Učení pomocí regularizačních sítí

Learning with Regularization Networks

Petra Kudová

Disertační práce

Osnova

Úvod

- neuronové sítě a učení z dat
- cíle disertační práce

Regularizační sítě

- teorie regularizace
- učení regularizační sítě, volba metaparametrů
- složené jádrové funkce

RBF sítě

- gradientní, třífázové a genetické učení
- hybridní metody

Srovnání regularizačních a RBF sítí

Závěr

- dosažené výsledky
- budoucí práce

Úvod

Učení z dat – učení s učitelem

- dána množina příkladů (vstup, výstup)
- najít odpovídající zobrazení (funkci)
- důraz na generalizaci

Cíle disertační práce

1. Prozkoumat učení pomocí regularizačních sítí
   - role jádrové funkce a regularizačního parametru
   - srovnání různých jádrových funkcí
2. Navrhnout autonomní učící algoritmus regularizační sítě
   - metaparametry základního algoritmu: regularizační parametr a jádrová funkce
   - zahrnout optimalizaci metaparametrů do učení

⋮

Cíle disertační práce

3. Prozkoumat a navrhnout učící algoritmy pro generalizované regularizační sítě
   - prozkoumat možnosti učení RBF sítí
   - hybridní metody

4. Experimentální studie zkoumaných algoritmů
   - vliv metaparametrů regularizační sítě na učení
   - otestovat a porovnat různé metody učení RBF sítí
   - srovnání regularizačních sítí a RBF sítí

Metodologie experimentů

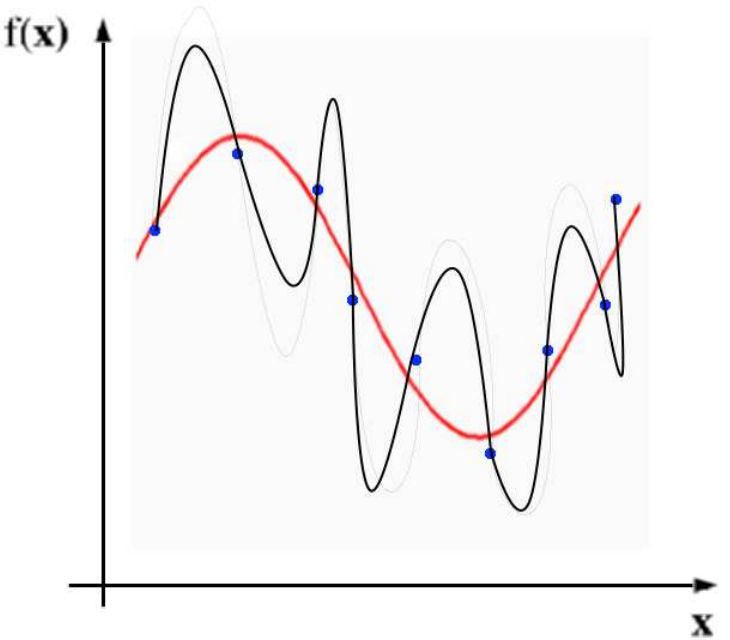
- Proben1 - sada úloh pro testování neuronových sítí
- trénovací a testovací množina
- implementace v systému Bang, LAPACK

Regularizační sítě

Učení z příkladů

Definice problému

- **Dáno:** množina vzorů $\{(\vec{x}_i, y_i) \in \mathbb{R}^d \times \mathbb{R}\}_{i=1}^N$
  množina vznikla navzorkováním nějaké neznámé funkce
- **Cíl:** rekonstruovat tuto funkci



Minimalizace empirické chyby

- najít f, která minimalizuje

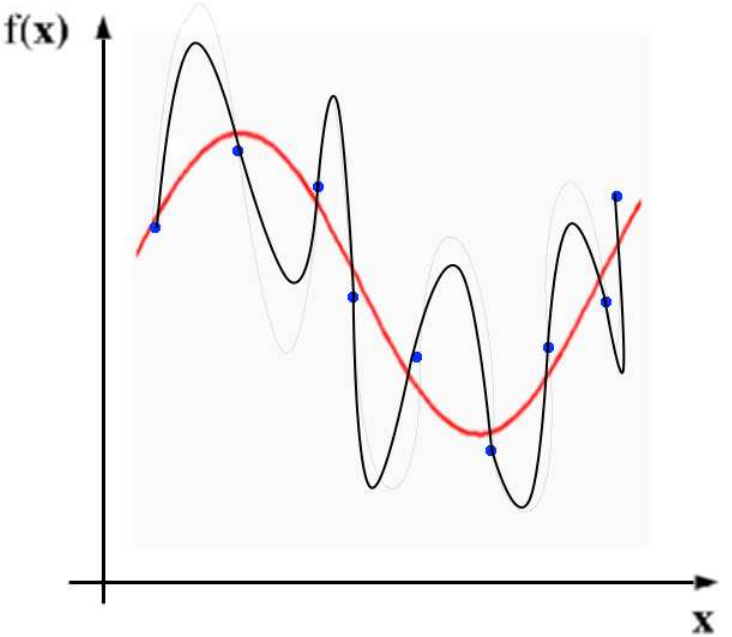
$$H[f] = \sum_{i=1}^{N} (f(\vec{x}_i) - y_i)^2$$

- obecně není dobře určená úloha
- vybrat jedno řešení na základě znalosti problému

Učení z příkladů

Definice problému

- **Dáno:** množina vzorů $\{(\vec{x}_i, y_i) \in \mathbb{R}^d \times \mathbb{R}\}_{i=1}^N$
  množina vznikla navzorkováním nějaké neznámé funkce
- **Cíl:** rekonstruovat tuto funkci



Regularizace

- přidat regularizační člen (stabilizátor)

$$H[f] = \sum_{i=1}^{N} (f(\vec{x}_i) - y_i)^2 + \gamma \Phi[f]$$

Stabilizátory založené na Fourierově transformaci

[Girosi, Jones, Poggio, 1995]

- stabilizátor ve tvaru:

$$\Phi[f] = \int_{\mathbb{R}^d} d\vec{s} \frac{|\tilde{f}(\vec{s})|^2}{\tilde{G}(\vec{s})}$$

$\tilde{G}$ pozitivní funkce
$\tilde{G}(\vec{s}) \to 0$ for $||s|| \to \infty$
$1/\tilde{G}$ *high pass* filtr

- pro širokou třídu stabilizátorů (G pozitivně definitní) má řešení tvar

$$f(x) = \sum_{i=1}^{N} w_i G(\vec{x} - \vec{x}_i)$$

váhy w_i lze nalézt řešením lineárního systému

$$(\gamma I + G)\vec{w} = \vec{y}$$

Hilbertovy prostory s reprodukčními jádry (RKHS)

[Poggio, Smale, 2003]

- zvol symetrickou, pozitivně definitní funkci $K = K(\vec{x}_1, \vec{x}_2)$
- vezmi $\mathcal{H}_K$ RKHS definovaný K

$$\mathcal{H}_K = \mathtt{compl}\{\sum_{i=1}^{n} a_i K_{x_i};\ x_i \in \Omega, a_i \in \mathbb{R}\}$$

- stabilizátor: norma $||\cdot||_K$ v $\mathcal{H}_K$

$$H[f] = \frac{1}{N}\sum_{i=1}^{N}(y_i - f(\vec{x}_i))^2 + \gamma||f||_K^2$$

- minimalizuj $H[f]$ na $\mathcal{H}_K \quad \longrightarrow \quad$ řešení:

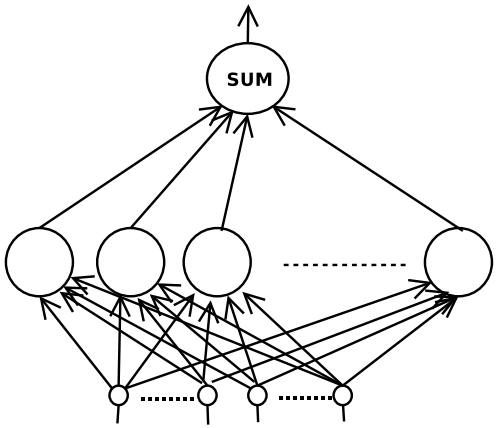
$$f(\vec{x}) = \sum_{i=1}^{N} w_i K_{\vec{x}_i}(\vec{x}) \qquad (N\gamma I + K)\vec{w} = \vec{y}$$

Regularizační síť

Architektura

- funkce sítě: $f(\vec{x}) = \sum_{i=1}^{N} w_i K(\vec{c_i}, \vec{x})$

$\vec{x}$	vstup
$\vec{c_i}$	středy
$K(\cdot,\cdot)$	jádrová funkce
f	výstup



Základní učící algoritmus [Algoritmus 4.1.1]

1. umísti středy jádrových funkcí do datových bodů
2. urči váhy řešením lineáního systému

$$(\gamma I + K)\vec{w} = \vec{y}$$

- metaparametry γ a K - definice problému

Učení regularizační sítě

- přímočarý algoritmus
- klíčová část - řešení lineárního systému

$$(\gamma I + K)\vec{w} = \vec{y}$$

- $\gamma I + K$ je striktně pozitivní $\rightarrow$ systém je **dobře určený**
- Je také dobře podmíněný ?

- Pro velká $\gamma \rightarrow$ dominantní diagonála $\rightarrow$ dobré

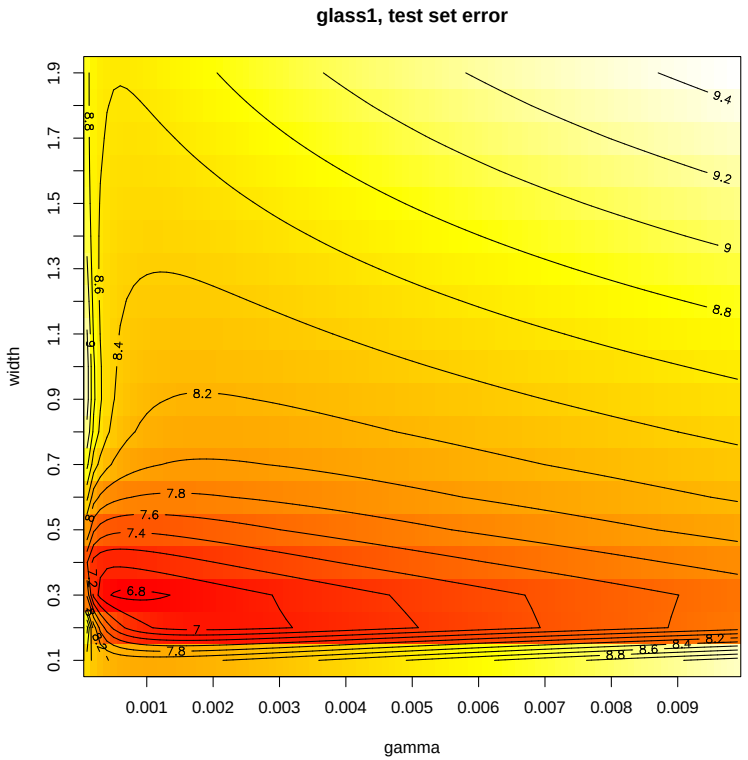

γ nelze volit libovolně

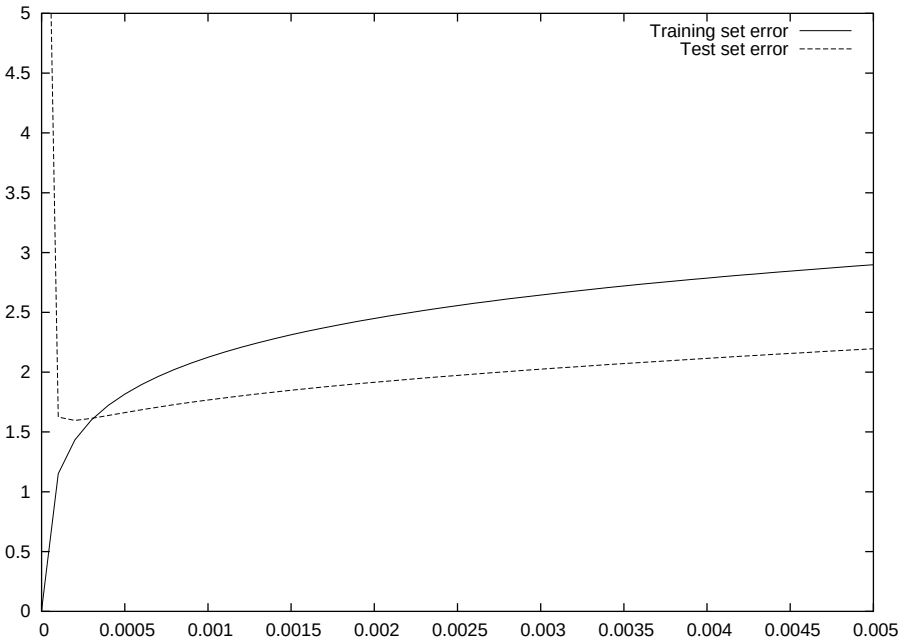

lze volit jádrovou funkci a její parametry

- volba γ a jádrové funkce je klíčová pro úspěšnost učení

Volba γ a jádrové funkce

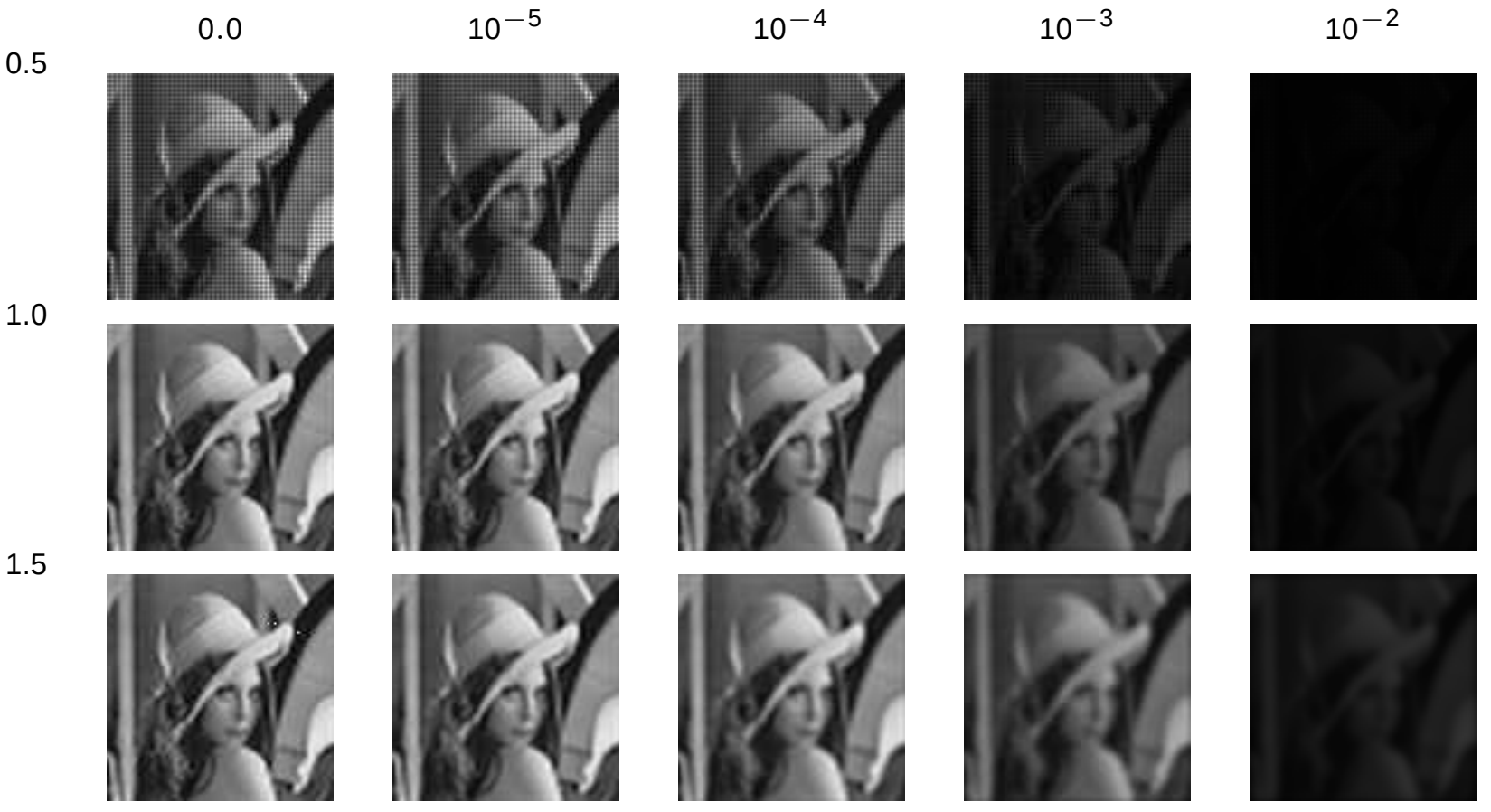
- Gaussova jádrová funkce
- závislost trénovací a testovací chyby na volbě γ a šířky Gassovy funkce





Volba γ and jádrové funkce

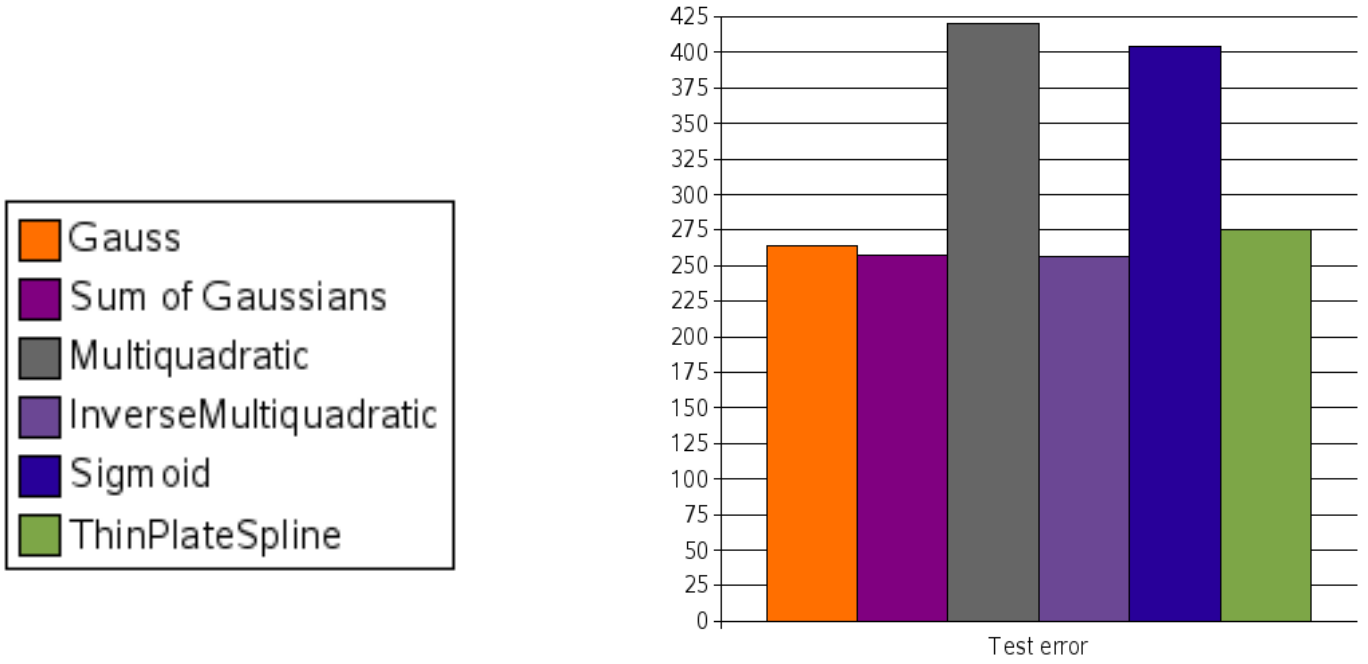
- aproximace obrázku pomocí reg. sítě s Gaussovými jádry
- trénovací data - 50x50, vygenerované obrázky - 100x100



Srovnání jádrových funkcí

[Kudová, ITAT, 2006]

- chyba na testovací množině na datech z Proben1



Testovací chyba

- lokální jádra - Gaussova a inverzní multikvadratická funkce

Autonomní učící algoritmus

Učení regularizační sítě

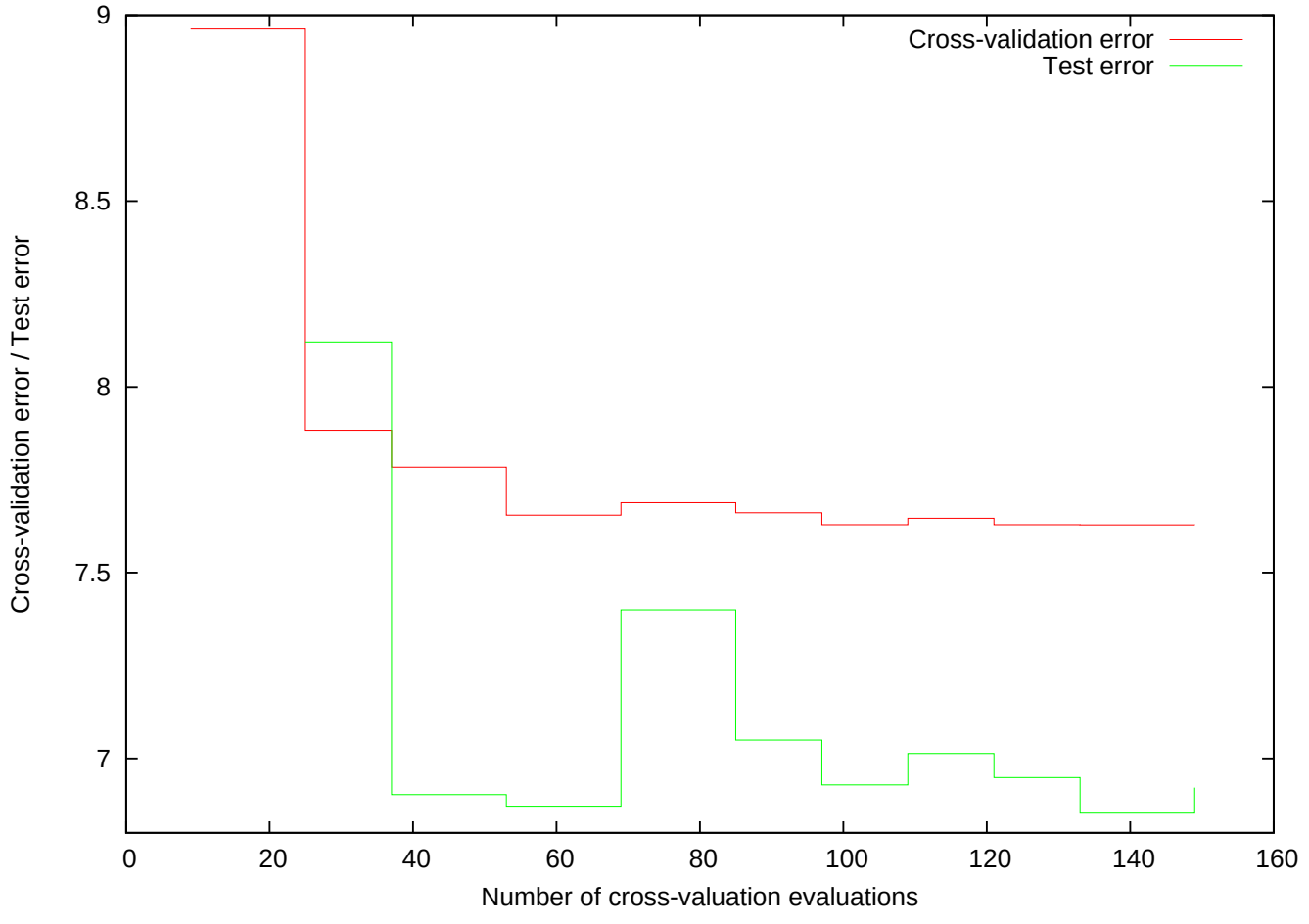
1. optimalizace metaparametrů
    1. typ jádrové funkce
    2. parametry jádrové funkce
    3. regularizační parametr
2. učení regularizační sítě s nalezenými metaparametry

Hledání vhodného γ a jádrová funkce

- γ a jádrová funkce reprezentují znalost problému
- minimalizace krosvalidační chyby
    - adaptivní mřížka [Algoritmus 4.5.1]
    - prohledávání pomocí GA [Algoritmus 4.6.1]

Hledání vhodného γ a šířky Gaussovy funkce

- průběh krosvalidační a odpovídající testovací chyby
- stačí desítky evaluací



Složené jádrové funkce

Motivace

- jádrová funkce reprezentuje znalost problému, předpoklad
- volba jádrové funkce závisí na daném problému
- v praxi jsou data často nehomogenní (odlišné typy atributů, různá hustota v různých oblastech vstupního prostoru, ...)

Součet a součin RKHS

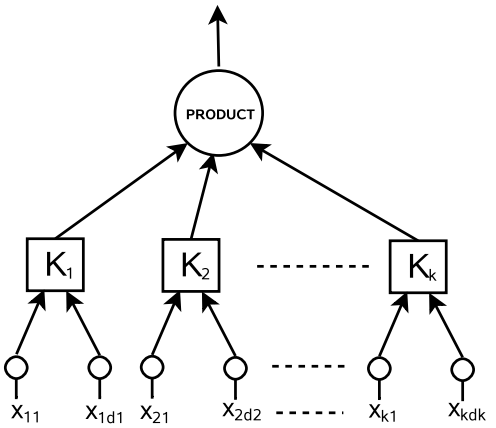
- založeno na Aronszajnově teorii
- součet a součin RKHS je opět RKHS
- jádrová funkce takového RKHS je součtem resp. součinen původních jádrových funkcí [Věta 3.2.5 a 3.4.6]

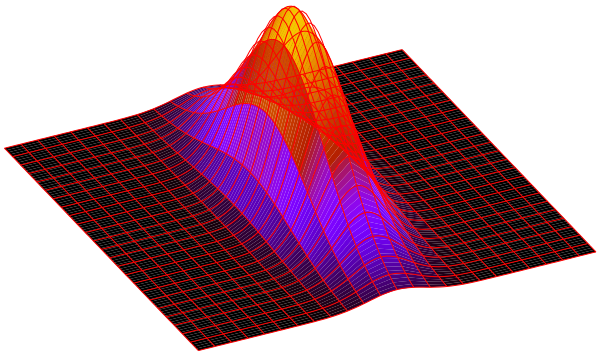
Součin jádrových funkcí

Motivace

- atributy různého typu
- pro různé atributy (skupiny atributů) různé jádrové funkce
- skupiny atributů zpracovávány nezávisle, mohou být různých typů

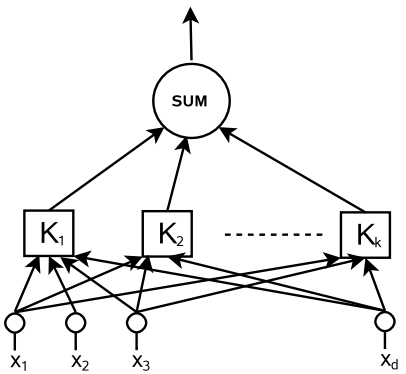
Součinová jednotka [Definice 3.3.3]

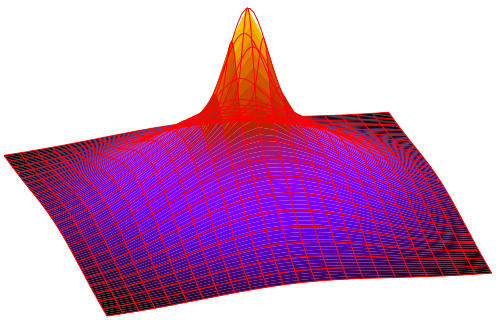




Součet jádrových funkcí

Součtová jednotka [Definice 3.5.3]





Omezená jádrová funkce [Definice 3.5.5]

- omezení na část vstupního prostoru

$$K_A(\vec{x},\vec{y}) = \begin{cases} K(\vec{x},\vec{y}) & \vec{x},\vec{y} \in A, \\ 0 & \text{jinak} \end{cases}$$

- data se liší v různých oblastech vstupního prostoru

Rozděl a panuj

Součet omezených jádrových funkcí

- pokud jsou příslušné oblasti disjunktní, výsledná síť se dá vyjádřit jako součet podsítí

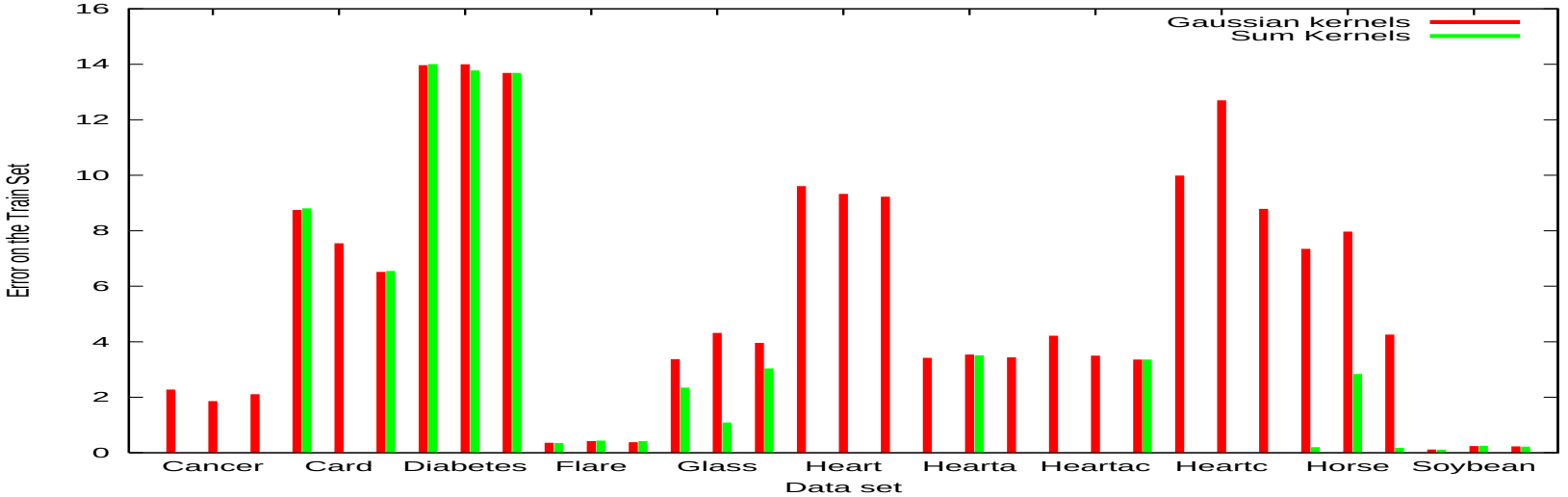
$$f(\vec{x}) = \sum_{\vec{x}_i \in A_1} w_i K_1(\vec{x}, \vec{x}_i) + \ldots + \sum_{\vec{x}_i \in A_k} w_i K_k(\vec{x}, \vec{x}_i)$$

Rozděl a panuj [Kapitola 3.5.2]

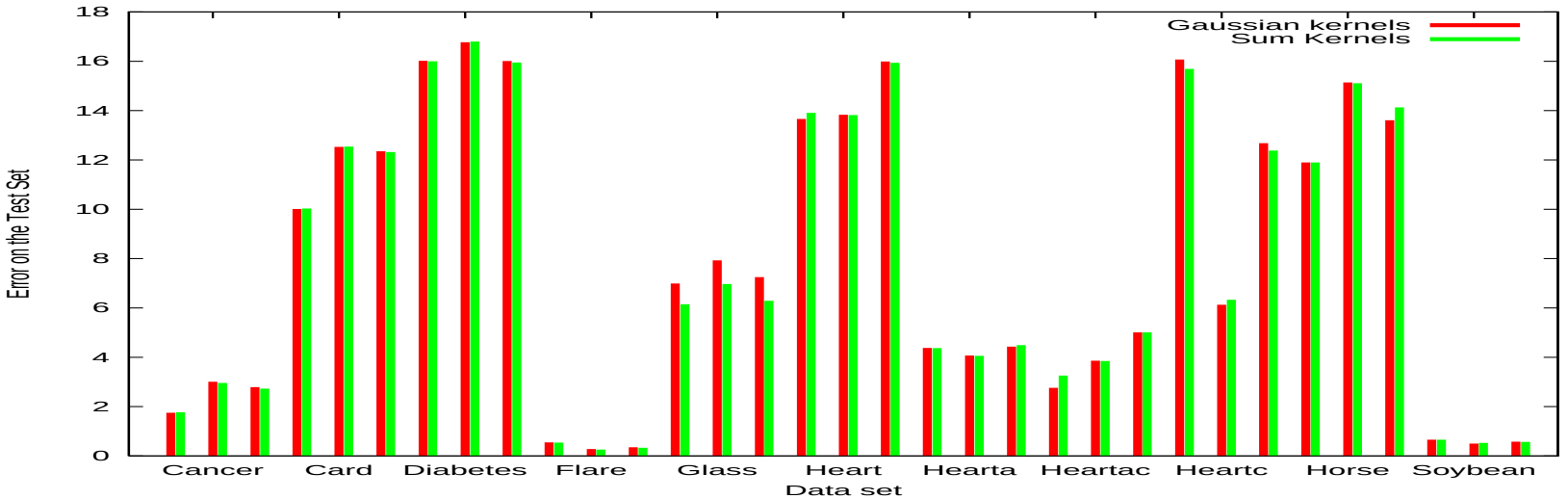
- data lze rozdělit na podmnožiny
- sítě se učí nezávisle, možno i paralelně
- snížení časové i prostorové složitosti učení

Součtová jádrová funkce

Chyba na trénovací množině

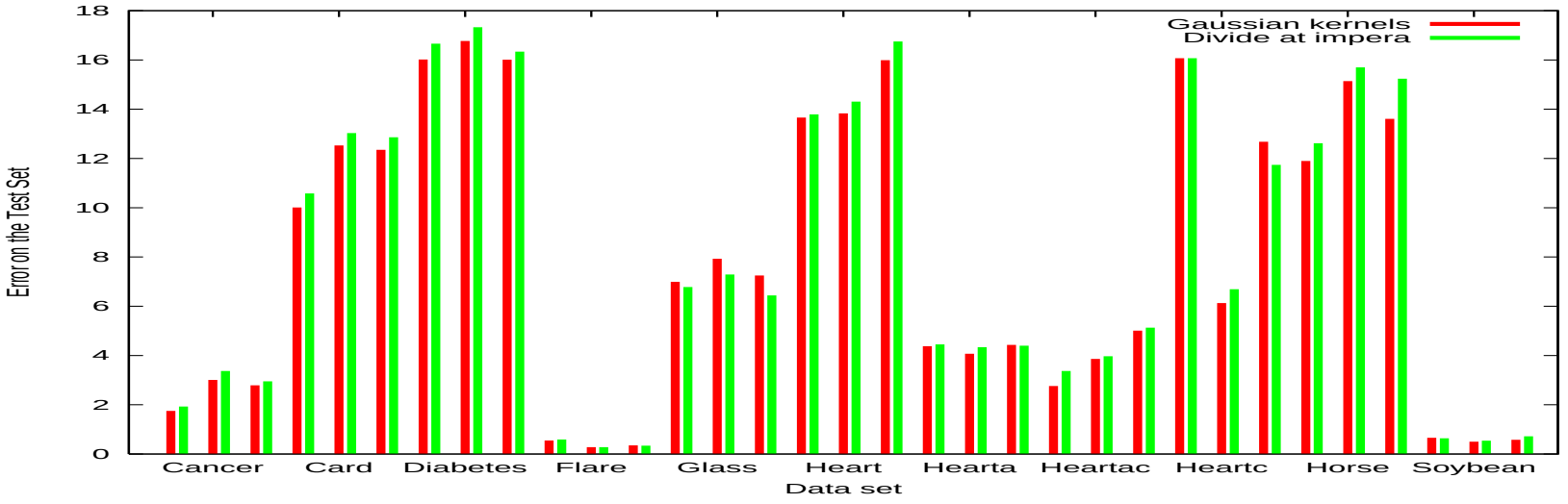


Chyba na testovací množině

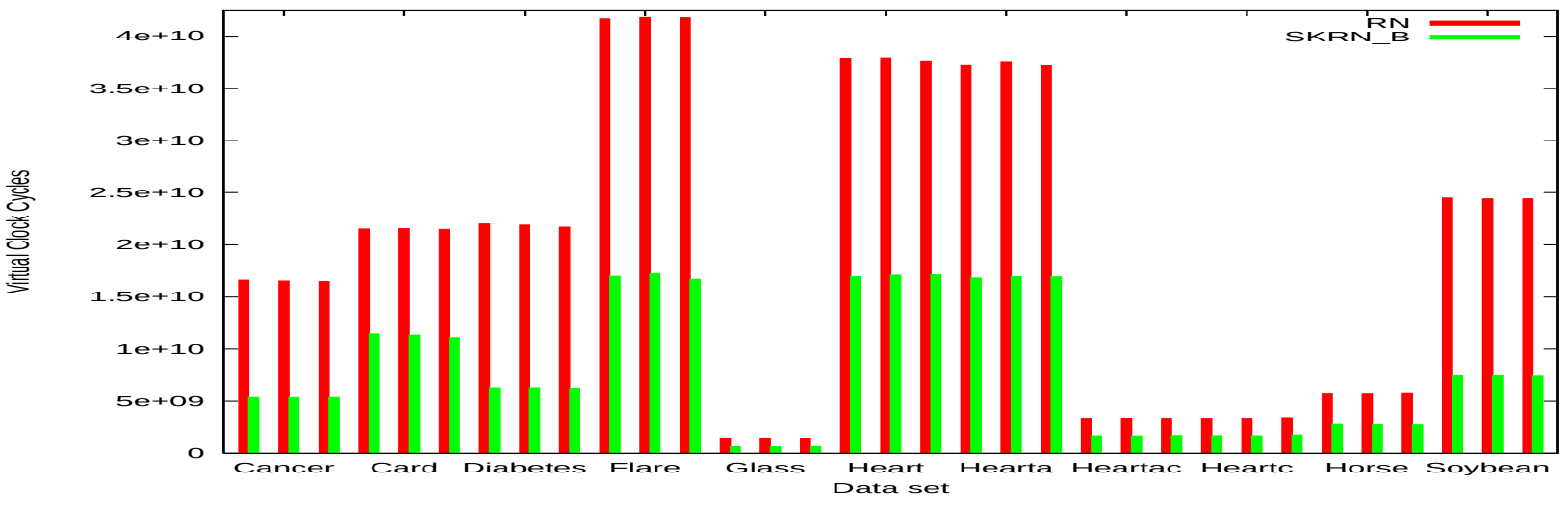


Rozděl a panuj

Chyba na testovací množině



Časové srovnání



RBF SÍTĚ

Generalizované regularizační sítě

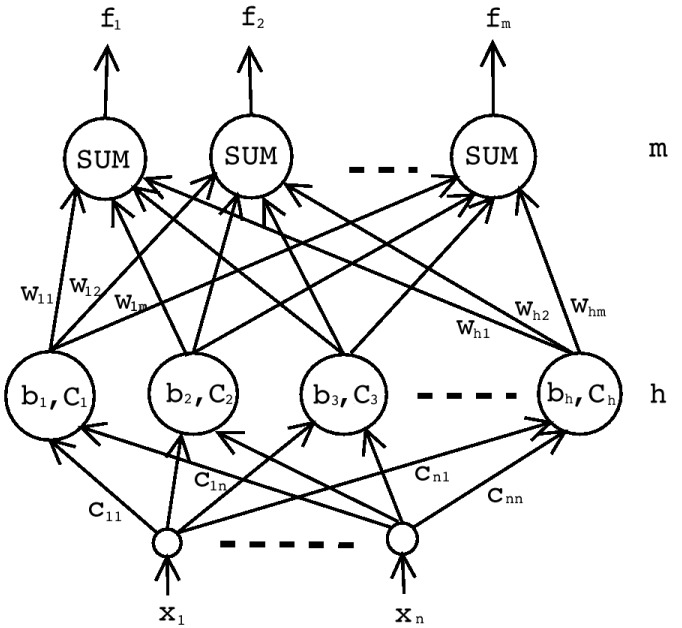
Generalizované regularizační sítě

- méně jednotek než datových vzorků
- parametry skrytých jednotek

RBF sítě

- skrytá vrstva - RBF jednotky
- lineární výstupní vrstva
- funkce sítě

$$f_s(\vec{x}) = \sum_{j=1}^{h} w_{js} \varphi \left(\frac{\|\vec{x} - \vec{c}_j\|_C}{b_j} \right)$$

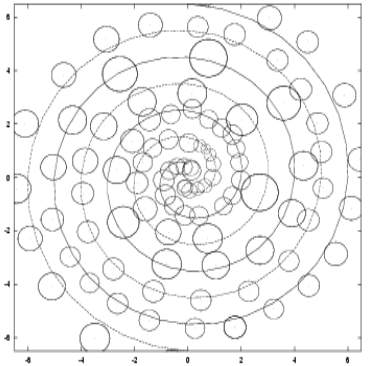


- $\| \vec{x} \|_C = (C\vec{x})^T (C\vec{x}) = \vec{x}^T C^T C \vec{x}$

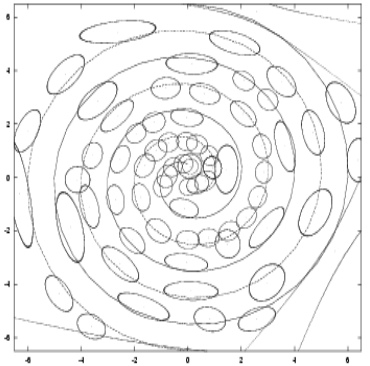
Použití vážené normy

- klasifikace bodů v rovině ležících na dvou spirálách
- RBF jednotky s a bez vážené normy

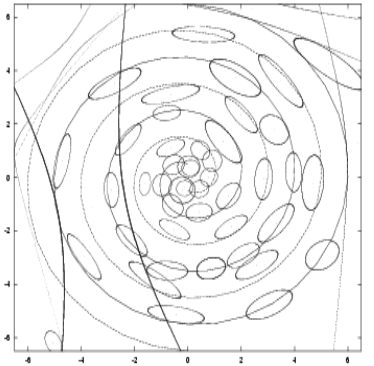
100 radiální



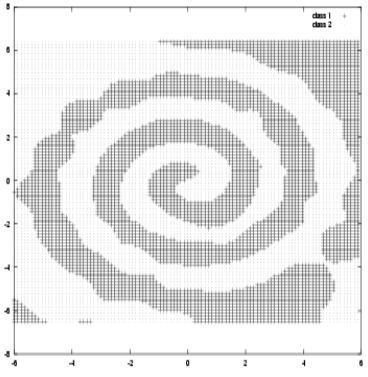
70 oválné

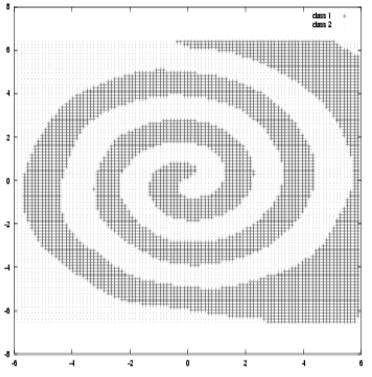


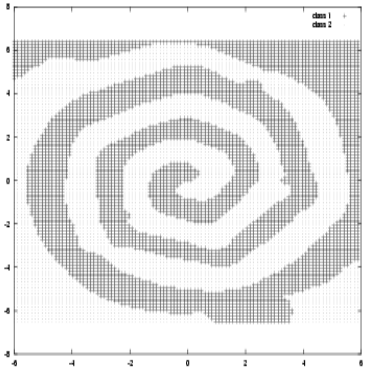
50 oválné



Výsledná klasifikace







Učení RBF sítí

Metody učení

- gradientní učení [Algoritmus 5.3.1]
  - inspirováno algoritmem zpětného šíření
  - jedna skrytá vrstva - jednodušší výpočet derivací
- třífázové učení [Algoritmus 5.4.1]
  - tři skupiny parametrů,
  - heuristiky, lineární optimalizace
- genetické učení [Algoritmus 5.5.3]
  - optimalizace použitím genetického učení

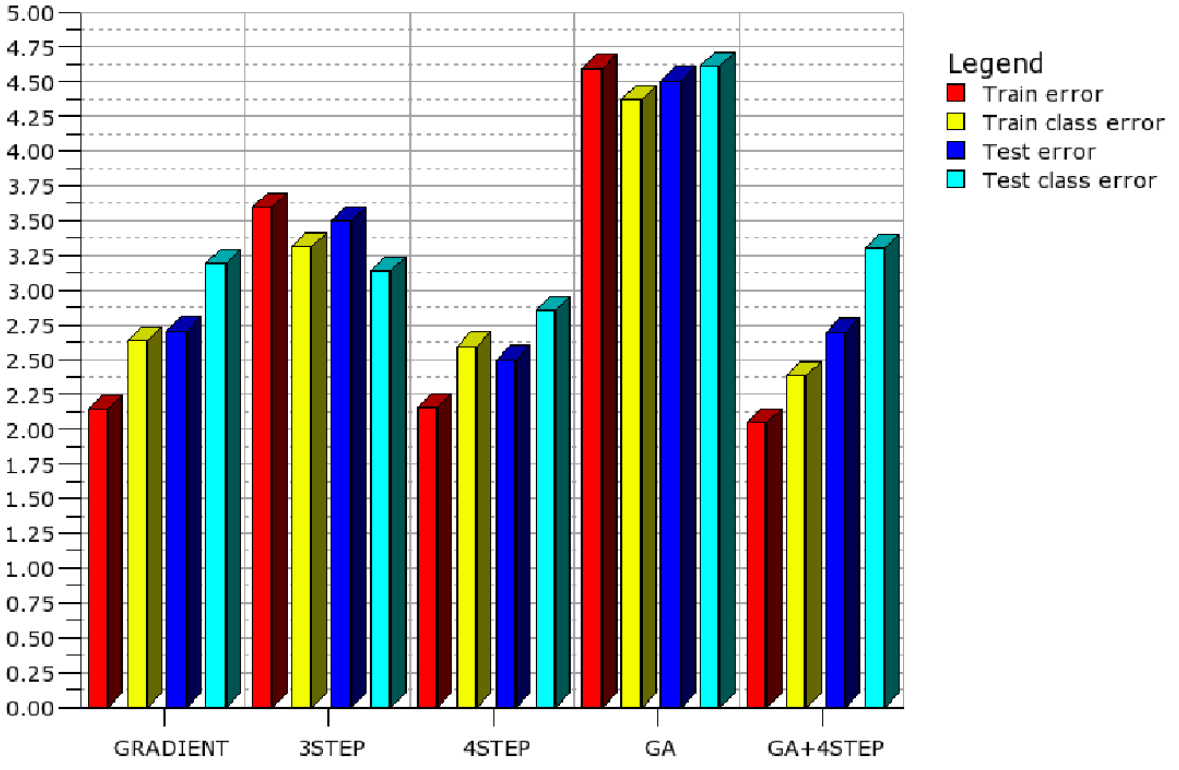
Hybridní učení

- hybridní genetické učení [Algoritmus 5.6.1]
  - kombinace genetického a třífázového učení
- čtyřfázové učení [Algoritmus 5.6.2]
  - třífázové učení následované gradientní fází

Srovnání učících algoritmů pro RBF sítě

[Neruda, Kudová, Future Generation Computer Systems, 2005]

- srovnání chyby na trénovací a testovací množině
- uvedena i chyba klasifikace



Srovnání regularizačních sítí a RBF sítí

Regularizační sítě vs. RBF sítě

Regularizační sítě

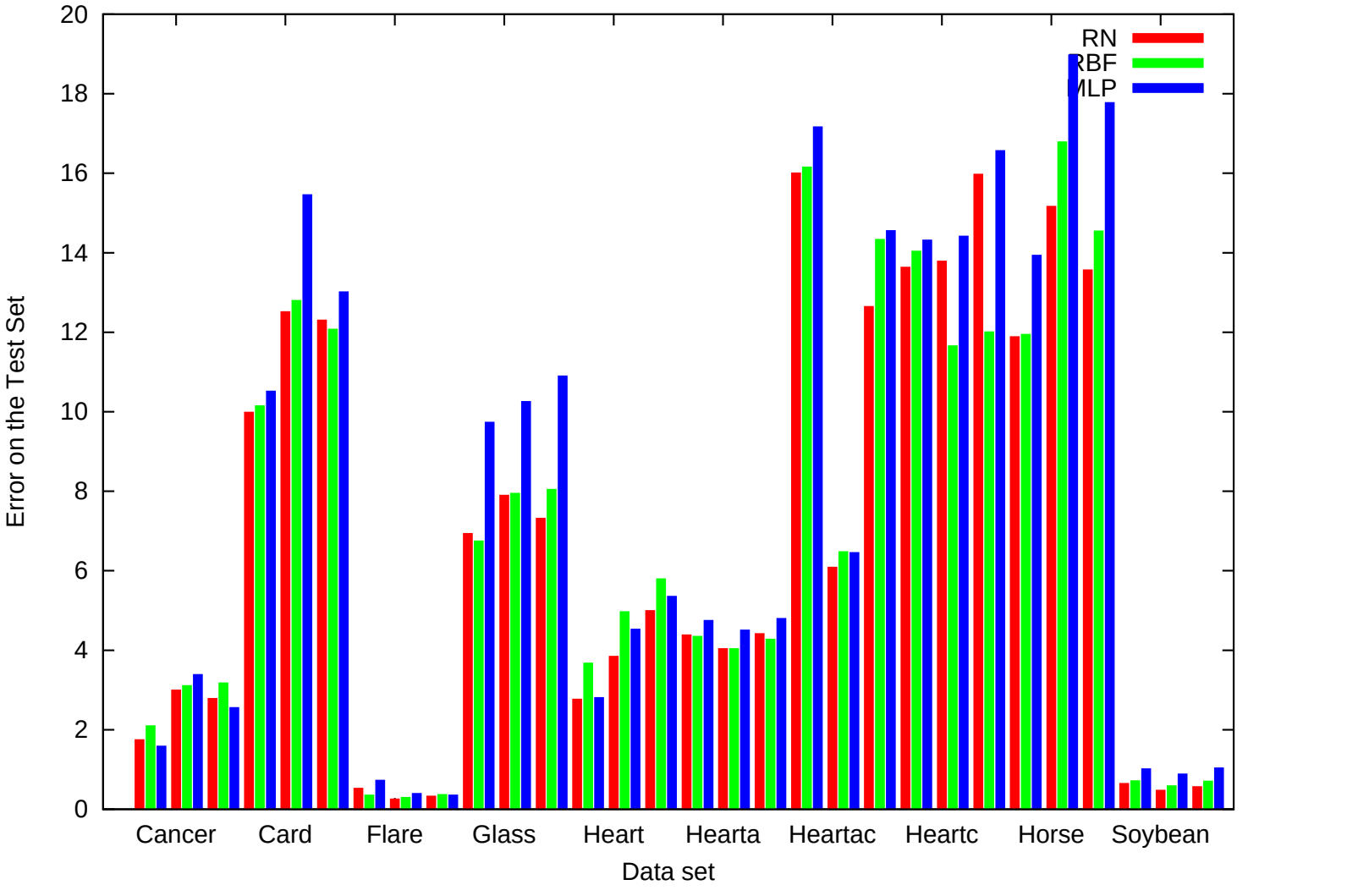
- dobré teoretické základy, snaha najít optimální řešení
- prostorová složitost sítě závisí na velikosti trénovací množiny
- učení - lineární optimalizace + krosvalidace
- metaparametry (γ, jádrová funkce)

RBF sítě

- hledáme přibližné řešení
- prostorová složitost závisí pouze na dimenzi vstupů, RBF jednotka má $(n + 1 + n(n + 1)/2)$ parametrů
- různé optimalizační algoritmy + heuristiky
- metaparametr h

Regularizační sítě vs. RBF sítě

[Kudová, Neruda, LNCS 3635, 2005]



Závěr

Výsledky

- sjednocující teoretický popis regularizačních a RBF sítí
- návrh učících algoritmů a modelů
  - autonomní učící algoritmus pro regularizační sítě
  - nové modely regularizačních sítí: součinové a součtové jádrové funkce
  - hybridní přístupy k učení RBF sítí
- rozsáhlá experimentální studie zkoumaných algoritmů
  - srovnání stávajících algoritmů pro RBF sítě
  - srovnání regularizačních a RBF sítí
  - aplikace na predikci průtoku na řece Ploučnici
- implementace všech algoritmů v prostředí Bang

Výhledy do budoucna

- redukce počtu jednotek
  - prořezávání sítě
  - volba počtu jednotek na základě teoretických výsledků
  - experimentální ověření
- složené jádrové funkce
  - jádrové funkce pro různé datové typy
  - kombinace jádrových funkcí pro nehomogenní data
- optimalizace pro velká data
  - redukce časové a prostorové složitosti
  - možnosti paralelizace

Diskuse

Závislost velikosti sítě a chyby aproximace

- zajímá nás schopnost generalizace
- více jednotek – lepší aproximace, větší náchylnost k přeučení

Experimentální výsledky s RBF sítěmi

# jednotek	data	Trénovací chyba	Testovací chyba
10	cancer	3.57	3.59
20		2.93	3.04
50		2.24	3.07
15	glass	7.50	9.90
30		5.94	13.05
50		4.61	112.95
30	hearta	4.32	4.74
40		4.11	4.66
50		4.05	4.75

Závislost velikosti sítě a chyby aproximace

- teoretické výsledky zabývající se závislostí velikosti sítě a chyby aproximace

  Xu, Krzyzak, Yuille,1994

  Corradi, White, 1995

  rychlost konvergence $O(n^{-2m/(2m+1)})$

  Kůrková, Sanguineti, 2005

  $O(\frac{1}{\sqrt{n}} G(N, \vec{y}, K, \gamma))$

- možnosti budoucí práce
  - využití teoretických výsledků v učících algoritmech
  - kriterium zastavení pro inkrementální učení
  - potřeba zkoumat vztah mezi teoretickými a experimentálními výsledky

Časová složitost jednotlivých algoritmů

- v experimentální části práce měřené v sekundách
- důvody:
  - různorodost používaných algoritmů
  - nelze porovnávat iterace
  - knihovna PAPI - technické problémy na některých platformách
- lze usuzovat na časovou náročnost obdobně velkých sítí na obdobně velkých datech
- na datech z Proben1 třífázové učení trvá řádově sekundy, gradientní minuty, genetické až hodiny
- rychlost konvergence (gradientní učení, GA) závisí na daném problému

PAC učení

- jakou velikost tréninkové množiny potřebujeme, aby se síť naučila s *velkou pravděpodobností* danou funkci s požadovanou přesností
- existují dílčí teoretické výsledky: Holden, Rayner; Generalization and PAC Learning: Some Results for the Class of Generalized Single-Layer Networks
- omezeno na klasifikaci do dvou tříd
- př. pro síť s $W = 200$ váhami, $\varepsilon = \frac{1}{4}$ a důvěryhodnost $1 - 8e^{-1.5W}$: požadovaný počet příkladů 124 000
- velmi volný odhad
- reálné výstupy - stále otevřené