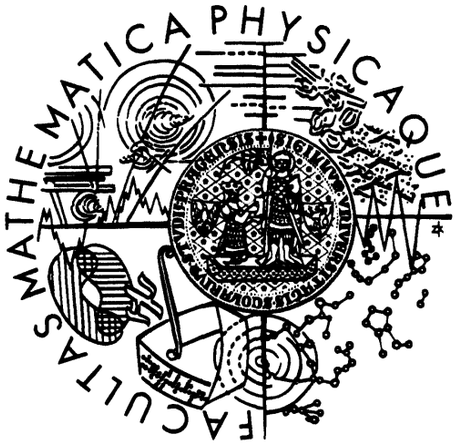
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE



Vojtěch Hlaveš

Metody srovnávání pravidel získaných z dat

Katedra softwarového inženýrství

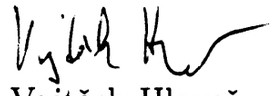
Vedoucí diplomové práce: Ing. RNDr. Martin Holeňa, CSc.

Studijní program: Informatika

2007

Na tomto místě bych chtěl poděkovat svému vedoucímu diplomové práce Ing. RNDr. Martinu Holeňovi, CSc. za cenné rady a připomínky, které přispěly k úspěšnému dokončení této práce.

Prohlašuji, že jsem svou diplomovou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.



V Praze dne 9. dubna 2007 Vojtěch Hlaveš

Obsah

Název práce: Metody srovnávání pravidel získaných z dat

Autor: Vojtěch Hlaveš

Katedra: Katedra softwarového inženýrství

Vedoucí diplomové práce: Ing. RNDr. Martin Holeňa, CSc.

E-mail vedoucího: martin@cs.cas.cz

Abstrakt: Dobývání znalostí z dat patří k nejrychleji se rozvíjejícím informačním technologiím. Jedním z nejpoužívanějších způsobů strukturovaného vyjádření znalostí jsou speciální typy tvrzení v nějaké logice, označované jako pravidla. Jelikož přístupy založenými na různých teoretických principech lze i ze stejných dat extrahovat naprosto rozdílné množiny pravidel, je žádoucí mít k dispozici metody, které budou umět tyto soubory pravidel měřit a srovnávat.

Příspěvkem k rozvoji těchto metod je i tato práce. Nejprve je předveden nový univerzální způsob měření souborů pravidel různých typů, pro který je ukázáno, že rozšiřuje metodu měření souborů klasifikačních pravidel, která byla již známa. Dále je provedeno zobecnění ROC prostorů, které je využito k demonstrování navrženého způsobu měření souborů pravidel při zkoumání závislosti kvality souborů pravidel na různých parametrech použitých metod na dobývání znalostí.

Klíčová slova: Dobývání znalostí, Pravidla, Soubory pravidel, Míry kvality

Title: Methods for comparing rules extracted from data

Author: Vojtěch Hlaveš

Department: Department of Software Engineering

Supervisor: Ing. RNDr. Martin Holeňa, CSc.

Supervisor's e-mail address: martin@cs.cas.cz

Abstract: Data mining has been rapidly developing recently. There is a great variety in types of structured knowledge, but the representation as a sentence of some formal logic is one of the most common. This structure is usually called rules. Because different methods usually produce different sets of rules, the importance of measures which can compare quality of sets of extracted rules grows.

This thesis participates in developing such measures. Firstly, a new and universal approach to measure whole sets of rules is demonstrated and it is shown that this apporach extends an already known method of measurement sets of classification rules. Further, ROC spaces are generalized and used to present the proposed approach, which also reveals dependence quality of sets of rules on various parameters of methods extracting data.

Keywords: Data mining, Rules, Sets of rules, Measures of quality

Kapitola 1

Dobývání znalostí z dat

Jedna z definic dobývání znalostí říká, že dobývání znalostí je netriviální proces získávání platných, nových, potenciálně využitelných a hlavně pochopitelných znalostí [15]. Tedy takových, které mohou sloužit i k učinění závažných rozhodnutí. Netriviální v definici říká, že za vlastním dobýváním musí být komplexní proces, platnost zaručuje platnost vztahů i pro nová data, novost požaduje dříve neznámé vztahy, využitelnost zajišťuje smysl vlastního procesu a pochopitelnost dává uživatelům možnost vztahy uplatnit. Uveďme ještě definici znalosti. Pro množinu faktů (vstupní data) F, jazyk L a míru jistoty C, je znalost tvrzení S v jazyce L, které popisuje vztah mezi podmnožinami F s jistotou C takovou, že S je v nějakém smyslu jednodušší než výčet všech faktů z F [15].

1.1 Historie

Počátky této informační technologie sahají až do 70. let minulého století. K jejímu rozvoji dal podnět především vznik databázových systémů a ohromný nárůst dat, která byla dostupná a která bylo možné analyzovat. Na začátku se používal ke zkoumání dat vždy pouze jeden ze dvou přístupů, založených na různých principech. Prvním byla explorativní analýza dat, jejímž principem je hledání hypotéz platných v datech, a druhý přístup byl založen na strojovém učení, které se snaží napodobovat lidský proces učení. Postupně se však začaly objevovat tendence využívat obou přístupů současně, což vyvrcholilo vznikem nové disciplíny - dobýváním znalostí z dat.

K vyhranění této disciplíny od ostatních, přispělo i uspořádání konference Knowledge Discovery in Databases (KDD-89) na přelomu 80. a 90.

let. Konference KDD-89 byla vcelku úspěšná, avšak původní záměr, aby se tato disciplína jmenovala shodně s touto konferencí nevyšel. V praxi se zažil termín 'Data Mining' [16].

V Česku se začala utvářet tato disciplína také v 70. letech 20. století, kdy se objevila jedna z prvních metod na dobývání znalostí z dat - metoda GUHA. Jako název se v Česku vžilo spojení 'Dobývání znalostí z dat'.

Dnes se již s dobýváním znalostí setkáme téměř všude. V bankovnictví ho lze využít ke zjišťování, zda klientovi poskytnout půjčku či nikoliv, v pojišťovnictví slouží k odhalování pojistných podvodů, ve zdravotnictví se objevují závislosti a příčiny různých chorob. Také metod sloužících k dobývání znalostí je dnes mnoho. Všechny však mají společné, že na vstupu očekávají data, které se budou analyzovat, a výstupem je množina znalostí.

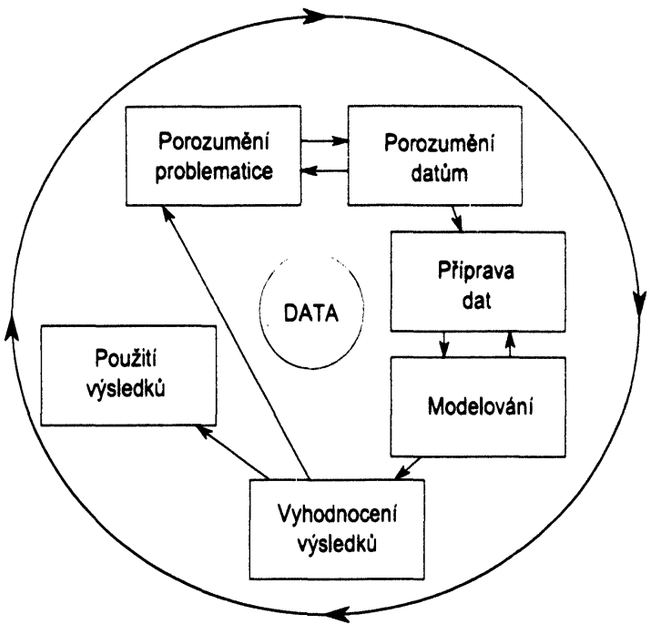
1.2 Proces dobývání znalostí z dat

Proces dobývání znalostí z dat není jen vlastní získávání zajímavých znalostí, ale zahrnuje i například přípravu dat a následné vyhodnocení výsledku. Tyto činnosti, které jsou k vlastnímu získávání znalostí nutné, mnohdy zabírají dohromady více času než samotné dolování. Celý proces se nechá snadno rozložit na více dílčích částí, které se běžně, při každé úloze dobývání znalostí, opakují. Tyto úlohy se pak nechají popsat a zařadit na příslušné místo v celém procesu. Příkladem může být CRISP-DM (CRoss-Industry Standard Process for Data Mining), který vznikl v rámci Evropského výzkumného projektu.

Jak je z obrázku 1.1 patrné, nemusí být celý proces přímočarý. Vše začíná u porozumění problematiky, tedy pochopení, proč se úloha bude provádět a jaké jsou očekávány výsledky. Následuje sběr vlastních dat a pochopení základních vlastností různých atributů a jejich významů.

Příprava dat bývá mnohdy nejdelší částí celého procesu a nejde v ní jen o samotné získání dat z různých databází a jejich pouhou transformaci do podoby, kterou očekává vlastní aplikace na dobývání znalostí.

I když je v dnešní době dostupné ohromné množství dat, nutně to automaticky neznamená možnost získání kvalitnějších znalostí. Je zřejmé, že obecně platné znalosti se z mála informací budují obtížně. Vždy totiž hrozí, že daný vzorek je pouze nějakým speciálním případem. Na druhou stranu podstatná většina souborů dat v sobě obsahuje různé chyby. Může jít o nevyplněné hodnoty některých atributů nebo dokonce mohou být tyto hodnoty chybné. Takovéto nesystematické chyby se pak nazývají šumem [18]. Na jeho odstranění se pracuje ve fázi přípravy dat. Avšak z důvodu, že jeho odstranění

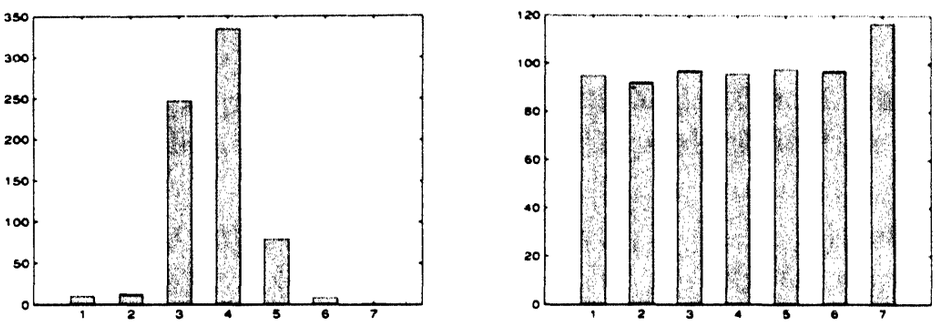


Obrázek 1.1: Životní cyklus CRISP-DM [2]

nebývá dokonalé, většina aplikací na dobývání znalostí již s možností výskytu šumu počítá a jejich výpočty jsou podle toho upraveny.

Druhým častým problémem jsou spojitá data, jelikož některé aplikace vyžadují pouze hodnoty diskrétní. Algoritmů na diskretizaci vstupních dat je také mnoho. Patrně nejjednodušším je rozdělení na intervaly o stejné nebo přibližné délce (ekvidistanční intervaly), neboť stačí znát rozsah hodnot atributů. Při ekvifrekvenčním rozdělení se původní rozsah atributů rozdělí na intervaly tak, aby do každého spadal přibližně stejný počet objektů. Výsledkem budou diskrétní hodnoty odpovídající informaci, do kterého intervalu původní hodnota patří (obrázek 1.2).

Samotné získávání znalostí spočívá v použití vlastní analytické procedury na již připravená data. Uživatel má obvykle možnost do tohoto výpočtu zasahovat určením hodnot některých parametrů. Jejich počet závisí na zvolené proceduře a kombinace těchto parametrů se nazývá nastavení. Výstup procedury budeme nazývat souborem znalostí. Vyhodnocení výsledků pak obnáší zkoumání získaných znalostí. Dochází k odstraňování znalostí, které jsou již známé, a také ke kontrole, že výsledek odpovídá zadání. Procesu odstraňování znalostí ze souboru znalostí budeme říkat prořezávání.



Obrázek 1.2: Ekvidistanční a ekvifrekvenční rozdělení hodnot Body Mass Index(BMI) do intervalů

1.3 Pravidla

Jednotlivé metody dobývání znalostí se od sebe liší reprezentací vlastních znalostí. Toto však není jediný rozdíl. Další rozdíly tvoří například vhodnost dané metody pro daná data nebo srozumitelnost nalezených znalostí pro uživatele. Reprezentace znalostí může být založena na klasifikačních nebo asociačních pravidlech, na klasifikačních stromech, na shlucích, regresní funkci či jiné struktuře. Většina těchto reprezentací je přímo spojena s určitou skupinou metod pro dobývání znalostí z dat. Příkladem může být regresní funkce pro regresní analýzu, shluky pro shlukovou analýzu apod. Avšak existuje zde jedna výjimka - reprezentace ve formě tvrzení nějaké formální logiky. Tato reprezentace, nazývaná stručně 'pravidla', se používá v různých metodách založených na zcela různých principech.

Definice 1 *Pravidlem budeme nazývat trojici*

$$<\sim, \varphi, \psi>$$

obvykle zapisovanou jako

$$\varphi \sim \psi \tag{1.1}$$

kde φ a ψ jsou formule v nějaké logice a $\sim$ je symbol jazyka této logiky. φ se nazývá antecedent, ψ consequent a $\sim$ udává vztah mezi nimi.

Souboru znalostí obsahující pravidla budeme říkat soubor pravidel. Pravidla dostávají podle vztahu mezi antecedentem a consequentem, a někdy i podle tvaru consequentu, různá jména.

1.3.1 Asociační a klasifikační pravidla

Mezi nejčastější typy pravidel se řadí asociační a klasifikační pravidla [15]. Označme vstupní data o_1,o_2,...o_n a budeme dále předpokládat, že $n \geq 1$.

Definice 2 *Určení přípustné hodnoty atributu* $A_j = \langle a_j, r_j, v_j \rangle$, *kde* a_j *udává atribut,* r_j *je operátor z* $\{<, >, =, \leq, \geq, \in\}$ *a* v_j *je hodnota, resp. množina hodnot v případě operátoru* $\in$, *je splněno pro objekt* o_i *ze vstupní datové sady právě tehdy, když pro hodnotu atributu* a_j *objektu* o_i *platí* r_j v_j.

Definice 3 *Asociační pravidlo je trojice*

$$\left\langle \rightarrow_{c,s}, \bigwedge_i A_{1i}, \bigwedge_j A_{2j} \right\rangle \tag{1.2}$$

kde $0 < s, c \leq 1$ *a* $A_{1i} = \langle a_{1i}, r_{1i}, v_{1i} \rangle$ *a* $A_{2j} = \langle a_{2i}, r_{2i}, v_{2i} \rangle$ *jsou určení přípustné hodnoty atributu, pro které platí* $\forall_j \forall_i a_{1i} \neq a_{2j}$. *Asociační pravidlo je splněno právě, když počet objektů, pro které je splněn antecedent i consequent, je větší nebo rovný* $s * n$ *a počet objektů, pro které je splněn antecedent i consequent je alespoň c krát počet objektů, pro které je splněn antecedent.*

Pravidlo asociační se obvykle zapisuje ve tvaru

$$A_{11} \wedge A_{12} \wedge ... \wedge A_{1n} \rightarrow_{c,s} A_{21} \wedge A_{22} \wedge ... \wedge A_{2m} \tag{1.3}$$

Pravidlo rozhodovací má v antecedentu pouze jeden atribut - třídu do které jsou data splňující antecedent pravidlem zařazena.

Definice 4 *Rozhodovací pravidlo je trojice*

$$\left\langle \rightarrow, \bigwedge_i A_i, C \right\rangle \tag{1.4}$$

kde $A_i = \langle a_i, r_i, v_i \rangle$ *a* $C = \langle a, =, v \rangle$ *jsou určení přípustné hodnoty atributu, pro které platí* $\forall_i a_i \neq a$, *a v je prvek konečné množiny* ξ, *kterou budeme nazývat množinou tříd. Symbol* $\rightarrow$ *značí implikaci ve výrokové logice.*

I pro rozhodovací pravidlo uvádíme jeho obvyklý tvar

$$A_1 \wedge A_2 \wedge ... \wedge A_n \rightarrow C \tag{1.5}$$

Definice 5 *Pravidla ze souboru pravidel, který obsahuje pouze pravidla ve tvaru (1.4) a pro který platí*

1. *Každá dvě pravidla p_i a p_j z daného souboru pravidel splňují: pokud existuje objekt o, pro který je splněn antecedent pravidla p_i a zároveň i antecedent pravidla p_j, pak $i = j$. Tuto vlastnost budeme nazývat jednoznačností.*

2. *Pro každý objekt o z prostoru atributů existuje v souboru pravidel pravidlo, jehož antecedent je pro o splněn.*

budeme nazývat klasifikačními pravidly a metody, které získávají soubory klasifikačních pravidel, budeme nazývat klasifikátory. Consequent pravidla p_i z první podmínky bude udávat třídu předpovídanou tímto souborem pravidel pro objekt o, kterou budeme označovat $\hat{y}_i$.

O rozhodovacím nebo asociačním pravidle, jehož antecedent je pro objekt o_i splněn, budeme říkat, že pokrývá objekt o_i.

Rozhodovací pravidla z jednoho souboru pravidel, přesněji jejich antecedenty, se nechají převést do disjunktivní normální formy (DNF). Formule v DNF je taková, která se skládá pouze z disjunkcí konjunkcí literálů nebo jejich negace. Po převodu budou všechny antecedenty pravidel se stejným consequentem sloučeny do jednoho pravidla. Dále budeme těmto pravidlům říkat pravidla v DNF, s ohledem na to, že tato malá nepřesnost snad povede k jednodušší terminologii.

Definice 6 *Rozhodovací pravidlo v DNF má tvar*

$$\left\langle \rightarrow, \bigvee_i \varphi_i, C \right\rangle \tag{1.6}$$

kde φ_i jsou antecedenty všech těch pravidel z jednoho souboru rozhodovacích pravidel, jejichž consequent tvoří C.

Po převedení všech původních pravidel, bude nových pravidel v DNF nejvýše tolik, kolik je tříd.

1.3.2 GUHA pravidla

Další typ pravidel byl definován v metodě GUHA. Tato pravidla jsou v podobě tvrzení observační logiky, což je booleovská predikátová logika se zobecněnými kvantifikátory. Pro vstupní data o n objektech je pravdivostním ohodnocením booleovského predikátu φ, resp. ψ, na těchto datech vektor $\|\varphi\| \in \{0,1\}^n$, resp. $\|\psi\| \in \{0,1\}^n$, avšak pravdivostním ohodnocením $\varphi \sim \psi$ je hodnota

$$\|\varphi \sim \psi\| = \mathrm{Tf}_{\sim}(\|\varphi\|, \|\psi\|)$$

kde $\mathrm{Tf}_{\sim}$ označuje 0,1-hodnotovou funkci na množině čtyřpolních čtvercových matic (tabulka 1.1), nazývanou pravdivostní funkce kvantifikátoru $\sim$ [12].

$\sim$	ψ	$\neg\,\psi$
φ	a	b
$\neg\,\varphi$	c	d

Tabulka 1.1: Čtyřpolní kontingenční tabulka

Hodnota a určuje počet objektů, pro které je splněno φ i ψ, b počet objektů pro které je φ splněno a ψ nesplněno, c udává počet objektů pro které je splněno ψ avšak není splněno φ a hodnota d je počet objektů které nesplňují ani φ ani ψ. Hodnota n bude rovná součtu hodnot a,b,c a d.

Konkrétním příkladem kvantifikátoru je fundovaná implikace $\Rightarrow_{p,Base}$ pro $0 < p \leq 1$ a $Base > 0$. Pro jeho pravdivostní funkci platí

$$\mathrm{Tf}_{\Rightarrow_{p,Base}} = \begin{cases} 1 & \text{pokud } \dfrac{\mathrm{a}}{\mathrm{a}+\mathrm{b}} \geq \mathrm{p} \ \wedge \ \mathrm{a} \geq \mathrm{Base} \\ 0 & \text{jinak} \end{cases} \tag{1.7}$$

Pravidlo s tímto kvantifikátorem odpovídá asociačnímu pravidlu [1], s tím malým rozdílem, že u asociačních pravidel se místo $a \geq Base$ uvažuje $\frac{a}{n} \geq Base$. Pro zjednodušení budeme místo $\|\varphi \sim \psi\|$ psát $\sim(a,b,c,d)$, kde a, b, c a d odpovídají hodnotám ze čtyřpolní tabulky spočítané pro daný antecedent φ a consequent ψ. Fundovaná implikace, jakožto funkce na čtyřpolní tabulce, se nechá zařadit do skupiny implikačních kvantifikátorů [19].

Definice 7 *Kvantifikátor je implikační jestliže z* $\sim(a,b,c,d) = 1$ *a z platnosti* $a' \geq a \wedge b \leq b'$ *plyne* $\sim(a',b',c',d') = 1$ *pro každou dvojici čtyřpolních tabulek* $\langle a,b,c,d\rangle$ *a* $\langle a',b',c',d'\rangle$.

Další kvantifikátor, který patří do této skupiny, je dolní kritická implikace $\Rightarrow^{!}_{\theta,\alpha}$ s prahem $\theta \in \langle 0,1\rangle$, založená na testu hypotézy $p_{\psi|\varphi} \leq \theta$ proti alternativě $p_{\psi|\varphi} \geq \theta$ pomocí binomiálního testu na hladině významnosti α, pro kterou platí

$$\mathrm{Tf}_{\Rightarrow^{!}_{\theta,\alpha}} = \begin{cases} 1 & \text{pokud } \sum_{i=a}^{a+b} \binom{a+b}{i} \theta^{i}(1-\theta)^{a+b-i} \leq \alpha \\ 0 & \text{jinak} \end{cases} \tag{1.8}$$

Definice 8 *Kvantifikátor je ekvivalenční jestliže* $z \sim (a,b,c,d) = 1$ *a z platnosti* $a' \geq a \wedge b' \leq b \wedge c' \leq c \wedge d \geq d'$ *plyne* $\sim (a',b',c',d') = 1$ *pro každou dvojici čtyřpolních tabulek* $\langle a,b,c,d\rangle$ *a* $\langle a',b',c',d'\rangle$.

Příkladem ekvivalenčního kvantifikátoru je fundovaná ekvivalence $\leftrightarrow_{p,Base}$ pro $0 < p \leq 1$ a $Base > 0$, pro kterou platí

$$\mathrm{Tf}_{\leftrightarrow_{p,Base}} = \begin{cases} 1 & \text{pokud } \frac{a+d}{a+b+c+d} \geq p \ \wedge\ a \geq Base \\ 0 & \text{jinak} \end{cases} \tag{1.9}$$

Do této skupiny patří i Fisherův kvantifikátor, založený na jednostranném Fisherově faktoriálovém testu nezávislosti hypotézy ohodnocení φ a ψ, jehož pravdivostní funkce je definována

$$\mathrm{Tf}_{\sim^{F}_{\alpha}} = \begin{cases} 1 & \text{pokud } ad \geq bc \wedge \sum_{i=a}^{\min(a+b,a+c)} \frac{\binom{a+c}{i}\binom{b+d}{a+b-i}}{\binom{a+b+c+d}{a+b}} \leq \alpha \\ 0 & \text{jinak} \end{cases} \tag{1.10}$$

Pravidla v podobě tvrzení observační logiky budeme nazývat GUHA pravidly, případně, pokud nás bude zajímat pouze některá skupina GUHA pravidel, budeme se na ni odkazovat přímo jejím jménem.

1.4 Vybrané metody dobývání pravidel

Vzhledem k velkému rozvoji disciplíny dobývání znalostí dnes existuje mnoho systémů a metod, určených k dobývání znalostí. Zde se zaměříme na ty metody, které byly použity k vlastní práci.

1.4.1 Metoda GUHA

GUHA je zkratkou slov General Unary Hypotheses Automaton. Současná nejznámější implementace této metody je LISp Miner [20].

Hledání pravidel pomocí procedury 4ft-Miner, která je součástí projektu LISp Miner, je založeno na práci s bitovými řetízky. Pro každý objekt v datech je vytvořen jeden bitový řetízek, který má délku rovnu počtu všech možných hodnot všech atributů. Nechť se první atribut nazývá A_1 a nabývá k hodnot. V řetízek bude na i. místě 1 pokud je hodnota A_1 rovna i. hodnotě, na ostatních místech bude 0. Pokud atribut A_1 bude nabývat hodnot z $\{1, 3, 7\}$ a A_2 hodnot z $\{5, 7, 8\}$ budou kusy řetízků vypadat tak, jak je zobrazeno v tabulce 1.2.

	Atributy				Řetízky			
	A_1	A_2	...	A_n	$A_1[1]$	$A_1[2]$	$A_1[3]$	...
r_1	1	5		8	1	0	0	
r_2	3	7		10	0	1	0	
...	...	...		...	...	...	...	
r_m	1	8		8	1	0	0	

Tabulka 1.2: Bitové řetízky

Všechny bitové operace po převedení atributů na řetízky jsou rychlé. Pokud chceme zkoumat případy kdy atribut A_1 nabývá hodnot 1 a 3, stačí spočítat disjunkci $A_1[1] \vee A_1[2] = A_1[1, 2]$. Při zkoumání případu, kdy atribut A_1 nabývá hodnot 1 nebo 3 a zároveň A_2 je 5 nebo 8 se stačí zabývat řádky pro které je $A_1[1, 2] \wedge A_2[1, 3] = 1$. Tudíž výpočet tabulky 1.1 a následné hledání velkého množství pravidel je možné.

Častým jevem je, že ve vstupních datech nejsou některé hodnoty atributů zadány a vyplněny. V metodě LISp Miner 4ft se toto řeší použitím devítipolní tabulky 1.3, která se spočítá na základě dat, a která je následně převáděna na čtyřpolní tabulku různými způsoby. Je možné neznámá data nezahrnovat, přidat hodnoty polí $i + m + o$ k hodnotě pole a a hodnoty $j + p$ k hodnotě v poli d, nebo opačně k b a c.

1.4.2 Metoda AQ

Metoda AQ je metoda strojového učení, jejíž počátky sahají také do 70. let minulého století a jejím výstupem jsou soubory rozhodovacích pravidel.

$\sim$	ψ	$?\psi$	$\neg\ \psi$
φ	a	i	b
$?\varphi$	o	m	p
$\neg\ \varphi$	c	j	d

Tabulka 1.3: Devítipolní kontingenční tabulka

Generování pravidel začíná výběrem jednoho objektu ze zkoumané třídy. Objekty patřící do této třídy budeme nazývat 'pozitivní případy', ostatní 'negativní'. Pro vybraný pozitivní případ se zkonstruuje pravidlo, které pokrývá právě tento případ. Toto pravidlo se opakovaně rozšiřuje pomocí operátoru 'extension-against' ('proti-rozšíření') použitého na zvolený pozitivní případ a postupně pro každý negativní případ. Výsledkem budou maximální rozšíření původního pravidla pokrývající vybraný pozitivní případ takové, že vždy nebudou pokrývat jeden negativní případ. Průnikem těchto rozšířených pravidel vzniká pravidlo, které se zařadí do předběžné výstupní množiny pravidel. Tomuto průniku se v literatuře říká hvězda. Všechny případy, které získané pravidlo pokrývá, se odstraní a začne se opět od výběru pozitivního případu ze zbylých případů [13].

Pravidla, která jsou zahrnuta do předběžné výstupní množiny pravidel jsou dále optimalizována (zobecňována). K tomu slouží operátory odstraňování podmínek z pravidla, uzavírání intervalů nebo jejich rozšiřování. Při žádné operaci však nesmí pravidlo pokrýt negativní případ. K výběru nejlepšího pravidla ze zobecněných pravidel slouží tzv. LEF (Lexicographical Evaluation Functional, 'lexikografický vyhodnocovací funkcionál'), ve které je možné klást požadavky na výstupní pravidla. V poslední fázi mohou být ještě prořezána (odstraněna) celá pravidla. Celý tento postup se ve verzi AQ21, což je poslední implementace této metody [21], označuje jako Theory Formation (TF, 'tvorba teorie').

V AQ21 jsou ještě implementovány dvě modifikace zmíněného algoritmu, které nepožadují, aby pravidla pokrývala pouze a zároveň všechny pozitivní případy. K tomu slouží míra kvality popisu. Mírám kvality pravidel se věnuje následující kapitola, kde lze nalézt i přesnou definici této míry. Uveďme tedy pouze její význam. Tato míra roste (přesněji neklesá), pokud pravidlo pokrývá více pozitivních případů a klesá (resp. neroste) s rostoucím počtem pokrytých negativních případů. První modifikace tzv. Approximate Theory Formation (ATF, 'tvorba přibližné teorie') se od TF liší tím, že z optimalizovaných pravidel se vybírají ta, pro která je míra kvality popisu nejvyšší. Pattern Discovery (PD, 'hledání vzorků') modifikuje již operátor 'extension-

against' a nepožaduje, aby se při zobecňování nepokryly negativní případy. Místo toho se pro možné rozšíření spočítá hodnota míry kvality popisu a pokud toto rozšíření vyhovuje - tj. míra kvality popisu je pro něj vyšší než zvolená hodnota - pracuje se s tímto rozšířením dále, jinak je zavrhnuto [23].

Metoda AQ umožňuje uživateli zasahovat do algoritmu generování pravidel tím, že lze nastavit některé požadavky, jednotlivě poměrně prostá (například minimální počet nově přidaných pozitivní případů při generalizaci pravidla), které mají výstupní pravidla splňovat. Pro zpřehlednění zápisu těchto požadavků byla definována LEF jako posloupnost (uspořádaná n-tice) dvojic

$$< (c_1, \tau_1), (c_2, \tau_2), ..., (c_n, \tau_n) >$$

kde c_i odpovídá kritériu a τ_i toleranci. Jednotlivé požadavky na pravidlo se vyhodnocují postupně. Každá hodnota kritéria se nesmí odlišovat od nejlepšího odhodnocení kritérií v generovaném souboru pravidel o více, než udávají tolerance. Jestliže se pro nějaké kritérium hodnota odlišuje o více, je pravidlo prohlášeno za horší než pravidla, která tuto podmínku splňují. Pokud dvě pravidla nesplňují stejnou podmínku, je za lepší považováno to pravidlo, pro které by stačilo menší zvýšení tolerance, aby již podmínku splňovalo. V případě, kdy dvě pravidla splňují všechny požadavky, jsou z hlediska LEF prohlášeny jako rovnocenné.

1.4.3 Získávání pravidel z fuzzy-neuronové sítě

Pravidla získaná touto metodou jsou generována v Lukasiewiczově, Gödlově nebo produkt-Lukasiewiczově fuzzy logice. Jejich antecedent je vždy v DNF a vztah mezi antecedentem a consequentem, který určuje jednu třídu, udává ekvivalence ve zvolené logice [17].

Fuzzy logika je rozšířením booleovské logiky. Narozdíl od booleovské logiky, ve které se používají pouze dva stupně pravdivosti - 1 pro pravdu a 0 pro nepravdu, je ve fuzzy logice oborem přípustných hodnot pro stupně pravdivosti interval $\langle 0, 1 \rangle$. Fuzzy logiky obsahují, obdobně jako booleovská logika, výrokové proměnné, logické spojky, konstantu $\bar{0}$ s pravdivostní hodnotou nula, axiomy a odvozovací pravidlo.

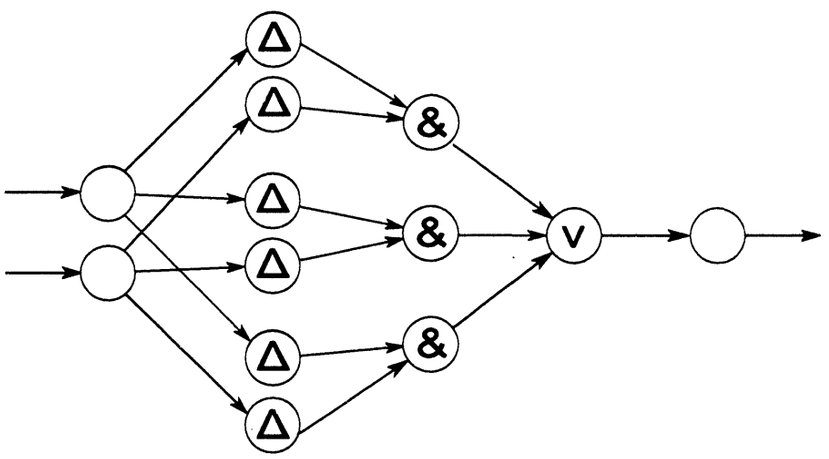
Zmíněné logiky se od sebe liší pravdivostními funkcemi konjunkce i disjunkce, zde nazývanými t-normy a t-konormy (t je zkratka slova trinagular a tyto názvy se užívají převážně z historických důvodů), které jsou v těchto logikách definovány na celém intervalu $\langle 0, 1 \rangle$. Jelikož jsou všechny tyto logiky rozšířením dvouhodnotové logiky, jejich výsledek na hodnotách $\{0, 1\}$ musí

Logika	t-norma	t-konorma
Lukasiewiczova	$max(0, a+b-1)$	$min(1, a+b)$
Gödlova	$min(a, b)$	$max(a, b)$
produkt-Lukasiewiczova	$a * b$	$a+b-a*b$

Tabulka 1.4: t-normy a t-konormy v různých logikách

být stejný jako v booleovské logice. Tabulka 1.4 ukazuje jejich předpis pro vstupní proměnné a a b.

Pravidla jsou získávána z naučené neuronové sítě, která má speciální tvar znázorněný na obrázku 1.3.



Obrázek 1.3: Architektura sítě, trojúhelník značí počítání stupně příslušnosti k fuzzy množině, &, resp. ∨, značí použití t-normy, resp. t-konormy, na vstupy příslušných neuronů

První skrytá vrstva je jediná adaptivní část sítě. Každý neuron z této vrstvy reprezentuje fuzzy množinu. Zvolená funkce příslušnosti, jejíž parametry se při učení přizpůsobují množině dat, přiřadí vstupu míru příslušnosti k dané fuzzy množině. Výstupy jednotlivých neuronů jsou přímo předávány druhé skryté vrstvě a leží tedy v intervalu $\langle 0, 1 \rangle$. Jako příklady funkcí příslušnosti uveďme Gaussovu funkci

$$g(x) = e^{-\left(\frac{x-s}{r}\right)^2}$$

kde s je střed a r rozptyl, sigmoidu

$$sig(x) = \frac{1}{1+e^{-\alpha(x-s)}}$$

a po částech lineární funkci trojúhelník

$$tri(x) = \begin{cases} 0 & \text{když } x \leq t_1 \text{ nebo } x > t_3 \\ \frac{x-t_1}{t_2-t_1} & \text{když } x > t_1 \text{ a současně } x \leq t_2 \\ \frac{t_3-x}{t_3-t_2} & \text{když } x > t_2 \text{ a současně } x \leq t_3 \end{cases}$$

Neurony ve druhé skryté vrstvě spočítají konjunkce (odpovídající t-normě v logice, kterou jsme si zvolili) měr příslušnosti jejich vstupů k fuzzy množinám obsažených v první skryté vrstvě. Počet neuronů této vrstvy se určuje experimentálně a je dán jen tím, kolik shluků vstupní data tvoří. Poslední vrstva obsahuje jen jeden neuron, který spočítá disjunkci (odpovídající t-konormě ve zvolené logice) hodnot na jeho vstupu. Jeho výstup, a tedy i výstup celé sítě, udává stupeň příslušnosti vstupního objektu o_i k fuzzy množině určené v consequentu. Pravdivostní hodnotu výrazu pro pravidlo, které má v consequentu třídu j, budeme pro vstupní objekt o_i značit $\|p_j\|^L_{o_i}$, kde L je zvolená fuzzy logika, a pravdivostní hodnotu negace výrazu pro dané pravidlo budeme značit $\|\neg p_j\|^L_{o_i}$.

Při učení sítě je cílem nalézt hodnoty adaptivních parametrů neuronové sítě tak, aby

$$\sum_i (y_i - \hat{y}_i)^2$$

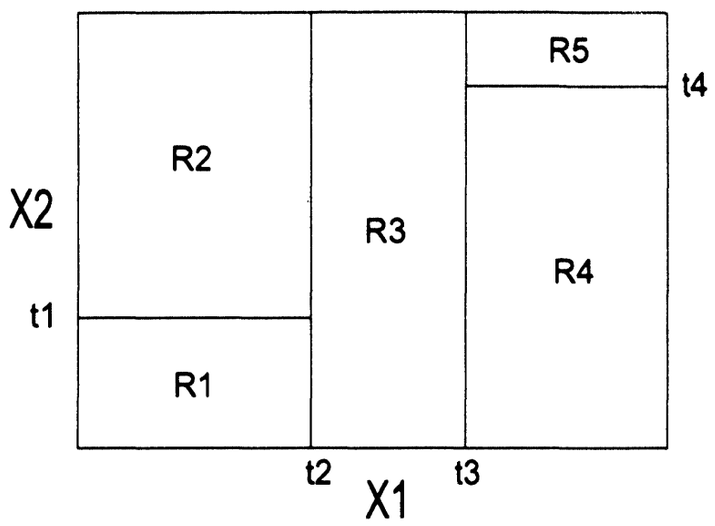
bylo co nejmenší. Výstup sítě pro i-tý objekt je zde označen jako $\hat{y}_i$ a y_i označuje skutečnou třídu vstupního objektu. Konkrétní metody řešení této úlohy jsou popsané v [17].

1.4.4 Klasifikační stromy

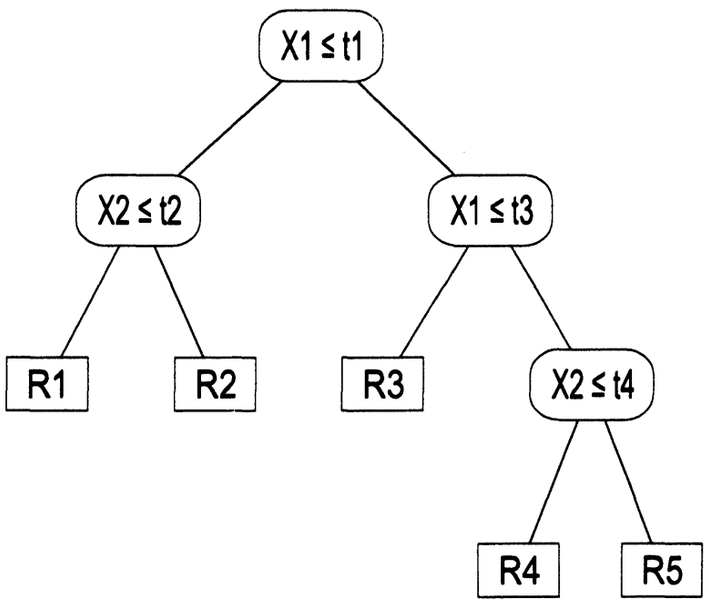
Stromové algoritmy pro klasifikaci se řadí mezi klasifikátory, a tedy pokrývají celý prostor atributů, který rozdělují celý prostor atributů na množinu kvádrů (např. obdélníků pro případ dvou atributů) [11]. Toto dělení není libovolné, ale rekurzivní štěpení vždy jednoho kvádru (obrázek 1.4). Opakovaně se vybírá atribut a hodnota, ve které se provede štěpení, dokud není splněna nějaká podmínka pro ukončení štěpení.

Svoje označení si tato metoda získala tím, že štěpení lze snadno zachytit v binárním stromě (obrázek 1.5) a to i pro případ více atributů. Listy tohoto stromu reprezentují jednotlivé oblasti po provedení štěpení a obsahují vstupní data.

Cílem štěpení uzlu je oddělit od sebe objekty z různých tříd. Budeme předpokládat označení těchto tříd 1,2,...,m. K dispozici jsou vstupní data,



Obrázek 1.4: Rozdělení prostoru dvou atributů



Obrázek 1.5: Binární strom odpovídající štěpení na obrázku 1.4

která jsou zařazena do tříd. Na jejich základě se musí rozhodnout, podle kterého atributu a ve které hodnotě se provede štěpení. Ke zjištění, zda má smysl štěpení provádět v uzlu v slouží tzv. index nečistoty $i(v)$. Označme potomky vzniklé po štěpení uzlu v jako l a r. Podíl objektů, které se po provedení štěpení dostaly do l označme $\pi(l)$, obdobně $\pi(r)$ bude podíl objektů, které se po provedení štěpení dostanou do r. Potom změna indexu nečistoty je

$$\Delta i(v) = i(v) - \pi(l)i(l) - \pi(r)i(r)$$

Pokud jsou hodnoty $\Delta i(v)$ vysoké, znamená to, že navrhované štěpení je dobré a nalezení štěpení s největším $\Delta i(v)$ lze získat vyzkoušením všech potenciálních štěpení.

Jako index nečistoty je možné volit například Gini Index

$$\text{Gini Index} = \sum_{j \neq k} \hat{p}(j|v)\hat{p}(k|v) \tag{1.11}$$

nebo entropii

$$\text{Entropie} = -\sum_{j=1}^{m} \hat{p}(j|v) log_2(\hat{p}(j|v)) \tag{1.12}$$

kde

$$\hat{p}(j|v) = \frac{N_j(v)}{N(v)}$$

a $N(v)$ značí počet objektů v uzlu v a $N_j(v)$ počet objektů, které jsou v uzlu v a patří do třídy j.

Po výběru konkrétního indexu nečistoty můžeme spočítat index nečistoty I celého stromu S

$$I(S) = \sum_{w \in T} N(w)i(w)$$

kde T je množina listů stromu S. Index nečistoty stromu po provedení štěpení s uzlu v je

$$I(S') = \sum_{w \in T-v} N(w)i(w) + N(l)i(l) + N(r)i(r)$$

Snížení indexu nečistoty celého stromu po provedení štěpení je tedy

$$\begin{aligned} \Delta I(S) &= I(S) - I(S') \\ &= i(v)N(v) - i(l)N(v)\pi(l) - i(r)N(v)\pi(r) \\ &= \Delta i(s,v)N(v) \end{aligned}$$

jelikož $N(l) = N(v)\pi(l)$ a $N(r) = N(v)\pi(r)$. Tedy změna indexu nečistoty celého stromu je odvozena pouze od změny indexu nečistoty ve štěpeném uzlu.

Na začátku jsme zmiňovali, že proces štěpení pokračuje až do doby, než je splněna nějaká podmínka. Touto podmínkou může být například nejvyšší počet objektů v listě. Obvykle se volí tato podmínka poměrně hodně přísná, v případě nejvyššího počtu objektů v listě je to malé číslo, což má za následek vytvoření stromu s větším počtem uzlů. Po dokončení konstrukce stromu se přiřadí listům třídy. Každému objektu, který se dostane do listu v, bude přiřazena třída k, jestliže $\hat{p}(k|v) = max_j \hat{p}(j|v)$. Pokud maxima nabývá více než jedna třída, pak se náhodně vybere jedna z nich.

Obvykle ještě následuje fáze, ve které probíhá prořezávání. U stromu s větším počtem uzlů totiž hrozí větší riziko, že se příliš přizpůsobil vstupním datům a mohl by hůře klasifikovat objekty, které ještě neviděl. Při prořezávání jsou odstraněny některé listy a nahrazeny jejich společným předkem. Určit, který uzel v je vhodné prořezat, lze podle hodnot $P(v) = max_j \hat{p}(j|v)N(v)$. Jelikož objekty v uzlu v jsou přiřazeny do třídy, která maximalizuje $\hat{p}(j|v)$, první část udává odhad, že objekty v tomto uzlu budou správně klasifikovány. Druhý faktor určuje velikost uzlu. Malé hodnoty $P(v)$ tedy udávají, že se jedná buď o malý uzel, který příliš neovlivní kvalitu celého stromu nebo se v uzlu v nachází relativně mnoho objektů i ze tříd jiných než je třída přiřazená uzlu v. Při pořezávání se vybere uzel s nejmenším $P(v)$.

Tato stromová reprezentace se převádí na soubor klasifikačních pravidel tak, že jedna větev ve stromě bude odpovídat jednomu pravidlu. Jelikož byl při konstrukci stromu pokryt celý prostor atributů a každý vstupní objekt spadá právě do jednoho listu, bude i výsledný soubor pravidel pokrývat celý prostor atributů a bude jednoznačný.

1.5 Křížová validace

Jelikož nás u pravidel zajímá, jak dobře se budou chovat na datech, která jim budeme předkládat až po vytvoření pravidel samotných, zohledňuje se toto již při jejich vytváření a následném měření. Data jsou rozdělena na dvě disjunktní podmnožiny - na data testovací a trénovací. Data trénovací jsou použita k vytvoření pravidel, na testovacích se pak měří jejich kvalita.

Toto rozdělení na dvě množiny není obvykle libovolné, ale používá se tzv. křížová validace [10]. Data jsou náhodně rozdělena na k přibližně stejně velkých částí a vždy jedna je použita jako data testovací a zbytek jako data

trénovací, výsledky jsou pak zprůměrovány. Tímto postupem se zaručí, že všechna data jsou použita ke konstrukci i testování pravidel, ale přitom nikdy nejsou použita zároveň v obou fázích.

Kapitola 2

Existující míry kvality

Většina existujících měr kvality je definována pouze pro jednotlivá rozhodovací nebo asociační pravidla. Vychází to z jejich použití - výsledkem generování těchto pravidel bývá obvykle objemný soubor pravidel, který není v lidských silách uchopit. Navíc, tento soubor pravidel může v některých případech obsahovat již známé skutečnosti. Je tedy snaha s využitím těchto měr vybrat ze souboru pravidel pouze některá pravidla, pokud možno tak, aby se tím nepřišlo o cenné znalosti, ale aby se tím počet pravidel dostatečně zredukoval.

Míry, které jsou konstruovány na měření souborů pravidel, jsou použitelné pouze na klasifikační pravidla a nelze je na jiný typ pravidel přímo použít. Také nám není známa žádná metoda na porovnávání souborů GUHA pravidel tak, aby byly zohledněny jejich různorodé vlastnosti.

V této kapitole jsou popsány nejdůležitější existující míry.

2.1 Míry založené na čtyřpolní kontingenční tabulce

Jedna z možností, jak ohodnotit jedno pravidlo, tedy jak mu přiřadit nějaké reálné číslo, vychází ze čtyřpolní kontingenční tabulky 1.1.

2.1.1 Základní míry

Mezi nejčastěji používanými mírami odvozenými ze čtyřpolní tabulky jsou konzistence a úplnost [4], jelikož jsou relativně jednoduché, ale jejich vy-

jadřovací schopnost je značná.

$$\text{Konzistence} = \frac{a}{a+b} \tag{2.1}$$

$$\text{Úplnost} = \frac{a}{a+c} \tag{2.2}$$

S jejich výskytem se setkáme v nejrůznějších aplikacích a navíc byly základem pro odvození měr složitějších.

Už při použití těchto dvou měr však nastává problém, byť ne tak závažný. Obvyklým cílem je získat (vybrat) pravidla, jejichž konzistence i úplnost je co nejvyšší. Většinou však při zvýšení jedné dojde k poklesu druhé. Vzniká tedy otázka, zda lze považovat pravidlo dosahující 91% konzistence a 71% úplnost jako lepší než pravidlo mající 98% konzistence a 60% úplnost. K vyřešení tohoto problému může pomoci zavedení měr na těchto dvou mírách nezávislých, jako například přesnost

$$\text{Přesnost} = \frac{a+d}{n} \tag{2.3}$$

nebo podpora

$$\text{Podpora} = \frac{a}{n} \tag{2.4}$$

Míry je možné také upravit nebo zkombinovat [4], příkladem je

$$\begin{aligned}(0.5 + \frac{1}{4} * \text{Úplnost}) * \text{Konzistence} + \\ (0.5 - \frac{1}{4} * \text{Konzistence}) * \text{Úplnost}\end{aligned} \tag{2.5}$$

nebo

$$\frac{n * \text{Konzistence} * \text{Úplnost} - a}{n * \text{Úplnost} - a}$$

Konzistence a podpora jsou využívány při získávání asociačních pravidel. Bylo však ukázáno, že kvůli tomu mohou být asociační pravidla zavádějící [3]. Jako příklad, který toto demonstruje, lze uvést úlohu nákupního košíku a souvislosti nákupu kávy a čaje (uvažujme pro jednoduchost pouze dva atributy Káva a Čaj, opakem $Káva = Ano$ je $Káva = Ne$, obdobně u čaje.

Za situace dané tabulkou 2.1 dosahuje pravidlo

$$\text{Čaj=Ano} \rightarrow \text{Káva=Ano}$$

→	Káva=Ano	Káva=Ne
Čaj=Ano	15	5
Čaj=Ne	70	10

Tabulka 2.1: Čtyřpolní kontingenční tabulka pro případ nákupního košíku

konzistence $\frac{15}{20} = 75\%$, což je poměrně vysoká hodnota. Pokud však užijeme toto pravidlo samostatně, bude to podle [3] zavádějící, protože pravděpodobnost, že si zákazník koupí kávu je 90%.

Mezi míry složitější patří míra navržená v systému AQ, která také kombinuje úplnost a konzistenci.

$$\text{Míra Kvality Popisu}(w) = \text{Úplnost}^{w} * \text{ZK}^{1-w} \tag{2.6}$$

Parametrem $w \in \langle 0, 1 \rangle$ lze klást důraz buď na složku konzistence nebo úplnosti. Míra ZK (Zisk Konzistence) má tvar

$$\text{ZK} = \left(\frac{a}{a+b} - \frac{a+c}{n} \right) * \left(\frac{n}{b+d} \right) \tag{2.7}$$

a je založená na srovnání konzistence pravidla a pravděpodobnosti, že náhodné určení consequentu bude správné [13].

Patrně z důvodů, že vznik většiny měr probíhal současně, nechá se objevit, že některé míry se chovají zcela stejně nebo velmi podobně, i když to nemusí být na první pohled úplně patrné [9].

2.1.2 Zajímavost

Při zkoumáních asociačních a rozhodovacích pravidel se zjistilo, že pravidla, která dosahují nejvyšších hodnot konzistence, jsou obvykle obecně známá, příkladem může být pravidlo

$$\text{BMI(20-25)} \rightarrow \text{Třída=Zdravý pacient}$$

dosahující konzistence 90%, které se nechá objevit v datovém souboru Pima indians diabetes database. Jelikož toto pravidlo nikoho nepřekvapí, je typicky manuálně v poslední fázi procesu dobývání znalostí odstraněno.

Bylo proto otázkou, zda by i toto nešlo automatizovat, což je nutné obzvláště v případě, kdy systém vyprodukuje obrovské množství pravidel [14]. Jedním možným řešením je použití měr zajímavosti pravidla.

Míry zajímavosti lze rozdělit do dvou skupin: objektivní a subjektivní [15].

Objektivní míry zajímavosti mohou být odvozeny například na základě statistické významnosti nalezených znalostí. Mezi tyto míry lze zařadit míru χ^2

$$\chi^2 = \frac{n * (a * d - b * c)^2}{(a+b) * (a+c) * (b+d) * (c+d)} \tag{2.8}$$

Snaha popsat míry zajímavosti vedla ke vzniku tří podmínek [8], které by míry zajímavosti měly splňovat:

- Zajímavost pravidla by měla být 0 v případě, že rozdělení dat pokrytých pravidlem je stejné jako v celé množině dat - tedy jeli $a = (a+b) * (a+c)/n$
- Zvýšení hodnoty a, při stejných hodnotách ostatních polí, by mělo zvýšit i zajímavost pravidla.
- Zajímavost pravidla by měla klesat, pokud dojde k jediné změně a to k poklesu úplnosti nebo přesnosti.

Mírou zajímavosti, která tyto podmínky splňuje je například míra

$$a - \frac{(a+b) * (a+c)}{S}$$

[8] nebo i míra kvality popisu (2.6) pro $w < 1$ [13].

Na druhou stranu subjektivní míry jsou založené na znalostech uživatele dané problematiky a tedy, narozdíl od měr objektivních, nelze postup ohodnocení pravidel podle subjektivní zajímavosti zcela zautomatizovat. Tyto znalosti se rozlišují podle jejich skutečného stavu. Mohou být buď správné, vágní nebo zcela nesprávné [14]. Základní dvě míry jsou neočekávanost, která měří, jak moc je uživatel daným pravidlem překvapen, a použitelnost, která požaduje, aby nalezené znalosti měly pro uživatele nějaký prospěch.

2.2 Míry založené na odhadu pravděpodobnosti

Míry, které slouží k měření kvality klasifikátorů, využívají obou vlastností souborů klasifikačních pravidel [10]. Narozdíl od měr založených na čtyřpolní

tabulce měří celé soubory. Tyto míry jsou založené na odhadu podmíněné pravděpodobnosti, že objekt splňující antecedent A patří do třídy j.

Označme trénovací data bez atributu třídy jako $x_1, x_2, \ldots, x_n$. Podmíněnou pravděpodobnost, že objekt ke klasifikaci x_i bude patřit do třídy $j \in \xi$ budeme značit $f(j|x_i)$. Cílem metody dobývání znalostí je zkonstruovat klasifikátor takový, jehož funkce $\hat{f}$ bude f aproximovat [10]. Definujme ještě $\delta(j|x_i)$, které je rovno 1, pokud objekt x_i doopravdy patří do třídy j, jinak nabývá hodnoty 0. Opět předpokládáme, že třídy jsou označeny $1, 2, \ldots, m$ a skutečnou třídu x_i označíme y_i.

Míry založené na odhadu podmíněné pravděpodobnosti můžeme rozdělit do několika kategorií, podle jejich vlastností. První z nich tvoří míry nesprávnosti (anglicky 'inaccuracy'), které udávají, jak špatně klasifikátor data klasifikuje. Tyto míry mají tvar

$$\text{TNesprávnost} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m} g\left[h(\delta(j|x_i)), h(\hat{f}(j|x_i))\right]$$

Na výběr funkcí g, h je kladen požadavek, aby pro pevné i

$$\sum_j f(j|x_i) g\left[h(\delta(j|x_i)), h(f(j|x_i))\right] \leq \sum_j f(j|x_i) g\left[h(\delta(j|x_i)), h(\hat{f}(j|x_i))\right]$$

tedy, aby nejlepší klasifikátor byl ten, který by určoval pro každý objekt jeho správnou třídu. Nejznámějším zástupcem této kategorie je

$$\text{Nesprávnost}_1 = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\delta(j|x_i) - \hat{f}(j|x_i)\right)^2 \tag{2.9}$$

pro

$$\begin{aligned} h_1(z) &= z \\ g_1(h_1(z_1), h_1(z_2)) &= (h_1(z_1) - h_1(z_2))^2 \end{aligned} \tag{2.10}$$

avšak může mít i tvar

$$\text{Nesprávnost}_2 = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\delta(j|x_i) log(\hat{f}(j|x_i)) = -\frac{1}{n}\sum_{1}^{n} log(\hat{f}(y_i|x_i)) \tag{2.11}$$

pro

$$\begin{aligned} h_2(z) &= z \\ g_2(h_2(z_1), h_2(z_2)) &= -h_2(z_1) * log(h_2(z_2)) \end{aligned} \tag{2.12}$$

Na rozdíl od nesprávnosti, která srovnává $\delta(j|x_i)$ a $\hat{f}(j|x_i)$, nespolehlivost srovnává $f(j|x_i)$ a $\hat{f}(j|x_i)$. V anglickém jazyce se této kategorii říká 'imprecision', avšak z důvodu, že jsme již zavedli míru přesnosti založenou na čtyřpolní tabulce, používáme označení nespolehlivost, abychom předešli nejednoznačnosti.

Střední hodnota $\varphi_{\alpha j}$ při pevném j se nechá vyjádřit jako

$$\int \varphi_{\alpha j}(x) f(x, j) dx$$

a toto lze pomocí Bayesova vzorce upravit na

$$\int \phi_{\alpha j}(x) f(x, j) dx = \int \phi_{\alpha j}(x) f(j|x) f(x) dx \tag{2.13}$$

kde volbou $\phi_{\alpha j}(x)$ získáme různé míry nespolehlivosti. Aritmetický průměr je nestranným odhadem (2.13). Označme

$$S_{\alpha j} = \frac{1}{n} \sum_{i=1}^{n} \phi_{\alpha j}(x_i) \delta(j|x_i)$$

Dále platí

$$\int \phi_{\alpha j}(x) \hat{f}(x, j) dx = \int \phi_{\alpha j}(x) \hat{f}(j|x) f(x) dx = \frac{1}{n} \sum_{i=1}^{n} \phi_{\alpha j}(x_i) \hat{f}(j|x_i)$$

a toto označíme

$$\hat{S}_{\alpha j} = \frac{1}{n} \sum_{i=1}^{n} \phi_{\alpha j}(x_i) \hat{f}(j|x_i)$$

Míry v kategorii nespolehlivosti mají tvar

$$\begin{aligned} \text{TNespolehlivost} &= \sum_{j} \left(S_{\alpha j} - \hat{S}_{\alpha j} \right) \\ &= \frac{1}{n} \sum_{j=1}^{m} \sum_{i=1}^{n} \left(\phi_{\alpha j}(x_i) \left[\delta(j|x_i) - \hat{f}(j|x_i) \right] \right) \end{aligned} \tag{2.14}$$

Příkladem konkrétních voleb ϕ jsou

$$\begin{aligned} \phi_{1_{\alpha j}}(x_i) &= (1 - \hat{f}(j|x_i))^2 \\ \phi_{2_{\alpha j}}(x_i) &= -log(\hat{f}(j|x_i)) \end{aligned}$$

a míry nespolehlivosti pro $o_{1_{a_j}}$ a $o_{2_{a_j}}$

$$\text{Nespolehlivost}_1 = \frac{1}{n}\sum_{j=1}^{m}\sum_{i=1}^{n}\left((1-\hat{f}(j|x_i))^2\right)\left(\delta(j|x_i)-\hat{f}(j|x_i)\right) \quad (2.15)$$

$$\text{Nespolehlivost}_2 = -\frac{1}{n}\sum_{j=1}^{m}\sum_{i=1}^{n} log(\hat{f}(j|x_i)\left(\delta(j|x_i)-\hat{f}(j|x_i)\right) \quad (2.16)$$

Lze se setkat i s dalšími dvěmi kategoriemi - neoddělitelnosti a podobnosti, které však nehodnotí to, zda klasifikátor určí pro objekt x_i třídu, do které skutečně patří [10]. Míra neoddělitelnosti je založená na tom, jak špatně jsou oddělené $f(j|x)$ a podobnost na tom, jak špatně jsou oddělené $f(j|\hat{f}(j|x))$. Příkladem míry neoddělitelnosti je

$$\text{Neoddělitelnost} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(f(j|x_i)^2 - \frac{\sum_j f(j|x_i)}{m}\right)$$

a míra podobnosti

$$\text{Podobnost} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(f(j|\hat{f}(j|x_i))^2 - \frac{\sum_j f(j|\hat{f}(j|x_i))}{m}\right)$$

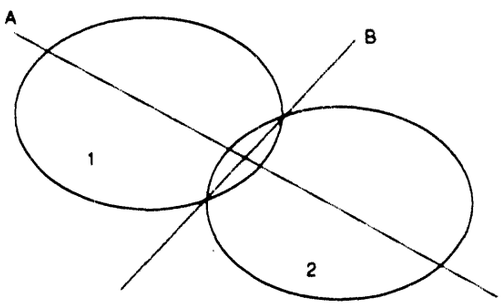
Jelikož

$$\frac{\sum_j f(j|x_i)}{m} = \frac{1}{m}$$

dostáváme se k tvaru

$$\begin{aligned}\text{Neoddělitelnost} &= -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(f(j|x_i)^2 - \frac{1}{m}\right)\\ &= \frac{1}{n}\sum_{i=1}^{n}\left(1-\sum_{j=1}^{m} f(j|x_i)^2\right)\\ &= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k\neq j} f(j|x_i)f(k|x_i)\end{aligned}$$

Neoddělitelnost nám tedy říká, jak podobné jsou skutečné pravděpodobnosti příslušností dat k třídám. Pokud pro každý objekt x_i jsou pravděpodobnosti blízké 0 kromě třídy, do které x_i patří, bude neoddělitelnost nízká.



Obrázek 2.1: Ukázka dvou tříd a dvou způsobů jak klasifikátor může oddělovat data

Podobnost popisuje, jak špatně klasifikátor odděluje třídu, do které x_i patří od ostatních. Pro názornost uvádíme obrázek 2.2.

Předpokládejme, že elipsy vyznačují obrys pro dvě třídy 1 a 2, které mají stejný počet prvků a $f(x|j)$ konstantní uvnitř každé elipsy a nula vně. Tato situace má nízkou hodnotu neoddělitelnosti. Budeme zkoumat klasifikátor, který data rozdělí podle přímky A, a předpokládejme, že všechny hodnoty pod i nad touto přímkou jsou konstantní a $\hat{f}(j|x)$ bude v obou oblastech zcela různá. Tento klasifikátor tedy od sebe výrazně oddělí data, avšak podobnost bude vysoká, protože skutečná pravděpodobnost příslušnosti k třídě 1 nebude příliš odlišná od skutečné pravděpodobnosti příslušnosti k třídě 2 v oblasti nad i pod přímkou. Klasifikátor, který data oddělí podle přímky B bude mít podobnost nižší. Toto nastane i pokud třídy 1 a 2 nebudou mít stejný počet prvků. V tom případě může být podobnost prvního klasifikátoru nižší než v minulém případě, avšak druhý klasifikátor bude mít podobnost ještě nižší.

2.3 Známé možnosti úprav současných měr

Míry mohou uvažovat zastoupení tříd v datech. Není obvykle těžké vygenerovat pravidla pro nejvíce zastoupenou třídu, avšak získat pravidla klasifikující do málo početné třídy může být problematické. Navíc, souborů dat, ve kterých se tato nerovnováha objevuje, je většina. Míry proto mohou zvýhodňovat pravidla, která se týkají méně zastoupené třídy, například vynásobením koeficientem.

$$\frac{n}{a+c}$$

Ze samotných dat však nelze říci, zda malá skupina dat tvoří pouze šum, nebo v sobě skrývá zásadní informace. Je tedy nutné interpretaci pravidel vzniklých z těchto dat důkladně rozvážit.

Míry mohou zohlednit i různou váhu atributů. Váha atributu, někdy nazývaná také cenou atributu, je reálné číslo. Cena celého pravidla pak může být součet cen atributů, které se v pravidle objevují. Může se stát, že pravidlo, které má menší cenu je použitelné, narozdíl od pravidla s vysokou cenou a vyšší mírou konzistence nebo přesnosti, které použitelné být nemusí. V případě, kdy zkoumáme a snažíme se určit diagnózu pacienta, se nelze opírat o pravidlo, které je sice velmi spolehlivé, ale odkazuje se na výsledek pitvy (tedy velmi cenný atribut). Je proto nutné použít pravidlo, které může být i méně spolehlivé, avšak je v danou chvíli použitelné. Míry však obvykle různé ceny atributů neuvažují.

2.4 ROC prostory

Pro porovnávání klasifikačních pravidel existuje metoda, jejíž hlavní výhodou je snadná a přehledná grafická reprezentace. ROC je zkratka slov Receiver Operating Characteristic ('charakteristika provozního přijímače') a její podstata byla převzata z teorie detekce signálu [7].

Opět se vychází ze čtyřpolní tabulky, objekty z jedné třídy jsou označeny jako 'pozitivní případy' a ostatní jako 'negativní' (tabulka 2.2).

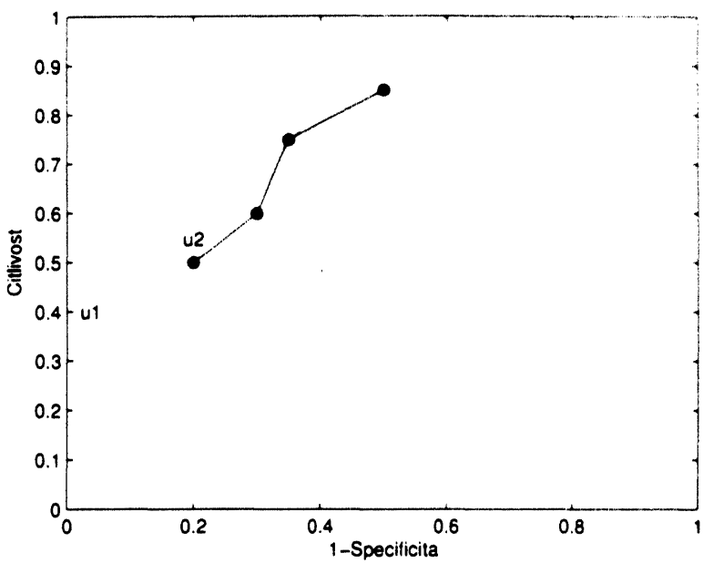
→	**Pozitivní případy**	**Negativní případy**
Predikce pozitivní	a	b
Predikce negativní	c	d

Tabulka 2.2: Čtyřpolní kontingenční tabulka pro ROC prostory

Pomocí této tabulky lze vyjádřit míru specificity

$$\text{Specificita} = \frac{d}{b+d} \tag{2.17}$$

a úplnosti (2.2), které se v tomto kontextu obvykle říká citlivost.



Obrázek 2.2: ROC křivka pro 4 různé soubory pravidel

Obor hodnot obou těchto měr je interval $\langle 0, 1 \rangle$, pro případ $a + c = 0$ se definuje hodnota míry citlivosti jako nula, obdobně i hodnota specificity v případě $b + d = 0$ se definuje jako nula.

Dále se zkonstruuje graf (obrázek 2.2), na jedné ose je hodnota citlivosti, na druhé hodnota specificity, a na něj se vynese bod podle hodnot měr pro daný zkoumaný soubor pravidel.

Množina bodů se stejnou hodnotou míry přesnosti (2.3) leží na jedné přímce. V případě, že pozitivních případů je stejně jako negativních, je tato přímka pouze posunutá osa souřadnic, jinak má jiný sklon. Takto lze porovnávat kvalitu klasifikačních pravidel klasifikujících do jedné zvolené třídy. Pro porovnání kvality pravidel klasifikujících do jiné třídy je třeba zkonstruovat novou tabulku a graf, neboť kvalita ostatních pravidel se v grafu neprojeví. Na obrázku 2.2 odpovídá úsečka u1 případu, kdy je počet pozitivních a negativních objektů stejný a prochází body s hodnotou přesnosti rovné hodnotě přesnosti souboru pravidel s nejvyšší hodnotou této míry. Pokud se v různém poměru uvažuje výsledek dvou vybraných souborů pravidel, lze simulovat soubory pravidel, které leží na spojnici těchto dvou bodů (například úsečka u2).

Takto lze srovnávat kvalitu souborů pravidel získaných z různých metod, avšak zajímavější je srovnávat, jak se mění kvalita souborů pravidel získaných při změně hodnoty právě jednoho parametru v nastavení jedné metody dobývání znalostí použité opakovaně na stejná data. Body, odpovídající

souborům pravidel při změně této hodnoty, lze spojit a tato křivka se nazývá ROC křivka. ROC prostorem budeme nazývat množinu $\{[x, y] : x, y \in \langle 0, 1 \rangle\}$, ve které se nachází všechny body ROC křivek.

Varianta ROC prostoru je PN prostor. Zde již nejsou hodnoty z prvního řádku čtyřpolní tabulky normovány do intervalu $\langle 0, 1 \rangle$, avšak na jednu osu se vynáší přímo a a na druhou b [9]. Přechod od PN prostoru k ROC prostoru lze provést tak, že se hodnoty opět normalizují.

2.5 Složitost

Z důvodů mnoha různých reprezentací znalostí samotná definice míry složitosti neexistuje. V případě souborů pravidel se za ni intuitivně považuje počet pravidel v souboru nebo počet podmínek v pravidlech. O složitosti hovoří i tzv. 'Occamova břitva'. Ze dvou souborů pravidel, které mají stejnou chybu na testovacích datech, je vhodné vybrat ten jednodušší, tedy menší, protože na uložení takového souboru je zapotřebí méně místa a pro uživatele je lépe zapamatovatelný [5].

K získání jednodušších souborů pravidel se obvykle používá strategie, při které se v metodě vytvoří větší seznam potenciálních pravidel a ten se dále upravuje a zjednodušuje - ubírá se počet pravidel či počet podmínek [5, 13]. Příkladem metody dobývání znalostí, ve které k tomuto dochází, je AQ21.

Kapitola 3

Navržené míry kvality

Výstupem různých metod jsou typicky různé soubory pravidel, kromě toho, i z jedné metody různým nastavením parametrů lze získat soubory pravidel zcela různé. Bylo by proto žádoucí umět tyto soubory porovnávat. V této kapitole předvedeme nový způsob na měření celých souborů pravidel, který zohledňuje specifické vlastnosti různých typů pravidel. Dále je provedeno rozšíření ROC prostorů na soubory pravidel jiných typů, než pro které byly původně konstruovány.

3.1 Rozšíření měr založených na kontingenční tabulce na soubory pravidel

Míry založené na čtyřpolní kontingenční tabulce slouží k měření jednotlivých pravidel. Cílem této sekce bude jejich rozšíření na celý soubor pravidel, s tím, že nejdůležitější pro nás bude rozšíření měr konzistence a přesnosti z důvodu, který bude zřejmý za chvíli.

3.1.1 Soubory klasifikačních pravidel

Začněme od případu booleovských klasifikačních pravidel. Pro tento případ již byly definovány míry v oddílu 2.2, avšak naším cílem bude ohodnotit celý soubor pravidel i pomocí měr založených na čtyřpolní tabulce (tabulka 1.1).

Rozšíření míry založené na čtyřpolní tabulce na celý soubor lze provést tak, že hodnota míry souboru je průměrná hodnota míry jednotlivých pravidel. Těmto mírám pro celý soubor pravidel budeme dávat přívlastek průměrná, například průměrná přesnost je průměrná přesnost pravidel v souboru.

Tento způsob zaručí, že míry zůstanou v jejich definovaném rozmezí - například průměrná konzistence nebo průměrná přesnost budou stále v intervalu $\langle 0, 1 \rangle$. Na druhou stranu, nevýhodou je typicky nízká průměrná podpora těchto pravidel.

Pokud by se uvažoval pouze součet hodnot míry jednotlivých pravidel, bude sice takto definovaná hodnota podpory pro celý soubor klasifikačních pravidel obvykle blízká jedné, avšak hodnoty některých měr (například konzistence nebo úplnosti) budou růst s počtem pravidel a bylo by i obtížné srovnávat různě velké (tj. s různým počtem pravidel) soubory pravidel.

3.1.2 Výhody DNF

V oddílu 1.3.1 jsme zmínili možnost převodu rozhodovacích pravidel do DNF. Nyní ukážeme, že reprezentace v podobě pravidel v DNF je vhodnější, než reprezentace jednotlivými pravidly.

Převedením pravidel na pravidla v DNF vzroste (přesněji neklesne) průměrná úplnost a podpora těchto pravidel. Uvažujme, že v souboru je k pravidel, která klasifikují do stejné třídy C, hodnoty a,b,c a d ze čtyřpolní tabulky pro i. pravidlo klasifikující do třídy C označíme a_i,b_i,c_i a d_i. Množina O_i obsahuje objekty pokryté i. pravidlem. Počet objektů pokrytých pravidlem v DNF po převedení těchto k pravidel je

$$|O_1 \cup O_2 \cup ... \cup O_k| \tag{3.1}$$

a hodnota a ze čtyřpolní tabulky pro pravidlo v DNF nebude menší než hodnoty a_i z čtyřpolních tabulek pro jednotlivá pravidla, protože

$$|O_1 \cup O_2 \cup ... \cup O_k| \geq max_i\,(a_i) \tag{3.2}$$

a jelikož $a_i + c_i$ je pro každé i konstantní, protože $a_i + c_i$ udává počet objektů v třídě C, při zvýšení hodnoty a musí dojít ke snížení hodnoty c. Hodnota $b_i + d_i$ je pro všechna i také konstantní, a tedy po převodu pravidel do DNF nemohlo dojít ke snížení ani podpory ani úplnosti.

Občas se požaduje, aby se pravidla s vysokou podporou měřila jinak, než pravidla s podporou nízkou [8]. Převedením pravidel do DNF se tento problém odstraní. Navíc již lze přímo srovnávat podporu pro různě velké soubory pravidel, získaných ze stejných dat.

3.1.3 Souvislost s mírami nespolehlivosti a nesprávnosti

Budeme se nyní snažit ukázat, že i když na první pohled nemají míry nespolehlivosti a nesprávnosti s průměrnými mírami nad čtyřpolními tabulkami nic společného, jejich sémantika je stejná a jejich výpočty se pro soubory klasifikačních pravidel v DNF nechají převést na výpočty měr definovaných na čtyřpolních tabulkách.

Definujme nejprve obecnou kontingenční tabulku velikosti m x m, označme ji A (tabulka 3.1). $A(i,j)$ udává počet objektů klasifikovaných do třídy i, které mají ve skutečnosti patřit do třídy j, tedy $\sum_{k:y_k=j} \hat{f}(i|x_k)$. Konstrukce této tabulky je možná díky jednoznačnosti. Pokud po převedení pravidel do DNF je jejich počet menší než m, dodáme do souboru pravidel i pravidla, která nejsou nikdy splněna a která mají v consequentu třídy, které se v původní množině consequentů nevyskytují. Pokud by se tato úprava neprovedla, nebylo by možné do měření zahrnout objekty, pro které by neexistovala pravidla s consequenty, které by obsahovaly třídy těchto objektů.

$\rightarrow$	C_1	C_2	...	C_m
P_1				
P_2				
...				
P_m				

Tabulka 3.1: Obecná kontingenční tabulka

Hodnota b_i pro čtyřpolní tabulku pravidla v DNF klasifikujícího do i. třídy je rovna počtu objektů, které nepatří do i. třídy, ale přitom dané pravidlo je do této třídy klasifikovalo, obdobně hodnota c_i je rovna počtu objektů, které patří do i. třídy, ale přitom byly zařazeny do jiné třídy. Hodnota a_i je rovna počtu správně klasifikovaných objektů pro třídu i. Tedy pro

čtyřpolní tabulku pro i. třídu platí

$$
\begin{aligned}
a_i &= A(i,i) \\
b_i &= \sum_{j:j\neq i} A(i,j) \\
c_i &= \sum_{j:j\neq i} A(j,i) \\
d_i &= \sum_{i=1}^{m}\sum_{j=1}^{m} A(i,j) - \left(\sum_{j:j\neq i} A(i,j) + \sum_{j:j\neq i} A(j,i) + A(i,i)\right)
\end{aligned}
$$

Toto dává návod na převedení tabulky A na m čtyřpolních tabulek, které budou odpovídat čtyřpolním tabulkám zkonstruované přímo pro jednotlivá pravidla v DNF. Také platí, že

$$\sum_i b_i = \sum_i \sum_{j:j\neq i} A(i,j) = \sum_i \sum_{j:j\neq i} A(j,i) = \sum_i c_i \tag{3.3}$$

Nyní ukážeme vztah mezi mírou nespolehlivosti a průměrnou podporou

pro soubor klasifikačních pravidel v DNF.

$$
\begin{aligned}
\text{Nespolehlivost}_1 &= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(1-\hat{f}(j|x_i)\right)^2\left(\delta(j|x_i)-\hat{f}(j|x_i)\right)\\
&= \frac{1}{n}\sum_{i=1}^{n}\sum_{j\in\{1,2,\ldots,m\}-\{y_i\}-\{\hat{y}_i\}}\left(1-\hat{f}(j|x_i)\right)^2\underbrace{\left(0-\hat{f}(j|x_i)\right)}_{=0}+\\
&\quad+\frac{1}{n}\sum_{i=1}^{n}\underbrace{\left(1-\hat{f}(\hat{y}_i|x_i)\right)^2}_{=0}\left(0-\hat{f}(\hat{y}_i|x_i)\right)+\\
&\quad+\frac{1}{n}\sum_{i=1}^{n}\left(1-\hat{f}(y_i|x_i)\right)^3\\
&= \frac{1}{n}\sum_{i=1}^{n}\left(1-\hat{f}(y_i|x_i)\right)^3\\
&= \frac{1}{n}\left(n-\sum_{k=1}^{m}\sum_{i:y_i=k}\hat{f}(y_i|x_i)\right)\\
&= 1-\frac{1}{n}\sum_{k=1}^{m}A(k,k)\\
&= 1-\frac{1}{n}\sum_{i=1}^{m}a_i\\
&= 1-\sum_{k=1}^{m}\frac{a_k}{a_k+b_k+c_k+d_k}\\
&= 1-\mathrm{PPo}*m \qquad (3.4)
\end{aligned}
$$

kde PPo je zkratka průměrné podpory

$$\mathrm{PPo}=\frac{\sum_k \frac{a_k}{a_k+b_k+c_k+d_k}}{m}$$

Dále ukážeme, že pro soubor klasifikačních pravidel v DNF souvisí průměrná přesnost s mírou nesprávnosti (2.9).

$$
\begin{aligned}
\text{Nesprávnost}_1 &= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\delta(j|x_i) - \hat{f}(j|x_i)\right)^2 = \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(\left(1-\hat{f}(y_i|x_i)\right)^2 + \sum_{j\in\{1,2,\ldots,m\}-\{y_i\}}\left(-\hat{f}(j|x_i)\right)^2\right) \\
&= \frac{1}{n}\left(n - \sum_{i=1}^{n}\left(\hat{f}(y_i|x_i)\right)^2 + \sum_{i=1}^{n}\sum_{j\in\{1,2,\ldots,m\}-\{y_i\}}\left(-\hat{f}(j|x_i)\right)^2\right) \\
&= \frac{1}{n}\left(n - \sum_{i=1}^{n}\left(\hat{f}(y_i|x_i)\right) + n - \sum_{i=1}^{n}\left(\hat{f}(y_i|x_i)\right)\right) \\
&= \frac{1}{n}\left(n - \sum_{j=1}^{m} A(j,j) + n - \sum_{j=1}^{m} A(j,j)\right) \\
&= \frac{1}{n}\sum_{k=1}^{m}\left(\sum_{j:j\neq k} A(j,k) + \sum_{j:j\neq k} A(j,k)\right) \\
&= \frac{1}{n}\left(\sum_{k}\sum_{j:j\neq k} A(j,k) + \sum_{j}\sum_{k:k\neq j} A(k,j)\right) \\
&= \frac{1}{n}\left(\sum_{k} b_k + \sum_{k} c_k\right) \\
&= \frac{1}{n}\sum_{k}(b_k + c_k) \\
&= \sum_{k}\frac{b_k + c_k}{a_k + b_k + c_k + d_k} \\
&= \left(1 - \frac{\sum_j \frac{a_k+d_k}{a_k+b_k+c_k+d_k)}}{m}\right) * m \\
&= (1 - \text{PPr}) * m
\end{aligned}
\tag{3.5}
$$

kde PPr je zkratka průměrné přesnosti

$$
\text{PPr} = \frac{\sum_k \frac{a_k+d_k}{a_k+b_k+c_k+d_k}}{m}
$$

Nyní použijeme předchozí odvození a ukážeme ještě vztah mezi nespoleh-

livostí a nesprávností.

$$\begin{aligned}
\text{Nespolehlivost}_1 &= \frac{1}{n}\left(n - \sum_{i=1}^{n} \hat{f}(y_i|x_i)\right) \\
&= \frac{1}{2n}\left(n - \sum_{i=1}^{n} \hat{f}(y_i|x_i) + n - \sum_{i=1}^{n} \hat{f}(y_i|x_i)\right) \\
&= \frac{1}{2}\,\text{Nesprávnost}_1
\end{aligned}$$

I přesto však budeme dále využívat obě dvě míry a budeme je rozšiřovat i na další typy pravidel, u kterých tato rovnost, která je zaručená jen pro klasifikační pravidla, nebude nutně platit.

3.1.4 Soubory rozhodovacích pravidel

V odstavci 3.1.2 jsme se zabývali pouze soubory klasifikačních pravidel. V případě souboru rozhodovacích pravidel nemusí být jednoznačnost klasifikace splněna. Příklad je uveden na obrázku 3.1. Předložíme-li tomuto souboru pravidel jeden objekt ke klasifikaci, čtyřpolní tabulky pro dvě třídy 'Pozitivní' a 'Negativní' budou obsahovat v jedné tabulce v poli a a ve druhé v poli b hodnotu 1, hodnota obou polí c bude rovna 0.

BMI(32-36) → Třída=Pozitivní
BMI(32-36) → Třída=Negativní

Obrázek 3.1: Soubor pravidel, který neklasifikuje jednoznačně

Navíc není ani nutné, aby soubor pravidel dokázal klasifikovat všechny objekty, příkladem může být opět obrázek 3.1. Pokud tomuto souboru předložím objekt, jehož BMI je 20, soubor nemůže určit třídu tohoto objektu.

Jednou z možností, jak přidělit nepokrytým objektům třídu, je přidělit ji náhodně s ohledem na rozložení tříd v datech.

Jelikož při použití měr nesprávnosti a nespolehlivosti se pravidlo nevyhodnocuje samostatně, ale uvažuje se celý soubor najednou, je možné nepokrytá data považovat za špatně zařazená. Další možností je, že se nebudou nepokrytá data pro vyhodnocování uvažovat.

Opět můžeme využít převodu do DNF. Předpokládejme nejprve, že soubor rozhodovacích pravidel klasifikuje jednoznačně. Z důvodu, že nemusí být

některé objekty pokryté, a že chceme, aby počet prvků v tabulce A byl roven n (celkovému počtu dat), rozšíříme tabulku A tak, že přidáme jeden řádek, označme ho '?', a zahrneme do jeho jednotlivých polí objekty, které se pro příslušné třídy nepodařilo klasifikovat. Pro tento typ pravidel rozšíříme míry nespolehlivosti a nesprávnosti tak, aby považovaly tato nepokrytá data za špatně zařazená. Pro nesprávnost bude místo (3.5) platit

$$\begin{aligned}
\text{Nesprávnost}_1 &= \frac{1}{n}\left(n - \sum_{i=1}^{n}\left(\hat{f}(y_i|x_i)\right)^2 + \sum_{i=1}^{n}\sum_{j\in\{1,2,\ldots,m\}-\{y_i\}}\left(-\hat{f}(j|x_i)\right)^2\right) \\
&= \frac{1}{n}\left(n - \sum_{k=1}^{m} A(j,j) + n - \sum_{k=1}^{m} A(j,j) - \sum_{j:j\leq m} A(m+1,j)\right) \\
&= \frac{1}{n}\sum_{k=1}^{m}\left(\sum_{j:j\neq k} A(j,k) + \sum_{j:j\leq m} A(m+1,j) + \sum_{j:j\neq k} A(j,k)\right. \\
&\quad \left. + \sum_{j:j\leq m} A(m+1,j) - \sum_{j:j\leq m} A(m+1,j)\right) \\
&= \frac{1}{n}\sum_{k=1}^{m}\left(\sum_{j:j\neq k} A(j,k) + \sum_{j:j\leq m} A(m+1,j) + \sum_{j:j\neq k} A(j,k)\right) \\
&= \frac{2}{n}\sum_{k=1}^{m}\sum_{j:j\neq k} A(k,j) + \frac{1}{n}\sum_{j:j\leq m} A(m+1,j) \qquad (3.6)
\end{aligned}$$

Hodnota míry nespolehlivosti zůstane stejná.

Pokud bychom tabulku A o řádek '?' nerozšířili, hodnota míry (3.5) bude hodnotou nesprávnosti pouze na pokrytých datech, a tedy dojde ke zmenšení hodnoty n. Jiný způsob, jak obejít penalizaci za nepokrytá data je nastavení implicitního určení třídy nepokrytých dat. Toto lze provést zvýšením hodnoty příslušné buňky tabulky A.

Nyní je třeba ještě vyřešit nejednoznačnost, která nám ztěžuje přiřazení objektu k jedné buňce tabulky A, po zkonstruování DNF pro rozhodovací pravidla. Dále budeme vždy uvažovat přidání řádku '?' pro nepokrytá data.

Definujme pro rozhodovací pravidla funkci

$$p(o_i) = \begin{cases} 1 & \text{pokud } (p.Antecedent \odot o_i \wedge p.Consequent \odot o_i) \\ 2 & \text{pokud } (p.Antecedent \otimes o_i) \\ 0 & \text{jinak} \end{cases} \qquad (3.7)$$

kde $\odot$, resp. $\otimes$, znamená, že daná část pravidla je pro objekt platná, resp. neplatná. Případy, ve kterých p nabývá hodnoty 1, resp. hodnoty 0, odpovídají tomu, že pravidlo platí, resp. neplatí, a hodnota 2 znamená, že pravidlo nedokáže rozhodnout. I když je pravdivostní hodnota výrazu pro jedno rozhodovací pravidlo ve výrokové logice vždy pouze 0 nebo 1, my zde využíváme i hodnoty 2, která souvisí se zavedením řádku '?'.

Pomocí funkce $p(o_i)$, je možné definovat platnost souboru P rozhodovacích pravidel p tak, že pro objekt o_i za předpokladu, že

$$\exists p \in P : p(o_i) = 1 \vee p(o_i) = 0 \tag{3.8}$$

soubor platí právě tehdy, když pro pravidla p ze souboru pravidel P a pro objekt o_i je splněno

$$|\{p \in P : p(o_i) = 1\}| \geq |\{p \in P : p(o_i) = 0\}| \tag{3.9}$$

Pokud nebude předpoklad (3.8) splněn, a tedy funkce $p(o_i)$ bude pro všechna pravidla ze souboru pravidel P pro vstupní objekt o_i nabývat hodnoty 2, zvýšíme v tabulce A o jedna hodnotu pole, které je určené řádkem '?' a sloupcem pro skutečnou třídu objektu o_i. Tato situace signalizuje, že pro soubor pravidel P nebude možné v případě objektu o_i získat hodnoty funkce $\hat{f}$.

V případě, že soubor pravidel platí, zvýšíme hodnotu příslušného pole na diagonále tabulky A o jedna. Případy, kdy soubor pravidel neplatí a nelze jednoznačně určit řádek, do kterého tuto informaci zaznamenat, vyřešíme později. Tento problém nastává, pokud existují dvě pravidla, která neplatí a přitom mají různý consequent.

3.1.5 Soubory pravidel dalších typů

I pro případ asociačních pravidel bychom chtěli zachovat využití tabulky A a mít možnost počítat obdobu měr (2.9) a (2.15). Pokud se podaří tuto tabulku definovat, bude možné přiřadit hodnoty i funkcím δ a $\hat{f}$.

Začněme od případu asociačních pravidel, které rozdělují hodnoty atributů na předem známé rozsahy. Pro tento případ je možné zkonstruovat tabulku, jejíž rozměry budou odpovídat počtu všech kombinací všech dílčích rozsahů všech atributů. Pro objekt se vyhledá sloupec, který bude spojen s takovou kombinací atributů, která tento objekt pokrývá. Pokud by objekt pokrývalo pouze jedno pravidlo, řádek bude určen rozsahem hodnot atributů,

které se nacházejí v consequentu a zbylé rozsahy atributu se určí tak, aby objekt byl pokryt.

Obdobně lze postupovat i v případě, kdy intervaly jednoho atributu mají v různých pravidlech po dvou společný průnik nebo jsou dopředu neznámé - stačí nelézt největší intervaly takové, které mají průnik s původními intervaly z pravidel buď nulový, nebo jsou celé jejich podmnožinami a jejich společné sjednocení odpovídá sjednocení původních intervalů. Jelikož je pravidel konečně, lze toto provést a tabulku A zkonstruovat.

Avšak i zde může nastat případ, kdy nepůjde určit řádek jednoznačně. Navíc od souboru rozhodovacích pravidel se soubor asociačních pravidel liší i tím, že mohou být splněna i dvě asociační pravidla, jejichž consequent je různý. To souvisí s tím, že původní pravidla nikdy nemůžou obsahovat v consequentu kombinaci všech atributů, neboť se tam nemůžou vyskytovat ty, které jsou již v antecedentu. Také celé consequenty některých pravidel mohou být součástí consequentů jiných pravidel a přitom ty více specifické (tj. s více atributy) nebudou platit a méně specifické (tj. s méně atributy) platit budou. Kromě toho, dramatické zvýšení velikosti tabulky A může vést k tomu, že některé míry (například průměrná přesnost nebo podpora) budou podávat neadekvátní informace. Průměrná podpora bude s rostoucí velikostí tabulky při stejném n klesat k nule a průměrná přesnost se bude blížit jedné. Také by mohla nastat situace, že by se po přidání jednoho pravidla výrazně změnil poměr platných a neplatných pravidel, protože by vznikly nové intervaly. Proto tento způsob považujeme za nevhodný.

3.2 Měření souboru založené na charakteristice souboru pravidel

V obou vzorcích (3.4) a (3.6) se rozlišují jednotlivé pole tabulky A pouze tím, zda se nacházejí na diagonále či nikoliv, resp. zda jsou navíc v řádku '?'. Celou tabulku A lze tedy zredukovat na vektor

$$V = \langle x, y, z \rangle$$

který budeme nazývat charakteristikou souboru pravidel. O charakteristice souboru pravidel budeme říkat, že popisuje chování souboru pravidel.

V případě jednoznačného souboru rozhodovacích pravidel bude hodnota x odpovídat počtu objektů, pro které soubor pravidel platí. Z důvodu jednoznačnosti bude toto rovno počtu o_i, pro které $\exists p : p(o_i) = 1$. Obdobně bude

hodnota y rovna počtu objektů, pro které $\exists p : p(o_i) = 0$ a hodnota z bude udávat počet objektů, pro které soubor pravidel nedokáže rozhodnout, tedy počet o_i, pro které $\forall p \in P : p(o_i) \neq 0 \wedge p(o_i) \neq 1$, neboli $n - x - y$.

Speciálně pro soubor klasifikačních pravidel budou platit následující rovnosti:

$$\begin{aligned} x &= \sum_{i=1}^{m} A(i,i) \\ y &= \sum_{i=1}^{m} \sum_{j:i \neq j} A(i,j) \\ z &= 0 \end{aligned} \tag{3.10}$$

a (3.4) a (3.5) budou odpovídat

$$\text{Nespolehlivost}_1 = 1 - \frac{1}{n} * x \tag{3.11}$$

resp.

$$\text{Nesprávnost}_1 = \frac{2}{n} * y \tag{3.12}$$

V tomto pohledu již není nutné určovat jeden konkrétní řádek, kam by se měl objekt v tabulce A vložit, a proto budeme vždy používat charakteristiku souboru pravidel V místo tabulky A. Ve všech předchozích situacích v sekcích 3.1.4 a 3.1.5, kde nebylo jasné, které pole se zvětší o jedna, použitím charakteristiky souboru pravidel V tento zásadní problém odpadne.

Definujme ještě funkci p, kterou využijeme při výpočtu charakteristiky souboru pravidel, pro asociační a GUHA pravidla. Pro asociační pravidlo a pravidlo s implikačním kvantifikátorem bude funkce p definována shodně jako v případě rozhodovacích pravidel, tedy

$$p(o_i) = \begin{cases} 1 & \text{pokud } p.Antecedent \odot o_i \wedge p.Consequent \odot o_i \\ 2 & \text{pokud } p.Antecedent \otimes o_i \\ 0 & \text{jinak} \end{cases} \tag{3.13}$$

a pro pravidlo s ekvivalenčním kvantifikátorem definujeme

$$p(o_i) = \begin{cases} 1 & \text{pokud } p.Antecedent \odot o_i \Leftrightarrow p.Consequent \odot o_i \\ 0 & \text{jinak} \end{cases} \tag{3.14}$$

kde $\odot$, resp. $\otimes$, opět znamená, že daná část pravidla je pro objekt platná, resp. neplatná. Případy, ve kterých p nabývá hodnoty 1, resp. hodnoty 0, odpovídají tomu, že pravidlo platí, resp. neplatí, a hodnota 2 znamená, že pravidlo nedokáže rozhodnout.

3.2.1 Většinový přístup

V tomto přístupu soubor asociačních, rozhodovacích a GUHA pravidel za podmínky (3.8) platí pro objekt o_i právě tehdy, když platí (3.9). Soubor fuzzy pravidel získaný z fuzzy-neuronové sítě bude platit právě tehdy, když

$$\sum_{j=1}^{m} \|p_j\|_{o_i}^{L} \geq \sum_{j=1}^{m} \|\neg p_j\|_{o_i}^{L} \tag{3.15}$$

Prvek charakteristiky souboru pravidel x, resp. y, bude odpovídat počtu objektů, pro které soubor pravidel platí, resp. neplatí, a hodnota z bude rovna $n - x - y$, speciálně pro případ souboru klasifikačních pravidel bude platit (3.10).

3.2.2 Konjunktivní přístup

Od souboru asociačních, rozhodovacích a GUHA pravidel je možné požadovat, aby v něm neexistovalo neplatné pravidlo, a tedy soubor pravidel pak za předpokladu (3.8) neplatí pro objekt o_i právě tehdy, když

$$\exists p \in P : p(o_i) = 0$$

Soubor fuzzy pravidel získaný z fuzzy-neuronové sítě bude platit právě tehdy, když

$$\forall j \; \|p_j\|_{o_i}^{L} \geq \frac{1}{2} \tag{3.16}$$

I zde bude prvek charakteristiky souboru pravidel x, resp. y, odpovídat počtu objektů, pro které soubor pravidel platí, resp. neplatí, a hodnota z bude rovna $n - x - y$. Navíc, x, y a z budou v případě souboru klasifikačních pravidel rovny (3.10) z důvodu dvou podmínek, které soubor klasifikačních pravidel musí splňovat.

3.2.3 Fuzzy přístup

Při výpočtu charakteristiky pravidel se nemusíme omezovat pouze na zjištění, zda soubor pravidel pro objekt platí nebo neplatí, tedy na hodnoty 0 a 1, ale můžeme v něm zahrnout informaci, že část souboru platí a část nikoliv.

Prvky charakteristiky souboru pravidel pro soubory rozhodovacích, asociačních a GUHA pravidel budou pro jeden objekt o_i odpovídat

$$
\begin{aligned}
x &= \begin{cases} \frac{|\{p:p(o_i)=1\}|}{|\{p:p(o_i)=0\}|+|\{p:p(o_i)=1\}|} & \text{pokud } |\{p : p(o_i) = 1\}| > 0 \\ 0 & \text{jinak} \end{cases} \\
y &= \begin{cases} \frac{|\{p:p(o_i)=0\}|}{|\{p:p(o_i)=0\}|+|\{p:p(o_i)=1\}|} & \text{pokud } |\{p : p(o_i) = 0\}| > 0 \\ 0 & \text{jinak} \end{cases} \\
z &= \begin{cases} 1 & \text{pokud } x + y = 0 \\ 0 & \text{jinak} \end{cases}
\end{aligned}
\tag{3.17}
$$

a pro soubor fuzzy pravidel získaný z fuzzy-neuronové sítě

$$
\begin{aligned}
x &= \frac{\sum_{j=1}^{m} \|p_j\|_{o_i}^{L}}{\sum_{j=1}^{m} \left(\|p_j\|_{o_i}^{L} + \|\neg p_j\|_{o_i}^{L} \right)} \\
y &= \frac{\sum_{j=1}^{m} \|\neg p_j\|_{o_i}^{L}}{\sum_{j=1}^{m} \left(\|p_j\|_{o_i}^{L} + \|\neg p_j\|_{o_i}^{L} \right)} \\
z &= 0
\end{aligned}
\tag{3.18}
$$

Charakteristika souboru pravidel pro celý datový soubor bude odpovídat součtu charakteristik pro každý objekt.

V případě souboru klasifikačních pravidel budou i při tomto přístupu x, y a z splňovat (3.10), jelikož jednoznačnost zaručí, že hodnoty x a y budou vždy rovny 0 nebo 1. Hodnota z bude rovna 0, což vyplývá z druhé podmínky pro klasifikační pravidla.

3.2.4 Míry založené na charakteristice souboru pravidel

Nyní, když máme k dispozici charakteristiku souboru pravidel, zkonstruujeme následující míry:

$$\text{Správnost} = \frac{x - y}{x + y + z} \tag{3.19}$$

$$\text{Podpora} = \frac{x}{x + y + z} \tag{3.20}$$

$$\text{Spolehlivost} = \begin{cases} \frac{x}{x+y} & \text{pokud } x + y > 0 \\ 0 & \text{jinak} \end{cases} \tag{3.21}$$

$$\text{Pokrytí} = \frac{x + y}{x + y + z} \tag{3.22}$$

(3.23)

Tyto míry lze využít na porovnání kvality různých souborů pravidel. Hodnoty měr spolehlivosti a podpory jsou pro soubory klasifikačních pravidel vždy stejné, neboť $z = 0$ a odpovídají hodnotě

$$\text{Spolehlivost} = \frac{x}{x + y} = \frac{x}{n} = 1 - \text{Nespolehlivost}_1 \tag{3.24}$$

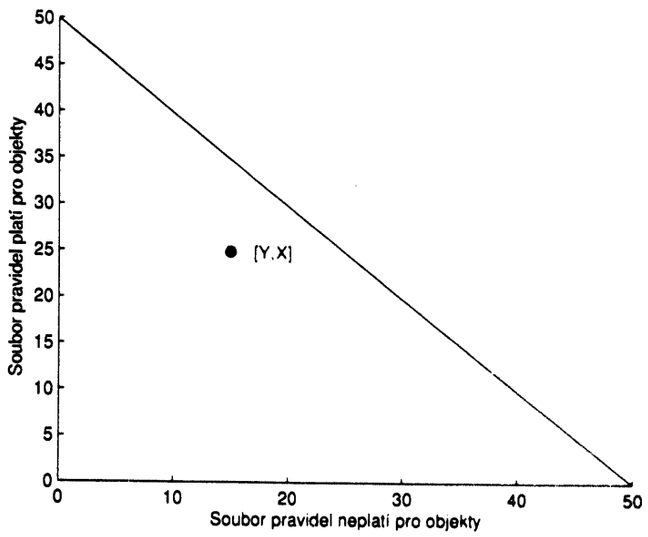
a správnost

$$\text{Správnost} = \frac{x - y}{x + y + z} = 1 - \frac{2y}{n} = 1 - \text{Nesprávnost}_1 \tag{3.25}$$

Míra pokrytí se od ostatních liší tím, že nemá svůj ekvivalent v mírách definovaných v sekci 2.2, jelikož pro soubor klasifikačních pravidel bude hodnota této míry vždy rovná 1. V případě souborů jiných typů pravidel, bude hodnota této míry udávat, o jaké části vstupních dat je možné rozhodnout, zda tyto soubory platí či nikoliv. Je zřejmé, že všechny míry rostou s hodnotou x a pokud jsou hodnoty y a z nulové, pak nabývají hodnoty 1. Nejmenší hodnota, kterou mohou míry kromě správnosti nabýt je 0, správnost může nabýt až -1.

3.3 Rozšíření ROC prostorů

Nyní, když už máme míry na porovnávání různých souborů různých typů pravidel, chtěli bychom mít obdobu ROC prostorů pro celé soubory pravidel. Samotné ROC i PN prostory nelze přímo použít, neboť ROC i PN prostory jsou použitelné pouze na soubory klasifikačních pravidel.



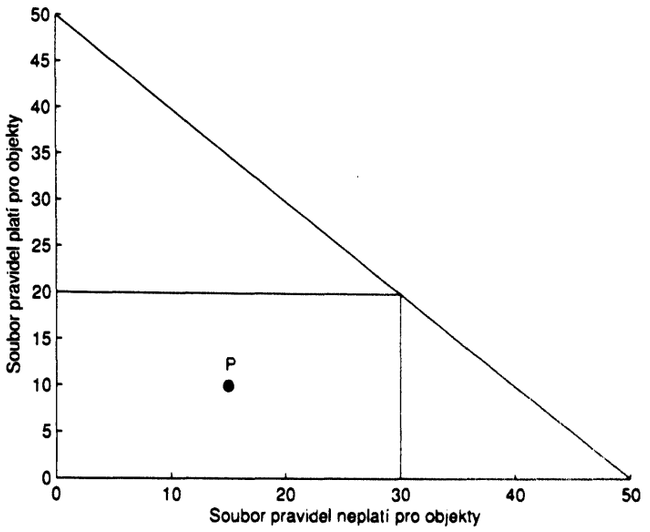
Obrázek 3.2: Ukázka charakteristiky souboru pravidel $\langle 25, 15, 10 \rangle$

3.3.1 Zavedení prostoru charakteristik souborů pravidel

Pro charakteristiku souboru pravidel $V = \langle x, y, z \rangle$ popisující chování souboru zkonstruujeme graf, na jehož vodorovnou osu budeme vynášet y a na druhou x (obrázek 3.2).

V těchto grafech budou všechny body popisující chování souboru ležet v prostoru levého dolního trojúhelníku, který je vymezen diagonálou u spojující body $[n, 0]$ a $[0, n]$. Součet hodnot y a x na diagonále je nejvyšší možný - tedy n. Tento prostor budeme nazývat prostorem charakteristik souborů pravidel.

Pokud si vezmeme, stejně jako v případě ROC nebo PN prostoru, ze souboru klasifikačních pravidel pouze ta pravidla klasifikující do zkoumané třídy, lze převést body odpovídající těmto souborům pravidel z prostoru charakteristik souborů pravidel do PN a tedy i ROC prostoru. Hodnoty x udávají ty případy, které jsou v tabulce 2.2 pro ROC prostory v poli a, neboť jsou to ty případy, které byly pro danou třídu zařazeny správně. Obdobně y odpovídá hodnotě b. Souřadnice bodů vynesených do PN prostoru budou odpovídat bodům v obdélníku o stejném rozměru jako PN prostor obsaženém přímo v prostoru charakteristik souborů pravidel, jejich souřadnice budou stejné (obrázek 3.3). Tedy i křivka v prostoru charakteristik souborů pravidel bude mít pro soubor pravidel klasifikujících do jedné třídy stejný tvar jako PN křivka. Levý dolní roh tohoto obdélníku bude mít souřadnici v počátku $[0, 0]$ a pravý horní roh se bude dotýkat diagonály u. Toto vyplývá z toho, že $x + y$



Obrázek 3.3: Ukázka prostoru charakteristik souboru pravidel s vyznačeným PN prostorem pro soubor pravidel klasifikujících do jedné třídy. Počet objektů ve vybrané třídě je 20, ve zbytku 30. Pro soubor pravidel P platí, že hodnota a z ROC tabulky je rovna 10 a hodnota b je rovna 15.

bude rovno n a z bude rovno 0.

3.3.2 Body se stejnými hodnotami měr

Znormujeme x,y a z tak, aby

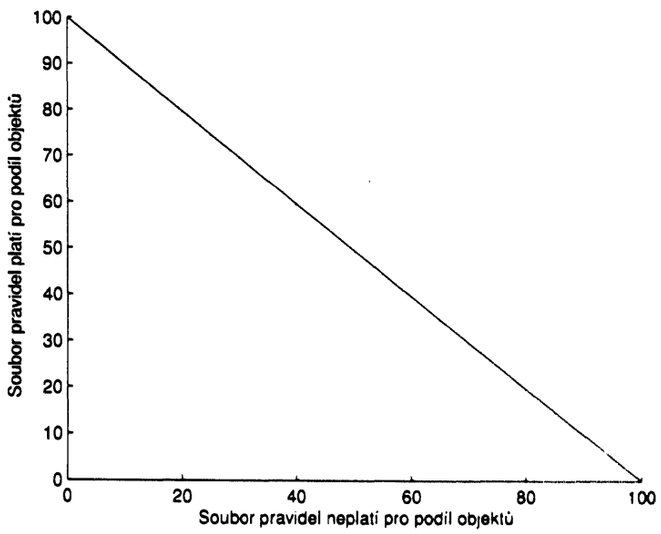
$$x + y + z = 100 \tag{3.26}$$

a nadále budeme uvažovat pouze normované charakteristiky. Toto provádíme proto, aby všechny grafy měly stejné měřítko a stejný rozsah os. Hodnoty všech měr definovaných v sekci 3.2.4 se po znormování x,y a z nezmění, jelikož koeficient, který bude čísla modifikovat, lze ze všech měr vytknout a zkrátit.

Body se stejnou správností mají tvar (pro $c \in \langle -1, 1 \rangle$)

$$\begin{aligned} c &= \frac{x-y}{x+y+z} = \frac{x-y}{100} \\ x &= f(y) = y + 100 * c \end{aligned} \tag{3.27}$$

tedy osa souřadnic posunutá o 100*c (obrázek 3.4).



Obrázek 3.4: Body se stejnými hodnotami správnosti

Jelikož oborem hodnot ostatních zavedených měr je interval $\langle 0, 1 \rangle$, bude dále platit, že $c \in \langle 0, 1 \rangle$. Body se stejnou podporou

$$\begin{aligned} c &= \frac{x}{100} \\ x &= f(y) = 100 * c \end{aligned} \tag{3.28}$$

tvoří úsečku rovnoběžnou s osou y (obrázek 3.5). Body se stejnou spolehlivostí mají tvar

$$\begin{aligned} c &= \frac{x}{x+y} \\ x &= f(y) = \frac{c}{1-c} * y \text{ pokud c} \in \langle 0, 1) \qquad (3.29) \\ y &= 0 \text{ pokud c} = 1 \qquad (3.30) \end{aligned}$$

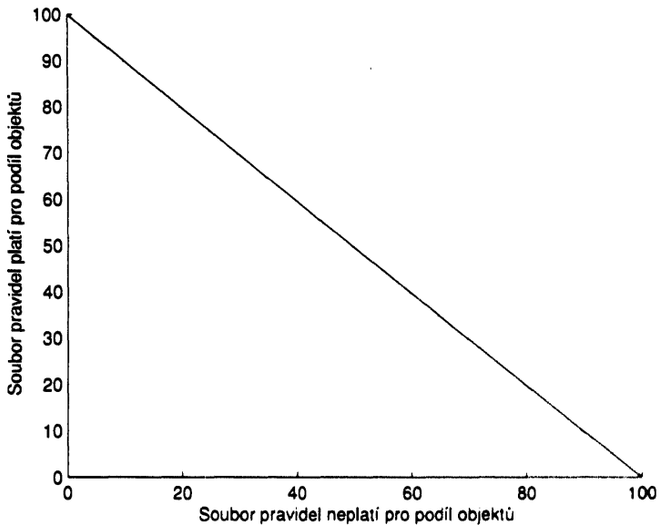
a jsou vyznačené na obrázku 3.6. Úsečky spojující body se stejnou hodnotou míry pokrytí

$$\begin{aligned} c &= \frac{x+y}{100} \\ x &= f(y) = c * 100 - y \end{aligned} \tag{3.31}$$

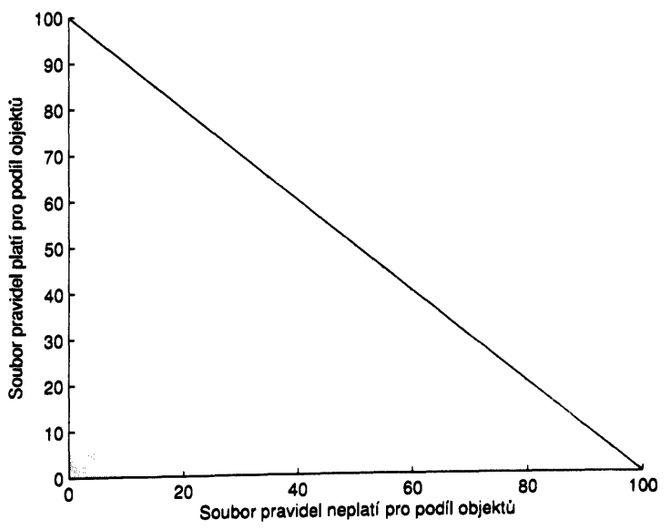
jsou rovnoběžné s diagonálou u a jsou zobrazené na obrázku 3.7.

3.3.3 Odečítání hodnot z grafu

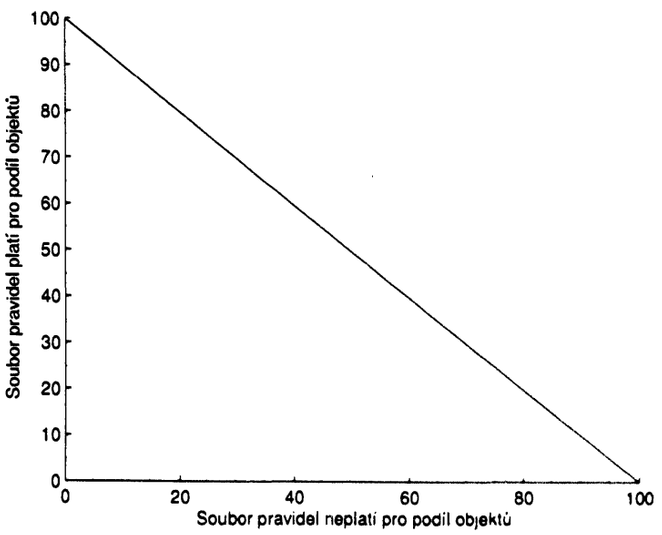
Přímo z grafu lze odečítat hodnoty správnosti, spolehlivosti, podpory i pokrytí. Hodnotu míry správnosti, podpory a pokrytí (v procentech) udává



Obrázek 3.5: Body se stejnými hodnotami podpory



Obrázek 3.6: Body se stejnými hodnotami spolehlivosti



Obrázek 3.7: Body se stejnými hodnotami pokrytí

druhá souřadnice bodu $[0, x']$ na svislé ose, ve kterém ji protne příslušná přímka spojující stejné hodnoty procházející bodem popisující chování souboru $[y, x]$, neboť pro všechny tři míry platí

$$\begin{aligned} c &= \frac{x'}{100} \\ x' &= c * 100 \end{aligned}$$

Hodnotu míry spolehlivosti, opět v procentech, udává druhá souřadnice bodu $[100 - x', x']$, který leží na průniku diagonály u a přímky spojující stejné hodnoty spolehlivosti procházející bodem popisující chování souboru $[y, x]$, neboť

$$\begin{aligned} c &= \frac{x'}{x' + 100 - x'} \\ x' &= c * 100 \end{aligned}$$

Kapitola 4

Ověření navrženého zobecnění

4.1 Použitá data

Vstupem pro metody na dobývání znalostí byly čtyři soubory dat: Iris plants database, BUPA liver disorders a Pima indians diabetes database získané ze stránek UCI Machine Learning Repository [22] a jedna data získaná z měření elektroencefalografu, která byla využita ke studiu mikrospánku [6].

Iris plants database tvoří celkem 150 záznamů, rozdělených ve stejném poměru do tří tříd, jedna třída určuje jeden druh kosatců: Setosa, Versicolour a Virginica. Každý záznam obsahuje šířku a výšku okvětních a kališních lístků.

BUPA liver disorders obsahuje dvě třídy, první obsahuje 145 záznamů o zdravých pacientech a druhá 200 záznamů o pacientech, kteří se léčí s chorobou jater. Každý záznam obsahuje množství vypitého alkoholu za den a průměrný obsah červených krvinek a jaterních enzymů v těle.

Pima indians diabetes database obsahuje 768 záznamů, z nichž 256 pacientů trpí cukrovkou. Atributy, které jsou k dispozici, tvoří BMI, věk, hladina cukru a inzulínu, tlak, tloušťka tukové řasy na tricepsu a koncentrace cukru v těle.

Data z elektroencefalografu tvoří 795 měření ze dvou elektrod které se skládají z 62 atributů a atributu třídy, který určuje stav, ve kterém se nacházela sledovaná osoba.

Každá datová sada byla rozdělena na 10 částí a byla použita křížová validace (viz sekce 1.5).

4.2 Postup ověřování

K získání pravidel z dat byly použity čtyři metody - 4ft-Miner, AQ 21, metoda založená na fuzzy-neuronových sítích a klasifikační stromy. Vyhodnocování kvality souborů pravidel probíhalo v prostředí Matlab, ve kterém jsme implementovali procedury určené k výpočtu hodnot měr a prostoru charakteristik souborů pravidel. Použité implementace rozhodovacích stromů a metody založené na fuzzy-neuronových sítích jsou také součástí prostředí Matlab. Pro metody 4ft-Miner a AQ 21, které nejsou součástí prostředí Matlab, jsme vytvořili procedury, které převádí soubory pravidel vygenerované pomocí těchto metod do prostředí Matlab. Zmíněné procedury a soubory pravidel jsou k dispozici na přiloženém médiu.

4.2.1 Nastavení pro metodu 4ft-Miner

Pro metodu 4ft-Miner byla vstupní spojitá data rozdělena ekvifrekvenčně do 4 až 8 tříd, podle velikosti vstupního datového souboru, s výjimkou případu, kdyby bylo v nějaké skupině příliš málo dat.

Zkoumány byly kvantifikátory fundované implikace, dolní kritické implikace, fundované ekvivalence a dále byl použit i Fisherův kvantifikátor. Jelikož byly výsledky pro tato pravidla srovnávány i s výsledky pro pravidla, která mají v consequentu vždy pouze atribut třídy, byly i pomocí 4ft-Mineru získávány převážně pravidla pouze s atributem třídy v consequentech. Pravidla s odlišnými consequenty byly získávány v případě pravidel s fundovanou ekvivalencí.

4.2.2 Nastavení pro metodu AQ21

V metodě AQ21 byly použity dvě modifikace hledání pravidel: PD a ATF, jelikož varianta TF není ovlivněna nastavením míry kvality popisu (viz oddíl 1.4.2). Pro obě modifikace byly zkoumány různé parametry ovlivňující tuto míru. První z nich byl Q, určující o kolik se může hodnota této míry pro pravidla v souboru pravidel odlišovat od pravidla s nejvyšší hodnotou této míry a druhý byl parametr váhy w.

4.2.3 Nastavení pro metodu založenou na fuzzy-neuronových sítích

Konfigurace této metody se provádí volbou logiky, počátečního přiblížení, funkce příslušnosti a počtu neuronů ve druhé skryté vrstvě, které se budou podílet na generování pravidel. Počet neuronů byl volen vždy v rozmezí jednoho až čtyř neuronů. Použité logiky byly Lukasiewiczova a produkt-Lukasiewiczova a tři funkce příslušnosti - sigmoida, Gaussova funkce a trojúhelník.

4.2.4 Nastavení pro klasifikační stromy

Při konstruování klasifikačních stromů byly použity Gini index a deviace, která vychází z entropie, jakožto indexy nečistoty. Vytvořené stromy pak byly prořezávány až do případu, kdy se celý strom skládal pouze z jednoho uzlu. Jelikož stromy konstruované z ne zcela stejných dat mohou vypadat zcela různě a obsahovat i různý počet uzlů, a tedy i pravidel, byl pro každou část křížové validace vybrán jeden soubor pravidel, který byl podle zvolené míry nejlepší. Výsledná hodnota měr odpovídala průměrné hodnotě měr těchto souborů pravidel.

Abychom získali i rámcový náhled na to, jak se liší chování pravidel získaných ze stromů o různé velikosti, zjišťovali jsme i hodnoty měr pro soubory pravidel získaných z rozhodovacích stromů, které vznikaly při postupném prořezávání původního stromu.

4.3 Výsledky

Nyní uvedeme prostory charakteristik souborů pravidel pro soubory pravidel získaných pomocí použitých metod a dat. Jejich cílem je i prozkoumat, jak se mění kvalita souborů pravidel při změně právě jednoho parametru. Tedy například, jak se chová fundovaná implikace při změně hodnoty p a stejné hodnotě Base.

4.3.1 Výsledky pro data o kosatcích

Pomocí metody 4ft-Miner jsme pro datovou sadu o kosatcích zkoumali chování souborů pravidel s fundovanou implikací a fundovanou ekvivalencí při změně hodnoty p i Base, dolní kritickou implikací při změně hodnoty p a α a

pravidla s Fisherovým kvantifikátorem při změně hodnoty α. Zkoumané hodnoty parametrů jsou vypsány v tabulce 4.1. Volené hodnoty parametru α pro Fisherův kvantifikátor i dolní kritickou implikaci byly 0.001, 0.005, 0.01, 0.05 a 0.1. Pro metodu 4ft-Miner uvádíme všechny tři přístupy k výpočtu charakteristiky souboru pravidel - většinový přístup, konjunktivní přístup a fuzzy přístup. Hodnoty měr se v jednotlivých přístupech odlišovaly, významné odchylky byly u míry spolehlivosti. Nejmenší spolehlivosti dosahovaly soubory při použití konjunktivního přístupu, rozdíl mezi spolehlivostí souborů při fuzzy přístupu a většinovém přístupu záležel na její hodnotě ve většinovém přístupu. Pokud byla tato hodnota velmi nízká, pak při fuzzy přístupu byla vyšší. I když tento jev nemusí nastat úplně pokaždé, na získaných souborech se projevil vždy. Patrný je například na obrázku 4.3.

GUHA kvantifikátor	Parametr	Od	Do	Krok
Fundovaná implikace	p	0.60	0.90	0.05
Fundovaná implikace	Base	0.03	0.21	0.02
Dolní kritická implikace	p	0.60	0.90	0.05
Dolní kritická implikace	α	0.001	0.1	-
Fundovaná ekvivalence	p	0.65	0.85	0.05
Fundovaná ekvivalence	Base	0.05	0.21	0.02
Fisherův kvantifikátor	α	0.001	0.1	-

Tabulka 4.1: Seznam zkoumaných GUHA kvantifikátorů a rozsahy parametrů pro metodu 4ft-Miner

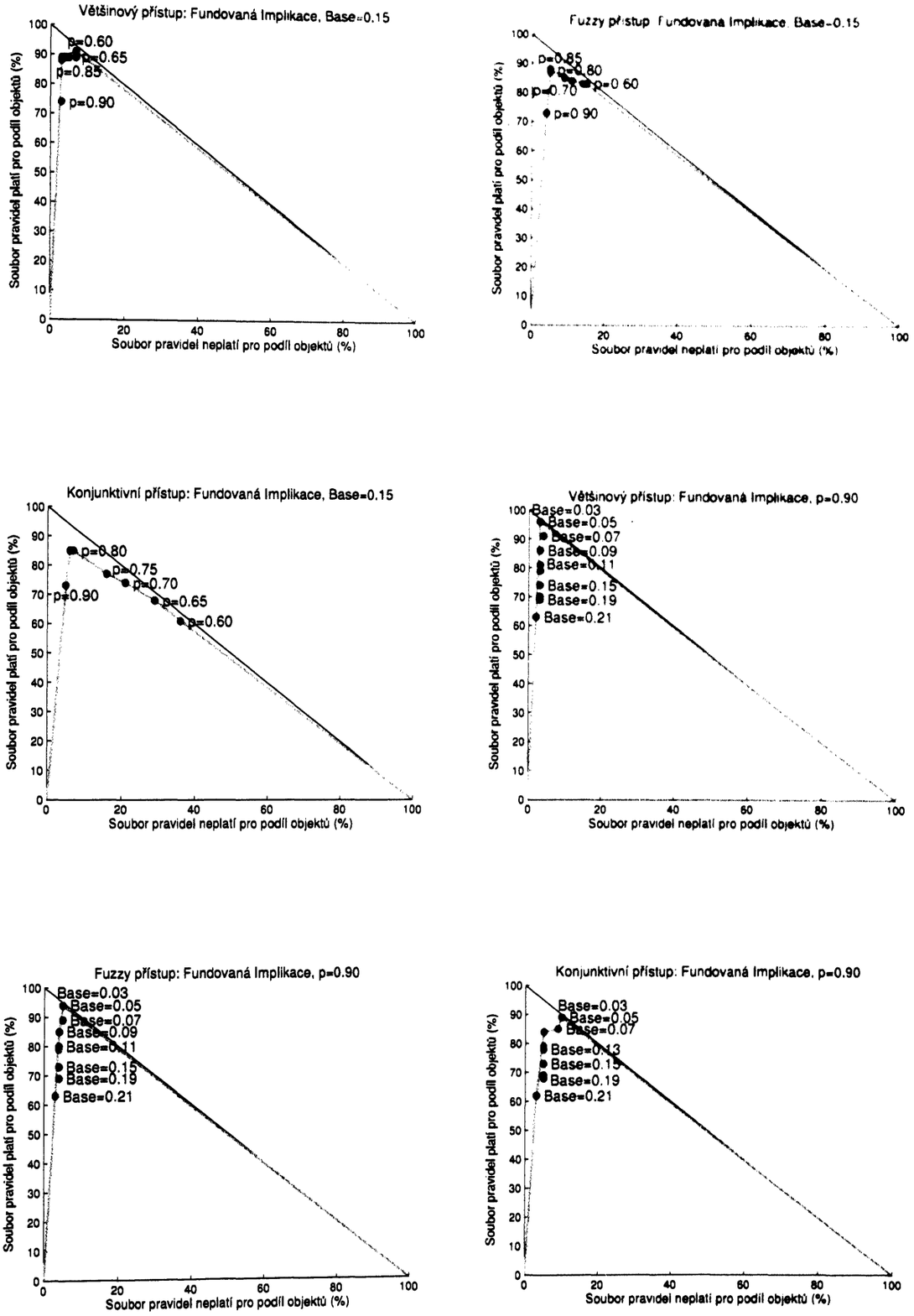
Pro metodu AQ 21 uvádíme změnu chování souboru pravidel jen při změně hodnoty parametru w v míře kvality popisu při použití modifikací ATF a PD a fuzzy přístupu, jelikož změna hodnoty parametru Q se na kvalitě pravidel projevovala pouze velmi nepatrně. Z obrázku 4.5 je patrné, že nejméně spolehlivé soubory pravidel byly ty, které byly získány při volbě hodnoty w blízké 0, kdy je kladen důraz na spolehlivost velmi malý nebo zcela žádný. Zkoumané hodnoty parametrů lze nalézt v tabulce 4.2.

Pro metodu založenou na fuzzy-neuronových sítích uvádíme vliv změny počtu neuronů ve druhé skryté vrstvě ve všech kombinacích obou logik a třech funkcí příslušnosti v případě fuzzy přístupu. Spolehlivost souborů pravidel získaných touto metodou byla různá. Při použití Gaussovy funkce a Lukasiewiczovy logiky byla velmi vysoká (obrázek 4.7), v případě produkt-Lukasiewiczovy logiky a sigmoidy (obrázek 4.6) nebo Gaussovy funkce byla spolehlivost poměrně nízká.

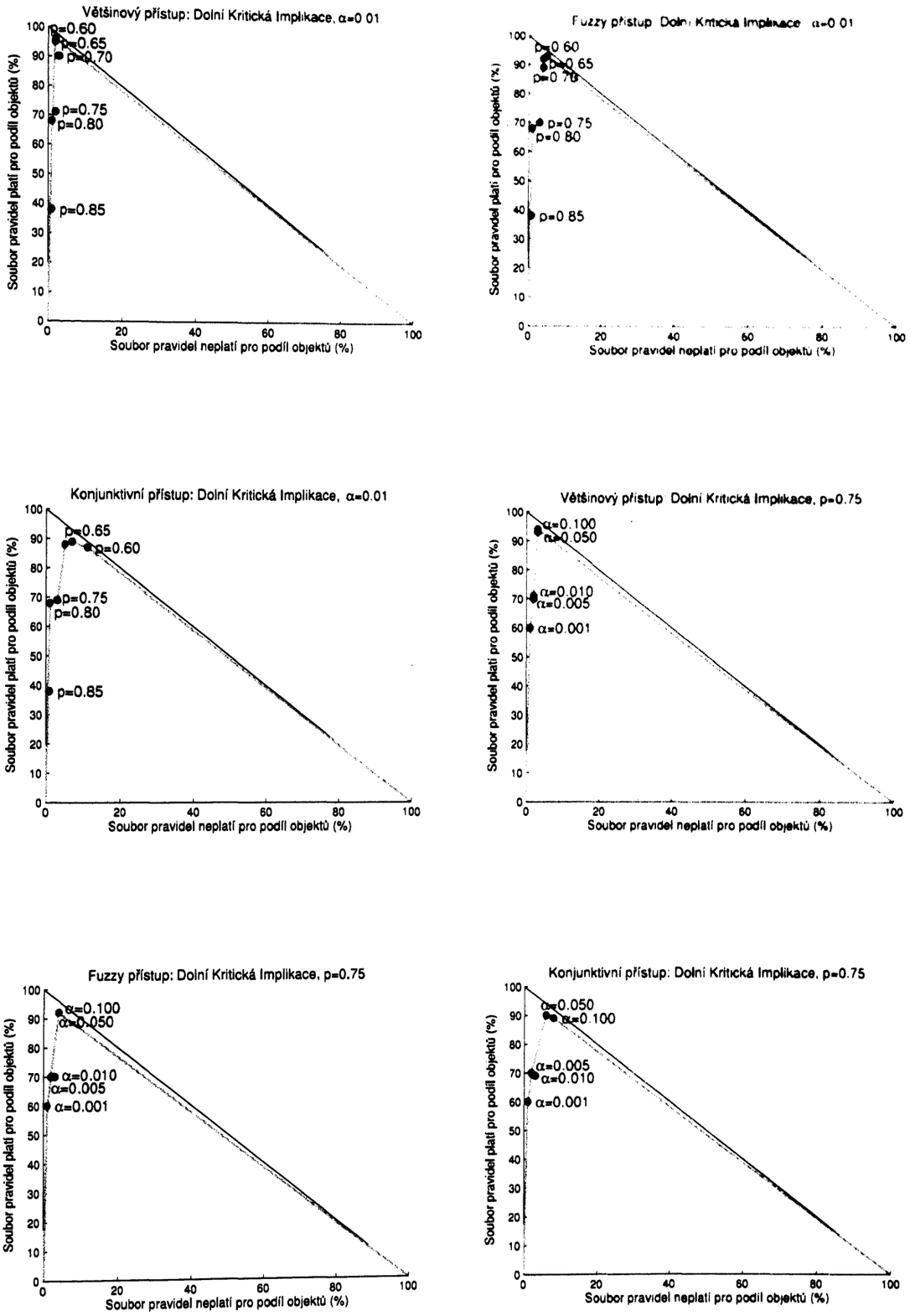
Modifikace	Parametr	Od	Do	Krok
ATF	w	0.0	1.0	0.25
PD	w	0.0	1.0	0.25
ATF	Q	0.4	0.8	0.1
PD	Q	0.4	0.8	0.1

Tabulka 4.2: Seznam zkoumaných modifikací a rozsah parametru w pro metodu AQ 21

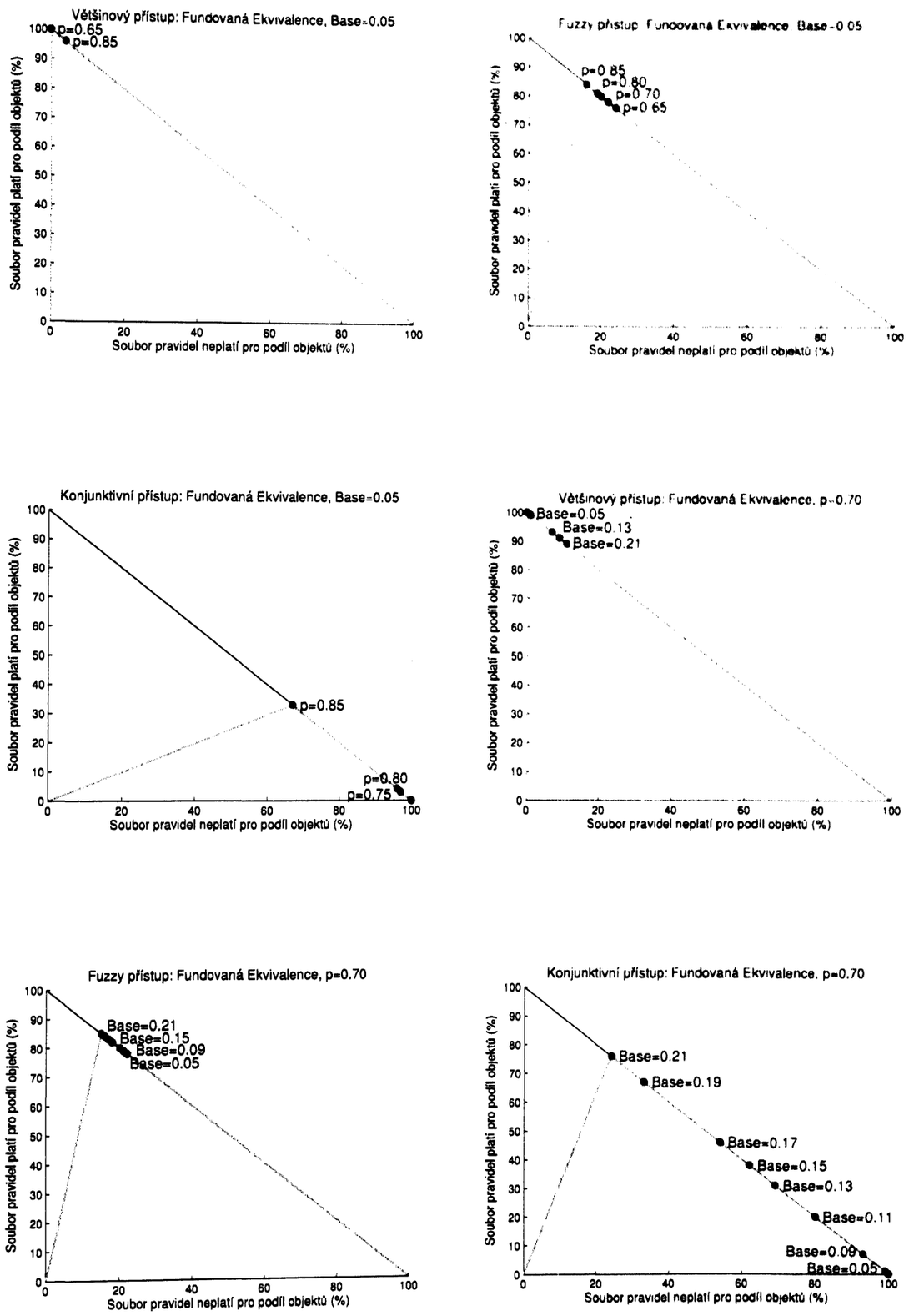
Z dat o kosatcích jsme v případe rozhodovacích stromů získali zcela shodné soubory pravidel při použití obou indexů nečistoty. V tabulce 4.3 uvádíme hodnoty měr správnosti, spolehlivosti, pokrytí, podpory a hodnoty ve sloupci označeném # udávají průměrný počet pravidel. Na obrázku 4.9 je uvedeno chování souborů pravidel, které vznikly ze stromů o různé velikosti.



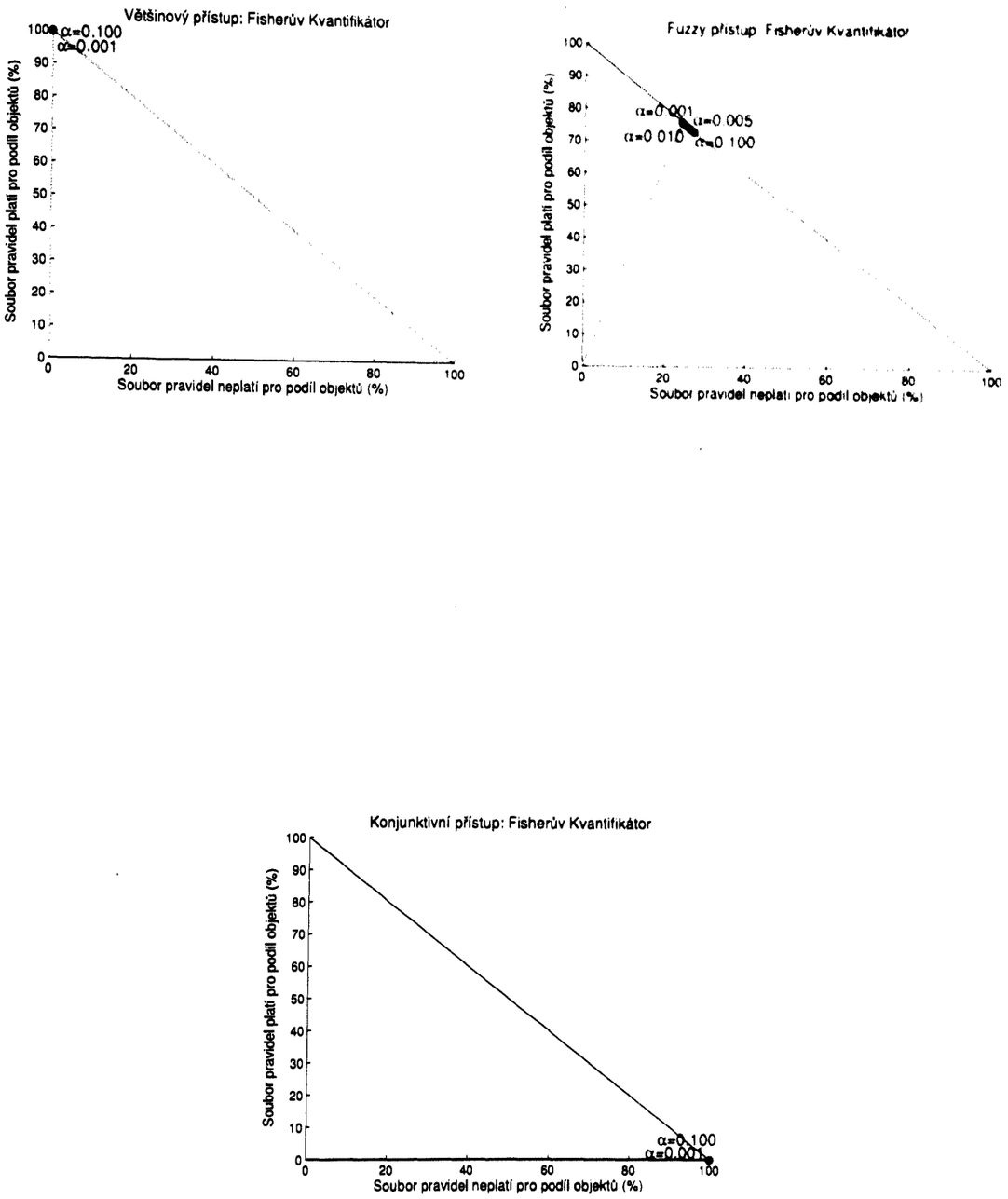
Obrázek 4.1: 4ft-Miner: Fundovaná implikace



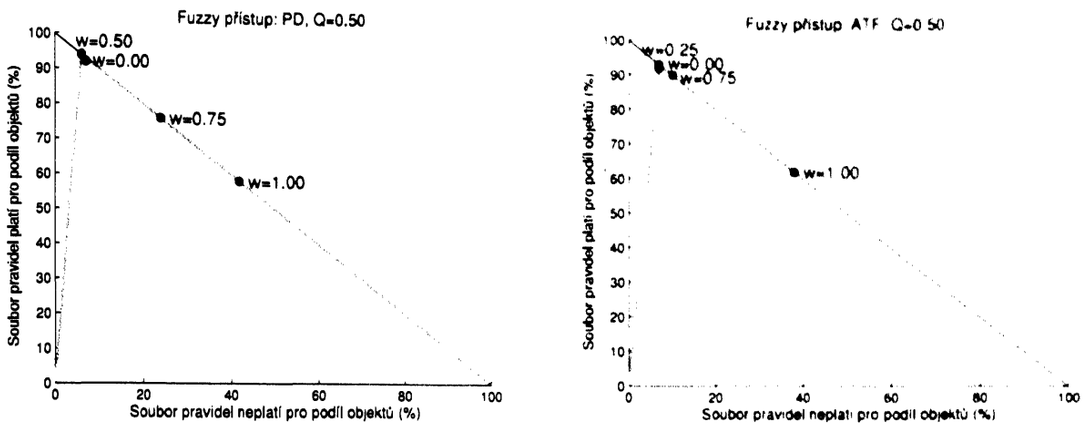
Obrázek 4.2: 4ft-Miner: Dolní kritická implikace



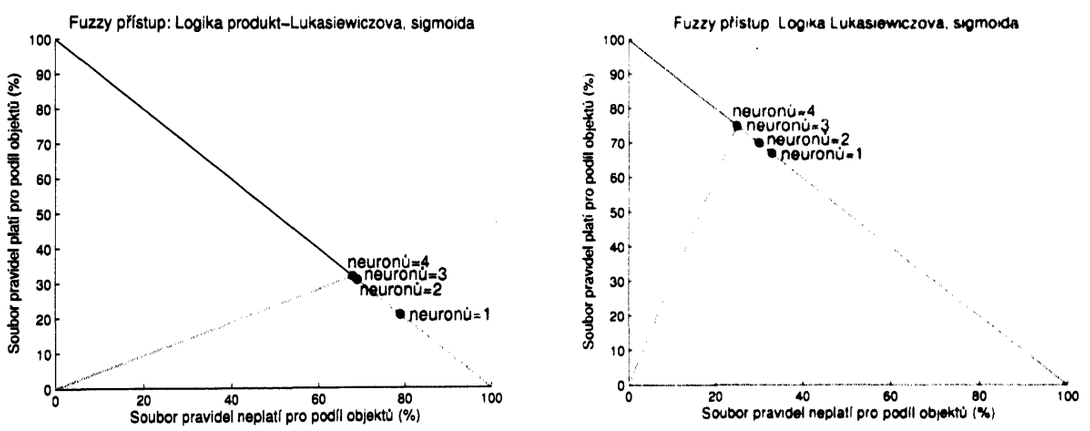
Obrázek 4.3: 4ft-Miner: Fundovaná ekvivalence



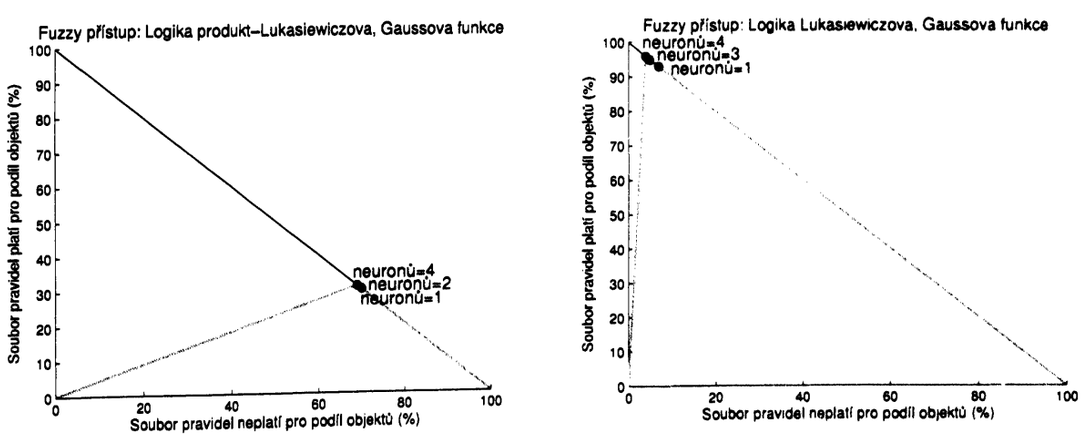
Obrázek 4.4: 4ff-Miner: Fisherův kvantifikátor



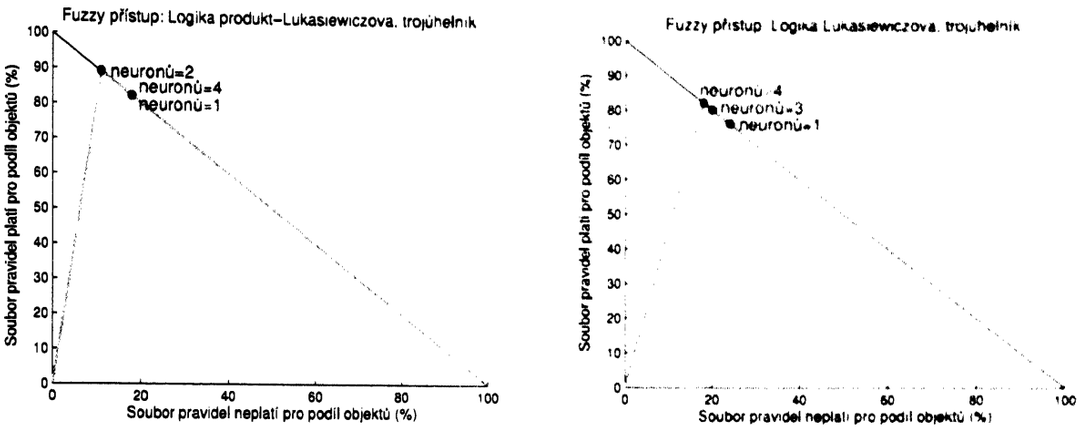
Obrázek 4.5: AQ21: Změna parametru w v míře kvality popisu



Obrázek 4.6: Fuzzy-neuronové sítě: sigmoida



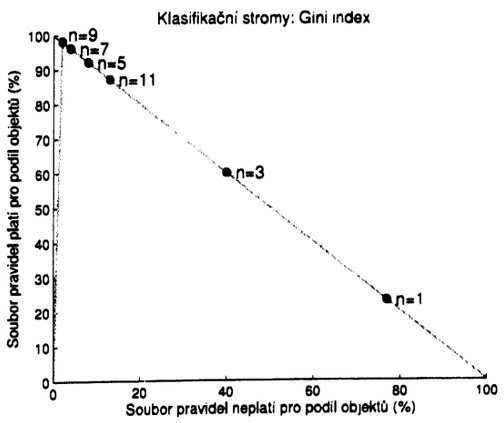
Obrázek 4.7: Fuzzy-neuronové sítě: Gaussova funkce



Obrázek 4.8: Fuzzy-neuronové sítě: trojúhelník

	Správnost	**Spolehlivost**	**Pokrytí**	**Podpora**	**#**
Gini index	0.86	0.93	1.0	0.93	5
Deviace	0.86	0.93	1.0	0.93	5

Tabulka 4.3: Výsledky pro soubory pravidel získaných z klasifikačních stromu



Obrázek 4.9: Klasifikační stromy s různou velikostí

4.3.2 Výsledky pro data o chorobách jater

Pomocí metody 4ft-Miner jsme na datech o chorobách jater zkoumali chování souborů pravidel s fundovanou implikací a fundovanou ekvivalencí při změně hodnoty p i Base, dolní kritickou implikací při změně hodnoty p a α a pravidla s Fisherovým kvantifikátorem při změně hodnoty α. Hodnoty těchto parametrů jsou zaznamenány v tabulce 4.4. Parametr α byl pro Fisherův kvantifikátor i dolní kritickou implikaci nastavován na hodnoty 0.001, 0.005, 0.01, 0.05 a 0.1. Výsledky uvádíme pro všechny tři přístupy k výpočtu charakteristiky souboru pravidel. Nejvíce se odlišuje konjunktivní přístup, při kterém jsou soubory pravidel výrazně méně spolehlivé, příkladem je průběh Fisherova kvantifikátoru na obrázku 4.13. Zajímavá je i změna kvality souborů pravidel s fundovanou implikací na obrázku 4.10, která spočívá ve zvýšení pokrytí, při snížení hodnoty parametru p nebo Base. Kromě konjunktivního přístupu nebylo toto zvýšení doprovázeno výrazným snížením spolehlivosti.

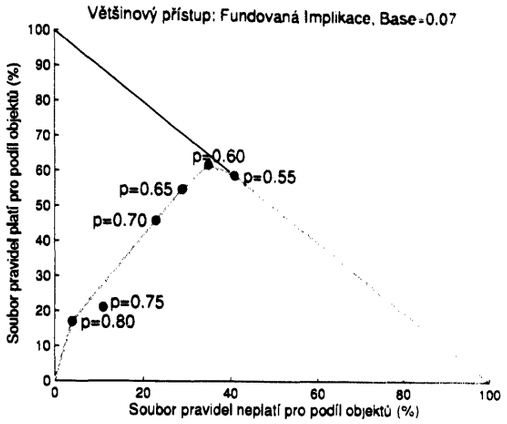
GUHA kvantifikátor	Parametr	Od	Do	Krok
Fundovaná implikace	p	0.55	0.80	0.05
Fundovaná implikace	Base	0.03	0.19	0.02
Dolní kritická implikace	p	0.55	0.80	0.05
Dolní kritická implikace	α	0.001	0.1	-
Fundovaná ekvivalence	p	0.70	0.85	0.05
Fundovaná ekvivalence	Base	0.05	0.19	0.02
Fisherův kvantifikátor	α	0.001	0.1	-

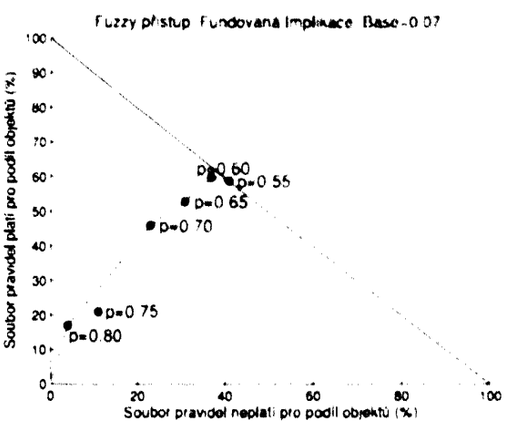
Tabulka 4.4: Seznam zkoumaných GUHA kvantifikátorů a rozsahy parametrů pro metodu 4ft-Miner

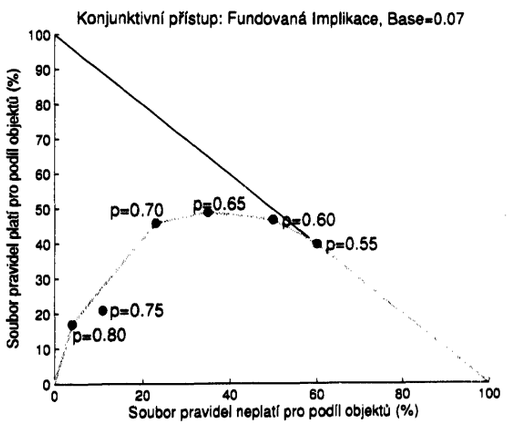
Rozdíly mezi modifikacemi ATF a PD a vliv změny hodnoty parametru w v případě fuzzy přístupu jsou zachyceny na obrázku 4.14 a zkoumané hodnoty jsou uvedeny v tabulce 4.2.

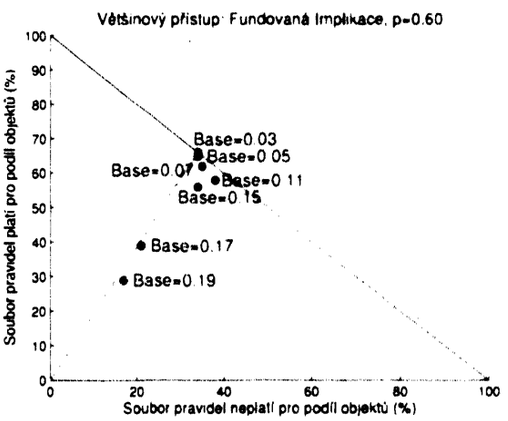
Pro metodu založenou na fuzzy-neuronových sítích opět uvádíme vliv počtu neuronů ve druhé skryté vrstvě na kvalitu souborů pravidel ve fuzzy přístupu. Tento vliv však nebyl příliš velký, neboť ve všech případech měly soubory velmi podobnou spolehlivost, což demonstruje obrázek 4.17, pouze v případě Lukasiewiczovy logiky a Gaussovy funkce na obrázku 4.16 spolehlivost relativně výrazněji rostla s počtem neuronů.

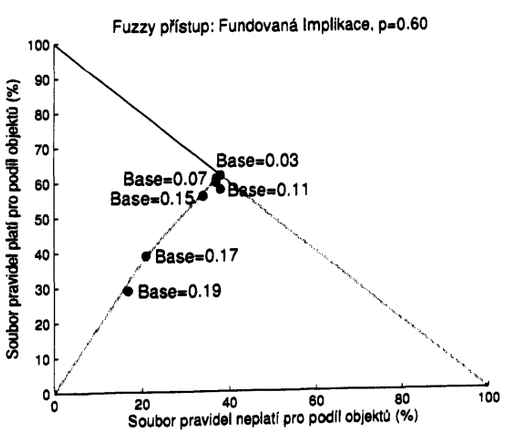
Rozdíly v hodnotách měr souborů pravidel získaných z klasifikačních stromů s použitím indexu Gini a deviace nebyly příliš velké, avšak soubory pravidel byly menší při použití Gini indexu. Toto demonstruje tabulka 4.5.

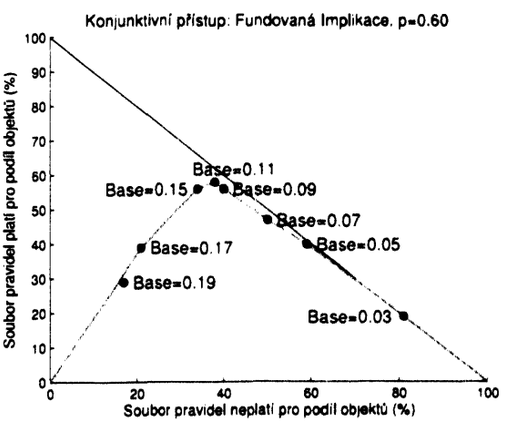




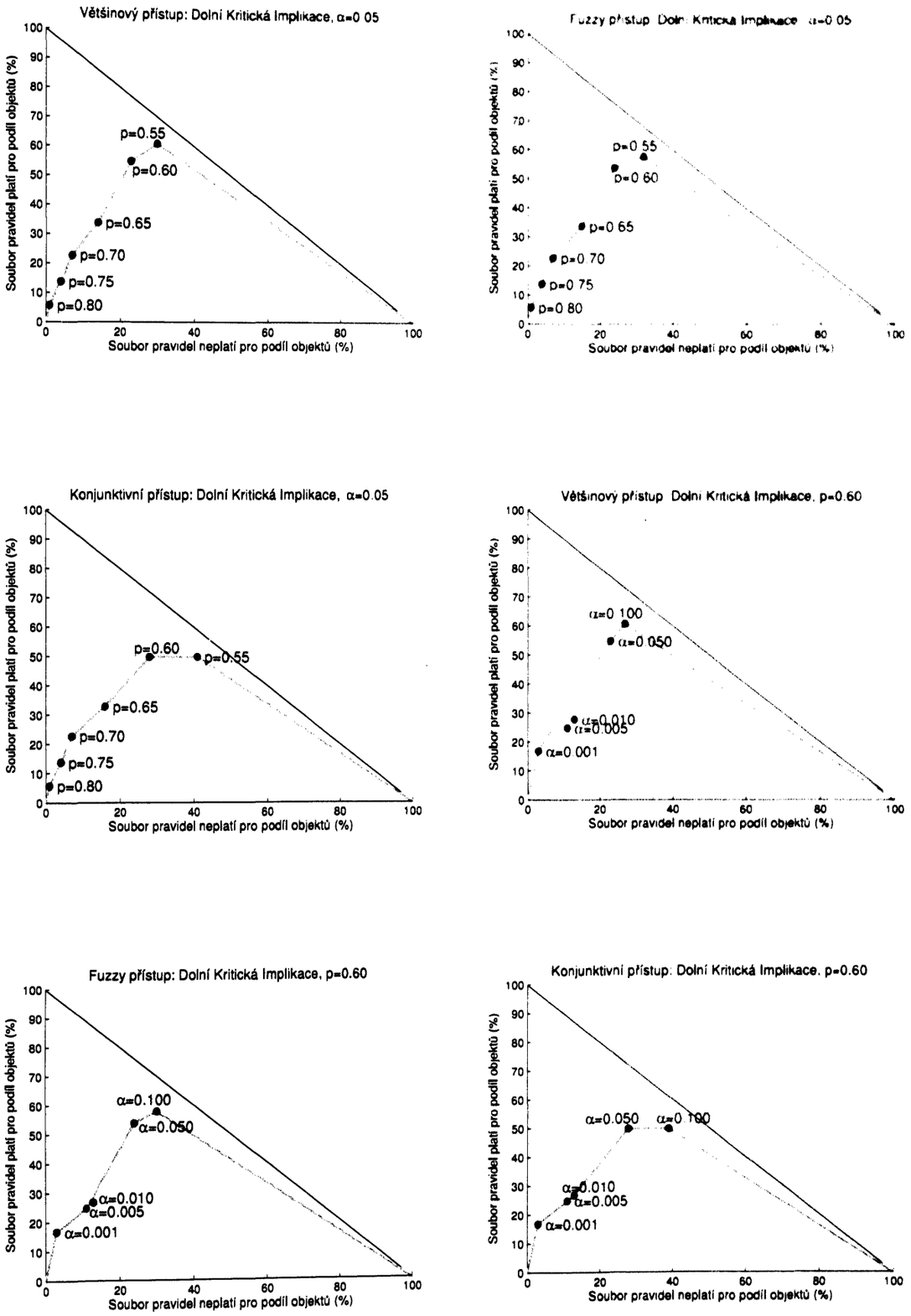




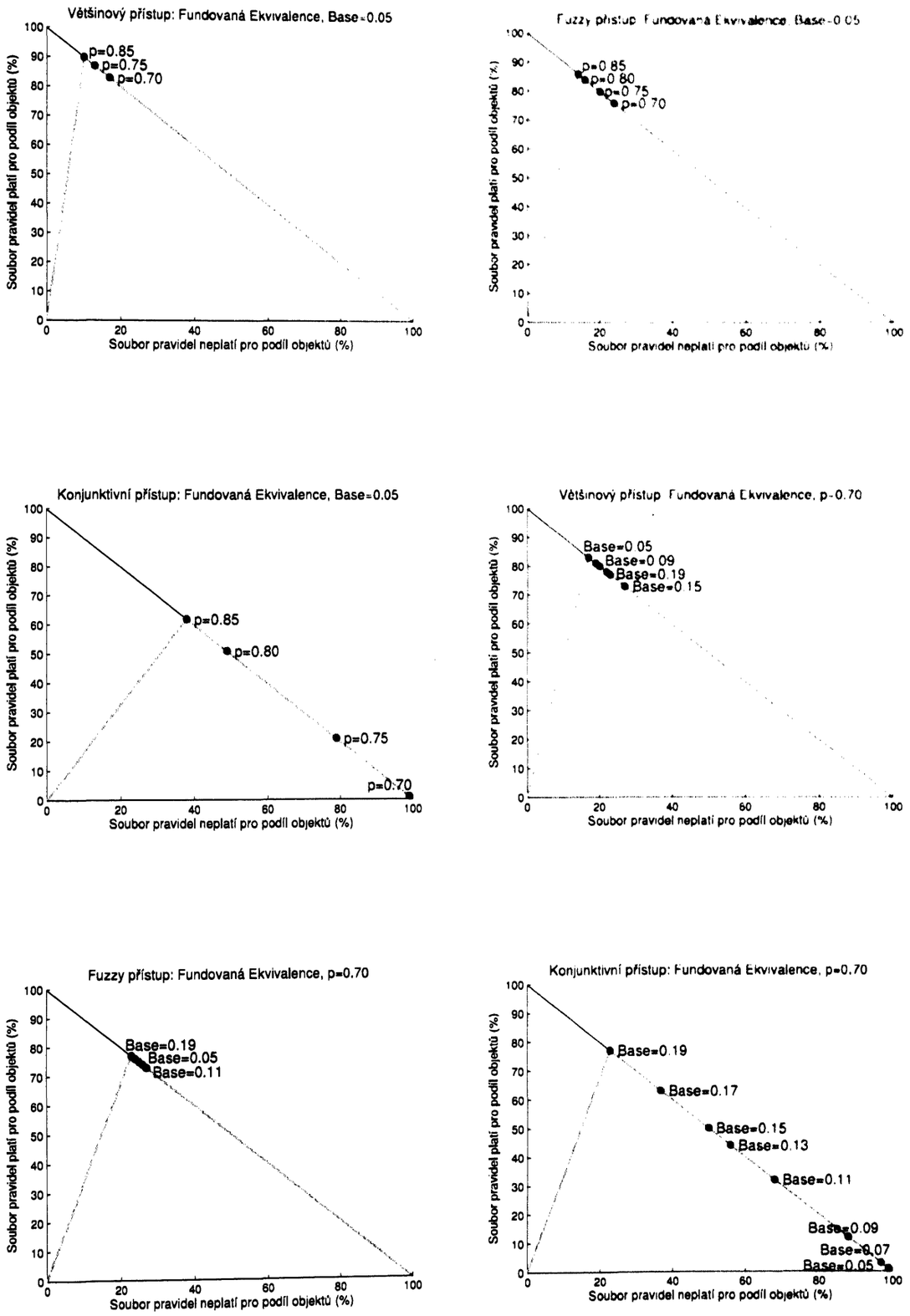




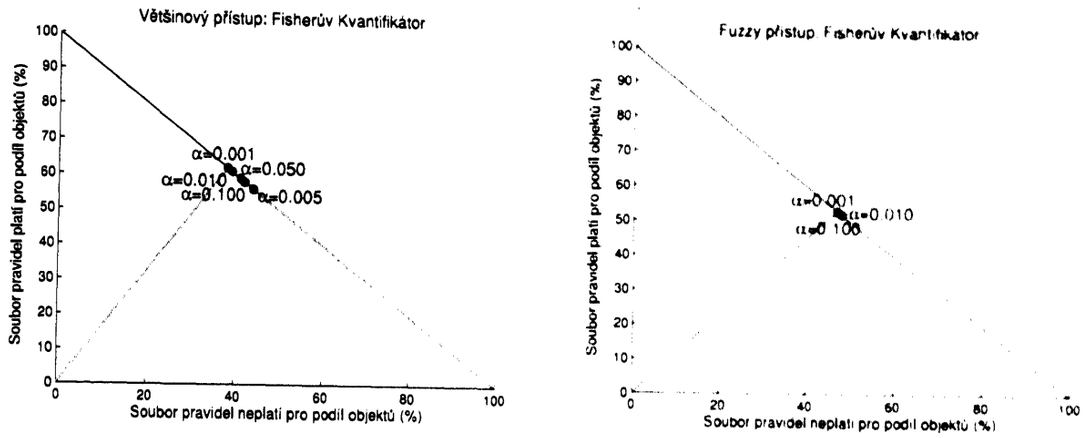
Obrázek 4.10: 4ft-Miner: Fundovaná implikace

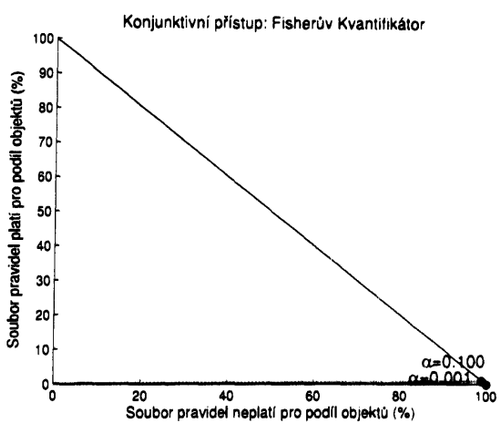


Obrázek 4.11: 4ft-Miner: Dolní kritická implikace

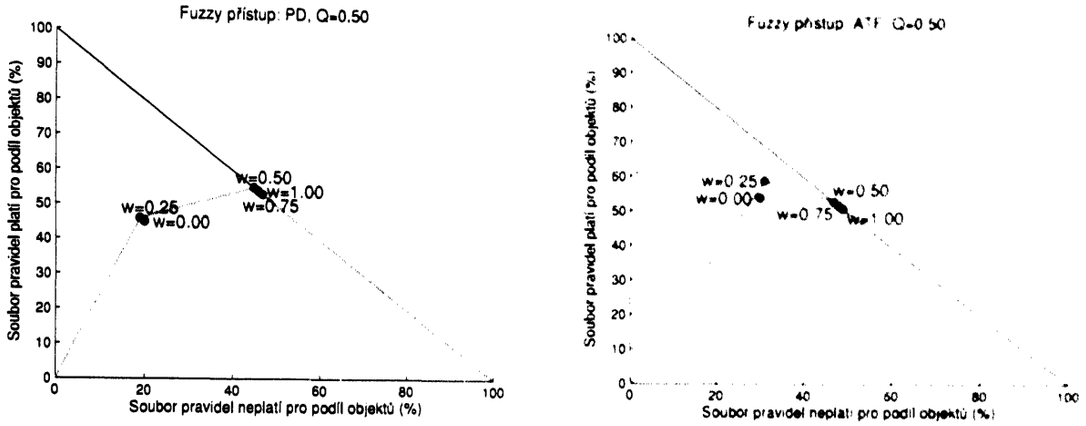


Obrázek 4.12: 4ft-Miner: Fundovaná ekvivalence

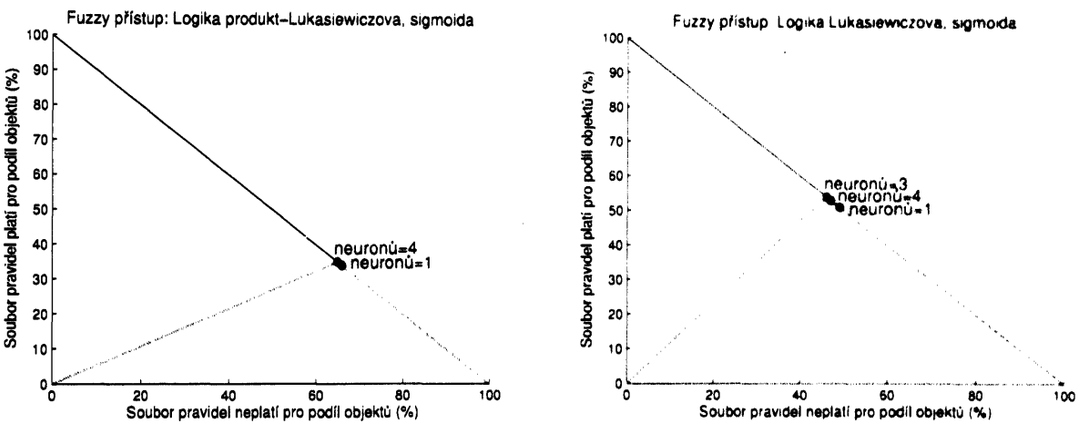




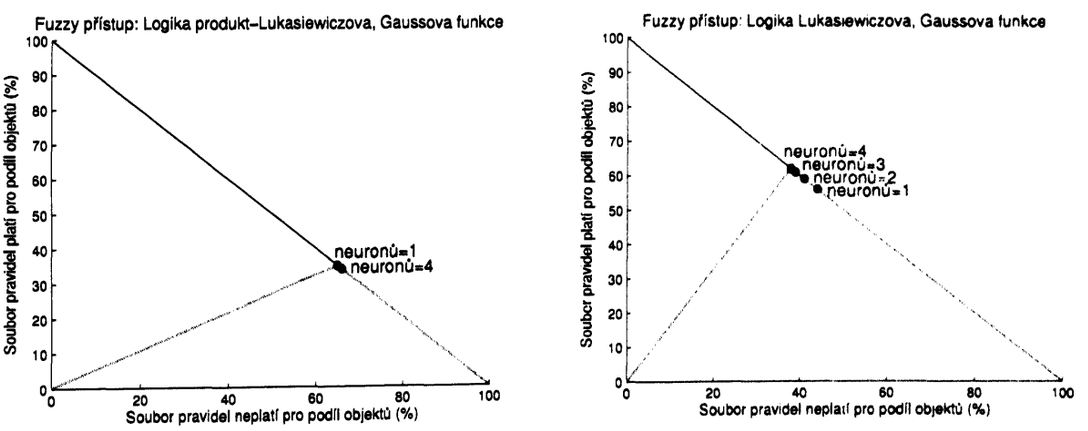
Obrázek 4.13: 4ft-Miner: Fisherův kvantifikátor



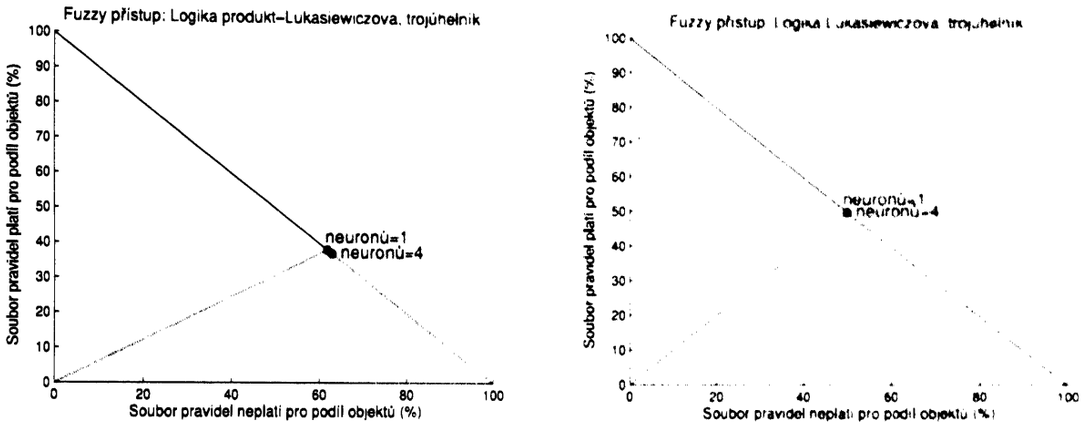
Obrázek 4.14: AQ21: Změna parametru w v míře kvality popisu



Obrázek 4.15: Fuzzy-neuronové sítě: sigmoida



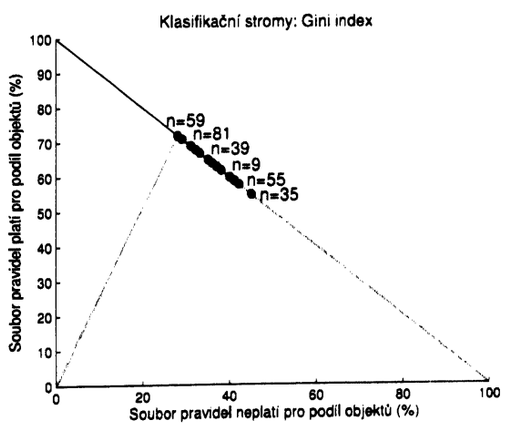
Obrázek 4.16: Fuzzy-neuronové sítě: Gaussova funkce



Obrázek 4.17: Fuzzy-neuronové sítě: trojúhelník

	Správnost	**Spolehlivost**	**Pokrytí**	**Podpora**	**#**
Gini index	0.44	0.72	1.00	0.72	11
Deviace	0.42	0.71	1.00	0.71	18

Tabulka 4.5: Výsledky pro soubory pravidel získaných z klasifikačních stromu pro data o chorobách jater



Obrázek 4.18: Klasifikační stromy s různou velikostí

4.3.3 Výsledky pro data o cukrovce

Pro data o cukrovce uvádíme prostory charakteristik souboru pravidel s fundovanou implikací a fundovanou ekvivalencí při změně hodnoty p i Base a dolní kritickou implikací při změně hodnoty p a α. Rozsahy těchto parametru jsou uvedeny v tabulce 4.6 a parametr α pro dolní kritickou implikaci a Fisherův kvantifikátor nabýval hodnot 0.001, 0.005, 0.01, 0.05 a 0.1. U souboru pravidel s fundovanou implikací (obrázek 4.19) opět docházelo při snížení hodnoty p nebo Base k nárůstu pokrytí, spolehlivost zůstávala ve fuzzy a většinovém přístupu přibližně stejná. Nejmenší rozdíly mezi jednotlivými přístupy nastaly u souborů s dolní kritickou implikací (obrázek 4.20), na kterých je patrná obdobná změna pokrytí při snížení hodnot parametru. Naopak největší rozdíly byly u souborů s fundovanou ekvivalencí (obrázek 4.21).

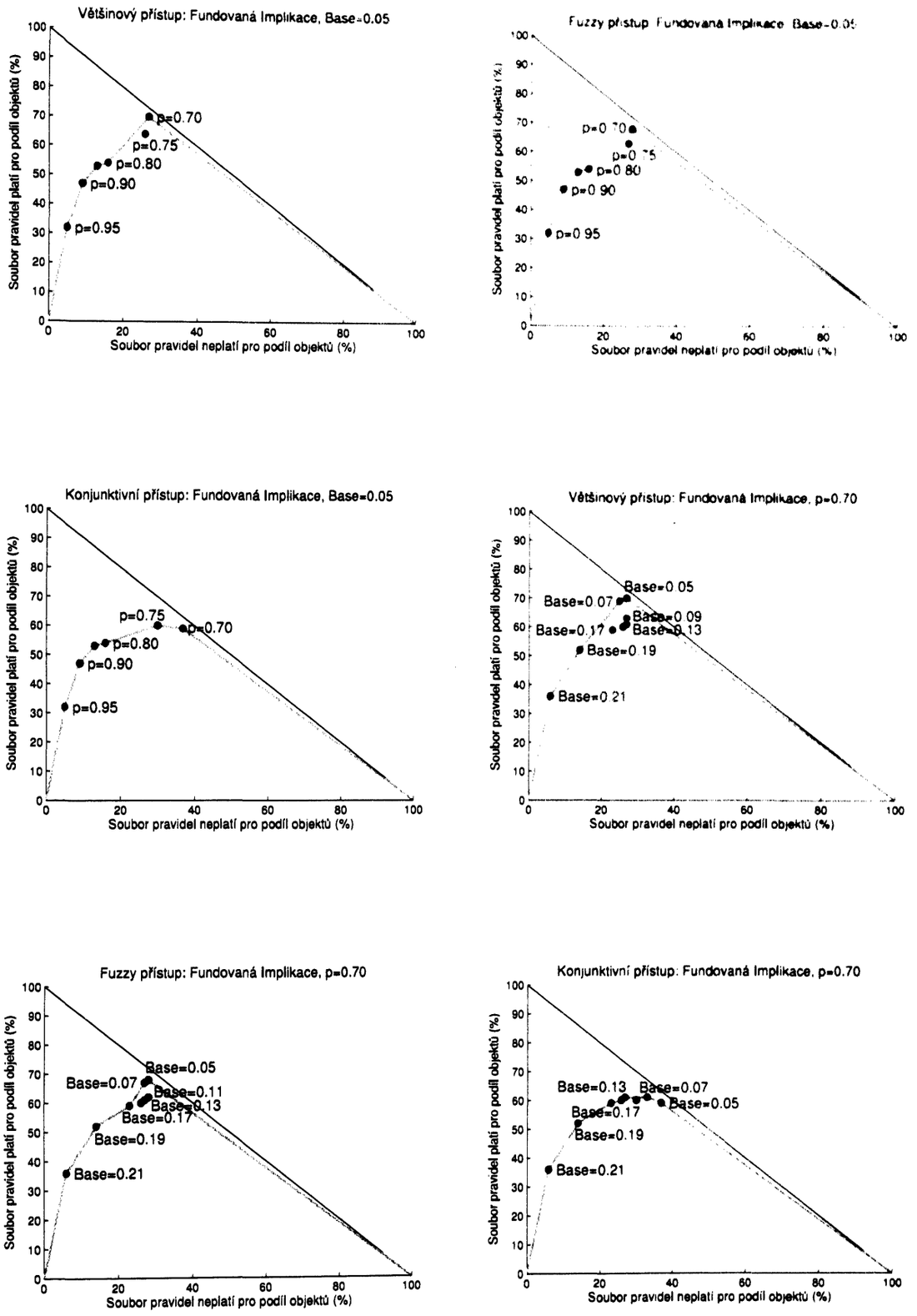
GUHA kvantifikátor	Parametr	Od	Do	Krok
Fundovaná implikace	p	0.70	0.95	0.05
Fundovaná implikace	Base	0.05	0.21	0.02
Dolní kritická implikace	p	0.60	0.85	0.05
Dolní kritická implikace	α	0.001	0.1	-
Fundovaná ekvivalence	p	0.60	0.80	0.05
Fundovaná ekvivalence	Base	0.05	0.21	0.02
Fisherův kvantifikátor	α	0.001	0.1	-

Tabulka 4.6: Seznam zkoumaných GUHA kvantifikátorů a rozsahy parametrů pro metodu 4ft-Miner

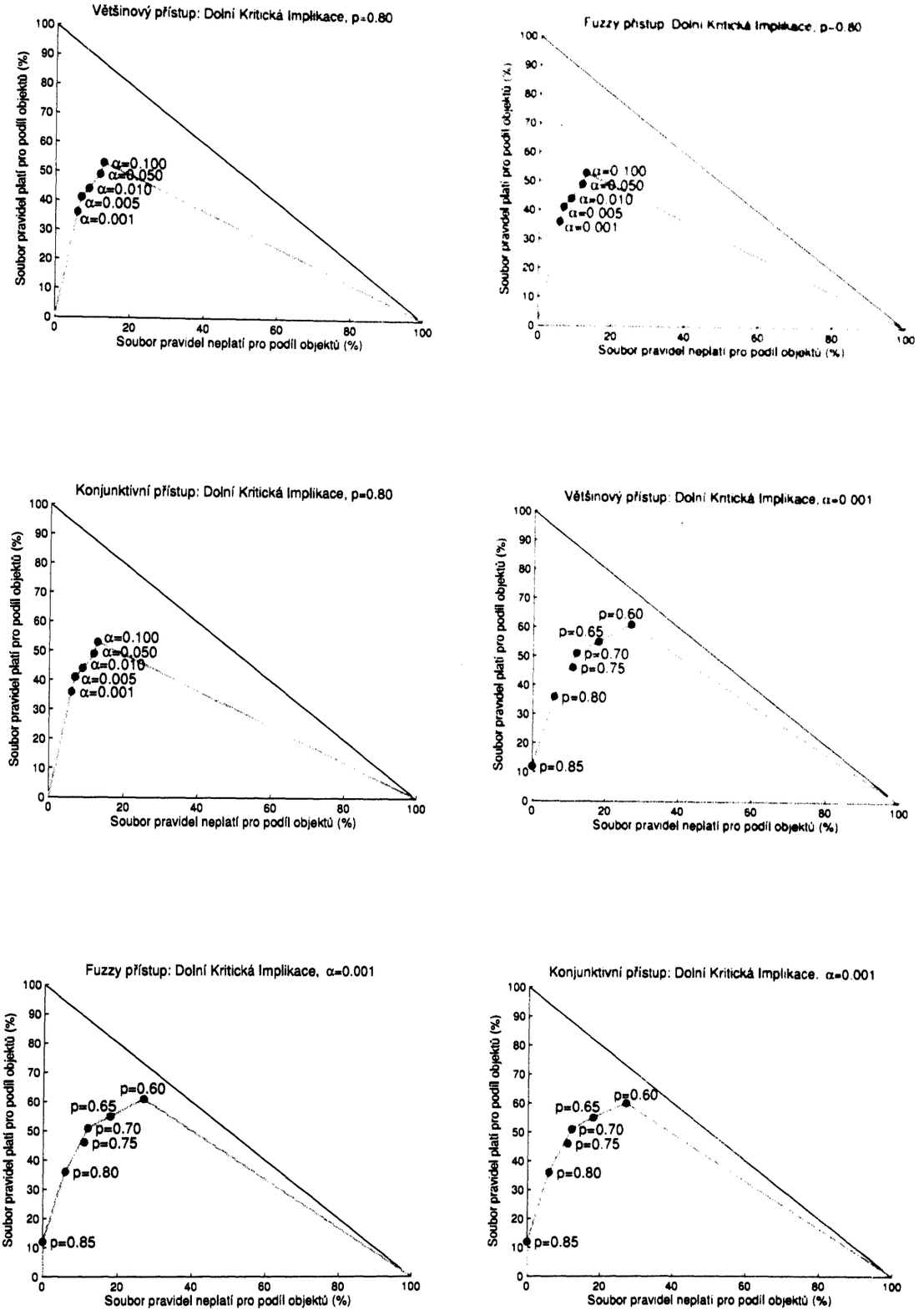
Metodou AQ21 se podařilo, při použití obou modifikací ATF i PD, získat soubory pravidel, jejichž spolehlivost nebyla příliš nízká (obrázek 4.23), avšak nedosahovala spolehlivosti souborů pravidel získaných z metody 4ft-Miner. Zkoumané parametry a volené hodnoty jsou zachyceny v tabulce 4.2.

Vliv změny počtu neuronů na kvalitu souborů pravidel získaných pomocí metody založené na fuzzy-neuronové síti opět uvádíme při volbě obou logik a tří funkcí příslušnosti v případě fuzzy přístupu. Význačným jevem bylo, že nejvyšších hodnot spolehlivosti dosahovaly soubory získané ze sítě s jedním neuronem ve druhé skryté vrstvě, což demonstruje obrázek 4.24. Při použití více neuronů byla spolehlivost stejná nebo nižší.

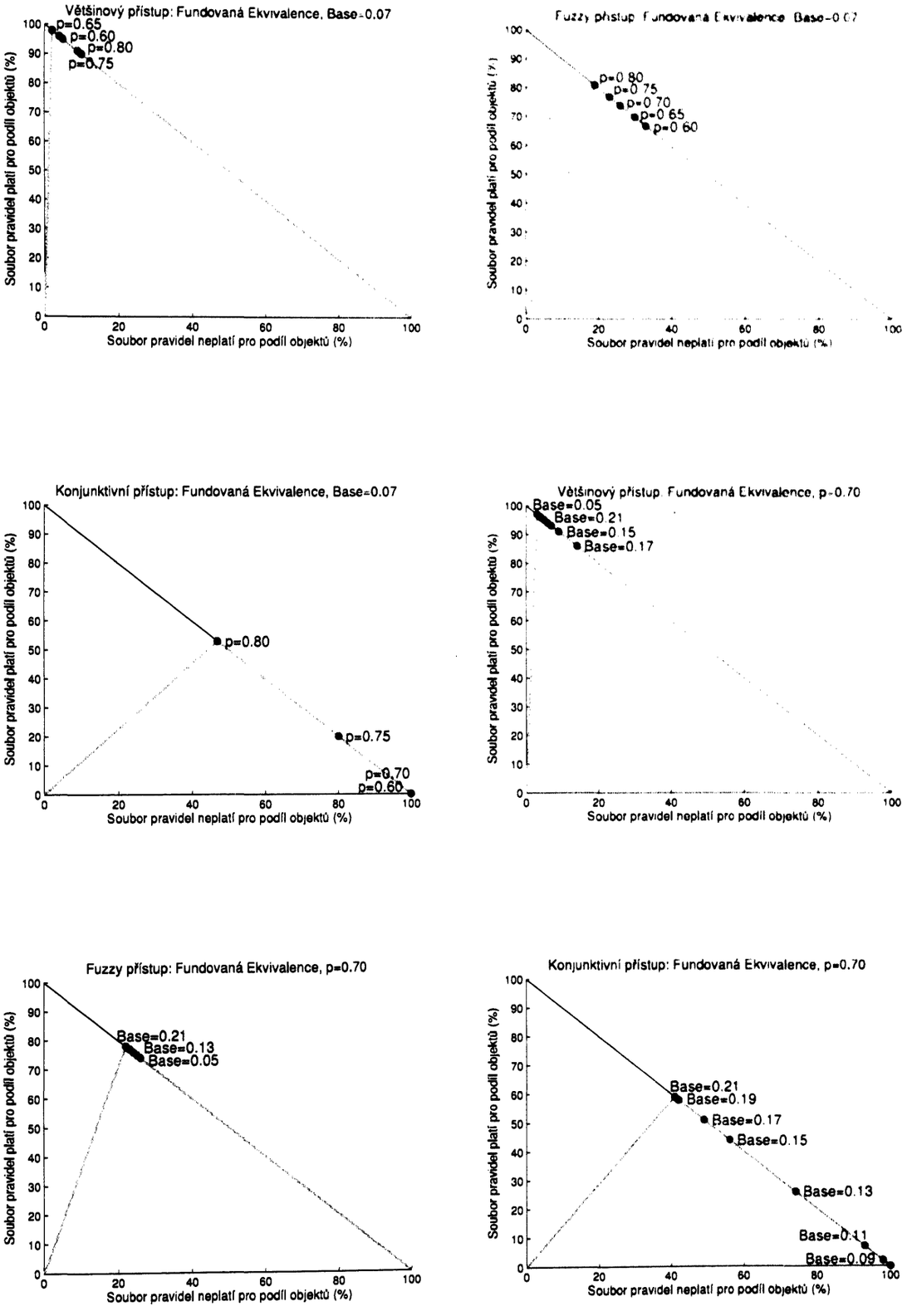
Soubory pravidel získané pomocí rozhodovacích stromů měli vysokou spolehlivost a navíc mají zaručené pokrytí rovno jedné. Vlastní hodnoty měr jsou zaznamenané v tabulce 4.7 a rozsah spolehlivosti souborů pravidel získaných ze stromů o různé velikosti je ukázán na obrázku 4.27.



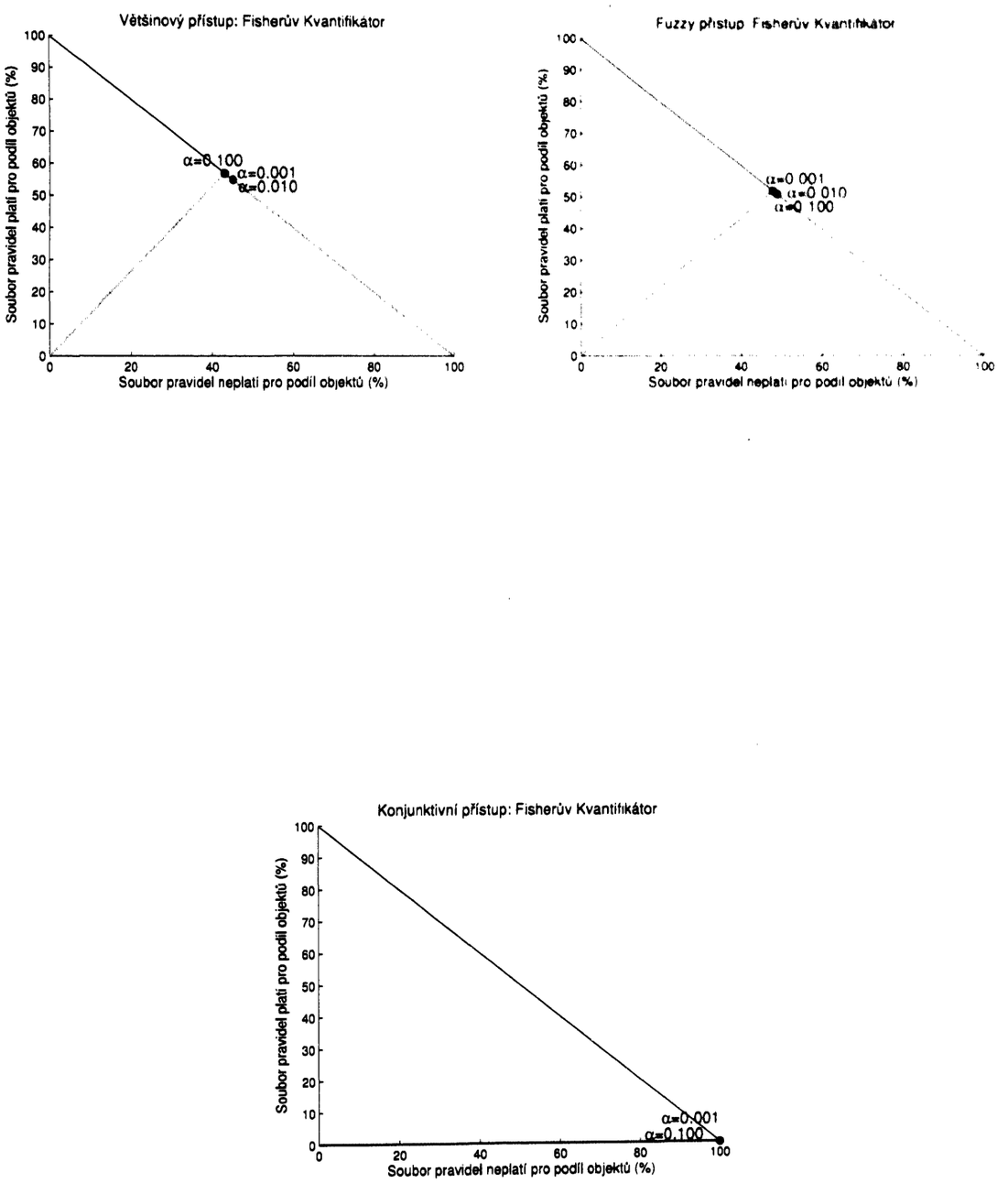
Obrázek 4.19: 4ft-Miner: Fundovaná implikace



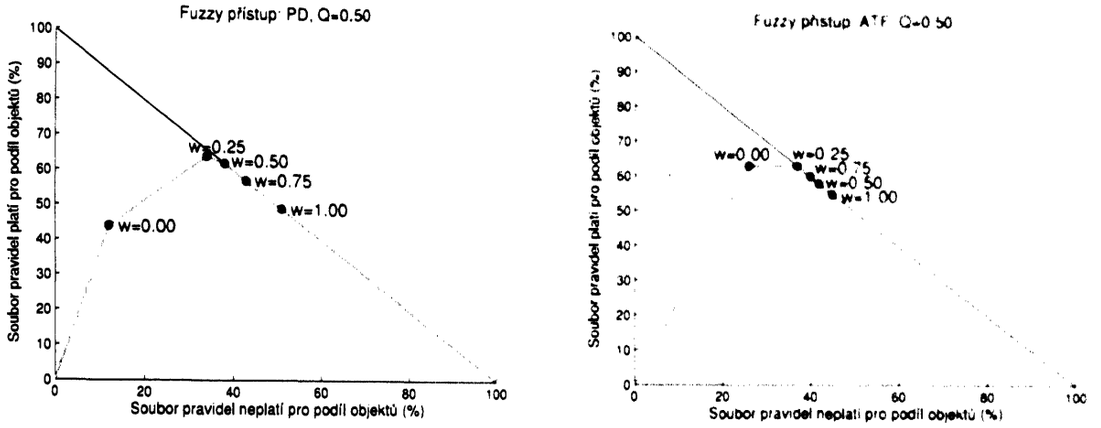
Obrázek 4.20: 4ft-Miner: Dolní kritická implikace



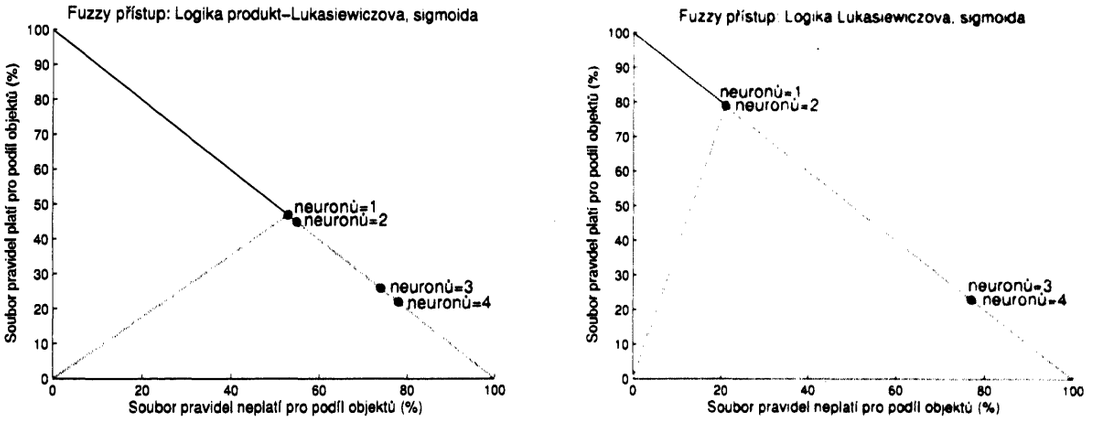
Obrázek 4.21: 4ft-Miner: Fundovaná ekvivalence



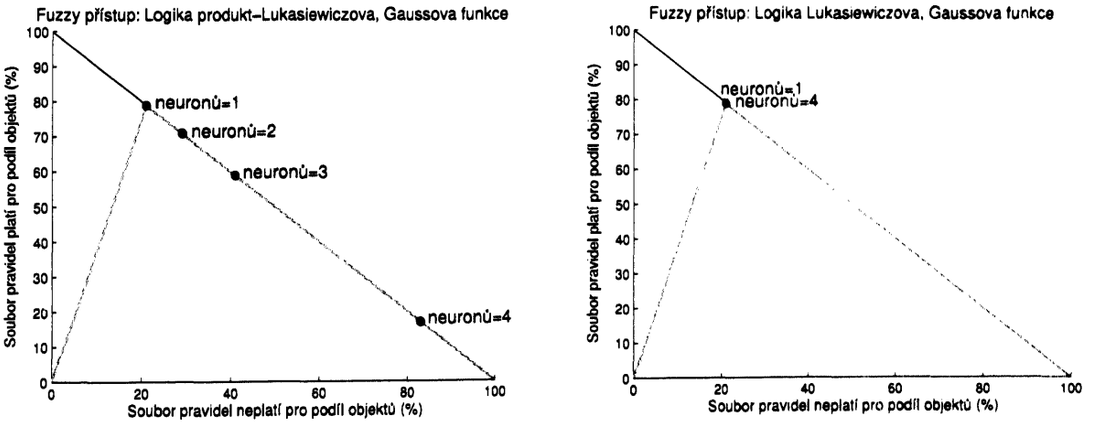
Obrázek 4.22: 4ft-Miner: Fisherův kvantifikátor



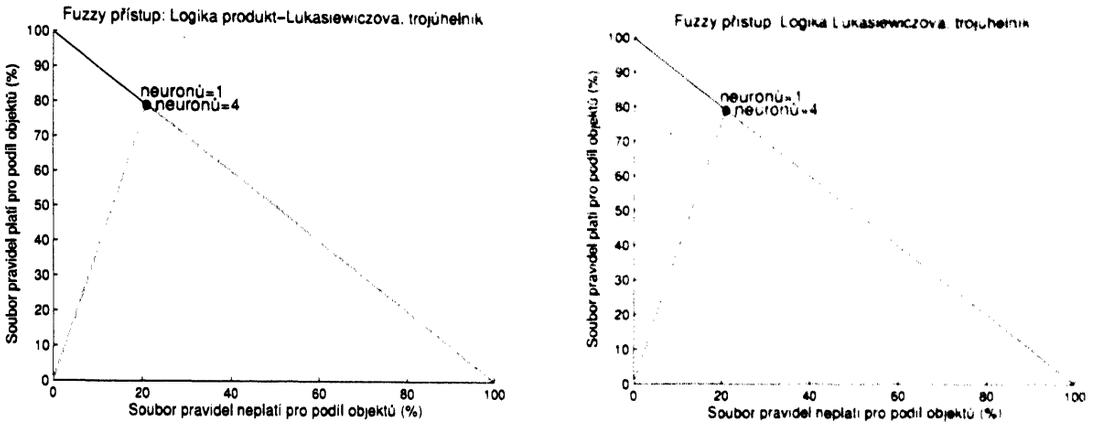
Obrázek 4.23: AQ21: Změna parametru w v míře kvality popisu



Obrázek 4.24: Fuzzy-neuronové sítě: sigmoida



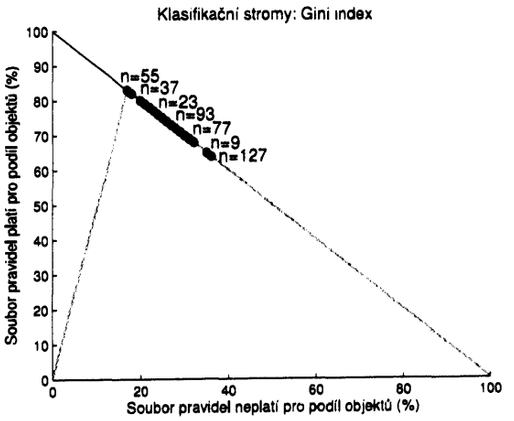
Obrázek 4.25: Fuzzy-neuronové sítě: Gaussova funkce



Obrázek 4.26: Fuzzy-neuronové sítě: trojúhelník

	Správnost	**Spolehlivost**	**Pokrytí**	**Podpora**	**#**
Gini index	0.56	0.78	1.0	0.78	26
Deviace	0.56	0.78	1.0	0.78	16

Tabulka 4.7: Výsledky pro soubory pravidel získaných z klasifikačních stromu



Obrázek 4.27: Klasifikační stromy s různou velikostí

4.3.4 Výsledky pro data z elektroencefalografu

I pro data z elektroencefalografu uvádíme průběh fundované implikace při změně hodnoty p i Base a dále průběh dolní kritické implikace při změně hodnoty p i α. Volené hodnoty parametrů p a Base jsou zachyceny v tabulce 4.8 a hodnoty parametru α pro dolní kritickou implikaci byly 0.001, 0.005, 0.01 a 0.05. Kvalitu souborů pravidel s fundovanou ekvivalencí a Fisherovým kvantifikátorem neuvádíme. Soubory pravidel s Fisherovým kvantifikátorem byly příliš velké i při nízké hodnotě α a soubory pravidel s fundovanou ekvivalencí byly příliš rozsáhlé při volbě $p = 0.95$ a Base=0.11 a při zvýšení hodnoty Base byly vygenerované soubory již prázdné. Mezi jednotlivými přístupy k výpočtu charakteristiky souborů pravidel v tomto případě nebyl příliš výrazný rozdíl. příkladem jsou průběhy fundované implikace při změně hodnot p i Base na obrázku 4.28.

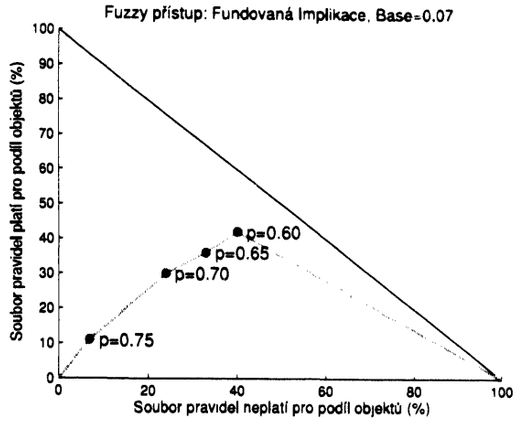
GUHA kvantifikátor	Parametr	Od	Do	Krok
Fundovaná implikace	p	0.60	0.75	0.05
Fundovaná implikace	Base	0.03	0.09	0.02
Dolní kritická implikace	p	0.60	0.80	0.05
Dolní kritická implikace	α	0.001	0.05	-

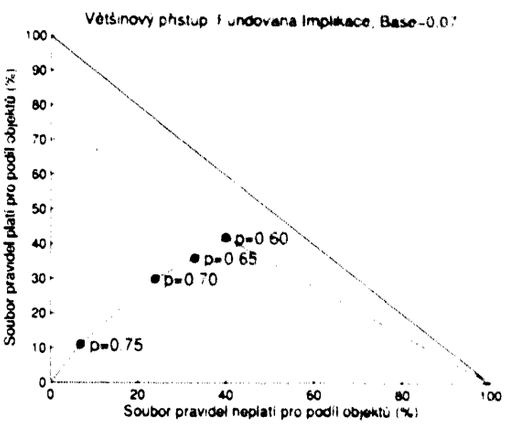
Tabulka 4.8: Seznam zkoumaných GUHA kvantifikátorů a rozsahy parametru pro metodu 4ft-Miner

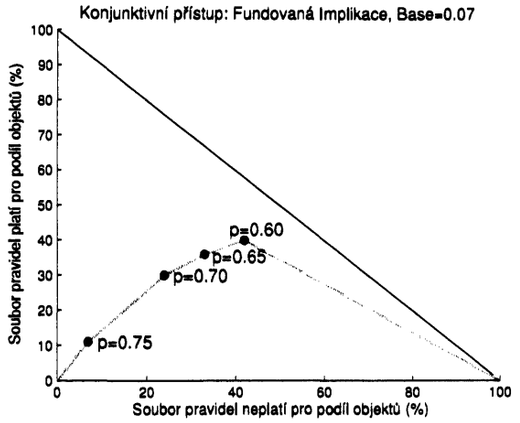
Zajímavá byla změna kvality souborů pravidel získaných pomocí metody AQ21 při různém nastavení parametru w. Pokud byla hodnota tohoto parametru nízká, spolehlivost souborů byla velmi vysoká. Naopak, v případě hodnoty parametru w blízké 1, byla spolehlivost velmi nízká. Pro modifikaci ATF i PD je zobrazena tato změna kvality souborů pravidel na obrázku 4.30 a nastavované hodnoty parametrů jsou popsány v tabulce 4.2.

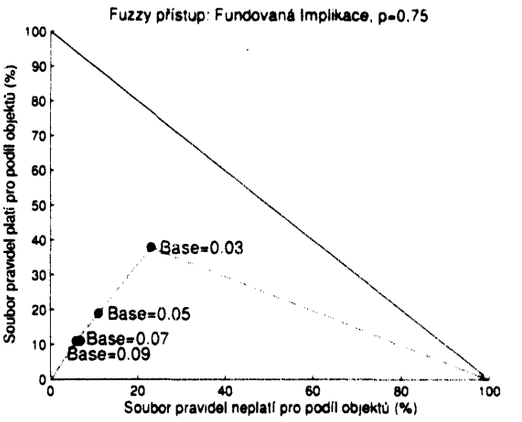
Pro metodu založenou na fuzzy-neuronových sítích nebylo možné uvést žádné výsledky, neboť používanou implementaci této metody nelze použít na data o takovém počtu atributů, které tato datová sada obsahuje.

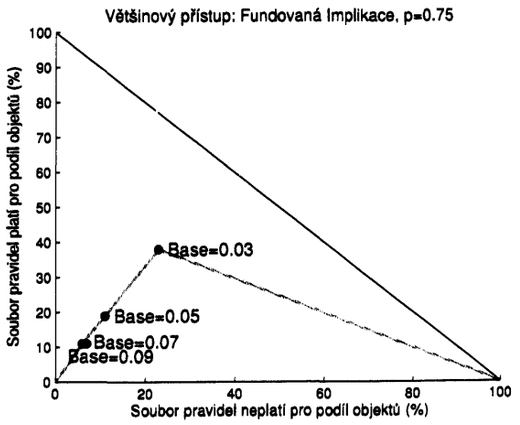
Jako poslední jsou v tabulce 4.9 uvedeny hodnoty pro soubory pravidel získané při použití obou indexů nečistoty a na obrázku 4.31 zobrazena změna spolehlivosti souborů pravidel závislých na velikosti klasifikačního stromu.

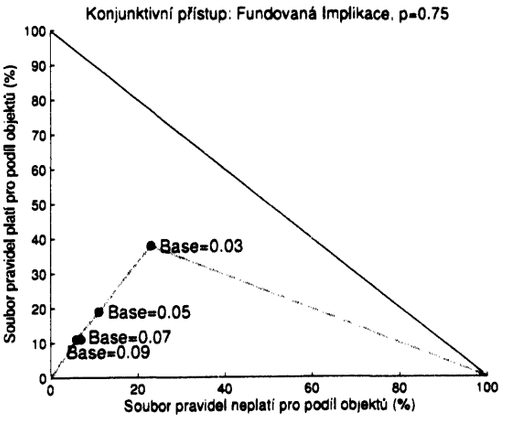




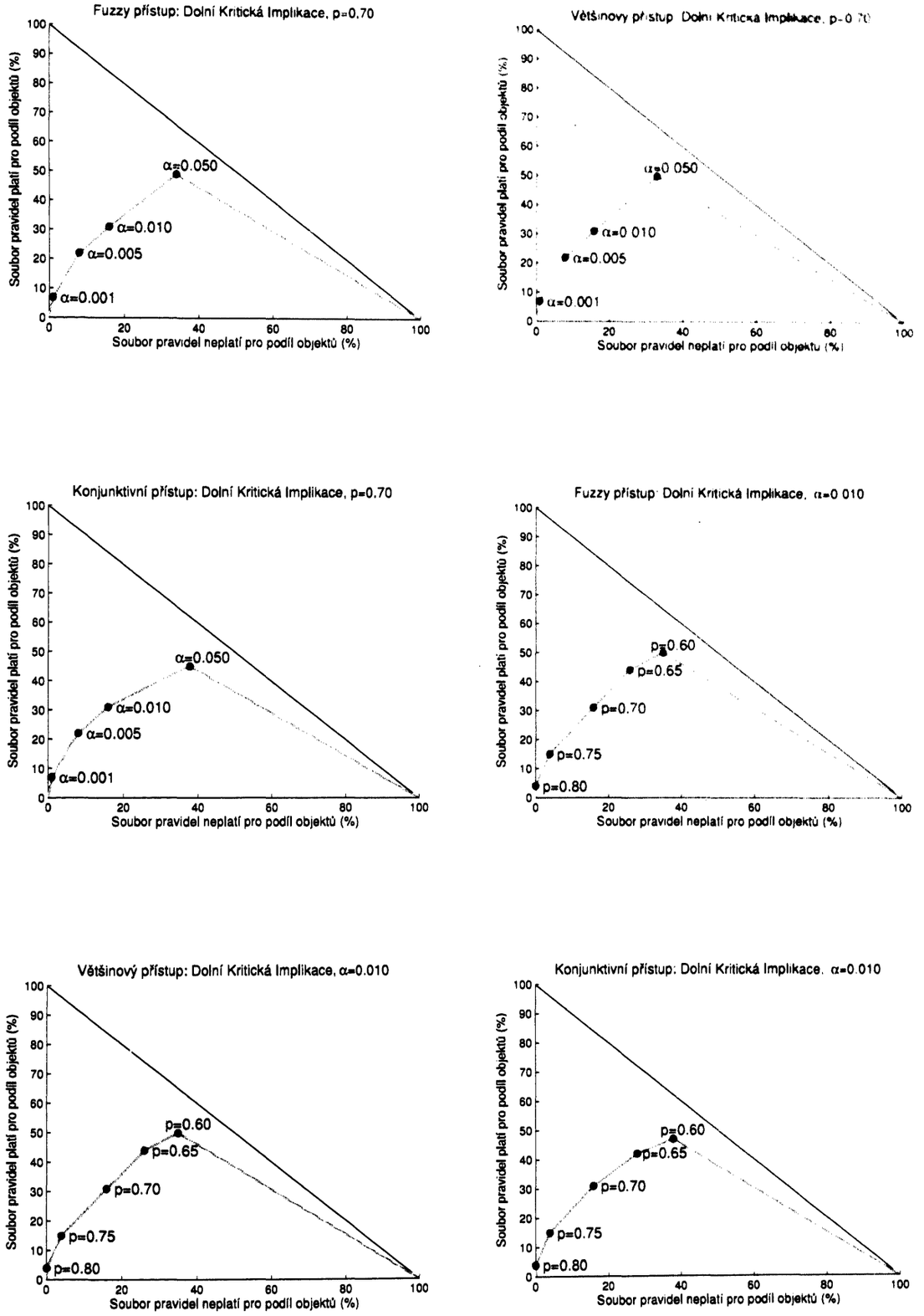




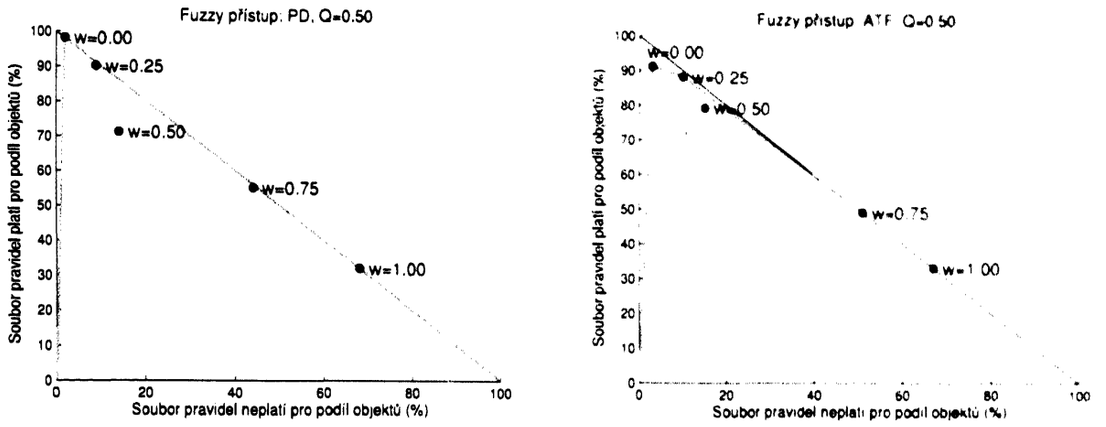




Obrázek 4.28: 4ft-Miner: Fundovaná implikace



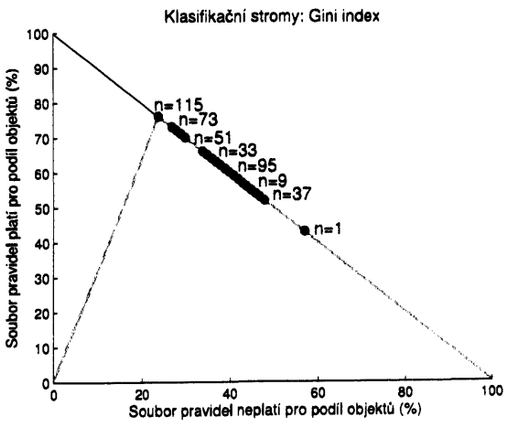
Obrázek 4.29: 4ft-Miner: Dolní kritická implikace



Obrázek 4.30: AQ21: Změna parametru w v míře kvality popisu

	Správnost	**Spolehlivost**	**Pokrytí**	**Podpora**	**#**
Gini index	0.36	0.68	1.0	0.68	29
Deviace	0.29	0.65	1.0	0.65	42

Tabulka 4.9: Výsledky pro soubory pravidel získaných z klasifikačních stromu



Obrázek 4.31: Klasifikační stromy s různou velikostí

Kapitola 5

Závěr

V první kapitole této práce jsme představili různé typy pravidel, které jsou nejčastěji používány k reprezentaci znalostí získaných z dat, a popsali jsme čtyři různé metody na jejich dobývání. Ve druhé kapitole jsme prozkoumali existující míry kvality, přičemž jsme zjistili, že většina těchto měr je zaměřena na měření pouze jednotlivých asociačních nebo rozhodovacích pravidel.

Ve třetí kapitole jsme navrhli způsob, jak měřit a srovnávat celé soubory pravidel různých typů, pomocí námi zavedené charakteristiky souboru pravidel. Uvedli jsme tři přístupy k výpočtu této charakteristiky a na jejím základě jsme definovali několik měr, které vystihují různé vlastnosti souboru pravidel. Navíc bylo ukázáno, že námi navržený přístup je ekvivalentní s již známou metodou měření souborů klasifikačních pravidel, a tedy ji dále rozšiřuje. Také jsme rozšířili teorii ROC prostorů na celé soubory pravidel různých typů a ukázali souvislosti mezi některými mírami pro klasifikační pravidla a pro asociační pravidla.

Navržený způsob jsme využili ke změření kvality souborů pravidel získaných z metod na dobývání znalostí popsaných v první kapitole, které jsme použili na čtyři různé vstupní datové soubory. Výsledky, které se nachází v páté kapitole, demonstrují navržený způsob a kladou si i za cíl prozkoumat, jak se změna jednoho parametru metody na dobývání znalostí promítne do změny kvality souborů pravidel.

Literatura

[1] Rakesh Agrawal, Tomasz Imielinski, and Arun N. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, D.C., May 26-28, 1993*, pages 207 216. ACM Press, 1993.

[2] Petr Berka. *Dobývání znalostí z databází.* Academia, 2003.

[3] Sergey Brin, Rajeev Motwani, and Craig Silverstein. Beyond market baskets: Generalizing association rules to correlations. In *SIGMOD 1997, Proceedings ACM SIGMOD International Conference on Management of Data, May 13-15, 1997, Tucson, Arizona, USA*, pages 265 276. ACM Press, 1997.

[4] Ivan Bruha and Sylva Kočková. Quality of decision rules: Empirical and statistical approaches. *Informatica (Slovenia)*, 17(3), 1993.

[5] Pedro Domingos. The role of Occam's razor in knowledge discovery. *Data Mining and Knowledge Discovery*, 3(4):409 425, 1999.

[6] J. Faber, M. Novák, P. Svoboda, and V. Tatarinov. Electrical brain wave analysis during hypnagogium. *Neural Network World*, 1:41 54, 2003.

[7] Peter A. Flach. The many faces of ROC analysis in machine learning, 2004.

[8] Alex A. Freitas. On rule interestingness measures. *Knowledge-Based Systems*, 12(5-6):309–315, October 1999.

[9] Johannes Fuernkranz and Peter A. Flach. ROC 'n' rule learning towards a better understanding of covering algorithms. *Machine Learning*, 58(1):39–77, January 2005.

[10] David J. Hand. *Construction and Assessment of Classification Rules.* John Wiley and Sons Ltd, 1997.

[11] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning.* Springer, 2001.

[12] Martin Holeňa. Získávání pravidel z dat. *Statistika*, 83(2):48 60, 2003.

[13] Kenneth A. Kaufman and Ryszard S. Michalski. An adjustable description quality measure for pattern discovery using the AQ methodology. *Journal of Intelligent Information Systems*, 14(2-3):199 216, 2000.

[14] Bing Liu, Wynne Hsu, and Shu Chen. Using general impressions to analyze discovered classification rules. In *Knowledge Discovery and Data Mining*, pages 31 36, 1997.

[15] Ken McGarry. A survey of interestingness measures for knowledge discovery. *Knowl. Eng. Rev.*, 20(1):39 61, 2005.

[16] Gregory Piatetsky-Shapiro. Knowledge discovery in databases: 10 years after. *SIGKDD Explorations*, 1(2):59 61, January 2000.

[17] Přemysl Posledník. *Extrakce pravidel fuzzy logiky z dat přímým prokládáním pravdivostních ohodnocení.* Fakulta jaderná a fyzikálně inženýrská, České vysoké učení technické, 2005.

[18] J. Ross Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81 106, 1986.

[19] Jan Rauch. Asociační pravidla a matematická logika. *Znalosti 2004*, pages 114–125, 2004.

[20] The official site of the LISp-Miner project. Dostupné z http://lispminer.vse.cz/.

[21] Machine Learning and Inference Laboratory, George Mason University. http://www.mli.gmu.edu/.

[22] Uci machine learning repository. Dostupné z ftp://ftp.ics.uci.edu/pub/machine-learning-databases/.

[23] Janusz Wojtusiak, Ryszard S. Michalski, Kenneth A. Kaufman, and Jaroslaw Pietrzykowski. Multitype pattern discovery via AQ21. *Reports of the Machine Learning and Inference Laboratory*, 2006.