JÁDROVÉ ODHADY
IMPLEMENTACE, ILUSTRATIVNÍ PŘÍKLADY, APLIKACE

VÍTĚZSLAV VESELÝ

Katedra aplikované matematiky PřF MU Brno

1. ÚVOD - SYSTÉM MATLAB

Příspěvek je věnován počítačové implementaci jádrových odhadů regresní funkce a navazuje na teoretický výklad J. Michálka publikovaný rovněž v tomto sborníku [1].

Pro implementaci metody jádrového vyhlazování byl zvolen systém MATLAB americké firmy MathWorks, Inc. [2]. Důvodů pro tuto volbu je možno vyjmenovat několik. Především je to však vysoká efektivita programátorské práce umožňující v krátké době odladit i poměrně logicky velmi složité algoritmy. Tato vlastnost MATLABu je dána v zásadě čtyřmi faktory:

a) Základním a vlastně jediným datovým typem jsou vektory a matice nad oborem komplexních čísel. Dokonce i textové řetězce jsou chápány jako vektory ASCII kódů odpovídajících znaků. Pro práci s maticemi lze používat operátory běžné v maticové algebře, což velmi usnadňuje a zpřehledňuje zápis i těch nejkomplikovanějších algoritmů.

b) Funkční procedury v MATLABu jsou vytvářeny jednoduchým způsobem jako běžné ASCII soubory (tzv. M-soubory), které není třeba kompilovat, neboť jazyk MATLABu je v zásadě interpretační. To výrazně urychluje celý proces ladění, neboť opakovaný cyklus opravy a běhu je bezprostřední. Přitom při maximálním využití vektorově-maticového aparátu lze většinou nevýhodu obecně nižší rychlosti spojené s interpretováním jednotlivých příkazů za běhu programu do značné míry potlačit, neboť maticové operace jsou realizovány interními kompilovanými procedurami. Tato možnost se nabízí i uživateli, který může výpočetně náročné části svého algoritmu realizovat v hostitelském programovacím jazyku ("C",FORTRAN) ve formě dynamicky připojovaných procedur, tzv. MEX-souborů, a tak finální verzi dostatečně optimalizovat i co se týče rychlosti.

c) Rozsáhlá nabídka hotových funkcí, které pokrývají nejen základní potřeby aplikované a numerické matematiky, ale i řadu speciálních oblastí jako je grafika (i prostorová), zpracování signálů, optimalizce, splajny, statistika, identifikace systémů, řízení a regulace procesů aj. Tyto specializované balíky funkcí (tzv. toolboxy) jsou zpravidla vytvářeny vedoucími specialisty v daném oboru. Aktivity tohoto druhu jsou široce rozšířeny především v oblasti zpracování dat a umožňují efektivní spolupráci a výměnu programů i s jinými spolupracujícími pracovišti.

d) Systém MATLAB je k dispozici na nejrůznějších počítačových platformách, a to zejména na IBM PC (AT 286, 386, 486) pod operačním systémem MS DOS i WINDOWS a na UNIXových systémech pod X-WINDOWS. Vzhledem k ASCII formátu M-souborů lze vytvořené programy snadno přenášet mezi jednotlivými platformami.

2. IMPLEMENTACE JÁDROVÝCH ODHADŮ

Soubor procedur (M-souborů) pro jádrové vyhlazování je koncepčně navržen jako otevřený modulární systém, který umožňuje řešit v zásadě úlohy dvojího typu.

(1) Nalézt výsledný průběh obecně k-té derivace (k=0,1,...) veličiny Y zatížené náhodnými vlivy v závislosti na veličině X, jejíž hodnoty jsou buď pevně předepsány (model s pevným plánem) a nebo jsou rovněž náhodného charakteru (model s náhodným plánem).

(2) Nalézt odhad hustoty pravděpodobnostního rozložení náhodné veličiny X. V tomto případě může být odhad hustoty hledán buď v kombinaci s vyhlazováním a nebo jako samostatná úloha, pokud data Y nejsou dána.

Typeset by $\mathcal{AMS}$-TEX

Modularita a otevřenost je motivována snahou rozložit problém na logicky relativně nezávislé celky zajišťující budoucí rozvoj variability funkcí systému (např. volbu různých typů vah a tvarů jader, výběr strategií pro hledání počátečních i optimálních odhadů šířek okna a pro korekce okrajových efektů).

V dalším bude popsána struktura a logika fungování jednotlivých komponent systému odpovídající danému stavu vývoje. Ten v současnosti umožňuje zpracovávat pouze 1-dimenzionální nekorelovaná data s homogenním rozptylem v ose Y, nemá implementovány odhady derivací (k=0), používá pouze jeden typ vah [1,(5)], poskytuje omezený výběr několika základních tvarů jader různé hladkosti (např. gaussovské, exponenciální, kvartické, obdélníkové aj.) a nabízí jen po jedné metodě pro optimalizaci šířky okna a korekci okrajových efektů.

V budoucnu se kromě rozšíření variability systému ve výše uvedených směrech počítá mimo jiné s
- optimalizací algoritmů z hlediska výpočetní rychlosti, založené především na využití rychlé Fourierovy transformace,
- implementací technik proměnné šířky okna, kdy lze vypustit předpoklad na homogenitu rozptylu dat Y a nekorelovanost dat,
- přechodem do více dimenzí.

Procedury systému (M-soubory) lze podle jejich účelu rozčlenit do čtyř skupin.

A. Výpočetní procedury tvaru jádra, resp. jeho derivace.

Pro zajištění dostatečné kompatibility a současně univerzálnosti použití mají tyto procedury předepsán následující standardizovaný formát vstupních a výstupních parametrů:

[k,k_p,k_i] = k_name(k_o,x),

kde

k_name udává jméno procedury pro výpočet jádra uvozené předponou ***k_*** (k=kernel).

Předpokládá se rovněž vhodně normalizovaná šířka jádra - např. nosič [-1,1] pro jádra s kompaktním nosičem.

<u>Vstupní parametry:</u>

k_o ... vektor řídících parametrů (options) pro výpočet jádra (např. řád derivace apod.); je-li zadán neúplně, musí procedura zajistit dosazení vhodných implicitních hodnot,

x ... vektor hodnot nezávisle proměnné, v nichž požadujeme spočíst průběh tvaru jádra se zadanými parametry ***k_o***.

<u>Výstupní parametry:</u>

Mají různý význam podle toho, zda byl nebo nebyl zadán vektor x.

a) *x nebyl zadán nebo x=[ ] (prázdná matice)*
V tomto případě proběhne fáze tzv. **identifikace typu jádra** - předchází zpravidla jako 1. krok před vlastním výpočtem.

k ... řetězec udávající jméno jádra, které musí být použito ve fázi výpočtu,
k_p ... vektor skutečných parametrů, kterých musí být použito ve fázi výpočtu,
k_i ... textová matice identifikující jádro a význam jeho parametrů:
1. řádek ***k_i*** udává stručnou charakteristiku jádra použitelnou např. jako popis do grafu. Zbývající řádky ***k_i*** podrobně popisují jádro a význam jeho parametrů, což je možno využít např. při interaktivním vkládání parametrů, kdy ***k_name*** je procedura zajišťující pouze volbu skutečného (elementárního) jádra ***k*** a jeho parametrů ***k_p***. Jestliže ***k_name*** reprezentuje elementární jádro, pak na rozdíl od předchozí situace je zpravidla ***k=k_name*** a ***k_p=k_o***.

b) *x byl zadán*
V tomto případě proběhne fáze vlastního výpočtu tvaru jádra.
k ... vektor hodnot jádra spočtených v hodnotách sítě x,
k_p, ***k_i*** jsou bezvýznamné parametry.

B. Výpočetní procedury pro počáteční odhad šířky okna.

Počítají asymptoticky optimální (pro velké n) hodnotu počátečního odhadu šířky okna $h = h(n)$, kde n udává délku vektoru dat X a Y.

Předpokládá se opět tento standardizovaný formát vstupních a výstupních parametrů:
[h,h_p,h_i] = h_name(h_o,n),

kde

h_name udává jméno procedury pro výpočet odhadu šířky okna uvozené předponou *h_*.

Vstupní a výstupní parametry mají analogický význam jako v předchozím případě.

C. Hlavní výkonná procedura.

[m,f,h_n,RSS,R,sig2] = k_smooth(x,X,Y,K,k_o,h,h_o,options,k_tit)

<u>Vstupní parametry:</u>

- x ... vektor hodnot (sítě) nezávisle proměnné, v nichž je požadován výpočet vyhlazeného průběhu m, resp. odhadu hustoty pravděpodobnosti f pro X;
- X ... vektor zadaných, resp. pozorovaných hodnot nezávisle proměnné X;
- Y ... vektor pozorovaných hodnot závisle proměnné Y, je-li Y prázdný (Y=[ ]), spočte se pouze odhad hustoty pravděpodobnosti f;
- K ... textový řetězec udávající jméno (*k_name*) procedury, jež má být použita pro výpočet tvaru jádra (viz A);
- *k_o* ... vektor řídících parametrů pro K;
- h ... textový řetězec udávající jméno (*h_name*) procedury, jež má být použita pro výpočet počátečního odhadu šířky okna (viz B), tento řetězec může být i prázdný - viz popis *h_o*;
- *h_o* ... vektor řídících parametrů pro h, resp. přímo dosazovaná šířka okna, pokud h = [ ];
- *options* ...

  vektor řídících parametrů procedury *k_smooth*, pokud je zadán neúplně, platí vhodné implicitní hodnoty;

  options(1) = 0 ... model s pevným plánem (implicitní),
  = 1 ... model s náhodným plánem,

  options(2) = 1 ... typ váhy (1,2, ...), implicitně 1

  options(3) = typ kritéria optimality (strategie) použité šířky okna *h_n*, (viz výstupní parametr R) implicitně 1;

  options(4) = řídící parametr optimalizace šířky okna
  = 0 ... neoptimalizovat (implicitní),
  < 0 ... optimalizovat bez kontrolního výpisu průběhu optimalizace,
  > 0 ... optimalizovat s kontrolním výpisem průběhu optimalizace,
  abs(options(4)) = mezní hodnota změny účelové funkce, po jejímž dosažení má být aproximační proces ukončen (menší hodnota znamená větší přesnost - viz standardní proceduru MATLABu *fmins*,

- *k_tit* ... nepovinný textový řetězec identifikující zpracovávanou úlohu a sloužící současně k řízení grafického výstupu. Jestliže tento parametr vynecháme, bude grafický výstup úplně potlačen.

  syntaxe *k_tit*: **'titulek>kreslicí zařízení?'**

  titulek ... titulní text, který se objeví v záhlaví grafu, jeho úvodní jeden až dva znaky mohou nepovinně udávat typ čáry pro vykreslení grafu vyhlazeného průběhu dle konvence příkazu *plot* MATLABu.

  >kreslicí zařízení? ...

  specifikuje kreslicí zařízení pro výstup grafu. Jestliže je tato část celá vynechána, bude graf vykreslen pouze na obrazovce,

  kreslicí zařízení :== prn nebo PRN ... výstup na tiskárnu
  :== jméno tiskového meta souboru MATLABu
  (implicitně meta soubor se jménem vytvořeným z úvodních znaků titulního textu),

  ? ... nepovinný znak; pokud je uveden, pak skutečné provedení výstupu bude vázáno na souhlas obsluhy až za běhu procedury.

Pro snadnější sestavení tohoto řetězce lze použít pomocnou proceduru *kplot* (viz D).

Výstupní parametry:

m ... vyhlazená veličina Y v hodnotách sítě x, m je prázdný vektor, pokud Y byl prázdný;
f ... odhad hustoty X v hodnotách sítě x; pro náhodný plán je na grafu příslušný jádrový odhad pro porovnání doplněn histogramem z X;
h_n ... skutečná šířka okna použitá při výpočtu m a f, tj. buď přímo dosazená jako vstupní parametr (při $h = [\,]$) nebo její asymptotický odhad spočtený pomocí h, resp. její optimalizovaná hodnota, pokud *options(4)*$\neq$ 0;
RSS ... reziduální součet čtverců (Residual Sum of Squares), viz [1];
R ... hodnota kritéria optimality (standardně [1,(K1)]) pro použitou šířku okna *h_n*. Typ použitého kritéria je určen parametrem *options(3)* a používá se jako účelová funkce při optimalizaci šířky - viz pomocnou proceduru *h_criter* v části D;
sig2 ... vektor odhadu rozptylu dat D(Y—X) ve směru osy Y několika metodami (v současnosti dva odhady - viz [1]).

D. Pomocné procedury.

(1) *options = koptions(parain)*

Procedura vrací úplný vektor řídících parametrů pro *k_smooth*. Hodnota *parain(k)* předepisuje hodnotu *k*-tého řídícího parametru, ostatní parametry nabudou svých implicitních hodnot.

(2) *k_tit = kplot(titulek,kreslicí_zařízení,znak_otazníku)*

Generuje řetězec *k_tit* pro *k_smooth* z jeho dílčích komponent.

(3) *r = h_criter(h,X,Y,K,k_o,options,k_tit)*

Vyhodnocuje hodnotu účelové funkce pro hledání optimální šířky *h_n* okna pomocí standardní optimalizační procedury *fmins* MATLABu [2].

r = abs(R), kde R je výstupní parametr získaný z

[*m,f,h_n,RSS,R*] = *k_smooth(X,X,Y,K,k_o,[ ],h,options,k_tit)*

s volbou *options(4)=0*.

Procedura *h_criter* je nepřímo volána prostřednictvím *fmins* v proceduře *k_smooth* v případě, že je požadována optimalizace (*options(4)* $\neq$ 0). Procedura *k_smooth* tak nepřímo rekurzívně vyvolává sama sebe tolikrát, kolikrát to vyžaduje *fmins* během jednotlivých optimalizačních kroků. Aby nedošlo k nekonečnému hnízdění rekurze, musíme pro *fmins* nastavit *options(4)=0*, ostatní řídící parametry a *X,Y,K,k_o* odpovídají vstupním hodnotám z prvního volání *k_smooth*. *h* je hodnota šířky okna dosazovaná automaticky procedurou *fmins*. Při spuštění *fmins* zadáme její počáteční odhad, po skončení optimalizace *fmins* vrací nalezenou optimální hodnotu.

3. ILUSTRATIVNÍ PŘÍKLADY

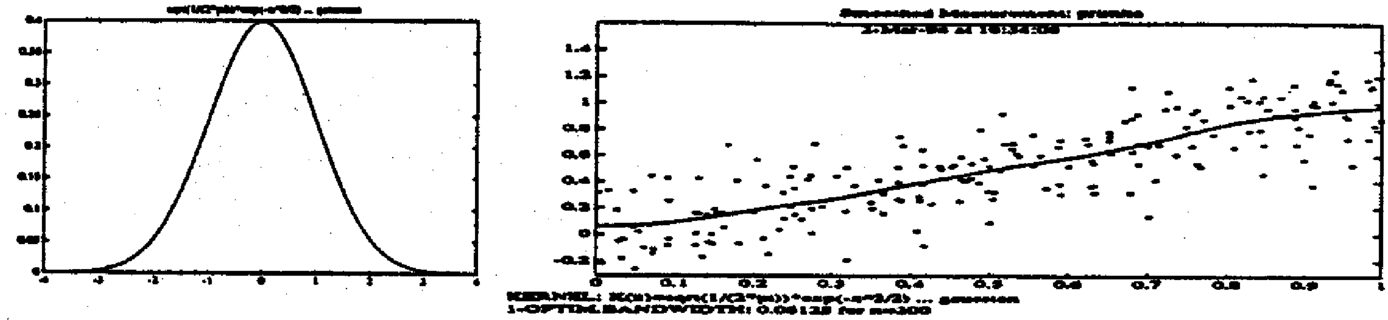
V tomto odstavci je metoda jádrového vyhlazování demonstrována na několika příkladech simulovaných i reálných dat. Pro každý typ příkladu je vždy uvedeno příslušné volání procedury *k_smooth*.

Grafický výstup zpracování jednotlivých příkladů je znázorněn na obrázcích. V záhlaví je vždy titulek určený parametrem *k_tit* při volání *k_smooth*. Ve spodní části je text identifikující použité jádro (*KERNEL*) a skutečná hodnota *h_n* šířky okna (*BANDWIDTH*). Text za *K(x)=* , resp. za *h(n)=* odpovídá řetězci vracenému v 1.řádku *k_i*, resp. *h_i* použité procedury *K*, resp. *h*. Použití optimalizace je indikováno např. textem *1-OPTIM.BANDWIDTH*, kde 1 značí typ použitého kritéria *R* (viz parametr *options(3)*).

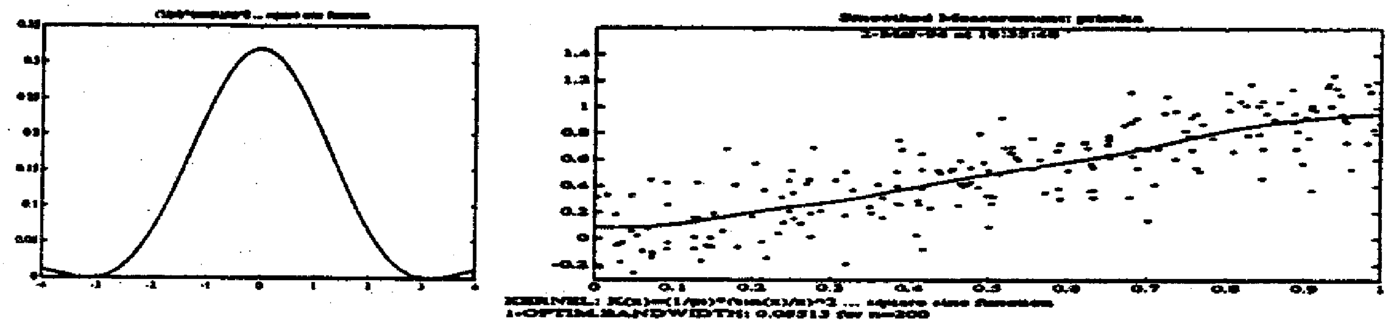
A. Simulovaná data.

(A1) Demonstrace vlivu hladkosti jádra (obr.1)

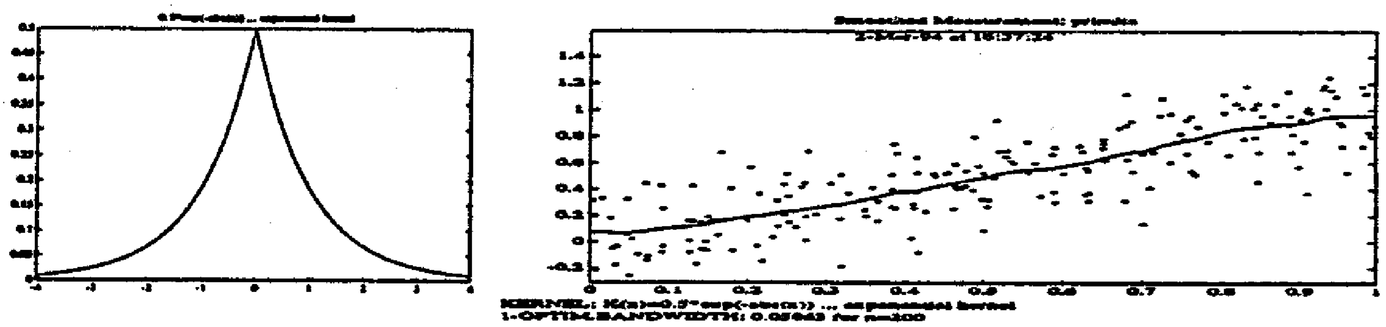
Popis simulace:
- Přímka $Y = X + e$ na intervalu [0,1] se simulovaným gaussovským šumem $e \sim N(0,0.04)$.
- Model s náhodným plánem, kde nezávisle proměnná X má simulované rovnoměrné rozložení na intervalu [0,1].
- Vyhlazení s použitím sedmi jader o různé hladkosti při optimalizované šířce okna (obr.1a-g):



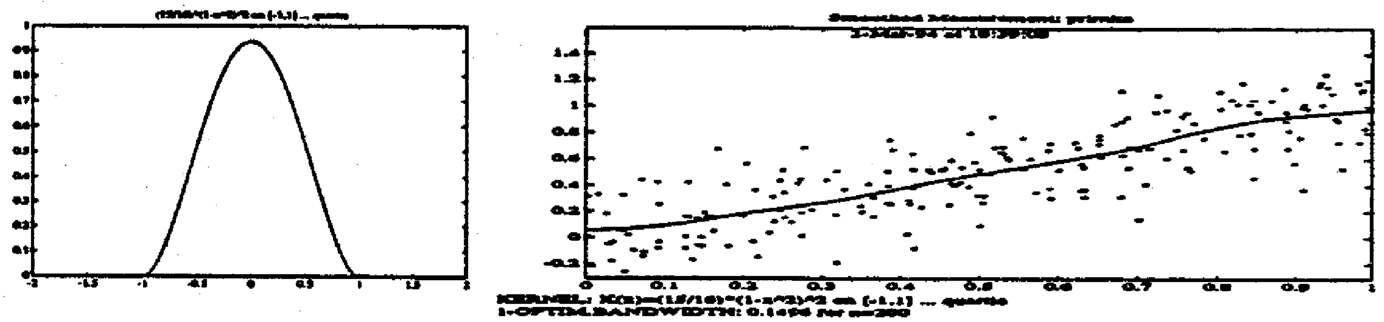
a) gaussovské jádro



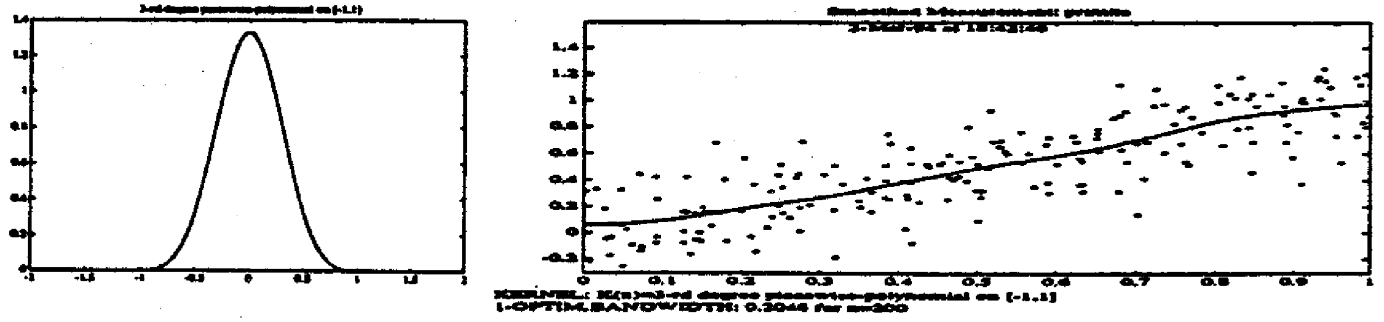
b) jádro sinc



c) exponenciální jádro

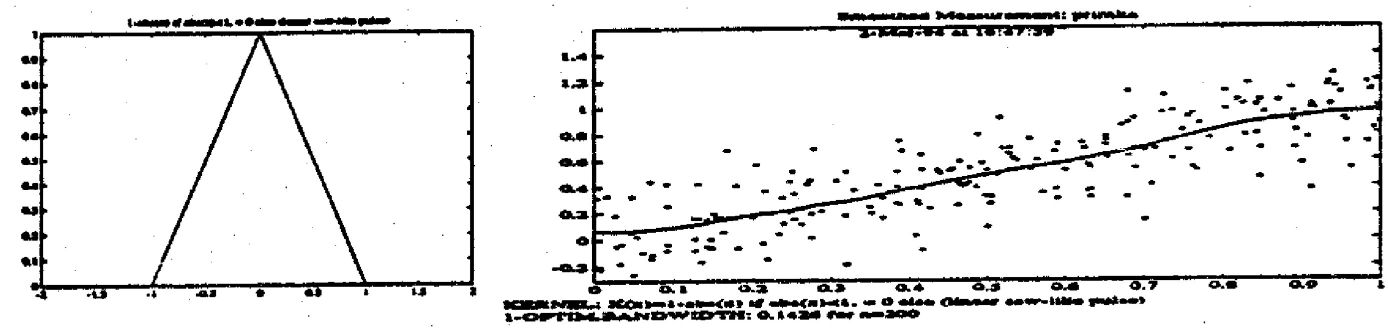


d) kvartické jádro

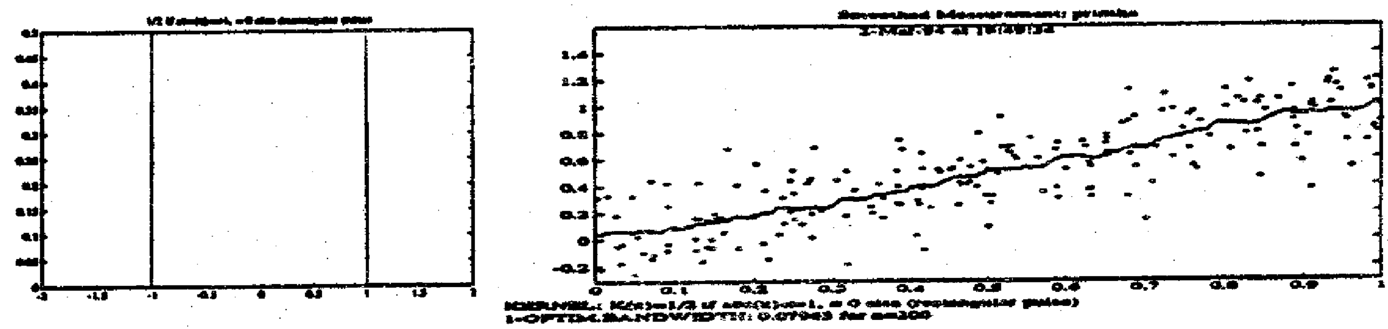


e) po částech polynomiální jádro

Obr. 1. Vliv hladkosti jádra na vyhlazení

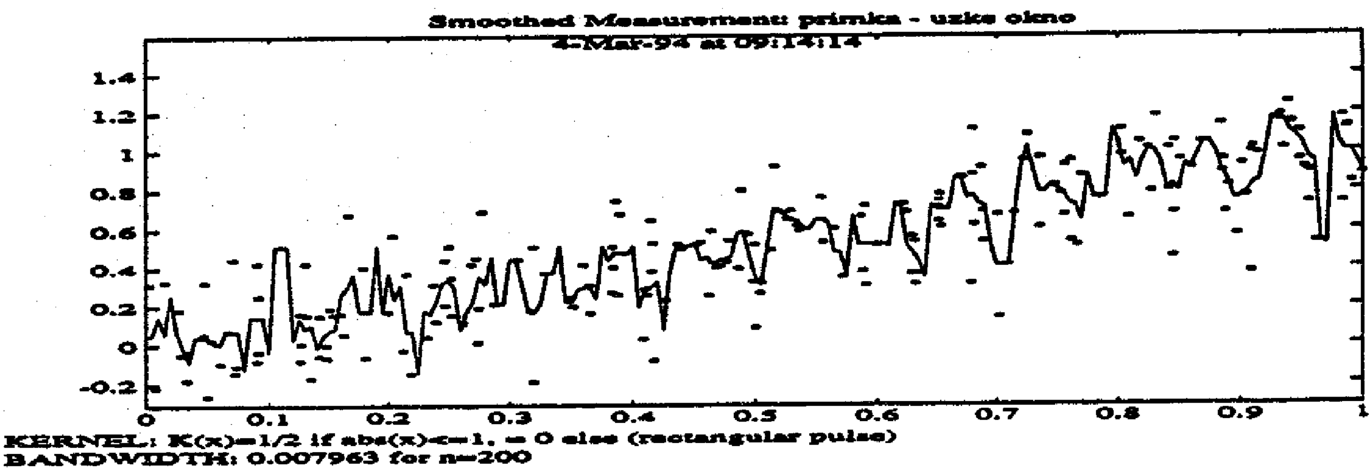


f) trojúhelníkové jádro

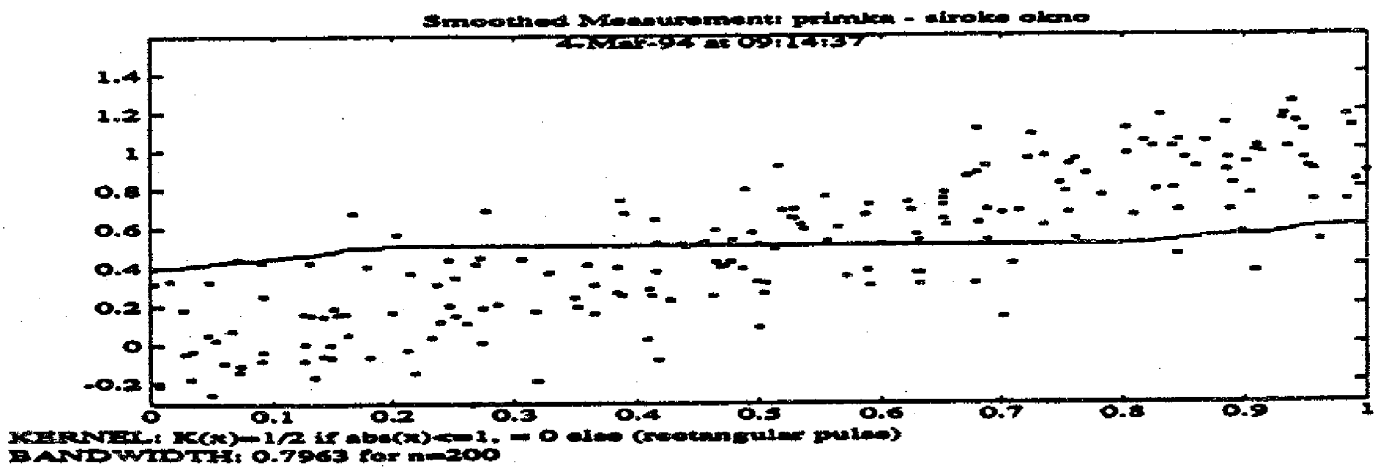


g) obdélníkové jádro

Obr. 1 (pokr.). Vliv hladkosti jádra na vyhlazení



a) malá šířka



b) velká šířka

Obr. 2. Vliv šířky obdélníkového jádra na vyhlazení

Parametry výpočtu:
options = [1,1,1,0.001];
h_0 = 0.07;
[*m,f,h_opt,RSS,R,sig2*] = *k_smooth(x,X,Y,kernel*,[ ],[ ],*h_0,options*,'primka>obr.1'*);*

Hodnoty výstupních parametrů:
sig2 = [*0.0476,0.0490*]
... odhad rozptylu je ve všech sedmi případech stejný (závisí pouze na Y) a dobře koresponduje s teoretickou hodnotou 0.04.

a) gaussovské jádro (*kernel*='k_gauss'): $\sqrt{1/(2\pi)}\,exp(-x^2/2)$
h_opt = 0.0613, *RSS* = 0.0445, *R* = 1.6930e-06.

b) kvadrát funkce *sinc* jako jádro (*kernel*='k_sinc'): $(1/\pi)(sin(x)/x)^2$
h_opt = 0.0551, *RSS* = 0.0448, *R* = -7.1607e-06.

c) exponenciální jádro (*kernel*='k_exp'): $0,5\,exp(-abs(x))$
h_opt = 0.0586, *RSS* = 0.0435, *R* = -1.1093e-05.

d) kvartické jádro (*kernel*='k_quart'): $(15/16)(1-x^2)^2$ na intervalu $[-1,1]$
h_opt = 0.1496, *RSS* = 0.0446, *R* = -2.5034e-06.

e) po částech polynomiální jádro 3. stupně (*kernel*='k_pol3'):
$4/3 - 8x^2 + 8abs(x)^3$ pro $abs(x) < 1/2$
$(8/3)(1-abs(x))^3$ pro $1/2 \le abs(x) \le 1$
0 jinak
h_opt = 0.2048, *RSS* = 0.0445, *R* = 2.0647e-06.

f) po částech lineární jádro - trojúhelníkový puls (*kernel*='k_lin'): $1 - abs(x)$ pro $abs(x) < 1$, 0 jinak
h_opt = 0.1426, *RSS* = 0.0442, *R* = 3.6570e-06.

g) obdélníkové jádro (*kernel*='k_rect'): 1/2 pro $abs(x) < 1$, 0 jinak
h_opt = 0.0796, *RSS* = 0.0445, *R* = -4.5599e-05.

<u>(A2) Demonstrace vlivu šířky okna (obr.2)</u>

Popis simulace:
- Simulovaná data i model jsou stejné jako v případě (A1).
- Vyhlazení s použitím obdélníkového jádra se šířkou výrazně odlišnou od optimální hodnoty *h_opt* = 0.0796 ad (A1g).

Parametry výpočtu: options = [1,1,1];

Hodnoty výstupních parametrů:

a) malá šířka (*h=h_opt/10=0.0080*):
[*m,f,h,RSS,R*] = *k_smooth(x,X,Y*,'k_rect',[ ],[ ],*h,options*,'primka - uzke okno>obr.2a'*);*
RSS = 0.0334, *R* = 0.0157.

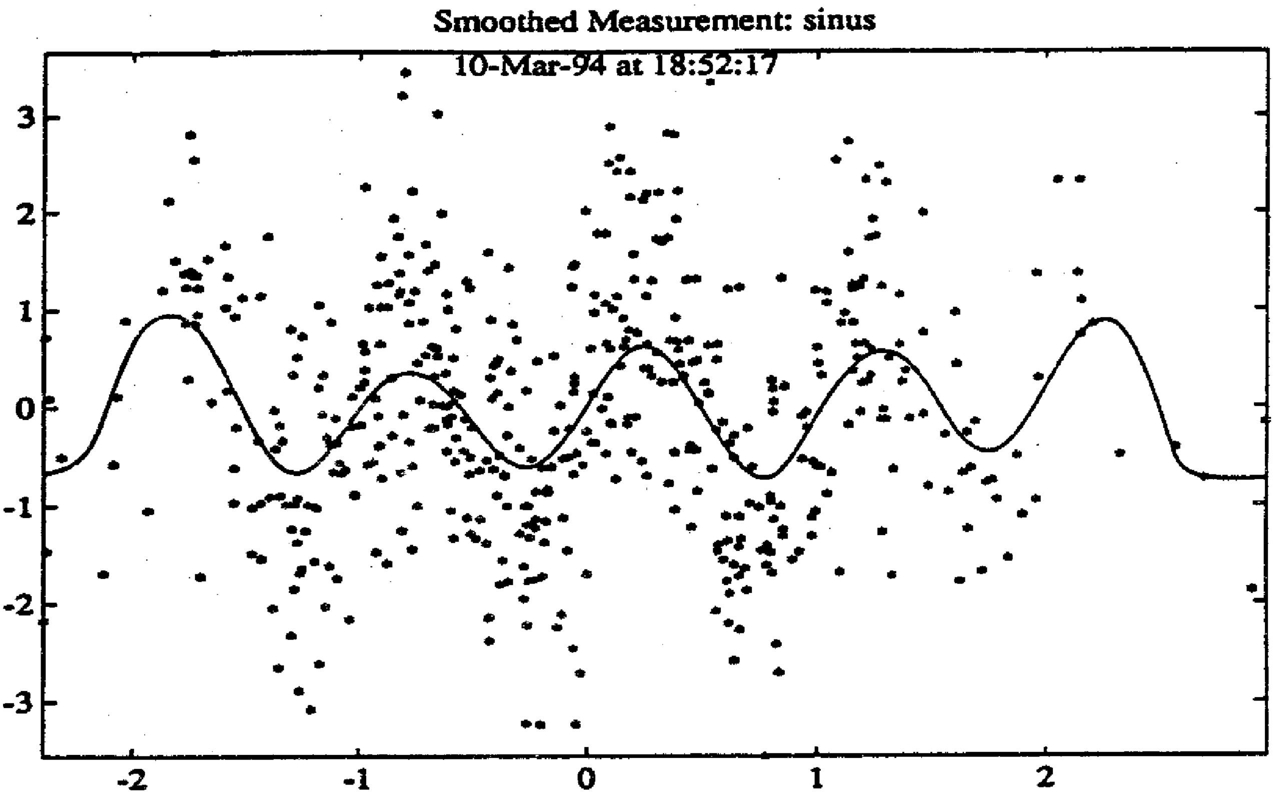
b) velká šířka (*h=h_opt*10=0.7963*):
[*m,f,h,RSS,R*] = *k_smooth(x,X,Y*,'k_rect',[ ],[ ],*h,options*,'primka - siroke okno>obr.2b'*);*
RSS = 0.1186, *R* = 0.0714.

<u>(A3) Detekce sinusového kmitu ze silného šumu (obr.3)</u>

Popis simulace:
- Sinusovka $Y = sin(2\pi X) + e$ se simulovaným gaussovským šumem $e \sim N(0,1)$.
- Model s náhodným plánem, kde nezávisle proměnná X má simulované normální rozložení $N(0,1)$.
- Vyhlazení a odhad hustoty X s použitím kvartického jádra při optimalizované šířce okna (obr.3a-b):

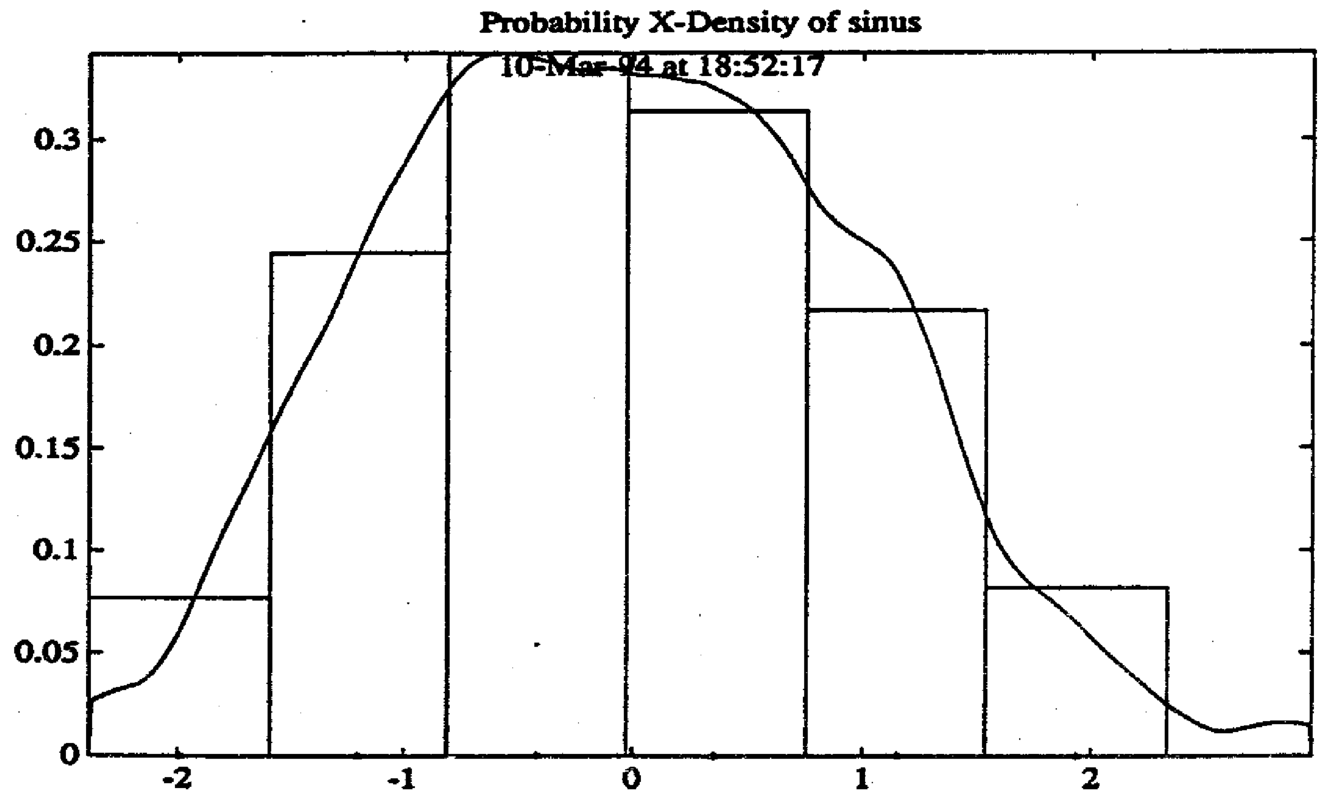
Parametry výpočtu:
options = [1,1,1,0.01]
h_0 = 0.35
[*m,f,h_opt,RSS,R,sig2*] = *k_smooth(x,X,Y*,'k_quart',[ ],[ ],*h_0,options*,'sinus>obr.3'*)*

Hodnoty výstupních parametrů:
h_opt = 0.4411, *RSS* = 1.1404 , *R* = -8.2873e-05, *sig2* = [1.1502,1.1613].



KERNEL: K(x)=(15/16)*(1-x^2)^2 on [-1,1] ... quartic
1-OPTIM.BANDWIDTH: 0.4411 for n=500

a) vyhlazená data



KERNEL: K(x)=(15/16)*(1-x^2)^2 on [-1,1] ... quartic
1-OPTIM.BANDWIDTH: 0.4411 for n=500

b) odhad hustoty v ose x

Obr. 3. Detekce sinusového kmitu ze silného šumu

B. Reálná data.

Data použitá v následujících příkladech pravděpodobně nesplňují striktně předpoklady nekorelovanosti a homogenity rozptylu. Přesto získané výsledky dobře ilustrují možnosti metody jádrového vyhlazování.

<u>(B1) Měření atmosférického tlaku (obr.4)</u>

Popis zpracování dat:
- Data reprezentují denní hodnoty atmosférického tlaku měřeného v Brně Tuřanech v období od 23.5.1985 do 30.9.1985.
- Model s pevným plánem, kde nezávisle proměnná X=den.
- Vyhlazení se provádí s použitím kvartického jádra při optimalizované šířce okna.

Parametry výpočtu:
options = [0,1,1,0.001]

Hodnoty výstupních parametrů:
[*m,f,h_opt,RSS,R,sig2*] = *k_smooth(den,den,tlak,*'k_quart',[ ],'h_x',[ ],*options,*'atmosfericky tlak>obr.4'*);*
h_opt = 4.7299, *RSS* = 9.9252, *R* = -6.9518e-05, *sig2* = [9.9554,7.7440].

<u>(B2) Sezónní měření atmosférických teplot za 90 let (obr.5)</u>

Popis zpracování dat:
- Data reprezentují průměrné podzimní teploty vzduchu měřené v Hurbanově v létech 1903 až 1992.
- Model s pevným plánem, kde nezávisle proměnná X=rok.
- Vyhlazení se provádí s použitím kvartického jádra jednak s šířkou zvolenou subjektivně na základě experimentování a jednak s optimalizovanou šířkou.

Parametry výpočtu:
options = [0,1,1,0.01]
h_0 = 4

Hodnoty výstupních parametrů:

a) Experimentálně odhadnutá šířka *h_0=4* (obr.5a):
[*m,f,h,RSS,R,sig2*] = ...
k_smooth(rok,rok,teplota,'k_quart',[ ],[ ],*h_0*,[ ], 'podzimni teploty - odhad sirky okna>obr.5a'*);*
h = 4, *RSS* = 0.9603, *R* = -0.4729, *sig2* = [1.4407,1.5383].

b) Následně optimalizovaná šířka (obr.5b):
[*m,f,h_opt,RSS,R,sig2*] = ...
k_smooth(rok,rok,teplota,'k_quart',[ ],[ ],*h_0,options,* 'podzimni teploty - optimalni sirka okna>obr.5b'*);*
h_opt = 21.3625, *RSS* = 1.4393, *R* = -1.3568e-05, *sig2* = [1.4407,1.5383].

Tento záměrně vybraný příklad dobře ilustruje, jak lokální změna charakteru dat (v místě evidentní skokové změny) zvyšuje celkový vysokofrekvenční obsah datové řady, což vede při globální optimalizaci k falešné snaze o vyšší vyhlazení všech dat. Proto zřejmě *h_opt* vychází výrazně větší, než subjektivní odhad *h_0*. Pro úlohy tohoto typu se proto nabízí jako výhodnější technika proměnné šířky okna.

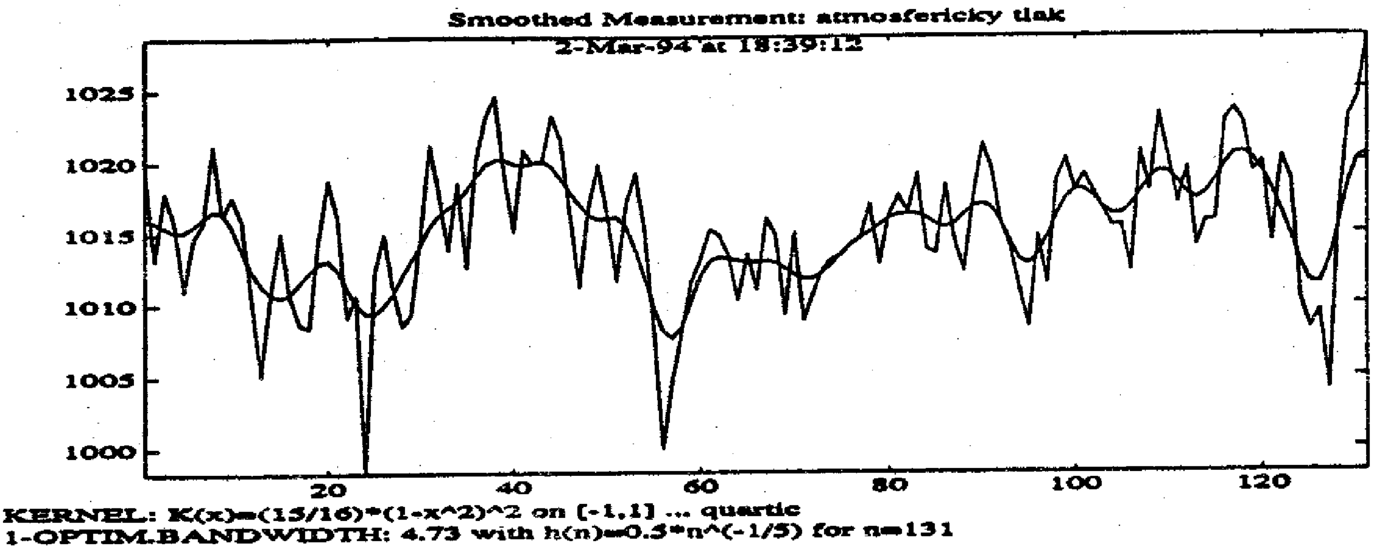
<u>(B3) Populační dynamika výskytu komárů (obr.6)</u>

Popis zpracování dat:
- Data reprezentují denní četnosti výskytu dvou druhů komárů v Želešicích u Brna v období od 24.5 - 1.10.1985 (f=samice).
- Model s pevným plánem, kde nezávisle proměnná X=den.
- Vyhlazení se provádí s použitím kvartického jádra při optimalizované šířce okna.

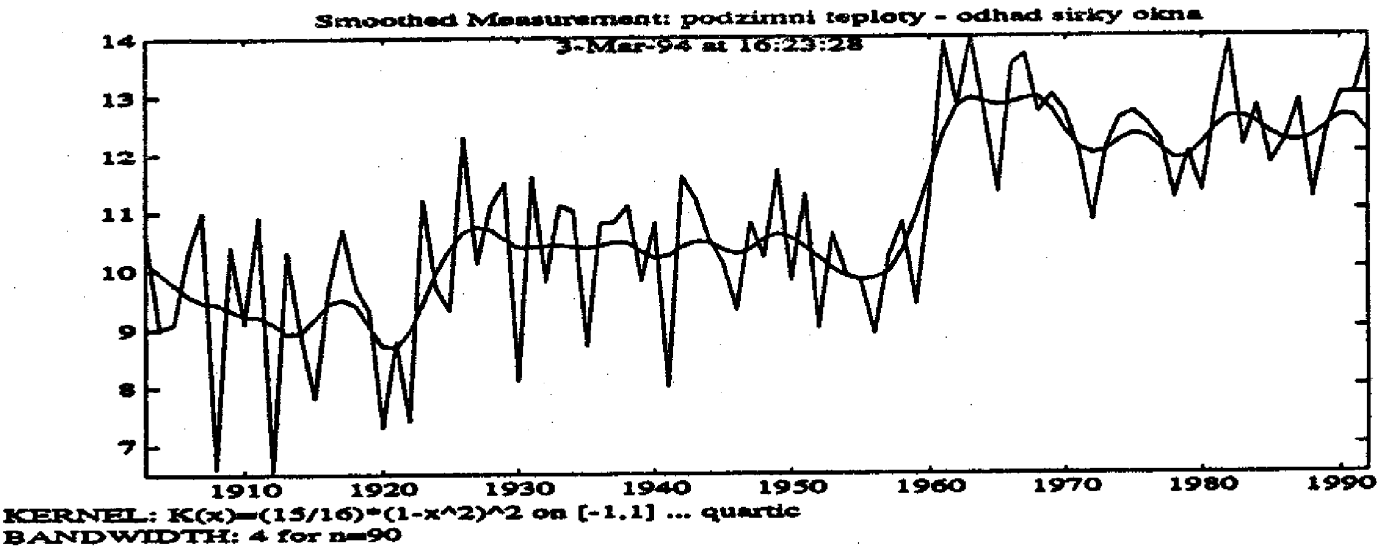
Parametry výpočtu:
options = [0,1,1,0.001]

Hodnoty výstupních parametrů:

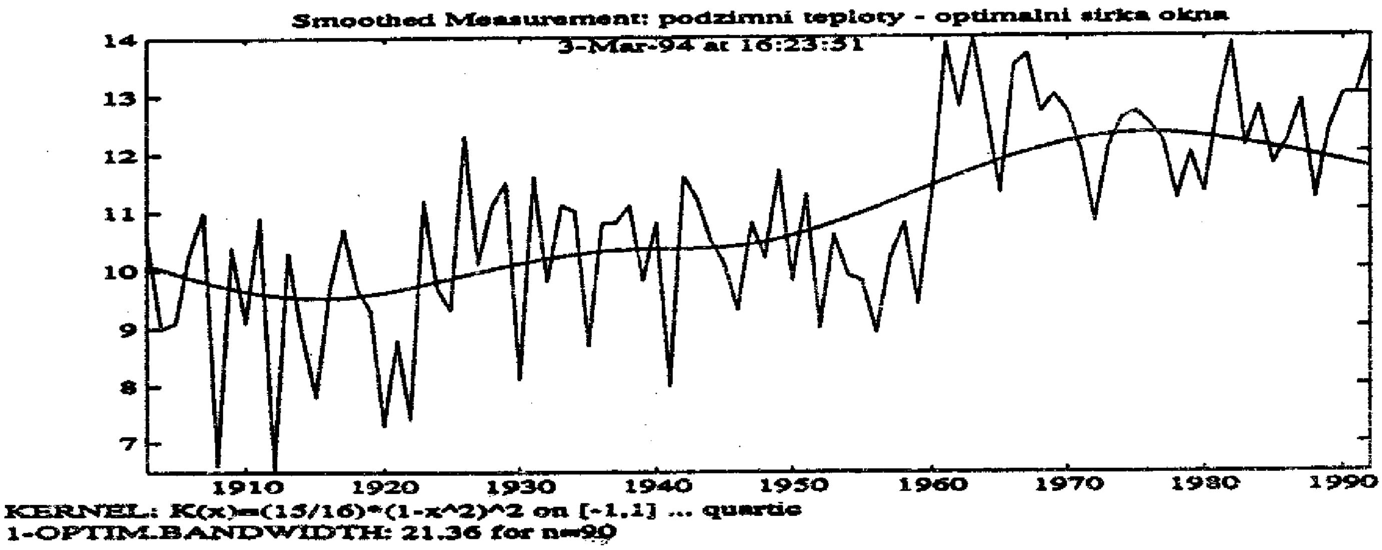
a) Druh komárů ***s21f*** (obr.6a):
[*m,f,h_opt,RSS,R,sig2*] = ...
k_smooth(den,den,s21f,'k_quart',[ ],'h_x',[ ],*options,*'druh komaru = s21f>obr.6a'*);*
h_opt = 14.5109, *RSS* = 23.7864, *R* = 2.3107e-05, *sig2* = [23.8101,23.0651].



OBR. 4. Měření atmosférického tlaku v Brně Tuřanech

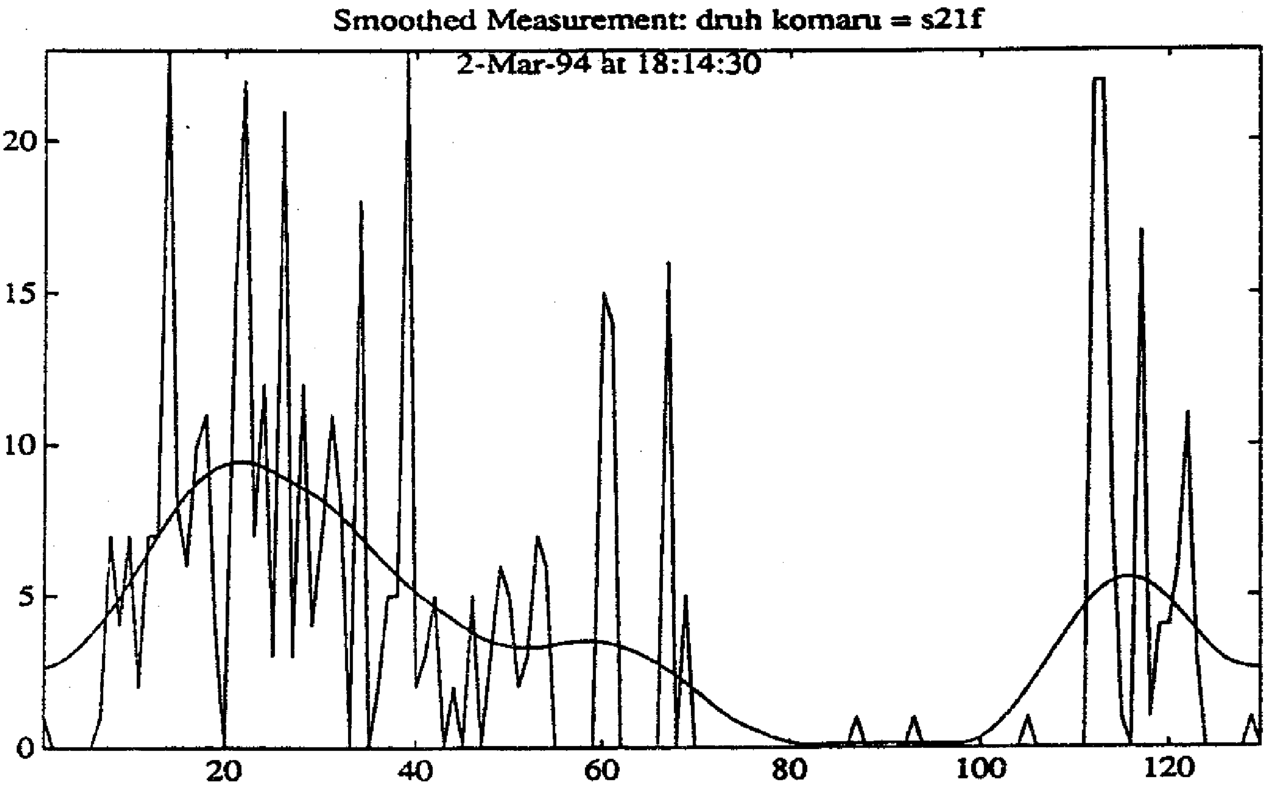


a) experimentálně odhadnutá šířka okna



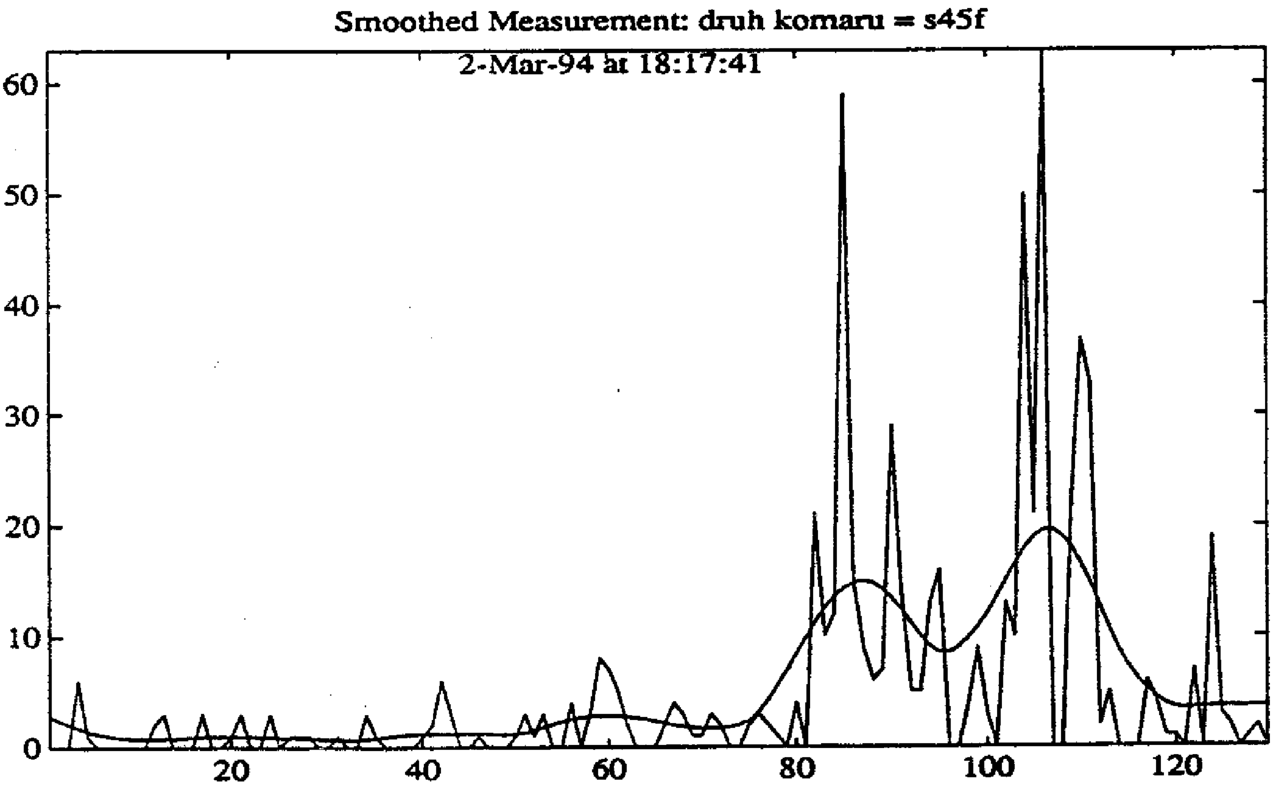
b) optimalizovaná šířka okna

OBR. 5. Průměrné podzimní teploty v Hurbanově



KERNEL: K(x)=(15/16)*(1-x^2)^2 on [-1,1] ... quartic
1-OPTIM.BANDWIDTH: 14.51 with h(n)=0.5*n^(-1/5) for n=130

a) druh komárů s21f



KERNEL: K(x)=(15/16)*(1-x^2)^2 on [-1,1] ... quartic
1-OPTIM.BANDWIDTH: 10.57 with h(n)=0.5*n^(-1/5) for n=130

b) druh komárů s45f

OBR. 6. Populační dynamika výskytu komárů v Želešicích u Brna

b) Druh komárů *s45f* (obr.6b):
[m,f,h_opt,RSS,R,sig2] = ...
k_smooth(den,den,s45f,'k_quart',[ ],'h_x',[ ],*options*,'druh komaru = s45f>obr.6b'*);*
h_opt = 10.5694, *RSS* = 66.4211, *R* = 2.3903e-04, *sig2* = [66.5116,64.7995].

LITERATURA

[1] J. Michálek: *Jádrové odhady - základní vlastnosti a optimální výběr parametrů odhadu.* sborník ROBUST'94 (1994).

[2] *MATLAB, High-Performance Numeric Computation and Visualization Software.* Reference Guide, The MathWorks, Inc. (1992).

Tato práce byla vypracována za finanční podpory Grantové Agentury České Republiky, reg. číslo grantu 201/93/2408.