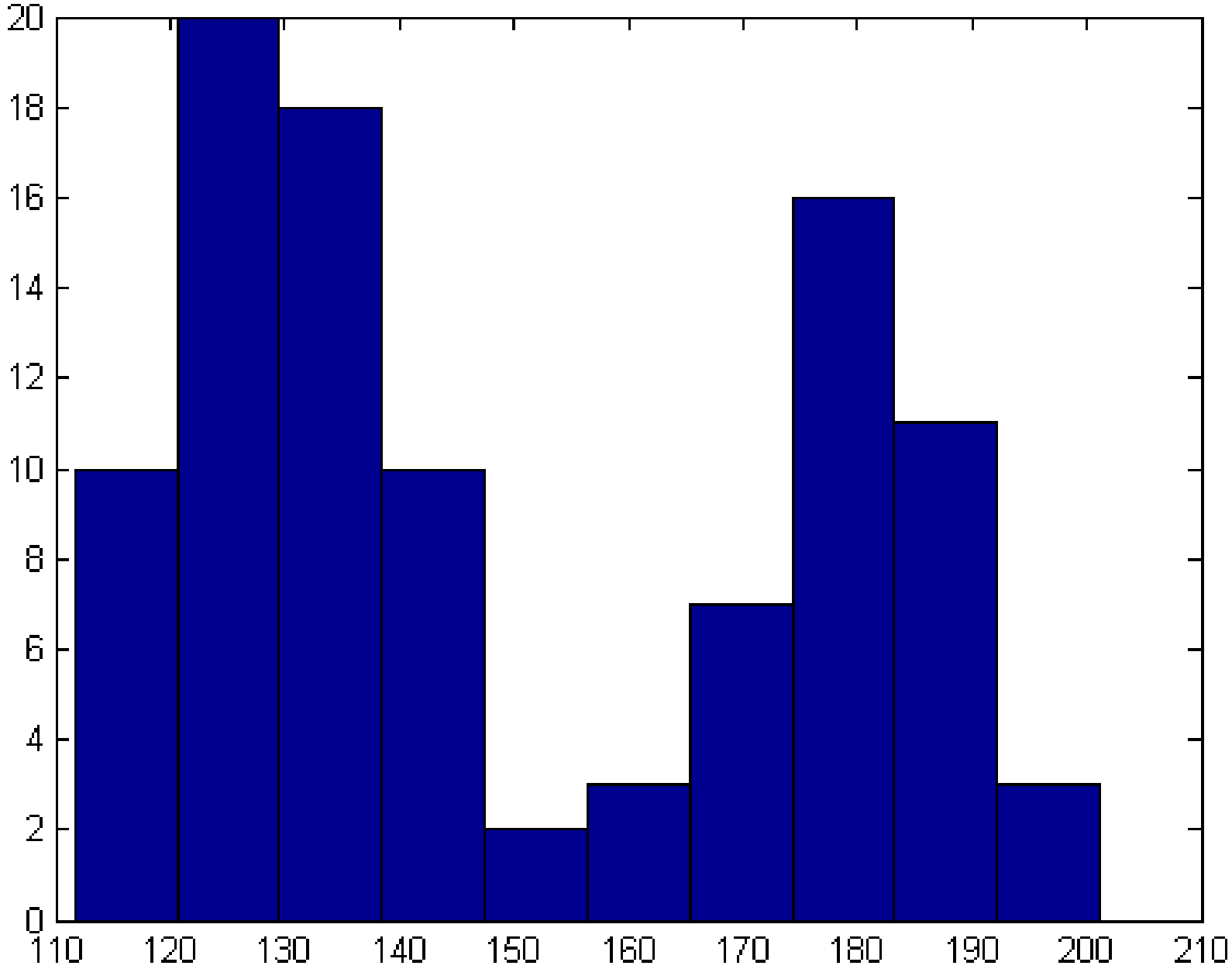
- *Většinou naivní* Bayesův klasifikátor, A_i: slova, metainformace
  - ošálitelné dodatečnými slovy A_i s $P(A_i|\text{ham}) \gg P(A_i|\text{spam})$

    $\Rightarrow$ ve výsledku $P(\text{ham}|A_1, \ldots, A_n) > P(\text{spam}|A_1, \ldots, A_n)$ (*Bayesian poisoning*)
- Proto alternativně používány jednoduché bayesovské sítě.
- Pro klienty | známější pro *příchozí servery:*

SpamAssassin, DSPAM, Bogofilter, SpamBayes (+ 3. třída: nejasné)

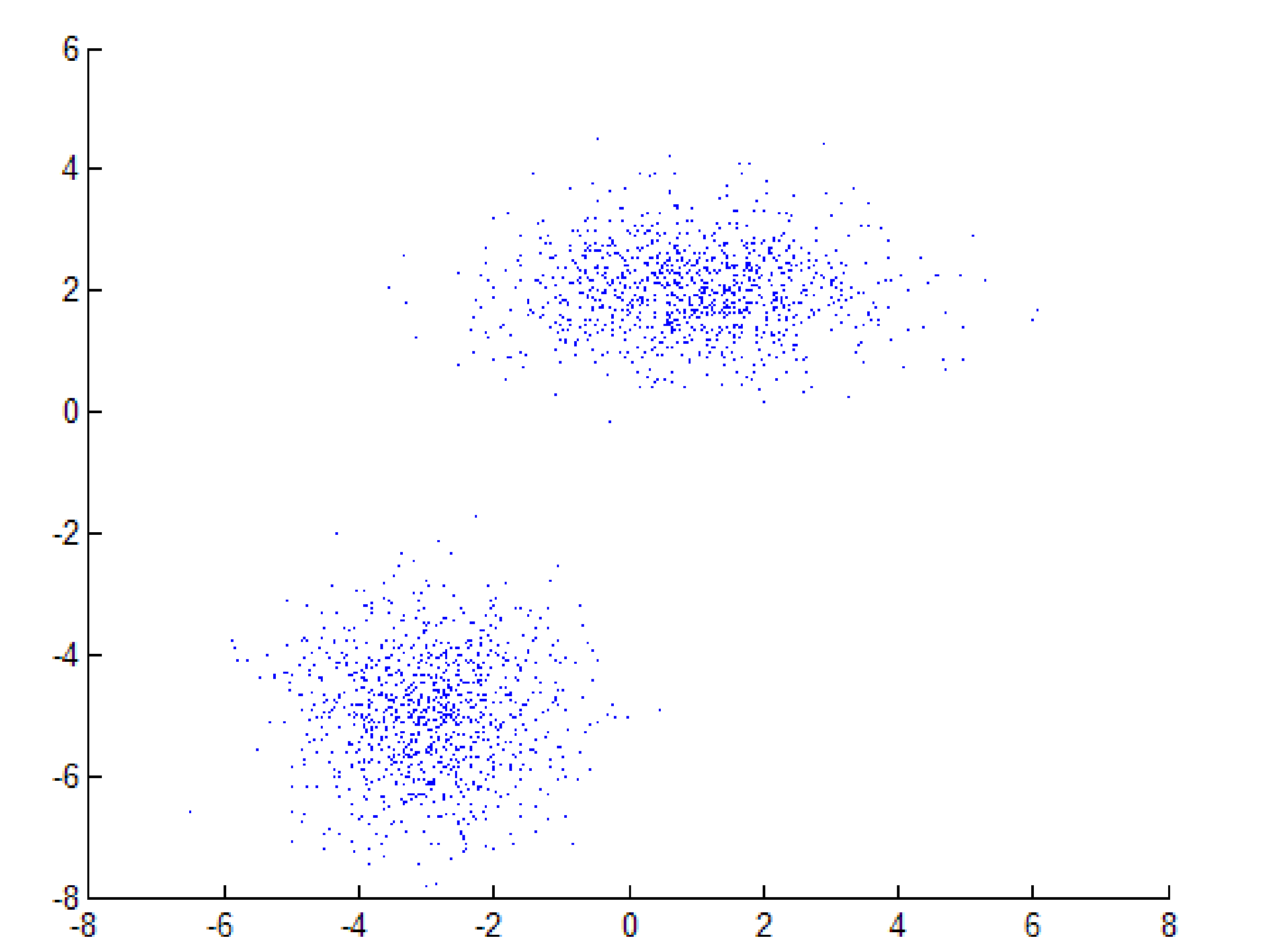
Využití rozdělení pravděpodobnosti

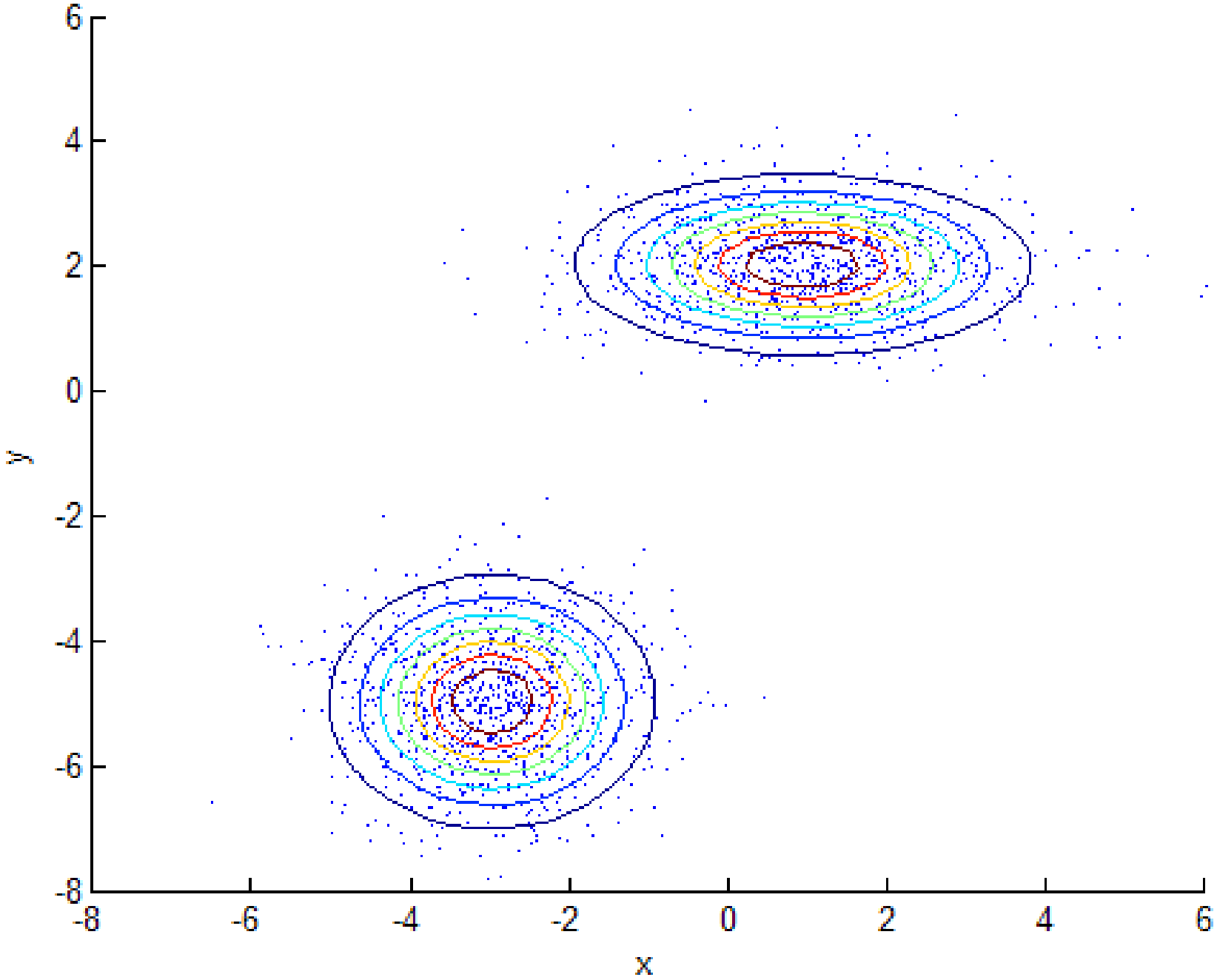
- *Diskriminační analýza* (DA): $P(C)$ – z rozdělení odpovídajícího C


20
18
16
14
12
10
8
6
4
2
0
110
120
130
140
150
160
170
180
190
200
210

Využití rozdělení pravděpodobnosti

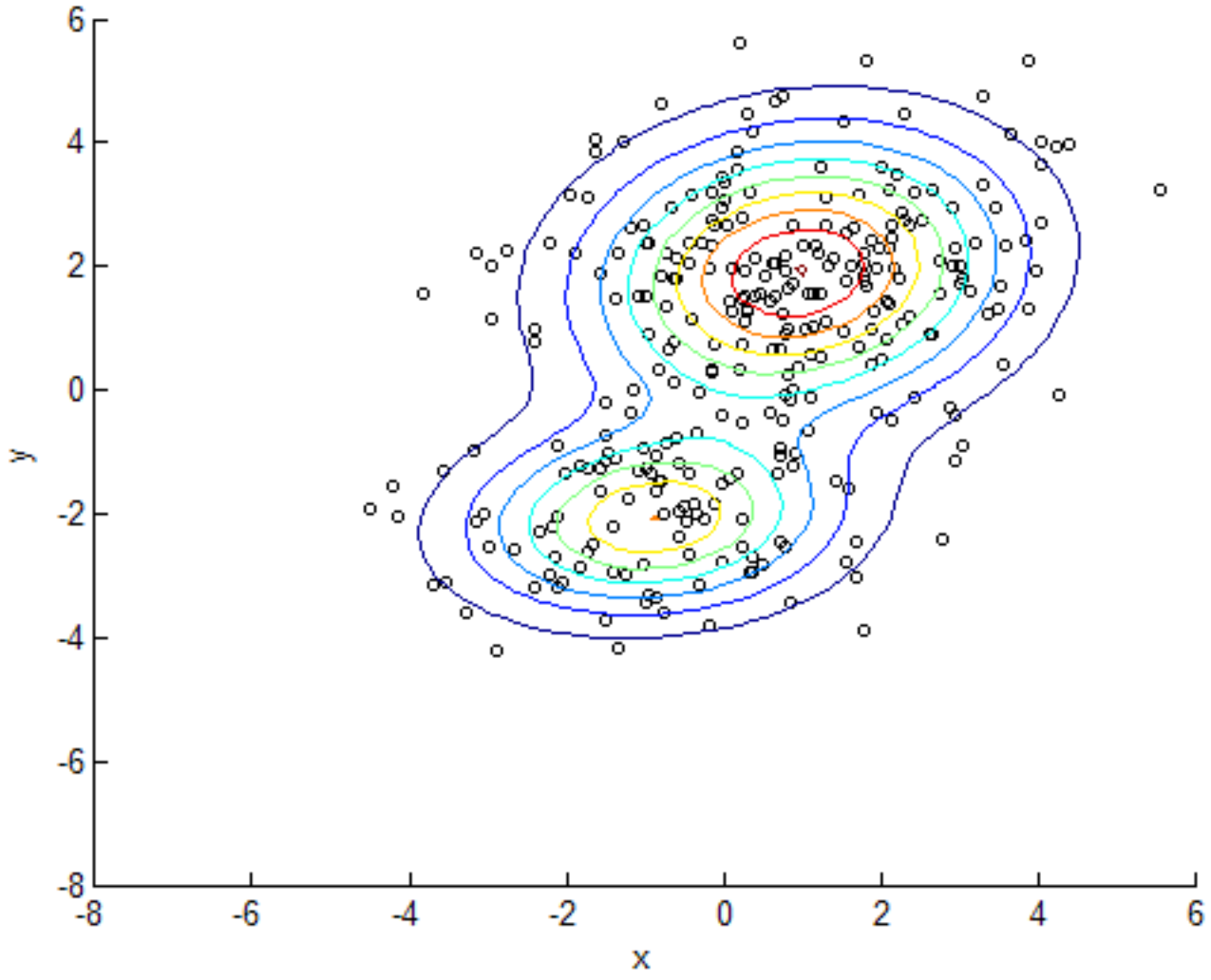
- *Diskriminační analýza* (DA): $P(C)$ – z rozdělení odpovídajícího C
  - určeno hustotou $f_C \in$ *rodina* $\mathcal{F} = \{f(\cdot|\theta) : \theta \epsilon \Theta\}$, Θ – možné parametry
  - originální (Fisherova) DA: $\mathcal{F}$ = vícerozměrná normální rozdělení
- *Učení* klasifikátoru: výpočet odhadů $\hat{\theta}$ parametrů pro $f_C, C \epsilon \mathcal{C}$
  - $\hat{\theta}$ – maximálně věrohodný odhad: $\hat{\theta} = \arg\max_{\theta \epsilon \Theta} \prod_{\text{učicí data}}^{x \epsilon} f(x|\theta)$
- Predikce pro nové x: $\arg\max_{C \epsilon \mathcal{C}} f_C(x|\hat{\theta})$


6
4
2
0
-2
-4
-6
-8
-8
-6
-4
-2
0
2
4
6
8


6
4
2
0
-2
-4
-6
-8
y
-8
-6
-4
-2
0
2
4
6
x

Lineární a kvadratická DA

- *Lineární (LDA):* $\mathcal{F} = \left\{ f(x|\beta) = \frac{1}{\sqrt{2\pi|\Sigma|}} e^{-\frac{1}{2}(x-\beta)^\top \Sigma^{-1}(x-\beta)} : \beta \in \mathbb{R}^d \right\}$ dimenze x
  - $\Sigma \in \mathbb{R}^{d,d}$: předem daná, stejná pro všechny třídy
  - pro $C \neq C'$ je množina $\{x \in \mathbb{R}^d : f_C(x) = f_{C'}(x)\}$ nadrovina (= lineární plocha)
- *Kvadratická (QDA):* $\mathcal{F} = \left\{ f(x|\beta) = \frac{1}{\sqrt{2\pi|\Sigma|}} e^{-\frac{1}{2}(x-\beta)^\top \Sigma^{-1}(x-\beta)} : \beta \in \mathbb{R}^d, \Sigma \in \mathbb{R}^{d,d} \right\}$
  - Σ – další parametr ⇒ parametry celkem $\frac{d(d+1)}{2}$ dimenzí (LDA: d)
  - pro $C \neq C'$ je množina $\{x \in \mathbb{R}^d : f_C(x) = f_{C'}(x)\}$ kvadratická plocha


6
4
2
0
-2
-4
-6
-8
y
-8
-6
-4
-2
0
2
4
6
x

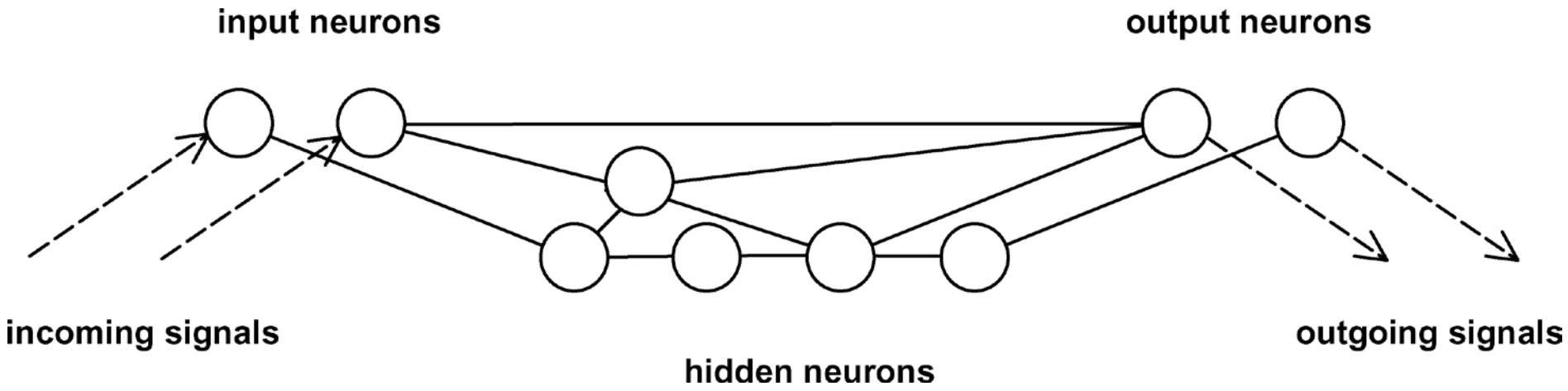
DA při klasifikaci obrázků a videí

- Čemu odpovídají jednotlivá pravděpodobnostní rozdělení?
  1. *konceptům* popisujícím zájmy uživatele
  2. *dodaným příkladům* zájmů uživatele
- *Atributy: statické* (obličeje,…), *dynamické* (pohyby, události)
- Využití: 1) doporučovací systémy − obsahové filtrování

  2) automatická detekce pornostránek

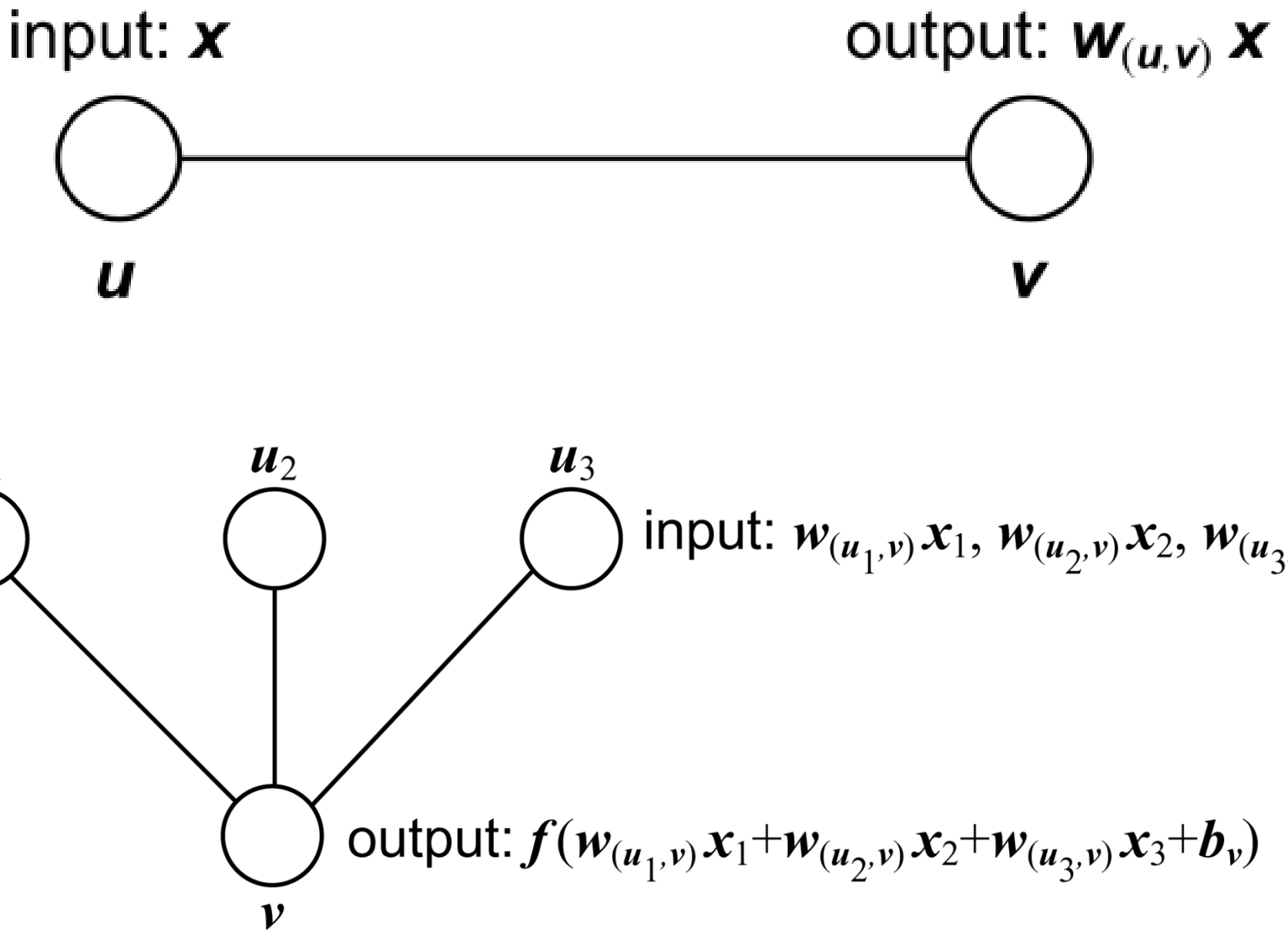
Umělé neuronové sítě

- Systémy implementující část funkcionality neuronových sítí
- *Neurony* − vstupní, výstupní, skryté


input neurons
output neurons
incoming signals
outgoing signals
hidden neurons

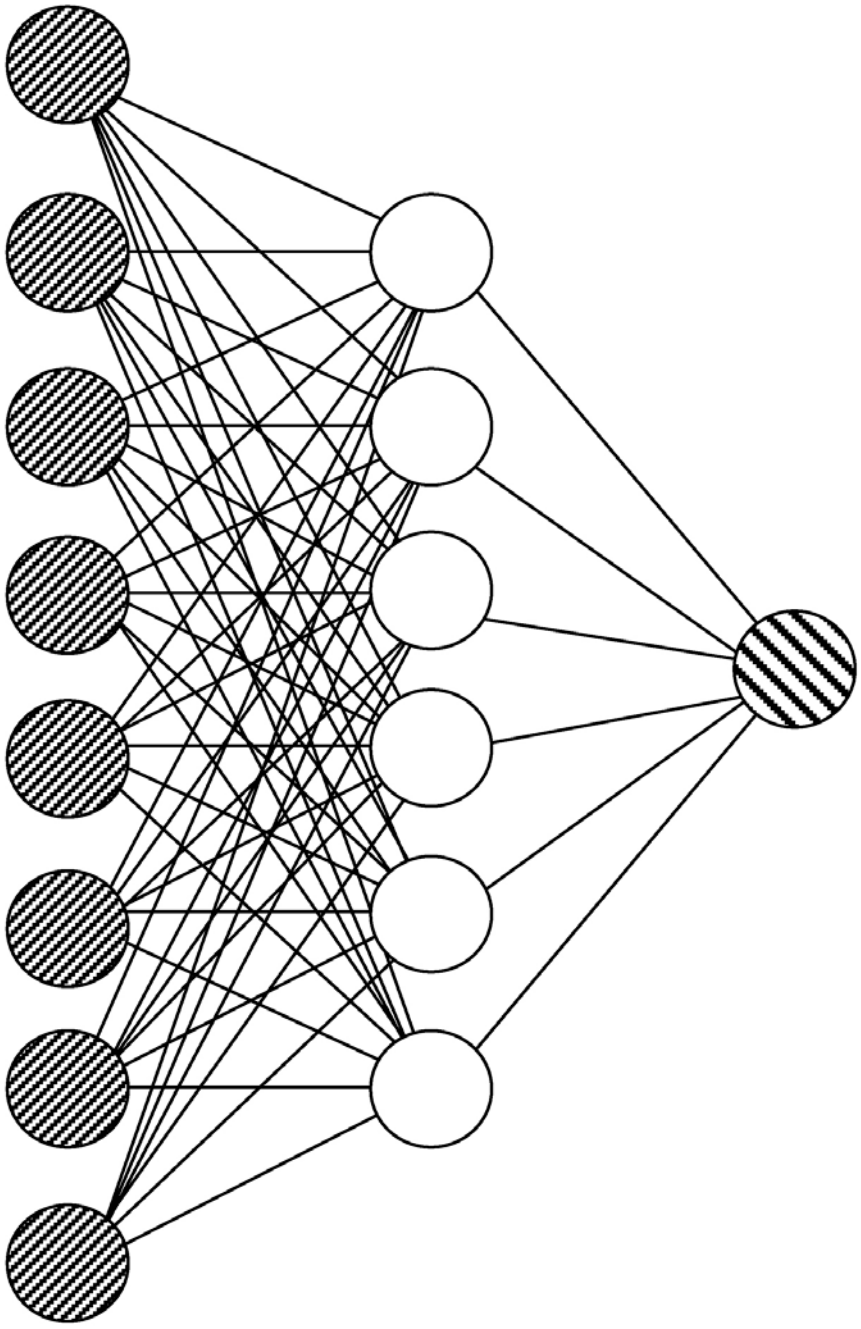
Umělé neuronové sítě

- Systémy implementující část funkcionality neuronových sítí
- *Neurony* − vstupní, výstupní, skryté
  - transformují signály (většinou nelineárně) − *somatické* operace
  - šíření signálů: vstupy → skryté neurony → výstupy
- *Spoje* − *synaptické* operace: většinou násobení vahou


input: x
output: $w_{(u,v)}\,x$
u
v
u_1
u_2
u_3
input: $w_{(u_1,v)}x_1,\ w_{(u_2,v)}x_2,\ w_{(u_3,v)}x_3$
output: $f(w_{(u_1,v)}x_1+w_{(u_2,v)}x_2+w_{(u_3,v)}x_3+b_v)$
v

Umělé neuronové sítě

- Systémy implementující část funkcionality neuronových sítí
- *Neurony* − vstupní, výstupní, skryté
  - transformují signály (většinou nelineárně) − *somatické* operace
  - šíření signálů: vstupy → skryté neurony → výstupy
- *Spoje* − *synaptické* operace: většinou násobení vahou
- *Topologie* − propojení neuronů spoji: nejčastěji *vrstevnatá*


input layer
hidden layer
output layer

Perceptrony a jejich zřetězení

- *Perceptron* − bez skrytých neuronů, somatická operace:

  skoková aktivační funkce Θ: $\Theta(\mathbb{R}_-) = 0, \Theta(\mathbb{R}_{+0}) = 1 \Rightarrow f_C(x) = \Theta(w^\intercal x + b)$

  - třída $\subset$ průnik poloprostorů $\Rightarrow$ vyžaduje lineární separabilitu

- *Vícevrstvý perceptron (MLP)* − zřetězení $\Rightarrow$ skryté vrstvy

  - somatická operace: sigmoidní aktivační funkce ($\sigma(x) = \frac{1}{1+e^{-x}}$, $= \frac{e^x - e^{-x}}{e^x + e^{-x}}$, ...)

  - nelineární klasifikátory, při 1 skryté $f_C = \sigma(\sum_{h \in H} w_{hC} \sigma(w_{\cdot h}{}^\intercal x + b_h) + b_C)$

MLP při filtraci spamu

- Výhoda: vysoká *přesnost*, zejména týmového použití
- Nevýhoda: *pomalé učení* (+ doučování nových spamů)
- ⇒ *Strategie* pro nasazení ve spamových filtrech:
  - na straně *klienta* (⇐ doučování méně časté)
  - doplňovány klasifikátory s rychlým doučováním (k-NN, …)
- Další použití: filtrace spamu přenášeného *obrázky*

MLP v doporučovacích systémech

- Při kolaborativním filtrování častěji než při obsahovém
- Výhoda: *přesnost*, zase hlavně v týmu
- Nevýhoda: *obtížná vysvětlitelnost* (> než pomalé učení) výsledku: proč je konkrétní objekt doporučován
- Alternativní použití: *predikce nesdělených* sociálních *atributů* (věk, příjmová skupina, …) indikujících, co doporučit

MLP při odhalování malware

- Odhalují podle statických vlastností, tj. kódu
  - vstupy: n-gramy bytů | operací, metadata binárek
  - kombinování s klasifikátory odhalujícími podle chování
- Problém: špatné učení třídy zastoupené vzácně
  - protiopatření: obohacení učicích dat zástupci malware
- Příklad komerčního použití MLP: antivirák IBM