CENTER FOR
MACHINE PERCEPTION

CZECH TECHNICAL
UNIVERSITY

MASTER THESIS

ISSN 1213-2365

Porovnání metod odhadování entropie pro registraci obrázků

Jiří Svoboda

svoboj8@fel.cvut.cz

CTU–CMP–2007–11

24. května 2007

Školitel: Dr. Ing. Jan Kybic

Research Reports of CMP, Czech Technical University in Prague, No. 11, 2007

Published by

Centrum strojového vnímání, Katedra kybernetiky
Fakulta elektrotechnická ČVUT
Technická 2, 166 27 Praha 6
fax: (02) 2435 7385, tel: (02) 2435 7637, www: http://cmp.felk.cvut.cz

České vysoké učení technické v Praze - Fakulta elektrotechnická

Katedra kybernetiky *Školní rok:* 2005/2006

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: Jiří Svoboda

Obor: Biomedicínské inženýrství

Název tématu: Porovnání metod odhadování entropie pro registraci obrázků

Zásady pro vypracování:

1. Seznamte se s existujícími metodami odhadování entropie, zvláště multidimenzionální.
2. Vybrané metody implementujte. Experimentálně vyhodnoťte jejich rychlost a statistické vlastnosti.

Seznam odborné literatury:

[1] Kraskov, A.; Stogbauer, H. and Grassberger, P.: Estimating mutual information, Phys. Rev. E (submitted), 2003, E-print, arXiv.org/cond-mat/0305641.

[2] J. Beirlant, E. Dudewicz, L. Gyorfi, and E. van der Meulen: Nonparametric entropy estimation: An overview. Int. J. Math. Stat. Sci. , 6(1):17-39, 1997

Vedoucí diplomové práce: Dr. Ing. Jan Kybic

Termín zadání diplomové práce: zimní semestr 2005/2006

Termín odevzdání diplomové práce: leden 2007

prof. Ing. Vladimír Mařík, DrSc.
vedoucí katedry

L.S.

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE Fakulta elektrotechnická -4-

v zast.

prof. Ing. Vladimír Kučera, DrSc.
děkan

V Praze dne 17.01.2006

Anotace

Registrace obrázků je důležitou disciplínou strojového vidění. V posledním desetiletí se začaly prosazovat pro unimodální i multimodální registraci metody využívající jako kriterium podobnosti mezi registrovanými obrázky vzájemnou informaci. Cílem práce je vyhodnotit statistické vlastnosti a rychlost implementací vybraných estimátorů entropie a vzájemné informace na datech různých dimenzionalit a pravděpodobnostních rozdělení. Jsou testovány estimátory využívající histogramování, vylepšení histogramu Parzenovým oknem či formou adaptivního binningu, estimátor založený na jádrovém odhadu hustoty pravděpodobnosti, estimátor založený na nejbližších sousedech i modifikace tohoto estimátoru urychlující vyhledávání záměnou nejbližších sousedů za přibližné nejbližší sousedy. Dále jsou vyhodnoceny vlastnosti estimátoru Rényi entropie z minimální kostry úplného grafu nad vzorky.

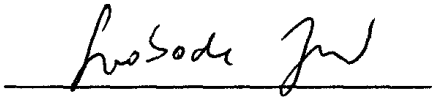
Abstract

Image registration is an important field of computer vision. In the last decade, methods using mutual information between registrered images as their similarity criterion have been gaining popularity. The aim of this work is to evaluate statistical properties and speed of certain entropy and mutual information estimator implementations on data of various dimensionalities and probability distributions. Among the estimators evaluated are: the histogram estimator, in its classical form and with enhancements such as histogram smoothing and adaptive binning, entropy and mutual information estimator based on kernel density estimation and a nearest-neighbor based estimator and its faster modifications replacing nearest-neighbors with approximate nearest-neighbors. Also, we assess statistical properties of an Renyi entropy estimator based on the length of a minimum spanning tree spanning the samples.

Prohlášení

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

V Praze dne 24. května 2007



Poděkování

Na tomto místě bych rád poděkoval všem, bez kterých by tato práce nemohla vzniknout. Zejména chci poděkovat vedoucímu mé práce Dr. Ing. Janu Kybicovi za cenné připomínky. Dále chci poděkovat osobám blízkým a rodině za trpělivost a podporu při studiu.

Obsah

Kapitola 1

Úvod

Cílem procesu registrace je nalézt transformaci, jež k sobě přiřadí odpovídající si body z dvou podobných obrázků; výstupem registrace je pak funkce, mapující na sebe tyto body. Registrování probíhá iterativně; jeden z obrázků je transformován funkcí, jež se každou iterací více blíží skutečné funkci korespondence. Za účelem popisu kvality transformace a jejího zvyšování v každém kroku je nutné definovat míru tuto kvalitu reflektující.

Při registraci často porovnávány podobné obrázky. Z tohoto důvodu se jeví logickou volba podobnostních měr porovnávajících absolutní hodnoty jasu pixelů sobě odpovídajících jako např. součet kvadrátů rozdílů či korelační koeficient. Tyto míry jsou však nevyhovující pro registraci obrázků bez prosté závislosti jasů odpovídajících si pixelů. Na prosté závislosti není omezena vzájemná informace, informačně teoretická veličina kvantifikující vzájemnou závislost náhodných proměnných. Vzájemná informace je proto vhodná k registraci dat pocházejících z různých modalit a v jiných případech nelineárních závislostí jasů korespondujících oblastí.

Obrázky jsou soubory vzorků, jež pochází z určitého neznámého pravděpodobnostního rozdělení. Z důvodu neznalosti tohoto rozdělení není možné počítat vzájemnou informaci analyticky. Existuje však řada postupů jak skutečnou hodnotu vzájemné informace odhadnout na základě vzorků obsažených v obrázku. V této práci jsou shrnuty výsledky experimentů s implementacemi vybraných estimátorů entropie a vzájemné informace. Testujeme jejich přesnost, odchylky od skutečných hodnot a rychlost na různých rozděleních vstupních dat. Estimátory byly napsány v jazyce C s ohledem na jednoduchost jejich integrace do jiných programů.

Kapitola 2

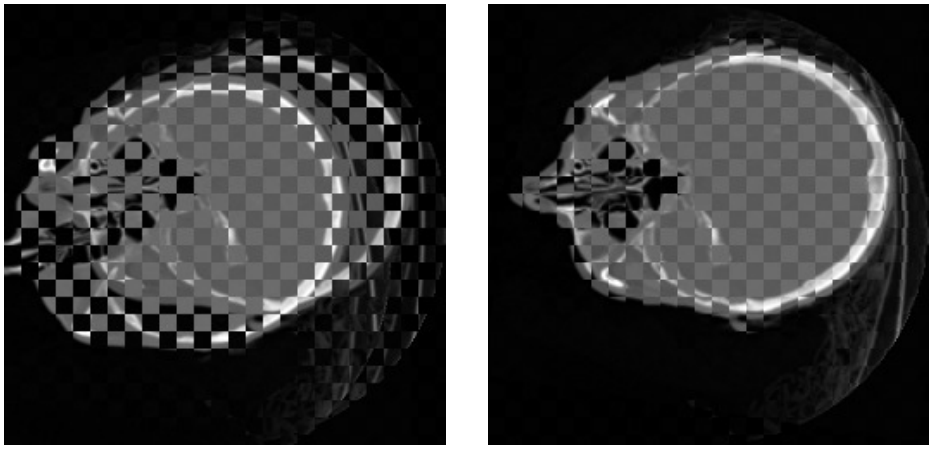
Základy registrace obrázků

2.1 Úvod

Registrace obrázků je jedním ze základních problémů strojového zpracování obrazové informace a je jednou z rychleji se vyvíjejících disciplín počítačového vidění. V mnoha úlohách se často setkáváme s nutností porovnat množinu obrázků, zjistit rozdíly mezi nimi či naopak nalézt podobné rysy, kvantifikovat míru odlišnosti, zjistit přítomnost určitého modelovaného objektu ve scéně či jej v této scéně lokalizovat. Tato úloha byla v minulosti z důvodu nedostatečného rozvoje výpočetní techniky možná pouze využitím práce operátora; strojová registrace obrázků je výsledkem úsilí převést řešení těchto úloh do výpočetní domény.

Registrace obrázků se uplatní v úloze rekonstrukce prostorového tvaru tělesa z odlišných fotografií; je třeba identifikovat korespondující body, aby bylo možno získat informaci o hloubce [17]. Díky registraci je umožněno skládání snímků do panoramatického záběru; zde je třeba určit změnu úhlu kamery mezi dílčími záběry [10]. Usnadňuje analýzu pohybu; není třeba detekovat hledaný objekt v každém snímku zvlášť, stačí vyhodnotit změny mezi snímky a tak získat informaci o rychlosti sledovaného objektu [43]. Registrace se také využívá při tvorbě obrázků se zvýšeným rozlišením *(superresolution images)*, kdy mírnou změnou úhlu snímání získáme informaci o bodech, nacházejících se mimo pravoúhlou mřížku některého z předchozích záběrů [40]. Širokou oblast využití poskytují geografické informační systémy. Informace z mnoha satelitních snímků mohou být sloučeny do jedné vrstvy a dále kombinovány s jinými než obrazovými daty [9, 44].

Systémy pro registraci obrázků jsou v současnosti široce využívány v mnoha aplikacích biomedicínského zobrazování; uplatní se při plánování i výkonu mnohých lékařských postupů. Pacienti během svého průchodu lékařskou péčí obvykle podstupují mnoho rozdílných anatomických i funkčních vyšetření, informace získané těmito postupy jsou obvykle komplementárního charakteru. Kombinací dat z různých zobrazovacích metod (např. výpočetní tomografie



Obrázek 2.1: Příklad registrace obrázků: dvojice podobných obrázků před (vlevo) a po registraci (vpravo) v šachovnicovém zobrazení. Data k dispozici na stránkách *The Stanford volume data archive*, <`http://graphics.stanford.edu/data/voldata/`>.

s magnetickou rezonancí aj.) je možné získat úplnější informaci o vyšetřovaném [25]. Častou aplikací je rovněž porovnání získaných dat s určitým standardem. Popis anatomickych struktur usnadňuje jejich registrace vůči „průměrným“ instancím s definovaným souřadným systémem (Talairachův atlas [37]). Častou aplikací je též porovnávání dat získaných s časovým odstupem (pooperační analýza, monitorování změn tkání aj.).

2.2 Formální definice registrace obrázků

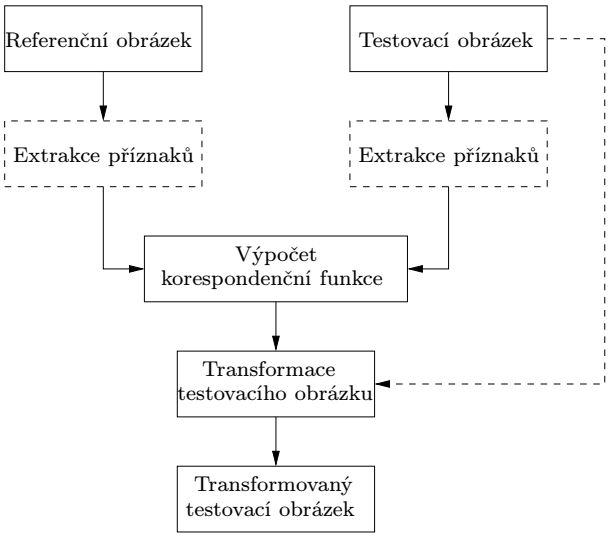
Mějme dva obrázky, zobrazující stejný či podobný objekt $p(\mathbf{x}_r)$ *(referenční obrázek, template)* a $q(\mathbf{x}_t)$ *(testovací obrázek, object)*. Pak *registrací obrazu (image registration)* nazýváme proces hledání funkce $\mathbf{f}$

$$\mathbf{x}_r = \mathbf{f}(\mathbf{x}_t). \tag{2.1}$$

Funkce $\mathbf{f}$ z rovnice (2.1) přiřazuje každému bodu z obrázku $q(\mathbf{x}_t)$ odpovídající bod z obrázku $p(\mathbf{x}_r)$. Funkci $\mathbf{f}$ nazýváme *funkcí korespondence (correspondence function)*.

Funkce $\mathbf{f}$ je vektorovou funkcí; lze na ní nahlížet jako na transformaci souřadného systému. Je-li $\mathbf{x}_r = (u, v)$ a $\mathbf{x}_t = (x, y)$, pak pro jednotlivé složky vektoru $\mathbf{x}_r$ platí vztah

$$\begin{aligned} u &= \mathbf{f}_x(x, y) \\ v &= \mathbf{f}_y(x, y). \end{aligned} \tag{2.2}$$



Obrázek 2.2: Blokové schema systému pro registraci obrazu.

2.3 Dílčí kroky procesu registrace obrázků

V současnosti neexistuje univerzální architektura systému pro registraci použitelná ve všech aplikacích. Při návrhu systému pro registraci je nutné brát ohled na předpokládanou oblast užití. I přes odlišnosti lze však rozložit většinu metod registrace v následující kroky:

- *Extrakce příznaků z dat* Z obrázků jsou získány množiny příznaků, které jsou v následujících fázích procesu srovnávány. Může se jednat o vyznačné body, křivky či oblasti.

- *Vyhledání korespondujících příznaků.* V referenčním a testovacím obrázku jsou hledány odpovídající si příznaky.

- *Vypočet korespondenční funkce.* Tato funkce je hledána prohledáváním *prostoru transformačních funkcí*, který určuje množinu přípustných transformací. Kritériem kvality transformačních funkcí je *účelová funkce*; registrační algoritmus usiluje o nalezení globálního maxima této funkce. S vyloučením marginálních případů, u nichž lze korespondenční funkci určit v jediném kroku přímým výpočtem, je registrace obrazu iterativní proces. V dobře definovaném registračním systému se v každé iteraci zvyšuje hodnota účelové funkce - aktuální transformační funkce se blíží funkci korespondence.

- *Transformace testovacího obrázku.* Testovací obrázek je transformován podle korespondenční funkce.

2.4 Taxonomie systémů pro registraci obrázků

Registrace obrazu je složitá úloha; každý systém pro registraci musí být navržen s ohledem na předpokládanou aplikaci a na základě apriorních znalostí dat. Této aplikaci je přizpůsobena volba každé z komponent systému. Podle volby některé z těchto komponent či účelu registrace je možné systémy pro registraci rozčlenit do kategorií; klasifikačními kritérii tedy mohou být:

- *Účel registrace*
- *Příznakový prostor algoritmu (feature space)*
- *Množina prohledávaných transformací (warp space, search space, model)*
- *Strategie prohledávání (search strategy)*
- *Účelová funkce (cost function)*

2.4.1 Účel registrace

Jedním z kritérií klasifikace systémů pro registraci obrázků jsou třídy, do kterých spadá oblast jejich aplikace. Systémy pro registraci pak dělíme na systémy pro registraci dat *pořízených z různých úhlů (multiview analysis)*, *pořízených v rozdílný čas (multitemporal analysis)*, *různými senzory (multimodal analysis)* a na *systémy umisťující model objektu do scény* [44, 22].

Data pořízena z odlišných úhlů pohledu

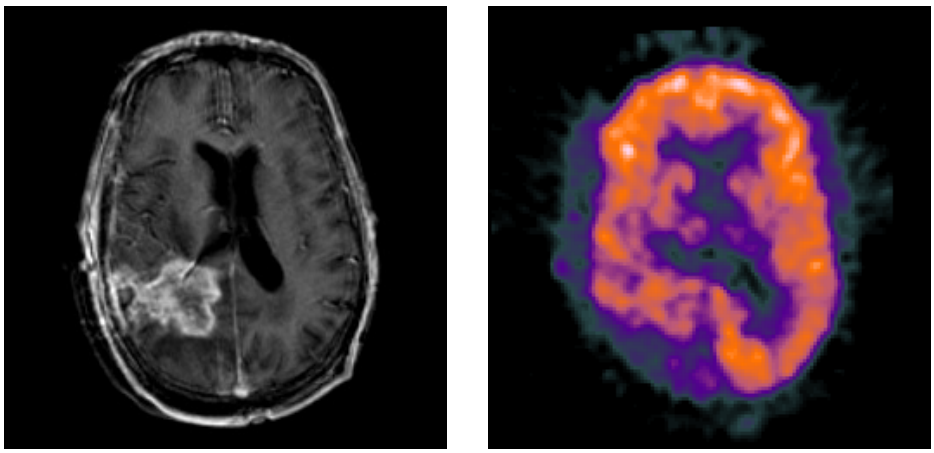
Cílem registrace dat pořízených z odlišných úhlů pohledu bývá obraz získaný fůzí dílčích obrázků či získání (úplnější) trojrozměrné reprezentace reality. Častou úlohou je extrakce trojrozměrného modelu objeku z fotografií (*„3D from stereo“*) nebo mozaikové skládání leteckých snímků [13].

Data pořízená v odlišný čas

Cílem této registrace je vyhodnotit změny snímané scény v čase. Příkladem může být lékařské zobrazování (porovnání pre- a post-iktálních SPECT snímků, monitoring růstu tkání [25]) nebo systém pro sledování pohybu.

Data odlišných modalit

Účelem registrace dat různých modalit je získat úplnější a detailnější informaci o snímaném objektu. Registrace dat různých modalit je častá u lékařského zobrazování. Příkladem je fůze dat získaných ultrazvukovým snímáním, PET, SPECT, vypočetní tomografií (CT) či magnetickou rezonancí (MRI) [25]. Příklad multimodálních dat je na Obr. 2.3.



Obrázek 2.3: Příklad dat odlišných modalit. MR (vlevo) a PET (vpravo) obraz řezu hlavou. Převzato z *The Whole Brain Atlas*, <`http://www.med.harvard.edu/AANLIB/home.html`>.

Registrace scény a modelu objektu

Účelem registrace scény a modelu objektu je lokalizovat model v obrázku či nalézt instanci modelu podobnou „průměrné“ instanci a vyhodnotit změny. Registrace scény a modelu je využíváno v průmyslu při optickém hodnocení kvality, při klasifikaci do tříd či v lékařství při porovnávání nasnímaných dat s anatomickými atlasy.

2.4.2 Příznakový prostor algoritmu

Extrakce příznaků z obrázků nahrazuje výběr význačných bodů odborníkem při ruční registraci. Pro strojovou registraci obrázků je třeba definovat takový postup získání příznaků, který minimalizuje množství příznaků beze svého protějšku v druhém registrovaném obrázku.

Algoritmy založené na význačných prvcích

Algoritmy založené na význačných prvcích (landmark-based algorithms) využívají výskytu snadno zjistitelných prvků v referenčním i testovacím obrázku. Mohou to být význačné plochy (extremální oblasti), křívky (hranice oblastí) či body (protnutí křivek, body na křivce s malým poloměrem křivosti, body nalezené Harrisovým detektorem). V obrázcích je nutno před registrací identifikovat korespondující páry příznaků; jako kritérium mohou sloužit např. jejich plošné uspořádání či hodnoty jasu sousedních pixelů. Tuto třídu algoritmů nelze použít, pokud jsou vyznačné struktury obtížně identifikovatelné nebo mezi registrovanými obrázky nekonzistentní.

Plošné algoritmy

Plošné (area-based) algoritmy z obrázku neextrahují z dat význačné struktury; příznakový prostor tvoří pixely obrázku, rozložené na pravoúhlé mřížce. Při transformaci takové mřížky v jednotlivých iteracích výpočtu korespondenční funkce však nastává situace, kdy polohy bodů transformovaného obrázku neodpovídají polohám na pravoúhlé mřížce; je nutno užít některé z interpolačních metod.

Algoritmy založené na vlastnostech integrálních transformací

Algoritmy založené na vlastnostech integrálních transformací využívají k registraci vlastností Fourierovy, waveletové a jiných integrálních transformací.

Fourierovské metody Registrace pomocí Fourierovy transformace využívá skutečnosti, že u většiny jednoduchých geometrických transformací obrázku lze snadno identifikovat jejich protějšek ve frekvenční oblasti. Tyto metody jsou robustní vůči frekvenčně závislému šumu. Lze je však užít pouze pro rigidní registraci (viz níže). Jednoduchá metoda identifikující posunutí v obrazové oblasti ze změny fáze spektra ve frekvenční oblasti byla navržena C. D. Kuglinem a D. C. Hinesem [21]

2.4.3 Množina prohledávaných transformací

Volba množiny prohledávaných transformací (modelu) je především určena apriorní znalostí o vlastnostech obrázků, které chceme systémem registrovat. Volba některého z možných modelů je kompromisem mezi množstvím hledaných parametrů a obecností možných řešení. Obecnější model také mnohdy vyžaduje sofistikovanější prohledávací strategii. Hierarchie modelů je znázorněna na Obr. 3.1.

Globální modely

Mezi globální modely patří modely založené na *rigidních transformacích* a *perspektivní transformaci*. Nejjednodušším modelem této třídy je translace, dále sem patří podobnostní, afinní a perspektivní transformace. Rigidní transformace jsou definovány jako

$$\mathbf{x}_r = \mathbb{A}(\mathbf{x}_t) + \mathbf{v}, \tag{2.3}$$

kde ortogonální matice $\mathbb{A}$ je *lineární* a vektor $\mathbf{v}$ *translační* komponenta transformace. Velmi obecným globálním modelem je perspektivní transformace. Umožňuje plně popsat deformace, které jsou kombinací posuvu, otočení, změny měřítka, zkosení a perspektivního zkreslení.

Lineární modely nejsou schopny modelovat lokální deformace. Výhodou těchto algoritmů je malý počet hledaných parametrů.

Semilokální modely

Množina *semilokálních modelů* obsahuje modely schopné popsat lokální deformace. Obvykle se jedná o globální modely s přidaným nelineárním členem. Nerigidní modely lze dále dělit podle typu nelineárního členu.

Radiální funkce Modely využívající *radiálních funkcí (radial-basis functions, RBF)* k polynomům popisujícím globální deformaci přidávají lineární kombinaci rotačně symetrických funkcí. Transformované souřadnice u, v lze zapsat jako

$$\begin{aligned} u &= \mathbf{p}_u(x,y) + \sum_{i=1}^{N} w_i g(\mathbf{x}, \mathbf{x}_i) \\ v &= \mathbf{p}_v(x,y) + \sum_{i=1}^{N} w_i g(\mathbf{x}, \mathbf{x}_i) \end{aligned} \tag{2.4}$$

kde $\mathbf{p}_u(x,y)$ a $\mathbf{p}_v(x,y)$ jsou polynomy popisující globální deformaci, $g(\mathbf{x}, \mathbf{x}_i)$ $i-$tá rotačně symetrická funkce a w_i parametry určující váhu příslušných funkcí. Označení radiální funkce vystihuje jejich důležitou vlastnost, a sice že funkční hodnota radiální funkce závisí pouze na vzdálenosti bodu z definičního oboru funkce ($\mathbf{x}$) od zvoleného charakteristického bodu funkce ($\mathbf{x}_i$). Často užívané RBF funkce jsou *thin-plate* funkce [44], u kterých nabývá nelineární člen v případě plošné registrace tvaru

$$g(\mathbf{x}, \mathbf{x}_i) = \|\mathbf{x} - \mathbf{x}_i\|^2 \ln(\|\mathbf{x} - \mathbf{x}_i\|). \tag{2.5}$$

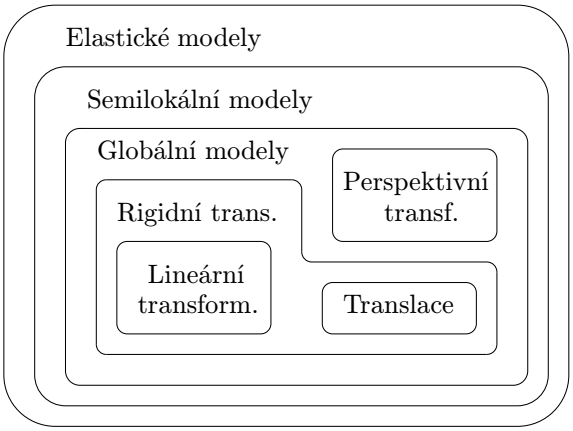
Elastické modely

Elastické modely jsou množinou transformačních funkcí, jež je natolik obecná, že je schopna reprezentovat téměř libovolnou nelineární korespondenční funkci $\mathbf{f}(\mathbf{x}_t)$. Deformace obrázku je modelována jako dvourozměrné pružné prostředí s definovanými vlastnostmi (pružnost, pevnost).

B-spliny Oblíbené elastické modely deformace jsou modely založené na *B-splinech*. B-spline je křivka složená z většího počtu dílčích polynomiálních křivek, spojených v *uzlech (knots)*. Pro B-spline řádu n je každá část polynom β^n řádu n určený polohou uzlů a $n-1$ podmínkami pro $\{1 \ldots n-1\}$-tou derivaci v uzlech. Pro výsledný B-spline $s(x)$ platí (pro jednorozměrný případ)

$$s(x) = \sum c(i)\beta^n(x-i), \tag{2.6}$$

kde $c(i)$ jsou parametry modelu (koeficienty B-splinů). Dílčí polynomiální funkce jsou definovány jako $(n+1)$-tá konvoluce dílčí polynomiální funkce řádu n s obdélníkovou funkcí



Obrázek 2.4: Hierarchie množin modelů transformačních funkcí.

β^0:

$$\beta^0(x) = \begin{cases} 1 & -\frac{1}{1} < x < \frac{1}{2} \\ \frac{1}{2} & x = \pm\frac{1}{2} \\ 0 & \text{jinde} \end{cases}$$
$$\beta^{n+1} = \beta^n * \beta^0 \tag{2.7}$$

B-spliny mají několik výhodných vlastností:

- Každý dílčí polynom je určen pouze podmínkami v několika sousedních uzlech → rychlost výpočtu
- Interpolaci lze realizovat filtrováním IIR filtrem
- n-rozměrnou filtraci lze rozložit na n jednorozměrných filtrací

2.4.4 Strategie prohledávání množiny transformačních funkcí

Přímý výpočet transformace

V určitých případech, pokud lze spočítat parametry extrému účelové funkce přímým výpočtem, jsme schopni zjistit podobu funkce korespondence v jediném kroku.

Prohledávání prostoru transformací hrubou silou

Je-li prostor transformací dostatečně malý, lze nalézt funkci korespondence prohledáváním prostoru transformací hrubou silou výpočtem účelové funkce pro všechny přípustné hodnoty parametrů modelu.

Optimalizace

Při optimalizačním přístupu hledáme maximum účelové funkce *optimalizačními metodami*. Hledáme takovou transformační funkci $\mathbf{f}$, pro kterou účelová funkce $C(\mathbf{f})$ dosahuje svého globálního maxima $C_m = \arg\max_{\mathbf{f}} C(\mathbf{f})$. Patří sem *metody 1. řádu* a *metody 2. řádu*.

Metody 1. řádu Často užívanými optimalizačními metodami jsou *metody 1. řádu (gradientní metody)*. Tyto metody hledají maximum účelové pohybem pracovního bodu proti směru jejího největšího spádu. Je-li $\mathbf{p} = (p_1, \ldots, p_n)$ vektor n parametrů transformačního modelu, pak je gradient účelové funkce $\nabla C(\mathbf{p})$ definován

$$\nabla C(\mathbf{p}) = \left(\frac{\partial C}{\partial p_1}, \ldots, \frac{\partial C}{\partial p_1} \right). \tag{2.8}$$

Soustava parciálních diferenciálních rovnic

Lze sestavit soustavu parciálních diferenciálních rovnic tak, aby korespondenční funkce byla jejím ustáleným řešením. Soustavy parciálních diferenciálních rovnic lze řešit metodou konečných prvků, metodou konečných diferencí a jinými metodami.

Jiné metody

K hledání korespondenční funkce lze dále užít: dynamické programování, genetické programování, prohledávání množiny transformačních funkcí hrubou silou s využitím heuristik a další metody.

2.4.5 Typ účelové funkce

Účelová funkce je volena tak, aby postihla optimalitu nalezené transformace z hlediska podobnosti transformovaného referenčního a testovacího obrázku a realističnosti transformace; hledané korespondenční funkci odpovídá globální maximum účelové funkce. Účelová funkce je funkcí parametrů zvoleného transformačního modelu; je definována jako

$$C(\mathbf{f}, p, q) = S(\mathbf{f}, p, q) + R(\mathbf{f}, p, q) \tag{2.9}$$

kde $S(\mathbf{f}, p, q)$ je *míra podobnosti* a $R(\mathbf{f}, p, q)$ *regularizační člen* účelové funkce. Míra podobnosti popisuje, nakolik si jsou dva obrázky podobné. Volba míry podobnosti závisí na apriorních znalostech o registrovaných obrázcích. Při registraci algoritmy založenými na význačných prvcích může být mírou podobnosti například *součet vzdáleností korespondujících význačných prvků*. Oblíbenými mírami podobnosti pro plošné metody jsou *součet čtverců rozdílů (SSD)*,

korelační koeficient (C) a *vzájemná informace (I)*. Regularizační člen účelové funkce znevýhodňuje nerealistické a nepravděpodobné transformace. Uvažováním tohoto členu dosáhneme dobře podmíněného a stabilního algoritmu, můžeme však také do optimalizace zakomponovat apriorní znalost o datech [29].

Součet čtverců rozdílů

Negativní součet čtverců rozdílů intenzit pixelů mezi obrázky je jednoduchá, rychlá a robustní podobnostní míra. Nelze ji však užít při registraci multimodálních dat; při užití této míry podobnosti předpokládáme, že senzor reaguje stejným způsobem na testovací i referenční obrázek a že intenzitní hodnoty odpovídajících si míst se liší pouze šumem. Pro dvojici obrázků $p(\mathbf{x}_r)$ a $q(\mathbf{f}(\mathbf{x}_t))$ je definována jako

$$SSD(\mathbf{f}, p, q) = -\sum_{i,j} \Big(q(\mathbf{f}(i,j)) - p(i,j) \Big)^2. \tag{2.10}$$

Korelační koeficient

Korelační koeficient popisuje míru lineární závislosti mezi dvěma náhodnými ději. Platí pro něj

$$C(\mathbf{f}, p, q) = \frac{\sum\limits_{i,j} \Big(q(\mathbf{f}(i,j)) - \overline{q} \Big) \Big(p(i,j) - \overline{p} \Big)}{\sqrt{\sum\limits_{i,j} \Big(q(\mathbf{f}(i,j)) - \overline{q} \Big)^2 \sum\limits_{i,j} \Big(p(i,j) - \overline{p} \Big)^2}} \tag{2.11}$$

kde $\overline{q}$ a $\overline{p}$ jsou průměrné hodnoty intenzity obrázků $q(\mathbf{f}(\mathbf{x}_t))$ a $p(\mathbf{x}_r)$. Korelační koeficient nabývá hodnot z intervalu $\langle -1; 1 \rangle$. Pro totožné obrázky je roven 1, pro navzájem inverzní obrázky -1. Pokud jsou hodnoty intenzit pixelů realizacemi nezávislých náhodných procesů, je očekávání jeho střední hodnoty rovno 0.

Vzájemná informace

Vzájemná informace je nejrozšířenější mírou podobnosti užívanou při registraci multimodálních obrázků; bývá proto často užívána pro registraci v lékařském zobrazování. Vzájemná informace je mírou statistické závislosti dvou náhodných dějů; pro dvě diskrétní náhodné veličiny X a Y je definována

$$I(X,Y) = \sum_{x_i \in X} \sum_{y_j \in Y} P(x_i, y_j) \log \frac{P(x_i, y_j)}{P(x_i) P(y_j)} \tag{2.12}$$

kde $P(x_i, y_j)$ je sdružená hustota pravděpodobnosti a $P(x_i)$ a $P(y_j)$ marginální hustoty pravděpodobností hodnot, kterých nabývají příslušné veličiny. Vzájemnou informaci lze vyjádřit

pomocí entropií $H(X)$, $H(Y)$ a sdružené entropie $H(X,Y)$ jako

$$\begin{aligned} I(X,Y) &= H(X) - H(X|Y) \\ &= H(Y) - H(X) + H(X,Y). \end{aligned} \tag{2.13}$$

Tohoto vztahu se obvykle užívá pro výpočet vzájemné informace na základě odhadů entropií. Registrované obrázky jsou v tomto smyslu realizacemi náhodných procesů X a Y.

Jedna z prvních prací navrhující užití vzájemné informace pro registraci je [42]. Autoři demonstrovali navržený postup registrací MRI obrázků a modelu objektu ve scéně.

Kapitola 3

Teorie informace, entropie a její odhad

3.1 Kořeny teorie informace

V první polovině 20. století již byly k dispozici poměrně pokročilé metody umožňující přenos informace (bezdrátová telegrafie, televizní přenos, frekvenční modulace, rozprozstřené spektrum, PCM kód, vokodér aj.). Většina však byla výsledkem inženýrského úsilí a málo z těchto metod se opíralo o rozsáhlejší teoretický základ, ačkoliv mnohé implicitně užívaly některé poznatky teorie informace jako důsledek intuice (např. kratší reprezentace frekventovaných znaků anglického jazyka v Morseově kódu).

Za zakladatele teorie informace je považován Claude Shannon, jeho práce však přímo navazuje na poznatky H. Nyquista a R. Hartleye z dvacátých let 20. století. Nyquist [30] uvádí dva faktory ovlivňující rychlost přenosu telegrafních informací *(intelligence rate)*:

1. Fyzická realizace telegrafního systému (konstrukce terminálu, parametry kabelu, úroveň signálu), určující horní mez přenášených frekvencí. Tento faktor byl všeobecně známý a hledala se nová řešení pro vylepšování přenosových charakteristik systémů.

2. Reprezentace zprávy v signálu. Telegrafisté tradičně užívali Morseova kódu, jehož definice zhruba respektuje frekvenci znaků anglického jazyka.

Nyquist uvádí vztah pro horní mez informace přenositelné telegrafem za časovou jednotku; platí

$$W = K \ln m \tag{3.1}$$

kde W je množství přenesené informace telegrafu, m množství proudových úrovní telegrafu a K konstanta. Také zmiňuje možnost zvýšení rychlosti přenosu informací při reprezentaci zprávy „optimálním kódem“.

R. Hartley ve své práci [15] rozvíjí podobné myšlenky jako H. Nyquist. Hartley zdůrazňuje, že informační kapacita vedení závisí pouze na schopnosti příjemce rozlišit v každém kroku

mezi možnými sekvenceni symbolů zvolených na opačném konci vedení a nikoliv na významu této informace. Hartley hledal veličinu, která bude pro abecedy velikostí s_1, s_2 a pro délky zpráv n_1, n_2 stejná, pokud $s_1^{n_1} = s_2^{n_2}$, dále která bude stoupat s počtem možných zpráv a která bude nulová pro minimální abecedu. To jej vedlo k závěru, že míra informace musí mít logaritmický tvar. Zobecňuje vztah (3.1); pro zprávu o n symbolech nad abecedou velikosti s pak pro množství přenesené informace H platí

$$H = \ln s^n. \tag{3.2}$$

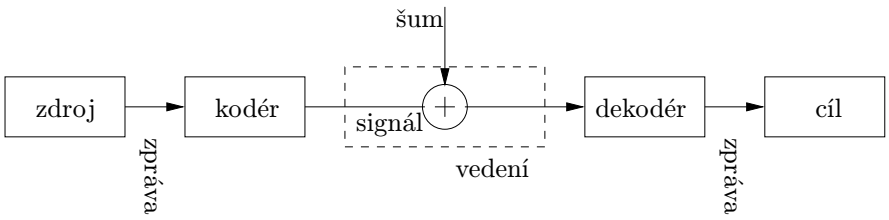
Veličina H je nazývána Hartleyova míra informace; vyjadřuje množství získané informace při přijmu zprávy délky n. Množství informace je úměrné délce zprávy a její logaritmus je úměrný velikosti abecedy. Nevýhodou Hartleyovy míry je, že předpokládá stejné relativní frekvence všech znaků abecedy, což nebývá zaručeno.

3.2 Některé poznatky teorie informace

Shannon se ve své práci úspěšně pokusil sestavit teorii umožňující pochopení a modelování telekomunikačních systémů. Rozpracoval její základy v prubehu 40. let a své výsledky zveřejnil v průlomovém článku *A Mathematical Theory of Communication* zveřejněném v periodiku Bellových laboratoří roku 1948 [36]. Shannon definuje model systémů pro přenos informace sestávající z následujících součástí (Obr. 3.1):

- *Zdroj informace* generuje zprávu, určenou k přenosu. Může se jednat o záznam hlasu účastníka telefonního hovoru či posloupnost písmen psané zprávy.
- *Kodér* ze zprávy generuje signál, který lze přenést zvoleným médiem. Akustický tlak je převeden na velikost proudu, psaná zpráva je reprezentována posloupností teček a čárek.
- *Vedení* je médium sloužící k přenosu signálu generovaného kodérem.
- *Dekodér* ze signálu rekonstruuje původní zprávu. Dekodér provádí inverzní operaci k činnosti kodéru.
- *Cíl* je entita které je zpráva určena.

Pokračování Shannonova článku lze rozdělit na dvě části; v první se zabývá uchopením pojmu informace a modelováním zdrojů informace. Zavádí pojem informační entropie a zabývá se modelováním přenosu informace po nezašuměném kanálu. Ve druhé části se věnuje přenosu informace zašuměným vedením a důsledky nedokonalosti přenosového kanálu.



Obrázek 3.1: Schema systému pro přenos informace.

3.2.1 Zdroj informace

Zdroj informace je matematický model objektu produkujícího posloupnost symbolů - *výstupů*. Množina všech možných výstupů se nazývá *abeceda*. Výstup zdroje informace je náhodný - ze znalosti výstupů do určitého okamžiku v čase nelze jednoznačně zjistit jejich budoucí průběh. Výstup zdroje informace lze modelovat pomocí pravděpodobnostního modelu jako náhodnou veličinu.

Pravděpodobnostní model

Pravděpodobnostní model je definován množinou $X = \{x_1 \dots x_n\}$ navzájem se vylučujících *elementárních jevů* a funkcí

$$P(A) : A \to \langle 0; 1 \rangle \tag{3.3}$$

kde A (*jev*) je libovolně zvolená podmnožina množiny elementárních jevů $A \subset X$. Funkce P přiřazuje množině A reálné číslo z intervalu $\langle 0; 1 \rangle$ vyjadřující relativní četnost jevu A. Funkce P je *pravděpodobnostní míra (pravděpodobnost)* jevu A. Pro pravděpodobnostní míru platí normalizační podmínky

$$\sum_{x_i \in X} P(x_i) = 1,$$
$$P(A) \geq 0 \text{ pro všechna } A \subset X.$$

Náhodnou veličinou pak je zobrazení

$$N : X \to \mathbb{R} \tag{3.4}$$

přiřazující každému elementárnímu jevu $x_i \in X$ reálné číslo.

Sdružená a podmíněná pravděpodobnost jevů

V přirozeném jazyce se některé skupiny znaků vyskytují s odlišnou frekvencí od jiných skupin a znalost předchozího znaku nám může něco říci o znaku následujícím - například v češtině po samohlásce obvykle následuje souhláska. V obrázku bývají podobně jasné pixely často blizko

u sebe. Vzájemné vztahy více náhodných veličin nám umožňuje popsat aparát podmíněné a sdružené pravděpodobnosti.

Sdružená pravděpodobnost jevů $A \subset X = \{a_1 \dots a_n\}$ a $B \subset Y = \{b_1 \dots b_m\}$ $P(A, B)$ vyjadřuje relativní četnost pokusů, jejichž výsledkem jsou jevy A a B. Sdružená pravděpodobnost v sobě obsahuje veškerou informaci o dílčích pravděpodobnostech jevů A a B. Tyto *marginální pravděpodobnosti* lze spočítat

$$\begin{aligned} P(A) &= \sum_{i=1}^{n} P(A, b_i), \\ P(B) &= \sum_{i=1}^{m} P(B, a_i). \end{aligned} \tag{3.5}$$

Pokud jsou jevy A a B vzájemně nezávislé, platí

$$P(A, B) = P(A)P(B). \tag{3.6}$$

Podmíněná pravděpodobnost $P(A|B)$ je definovaná jako pravděpodobnost jevu A pokud s jistotou víme že jev B nastal. Platí

$$P(A|B) = \frac{P(A, B)}{P(B)}. \tag{3.7}$$

3.2.2 Entropie

Entropie je jednou z číselných charakteristik náhodných veličin. Entropie je funkcí pravděpodobností elementárních jevů $H(P_1, P_2 \dots P_n)$ splňující následující podmínky:

1. Je spojitou funkcí pravděpodobností na intervalu $\langle 0; 1 \rangle$
2. Záměna libovolných dvou argumentů funkce H nezmění její hodnotu
3. Pokud lze elementární jev x_1 s pravděpodobností P_1 rozložit na dva dílčí elementární jevy x_1', x_1'' s pravděpodobnostmi výskytu P_1' a P_1'' (*zjemnění množiny elementárních jevů*), pak musí platit

$$H(P_1', P_1'', P_2 \dots P_n) = H(P_1, P_2 \dots P_n) + P_1 H\Big(\frac{P_1'}{P_1}, \frac{P_1''}{P_1}\Big). \tag{3.8}$$

Lze ukázat, že výše zmíněným podmínkám vyhovuje pouze funkce tvaru $-K \sum_{i=1}^{n} P(x_i) \ln P(x_i)$ kde K je libovolná nezáporná konstanta. Entropie na množině jevů $X = \{x_1 \dots x_n\}$ je tedy definována

$$H(X) = -\sum_{x_i \in X} P(x_i) \ln P(x_i). \tag{3.9}$$

Entropie má několik vlastností, z nichž některé odpovídají intuitivním požadavkům kladeným na informační míru:

1. $H = 0$ pokud P_i všech elementárních jevů mimo jednoho jsou rovny nule.

2. Pro n elementárních jevů H nabývá maxima pokud $P_i = \frac{1}{n}$ pro $i = 1 \dots n$.

3. *Sdružená entropie* $H(X, Y)$ množin elementárních jevů $X = \{x_1 \dots x_n\}$ a $Y = \{y_1 \dots y_m\}$ je entropie na kartézském součinu množin (X, Y), tedy

$$H(X, Y) = -\sum_{x_i \in X} \sum_{y_j \in Y} P(x_i, y_j) \ln P(x_i, y_j), \tag{3.10}$$

   kde $P(x_i, y_j)$ je sdružená pravděpodobnost jevů x_i a y_j. Marginální entropie podle množin X a Y získáme

$$\begin{aligned} H(X) &= -\sum_{x_i \in X} \left(\sum_{y_j \in Y} P(x_i, y_j) \right) \ln \sum_{y_j \in Y} P(x_i, y_j), \\ H(Y) &= -\sum_{y_j \in Y} \left(\sum_{x_i \in X} P(x_i, y_j) \right) \ln \sum_{x_i \in X} P(x_i, y_j). \end{aligned} \tag{3.11}$$

   Porovnáním rovnic (3.10) a (3.11) získáme vztah

$$H(X, Y) \leq H(X) + H(Y) \tag{3.12}$$

   s rovností, pokud platí $P(x_i, y_j) = P(x_i)P(y_j)$ pro $i = 1 \dots n$, $j = 1 \dots m$.

4. Entropii jevu X za předpokladu že nastal jev y_j lze vyjádřit

$$H(X|y_j) = -\sum_{x_i \in X} P(x_i|y_i) \ln P(x_i|y_i). \tag{3.13}$$

   Pak střední hodnota entropií $H(X|y_j)$ pro $j = 1 \dots m$

$$\begin{aligned} H(X|Y) &= -\sum_{y_j \in Y} P(y_j) H(X|y_j) \\ &= -\sum_{x_i \in X} \sum_{y_j \in Y} P(x_i, y_j) \ln \frac{P(x_i, y_j)}{P(y_j)} \end{aligned} \tag{3.14}$$

   se nazývá *podmíněná entropie* množiny X na množině Y. Z porovnání (3.14) a (3.9) vyplývá, že podmíněnou entropii lze zapsat jako

$$H(X|Y) = H(X, Y) - H(Y). \tag{3.15}$$

5. Z porovnání vztahů (3.11), (3.14) a z principu symetrie vyplývá

$$\begin{aligned} H(X) &\geq H(X|Y), \\ H(Y) &\geq H(Y|X) \end{aligned} \tag{3.16}$$

   s rovností pokud jsou X a Y nezávislé.

Zobecněním vztahu (3.9) na spojitá rozdělení je možno popsat *diferenciální entropii* spojité náhodné veličiny. Diferenciální entropie náhodné veličiny X je dána

$$H_c(X) = \int_{-\infty}^{\infty} f(x) \ln f(x) \mathrm{d}x \tag{3.17}$$

kde $f(x)$ je hustota pravděpodobnosti náhodné veličiny X. Narozdíl od entropie diskrétního rozdělení může diferenciální entropie nabývat záporných hodnot. Ve spojité verzi je entropie relativní mírou neurčitosti vzhledem ke zvolenému systému souřadnic. Pro spojitou entropii platí obdobné vztahy jako pro entropii diskrétní. V praxi odhadujeme spojitou entropii hustoty pravděpodobnosti pomocí diskrétní entropie, více viz kap. 4.

Interpretace

Informační entropii zavedl C. Shannon za účelem popisu chování zdroje informace. Entropie je mírou neurčitosti před přijetím symbolu zprávy; čím více neurčitosti připadá zdroji, tím těžší je symboly předvídat, tj. tím více informace získáme po obdržení symbolu. Entropie je mírou šíře pravděpodobnostního rozdělení; u rozdělení s několika úzkými dominujícími peaky je nižší než pro široká rozdělení.

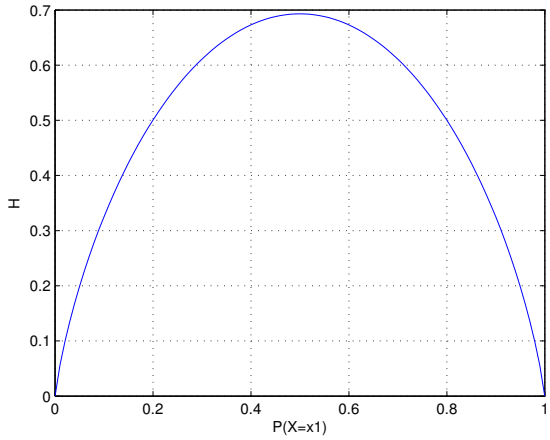
Pro zdroj s degenerovaným pravděpodobnostním rozdělením symbolů na výstupu je entropie nulová (bod 1). Neexistuje neurčitost co se následujících symbolů týče, tudíž jejich přijímáním nezískáváme žádnou informaci. Pro zdroje s rovnoměrným rozdělením pravděpodobnosti symbolů (bod 2) je neurčitost maximální; žádný symbol není pravděpodobnější než jiné. Závislost hodnoty informační entropie na hodnotách pravděpodobností elementárních jevů lze ilustrovat na Bernoulliho pokusu. Bernoulliho pokus je náhodná veličina, která může nabývat pouze dvou hodnot x_1 a x_2 s pravděpodobnostmi $P(X = x_1)$ a $P(X = x_2) = 1 - P(X = x_1)$. Průběh hodnoty informační entropie Bernoulliho pokusu v závislosti na $P(X = x_1)$ je znázorněn na Obr. 3.2.

Rovnice (3.12) tvrdí, že entropie složeného jevu není větší než součet entropií dílčích jevů; dílčí jev v mezním případě neobsahuje žádnou informaci o druhém dílčím jevu. Názornějším vyjádřením téže myšlenky jsou rovnice (3.16). Pokud nastal jev X, nejistota o jevu Y nevzroste; informace není záporná.

3.2.3 Vzájemná informace

Nechť $X = \{x_1 \dots x_n\}, Y = \{y_1 \dots y_m\}$ jsou množiny elementárních jevů. *Vzájemná informace (mutual information)* množin X a Y je definována jako

$$I(X, Y) = H(X) - H(X|Y). \tag{3.18}$$



Obrázek 3.2: Entropie Bernoulliho pokusu v závislosti na pravděpodobnosti jednoho z výsledků. Entropie nabývá maxima pro $P(X = x_1) = P(X = x_2)$, tj. pro rovnoměrné rozdělení pravděpodobností.

Po dosazení za $H(Y|X)$, resp. $H(X)$, $H(Y)$ a $H(X, Y)$ lze odvodit analogické vztahy pro vzájemnou informaci

$$I(X, Y) = H(X) + H(Y) - H(X, Y) \tag{3.19}$$

$$= \sum_{x_i \in X} \sum_{y_j \in Y} P(x_i, y_i) \ln \frac{P(x_i, y_j)}{P(x_i)P(y_j)}. \tag{3.20}$$

Vzájemná informace je mírou vzájemné závislosti náhodných veličin; vyjadřuje o kolik se v průměru sníží nejistota X pokud známe Y, tj. množství informace, které obsahuje Y o X. Vzájemná informace je, stejně jako entropie, symetrická vzhledem k argumentům. Vztah vzájemné informace k entropii je znázorněn na Obr. 3.3.

Vzájemná informace popisuje vzdálenost skutečného sdruženého rozložení pravděpodobnosti X, Y a sdruženého rozložení pravděpodobnosti, pokud by byly X a Y nezávislé. Pro vzájemnou informaci platí

$$I(X, Y) = 0 \text{ pokud } X \text{ a } Y \text{ jsou nezávislé} \tag{3.21}$$

a

$$I(X, Y) = H(X) = H(Y) \text{ pokud } X = Y. \tag{3.22}$$

Vzájemná informace a registrace obrázků

Pro registraci obrázků pomocí vzájemné informace je klíčová skutečnost, že body ve sdruženém příznakovém prostoru mění svojí polohu v závislosti na míře „registrovanosti“ obrázků.

Ze vztahu (3.18) vyplývá, že maximalizace $I(X,Y)$ odpovídá minimalizaci $H(X,Y)$. Pokud jsou obrázky dobře registrovány, korespondující oblasti se zhruba stejnou intenzitou vytvoří ve sdruženém příznakovém prostoru shluky bodů. Pak znalost hodnoty intenzity určitého pixelu z jednoho obrázku nám s pravděpodobností větší než u neregistrovaných obrázků poskytne informaci o intenzitě ve druhém obrázku, $H(X,Y)$ se zmenší a $I(X,Y)$ vzroste. U neregistrovaných obrázků odpovídá určité hodnotě intenzity v jednom obrázku více intenzit v obrázku druhém, sdružený příznakov prostor je rozptýlenější, nejistota a tudíž hodnota $H(X,Y)$ je větší a $I(X,Y)$ nižší (Obr. 3.4). Pro registraci však nelze jako kriterium použít pouze hodnotu sdružené entropie $H(X,Y)$; Vzhledem k faktu, že sdružený příznakový prostor je konstruován z překrývajících se částí obrázku, mohou nízké hodnoty $H(X,Y)$ nastat v případech, kdy se obrázky překrývají např. pouze okraji s homogenními hodnotami jasu a ve sdruženém příznakovém prostoru je pak pouze jeden peak odpovídající těmto hodnotám. Pro vzájemnou informaci $I(X,Y)$ toto omezení neplatí, neboť ve výše zmíněném nežádoucím případě budou také nízké hodnoty $H(X)$ a $H(Y)$ odpovídající stejným homogenním oblastem a $I(X,Y)$ bude ve výsledku malé.

3.3 Odhady entropie

V praxi nemůžeme přímo počítat entropii pravděpodobnostního rozdělení z důvodu neznalosti skutečné hustoty pravděpodobnosti generující data. V případech, kdy nemůžeme spočítat skutečnou hodnotu entropie analyticky mohou pomoci některé z metod jejího odhadu na základě vzorků z $f(x)$.

Mějme množinu n vzorků $\mathcal{X}_n = \{\mathbf{x}_1 \dots \mathbf{x}_n\}$, $\mathbf{x}_i \in \mathbb{R}^d$ náhodné proměnné s d-rozměrnou hustotou pravděpodobnosti $f(\mathbf{x})$. *Odhad entropie* je funkce $\hat{H}(\mathcal{X}_n)$ taková, že $|\hat{H}(\mathcal{X}_n) - H(f(\mathbf{x}))|$ je malé.

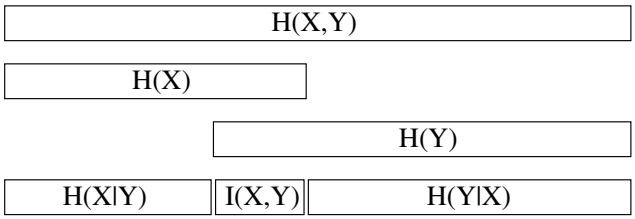
3.3.1 Plug-in odhad

Plug-in odhad je takový odhad, který nejprve odhaduje $f(\mathbf{x})$ na základě vzorků $\mathbf{x}_1 \dots \mathbf{x}_n$, tj. $\hat{H}(\hat{f}(\mathbf{x}_1 \dots \mathbf{x}_n))$, kde $\hat{f}(\mathbf{x}_1 \dots \mathbf{x}_n)$ je histogramový nebo jádrový odhad hustoty $f(\mathbf{x})$.

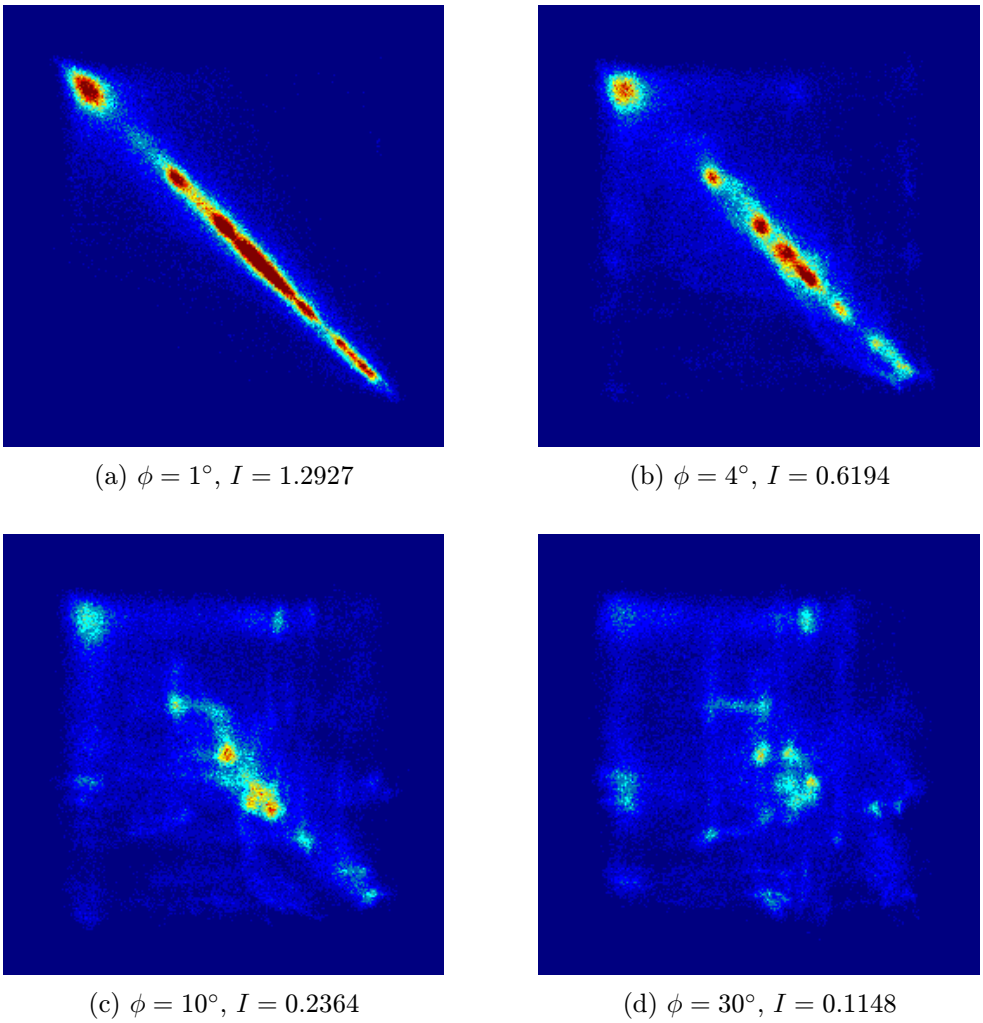
- *Integrální odhad* entropie počítá hodnotu $\hat{H}$ přímo podle definice 3.9, tj.

$$\hat{H} = -\int \hat{f}(\mathbf{x}) \ln \hat{f}(\mathbf{x}) \mathrm{d}\mathbf{x}, \tag{3.23}$$

  kde $\hat{f}(\mathbf{x})$ je odhad hustoty pravděpodobnosti $f(\mathbf{x})$. Nevýhodou integrálního estimátoru je nutnost numerické integrace, pokud je $\hat{f}(\mathbf{x})$ jádrový odhad hustoty $f(\mathbf{x})$ [4].



Obrázek 3.3: Znázornění vztahu sdružené entropie $H(X,Y)$, marginálních entropií $H(X)$ a $H(Y)$, podmíněných entropií $H(X|Y)$ a $H(Y|X)$ a vzájemné informace $I(X,Y)$ dvou náhodných veličin X a Y.



(a) $\phi = 1°$, $I = 1.2927$

(b) $\phi = 4°$, $I = 0.6194$

(c) $\phi = 10°$, $I = 0.2364$

(d) $\phi = 30°$, $I = 0.1148$

Obrázek 3.4: Sdružený příznakový prostor a hodnota $I(X,Y)$ obrázku a téhož obrázku natočeného o úhel ϕ. Protože se jedná o týž obrázek, jsou příznaky koncentrovány podél diagonály.

- *Odhad pomocí střední hodnoty logaritmu pravděpodobnosti* nabývá podoby

$$\hat{H} = -\frac{1}{n}\sum_{i=1}^{n} \ln \hat{f}(\mathbf{x}_i) \tag{3.24}$$

kde $\hat{f}(\mathbf{x})$ je odhad hustoty $f(\mathbf{x})$. Tento odhad entropie byl navržen v [1].

- *Odhad dělící vzorky (splitting-data estimate)* [14] dělí vzorky $\mathcal{X}_n = \{\mathbf{x}_1 \dots \mathbf{x}_n\}$ do dvou skupin $\mathcal{X}_k{}' = \{\mathbf{x}'_1 \dots \mathbf{x}'_k\}$, $\mathcal{X}_l{}'' = \{\mathbf{x}''_1 \dots \mathbf{x}''_l\}$ tak, že $k+l = n$. Z první podskupiny vzorků $\mathcal{X}_k{}'$ je sestrojen jádrový odhad $\hat{f}_k$ a ze druhé podskupiny $\mathcal{X}_l{}''$ posléze vypočítán odhad entropie $\hat{H}$, tj.

$$\hat{H} = -\frac{1}{l}\sum_{i=1}^{l} \ln \hat{f}_k(\mathbf{x}''_i). \tag{3.25}$$

- *Odhad s vynecháním (cross-validation estimate)* podle [18] je

$$\hat{H} = -\frac{1}{n}\sum_{i=1}^{n} \delta_{\mathbf{x}_i \in \mathcal{X}_n - \{\mathbf{x}_i\}} \ln \hat{f}_i(\mathbf{x}_i), \tag{3.26}$$

kde $\hat{f}_i(\mathbf{x})$ je odhad hustoty $f(\mathbf{x})$ získaný z $\mathcal{X}_n$ s vynecháním vzorku $\mathbf{x}_i$ a $\delta_{\mathbf{x}_i \in \mathcal{X}_n - \{\mathbf{x}_i\}}$ charakteristická funkce podmínky $\mathbf{x}_i \in \mathcal{X}_n - \{\mathbf{x}_i\}$.

3.3.2 Odhad založený na nejbližších sousedech

Kozačenkův-Leoněnkův odhad entropie [19] nevyžaduje odhad hustoty $f(\mathbf{x})$ a je použitelný pro libovolné d. Je definován jako

$$\hat{H} = \frac{d}{n}\sum_{i=1}^{n} \ln \rho_i + \ln \frac{(n-1)\pi^{\frac{d}{2}}}{\Gamma(1+\frac{d}{2})} + \gamma, \tag{3.27}$$

kde γ je Eulerova-Mascheroniho konstanta ($\gamma \approx 0.57721$), $\Gamma(x)$ je funkce Gamma a ρ_i vzdálenost bodu $\mathbf{x}_i$ k nejbližšímu sousedu. Pro soubor vzorků $\mathbf{x}_1 \dots \mathbf{x}_n$, kde alespoň pro dvě různá $\mathbf{x}_i$, $\mathbf{x}_j$ platí $\mathbf{x}_i = \mathbf{x}_j$, lze použít pro $\hat{H}$

$$\hat{H} = \frac{d}{n}\sum_{i=1}^{n} \ln \max(\rho_i, \varepsilon) + \ln \frac{(n-1)\pi^{\frac{d}{2}}}{\Gamma(1+\frac{d}{2})} + \gamma, \tag{3.28}$$

kde ε je konstanta a funkce $\max(a, b)$ vrací větší z čísel a, b.

3.3.3 Odhad založený na objemu buněk Voronoi diagramu

Mějme množinu vzorků $\mathcal{X}_n = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$, $\mathbf{x}_i \in \mathbb{R}^d$. *Voronoi diagram* na množině $\mathcal{X}_n$ získáme přiřazením každého bodu v $\mathbb{R}^d$ bodu z $\mathcal{X}_n$ s nejmenší euklidovskou vzdáleností. Oblasti

v prostoru přiřazené k jednomu bodu z $\mathcal{X}_n$ se nazývají *Voronoi buňky*. Odhad entropie $\hat{H}$ pravděpodobnostního rozložení $p(\mathbf{x})$ generujícího $\mathcal{X}_n$ je pak [26]

$$\hat{H} = \frac{1}{N_{fin}} \sum_{i=1}^{n} \delta_{V_i \neq \infty} \ln(nV_i), \tag{3.29}$$

kde V_i značí objem Voronoi buňky odpovídající bodu $\mathbf{x}_i$ a $\delta_{V_i \neq \infty} = 1$ pokud je V_i konečné velikosti, jinak $\delta_{V_i \neq \infty} = 0$ a N_{fin} je počet Voronoi buněk konečné velikosti. Entropii lze odhadovat i pomocí Delaunayovy triangulace, duálního problému ke konstrukci Voronoi diagramu [26].

3.4 α-entropie

α-entropie (též Rényiho entropie) je generalizací Shannonovy entropie [32]. α-entropie řádu α je pro náhodnou veličinu s hustotou pravděpodobnosti $p(x)$ definována

$$H_\alpha = \frac{1}{1-\alpha} \ln \left(\int_{-\infty}^{\infty} p(x)^\alpha \right), \tag{3.30}$$

$\alpha \geq 0$. Platí, že pro $\alpha \to 1$ α-entropie konverguje ke Shannonově entropii.

Nechť $\mathcal{X}_n = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ je soubor n d-rozměrných nezávislých vzorků. Pak můžeme odhadnout α-entropii pravděpodobnostního rozložení $f(x)$ generujícího $\mathcal{X}_n$ pomocí *délky některého z minimálních grafů* nad vzorky z $\mathcal{X}_n$. Minimálním grafem může být Steinerův strom, minimální kostra grafu (minimum spanning tree, MST), graf řešící problém obchodního cestujícího či graf nejbližších sousedů [27]. Délka minimálního grafu je definována

$$L(\mathcal{X}_n) = \sum_{e \in E} e^\gamma(\mathcal{X}_n), \tag{3.31}$$

kde E je množina hran patřících k minimálnímu grafu nad množinou bodů z $\mathcal{X}_n$, e je Euklidovská délka hrany z E a γ je konstanta, $0 < \gamma < d$. Odhad α-entropie je pak [16]

$$\hat{H}_\alpha(\mathcal{X}_n) = \frac{1}{1-\alpha} \left(\ln \frac{L(\mathcal{X}_n)}{n^\alpha} - K \right), \tag{3.32}$$

K je konstanta závislá na typu použitého minimálního grafu a

$$\alpha = \frac{d-\gamma}{d}. \tag{3.33}$$

Kapitola 4

Odhad hustoty pravděpodobnosti

Mějme soubor nezávislych realizací náhodné proměnné $\mathcal{X}_n = \{\mathbf{x}_1 \dots \mathbf{x}_n\}$, $\mathbf{x}_i \in \mathbb{R}^d$ s neznámou hustotou pravděpodobnosti $f(\mathbf{x})$. Odhadem hustoty pravděpodobnosti pak nazýváme usuzování z množiny $\mathcal{X}_n$ na podobu funkce $f(\mathbf{x})$. Odhady hustoty pravděpodobnosti můžeme rozdělit do dvou skupin:

- *Neparametrické odhady* poskytují metodologii aproximace velké množiny neznámých hustot pravděpodobnosti ze vzorků. Oproti parametrickým odhadům jsou však méně přesné.
- *Parametrické odhady* předpokládají znalost analytické podoby hustoty pravděpodobnosti. Parametrický odhad pak spočívá v hledání neznámých parametrů tohoto modelu. Za předpokladu správné volby modelu jsou přesnější než neparametrické estimátory.

4.1 Histogramování

Nechť $\mathcal{X}_n = \{x_1 \dots x_n\}$, $x_i \in \mathbb{R}$ je soubor vzorků pocházejících z neznámé hustoty pravděpodobnosti $f(x)$. Pak *histogram* přiřazuje každému prvku z $\mathcal{X}_n$ některý z k navzájem disjunktních *binů* $B_1 \dots B_k$ na základě hodnoty prvku x_i. Pro šířku binu h jsou do i-tého binu zahrnuty prvky z intervalu $\langle o+(i-1)h, o+ih)$, kde $o = \min(\mathcal{X}_n)$; histogram rozděluje prostor jevů na přihrádky o šířce h. Pro histogram platí

$$n = \sum_{i=1}^{k} m_i, \tag{4.1}$$

kde m_i je počet vzorků v i-tém binu. Pravděpodobnost, že náhodný vzorek připadne i-tému binu lze zapsat jako

$$P_i = \int_{o+(i-1)h}^{o+ih} f(x)\mathrm{d}x. \tag{4.2}$$

Pravděpodobnost binu B_i lze pak zpětně odhadnout podle četností jako

$$P_i = \frac{m_i}{n}. \tag{4.3}$$

Odhad integrálu hustoty pravděpodobnosti $\hat{f}(x)$ je pak pro interval hodnot $\langle o+(i-1)h, o+ih)$ roven P_i.

Odhad hustoty pravděpodobnosti získaný za pomocí histogramování lze považovat za neparametrický odhad hustoty pravděpodobnosti, neboť s jeho pomocí lze modelovat pro $n \to \infty$ téměř libovolnou hustotu $f(x)$. Histogram lze zobecnit pro větší počet dimenzí. Biny jsou pak určeny svými souřadnicemi v každé dimenzi. Je-li $k_1, k_2 \ldots k_n$ počet binů v $1, 2 \ldots n$-té dimenzi, pak pro celkový počet binů N platí

$$N = k_1 k_2 \ldots k_n. \tag{4.4}$$

U vícerozměrných histogramů se manifestuje nepříjemný jev známý jako prokletí dimenzionality *(curse of dimensionality)*. Označení popisuje jev exponenciálního narůstání velikosti prostoru s lineárním nárustem dimenzí.

Šířku binů h je vhodné volit na základě apriorních znalostí o hustotě pravděpodobnosti analyzovaných dat. V současnosti existuje pro tuto volbu několik empirických pravidel. V [12] je uvedeno pravidlo pro šíři binu jednorozměrného histogramu

$$h \approx \frac{2(Q_1 - Q_3)}{n^{1/3}}, \tag{4.5}$$

kde $Q_1 - Q_3$ je mezikvartilové rozpětí dat a n počet vzorků. Scott v [33] odvozuje vztah pro optimální šířku binu, v [34] pak vztah zobecňuje pro obecný počet dimenzí. Za předpokladu normálního rozdělení dat s diagonální kovarianční maticí pro šířku binu v dimenzi i platí

$$h_i \approx \frac{3.5\hat{\sigma}_i}{n^{1/(2+d)}}, \tag{4.6}$$

kde $\hat{\sigma}_i$ je odhad směrodatné odchylky dat v dimenzi i.

Pro výpočet diferenciální entropie histogramováním je nutné aplikovat korekce v důsledku konečné šířky binů. Mezi diferenciální entropii H_d a diskretizovanou entropií H_h platí vztah [41]

$$H_d = H_h + \ln b, \tag{4.7}$$

kde b je šířka binů histogramu.

4.1.1 Vylepšení histogramu oknem

Jednou z metod kompenzace malé relativní hustoty vzorků ve vyšších dimenzích je vylepšení histogramu oknem. Nechť m_i je četnost vzorků v i-tém binu jednorozměrného histogramu a

$\varphi(x)$ je okenní funkce. Pro nové četnosti ve vyhlazeném histogramu pak platí

$$m'_j = \sum_i m_i \varphi\left(\frac{(j-i)h}{b}\right), \tag{4.8}$$

kde m_i je četnost i-tého binu před vyhlazením oknem, h šíře histogramových binů a b šířka pásma okna. Nové četnosti obecně nemusí být celočíselné. Pravděpodobnosti binů pak získáme jako

$$P'_j = \frac{m'_j}{\sum_i m'_i}. \tag{4.9}$$

4.1.2 Adaptivní histogramování

V Darbellay a Vajda [11] byla navržena metoda histogramování pro účely odhadu vzájemné informace přihlížející k uspořádání vzorků v příznakovém prostoru. Na rozdíl od obvyklého histogramování s konstantní šíří binů tato metoda iterativně zjemňuje rozlišení histogramu v závislosti na míře neuniformity rozdělení obsažených vzorků.

Adaptivní binning podle [11] je definován pro dvě náhodné proměnné $X \in \mathbb{R}^{d_1}$ a $Y \in \mathbb{R}^{d_2}$. Nechť $(\mathbf{x}_1, \mathbf{y}_1) \dots (\mathbf{x}_n, \mathbf{y}_n) \in \mathbb{R}^{(d_1+d_2)}$ jsou realizace náhodných proměnných X a Y a $F_{X,Y}$, F_X, F_Y sdružená a marginální distribuční funkce odhadnuté ze vzorků. Dále nechť r a s jsou celočíselné kladné parametry algoritmu. Autoři navrhli následující postup pro konstrukci histogramu:

1. Rozdělíme sdružený prostor jevů $\mathcal{C}^0$ podle kvantilů na r intervalů v každé dimenzi tak, že pro každou dimenzi $j = \{1 \dots d_1 + d_2\}$ a i-tý kvantil platí

$$\begin{aligned} j \le d_1 &: a_{i,j} = F^{-1}_{X,j}(i/r), && 1 \le i \le r-1, \\ j > d_1 &: a_{i,j} = F^{-1}_{Y,j-d_1}(i/r), && 1 \le i \le r-1, \end{aligned} \tag{4.10}$$

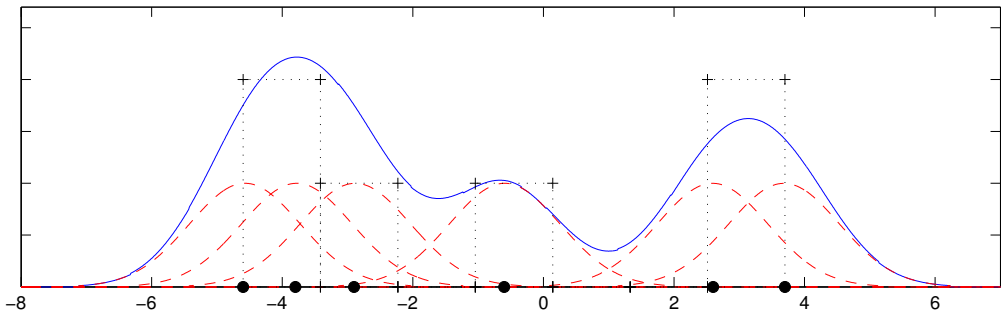
   kde $F_{A,j}$ je projekce[1] marginální distribuční funkce F_A do j-té dimenze. Výsledkem tohoto kroku je rozdělení prostoru jevů na $r^{d_1+d_2}$ disjunktních částí (buněk) $\mathcal{C}^1_k, k = \{1 \dots r^{(d_1+d_2)}\}$. Kvantily $a_{i,j}$ definují souřadnice i-tého dělení v dimenzi j.

2. Každou buňku $\mathcal{C}^p$ rozdělíme na $s^{d_1+d_2}$ buněk podle kvantilů

$$\begin{aligned} j \le d_1 &: a_{i,j} = (F^{-1}_{X,j}|\mathcal{C}^p)^{-1}(i/s), && 1 \le i \le s-1, \\ j > d_1 &: a_{i,j} = (F^{-1}_{Y,j-d_1}|\mathcal{C}^p)^{-1}(i/s), && 1 \le i \le s-1, \end{aligned} \tag{4.11}$$

   kde $F_{A,j}|\mathcal{C}^p$ je projekce podmíněné distribuční funkce $F_A|\mathcal{C}^p$ do dimenze j. Vzniklé buňky $\mathcal{C}^{p+1}_k$ pak otestujeme na rovnoměrné zastoupení vzorků. K testování je možno zvolit libovolný statistický test. Pokud je potvrzena hypotéza o rovnoměrném zastoupení vzorků,

[1] Projekcí distribuční funkce do dimenze j je míněna marginální distribuční funkce v dimenzi j.



Obrázek 4.1: Šest gaussiánů a jejich suma jako jádrový odhad hustoty pravděpodobnosti ze vzorků. Jádrový odhad je hladký a spojitý, na rozdíl od histogramu sestrojeného nad týmiž vzorky.

skončí zjemňování histogramu na úrovni buněk $\mathcal{C}^p$. V opačném případě zjemníme $\mathcal{C}^p$ podle vztahu (4.11) s nahrazením parametru s parametrem r. Pro každou dceřinnou buňku pak opakujeme postup tohoto bodu.

Odhad vzájemné informace náhodných proměnných X a Y na adaptivním histogramu sestrojeném výše popsaným postupem je pak

$$\hat{I}(X,Y) = \sum_{\mathcal{C}\in\mathcal{C}_{term}} P_{X,Y}(\mathcal{C}) \ln \frac{P_{X,Y}(\mathcal{C})}{P_X(\mathcal{C})P_Y(\mathcal{C})}, \tag{4.12}$$

kde $\mathcal{C}_{term}$ je množina terminálních buněk histogramu a $P_{X,Y}(\mathcal{C})$, $P_X(\mathcal{C})$, $P_Y(\mathcal{C})$ sdružená a marginální pravděpodobnosti buňky $\mathcal{C}$.

4.2 Jádrový odhad

Modelování hustoty pravděpodobnosti histogramováním má několik nepříjemných vlastností:

- Histogramy nemodelují hustotu pravděpodobnosti hladce
- Jsou závislé na volbě počátku binů
- Jsou závislé na volbě šířky pásma

První dva nedostatky lze odstranit nahrazením histogramu *jádrovým odhadem* hustoty pravděpodobnosti.

Nechť $\mathcal{X}_n = \{\mathbf{x}_1 \ldots \mathbf{x}_n\}$, $\mathbf{x}_i \in \mathbb{R}^d$ jsou vzorky pocházející z hustoty pravděpodobnosti $f(\mathbf{x})$. Jádrový odhad hustoty $f(\mathbf{x})$ je definován jako

$$\hat{f}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} K(\mathbf{x}-\mathbf{x}_i). \tag{4.13}$$

Funkci K označujeme jako *jádrovou funkci* pro kterou platí

$$\int_{\mathbb{R}^d} K(\mathbf{x})\mathrm{d}\mathbf{x} = 1. \tag{4.14}$$

Jádrový odhad řadíme stejně jako histogram k neparametrickým odhadům hustoty pravděpodobnosti. U jádrového odhadu každý vzorek přispívá k odhadu hustoty pravděpodobnosti hodnotou závislou na vzdálenosti od vzorku (Obr. 4.1). Jádrovou funkci obvykle volíme za účelem hladkosti odhadu spojitou. Tato funkce je definována kromě předpisu určujícího její typ (trojúhelníkové jádro, gaussián ...) také *šířkou pásma* určující rozpětí jádrové funkce. Volba šířky pásma výrazně ovlivňuje podobu modelované výsledné hustoty pravděpodobnosti; při volbě příliš malé šířky pásma dochází k výskytu falešných modů v hustotě, naopak volba příliš velké šířky pásma vede ke ztrátě informace o struktuře hustoty pravděpodobnosti.

Nejčastěji užívanou jádrovou funkcí je gaussián

$$K(\mathbf{x} - \mathbf{x}_i) = \frac{1}{(2\pi)^{d/2}|\Sigma_K|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}_i)^\top \Sigma_K^{-1}(\mathbf{x} - \mathbf{x}_i)\right). \tag{4.15}$$

V [35] je uvedeno doporučení pro kovarianční matici gaussovského jádra za předpokladu, že neznámá hustota má podobu vícerozměrného normálního rozdělení

$$\begin{aligned} \Sigma_K &= h^2\Sigma, \\ h &= n^{-1/(d+4)}, \end{aligned} \tag{4.16}$$

kde Σ je odhad kovarianční matice hustoty pravděpodobnosti.

Kapitola 5

Některé poznatky výpočetní geometrie

V mnoha úlohách je nutné strojově vyřešit určité geometrické problémy v prostorech různých dimenzí. Zobecněním postupů klasické geometrie, jejich algoritmizací a reprezentací reálných objektů v paměti počítačů se zabývá *výpočetní geometrie*. Výpočetní geometrii lze rozdělit do dvou hlavních směrů:

- *Algoritmická geometrie* se zabývá algoritmizací geometrických problémů. Úlohy algoritmické geometrie lze rozdělit na úlohy statické, hledající řešení na neměnné množině, úlohy dynamické, jež se zabývají optimalitou dynamických struktur indexace prostoru a úlohy geometrického vyhledávání. Mezi typické statické úlohy algoritmické geometrie patří konstrukce Voronoi diagramu a Delaunayovy triangulace, closest-pair problém, lineární programování aj. Dynamické úlohy se zabývají optimalitou datových struktur pro reprezentaci proměnné množiny bodů v prostoru. Úlohy geometrického vyhledávání zahrnují určení souřadnic bodu v prostoru, resp. ve struktuře tento prostor reprezentující, úlohu vyhledávání nejbližších sousedů aj. [31]
- *Numerická výpočetní geometrie* též nazývaná *geometrické modelování* se zabývá hledáním optimální formy reprezentace objektů skutečného světa pro účely modelování, simulace a počítačem asistovaného designu (CAD).

5.1 Vyhledávání nejbližších sousedů

Problém *hledání nejbližších sousedů (nearest-neighbor search, post-office problem)* je velmi důležitým problémem výpočetní geometrie. Jeho důležitost vyplývá také z nedávného vývoje disciplín strojového vidění, kde je častou úlohou vyhledávání nejpodobnějšího objektu z databáze modelů vzhledem k nějakému obrazu reality. Objekty mohou být reprezentovány *příznaky* ve vícerozměrném prostoru a úloha určení nejpodobnějšího z modelů je tak převedena na problém hledání nejbližších sousedů [39].

Problémem hledání nejbližších sousedů nazýváme úlohu identifikace bodu $\mathbf{x}_i$ z množiny $\mathcal{X}_n = \{\mathbf{x}_1 \dots \mathbf{x}_n\} \in \mathbb{R}^d$, pro který platí

$$D(\mathbf{x}_i - \mathbf{x}) < D(\mathbf{x}_j - \mathbf{x}) \quad \forall j \neq i, \tag{5.1}$$

kde $\mathbf{x}$ je *referenční bod (query point)* a D *vzdálenost*. Vzdálenost od referenčního bodu k bodu, jenž je výsledkem dotazu, je menší než vzdálenost referenčního bodu a libovolného jiného bodu z množiny $\mathcal{X}_n$. Vzdálenost může být definována různými způsoby; mezi body $\mathbf{x} = (x_1 \dots x_d)$, $\mathbf{y} = (y_1 \dots y_d)$ definujeme vzdálenost normy p jako

$$D_p = \left(\sum_{i=1}^{d} |x_i - y_i|^p \right)^{1/p}. \tag{5.2}$$

Nejčastěji užívané vzdálenosti jsou euklidovská vzdálenost ($p = 2$), Manhattan vzdálenost (též norma taxíku, $p = 1$) a Čebyševova vzdálenost ($p \to \infty$).

Existují tři základní postupy užívané pro vyhledávání nejbližších sousedů:

1. *Vyhledávání hrubou silou.* Tato metoda určí vzdálenost pro každý bod množiny $\mathcal{X}_n$ a vrátí nejbližší z bodů. Jedná se o nejjednodušší metodu, ve většině případů však nejpomalejší. Její složitost je lineární, $O(n)$.

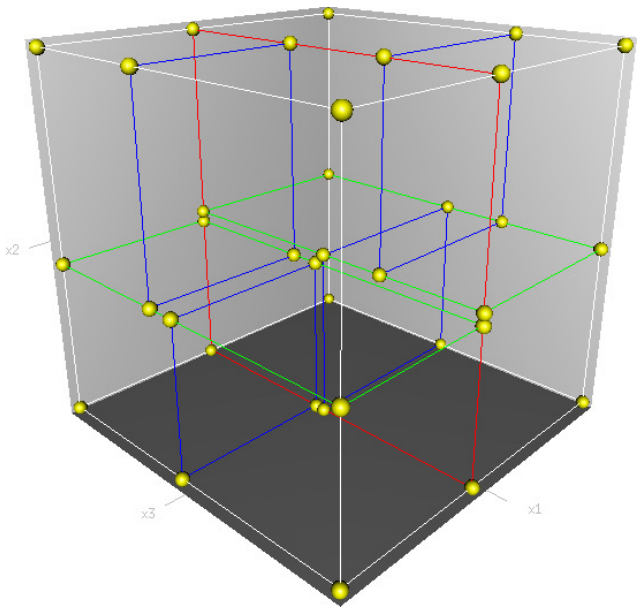
2. *Dělení prostoru (space partitioning).* Metody dělení prostoru dělí prostor do dvou či více disjunktních částí. Dělení prostoru je obvykle hierarchické, což umožňuje reprezentaci prostoru ve stromové struktuře.

3. *Prostorově citlivé hašování.* [24]

Problém vyhledávání nejbližších sousedů je vyřešen téměř optimálně pro nízké dimenzionality, pro vyšší dimenze se však stále jedná o komplikovanou úlohu. U vyhledávání nejbližších sousedů se ve vysokodimenzionálních prostorech také manifestuje „prokletí dimenzionality“, exponenciální nárust složitosti s počtem dimenzí; v [7] bylo experimentálně dokázáno, že pro dimenzionalitu $d > \ln n$, kde n je počet vzorků, je vyhledávání hrubou silou srovnatelně rychlé se space-partitioning algoritmy.

5.1.1 *k*-d stromy

k-d strom [6] je jednou z nejčastěji užívaných space-partitioning metod. k-d strom je binárním stromem; každý uzel má právě dva potomky. Na každé úrovni stromu jsou data rozdělena podle dimenze s největším rozptylem do dvou stejně velkých množin tak, že dělící nadrovina prochází mediánem bodů kolmo k souřadnici dělené dimenze. Každá vzniklá množina je pak přiřazena k jednomu z potomků děleného uzlu. Proces je opakován, dokud není dosaženo uzlů



Obrázek 5.1: Příklad trojrozměrného k-d stromu o osmi listech. Celý prostor dat je obsažen v bílé krychli. Řez první úrovně je uskutečněn jednou rovinou (červeně), obě poloviny dat jsou rozděleny zelenými rovinami na čtvrtiny. Na poslední úrovni jsou čtvrtiny rozděleny modrými rovinami na osminy. Převzato z <`http://www.stat.purdue.edu/~btyner/packages.html`>.

o velikosti menší než je dvojnásobek minimální velikosti listů N_{min}. Výsledkem tohoto procesu je pak vyvážený strom o hloubce $h = \lfloor \log_2(\frac{n}{N_{min}}) \rfloor$. Příklad jednoduchého k-d stromu je na Obr. 5.1.

Postup vyhledávání nejbližších sousedů v k-d stromech lze zapsat následujícím způsobem:

1. *Lokalizace referenčního bodu.* V k-d stromu nalezni list, jemuž odpovídající část prostoru obsahuje referenční bod.

2. *Určení horní hranice vzdálenosti.* Množinu bodů patřící listu s referenčním bodem prohledej a urči vzdálenost D_{min} k nejbližšímu sousedu referenčního bodu.

3. *Prohledání okolních listů.* Dále prohledej listy, jejichž vzdálenost k referenčnímu bodu je menší než D_{min}.

V nejhorším případě algoritmus vyhledávání prohledá cely strom, mezní složitost tedy je $O(n)$. Průměrný čas vyhledávání nejbližších sousedů v k-d stromech činí $O(\log_2 n)$.

Best Bin First vyhledávání v *k*-d stromech

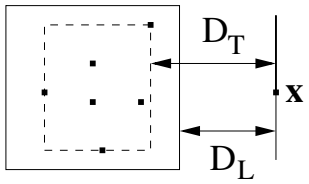
Best Bin First (BBF) [5] je modifikace algoritmu vyhledávání nejbližších sousedů v k-d stromech. Metoda BBF je užitečná, pokud nám stačí vyhledání nejbližšího souseda ve většině případů a v malé části případů souseda velmi blízkého. Algoritmu je předán jako parametr maximální počet prohledávaných listů N_{max}; k určení pořadí prohledávání stromu je užita *prioritní fronta*. Algoritmus vyhledávání BBF vracející vzdálenost k nejbližšímu nalezenému sousedu D_{nn} lze zapsat následujícím způsobem:

1. *Inicializace.* $N = 0$.
2. *Lokalizace referenčního bodu.* Pro každou úroveň k-d větvení vlož do prioritní fronty ukazatel na uzel, jehož cestou se nevydáš.
3. *Určení meze vzdálenosti.* List obsahující referenční bod prohledej a urči vzdálenost D_{min} k nejbližšímu bodu v listu. $N = N + 1$.
4. *Prořezání stromu.* Z prioritní fronty jsou odstraň ukazatele na uzly, jejichž vzdálenost od reference je $D \geq D_{min}$.
5. *Kontrola ukončení.* Pokud je prioritní fronta prázdná či pokud $N = N_{max}$, algoritmus končí. $D_{nn} = D_{min}$
6. *Průchod stromem.* Vyjmi z fronty ukazatel na uzel s nejmenší vzdáleností D k referenci. Pokud není list, procházej stromem sestupně k listům, vol cestu s menší vzdáleností k referenci. Ukazatele na uzly, které nebyly zvoleny k průchodu vlož do prioritní fronty pokud je jejich vzdálenost k referenci $D \leq D_{min}$.
7. *Prohledání listu a update meze vzdálenosti.* List prohledej, v případě nutnosti updatuj hodnotu D_{min}. $N = N + 1$. Pokračuj bodem 4.

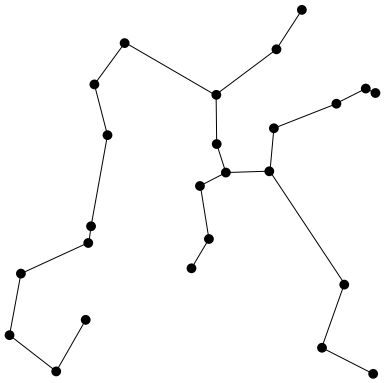
Algoritmus při hledání nejbližšího souseda prohledává listy k-d stromu v pořadí podle jejich vzdálenosti k referenčnímu bodu vzestupně. Z tohoto důvodu je efektivnější než naivní hledání v k-d stromu i v případě $N_{max} = \infty$. Alternativně lze zavést namísto počtu procházených uzlů omezení na dobu běhu.

Těsné hranice *k*-d stromů

Vzdálenost referenčního bodu od některého z uzlů je obvykle odhadována jako vzdálenost k hranici (nad)krychle prostoru, kterou daný uzel reprezentuje. Těsné hranice k-d stromu [38] jsou zavedeny za účelem lepšího odhadu nejmenší možné vzdálenosti referenčního bodu k nejbližšímu bodu z uzlu. Těsná hranice je nejmenší možná (nad)krychle obklopující body v uzlu (Obr. 5.2).



Obrázek 5.2: Ilustrace znázorňující těsnou (přerušovaná čára) a volnou (plná čára) hranici uzlu. Volná hranice je totožná s částí prostoru pokrytou uzlem k-d stromu. Vzdálenost bodu $\mathbf{x}$ k těsné hranici uzlu (D_T) je lepším odhadem vzdálenosti k nejbližšímu bodu z uzlu než vzdálenost k volné hranici uzlu (D_L).



Obrázek 5.3: Příklad minimální kostry rovinného euklidovského grafu. Existuje cesta mezi všemi body a celková délka kostry je nejmenší možná.

5.2 Minimální kostra euklidovského grafu

Euklidovský graf je uspořádaná dvojice $G = (V, E)$, kde $V = (\mathbf{v}_1 \dots \mathbf{v}_n), \mathbf{v}_i \in \mathbb{R}^d$ je množina *vrcholů* a $E = (e_1 \dots e_m)$ je množina *hran*. Každá hrana euklidovského grafu e_i je definována množinou právě dvou prvků z V, $e_i = (\mathbf{v}_j, \mathbf{v}_k)$ a *délka hrany* je euklidovskou vzdáleností bodů $\mathbf{v}_j$, $\mathbf{v}_k$. *Minimální kostra euklidovského grafu (minimum spanning tree)* je pak takový podgraf $G' = (V', E')$ grafu G, pro který platí:

1. $V' = V$, $E' \subset E$,
2. Pro všechna $\mathbf{v}_i, \mathbf{v}_j \in V'$ existuje cesta mezi $\mathbf{v}_i$ a $\mathbf{v}_j$,
3. Pro celkovou délku L_E všech hran z E' platí $L_E = \text{argmin}_{E'} \sum_{i=1}^{|E'|} |e_i|$, kde $|e_i|$ značí euklidovskou vzdálenost mezi vrcholy hrany e_i.

Minimální kostra grafu spojuje všechny vrcholy grafu kombinací hran s nejmenší celkovou délkou (Obr. 5.3).

Problém minimální kostry euklidovského grafu byl poprvé formulován O. Borůvkou r. 1926 [8]. Borůvka řešil praktický problém návrhu elektrorozvodné sítě. K formulaci problému minimální kostry jej vedl požadavek minimalizace nákladů na výstavbu sítě – tedy minimální délky elektrorozvodného vedení [28].

Kapitola 6

Implementace

Vybrané estimátory entropie a vzájemné informace byly implementovány v jazyce C. Estimátory jsou napsány jako statické moduly. V této kapitole jsou popsány datové struktury a hlavičkové soubory nezbytné k volání estimátorů z jiných modulů a hlavní myšlenky implementace estimátorů. Dále jsou popsány volby parametrů pro experimenty. Důležité vnitřní funkce estimátorů jsou popsány v příloze, podrobné informace o funkcích a datových strukturách jsou uvedeny v komentářích ve zdrojovém kódu.

V této kapitole je pro popis funkcí a datových struktur užito notace jazyka C.

6.1 Obecné informace o implementaci estimátorů, komunikace s vnějšími moduly

Všechny estimátory byly napsány na platformě Linux. Estimátory jsou určeny k sestavení kompilátorem GCC.

Všechny moduly užívají za účelem rychlosti v maximální možné míře ukazatele na data obsažená ve vstupní struktuře `sImage`. Tato struktura je definována

```
typedef struct sImage {
   int samples;
   int dimension;
   double **data;
}  sImage
```

kde `int samples` je počet vzorků, `int dimension` dimenze dat a `double **data` ukazatel velikosti `samples` na ukazatele velikosti `dimension` na prvky typu `double`. Struktura `sImage` je definována v hlavičkovém souboru `imtype.h`. Strukturu `imtype.h` je nutno zahrnout do

volajícího programu stejně jako některý z následujících souborů, deklarujících globální funkce estimátorů:

- `histo.h` – Hlavičkový soubor deklarující globální funkce histogramového estimátoru entropie a vzájemné informace (včetně modifikace s vyhlazením histogramu) a strukturu parametrů `sParamHist`.
- `kdest.h` – Hlavičkový soubor deklarující globální funkce estimátoru entropie a vzájemné informace založeného na nejbližších sousedech a estimátoru α-entropie a vzájemné informace založeného na délce minimální kostry grafu nad vzorky. Deklaruje strukturu parametrů `sParamTree`.
- `kdens.h` – Hlavičkový soubor deklarující globální funkce estimátoru entropie a vzájemné informace s jádrovým odhadem.
- `ada.h` – Hlavičkový soubor deklaruje globální funkce estimátoru vzájemné informace s adaptivním histogramováním a strukturu parametrů `sParamAda`.

6.2 Estimátor entropie a vzájemné informace s histogramováním

Estimátor entropie a vzájemné informace s histogramováním je volán funkcemi

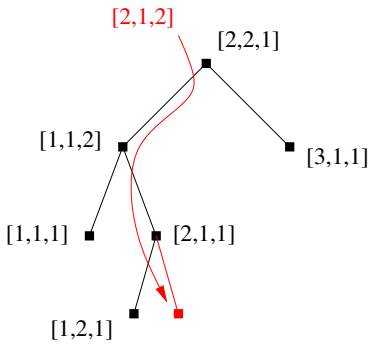
- `double histH(sImage *im1, sParamHist *par)`
- `double histMI(sImage *im1, sImage *im2, sParamHist *par)`

Parametrická struktura estimátoru je definována

```
typedef struct sParamHist {
  int parzen;
  double parzenMargin;
  int chopHist;
  int align;
} sParamHist;
```

Hodnota parametru `parzen` musí být pro tento estimátor inicializována 0. Nastavením parametru `align` na 1 je možno upravit v každé dimenzi šíře histogramových binů na nejbližší vyšší hodnotu, jejíž celočíselný násobek je roven rozdílu krajních hodnot dat. Jiných parametrů estimátor nevyužívá.

Estimátor s histogramováním modeluje hustotu pravděpodobnosti vícerozměrným histogramem. Algoritmus estimátoru lze rozčlenit do následujících kroků:



Obrázek 6.1: Přiklad vložení prvku do stromu s neexistujícím příslušným binem. Algoritmus vytvoří nový uzel reprezentující bin o souřadnicích $[2, 1, 2]$.

1. *Určení parametrů výpočtu.* Nejprve algoritmus určí rozptyl, počty a šíři binů v jednotlivých dimenzích podle Scottova pravidla.

2. *Naplnění histogramu.* Samotný histogram je v operační paměti reprezentován binárním stromem s naivním vkládáním prvků. Jako klíč jsou zvoleny souřadnice histogramového binu, kde první dimenze zastává úlohu nejvýznamnější a poslední dimenze nejméně významné číslice v popisovači. Pokud algoritmus při vkládání prvku nalezne ve stromové struktuře správný bin, inkrementuje jej o jednotku, v opačném případě vytvoří nový bin (Obr. 6.1).

3. *Estimace entropie.* Algoritmus prochází binární strom do hloubky, uvolňuje operační paměť a počítá entropii podle četností dat v binech. Po průchodu celým stromem jsou aplikovány korekce entropie v důsledku konečné šířky binů [41].

Pro výpočet vzájemné informace jsou výše uvedené kroky provedeny pro marginální a sdruženou entropii. Výsledek je pak spočten podle vztahu (3.19). Parametr `align` byl v experimentech nastaven na 0, není-li uvedeno jinak.

6.3 Estimátor entropie a vzájemné informace s histogramováním a vylepšením histogramu oknem

Estimátor je volán shodnými funkcemi jako předchozí. Hodnota `parzen` paramerické struktury musí být inicializována 1.

Estimátor s histogramováním a vylepšením histogramu oknem je implementován podobně jako předchozí estimátor s několika odchylkami.

1. *Určení parametrů výpočtu.* Algoritmus opět určí rozptyl, počty a šíře binů v jednotlivých dimenzích, navíc určí šířku pásma okna v jednotlivých dimenzích na základě Silvermanova pravidla [3]

$$\sigma_{BW} = 0.9An^{1/5}, \quad A = \min\left\{\sigma, \frac{Q_3 - Q_1}{1.34}\right\}, \tag{6.1}$$

   kde σ je odhad směrodatné odchylky a $Q_3 - Q_1$ mezikvartilové rozpětí. Na základě těchto šířek pásma a parametru určujícího dolní mez inkrementace histogramu je spočtena maximální vzdálenost vyhlazování v jednotlivých dimenzích.

2. *Naplnění histogramu.* První část tohoto kroku je totožná s předchozím estimátorem. Po vložení každého bodu ze vstupních dat do histogramu jsou však navíc inkrementovány okolní biny o hodnotu (resp. vytvořeny nové biny s hodnotou) odpovídající hodnotě vícerozměrného gaussiánu s diagonální kovarianční maticí ve vzdálenosti x od počátku souřadnic, kde x je vzdálenost středu okolního binu od středu binu s původním (centrálním) bodem v přirozených jednotkách dat. Dolní limit inkrementace je omezen hodnotou položky `parzenMargin` parametrické struktury.

3. *Estimace entropie.* Tento krok je totožný jako u předchozího estimátoru.

U estimátoru byla zvolena dolní mez inkrementace binů jako 0.1% centrální hodnoty.

6.4 Estimátor entropie a vzájemné informace s jádrovým odhadem

Estimátor je volán funkcemi

- `double kdensH(sImage *img1)`
- `double kdensMI(sImage *img1, sImage *img2)`

Estimátor nemá žádné vstupní parametry.

Algoritmus počítá hodnotu entropie (vzájemné informace) s využitím vztahu (3.24). Hustota pravděpodobnosti je odhadnuta pro všechny body z množiny vzorků na vstupu algoritmu.

1. *Výpočet kovarianční matice dat.* Algoritmus určí kovarianční matici vstupních dat. Tento krok je nezbytný k výpočtu podoby jádrové funkce.

2. *Určení podoby jádra.* Jádrová funkce je gaussián s obecnou kovarianční maticí získanou podle předpisů (4.16).

3. *Výpočet hodnoty hustoty pravděpodobnosti v bodech z dat.* Hustota je počítána jako součet příspěvků jednotlivých bodů ze vstupních dat. Algoritmus uvažuje k výpočtu hustoty pravděpodobnosti příspěvky od všech bodů, pracuje tedy s kvadratickou složitostí.

4. *Odhad entropie.* K odhadu je přímo užito vztahu (3.24).

6.5 Estimátor entropie a vzájemné informace s vyhledáváním nejbližších sousedů

Estimátor je volán funkcemi

- `double kdNnMI(sImage *i1, sImage *i2, sParamTree *par)`
- `double kdNnH(sImage *i1, sParamTree *par)`

Parametry estimátoru jsou uloženy ve struktuře

```
typedef struct sParamTree {
  int minLeafSize;
  int tightBoxes;
  int maxStepsBBF;
  int deMethod;
} sParamTree;
```

Estimátor vyhledává nejbližší sousedy Best Bin First (BBF) vyhledáváním v k-d stromech. Algoritmus lze rozdělit do následujících kroků:

1. *Konstrukce k-d stromu.* Nejprve je vytvořena k-d strom půlením dat v dimenzi s největším rozptylem. Konstrukce je zastavena, pokud počet bodů v uzlu stromu poklesne pod dvojnásobek hodnoty `minLeafSize`.

2. *Vyhledávání nejbližších sousedů.* Pro každý bod z množiny vstupních dat je určena vzdálenost k nejbližšímu sousedu. V k-d stromech je vyhledáváno BBF vyhledáváním. Je možno omezit maximální počet procházených uzlů či jej ponechat neomezený (položka `maxStepsBBF` struktury parametrů). Omezení počtu procházených binů značně urychlí výpočet, ne vždy je však nalezen skutečný nejbližší soused.

3. *Uvolnění pamětu a výpočet entropie.* Entropie je spočtena podle Kozačenkova-Leoněnkova vztahu.

Nastavením parametru `tightBoxes` na 1 nebo 0 je možno zvolit užití těsných hranic uzlu pro výpočet vzdálenosti referenčního bodu od oblasti reprezentované uzlem. Volbou parametru `deMethod` je možno aktivovat užití varianty Kozačenko-Leoněnkova vztahu pro degenerované rozdělení. Parametr `minLeafSize` definuje minimální velikost listů k-d stromů.

Pro všechny experimenty bylo zvoleno `tightBoxes` $= 1$ a `deMethod` $= 0$. Minimální velikost listů k-d stromu byla určena experimentálně. Byly testovány 3 varianty estimátoru s hodnotami `maxStepsBBF` $= \{\infty, 20, 10\}$.

6.6 Estimátor α-entropie a vzájemné informace

Estimátor je volán

- `double kdMstMI(sImage *i1, sImage *i2, sParamTree *par)`
- `double kdMstH(sImage *i1, sParamTree *par)`

Struktura parametrů je shodná jako u předchozího estimátoru.

Estimátor α-entropie odhaduje hodnotu α-entropie podle délky minimální kostry úplného grafu, sestrojeného nad body z množiny vstupních dat. K reprezentaci prostoru je využit k-d strom, na počet procházených uzlů v dotazech není uvaleno omezení co se počtu týče. Algoritmus užívá k výpočtu délky minimální kostry Primova algoritmu. Z k-d stromu je vždy odejmut bod nejbližší libovolnému bodu z dosud zkonstruované části kostry a jeho vzdálenost od kostry je připočtena k celkové délce. Estimátor pracuje v následujících krocích:

1. *Konstrukce k-d stromu.* Totožné s předchozím estimátorem.
2. *Výpočet délky minimální kostry.* Estimátor počítá délku minimální kostry nad všemi body vstupních dat. Algoritmus vždy hledá bod nejblíže libovolnému body z kostry dotazem na nejbližšího souseda pro všechny body z kostry. Bod, který je výsledkem tohoto dotazu je pak přidán do kostry a odstraněn z k-d stromu. Jeho vzdálenost od kostry je přičtena do proměnné reprezentující váženou délku kostry. Postup se opakuje až do úplného vyprázdnění k-d stromu. Váhovací konstanta byla zvolena $\gamma = 0.5$ v souladu s [16], hodnota α tedy je $\alpha = \frac{d-\gamma}{d}$ kde d je dimenze dat.
3. *Výpočet α-entropie.* Výpočet je realizován podle (3.32).

U estimátoru bylo zvoleno `maxStepsBBF` $= \infty$, `tightBoxes` $= 1$. Minimální velikost listů k-d stromu byla určena experimentálně.

6.7 Estimátor vzájemné informace s adaptivním histogramováním

Estimátor je volán funkcí

- `double adaptiveMI(sImage * im1, sImage * im2, sParamAda * par);`

Struktura s parametry nabývá podoby

```
typedef struct sParamAda {
  int r;
  int s;
  int minLeaf;
} sParamAda;
```

Volbou `minLeaf` je možno stanovit minimální počet prvků obsažených v buňce.

Estimátor implementuje algoritmus adaptivního histogramování podle [11]. Algoritmu je předán jako parametr počet dělení v každé dimenzi (r) a počet dělení v každé dimenzi pro testování uniformity obsažených vzorků (s). Algoritmus pracuje v následujících krocích:

1. *Inicializace histogramu.* Algoritmus inicializuje struktury reprezentující histogram a naplní proměnné nezbytné pro běh.

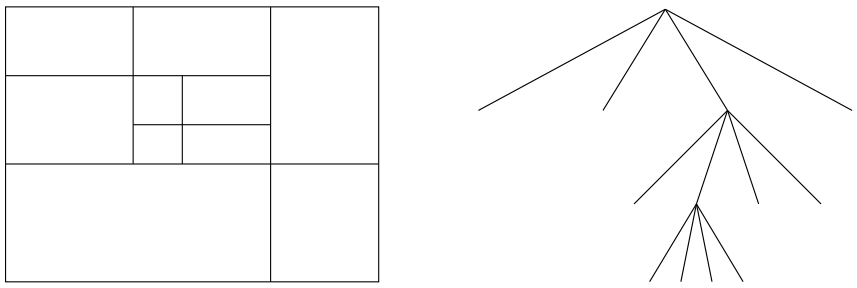
2. *Prvotní partitioning vzorků.* Algoritmus rozdělí prostor vzorků na r^d buněk v souladu s definicí algoritmu.

3. *Rekurzivní dělení a testování uniformity rozdělení pro potomky.* Každá část histogramu je rozdělena podle marginálních kvantilů pravděpodobností na s^d buněk. Na těchto buňkách je poté testována uniformita zastoupení vzorků. V případě potvrzení hypotézy o uniformitě je dělení buněk stornováno, v opačném případě je ponecháno pokud $r = s$ či přepočítáno pro parametr r pokud $r \neq s$.

4. *Výpočet vzájemné informace.* Vzájemní informace je spočtena na základě marginálních a sdružených pravděpodobností buněk podle (3.20).

Adaptivní histogram je v paměti reprezentován stromem (Obr. 6.2). Každý neterminální uzel stromu má r^d potomků, kde r je parametr algoritmu a d dimenze dat. Každý uzel rovněž nese informaci o sdružené pravděpodobnosti a o marginálních pravděpodobnostech buňky, kterou reprezentuje. Jako test uniformity byl zvolen χ^2 test dobré shody s mezní hladinou významnosti 5%.

Pro experimenty prováděné s estimátorem využívajícím adaptivní histogramování byly zvoleny parametry $r = 2$ a $s = 2$ v souladu s doporučením autorů.



(a) (b)

Obrázek 6.2: Reprezentace adaptivního histogramu stromem. Adaptivní histogram (a) a odpovídající strom (b).

Kapitola 7

Experimenty

V této kapitole jsou prezentovány experimenty uskutečněné s estimátory entropie a vzájemné informace popsanými v kapitole 6. Těžiště kapitoly představuje vyhodnocení vlastností estimátorů pro odhad entropie normálního rozdělení; jejich přesnosti, rozptylu, střední kvadratické chyby a časů běhu. Estimátory jsou dále testovány na rovnoměrném rozdělení. V závěru kapitoly jsou srovnány strategie histogramování pro odhad vzájemné informace.

V textu jsou užita následující označení estimátorů:

- $\hat{H}_h$, $\widehat{MI}_h$ - histogramový estimátor (viz. 6.2).
- $\hat{H}_{hs}$, $\widehat{MI}_{hs}$ - histogramový estimátor s vyhlazením histogramu oknem (6.3).
- $\widehat{MI}_{ada}$ - histogramový estimátor vzájemné informace s adaptivním histogramováním (6.7).
- $\hat{H}_{kde}$ - estimátor využívající jádrový odhad hustoty pravděpodobnosti (6.4).
- $\hat{H}_{nn}$ - estimátor založený na nejbližších sousedech s BBF vyhledáváním (6.5).
- $\hat{H}_{nn,10}$ - estimátor založený na nejbližších sousedech s BBF vyhledáváním a nejvýše 10 prohledávanými listy.
- $\hat{H}_{nn,20}$ - estimátor založený na nejbližších sousedech s BBF vyhledáváním a nejvýše 20 prohledávanými listy.
- $\hat{H}_{\alpha}$ - estimátor α-entropie.

7.1 Určení optimální velikosti listů *k*-d stromů

Estimátor založený na vyhledávání nejbližších sousedů a estimátor α-entropie využívají pro space-partitioning k-d stromy. Hloubka k-d stromu je určena minimální velikostí listů N_{min}.

Cílem experimentu je určit optimální velikost tohoto parametru pro různé dimenze dat.

V rámci tohoto experimentu byla měřena rychlost výpočtu entropie pro 10^4 prvků, dimenzionalitu dat $d = \{1, 3, 6, 9, 12\}$ a pro minimální velikosti listů $N_{min} = \{5, 10, 15, 20, 25, 30\}$. Data pocházela z normálního rozdělení $N(0, 1)$, resp. z vícerozměrného normálního rozdělení $N(\mathbf{o}_n, I_n)$. Maximální počet prohledávaných listů nebyl omezen. Výsledky (Tab. 7.1) jsou získány průměrováním z 50 běhů. Na základě těchto výsledků byla pro následující experimenty u estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$ zvolena pro dimenze $d \leq 4$ minimální velikost listů $N_{min} = 5$ a pro dimenze $d \geq 5$ minimální velikost listů $N_{min} = 10$.

d/N_{min}	5	10	15	20	25	30
1	**0,086**	0,087	0,087	0,102	0,102	0,103
3	**0,185**	0,187	0,193	0,227	0,224	0,226
6	0,861	**0,859**	0,915	1,1027	1,115	1,116
9	3,956	**3,903**	3,989	4,612	4,680	4,705
12	17,148	**15,033**	16,884	16,374	16,778	15,234

Tabulka 7.1: Průměrné doby výpočtu entropie pro nearest-neighbor estimátor. Nejkratší časy jsou znázorněny tučně.

7.2 Statistické vlastnosti, přesnost a rychlost estimátorů entropie

Cílem experimentů je vyhodnotit rychlost, průměrnou odchylku od skutečné hodnoty a rozptyl estimátorů entropie. Výsledky byly získány pro 100 běhů estimátorů pro počty prvků $n < 10^5$, pro 40 běhů u $10^5 \leq n < 10^6$ a pro 20 běhů v případě $n = 10^6$.

V tabulkách je užito následujících označení:

- $\overline{\hat{H}}$ – průměrná hodnota odhadu entropie v natech.
- σ – směrodatná odchylka odhadu entropie.
- bias – průměrná odchylka odhadu od skutečné hodnoty entropie

$$\text{bias} = \overline{\hat{H}} - H_{true} \tag{7.1}$$

kde H_{true} je skutečná hodnota entropie hustoty pravděpodobnosti generující data. Pro účely vykreslení v logaritmických souřadnicích je u grafů uvažována absolutní hodnota bias.

- T – průměrný čas běhu v s.

7.2.1 Odhad entropie normálního rozdělení

Cílem experimentu je vyhodnotit rychlost a přesnost estimátorů entropie pro normální rozdělení různých dimenzionalit. Testovací data pochází z normálního rozdělení s diagonální kovarianční maticí. Pro skutečnou hodnotu diferenciální entropie normálního rozdělení s kovarianční maticí Σ platí [2]

$$H = \frac{1}{2}\ln((2\pi e)^d|\Sigma|). \tag{7.2}$$

Test byl proveden pro dimenze $d = \{1, 3, 6, 9, 12\}$. Pro $n \leq 10^2$ nebyl měřen čas běhu. Testování estimátorů bylo ukončeno, pokud čas běhu překročil 400 s.

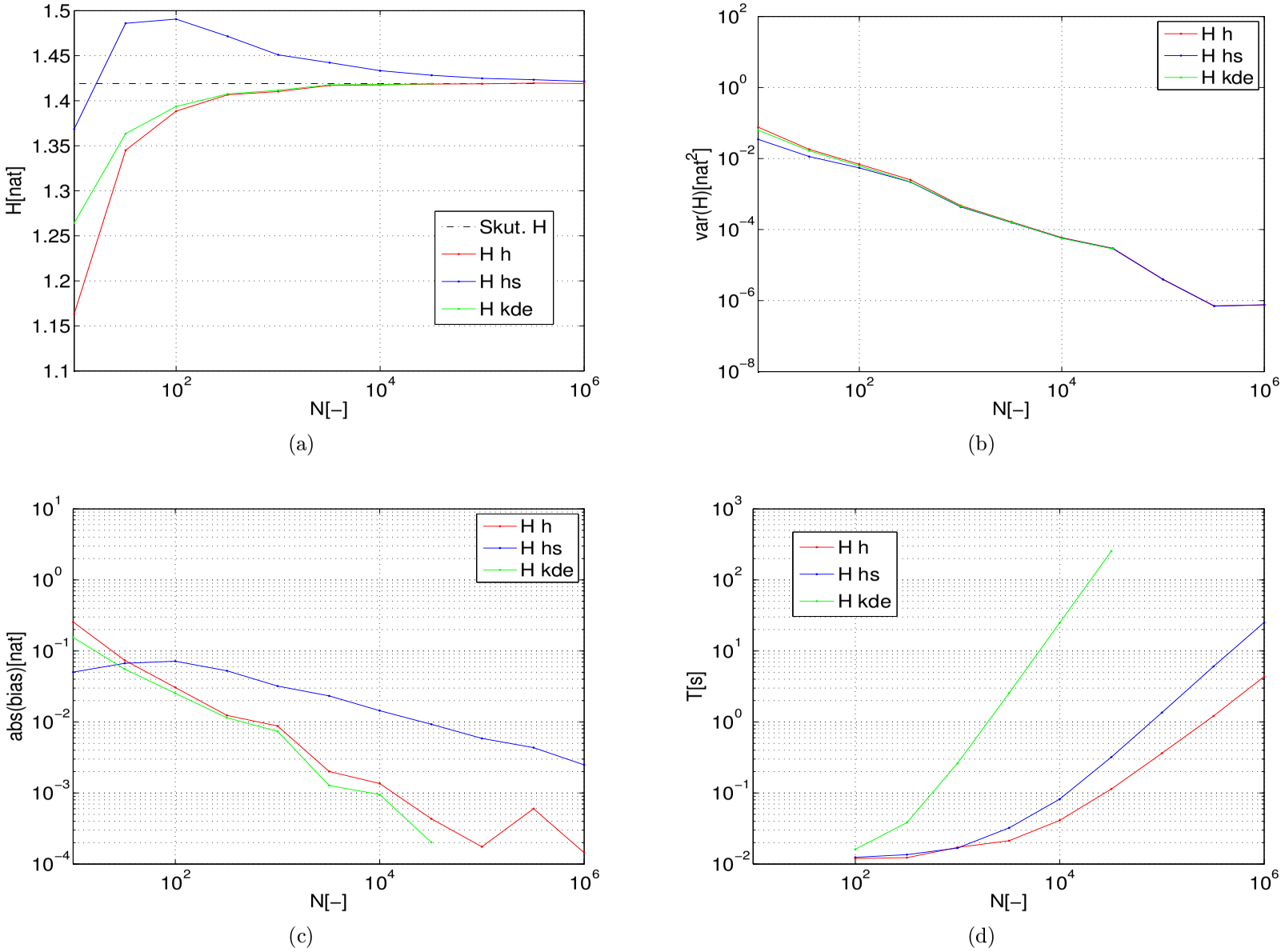
Střední kvadratická chyba a časy běhů všech estimátorů entropie jsou porovnány v části 7.2.3.

	$\hat{H}_h$				$\hat{H}_{hs}$				$\hat{H}_{kde}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	1.1631	0.2767	-0.2558		1.3686	0.1864	-0.0504		1.2642	0.2512	-0.1547	
$3.2 * 10^1$	1.3450	0.1347	-0.0739		1.4859	0.1067	0.0670		1.3635	0.1286	-0.0555	
10^2	1.3884	0.0830	-0.0306	0.0120	1.4905	0.0738	0.0716	0.0124	1.3936	0.0789	-0.0254	0.0161
$3.2 * 10^2$	1.4066	0.0502	-0.0124	0.0123	1.4715	0.0466	0.0526	0.0135	1.4075	0.0471	-0.0115	0.0383
10^3	1.4102	0.0218	-0.0088	0.0171	1.4510	0.0209	0.0321	0.0168	1.4116	0.0213	-0.0074	0.2620
$3.2 * 10^3$	1.4169	0.0128	-0.0020	0.0213	1.4422	0.0125	0.0233	0.0323	1.4177	0.0126	-0.0013	2.5597
10^4	1.4176	0.0077	-0.0014	0.0411	1.4334	0.0076	0.0144	0.0820	1.4180	0.0075	-0.0010	24.9062
$3.2 * 10^4$	1.4185	0.0054	-0.0004	0.1136	1.4282	0.0054	0.0093	0.3198	1.4187	0.0054	-0.0002	254.4511
10^5	1.4188	0.0020	-0.0002	0.3624	1.4248	0.0020	0.0059	1.3530				
$3.2 * 10^5$	1.4195	0.0008	0.0006	1.2154	1.4233	0.0008	0.0043	6.0610				
10^6	1.4191	0.0009	0.0001	4.3569	1.4214	0.0009	0.0025	25.2037				

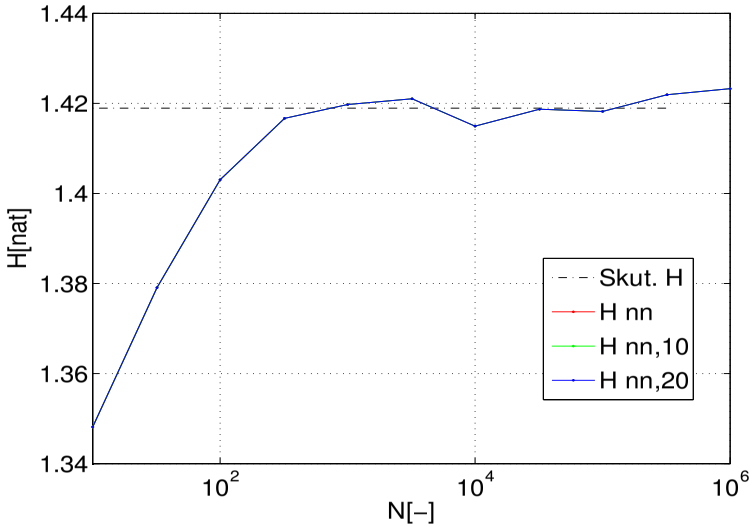
Tabulka 7.2: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho směrodatná odchylka, bias a čas běhu pro $N(0,1)$ s $d = 1$. Skutečná hodnota entropie $H = 1.4189$.

	$\hat{H}_{nn}$				$\hat{H}_{nn,10}$				$\hat{H}_{nn,20}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	1.3482	0.5017	-0.0708		1.3482	0.5017	-0.0708		1.3482	0.5017	-0.0708	
$3.2 * 10^1$	1.3791	0.2719	-0.0398		1.3791	0.2719	-0.0398		1.3791	0.2719	-0.0398	
10^2	1.4030	0.1701	-0.0159	0.0116	1.4030	0.1701	-0.0159	0.0112	1.4030	0.1701	-0.0159	0.0126
$3.2 * 10^2$	1.4167	0.0903	-0.0023	0.0131	1.4167	0.0903	-0.0023	0.0114	1.4167	0.0903	-0.0023	0.0116
10^3	1.4197	0.0516	0.0008	0.0199	1.4197	0.0516	0.0008	0.0201	1.4197	0.0516	0.0008	0.0203
$3.2 * 10^3$	1.4210	0.0225	0.0021	0.0519	1.4210	0.0225	0.0021	0.0453	1.4210	0.0225	0.0021	0.0496
10^4	1.4149	0.0168	-0.0040	0.1165	1.4149	0.0168	-0.0040	0.1089	1.4149	0.0168	-0.0040	0.1212
$3.2 * 10^4$	1.4187	0.0090	-0.0002	0.4085	1.4187	0.0090	-0.0002	0.3933	1.4187	0.0090	-0.0002	0.3913
10^5	1.4182	0.0058	-0.0007	1.5033	1.4182	0.0058	-0.0007	1.5018	1.4182	0.0058	-0.0007	1.5427
$3.2 * 10^5$	1.4219	0.0012	0.0030	7.2580	1.4219	0.0012	0.0030	6.7938	1.4219	0.0012	0.0030	6.5997
10^6	1.4233	0.0014	0.0043	24.2972	1.4233	0.0014	0.0043	23.9986	1.4233	0.0014	0.0043	23.9939

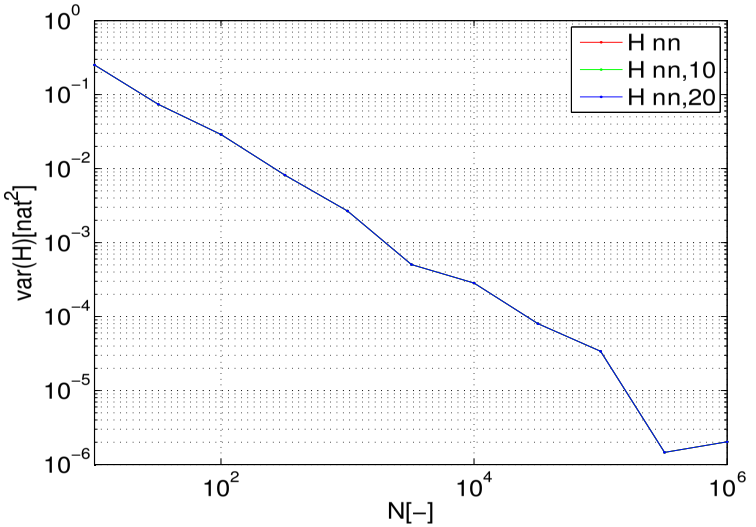
Tabulka 7.3: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho směrodatná odchylka, bias a čas běhu pro $N(0,1)$ s $d = 1$. Skutečná hodnota entropie $H = 1.4189$.



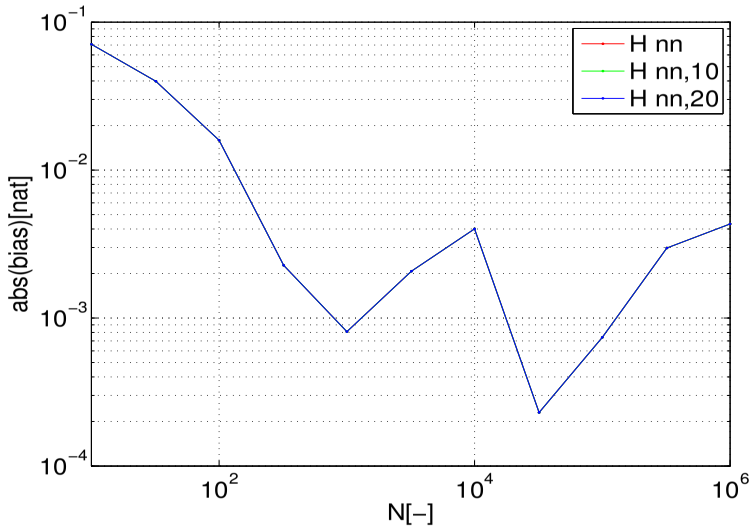
Obrázek 7.1: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho rozptyl, bias a čas běhu pro $N(0,1)$ s $d = 1$. Skutečná hodnota entropie $H = 1.4189$.



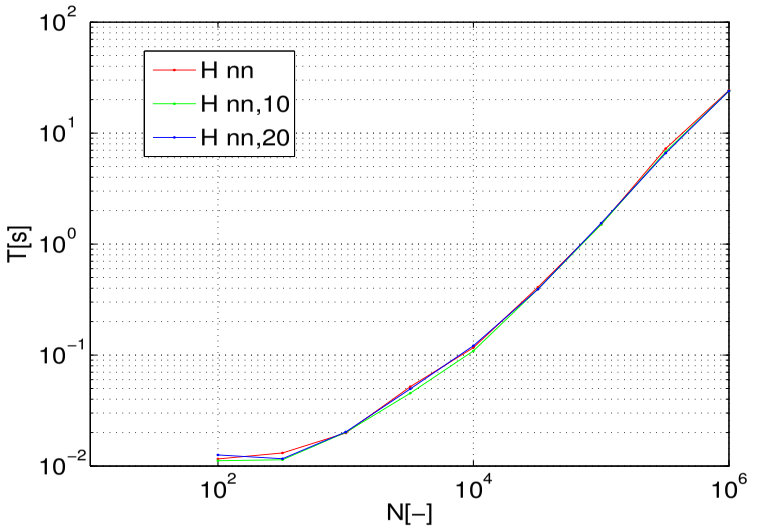
(a)



(b)



(c)



(d)

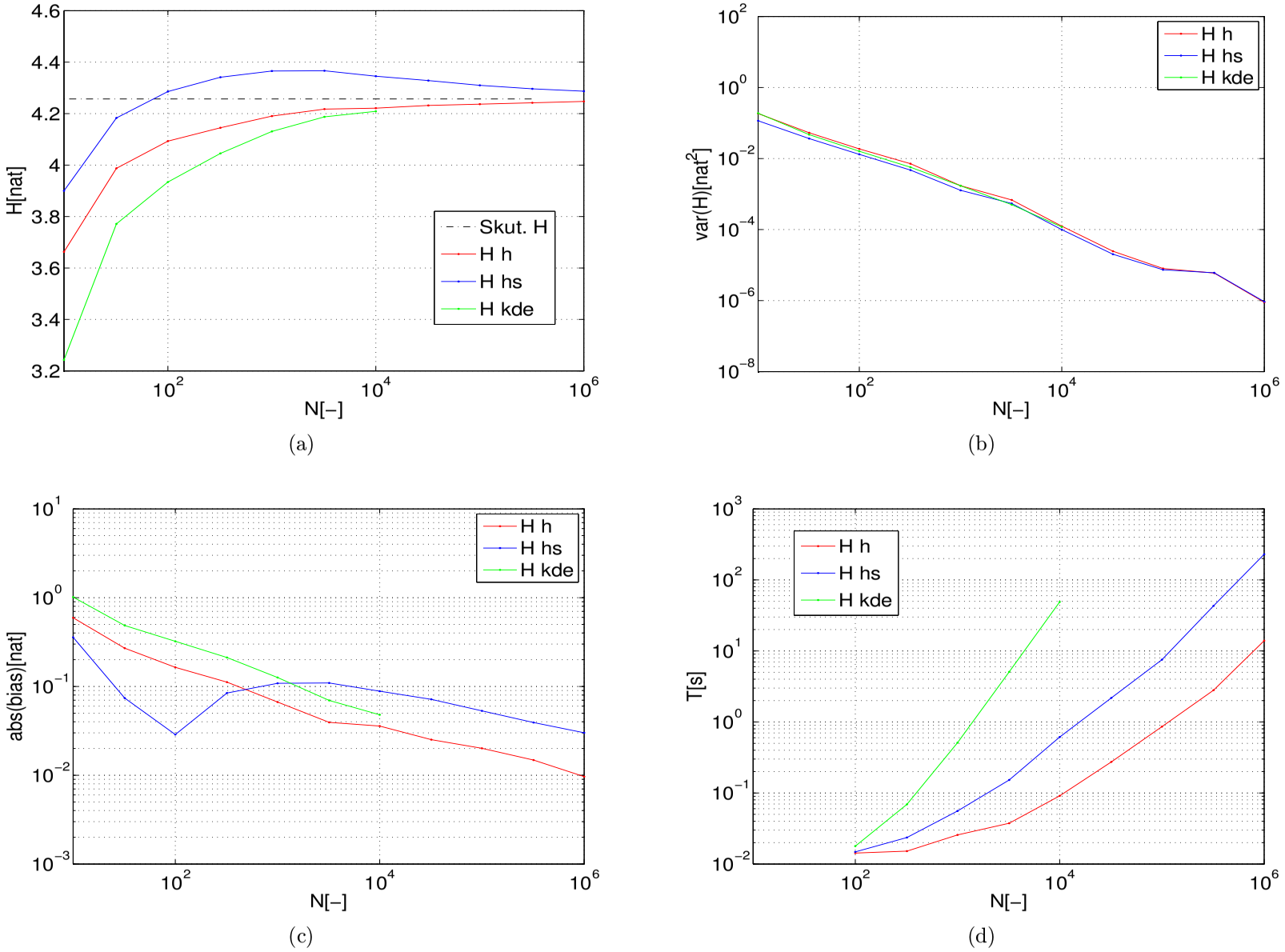
Obrázek 7.2: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho rozptyl, bias a čas běhu pro $N(0,1)$ s $d = 1$. Skutečná hodnota entropie $H = 1.4189$.

	$\hat{H}_h$				$\hat{H}_{hs}$				$\hat{H}_{kde}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	3.6629	0.4300	-0.5939		3.8998	0.3406	-0.3570		3.2430	0.4346	-1.0138	
$3.2 * 10^1$	3.9873	0.2307	-0.2695		4.1828	0.1912	-0.0740		3.7714	0.2171	-0.4854	
10^2	4.0928	0.1368	-0.1640	0.0143	4.2856	0.1146	0.0287	0.0148	3.9340	0.1267	-0.3228	0.0178
$3.2 * 10^2$	4.1449	0.0846	-0.1119	0.0152	4.3411	0.0688	0.0843	0.0236	4.0452	0.0755	-0.2116	0.0692
10^3	4.1901	0.0415	-0.0667	0.0257	4.3654	0.0357	0.1086	0.0558	4.1306	0.0413	-0.1262	0.5113
$3.2 * 10^3$	4.2174	0.0261	-0.0394	0.0376	4.3663	0.0233	0.1095	0.1525	4.1873	0.0224	-0.0695	5.0740
10^4	4.2210	0.0112	-0.0358	0.0911	4.3451	0.0100	0.0883	0.6126	4.2089	0.0109	-0.0479	48.9700
$3.2 * 10^4$	4.2317	0.0050	-0.0252	0.2741	4.3283	0.0045	0.0715	2.1834				
10^5	4.2367	0.0028	-0.0201	0.8596	4.3099	0.0027	0.0531	7.5214				
$3.2 * 10^5$	4.2420	0.0024	-0.0148	2.7996	4.2961	0.0025	0.0392	43.1696				
10^6	4.2472	0.0009	-0.0096	13.9318	4.2869	0.0010	0.0301	229.2739				

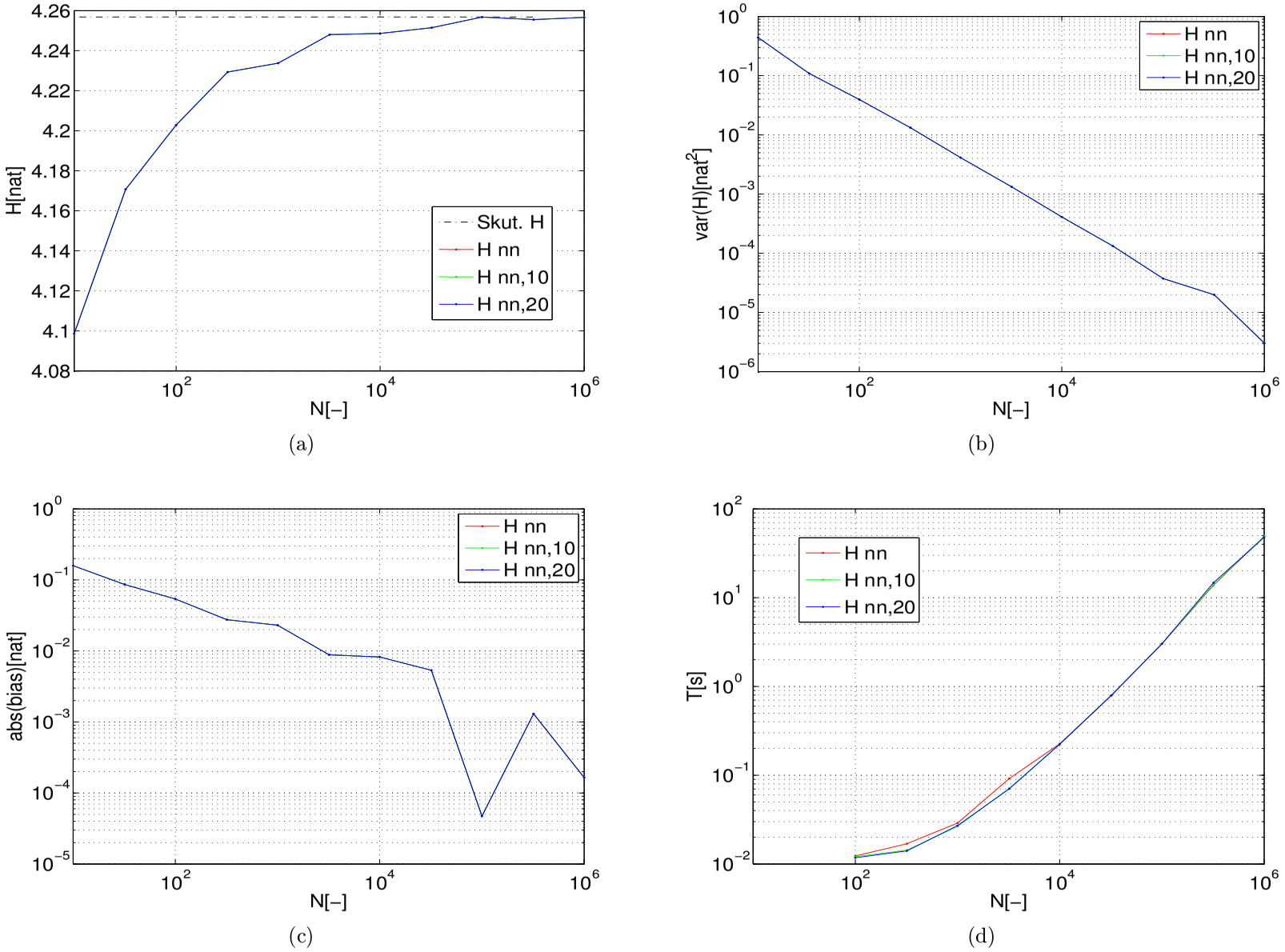
Tabulka 7.4: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho směrodatná odchylka, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 3$. Skutečná hodnota entropie $H = 4.2568$.

	$\hat{H}_{nn}$				$\hat{H}_{nn,10}$				$\hat{H}_{nn,20}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	4.0985	0.6642	-0.1583		4.0985	0.6642	-0.1583		4.0985	0.6642	-0.1583	
$3.2 * 10^1$	4.1707	0.3301	-0.0861		4.1707	0.3301	-0.0861		4.1707	0.3301	-0.0861	
10^2	4.2028	0.1988	-0.0540	0.0123	4.2028	0.1988	-0.0540	0.0121	4.2028	0.1988	-0.0540	0.0118
$3.2 * 10^2$	4.2293	0.1149	-0.0275	0.0169	4.2293	0.1149	-0.0275	0.0144	4.2293	0.1149	-0.0275	0.0140
10^3	4.2337	0.0642	-0.0231	0.0289	4.2337	0.0642	-0.0231	0.0265	4.2337	0.0642	-0.0231	0.0271
$3.2 * 10^3$	4.2480	0.0364	-0.0088	0.0915	4.2480	0.0364	-0.0088	0.0710	4.2480	0.0364	-0.0088	0.0701
10^4	4.2486	0.0202	-0.0082	0.2257	4.2486	0.0202	-0.0082	0.2228	4.2486	0.0202	-0.0082	0.2235
$3.2 * 10^4$	4.2515	0.0115	-0.0053	0.7921	4.2515	0.0115	-0.0053	0.7966	4.2515	0.0115	-0.0053	0.7913
10^5	4.2568	0.0061	-0.0000	3.0198	4.2568	0.0061	-0.0000	3.0171	4.2568	0.0061	-0.0000	3.0225
$3.2 * 10^5$	4.2555	0.0045	-0.0013	13.8128	4.2555	0.0045	-0.0013	14.0001	4.2555	0.0045	-0.0013	14.7414
10^6	4.2567	0.0017	-0.0002	48.8933	4.2567	0.0017	-0.0002	48.9804	4.2567	0.0017	-0.0002	47.8079

Tabulka 7.5: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho směrodatná odchylka, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 3$. Skutečná hodnota entropie $H = 4.2568$.



Obrázek 7.3: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho rozptyl, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 3$. Skutečná hodnota entropie $H = 4.2568$.



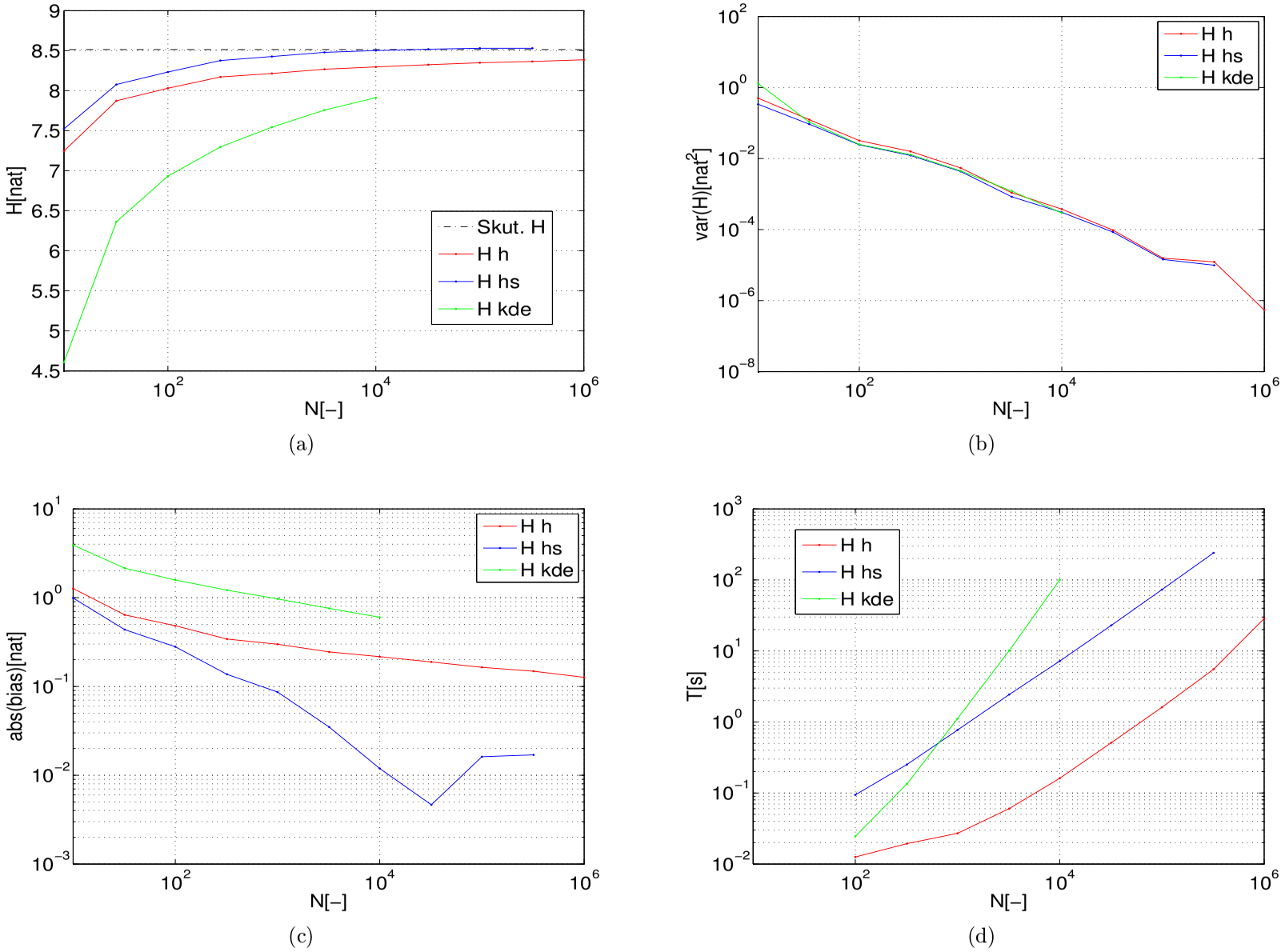
Obrázek 7.4: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho rozptyl, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 3$. Skutečná hodnota entropie $H = 4.2568$.

	$\hat{H}_h$				$\hat{H}_{hs}$				$\hat{H}_{kde}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	7.2466	0.7071	-1.2670		7.5231	0.5823	-0.9905		4.6083	1.1395	-3.9053	
$3.2 * 10^1$	7.8736	0.3533	-0.6400		8.0773	0.3048	-0.4363		6.3628	0.3268	-2.1508	
10^2	8.0309	0.1789	-0.4828	0.0125	8.2336	0.1562	-0.2801	0.0942	6.9329	0.1583	-1.5807	0.0245
$3.2 * 10^2$	8.1715	0.1260	-0.3422	0.0194	8.3765	0.1107	-0.1371	0.2519	7.2980	0.1136	-1.2157	0.1343
10^3	8.2147	0.0739	-0.2989	0.0270	8.4271	0.0659	-0.0865	0.7709	7.5443	0.0672	-0.9693	1.1156
$3.2 * 10^3$	8.2685	0.0330	-0.2451	0.0602	8.4788	0.0290	-0.0349	2.4346	7.7569	0.0348	-0.7567	10.4066
10^4	8.2973	0.0194	-0.2164	0.1610	8.5017	0.0175	-0.0119	7.1844	7.9130	0.0171	-0.6007	99.9724
$3.2 * 10^4$	8.3249	0.0098	-0.1888	0.5112	8.5183	0.0092	0.0046	22.9543				
10^5	8.3495	0.0040	-0.1642	1.6072	8.5298	0.0038	0.0162	72.8999				
$3.2 * 10^5$	8.3651	0.0035	-0.1486	5.5250	8.5306	0.0031	0.0170	240.3516				
10^6	8.3869	0.0007	-0.1267	28.5823								

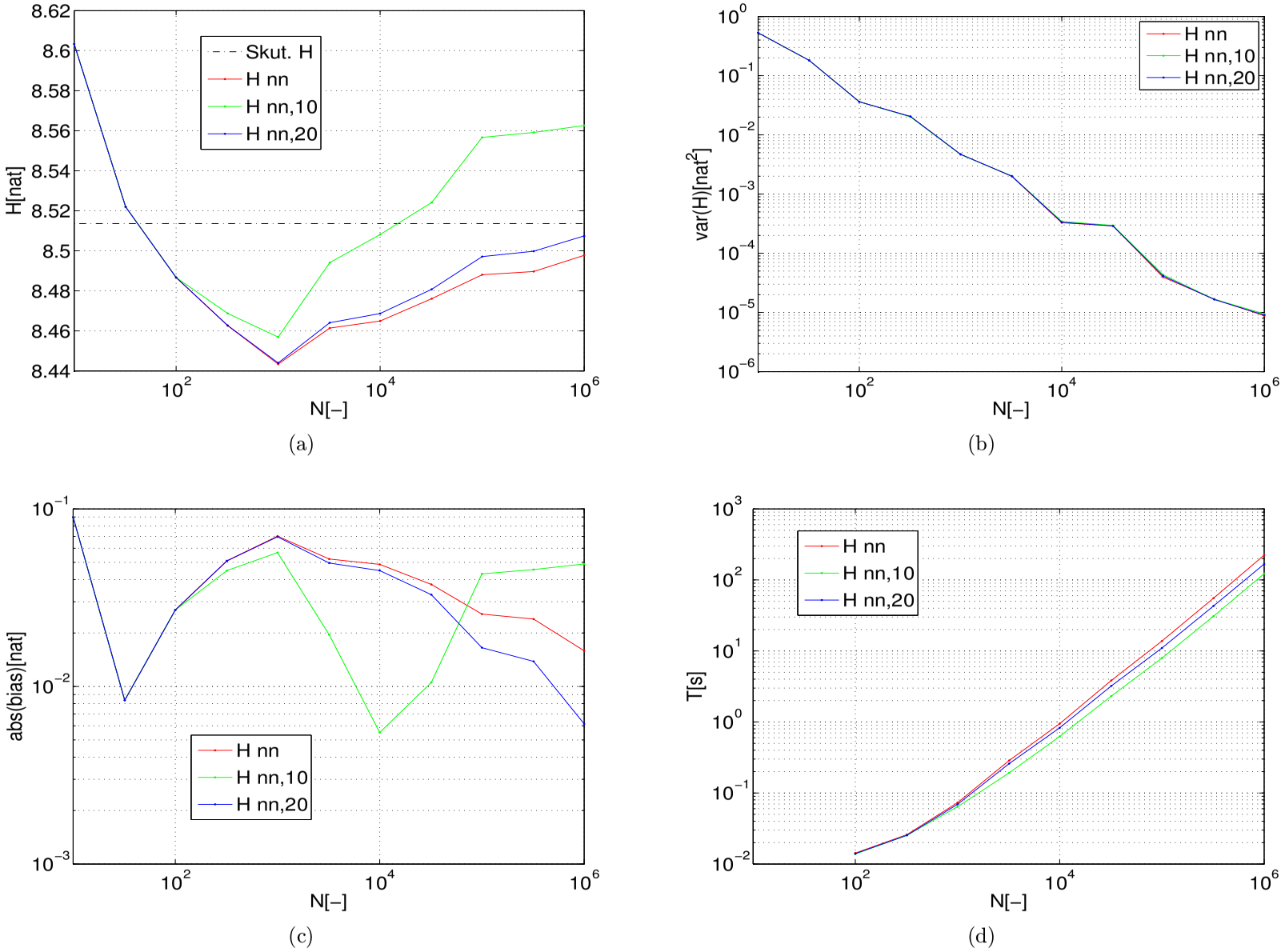
Tabulka 7.6: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho směrodatná odchylka, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 6$. Skutečná hodnota entropie $H = 8.5136$.

	$\hat{H}_{nn}$				$\hat{H}_{nn,10}$				$\hat{H}_{nn,20}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	8.6033	0.7280	0.0897		8.6033	0.7280	0.0897		8.6033	0.7280	0.0897	
$3.2 * 10^1$	8.5220	0.4259	0.0084		8.5220	0.4259	0.0084		8.5220	0.4259	0.0084	
10^2	8.4867	0.1903	-0.0269	0.0143	8.4867	0.1903	-0.0269	0.0138	8.4867	0.1903	-0.0269	0.0140
$3.2 * 10^2$	8.4626	0.1429	-0.0510	0.0260	8.4687	0.1418	-0.0449	0.0254	8.4627	0.1430	-0.0509	0.0254
10^3	8.4434	0.0685	-0.0702	0.0732	8.4569	0.0681	-0.0567	0.0639	8.4440	0.0686	-0.0697	0.0692
$3.2 * 10^3$	8.4614	0.0447	-0.0522	0.2869	8.4941	0.0450	-0.0196	0.1920	8.4641	0.0448	-0.0496	0.2590
10^4	8.4649	0.0181	-0.0487	0.9450	8.5081	0.0186	-0.0055	0.6243	8.4687	0.0183	-0.0449	0.8280
$3.2 * 10^4$	8.4761	0.0171	-0.0376	3.8455	8.5242	0.0172	0.0105	2.3153	8.4808	0.0170	-0.0329	3.2205
10^5	8.4881	0.0063	-0.0256	13.7850	8.5567	0.0066	0.0430	7.9259	8.4971	0.0064	-0.0165	10.9819
$3.2 * 10^5$	8.4897	0.0041	-0.0240	55.3715	8.5591	0.0041	0.0455	30.8125	8.4998	0.0041	-0.0138	43.0208
10^6	8.4978	0.0030	-0.0159	223.3196	8.5626	0.0031	0.0490	122.2103	8.5075	0.0030	-0.0062	167.5396

Tabulka 7.7: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho směrodatná odchylka, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 6$. Skutečná hodnota entropie $H = 8.5136$.



Obrázek 7.5: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho rozptyl, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 6$. Skutečná hodnota entropie $H = 8.5136$.



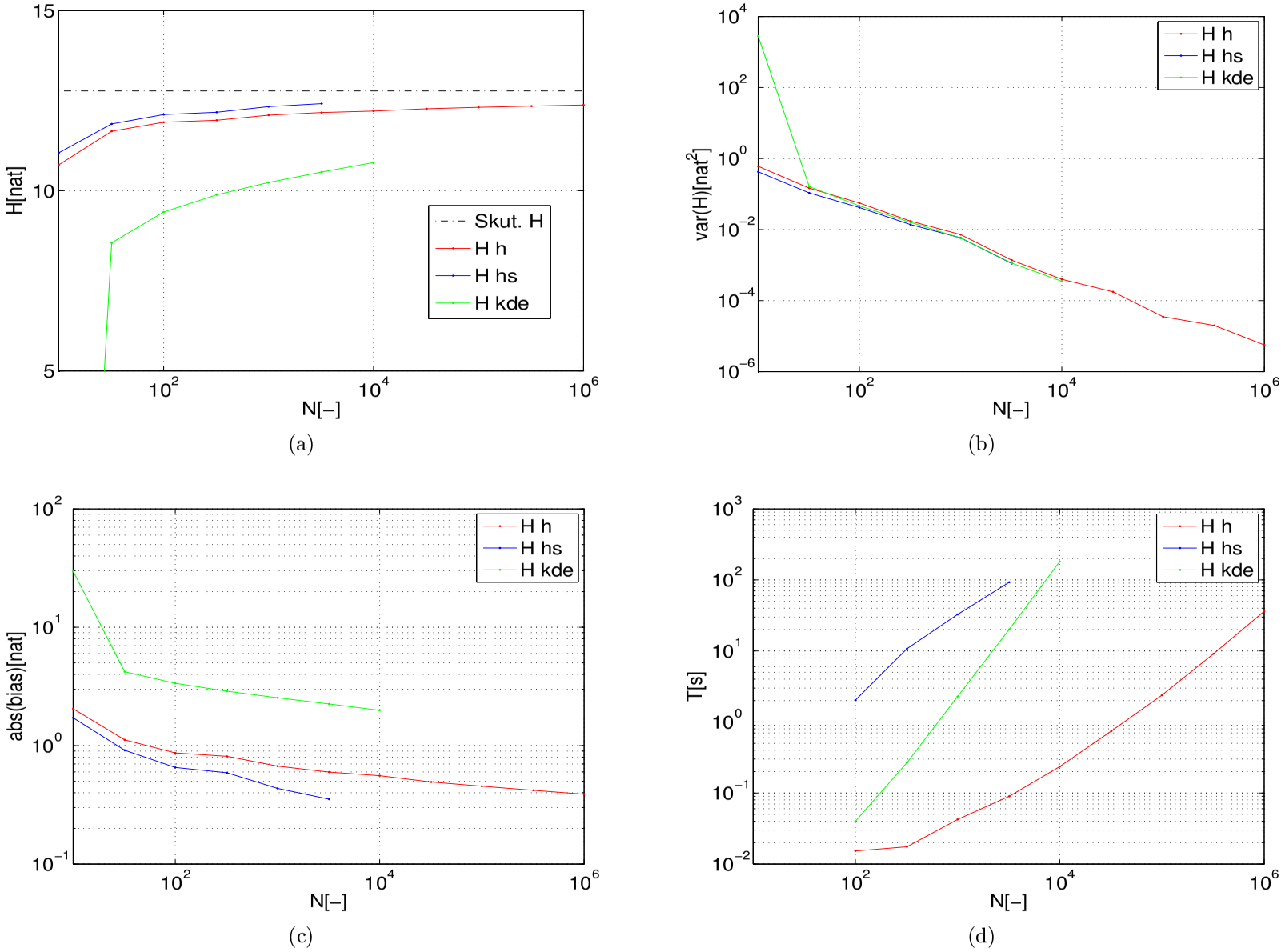
Obrázek 7.6: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho rozptyl, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 6$. Skutečná hodnota entropie $H = 8.5136$.

	$\hat{H}_h$				$\hat{H}_{hs}$				$\hat{H}_{kde}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	10.7213	0.7783	-2.0492		11.0533	0.6516	-1.7171		-16.9107	52.9021	-29.6812	
$3.2 * 10^1$	11.6517	0.3835	-1.1188		11.8561	0.3285	-0.9144		8.5564	0.4019	-4.2140	
10^2	11.9022	0.2366	-0.8682	0.0153	12.1161	0.2042	-0.6544	2.0255	9.4109	0.2155	-3.3595	0.0399
$3.2 * 10^2$	11.9574	0.1316	-0.8130	0.0175	12.1801	0.1174	-0.5904	10.7622	9.8886	0.1265	-2.8818	0.2671
10^3	12.0991	0.0850	-0.6713	0.0424	12.3348	0.0764	-0.4356	32.6467	10.2305	0.0760	-2.5400	2.2761
$3.2 * 10^3$	12.1727	0.0370	-0.5977	0.0898	12.4176	0.0332	-0.3528	92.7648	10.5212	0.0337	-2.2492	20.1267
10^4	12.2133	0.0200	-0.5572	0.2348					10.7837	0.0187	-1.9867	180.9370
$3.2 * 10^4$	12.2771	0.0133	-0.4934	0.7466								
10^5	12.3163	0.0059	-0.4541	2.3834								
$3.2 * 10^5$	12.3509	0.0045	-0.4196	9.1330								
10^6	12.3814	0.0024	-0.3890	36.1506								

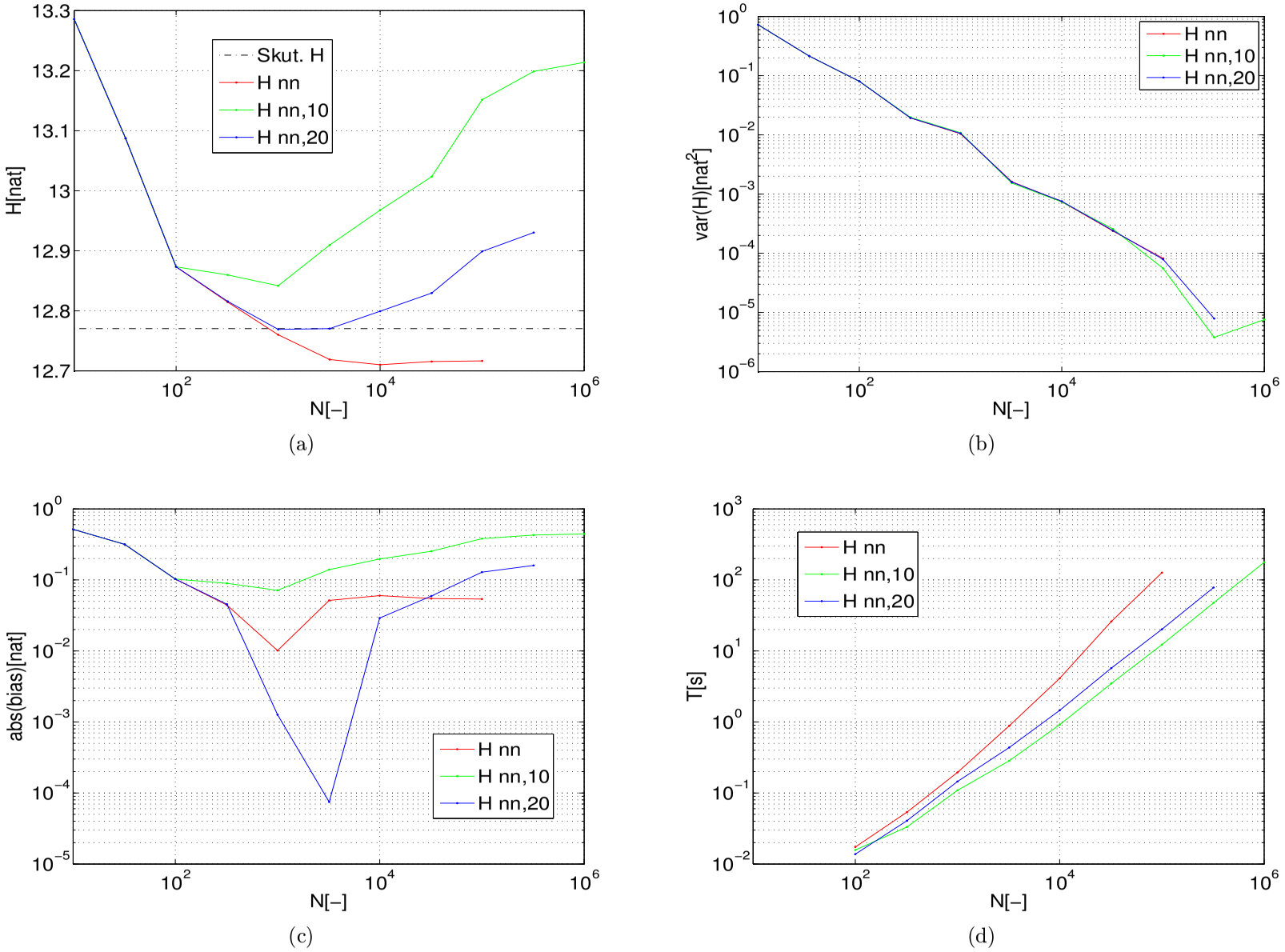
Tabulka 7.8: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho směrodatná odchylka, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 9$. Skutečná hodnota entropie $H = 12.7704$.

	$\hat{H}_{nn}$				$\hat{H}_{nn,10}$				$\hat{H}_{nn,20}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	13.2858	0.8517	0.5154		13.2858	0.8517	0.5154		13.2858	0.8517	0.5154	
$3.2 * 10^1$	13.0872	0.4635	0.3167		13.0872	0.4635	0.3167		13.0872	0.4635	0.3167	
10^2	12.8733	0.2833	0.1028	0.0174	12.8733	0.2833	0.1028	0.0157	12.8733	0.2833	0.1028	0.0138
$3.2 * 10^2$	12.8147	0.1390	0.0442	0.0537	12.8599	0.1408	0.0894	0.0332	12.8158	0.1392	0.0454	0.0407
10^3	12.7603	0.1021	-0.0101	0.1954	12.8418	0.1041	0.0713	0.1091	12.7692	0.1028	-0.0013	0.1450
$3.2 * 10^3$	12.7190	0.0402	-0.0514	0.8797	12.9097	0.0391	0.1393	0.2837	12.7705	0.0399	0.0001	0.4363
10^4	12.7103	0.0272	-0.0601	4.1125	12.9677	0.0271	0.1972	0.9144	12.7995	0.0275	0.0290	1.4559
$3.2 * 10^4$	12.7157	0.0154	-0.0547	25.9718	13.0235	0.0160	0.2530	3.4790	12.8297	0.0155	0.0592	5.7302
10^5	12.7167	0.0090	-0.0538	126.2669	13.1516	0.0075	0.3811	12.2616	12.8991	0.0089	0.1286	20.1927
$3.2 * 10^5$					13.1987	0.0019	0.4283	47.6008	12.9303	0.0028	0.1599	78.0887
10^6					13.2138	0.0028	0.4433	176.6311				

Tabulka 7.9: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho směrodatná odchylka, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 9$. Skutečná hodnota entropie $H = 12.7704$.



Obrázek 7.7: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho rozptyl, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 9$. Skutečná hodnota entropie $H = 12.7704$.



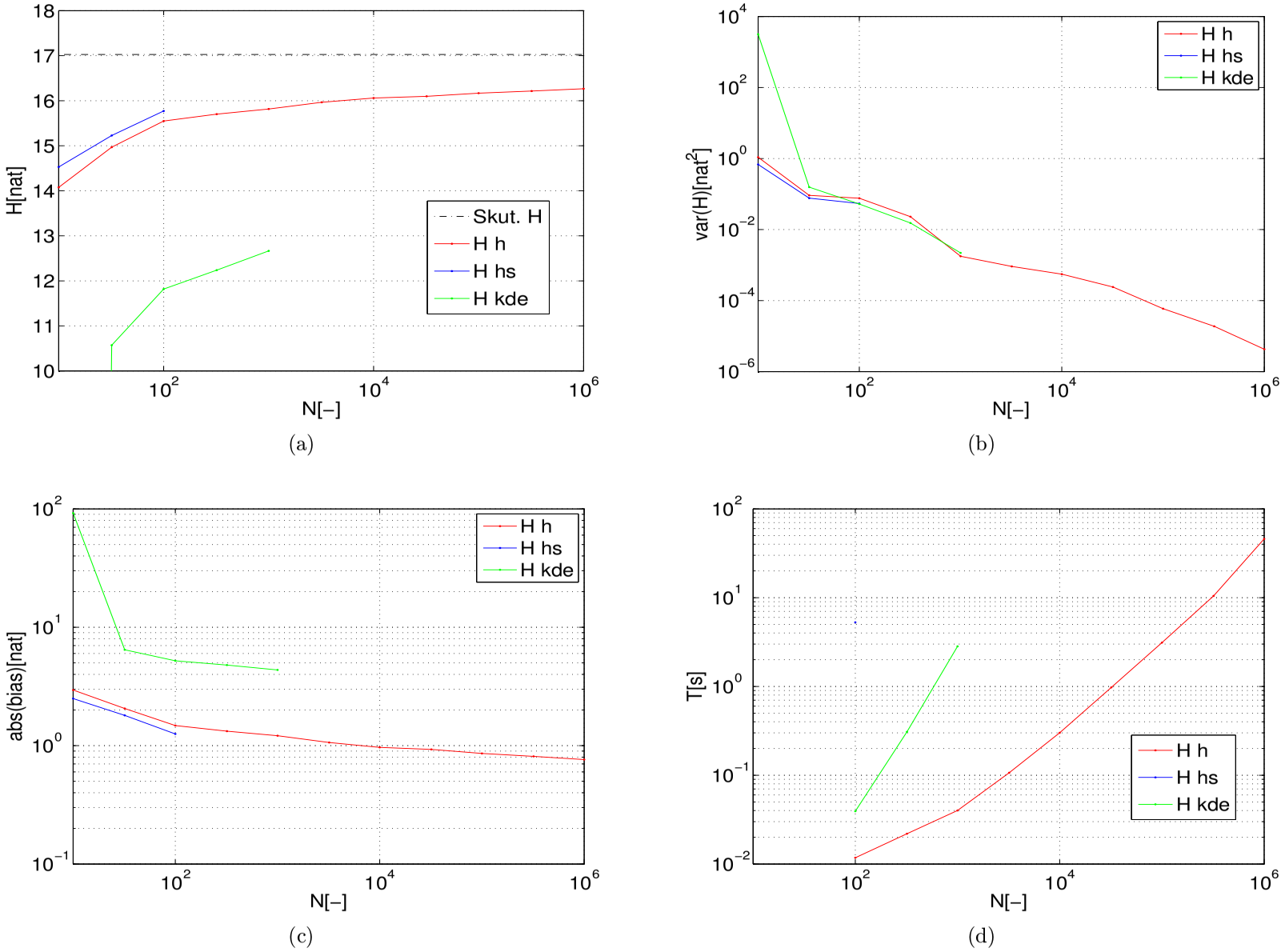
Obrázek 7.8: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho rozptyl, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 9$. Skutečná hodnota entropie $H = 12.7704$.

	$\hat{H}_h$				$\hat{H}_{hs}$				$\hat{H}_{kde}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	14.0771	1.0426	-2.9502		14.5287	0.8257	-2.4986		-75.0625	56.1913	-92.0898	
$3.2 * 10^1$	14.9680	0.3034	-2.0592		15.2266	0.2769	-1.8007		10.5709	0.3958	-6.4564	
10^2	15.5488	0.2766	-1.4785	0.0118	15.7727	0.2328	-1.2546	5.2797	11.8169	0.2289	-5.2104	0.0393
$3.2 * 10^2$	15.7026	0.1518	-1.3247	0.0219					12.2380	0.1239	-4.7892	0.3081
10^3	15.8149	0.0421	-1.2123	0.0401					12.6657	0.0470	-4.3616	2.8268
$3.2 * 10^3$	15.9645	0.0303	-1.0627	0.1066								
10^4	16.0605	0.0236	-0.9668	0.3009								
$3.2 * 10^4$	16.0974	0.0155	-0.9298	0.9756								
10^5	16.1682	0.0077	-0.8591	3.1173								
$3.2 * 10^5$	16.2156	0.0043	-0.8116	10.4523								
10^6	16.2650	0.0021	-0.7623	45.8711								

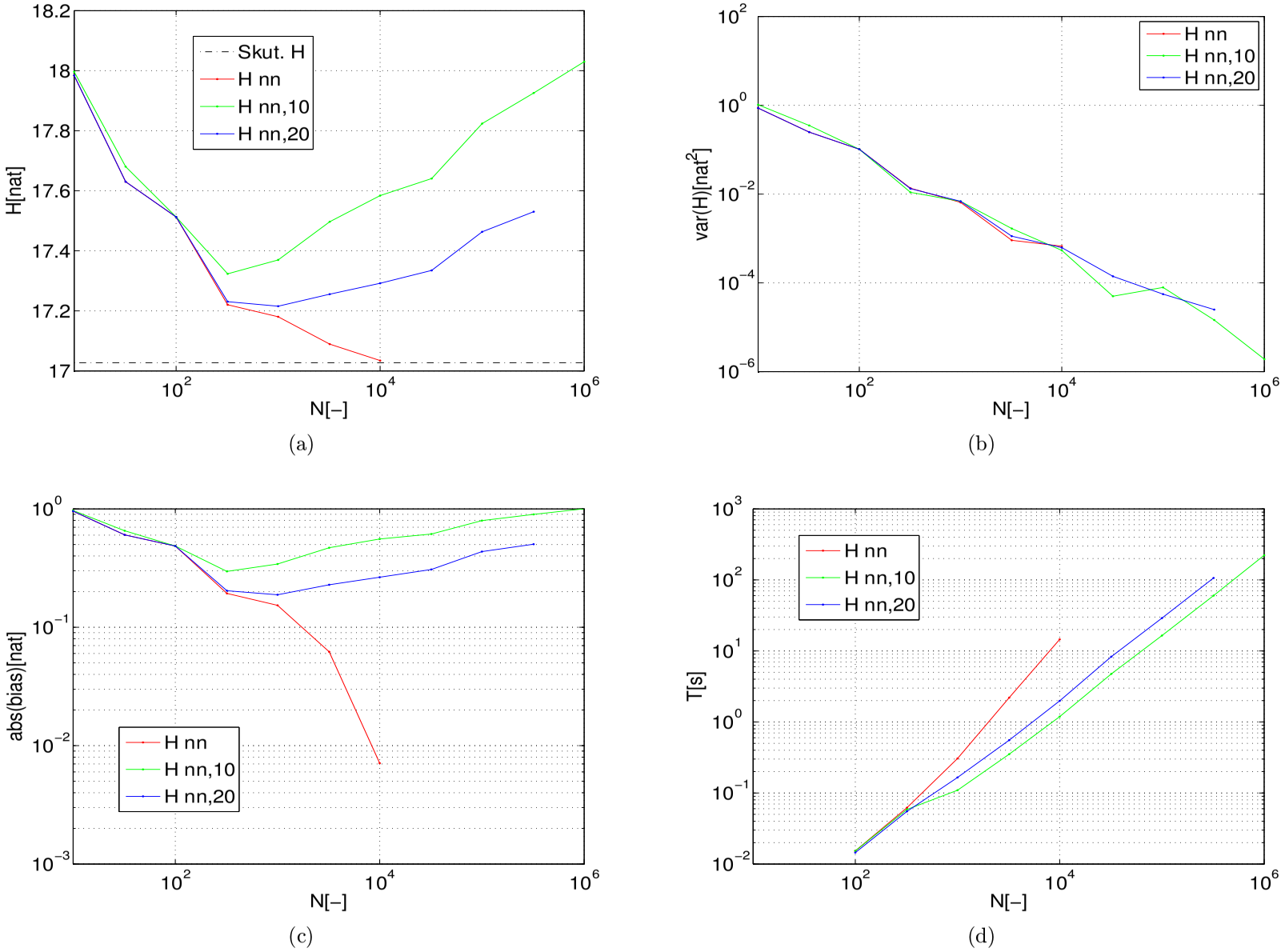
Tabulka 7.10: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho směrodatná odchylka, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 12$. Skutečná hodnota entropie $H = 17.0272$.

	$\hat{H}_{nn}$				$\hat{H}_{nn,10}$				$\hat{H}_{nn,20}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	17.9840	0.9289	0.9567		17.9962	1.0153	0.9690		17.9840	0.9289	0.9567	
$3.2 * 10^1$	17.6304	0.4987	0.6031		17.6805	0.5924	0.6532		17.6304	0.4987	0.6031	
10^2	17.5127	0.3198	0.4854	0.0153	17.5127	0.3198	0.4854	0.0153	17.5127	0.3198	0.4854	0.0145
$3.2 * 10^2$	17.2203	0.1164	0.1931	0.0620	17.3234	0.1046	0.2962	0.0586	17.2307	0.1152	0.2034	0.0551
10^3	17.1804	0.0805	0.1531	0.3062	17.3696	0.0829	0.3423	0.1093	17.2155	0.0829	0.1882	0.1656
$3.2 * 10^3$	17.0894	0.0301	0.0621	2.2008	17.4967	0.0409	0.4695	0.3515	17.2557	0.0335	0.2284	0.5541
10^4	17.0344	0.0259	0.0071	14.5424	17.5840	0.0232	0.5568	1.1875	17.2920	0.0250	0.2647	1.9833
$3.2 * 10^4$					17.6408	0.0071	0.6135	4.7447	17.3348	0.0118	0.3075	8.3075
10^5					17.8237	0.0089	0.7964	16.4662	17.4632	0.0075	0.4360	28.9447
$3.2 * 10^5$					17.9256	0.0038	0.8984	60.0518	17.5303	0.0050	0.5031	106.4001
10^6					18.0299	0.0014	1.0026	221.8968				

Tabulka 7.11: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho směrodatná odchylka, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 12$. Skutečná hodnota entropie $H = 17.0272$.



Obrázek 7.9: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho rozptyl, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 12$. Skutečná hodnota entropie $H = 17.0272$.



Obrázek 7.10: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho rozptyl, bias a čas běhu pro $N(\mathbf{o}, I_d)$ s $d = 12$. Skutečná hodnota entropie $H = 17.0272$.

7.2.2 Odhad α-entropie normálního rozdělení

V následujícím experimentu byla vyhodnocena přesnost a rychlost estimátoru α-entropie $\hat{H}_\alpha$ pro dimenzionality $d = \{1, 3, 6, 9, 12\}$. Pro teoretickou hodnotu α-entropie normálního rozdělení s kovarianční maticí Σ platí [23]

$$H_\alpha = H + \frac{d}{2}\left(\frac{\ln \alpha}{1-\alpha}\right), \tag{7.3}$$

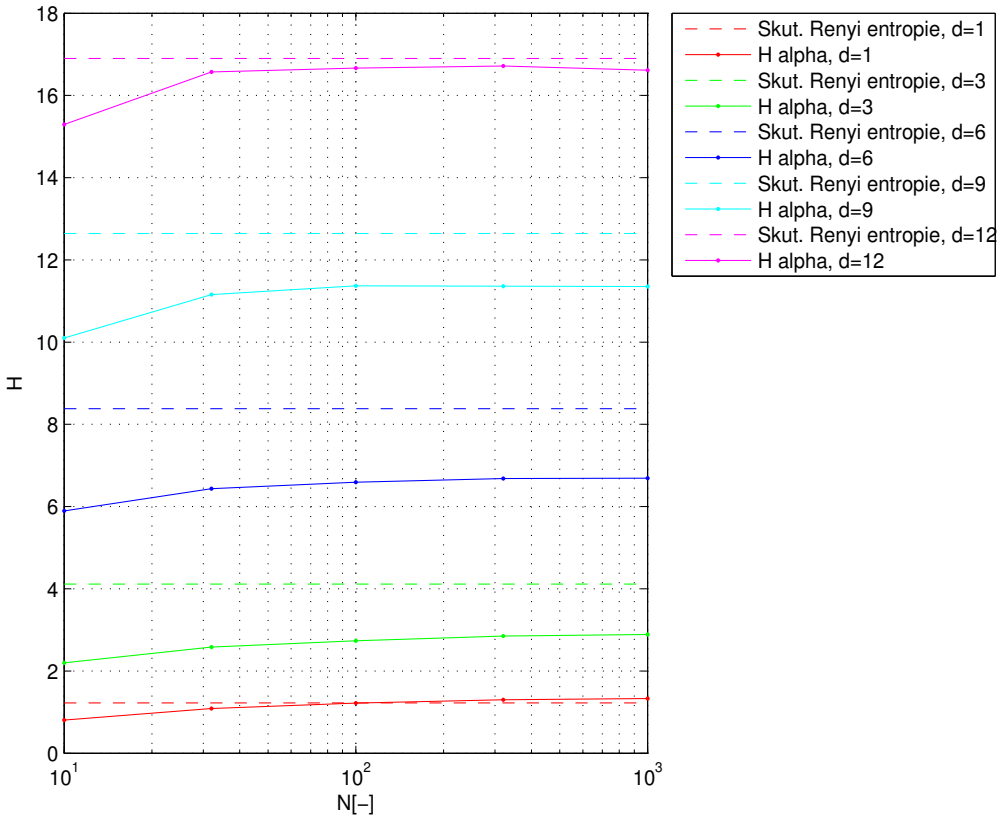
kde H značí hodnotu Shannonovy entropie rozdělení.

Střední kvadratická chyba estimátoru α-entropie je srovnána s chybami estimátorů Shannonovy entropie v části 7.2.3.

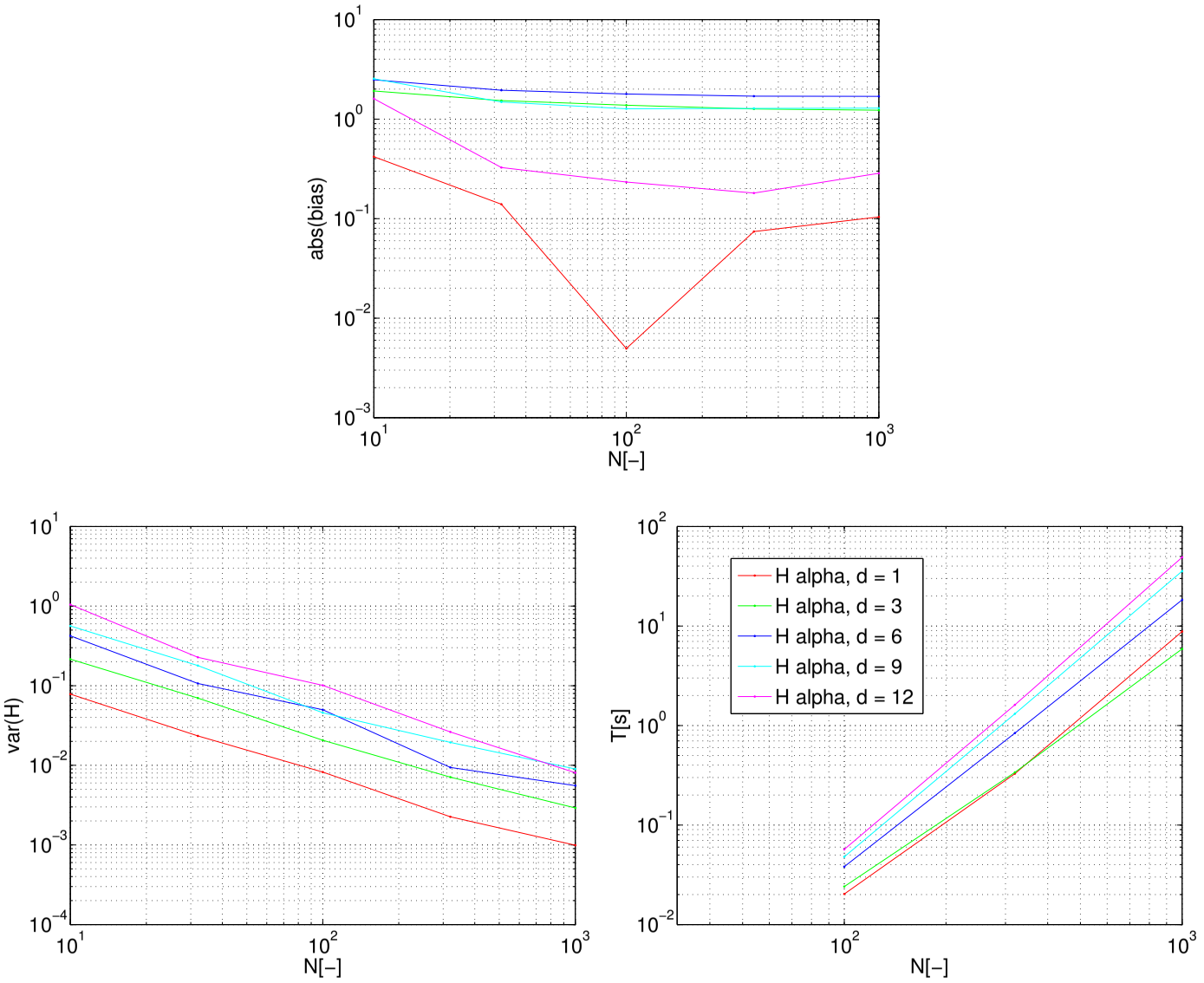
	$\hat{H}_\alpha, d=1$				$\hat{H}_\alpha, d=3$				$\hat{H}_\alpha, d=6$			
N[-]	$\overline{\hat{H}}$	σ	bias	T[s]	$\overline{\hat{H}}$	σ	bias	T[s]	$\overline{\hat{H}}$	σ	bias	T[s]
10^1	0.8072	0.2801	-0.6117		2.2010	0.4643	-2.0558		5.8960	0.6503	-2.6176	
$3.2 * 10^1$	1.0863	0.1531	-0.3326		2.5813	0.2643	-1.6755		6.4329	0.3265	-2.0807	
10^2	1.2208	0.0907	-0.1981	0.0203	2.7367	0.1438	-1.5201	0.0241	6.5935	0.2235	-1.9201	0.0382
$3.2 * 10^2$	1.2998	0.0475	-0.1191	0.3284	2.8517	0.0843	-1.4051	0.3385	6.6802	0.0970	-1.8334	0.8384
10^3	1.3296	0.0315	-0.0894	8.0039	2.8885	0.0540	-1.3683	7.8862	6.6884	0.0745	-1.8252	18.2988

	$\hat{H}_\alpha, d=9$				$\hat{H}_\alpha, d=12$			
N[-]	$\overline{\hat{H}}$	σ	bias	T[s]	$\overline{\hat{H}}$	σ	bias	T[s]
10^1	10.1000	0.7506	-2.6704		15.2960	1.0223	-1.7312	
$3.2 * 10^1$	11.1551	0.4233	-1.6153		16.5730	0.4771	-0.4543	
10^2	11.3700	0.2138	-1.4004	0.0480	16.6656	0.3180	-0.3616	0.0569
$3.2 * 10^2$	11.3589	0.1394	-1.4115	1.3115	16.7180	0.1619	-0.3093	1.6125
10^3	11.3529	0.0949	-1.4175	35.5812	16.6130	0.0898	-0.4143	49.0376

Tabulka 7.12: Průměrná hodnota odhadu α-entropie estimátoru $\hat{H}_\alpha$, jeho směrodatná odchylka, bias a čas běhu pro normální rozdělení.



Obrázek 7.11: Průměrné hodnoty odhadu α-entropie pro dimenze $d = \{1, 3, 6, 9, 12\}$. Skutečné hodnoty α-entropie jsou znázorněny přerušovanou čarou.



Obrázek 7.12: Rozptyl, bias a čas běhu estimátoru α-entropie pro dimenze $d = \{1, 3, 6, 9, 12\}$.

7.2.3 Střední kvadratická chyba a doba výpočtu: srovnání estimátorů entropie a α-entropie

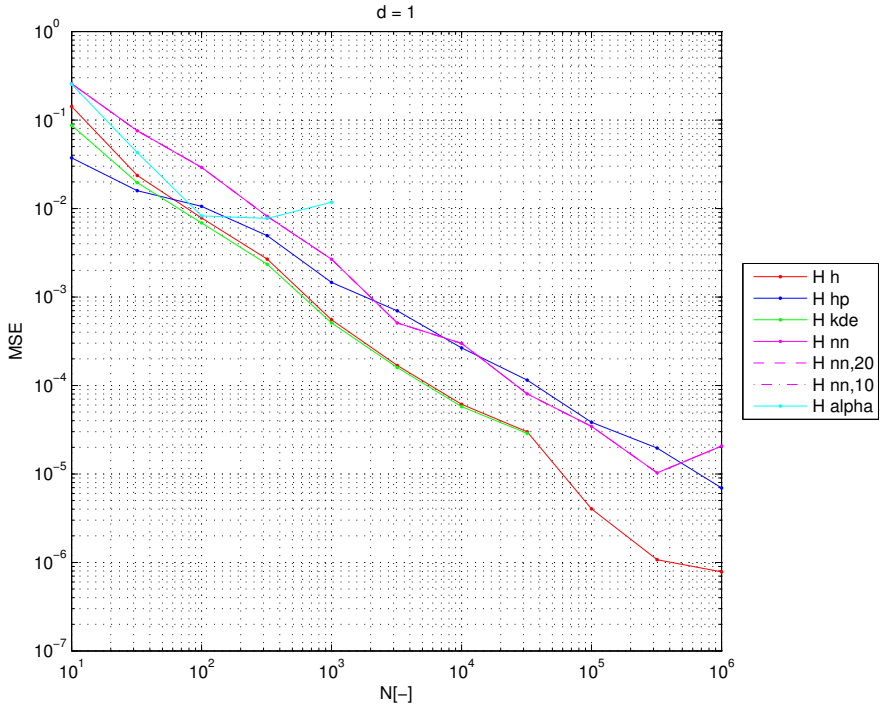
Střední kvadratická chyba estimátoru parametru θ je definována jako střední hodnota kvadrátu odchylky estimátoru

$$MSE(\hat{\theta}) = E((\hat{\theta} - \theta)^2), \tag{7.4}$$

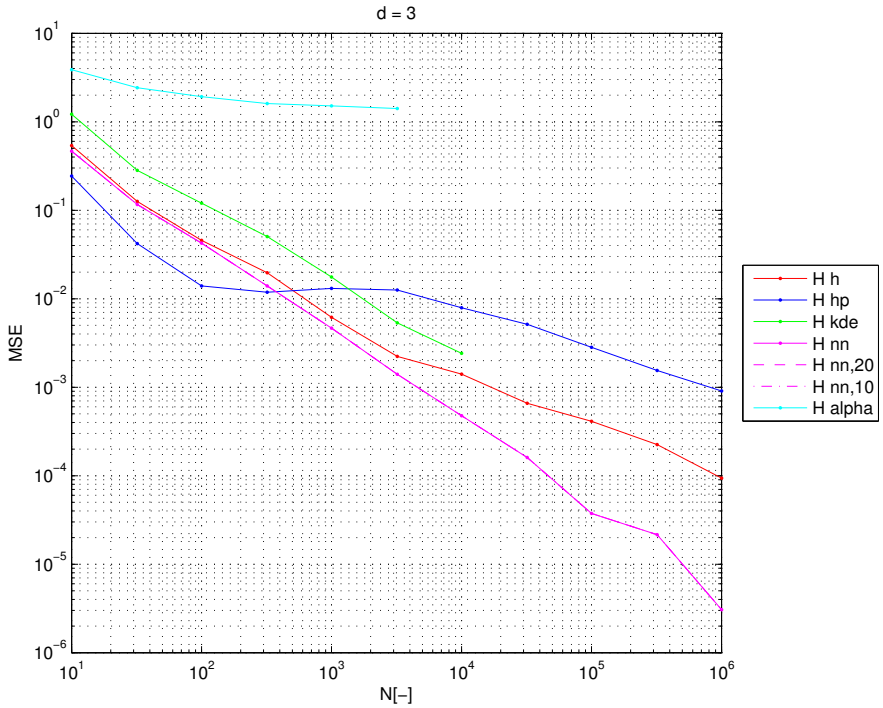
kde $\hat{\theta}$ je odhad skutečné hodnoty parametru θ. Střední kvadratická chyba může být vyjádřena pomocí rozptylu σ^2 a bias jako

$$MSE(\hat{\theta}) = \sigma^2(\hat{\theta}) + (\text{bias}(\hat{\theta}, \theta))^2. \tag{7.5}$$

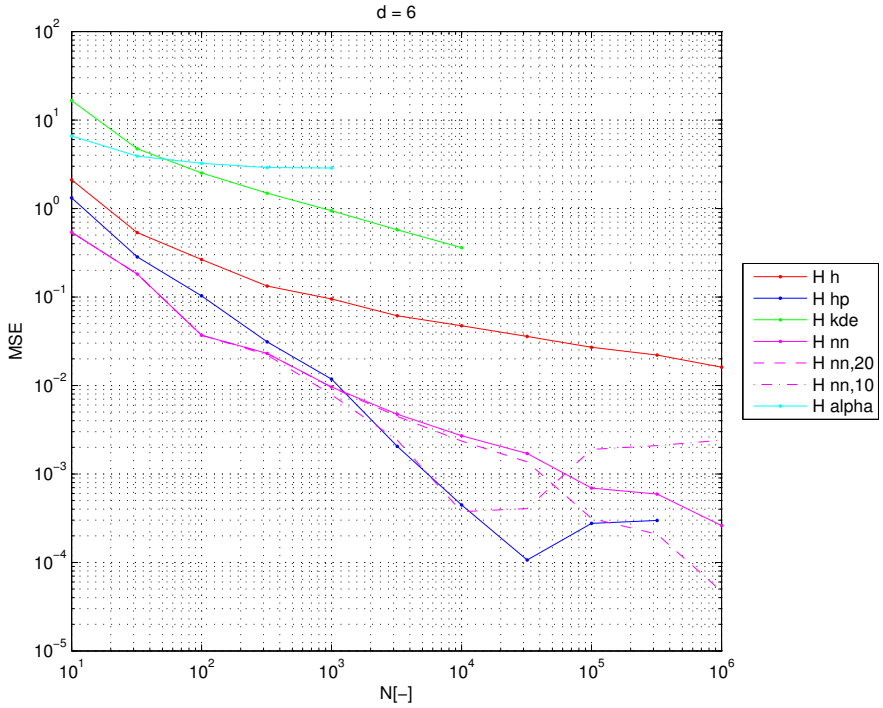
V následujících grafech jsou vyneseny závislosti střední kvadratické chyby estimátorů entropie a α-entropie v závislosti na počtu vzorků pro dimenze $d = \{1, 3, 6, 9, 12\}$ a v závislosti na dimenzi pro počty vzorků $n = \{10^3, 10^4, 10^5, 10^6\}$ z normálního rozdělení s diagonální kovarianční maticí. Užito bylo dat z předchozích experimentů.



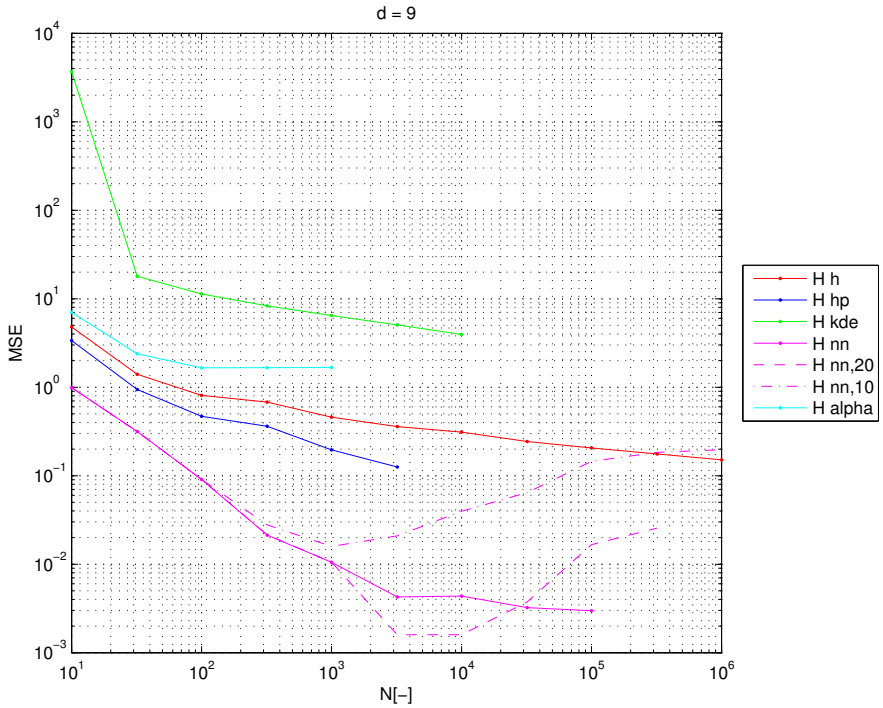
Obrázek 7.13: Střední kvadratická chyba estimátorů v závislosti na počtu vzorků pro $d = 1$.



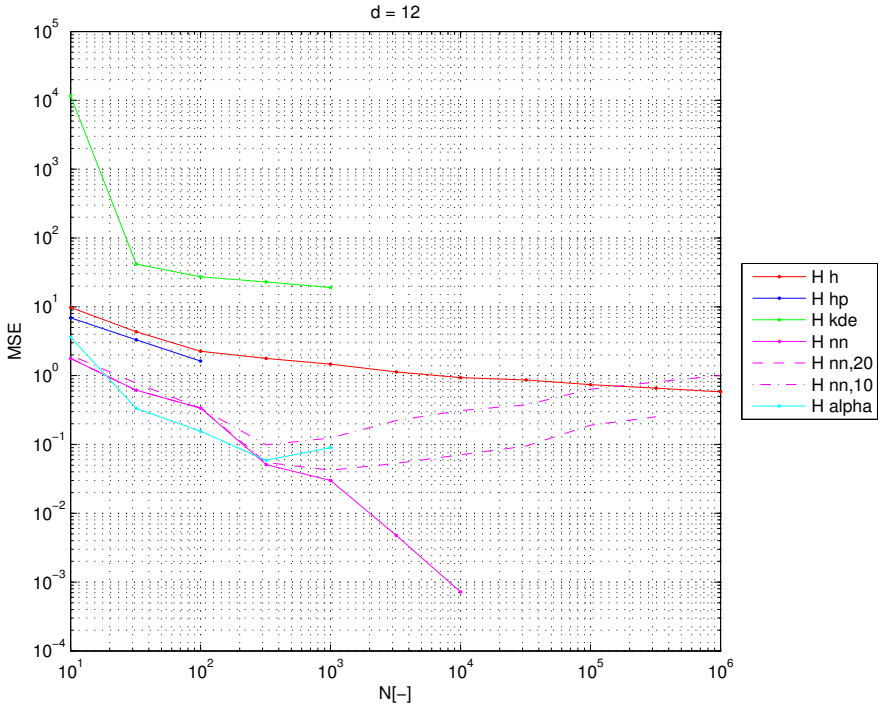
Obrázek 7.14: Střední kvadratická chyba estimátorů v závislosti na počtu vzorků pro $d = 3$.



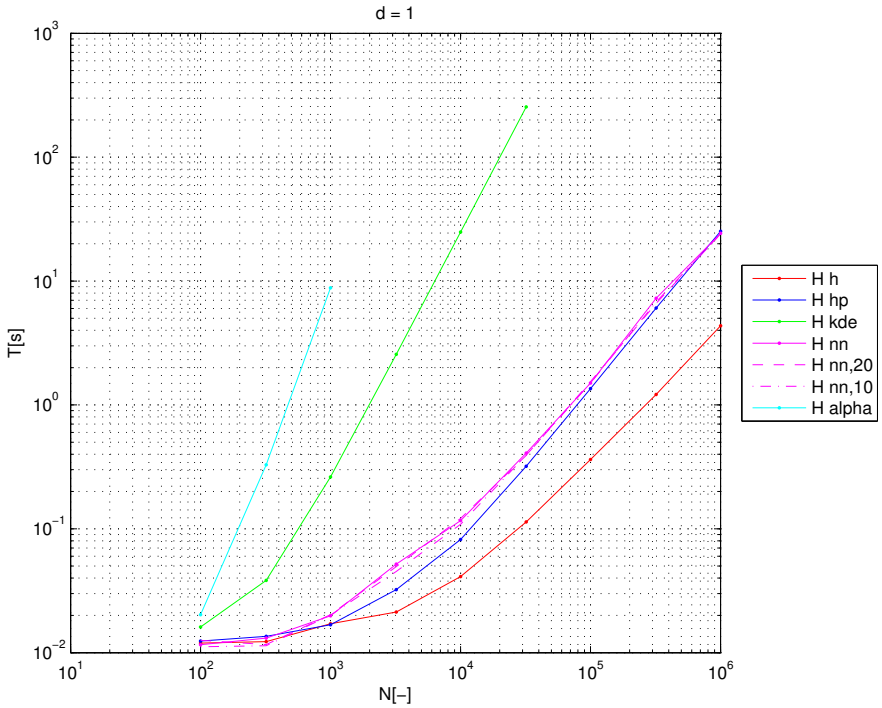
Obrázek 7.15: Střední kvadratická chyba estimátorů v závislosti na počtu vzorků pro $d = 6$.



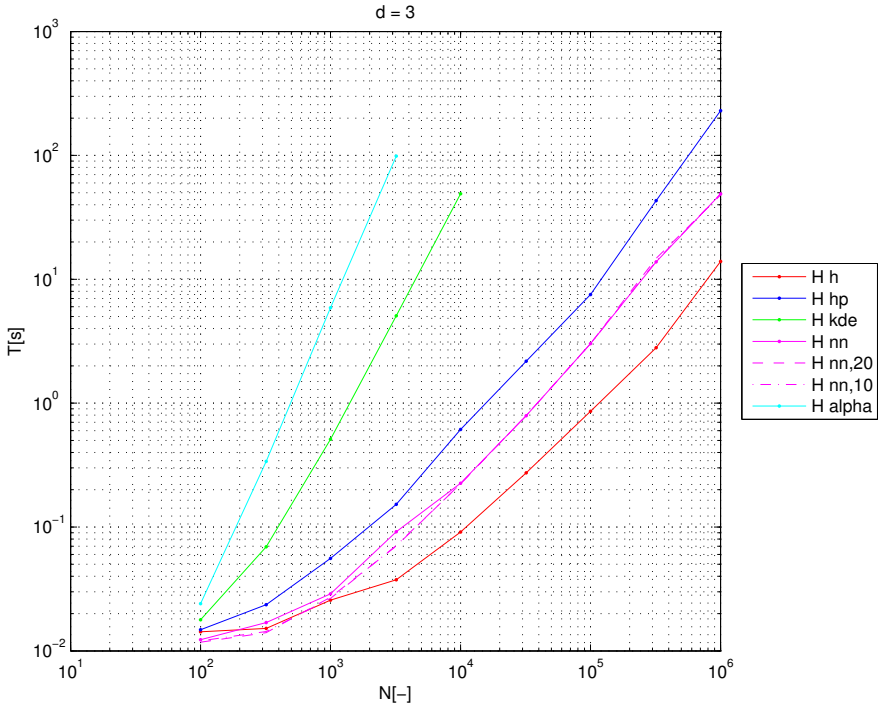
Obrázek 7.16: Střední kvadratická chyba estimátorů v závislosti na počtu vzorků pro $d = 9$.



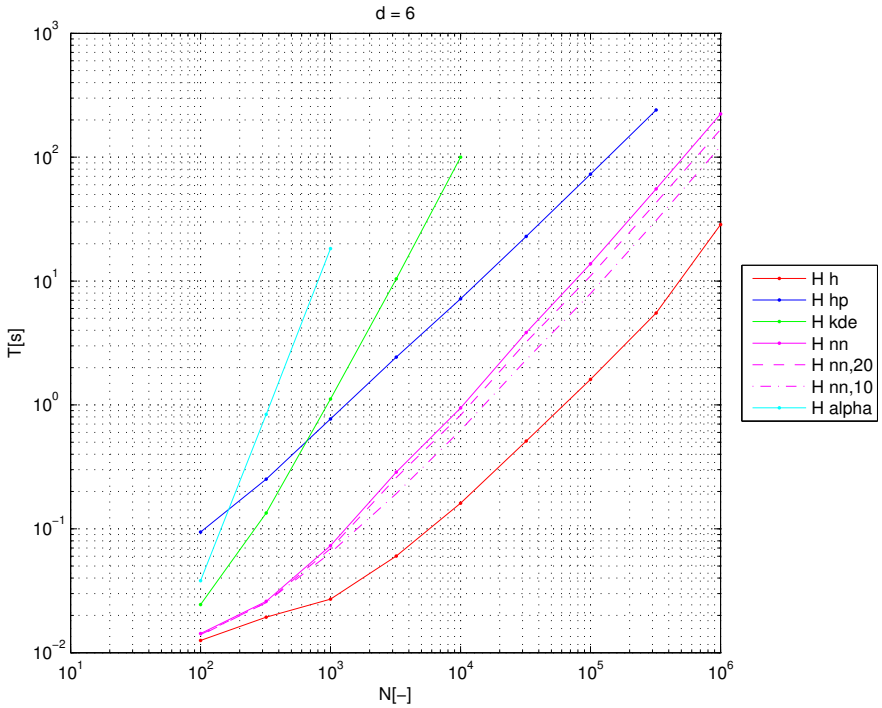
Obrázek 7.17: Střední kvadratická chyba estimátorů v závislosti na počtu vzorků pro $d = 12$.



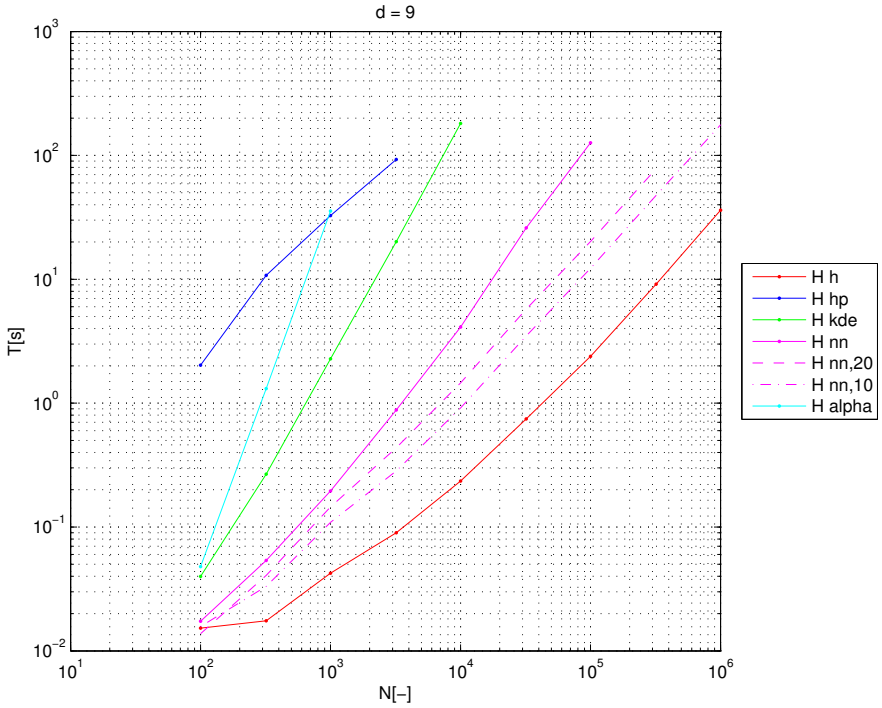
Obrázek 7.18: Průměrný čas běhu estimátorů v závislosti na počtu vzorků pro $d = 1$.



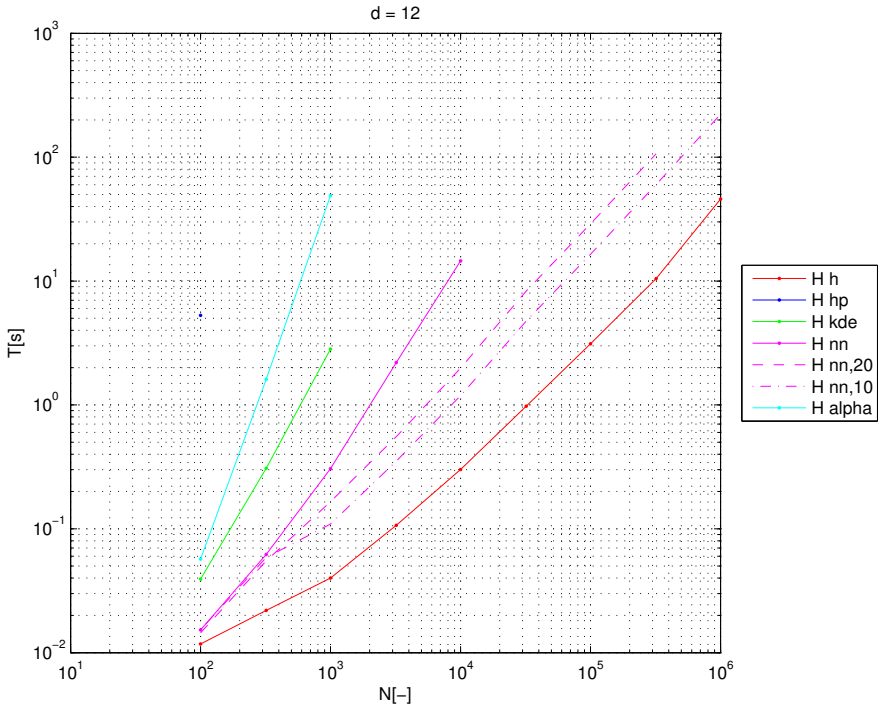
Obrázek 7.19: Průměrný čas běhu estimátorů v závislosti na počtu vzorků pro $d = 3$.



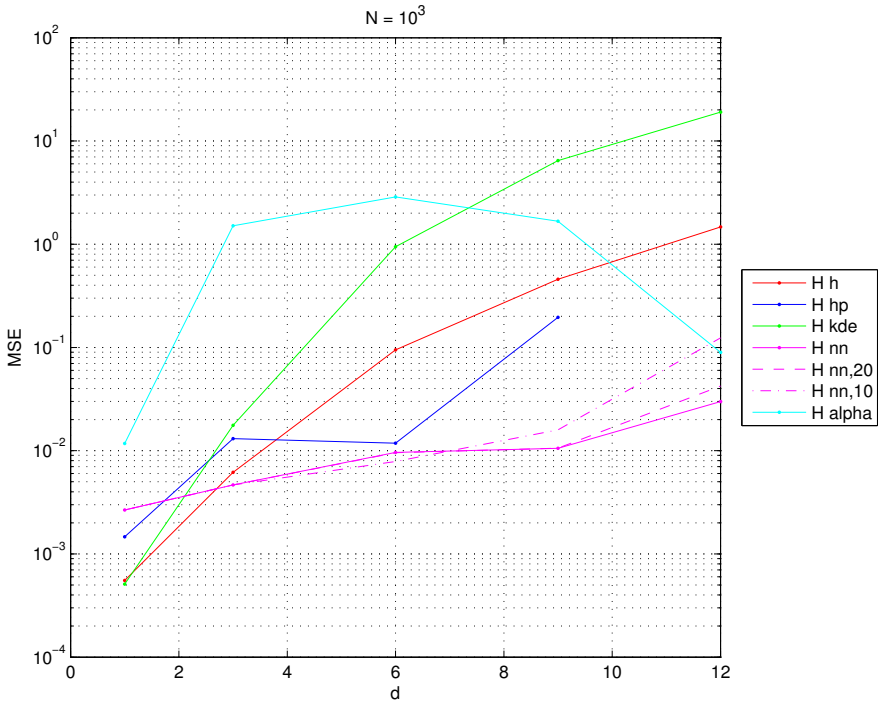
Obrázek 7.20: Průměrný čas běhu estimátorů v závislosti na počtu vzorků pro $d = 6$.



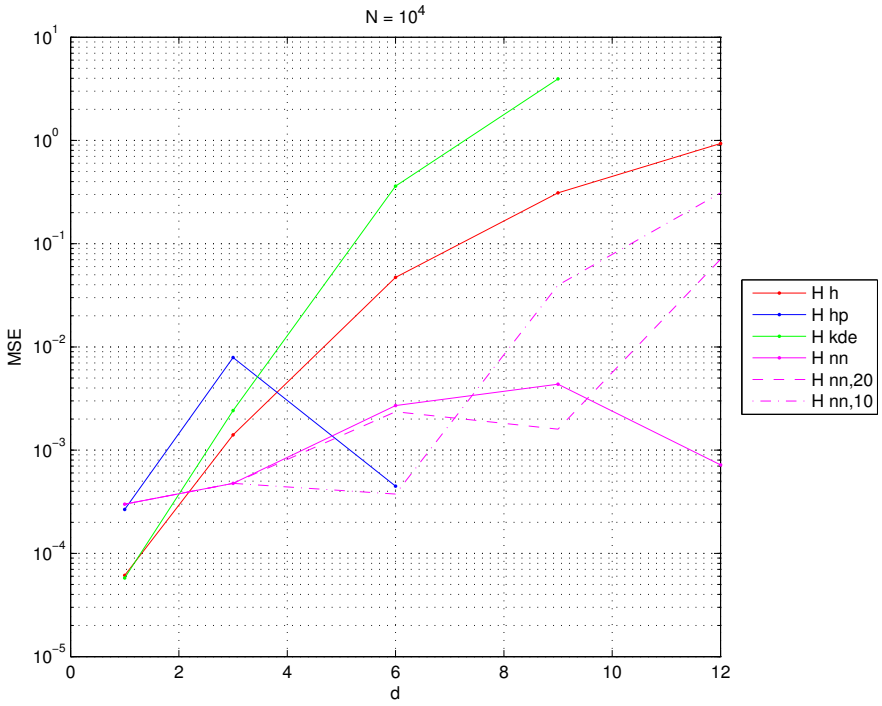
Obrázek 7.21: Průměrný čas běhu estimátorů v závislosti na počtu vzorků pro $d = 9$.



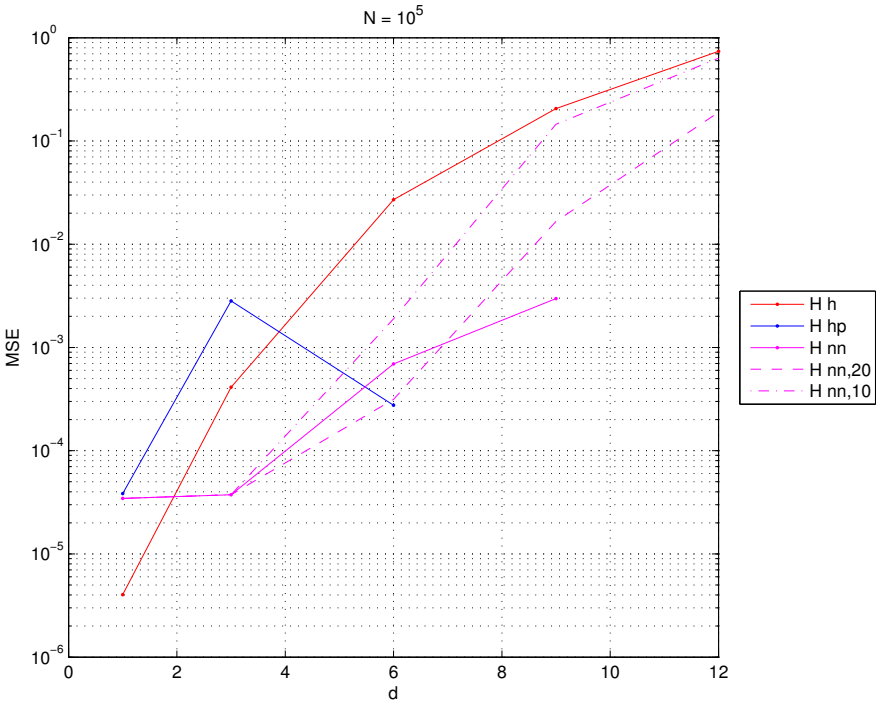
Obrázek 7.22: Průměrný čas běhu estimátorů v závislosti na počtu vzorků pro $d = 12$.



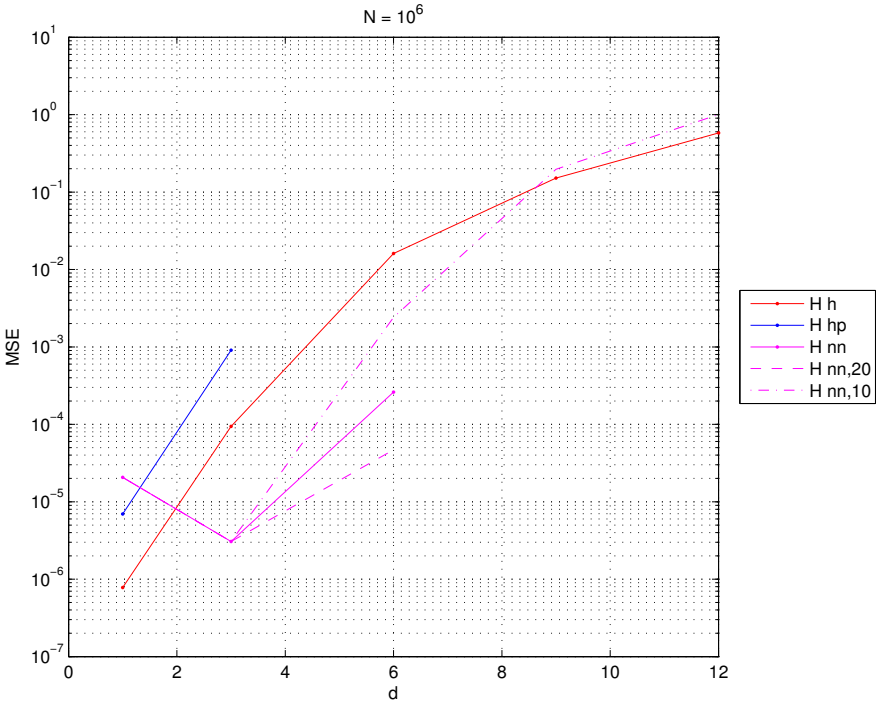
Obrázek 7.23: Střední kvadratická chyba estimátorů v závislosti na dimenzi pro $N = 10^3$.



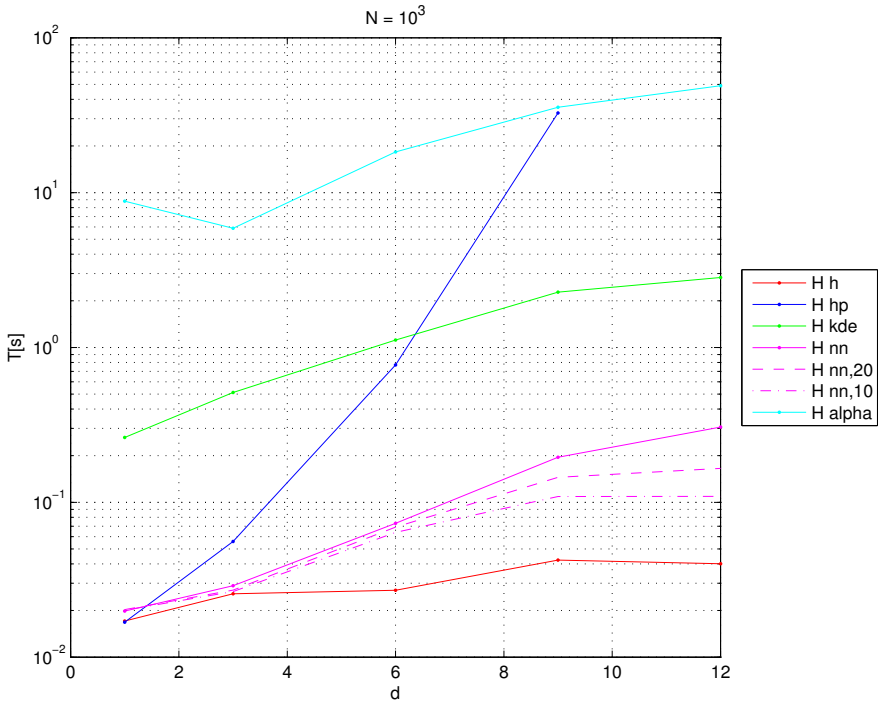
Obrázek 7.24: Střední kvadratická chyba estimátorů v závislosti na dimenzi pro $N = 10^4$.



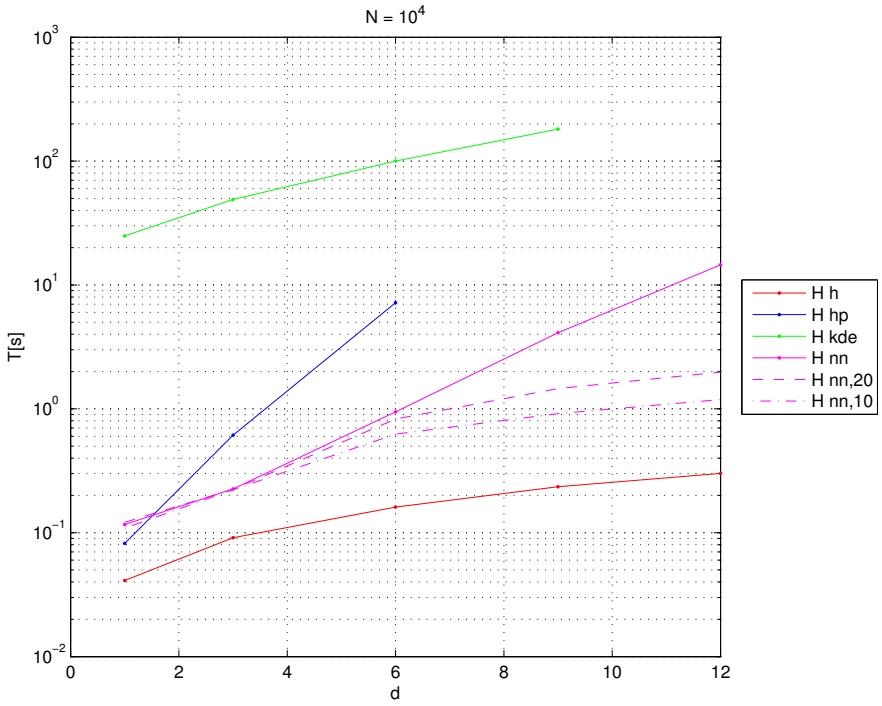
Obrázek 7.25: Střední kvadratická chyba estimátorů v závislosti na dimenzi pro $N = 10^5$.



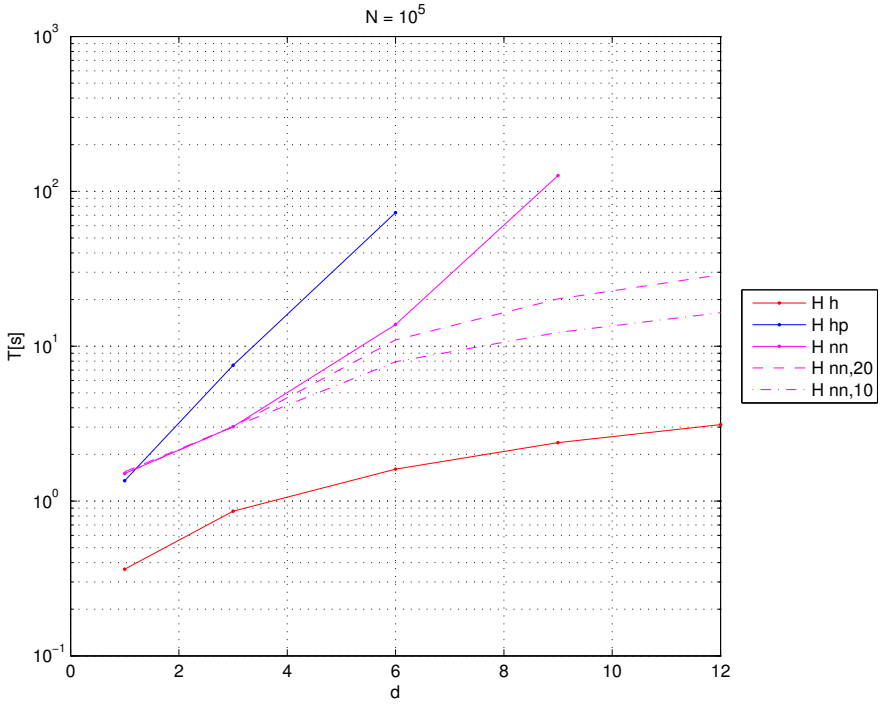
Obrázek 7.26: Střední kvadratická chyba estimátorů v závislosti na dimenzi pro $N = 10^6$.



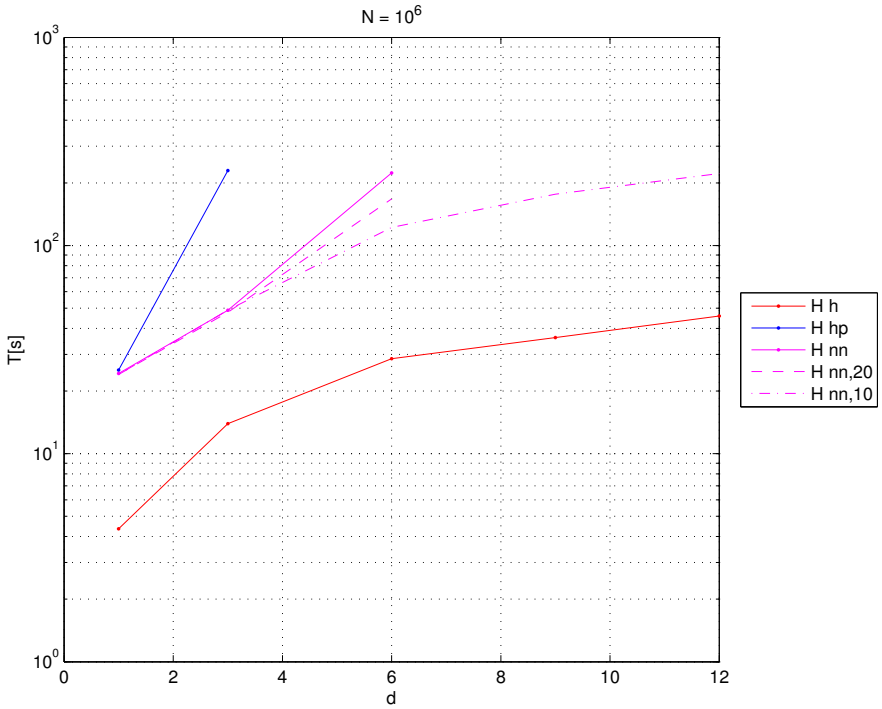
Obrázek 7.27: Průměrný čas běhu estimátorů v závislosti na dimenzi pro $N = 10^3$.



Obrázek 7.28: Průměrný čas běhu estimátorů v závislosti na dimenzi pro $N = 10^4$.



Obrázek 7.29: Průměrný čas běhu estimátorů v závislosti na dimenzi pro $N = 10^5$.



Obrázek 7.30: Průměrný čas běhu estimátorů v závislosti na dimenzi pro $N = 10^6$.

7.2.4 Odhad entropie rovnoměrného rozdělení

Estimátory $\hat{H}_h$, $\hat{H}_{hs}$, $\hat{H}_{kde}$, $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$ byly užity k estimaci entropie jedno- a vícerozměrného rovnoměrného rozdělení.

Test byl proveden pro dimenze $d = \{1, 3, 6, 9, 12\}$. Pro $n \leq 10^2$ nebyl měřen čas běhu. Testování estimátorů bylo ukončeno, pokud čas běhu překročil 400 s.

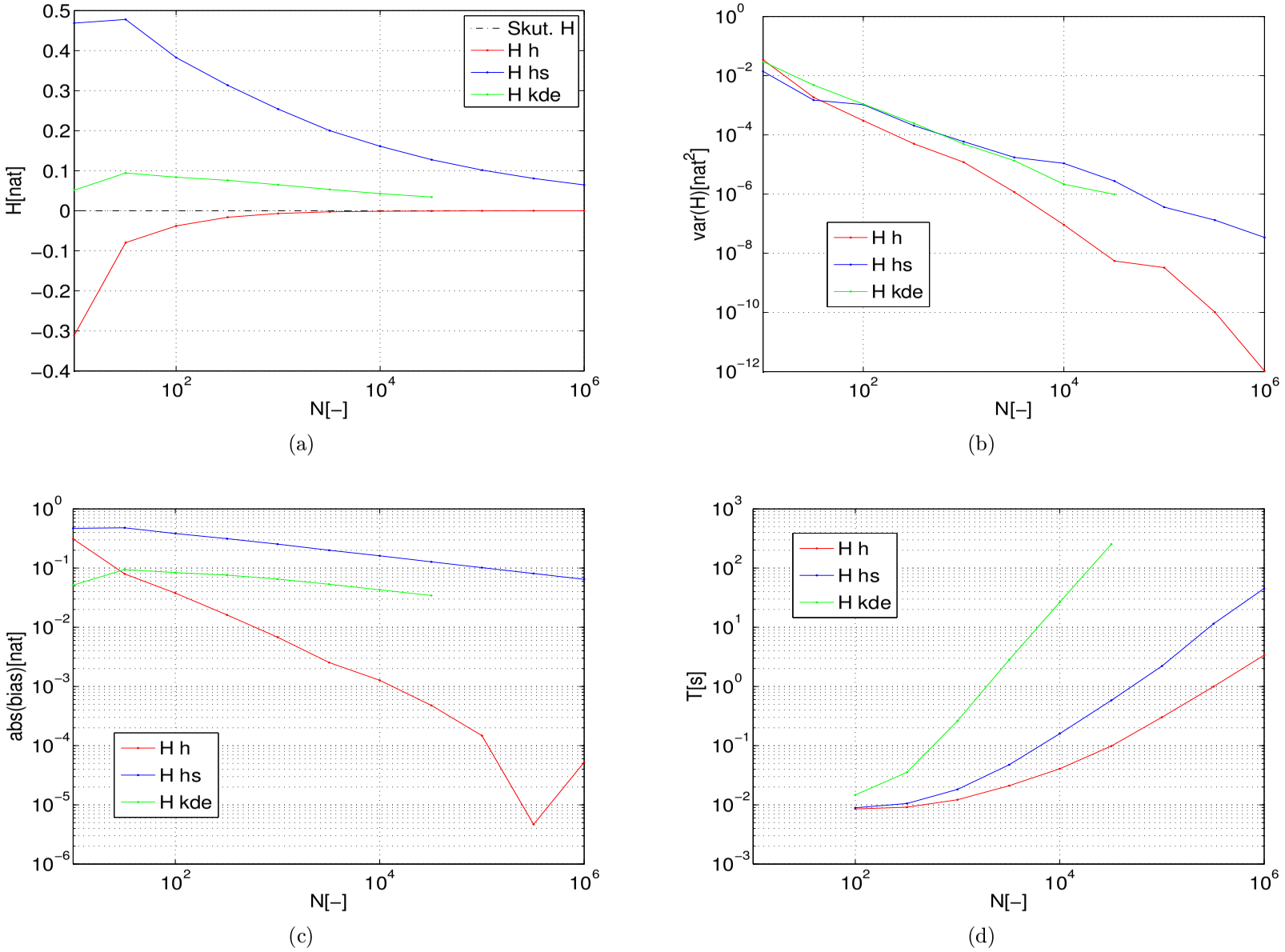
Pro účely potlačení artefaktů byly u histogramových estimátorů upraveny šíře binů na nejbližší vyšší hodnotu oproti hodnotě odhadnuté Scottovým pravidlem, jejíž celočíselný násobek odpovídal rozdílu krajních hodnot v jednotlivých dimenzích (viz. 6.2).

	$\hat{H}_h$				$\hat{H}_{hs}$				$\hat{H}_{kde}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	-0.3111	0.1863	-0.3111		0.4688	0.1180	0.4688		0.0516	0.1727	0.0516	
$3.2 * 10^1$	-0.0796	0.0430	-0.0796		0.4781	0.0387	0.4781		0.0945	0.0694	0.0945	
10^2	-0.0378	0.0174	-0.0378	0.0085	0.3828	0.0324	0.3828	0.0089	0.0840	0.0331	0.0840	0.0147
$3.2 * 10^2$	-0.0161	0.0071	-0.0161	0.0091	0.3136	0.0143	0.3136	0.0105	0.0761	0.0157	0.0761	0.0354
10^3	-0.0068	0.0035	-0.0068	0.0122	0.2540	0.0077	0.2540	0.0182	0.0651	0.0070	0.0651	0.2626
$3.2 * 10^3$	-0.0025	0.0011	-0.0025	0.0210	0.2002	0.0042	0.2002	0.0475	0.0532	0.0037	0.0532	2.8249
10^4	-0.0013	0.0003	-0.0013	0.0406	0.1613	0.0033	0.1613	0.1592	0.0429	0.0015	0.0429	26.4055
$3.2 * 10^4$	-0.0005	0.0001	-0.0005	0.0985	0.1276	0.0017	0.1276	0.5831	0.0344	0.0010	0.0344	253.4937
10^5	-0.0001	0.0001	-0.0001	0.3021	0.1016	0.0006	0.1016	2.2095				
$3.2 * 10^5$	-0.0000	0.0000	-0.0000	0.9956	0.0808	0.0004	0.0808	11.4707				
10^6	0.0001	0.0000	0.0001	3.3820	0.0646	0.0002	0.0646	45.2546				

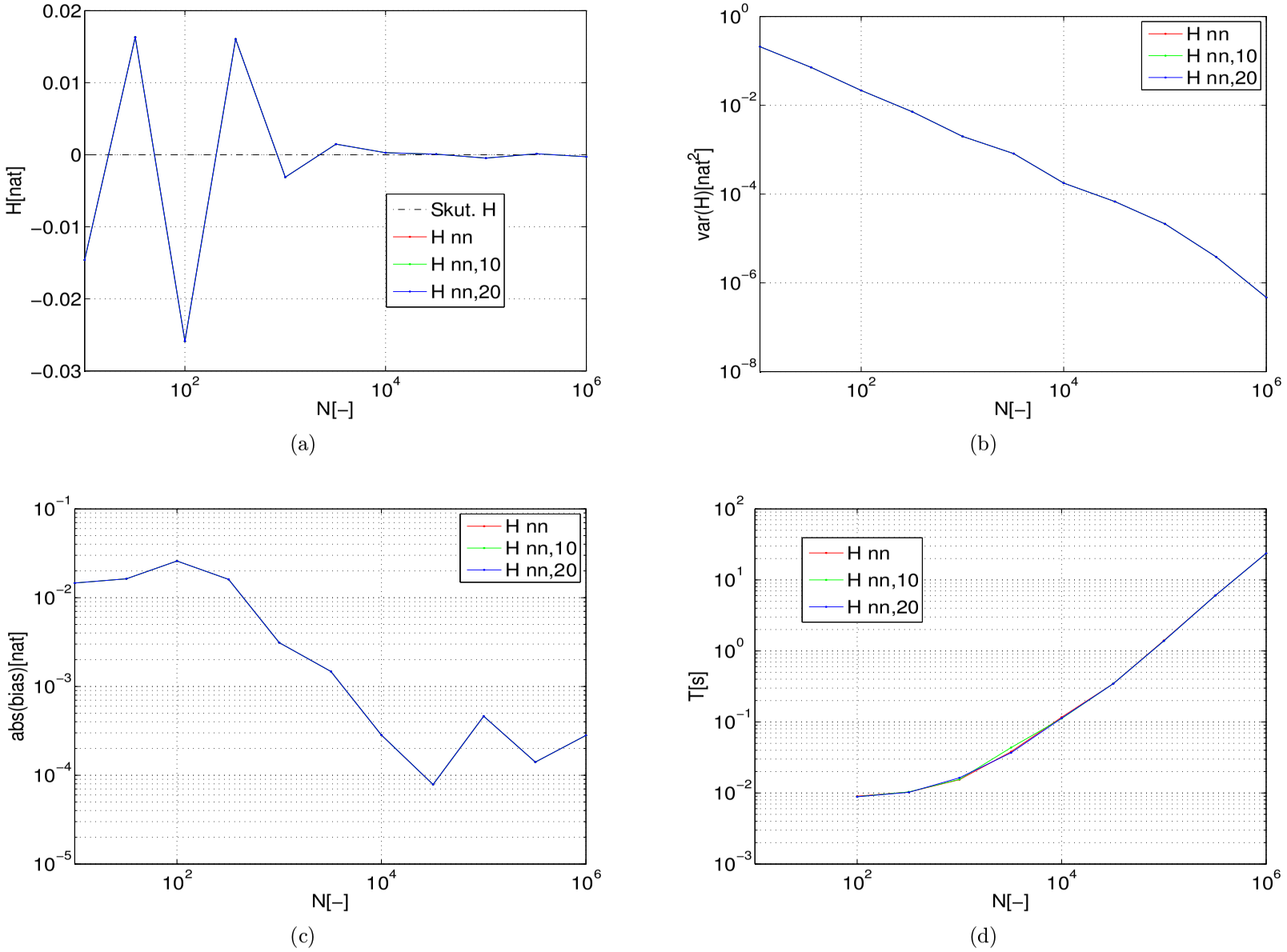
Tabulka 7.13: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho směrodatná odchylka, bias a čas běhu pro rovnoměrné rozdělení s $d = 1$. Skutečná hodnota entropie $H = 0$.

	$\hat{H}_{nn}$				$\hat{H}_{nn,10}$				$\hat{H}_{nn,20}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	-0.0146	0.4572	-0.0146		-0.0146	0.4572	-0.0146		-0.0146	0.4572	-0.0146	
$3.2 * 10^1$	0.0163	0.2674	0.0163		0.0163	0.2674	0.0163		0.0163	0.2674	0.0163	
10^2	-0.0259	0.1469	-0.0259	0.0090	-0.0259	0.1469	-0.0259	0.0089	-0.0259	0.1469	-0.0259	0.0088
$3.2 * 10^2$	0.0161	0.0845	0.0161	0.0103	0.0161	0.0845	0.0161	0.0104	0.0161	0.0845	0.0161	0.0102
10^3	-0.0031	0.0446	-0.0031	0.0154	-0.0031	0.0446	-0.0031	0.0155	-0.0031	0.0446	-0.0031	0.0164
$3.2 * 10^3$	0.0015	0.0285	0.0015	0.0382	0.0015	0.0285	0.0015	0.0436	0.0015	0.0285	0.0015	0.0368
10^4	0.0003	0.0133	0.0003	0.1167	0.0003	0.0133	0.0003	0.1117	0.0003	0.0133	0.0003	0.1130
$3.2 * 10^4$	0.0001	0.0082	0.0001	0.3464	0.0001	0.0082	0.0001	0.3469	0.0001	0.0082	0.0001	0.3455
10^5	-0.0005	0.0046	-0.0005	1.4051	-0.0005	0.0046	-0.0005	1.3839	-0.0005	0.0046	-0.0005	1.3849
$3.2 * 10^5$	0.0001	0.0020	0.0001	6.0643	0.0001	0.0020	0.0001	6.0596	0.0001	0.0020	0.0001	6.0619
10^6	-0.0003	0.0007	-0.0003	23.7224	-0.0003	0.0007	-0.0003	23.6627	-0.0003	0.0007	-0.0003	23.6850

Tabulka 7.14: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho směrodatná odchylka, bias a čas běhu pro rovnoměrné rozdělení s $d = 1$. Skutečná hodnota entropie $H = 0$.



Obrázek 7.31: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho rozptyl, bias a čas běhu pro rovnoměrné rozdělení s $d = 1$. Skutečná hodnota entropie $H = 0$.



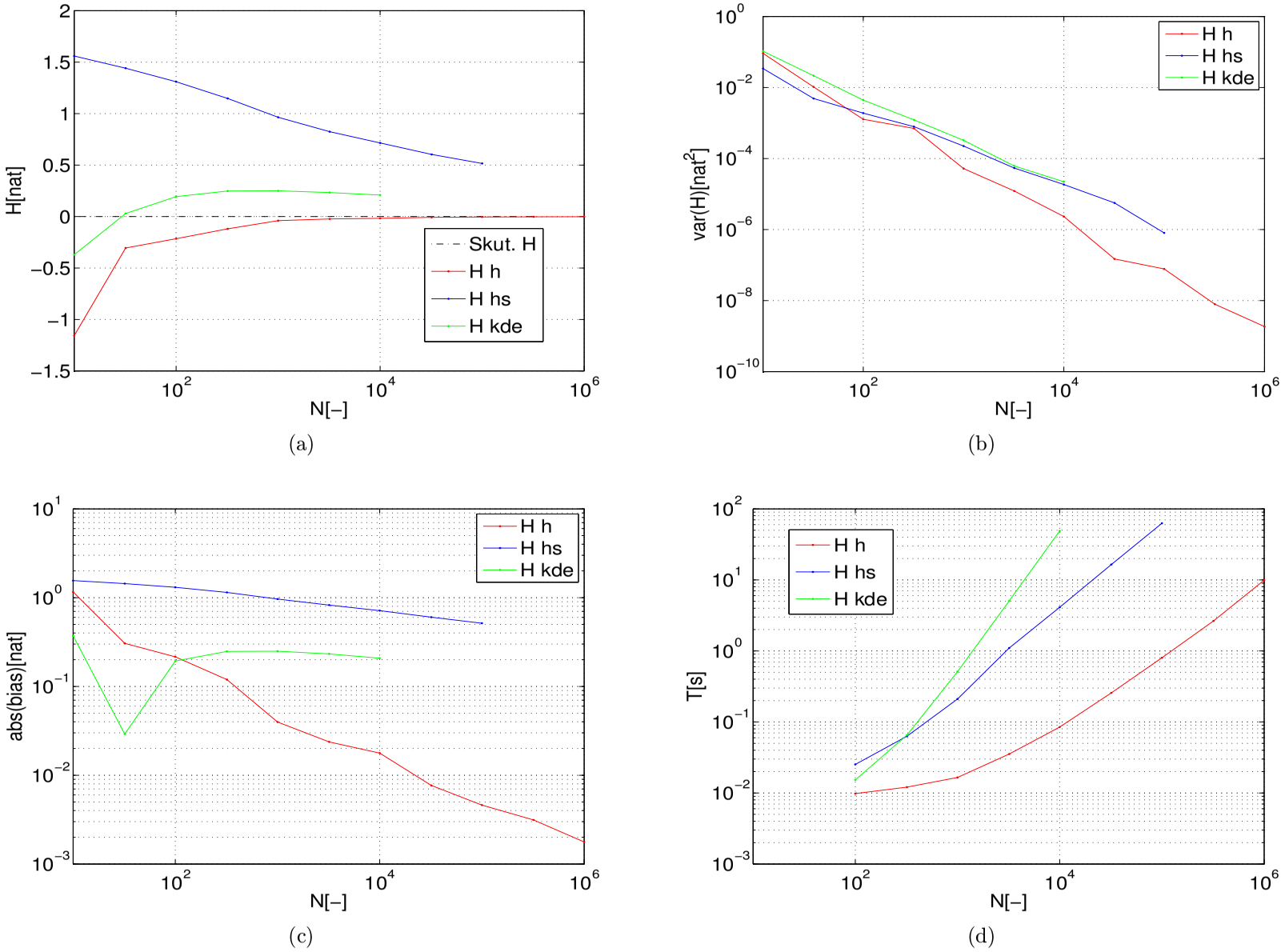
Obrázek 7.32: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho rozptyl, bias a čas běhu pro rovnoměrné rozdělení s $d = 1$. Skutečná hodnota entropie $H = 0$.

	$\hat{H}_h$				$\hat{H}_{hs}$				$\hat{H}_{kde}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	-1.1575	0.3029	-1.1575		1.5589	0.1853	1.5589		-0.3712	0.3240	-0.3712	
$3.2 * 10^1$	-0.3060	0.1019	-0.3060		1.4410	0.0699	1.4410		0.0291	0.1467	0.0291	
10^2	-0.2155	0.0357	-0.2155	0.0098	1.3092	0.0436	1.3092	0.0253	0.1933	0.0664	0.1933	0.0153
$3.2 * 10^2$	-0.1194	0.0266	-0.1194	0.0121	1.1467	0.0281	1.1467	0.0629	0.2481	0.0351	0.2481	0.0653
10^3	-0.0397	0.0072	-0.0397	0.0165	0.9637	0.0150	0.9637	0.2107	0.2494	0.0180	0.2494	0.5117
$3.2 * 10^3$	-0.0237	0.0035	-0.0237	0.0354	0.8245	0.0073	0.8245	1.0993	0.2328	0.0079	0.2328	5.0693
10^4	-0.0178	0.0015	-0.0178	0.0848	0.7150	0.0043	0.7150	4.1172	0.2079	0.0047	0.2079	48.9606
$3.2 * 10^4$	-0.0077	0.0004	-0.0077	0.2569	0.6032	0.0024	0.6032	16.4738				
10^5	-0.0046	0.0003	-0.0046	0.8011	0.5151	0.0009	0.5151	62.7209				
$3.2 * 10^5$	-0.0031	0.0001	-0.0031	2.6450								
10^6	-0.0018	0.0000	-0.0018	9.9159								

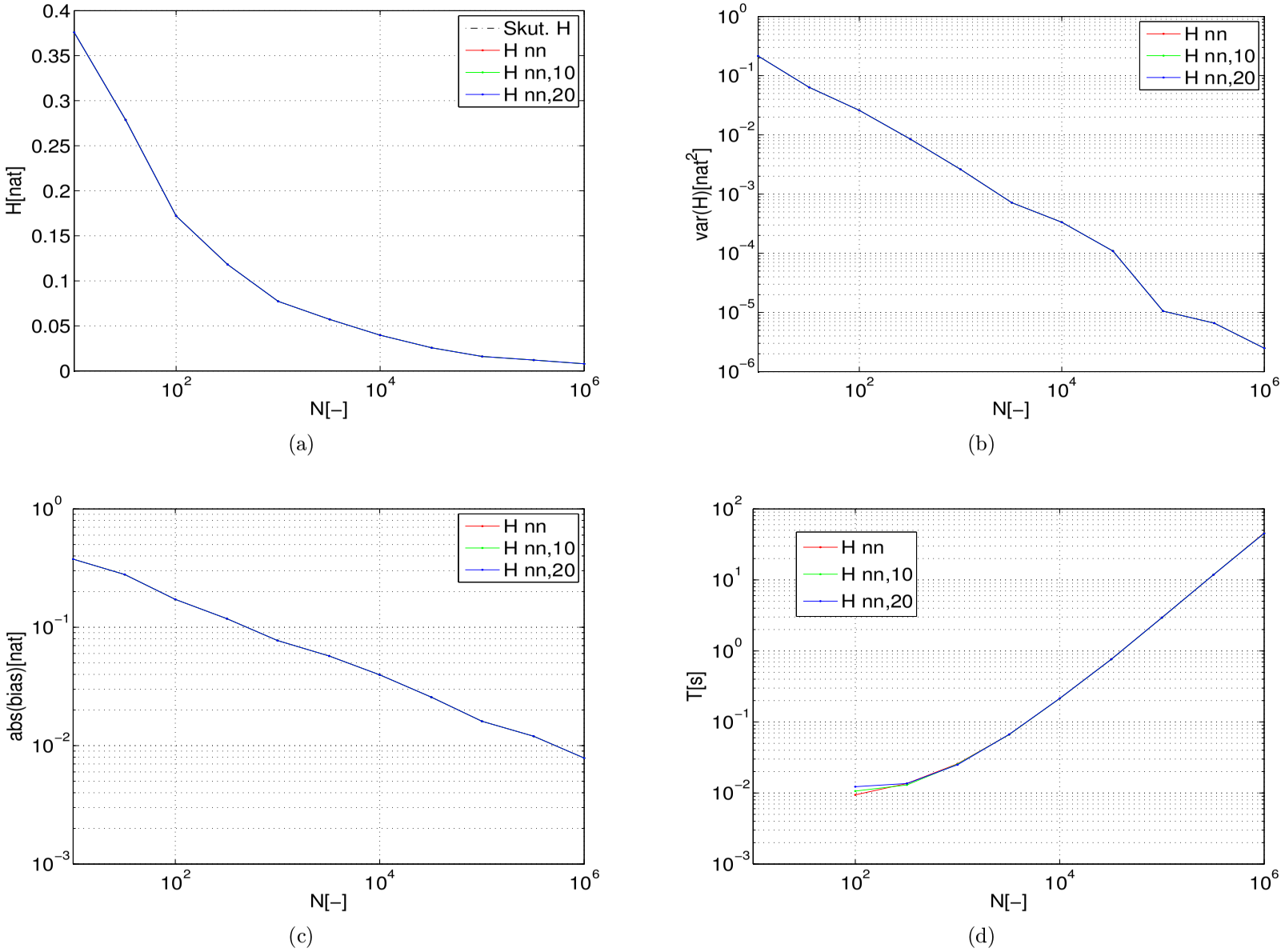
Tabulka 7.15: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho směrodatná odchylka, bias a čas běhu pro rovnoměrné rozdělení s $d = 3$. Skutečná hodnota entropie $H = 0$.

	$\hat{H}_{nn}$				$\hat{H}_{nn,10}$				$\hat{H}_{nn,20}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	0.3760	0.4639	0.3760		0.3760	0.4639	0.3760		0.3760	0.4639	0.3760	
$3.2 * 10^1$	0.2787	0.2519	0.2787		0.2787	0.2519	0.2787		0.2787	0.2519	0.2787	
10^2	0.1721	0.1611	0.1721	0.0094	0.1721	0.1611	0.1721	0.0106	0.1721	0.1611	0.1721	0.0123
$3.2 * 10^2$	0.1181	0.0920	0.1181	0.0136	0.1181	0.0920	0.1181	0.0130	0.1181	0.0920	0.1181	0.0136
10^3	0.0772	0.0511	0.0772	0.0258	0.0772	0.0511	0.0772	0.0254	0.0772	0.0511	0.0772	0.0250
$3.2 * 10^3$	0.0572	0.0268	0.0572	0.0665	0.0572	0.0268	0.0572	0.0665	0.0572	0.0268	0.0572	0.0669
10^4	0.0397	0.0183	0.0397	0.2141	0.0397	0.0183	0.0397	0.2140	0.0397	0.0183	0.0397	0.2144
$3.2 * 10^4$	0.0257	0.0104	0.0257	0.7648	0.0257	0.0104	0.0257	0.7633	0.0257	0.0104	0.0257	0.7620
10^5	0.0161	0.0032	0.0161	2.9487	0.0161	0.0032	0.0161	2.9387	0.0161	0.0032	0.0161	2.9429
$3.2 * 10^5$	0.0120	0.0026	0.0120	11.8486	0.0120	0.0026	0.0120	11.8653	0.0120	0.0026	0.0120	11.8540
10^6	0.0079	0.0016	0.0079	45.8183	0.0079	0.0016	0.0079	45.2995	0.0079	0.0016	0.0079	45.2504

Tabulka 7.16: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho směrodatná odchylka, bias a čas běhu pro rovnoměrné rozdělení s $d = 3$. Skutečná hodnota entropie $H = 0$.



Obrázek 7.33: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho rozptyl, bias a čas běhu pro rovnoměrné rozdělení s $d = 3$. Skutečná hodnota entropie $H = 0$.



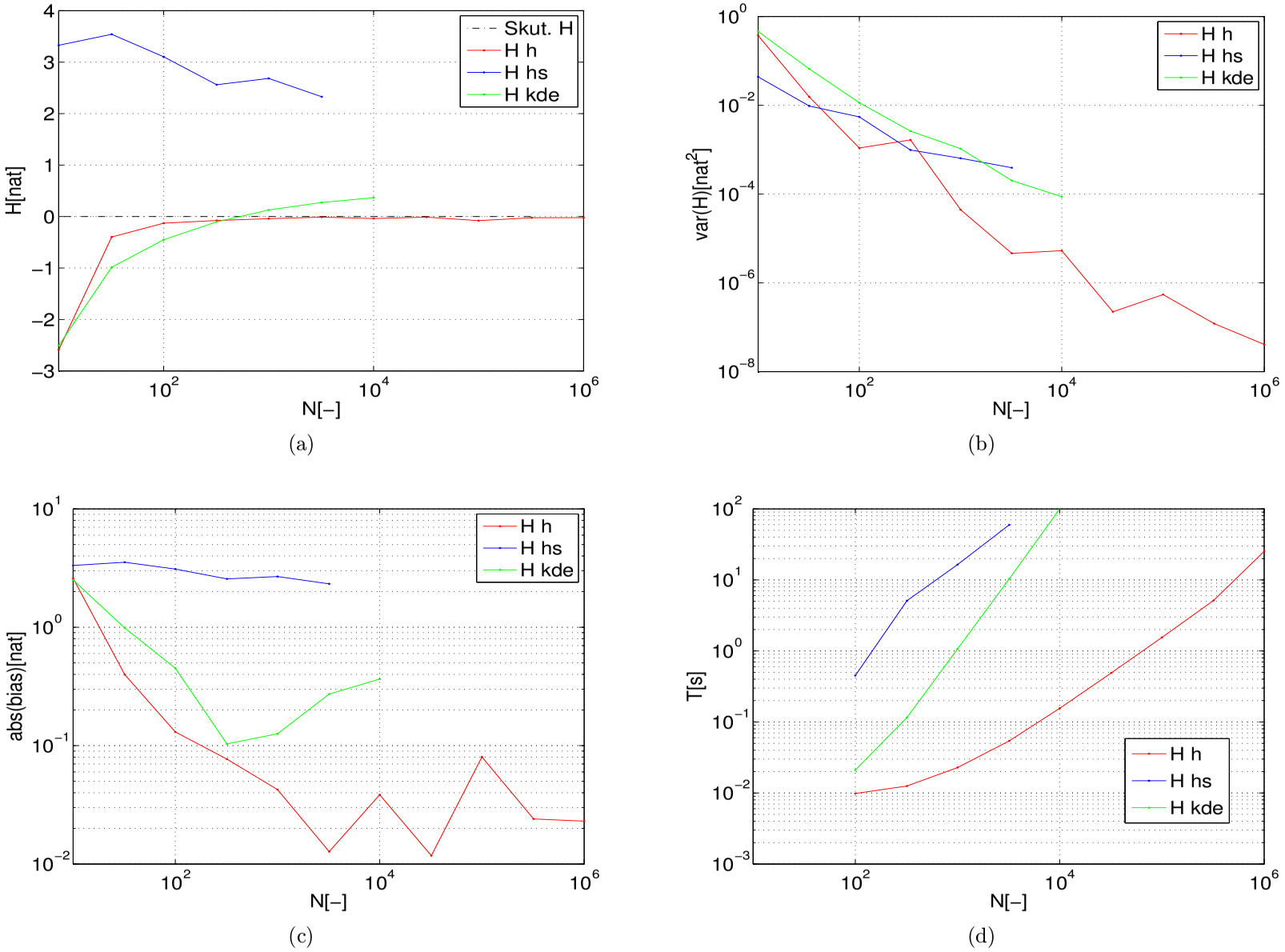
Obrázek 7.34: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho rozptyl, bias a čas běhu pro rovnoměrné rozdělení s $d = 3$. Skutečná hodnota entropie $H = 0$.

	$\hat{H}_h$				$\hat{H}_{hs}$				$\hat{H}_{kde}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	-2.5886	0.6111	-2.5886		3.3266	0.2090	3.3266		-2.5050	0.6777	-2.5050	
$3.2 * 10^1$	-0.3982	0.1244	-0.3982		3.5410	0.0982	3.5410		-0.9866	0.2570	-0.9866	
10^2	-0.1305	0.0330	-0.1305	0.0098	3.1028	0.0740	3.1028	0.4502	-0.4513	0.1069	-0.4513	0.0210
$3.2 * 10^2$	-0.0769	0.0407	-0.0769	0.0125	2.5610	0.0313	2.5610	5.0877	-0.1035	0.0512	-0.1035	0.1155
10^3	-0.0425	0.0067	-0.0425	0.0228	2.6829	0.0253	2.6829	16.4022	0.1260	0.0325	0.1260	1.0656
$3.2 * 10^3$	-0.0127	0.0021	-0.0127	0.0541	2.3275	0.0198	2.3275	59.4896	0.2727	0.0142	0.2727	10.3103
10^4	-0.0384	0.0023	-0.0384	0.1558					0.3652	0.0093	0.3652	100.0697
$3.2 * 10^4$	-0.0118	0.0005	-0.0118	0.4916								
10^5	-0.0802	0.0007	-0.0802	1.5498								
$3.2 * 10^5$	-0.0240	0.0003	-0.0240	5.1341								
10^6	-0.0230	0.0002	-0.0230	25.2661								

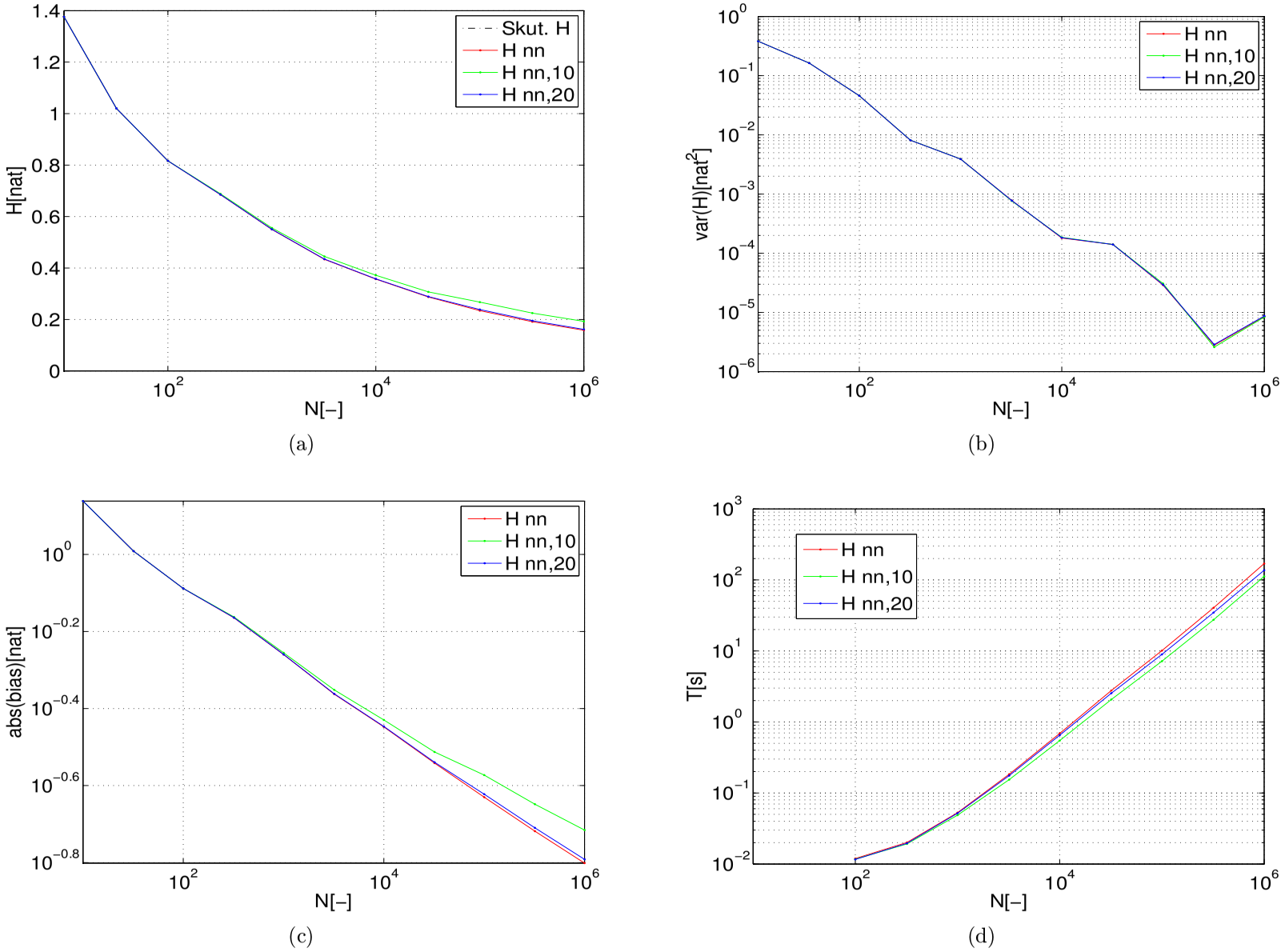
Tabulka 7.17: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho směrodatná odchylka, bias a čas běhu pro rovnoměrné rozdělení s $d = 6$. Skutečná hodnota entropie $H = 0$.

	$\hat{H}_{nn}$				$\hat{H}_{nn,10}$				$\hat{H}_{nn,20}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	1.3763	0.6182	1.3763		1.3763	0.6182	1.3763		1.3763	0.6182	1.3763	
$3.2 * 10^1$	1.0200	0.4039	1.0200		1.0200	0.4039	1.0200		1.0200	0.4039	1.0200	
10^2	0.8164	0.2141	0.8164	0.0120	0.8164	0.2141	0.8164	0.0117	0.8164	0.2141	0.8164	0.0116
$3.2 * 10^2$	0.6853	0.0900	0.6853	0.0201	0.6882	0.0902	0.6882	0.0192	0.6854	0.0901	0.6854	0.0195
10^3	0.5498	0.0626	0.5498	0.0528	0.5553	0.0626	0.5553	0.0487	0.5501	0.0625	0.5501	0.0519
$3.2 * 10^3$	0.4337	0.0279	0.4337	0.1833	0.4450	0.0277	0.4450	0.1540	0.4346	0.0279	0.4346	0.1756
10^4	0.3568	0.0134	0.3568	0.6885	0.3718	0.0136	0.3718	0.5464	0.3577	0.0135	0.3577	0.6517
$3.2 * 10^4$	0.2873	0.0119	0.2873	2.7722	0.3070	0.0119	0.3070	2.0709	0.2888	0.0118	0.2888	2.5422
10^5	0.2343	0.0054	0.2343	10.1122	0.2674	0.0055	0.2674	7.1745	0.2385	0.0054	0.2385	8.9436
$3.2 * 10^5$	0.1912	0.0017	0.1912	40.4627	0.2246	0.0016	0.2246	27.4234	0.1951	0.0017	0.1951	34.6405
10^6	0.1580	0.0029	0.1580	169.6326	0.1925	0.0029	0.1925	112.1506	0.1615	0.0030	0.1615	136.8935

Tabulka 7.18: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho směrodatná odchylka, bias a čas běhu pro rovnoměrné rozdělení s $d = 6$. Skutečná hodnota entropie $H = 0$.



Obrázek 7.35: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho rozptyl, bias a čas běhu pro rovnoměrné rozdělení s $d = 6$. Skutečná hodnota entropie $H = 0$.



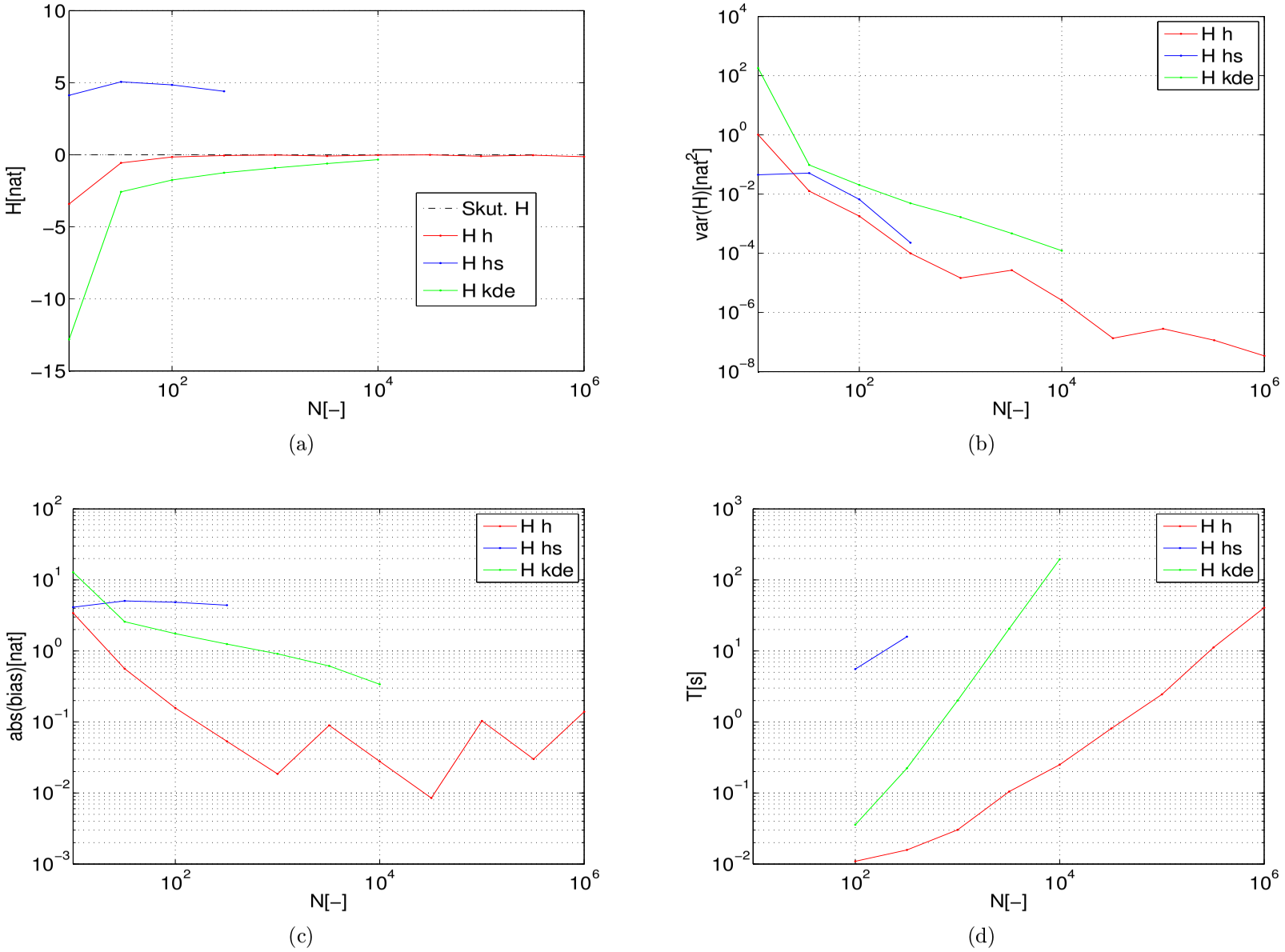
Obrázek 7.36: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho rozptyl, bias a čas běhu pro rovnoměrné rozdělení s $d = 6$. Skutečná hodnota entropie $H = 0$.

	$\hat{H}_h$				$\hat{H}_{hs}$				$\hat{H}_{kde}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	-3.4127	1.0063	-3.4127		4.1262	0.2115	4.1262		-12.8140	13.4092	-12.8140	
$3.2 * 10^1$	-0.5612	0.1125	-0.5612		5.0596	0.2261	5.0596		-2.5749	0.3107	-2.5749	
10^2	-0.1573	0.0424	-0.1573	0.0109	4.8449	0.0813	4.8449	5.5429	-1.7519	0.1427	-1.7519	0.0357
$3.2 * 10^2$	-0.0534	0.0100	-0.0534	0.0158	4.4013	0.0151	4.4013	15.8738	-1.2480	0.0699	-1.2480	0.2233
10^3	-0.0185	0.0038	-0.0185	0.0302					-0.9113	0.0408	-0.9113	2.0006
$3.2 * 10^3$	-0.0895	0.0052	-0.0895	0.1046					-0.6133	0.0216	-0.6133	20.5401
10^4	-0.0279	0.0016	-0.0279	0.2502					-0.3391	0.0110	-0.3391	195.4916
$3.2 * 10^4$	-0.0085	0.0004	-0.0085	0.8098								
10^5	-0.1034	0.0005	-0.1034	2.4436								
$3.2 * 10^5$	-0.0300	0.0003	-0.0300	11.1873								
10^6	-0.1393	0.0002	-0.1393	40.6345								

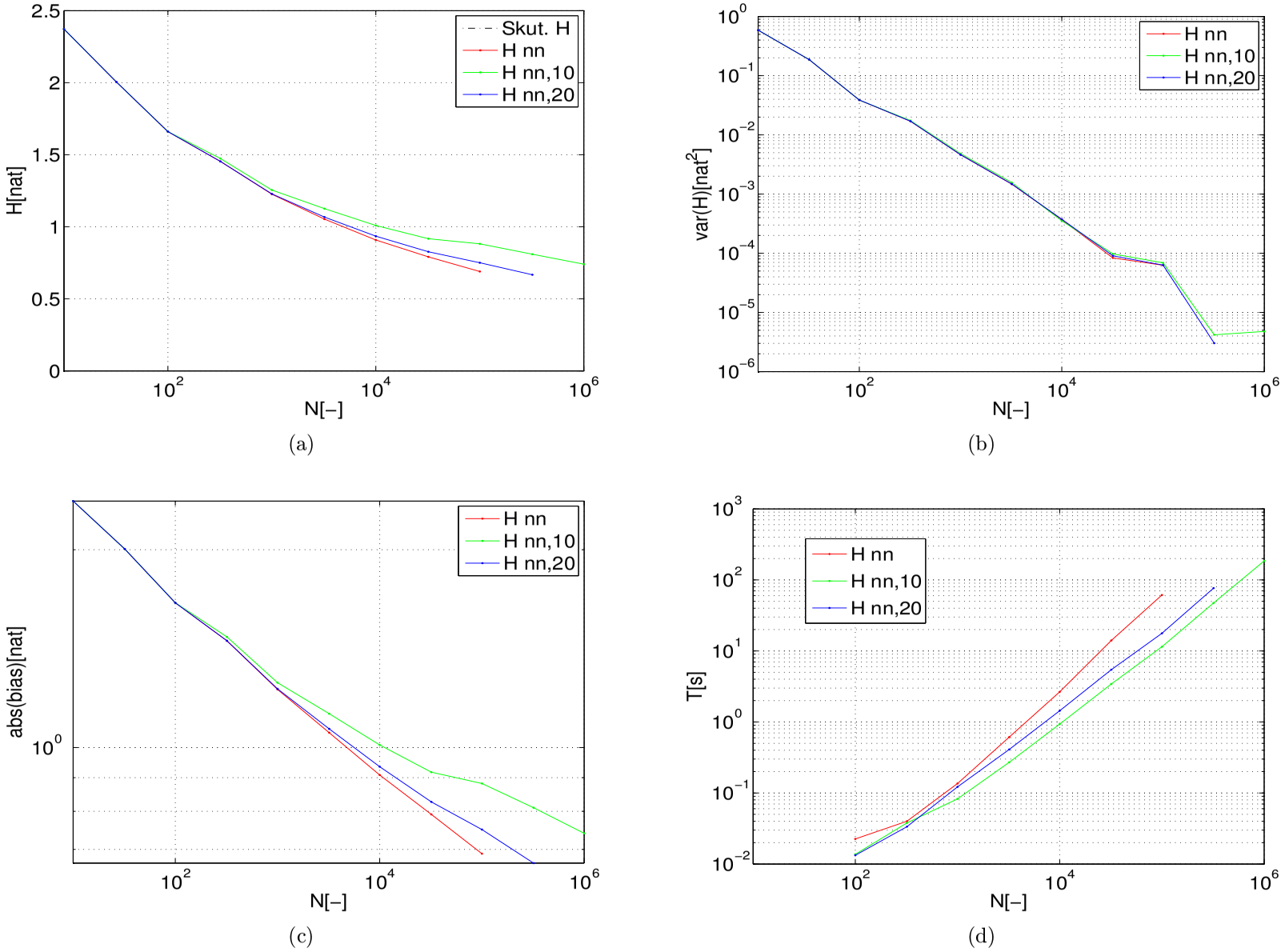
Tabulka 7.19: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho směrodatná odchylka, bias a čas běhu pro rovnoměrné rozdělení s $d = 9$. Skutečná hodnota entropie $H = 0$.

	$\hat{H}_{nn}$				$\hat{H}_{nn,10}$				$\hat{H}_{nn,20}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	2.3717	0.7672	2.3717		2.3717	0.7672	2.3717		2.3717	0.7672	2.3717	
$3.2 * 10^1$	2.0051	0.4322	2.0051		2.0051	0.4322	2.0051		2.0051	0.4322	2.0051	
10^2	1.6605	0.1969	1.6605	0.0225	1.6605	0.1969	1.6605	0.0137	1.6605	0.1969	1.6605	0.0132
$3.2 * 10^2$	1.4535	0.1303	1.4535	0.0400	1.4735	0.1321	1.4735	0.0379	1.4540	0.1305	1.4540	0.0335
10^3	1.2259	0.0681	1.2259	0.1358	1.2551	0.0693	1.2551	0.0827	1.2284	0.0679	1.2284	0.1219
$3.2 * 10^3$	1.0535	0.0383	1.0535	0.6114	1.1261	0.0395	1.1261	0.2702	1.0677	0.0384	1.0677	0.4094
10^4	0.9084	0.0192	0.9084	2.6548	1.0093	0.0188	1.0093	0.9313	0.9352	0.0194	0.9352	1.4376
$3.2 * 10^4$	0.7914	0.0091	0.7914	14.0568	0.9175	0.0099	0.9175	3.4279	0.8267	0.0095	0.8267	5.4369
10^5	0.6897	0.0079	0.6897	61.2986	0.8820	0.0083	0.8820	11.4344	0.7505	0.0080	0.7505	17.6827
$3.2 * 10^5$					0.8100	0.0020	0.8100	47.2783	0.6673	0.0017	0.6673	76.6700
10^6					0.7406	0.0022	0.7406	185.0576				

Tabulka 7.20: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho směrodatná odchylka, bias a čas běhu pro rovnoměrné rozdělení s $d = 9$. Skutečná hodnota entropie $H = 0$.



Obrázek 7.37: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho rozptyl, bias a čas běhu pro rovnoměrné rozdělení s $d = 9$. Skutečná hodnota entropie $H = 0$.



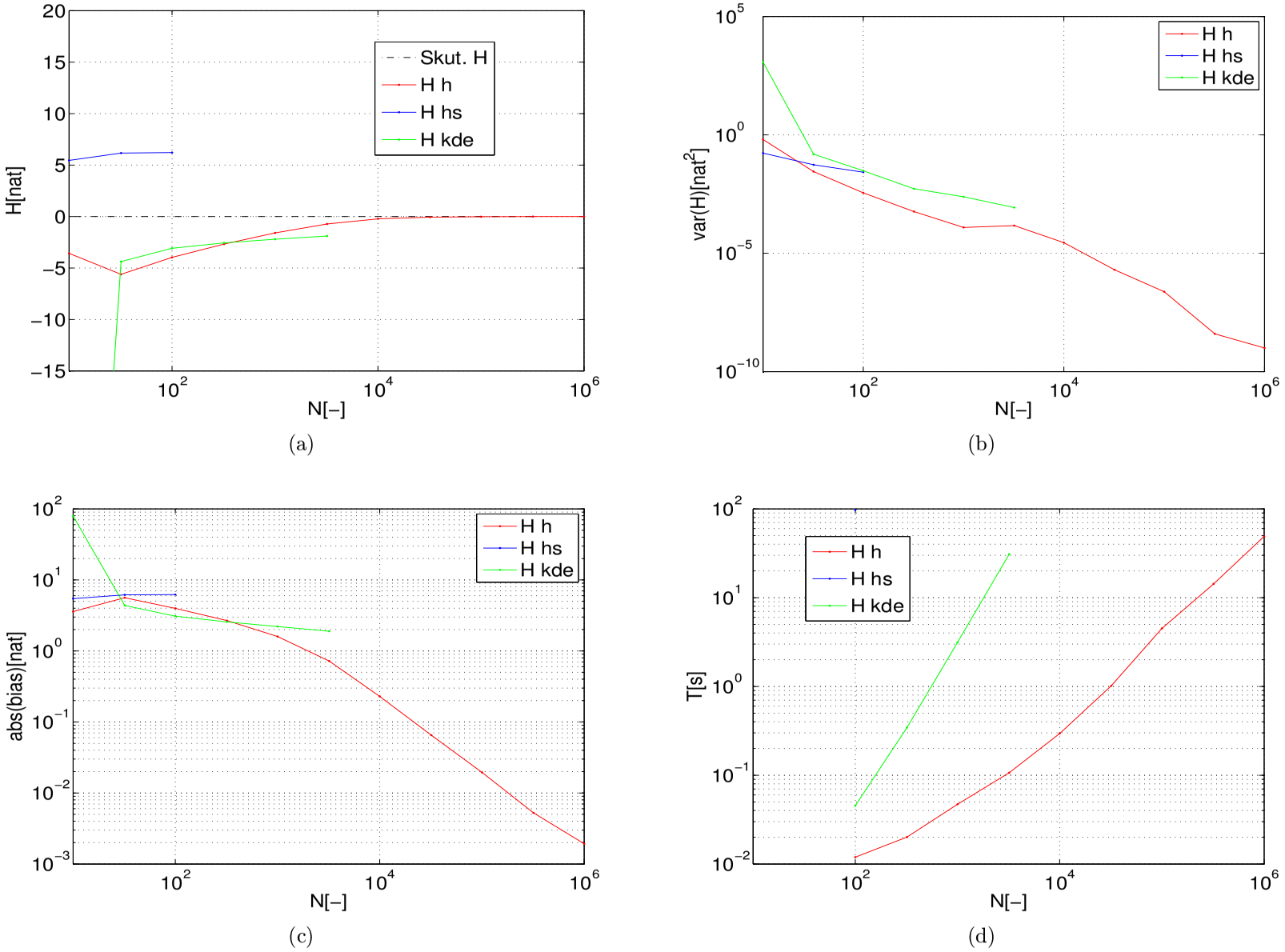
Obrázek 7.38: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho rozptyl, bias a čas běhu pro rovnoměrné rozdělení s $d = 9$. Skutečná hodnota entropie $H = 0$.

	$\hat{H}_h$				$\hat{H}_{hs}$				$\hat{H}_{kde}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	-3.5822	0.7987	-3.5822		5.4546	0.4114	5.4546		-78.9698	34.4501	-78.9698	
$3.2 * 10^1$	-5.6180	0.1676	-5.6180		6.1631	0.2336	6.1631		-4.3756	0.3907	-4.3756	
10^2	-3.9724	0.0597	-3.9724	0.0119	6.2158	0.1626	6.2158	97.0525	-3.0867	0.1748	-3.0867	0.0453
$3.2 * 10^2$	-2.6782	0.0240	-2.6782	0.0201					-2.5625	0.0729	-2.5625	0.3434
10^3	-1.5941	0.0111	-1.5941	0.0471					-2.2078	0.0492	-2.2078	3.1402
$3.2 * 10^3$	-0.7205	0.0121	-0.7205	0.1066					-1.8954	0.0292	-1.8954	30.8721
10^4	-0.2292	0.0053	-0.2292	0.2946								
$3.2 * 10^4$	-0.0650	0.0014	-0.0650	1.0180								
10^5	-0.0195	0.0005	-0.0195	4.4974								
$3.2 * 10^5$	-0.0052	0.0001	-0.0052	14.2774								
10^6	-0.0019	0.0000	-0.0019	49.0251								

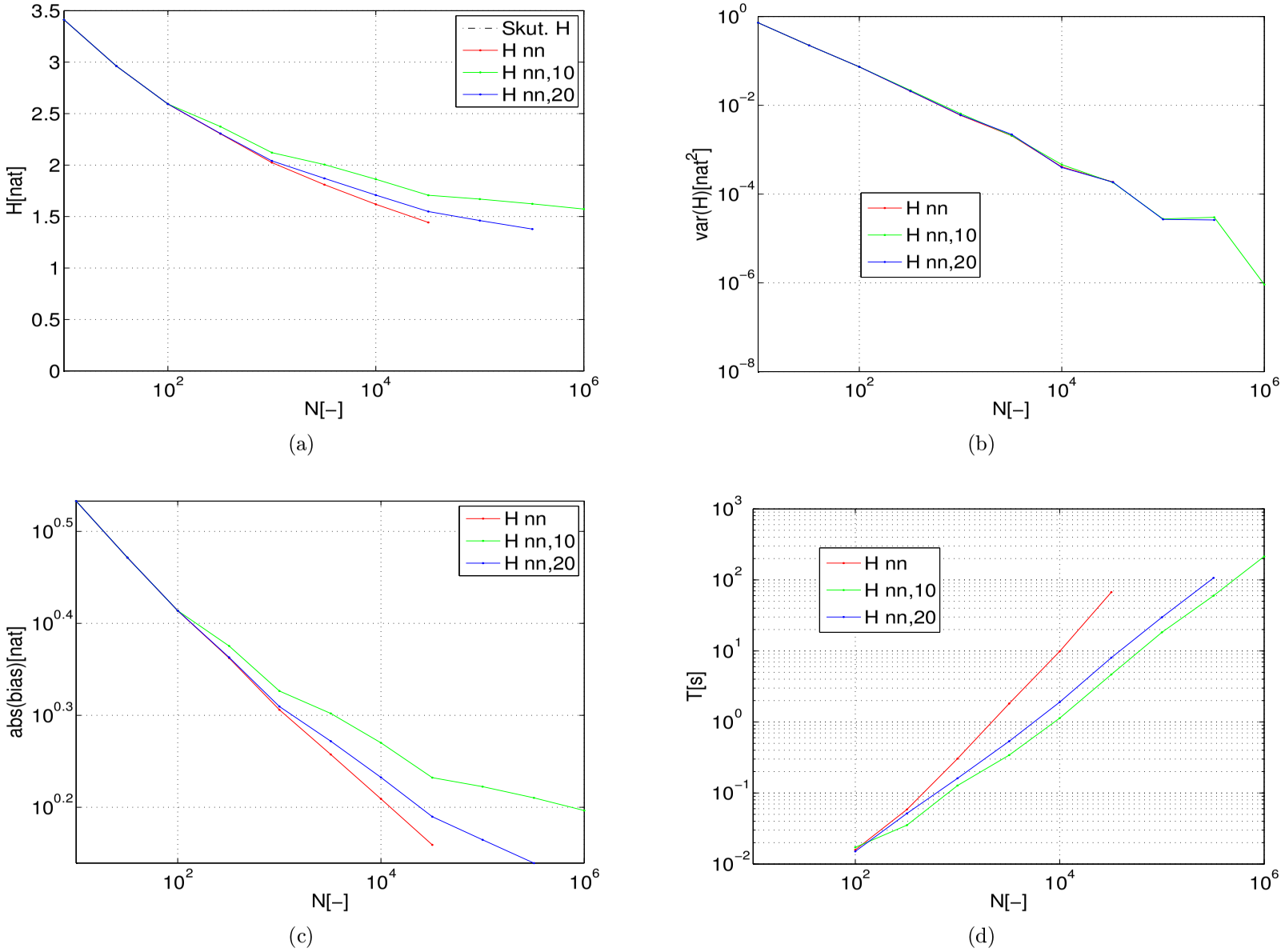
Tabulka 7.21: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho směrodatná odchylka, bias a čas běhu pro rovnoměrné rozdělení s $d = 12$. Skutečná hodnota entropie $H = 0$.

	$\hat{H}_{nn}$				$\hat{H}_{nn,10}$				$\hat{H}_{nn,20}$			
N[-]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\hat{H}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	3.4121	0.8510	3.4121		3.4121	0.8510	3.4121		3.4121	0.8510	3.4121	
$3.2 * 10^1$	2.9617	0.4719	2.9617		2.9617	0.4719	2.9617		2.9617	0.4719	2.9617	
10^2	2.5930	0.2710	2.5930	0.0158	2.5930	0.2710	2.5930	0.0171	2.5930	0.2710	2.5930	0.0151
$3.2 * 10^2$	2.3033	0.1440	2.3033	0.0586	2.3739	0.1473	2.3739	0.0353	2.3071	0.1444	2.3071	0.0515
10^3	2.0235	0.0771	2.0235	0.3044	2.1205	0.0806	2.1205	0.1272	2.0398	0.0778	2.0398	0.1610
$3.2 * 10^3$	1.8092	0.0454	1.8092	1.8150	2.0046	0.0455	2.0046	0.3417	1.8703	0.0469	1.8703	0.5357
10^4	1.6185	0.0203	1.6185	9.8530	1.8625	0.0214	1.8625	1.1322	1.7081	0.0199	1.7081	1.9028
$3.2 * 10^4$	1.4424	0.0137	1.4424	67.1895	1.7068	0.0135	1.7068	4.6586	1.5480	0.0136	1.5480	8.0428
10^5					1.6691	0.0052	1.6691	18.3061	1.4609	0.0052	1.4609	29.7482
$3.2 * 10^5$					1.6227	0.0055	1.6227	59.9747	1.3783	0.0051	1.3783	106.7323
10^6					1.5721	0.0010	1.5721	213.8410				

Tabulka 7.22: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho směrodatná odchylka, bias a čas běhu pro rovnoměrné rozdělení s $d = 12$. Skutečná hodnota entropie $H = 0$.



Obrázek 7.39: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_h$, $\hat{H}_{hs}$ a $\hat{H}_{kde}$, jeho rozptyl, bias a čas běhu pro rovnoměrné rozdělení s $d = 12$. Skutečná hodnota entropie $H = 0$.



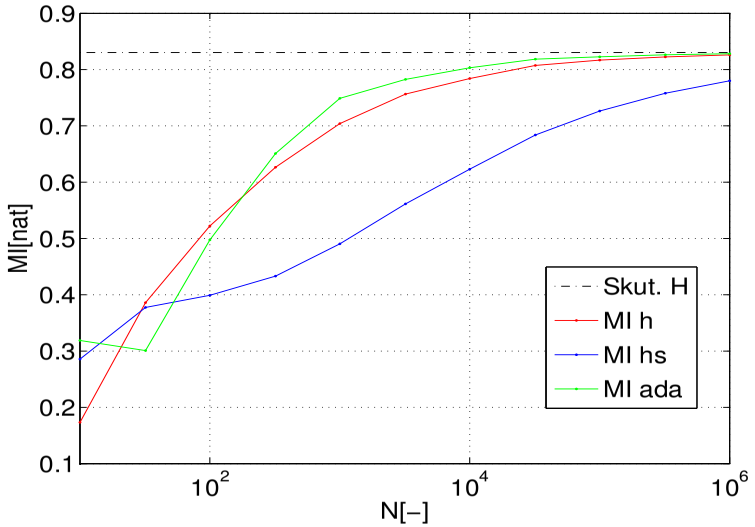
Obrázek 7.40: Průměrná hodnota odhadu entropie estimátorů $\hat{H}_{nn}$, $\hat{H}_{nn,10}$ a $\hat{H}_{nn,20}$, jeho rozptyl, bias a čas běhu pro rovnoměrné rozdělení s $d = 12$. Skutečná hodnota entropie $H = 0$.

7.2.5 Porovnání variant histogramových estimátorů pro odhad vzájemné informace

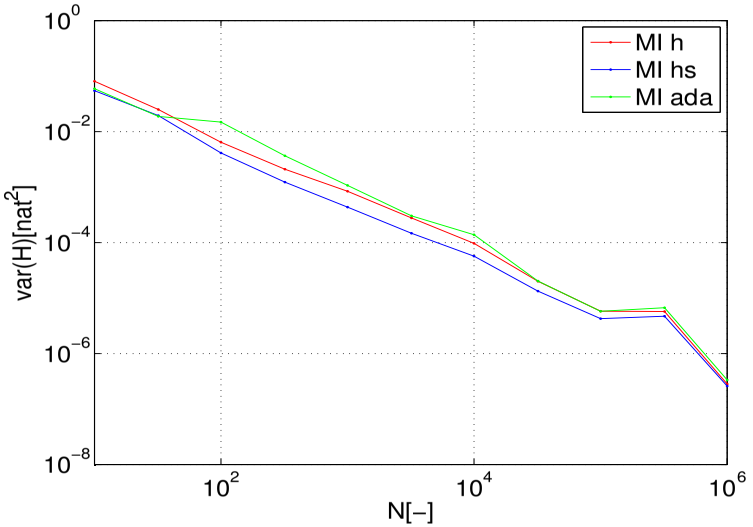
Pro účely vyhodnocení přesnosti a rychlosti histogramového estimátoru s adaptivním histogramováním $\widehat{MI}_{ada}$ byly porovnány jeho statistické vlastnosti a rychlost s estimátory vzájemné informace $\widehat{MI}_h$ a $\widehat{MI}_{hs}$. Experiment byl uskutečněn pro sdružené dimenze $d = \{2, 4, 6\}$. Data byla generována z vícerozměrného normálního rozdělení se stopou kovarianční matice $\mathrm{tr}(\Sigma) = d$.

	$\widehat{MI}_h$				$\widehat{MI}_{hs}$				$\widehat{MI}_{ada}$			
N[-]	$\overline{\widehat{MI}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\widehat{MI}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\widehat{MI}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	0.1734	0.2840	-1.8342		0.2856	0.2338	-1.7219		0.3189	0.2435	-1.6886	
$3.2 * 10^1$	0.3861	0.1583	-1.6215		0.3775	0.1394	-1.6300		0.3009	0.1369	-1.7066	
10^2	0.5223	0.0801	-1.4852	0.0097	0.3991	0.0641	-1.6084	0.0096	0.4975	0.1218	-1.5100	0.0104
$3.2 * 10^2$	0.6266	0.0458	-1.3810	0.0146	0.4331	0.0350	-1.5745	0.0158	0.6509	0.0606	-1.3566	0.0117
10^3	0.7041	0.0290	-1.3034	0.0148	0.4904	0.0209	-1.5171	0.0260	0.7488	0.0327	-1.2587	0.0225
$3.2 * 10^3$	0.7565	0.0167	-1.2510	0.0299	0.5613	0.0121	-1.4462	0.1042	0.7827	0.0174	-1.2248	0.0622
10^4	0.7840	0.0098	-1.2235	0.0789	0.6230	0.0075	-1.3845	0.4049	0.8032	0.0118	-1.2043	0.2307
$3.2 * 10^4$	0.8073	0.0045	-1.2002	0.2540	0.6838	0.0037	-1.3237	1.5431	0.8186	0.0045	-1.1889	1.0433
10^5	0.8168	0.0024	-1.1907	0.8499	0.7264	0.0021	-1.2812	5.5261	0.8227	0.0024	-1.1848	6.0361
$3.2 * 10^5$	0.8226	0.0024	-1.1849	3.1550	0.7579	0.0022	-1.2496	28.1957	0.8263	0.0026	-1.1812	37.8809
10^6	0.8261	0.0005	-1.1814	11.8156	0.7801	0.0005	-1.2274	126.9728	0.8282	0.0006	-1.1793	195.9860

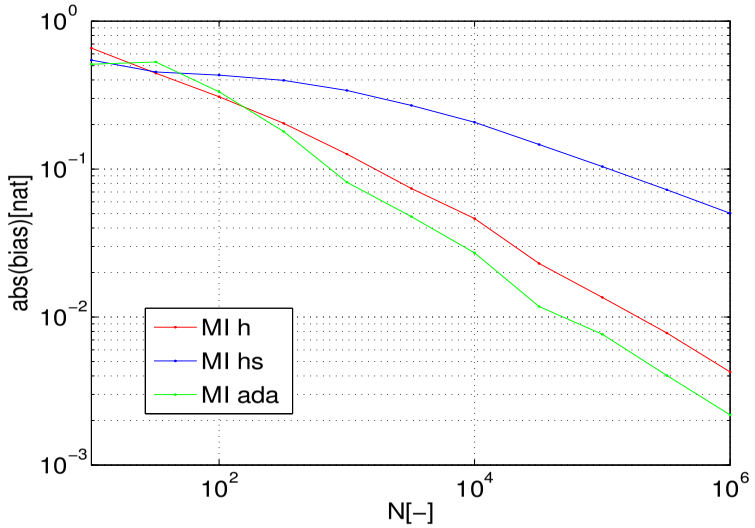
Tabulka 7.23: Průměrná hodnota odhadu vzájemné informace estimátorů $\widehat{MI}_h$, $\widehat{MI}_{hs}$ a $\widehat{MI}_{ada}$, jeho směrodatná odchylka, bias a čas běhu pro sdruženou dimenzionalitu $d_1 + d_2 = 2$. Skutečná hodnota vzájemné informace $MI = 0.8303$.



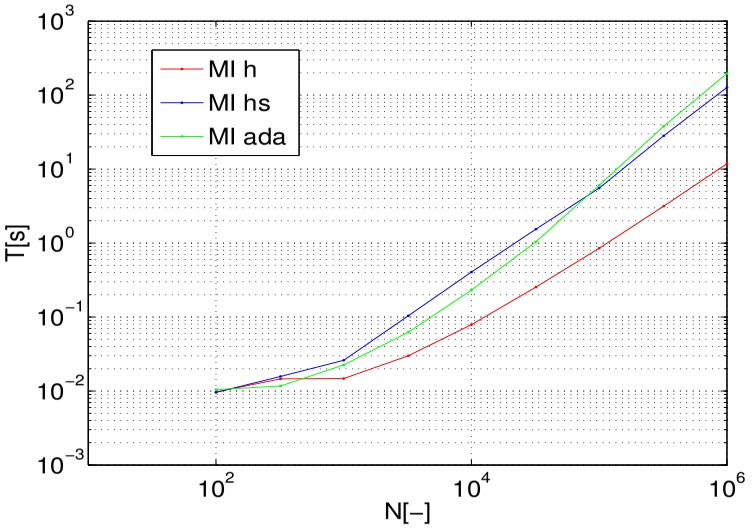
(a)



(b)



(c)

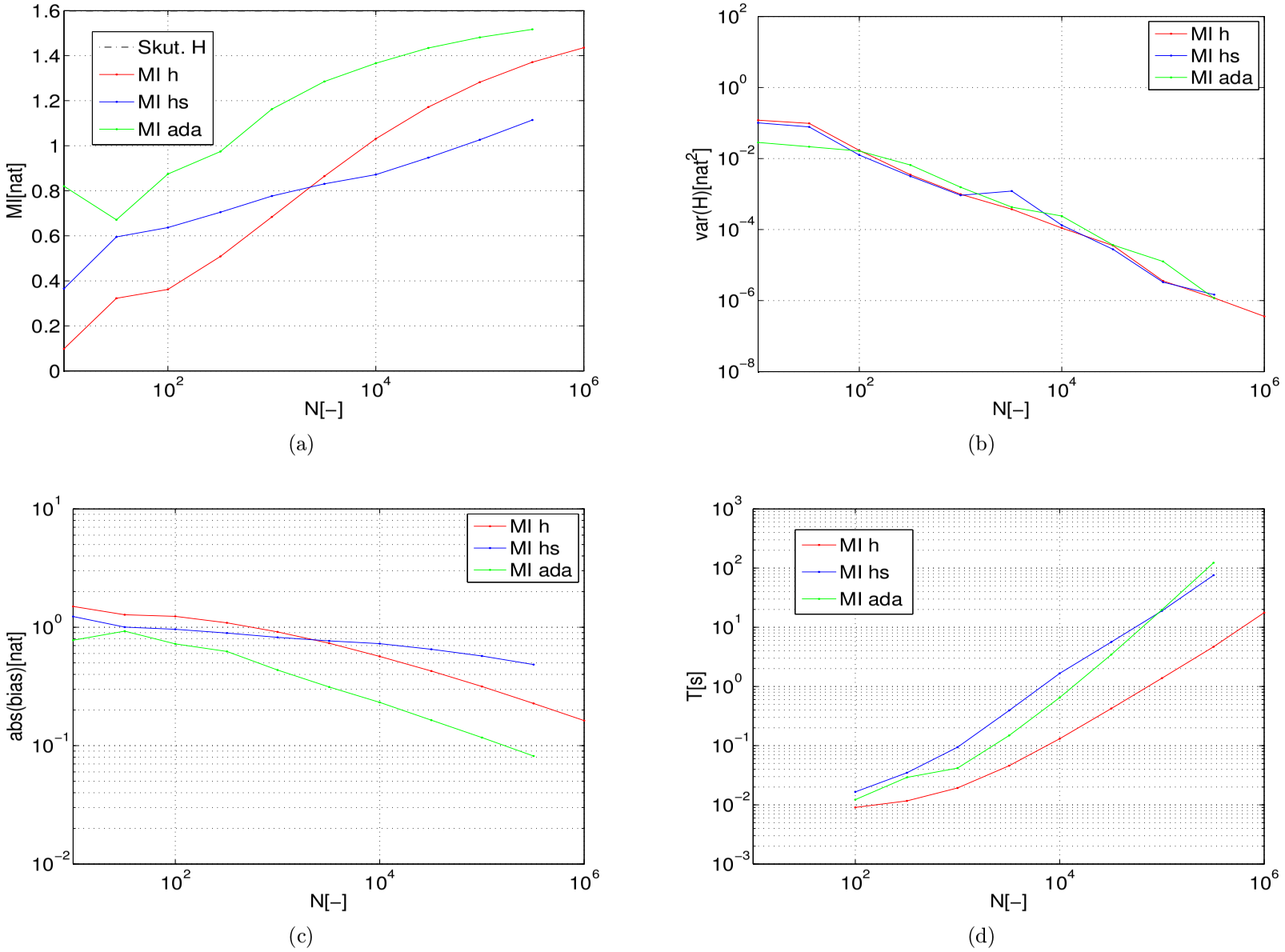


(d)

Obrázek 7.41: Průměrná hodnota odhadu vzájemné informace estimátorů $\widehat{MI}_h$, $\widehat{MI}_{hs}$ a $\widehat{MI}_{ada}$, jeho rozptyl, bias a čas běhu pro $d_1 + d_2 = 2$. Skutečná hodnota $MI = 0.8303$.

	$\widehat{MI}_h$				$\widehat{MI}_{hs}$				$\widehat{MI}_{ada}$			
N[-]	$\overline{\widehat{MI}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\widehat{MI}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\widehat{MI}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	0.0982	0.3468	-2.9576		0.3653	0.3175	-2.6904		0.8204	0.1686	-2.2353	
$3.2 * 10^1$	0.3226	0.3123	-2.7331		0.5952	0.2793	-2.4606		0.6713	0.1467	-2.3845	
10^2	0.3621	0.1301	-2.6937	0.0090	0.6368	0.1119	-2.4189	0.0165	0.8744	0.1277	-2.1814	0.0122
$3.2 * 10^2$	0.5084	0.0591	-2.5473	0.0116	0.7048	0.0563	-2.3509	0.0349	0.9742	0.0810	-2.0816	0.0291
10^3	0.6839	0.0314	-2.3719	0.0193	0.7767	0.0304	-2.2790	0.0941	1.1629	0.0394	-1.8928	0.0417
$3.2 * 10^3$	0.8650	0.0194	-2.1907	0.0458	0.8309	0.0349	-2.2249	0.3932	1.2852	0.0207	-1.7705	0.1487
10^4	1.0312	0.0105	-2.0245	0.1309	0.8716	0.0115	-2.1842	1.6630	1.3665	0.0155	-1.6893	0.6483
$3.2 * 10^4$	1.1719	0.0060	-1.8839	0.4240	0.9469	0.0053	-2.1088	5.6419	1.4342	0.0061	-1.6215	3.4561
10^5	1.2825	0.0019	-1.7732	1.3789	1.0265	0.0018	-2.0292	18.8999	1.4814	0.0036	-1.5743	19.7351
$3.2 * 10^5$	1.3713	0.0011	-1.6844	4.6867	1.1142	0.0012	-1.9415	75.7024	1.5167	0.0011	-1.5390	123.0974
10^6	1.4355	0.0006	-1.6202	17.6440								

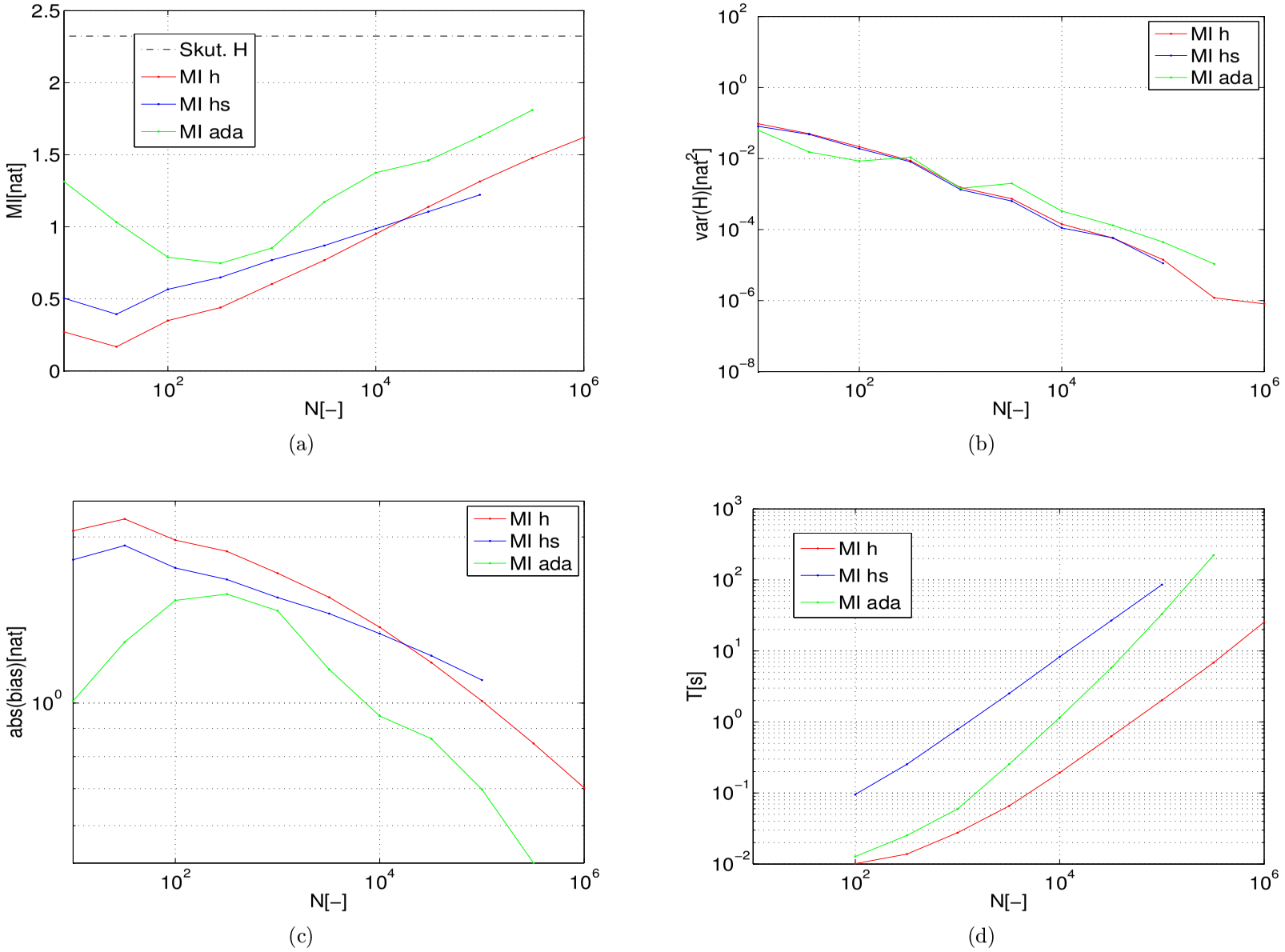
Tabulka 7.24: Průměrná hodnota odhadu vzájemné informace estimátorů $\widehat{MI}_h$, $\widehat{MI}_{hs}$ a $\widehat{MI}_{ada}$, jeho směrodatná odchylka, bias a čas běhu pro sdruženou dimenzionalitu $d_1 + d_2 = 4$. Skutečná hodnota vzájemné informace $MI = 1.598$.



Obrázek 7.42: Průměrná hodnota odhadu vzájemné informace estimátorů $\widehat{MI}_h$, $\widehat{MI}_{hs}$ a $\widehat{MI}_{ada}$, jeho rozptyl, bias a čas běhu pro $d_1 + d_2 = 4$. Skutečná hodnota $MI = 1.598$.

	$\widehat{MI}_h$				$\widehat{MI}_{hs}$				$\widehat{MI}_{ada}$			
N[-]	$\overline{\widehat{MI}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\widehat{MI}}$[nat]	σ[nat]	bias[nat]	T[s]	$\overline{\widehat{MI}}$[nat]	σ[nat]	bias[nat]	T[s]
10^1	0.2709	0.3094	-5.2264		0.5048	0.2839	-4.9925		1.3141	0.2510	-4.1831	
$3.2 * 10^1$	0.1679	0.2235	-5.3294		0.3933	0.2190	-5.1040		1.0327	0.1234	-4.4645	
10^2	0.3494	0.1468	-5.1479	0.0101	0.5659	0.1379	-4.9314	0.0954	0.7884	0.0921	-4.7088	0.0128
$3.2 * 10^2$	0.4399	0.0933	-5.0574	0.0138	0.6487	0.0901	-4.8485	0.2537	0.7472	0.1045	-4.7501	0.0252
10^3	0.6036	0.0391	-4.8937	0.0276	0.7696	0.0365	-4.7277	0.7864	0.8530	0.0381	-4.6443	0.0596
$3.2 * 10^3$	0.7686	0.0271	-4.7287	0.0657	0.8703	0.0252	-4.6270	2.5221	1.1722	0.0447	-4.3251	0.2544
10^4	0.9506	0.0119	-4.5467	0.1932	0.9868	0.0105	-4.5105	8.2564	1.3757	0.0182	-4.1216	1.1558
$3.2 * 10^4$	1.1387	0.0076	-4.3585	0.6289	1.1048	0.0076	-4.3925	26.7902	1.4610	0.0115	-4.0362	5.7778
10^5	1.3139	0.0038	-4.1834	2.0210	1.2221	0.0034	-4.2752	85.5896	1.6254	0.0067	-3.8719	33.1073
$3.2 * 10^5$	1.4781	0.0011	-4.0192	6.8432					1.8101	0.0033	-3.6872	222.6091
10^6	1.6207	0.0009	-3.8766	25.4684								

Tabulka 7.25: Průměrná hodnota odhadu vzájemné informace estimátorů $\widehat{MI}_h$, $\widehat{MI}_{hs}$ a $\widehat{MI}_{ada}$, jeho směrodatná odchylka, bias a čas běhu pro sdruženou dimenzionalitu $d_1 + d_2 = 6$. Skutečná hodnota vzájemné informace $MI = 2.3232$.



Obrázek 7.43: Průměrná hodnota odhadu vzájemné informace estimátorů $\widehat{MI}_h$, $\widehat{MI}_{hs}$ a $\widehat{MI}_{ada}$, jeho rozptyl, bias a čas běhu pro $d_1 + d_2 = 6$. Skutečná hodnota $MI = 2.3232$.

Kapitola 8

Závěr

V této práci byla rozebrána problematika odhadu entropie a vzájemné informace pro registraci obrázků. Těžiště práce spočívá v implementaci vybraných estimátorů entropie a vzájemné informace v jazyce C a v experimentálním porovnání jejich přesnosti, rychlosti a statistických vlastností. Součástí práce je též implementace estimátoru α-entropie z délky minimální kostry úplného grafu nad vzorky. Estimátory byly napsány s ohledem na jejich snadnou integraci v jiných programech.

Z implementovaných estimátorů se ukázal jako nejrychlejší histogramový estimátor. V experimentech se však projevila jeho pomalá konvergence ke skutečné hodnotě entropie pro vyšší dimenzionality. Tento jev částečně kompenzuje vyhlazení histogramu Parzenovým oknem; nevýhodou toho přístupu je však nedostatečná rychlost pro vyšší dimenze.

Ve vyšších dimenzích dat vykazovaly pro normální rozdělení lepší konvergenci ke skutečné hodnotě entropie oproti histogramovému estimátoru estimátory založené na vyhledávání nejbližších sousedů. V nízkých dimenzích byl však patrný jejich větší rozptyl. Pro rovnoměrné rozdělení se pro $d \geq 3$ projevily oproti histogramovému estimátoru vychýlenější.

Omezení počtu prohledávaných uzlů při BBF vyhledávání urychlí výpočty pro větší dimenzionality. Nevýhodou ovšem je, že pro vyšší počty vzorků ($n > 10^4$) a dimenze $d \geq 6$ se s počtem vzorků zvyšuje výchylka estimátoru.

Pro odhad vzájemné informace prokázal dobré výsledky estimátor využívající adaptivní histogramování. Estimátor měl oproti histogramovému estimátoru menší výchylku pro dimenzi $d = 2$ a počty vzorků $n > 3,2.10^3$ a ve vyšších dimenzích pro všechny testované počty vzorků.

Literatura

[1] AHMAD, I. A. – LIN, P. E. A Nonparametric Entropy Estimation for Absolutely Continuous Distributions. *IEEE Transactions on Information Theory*. 1976, 22, s. 372–375.

[2] AHMED, N. A. – GOKHALE, D. V. Entropy expressions and their estimators for multivariate distributions. *IEEE Transactions on Information Theory*. 1989.

[3] ASSENZA, A. et al. Assessment of Probability Density Estimation Methods: Parzen Window and Finite Gaussian Mixture. In *Proceedings of the 2006 IEEE International Symposium on Circuits and Systems*, s. 3245–3248, 2006.

[4] BEIRLANT, J. et al. Nonparametric entropy estimation: an overview. *International J. Math. Stat. Sci.* 1997, 6, 1, s. 17–39.

[5] BEIS, J. S. – LOWE, D. G. Shape Indexing Using Approximate Nearest-Neighbour Search in High-DimensionalSpaces. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, s. 1000–1006, June 1997.

[6] BENTLEY, J. L. Multidimensional binary search trees used for associative searching. *Commun. ACM*. 1975, 18, 9, s. 509–517.

[7] BEYER, K. et al. When Is "Nearest Neighbor" Meaningful? In *Proc. 7th Conf. Data Theory*, s. 217–235, 1999.

[8] BORŮVKA, O. O jistém problému minimálním. *Práce mor. přírodověd. spol. v Brně III*. 1926, 3, s. 37–58.

[9] BROWN, L. A survey of image registration techniques. *ACM Computing Surveys*. 1992, 24, 4, s. 326–376.

[10] CHEN, H. Gradient-based approach for fine registration of panorama images. *Journal of Computer Science and Technology*. 2004, 19, 5, s. 691–697.

[11] DARBELLAY, G. A. – VAJDA, I. Estimation of the information by an adaptive partitioning of the observation space. *IEEE Transactions on Information Theory*. 1999, 45, 4, s. 1315–1321.

[12] FREEDMAN, D. – DIACONIS, P. On the histogram as a density estimator: L2 theory. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*. 1979, 57, 4, s. 453?476.

[13] GUMUSTEKIN, S. *An Introduction to Image Mosaicing* [online]. Revize červenec 1999 [citováno 2007-01-14]. <http://www.iyte.edu.tr/eee/sevgum/research/mosaicing99/>.

[14] GYÖRFI, L. – VAN DER MEULEN, E. C. An Entropy Estimate Based on a Kernel Density Estimation. *Limit Theorems in Probability and Statistics*. 1990, s. 229–240.

[15] HARTLEY, R. V. L. Transmission of Information. *Bell Systems Technical Journal*. 1928, 7, s. 535–563.

[16] HERO, A. O. et al. Applications of entropic spanning graphs. *IEEE Signal Proc. Magazine*. September 2002, 19, 5, s. 85–95.

[17] HUTTON, B. F. et al. A hybrid 3-D reconstruction/registration algorithm for correction of head motion in emission tomography. *IEEE Transactions on Nuclear Science*. 2002, 49, s. 188–194.

[18] IVANOV, A. V. – ROZHKOVA, M. K. Properties of the statistical estimate of the entropy of a random vector with a probability. *Problems of Information Transmission*. 1981, 17, s. 171–178.

[19] KOZACHENKO, L. F. – LEONENKO, N. N. On statistical estimation of entropy of random vector. *Problems of Information Transmission*. 1987, 23, 2, s. 95–101.

[20] KRASKOV, A. – STÖGBAUER, H. – GRASSBERGER, P. Estimating Mutual Information. *Physical Review E*. 2004, 69, 6 pt. 2.

[21] KUGLIN, C. D. – HINES, D. C. The Phase Correlation Image Alignment Method. In *Proceedings of the IEEE 1975 International Conference on Cybernetics andSociety*, s. 163–165, September 1975.

[22] KYBIC, J. *Elastic Image Registration Using Parametric Deformation Models*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, 2001.

[23] LEONENKO, N. – PRONZATO, L. – SAVANI, V. *A class of Rényi information estimators for multidimensional densities*. Technical report, Laboratoire I3S, Sophia Antipolis, 2005.

[24] LOWE, D. G. Object recognition from local scale-invariant features. In *Proceedings of the International Conference on Computer Vision*, 2, s. 1150, 1999.

[25] MAINTZ, J. – VIERGEVER, M. A survey of medical image registration. *Medical Image Analysis*. 1998, 2, 1, s. 1–36.

[26] MILLER, E. G. A new class of entropy estimators for multi-dimensional densities. In *Proceedings of ICASSP2003*, 2003.

[27] NEEMUCHWALA, H. F. – HERO, A. Image registration in high-dimensional feature space. In *Proceedings of the SPIE Conference on Electronic Imaging*, s. 99–113, 2005.

[28] NEŠETŘIL, J. – MILKOVÁ, E. – NEŠETŘILOVÁ, H. Otakar Boruvka on Minimum Spanning Tree Problem: Translation of Both the 1926 Papers, Comments, History. *DMATH: Discrete Mathematics*. 2001, 233.

[29] NIELSEN, L. K. *Elastic Registration of Medical MR Images*. PhD thesis, University of Bergen, January 2003.

[30] NYQUIST, H. Certain Factors Affecting Telegraph Speed. *Bell Systems Technical Journal*. 1924, 3, s. 324–352.

[31] PREPARATA, F. P. – SHAMOS, M. I. *Computational Geometry - An Introduction*. Springer-Verlag, 1985.

[32] RÉNYI, A. On measures of information and entropy. In *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability*, s. 547–561, 1960.

[33] SCOTT, D. W. On optimal and data-based histograms. *Biometrika*. 1979, 66, s. 605–610.

[34] SCOTT, D. W. *Multivariate Density Estimation: Theory, Practice, and Visualization*. Wiley Series in Probability and Statistics. John Wiley & Sons, 1992.

[35] SCOTT, D. W. – SAIN, S. R. Multi-Dimensional Density Estimation. In RAO, C. – WEGMAN, E. (Ed.) *Handbook of Statistics*, 23. Elsevier, 2004.

[36] SHANNON, C. E. A mathematical theory of communication. *Bell Systems Technical Journal*. 1948, 27, s. 379–423, 623–656.

[37] TALAIRACH, J. – TOURNOUX, P. *Co-planar Stereotaxic Atlas of the Human Brain*. Thieme Medical Publishers, 1988.

[38] TATE, S. – XU, K. General-purpose spatial decomposition algorithms: experimental results. In *Proceedings of the 2nd Workshop on Algorithm Engineering and Experimentation*, s. 197–216, 2000.

[39] TSAPARAS, P. *Nearest Neighbor Search in Multidimensional Spaces* [online]. [citováno 2007-05-08]. <http://www.cs.helsinki.fi/u/tsaparas/publications/depth.ps>.

[40] VANDEWALLE, P. – SÜSSTRUNK, S. – VETTERLI, M. Superresolution Images Reconstructed from Aliased Images. In *Proceedings of the SPIE Visual Communication and Image Processing Conference*, s. 1398–1405, 2003.

[41] VICTOR, J. D. Binless strategies for estimation of information from neural data. *Physical Review E.* November 2002, 66, 5, s. 051903(15).

[42] VIOLA, P. A. Alignment by maximization of mutual information. Technical report, Massachussets Institute Of Technology, June 1995.

[43] WANG, J. – CHUN, J. Image registration for an imaging system on-board fast moving military vehicle. In *Proceedings of the IEEE National Aerospace and Electronics Conference*, s. 239–243, 2000.

[44] ZITOVÁ, B. – FLUSSER, J. Image registration methods: a survey. *Image and Vision Computing.* 2003, 21, s. 977–1000.

Příloha A

Poznámky k implementaci

V této příloze jsou popsány důležité funkce jednotlivých modulů. Podrobné informace jsou uvedeny v komentářích ve zdrojových souborech.

Program je uložen v adresáři `code/` na přiloženém cd. V adresáři jsou obsaženy následující soubory:

`./Makefile`	Makefile demonstračního programu
`./bin`	
`./src`	adresář se zdrojovými kódy
`./src/example.c`	jednoduchý program demonstrující deklaraci struktur parametrů a volání funkcí
`./src/ada.h`	
`./src/ada.c`	zdrojový soubor a hlavička implementující adaptivní histogram
`./src/histo.h`	
`./src/histo.c`	zdrojový soubor a hlavička implementující histogramový estimátor včetně varianty s vyhlazením histogramu
`./src/imtype.h`	hlavička deklarující vstupní strukturu dat `sImage`
`./src/kd.h`	
`./src/kd.c`	zdrojový soubor a hlavička implementující funkce nad k-d stromy.
`./src/kdens.h`	
`./src/kdens.c`	zdrojový soubor a hlavička estimátoru s jádrovým odhadem, závisí na modulu `matrixops`
`./src/kdest.h`	
`./src/kdest.c`	zdrojový soubor a hlavička nearest-neighbor estimátoru, závisí na modulu `kd`

`./src/matrixops.h`	
`./src/matrixops.c`	modul funkcí nad maticemi (inverze, determinant)
`./src/misc.h`	
`./src/misc.c`	modul s chybovými hlášeními estimátorů
`./src/read.h`	
`./src/read.c`	implementace generátorů náhodných dat a funkcí načítajících data ze souborů

Soubor example.c

Soubor `example.c` uvádí příklad užití modulů s estimátory jiným programem. V souboru jsou inkludovány nezbytné hlavičky (hlavička `imtype.h` definující datovou strukturu, s níž estimátory pracují, a hlavičkové soubory některých estimátorů). Na počátku souboru je také inkludován hlavičkový soubor `read.h`. V modulu `read` jsou implementovány některé generátory náhodných dat a funkce pro čtení dat ze souborů.

Ukázkový program po spuštění vygeneruje data (závislá na inicializaci parametrů v souboru `example.c`) a odhadne jejich entropii různými estimátory. Ukázkový program lze zkompilovat příkazem `make`. Makefile podporuje cíle `all` a `clean`.

Soubor ada.c

Globální funkcí modulu `ada` je funkce

- `double adaptiveMI(sImage * im1, sImage * im2, sParamAda * par)`

Důležité vnitřní funkce souboru `ada.c` jsou:

- `void histInit(sImage * i1, sImage * i2, int r, int s, double delta, int minLeaf)` – funkce alokuje paměť pro globální histogramovou strukturu a pomocné proměnné. Funkce zapíše hodnoty parametrů do histogramové struktury,
- `int partitionH(sAdaTreeNode * node, int r, char mode)` – funkce je volána nad každou buňkou histogramového stromu. Funkci je předán parametr určující počet dělení v každé dimenzi. Za účelem vyhledávání kvantilů v datech volá funkci `searchb`, implementující binární vyhledávání. Na konci běhu je proveden χ^2 test nad dceřinnými buňkami a vrácena hodnota na základě jeho výsledku.
- `void getMI(sAdaTreeNode * root)` – funkce prochází histogram a sčítá příspěvky terminálních buněk histogramu k celkové vzájemné informaci.

- `int searchb(double *** array, int N, double *** key, int dimension)` – implementace binárního vyhledávání.

Soubor histo.c

Globální funkcí modulu jsou

- `double histH(sImage * im1, sParamHist * par)` – funkce alokuje paměť, volá funkci `initHist` pro inicializaci histogramu, funkci `fillHist` pro naplnění histogramu, počítá entropii a uvolní paměť.
- `double histMI(sImage * im1, sImage * im2, sParamHist * par)` – viz. výše, funkce dále volá `histH` nad marginálními soubory dat.

Důležité lokální funkce jsou

- `void initHist()` – inicializace histogramové struktury, výpočet velikosti binů a jejich počtu.
- `void fillHist()` – naplnění histogramu vzorky. Pokud je zvoleno vylepšení histogramu oknem, vkládá vzorky pomocí funkce `parIns`.

Soubor kd.c

Modul implementuje operace nad k-d stromy. Globálními funkcemi jsou:

- `extern void kdBuild(sKdTree * kdbase)` – konstruuje k-d strom nad inicializovanou strukturou typu `sKdTree`. Využívá funkcí `getMaxVarDim` pro určení dimenze s nejvyšším rozptylem dat a `splitData` pro rozdělení obsahu uzlu.
- `double kdNNdist(sKdNode * node, double ** target, int queryIsInSet, int BBFlimit)` – vyhledává nejbližší sousedy metodou BBF. Parametr `queryIsInSet` udává zda je referenční bod obsažen v k-d stromu, tj. zda je nutno jej vyřadit z množiny prohledávaných bodů. Funkce v průběhu vyhledávání vkládá ukazatele na uzly k-d stromu společně s jejich vzdáleností od referenčního bodu do prioritní fronty pomocí funkce `insertInQueue` a vyjímá pomocí `retrieveFromQueue`.
- `double ** kdRemovePoint(sKdNode * root, double ** point)` – odstraní zadaný bod z k-d stromu a aktualizuje informaci o hranici uzlu. Pokud se těsná hranice uzlu změnila, propaguje změny směrem ke kořenu k-d stromu.

Důležité lokální funkce:

- `void pruneQueue(double dist)` – funkce odstraní z prioritní fronty uzly se vzdáleností větší než je parametr funkce.
- `sKdNode * findLeaf(sKdNode * node, double ** query)` – nalezne list k-d stromu obsahující zadaný bod.
- `double ** findNN(sKdNode * node, double ** query)` – nalezne nejbližší bod k referenci v listu `node`.

Soubor kdens.c

Globální funkce modulu:

- `double kdensH(sImage * img1)` – funkce počítá odhad hustoty pravděpodobnosti pro všechny body. Nejprve je odhadnuta kovarianční matice dat. Funkce pak volá `scaleCM` k výpočtu kovarianční matice jádra z kovarianční matice dat a funkci `MVN` k výpočtu příspěvků bodů k výsledné hustotě pravděpodobnosti.
- `double kdensMI(sImage * img1, sImage * img2)` – viz výše. Funkce volá pro výpočet marginálních entropií funkci `kdensH`.

Soubor kdest.c

Globální funkce modulu:

- `double kdNnMI(sImage * i1, sImage * i2, sParamTree * par)` – funkce počítá vzájemnou informace mezi daty z `i1` a `i2` přes délku grafu nejbližších sousedů. Funkce využívá vnitřní funkce modulu `kd` a funkci `kdNnH`.
- `double kdNnH(sImage * i1, sParamTree * par)` – výpočet entropie vyhledáváním nejbližších sousedů. Funkce využívá vnitřní funkce modulu `kd`.
- `double kdMstMI(sImage * i1, sImage * i2, sParamTree * par)` – Funkce počítá α vzájemnou informaci mezi daty `i1` a `i2` pomocí délky minimální kostry úplného grafu nad vzorky. Funkce využívá vnitřní funkce modulu `kd` a funkci `kdMstH`..
- `double kdMstH(sImage * i1, sParamTree * par)` – Výpočet α-entropie pomocí délky minimální kostry úplného grafu nad vzorky. Funkce využívá vnitřní funkce modulu `kd`.

Soubor matrixops.c

- `void inverse(double ** a, double ** e, int n);` – funkce počítá inverzní matici k čtvercové matici `a` o `nxn` prvcích. Funkce předpokládá `e` matici identity stejného rozměru.
- `double ** multiply(double ** a, double ** b, double ** c,int m, int n, int o)` – násobení matic `a` a `b`. Výsledek je uložen do matice `c`. Funkce předpokládá rozměry matice `a` `oxm` a rozměry matice `b` `nxo`.
- `double determinant(double ** a, int n)` – funkce počítá determinant matice `a` o rozměrech `nxn` Gaussovou eliminační metodou.

Soubor misc.c

- `void error(int errcode)` – funkce chybových hlášení estimátorů.

Soubor read.c

Globální funkce modulu:

- `sImage * testUniform(double lowerbound, double upperbound, int samples, int dimension)` – generátor dat rovnoměrného rozložení. Parametry `lowerbound` a `upperbound` vymezují rozsah. Parametr `samples` určuje počet generovaných vzorků a parametr `dimension` dimenzi dat.
- `sImage * testNormal(double mean, double sigma, int quality, int samples, int dimension)` – generátor dat normálního rozložení s diagonální kovarianční maticí. Parametr `sigma` určuje směrodatnou odchylku v jednotlivých dimenzích. Parametr `samples` určuje počet generovaných vzorků a parametr `dimension` dimenzi dat. Vzorky jsou generovány z rovnoměrného rozdělení díky platnosti centrální limitní věty; parametr `quality` určuje počet realizací náhodné proměnné s rovnoměrným rozdělením na jeden vzorek.
- `sImage * imgRead(char jmeno[])` – načítání obrázků formátu 24-bitový Windows Bitmap bez komprese.
- `sImage * textRead(char jmeno[]);` – načítání dat z textového souboru. Funkce předpokládá následující podobu textového souboru (znak _ označuje mezeru):

```
<číslo typu float>_<číslo typu float>  ... <číslo typu float>\n
<číslo typu float>_<číslo typu float>  ... <číslo typu float>\n
.
.
.
<číslo typu float>_<číslo typu float>  ... <číslo typu float>\n
#EOF
```

Funkce interpretuje sloupce jako dimenze a řádky jako vzorky. Funkce užívá vestavěných funkcí formátovaného čtení jazyka C. Předpokládány jsou konce řádků unixového typu.

Příloha B

Obsah přiloženého CD

Obsah přiloženého CD je následující:

Adresář na CD	**Popis**
/text	Adresář obsahující text diplomové práce ve formátech PostScript a PDF.
/code	Adresář obsahující podadresáře a soubory programu. V tomto adresáři je uložen Makefile demonstračního programu.
/code/bin	Do tohoto adresáře se kompiluje ukázkový program.
/code/src	Adresář se zdrojovými a hlavičkovými soubory programu.