

Zadání prvních dvou cvičení na následujících třech stranách je zaměřeno na seznámení se s induktivním generátorem rozhodovacích stromů a pravidel (demonstrační verze) See5/c5, dále na způsob přípravy dat pro trénování a testování, a to jak pro data reálná, tak umělá.

1) REÁLNÁ DATA IRIS

Pro zpracování dat **IRIS** bude použita demonstrační verze programu **See5** (od plné verze se liší tím, že má omezení na maximální počet trénovacích/testovacích příkladů ≤ 400). See5 je verze pro Windows, verze pro Unix/Linux má název **c5**, nekomerční předchůdce byl označen jako **c4.5** (viz např. <http://www.cse.unsw.edu.au/~quinlan> a <http://www.rulequest.com/>).

Data jsou v souboru **IRIS.TXT**. Každý řádek představuje jeden příklad, jehož struktura je následující:

$$a_1, a_2, a_3, a_4, C_i$$

kde a_j je numerická hodnota (reálné číslo, $j=1,2,3,4$) a C_i je klasifikace, $i=1,2,3$.

Data popisují parametry tří variant rostliny iris:

$$C_1=\text{Iris Setosa}, C_2=\text{Iris Versicolor}, \text{ a } C_3=\text{Iris Virginica}.$$

Význam atributů (některé parametry rostlin): a_1 je délka kališního lístku v cm, a_2 je šířka, a_3 je délka korunního plátku, a a_4 je jeho šířka.

Četnost každé ze tří tříd je zastoupena v datech 1/3 (po 50, tj. celkem je 150 příkladů dat). Vlastnosti jednotlivých tříd: každá třída je lineárně separovatelná od ostatních dvou (které od sebe ale lineárně separovatelné nejsou). Řešený problém je, zda na základě hodnot čtyř atributů (délka a šířka kališního lístku a korunního plátku) lze spolehlivě klasifikovat (zařadit do tříd, odlišit od sebe) jednotlivé varianty rostliny; zároveň je otázkou, zda je to možné pomocí těchto atributů nebo zda by bylo nutno ještě nějaké jiné atributy přidat, anebo zda by bylo možno některý atribut (nebo některé atributy) ze zadaných čtyř vynechat bez omezení přesnosti klasifikace (případně jen s malým poklesem přesnosti). Použité atributy se objeví ve vygenerovaném stromu, v kořeni je atribut nejvýznamnější (z hlediska entropie, na jejímž využití je generátor rozhodovacích stromů See5/c5/c4.5 založen—postupně jsou hledány atributy, které rozdělují nehomogenní množiny dat do homogennějších, v listech stromu jsou pak množiny homogenní, v nejhorším případě jednoprvkové). V uzlech stromu jsou testy na hodnoty atributů (probírá se na přednáškách).

ZADÁNÍ

Pomocí příkladů z dat **IRIS** a programu **See5** vytvořte rozhodovací stromy. Postupně experimentujte podle následujících pokynů a), b) a c). Výsledky si zaznamenejte.

a) Vygenerujte jeden rozhodovací strom pomocí všech dat **IRIS** (překopírujte údaje z **IRIS.TXT** do **IRIS.DATA**, soubor popisu dat **IRIS.NAMES** je vždy stejný). Zaznamenejte si výsledek (See5 ukládá výsledky do souboru s příponou **out**, např. pro **IRIS.NAMES** a **IRIS.DATA** vznikne textový soubor **IRIS.OUT**).

b) Rozdělte data **IRIS** na **IRIS.DATA** a **IRIS.TEST**. Soubory, které musí obsahovat různé příklady (aby test byl na jiných datech než trénování) vytvoříte ze základního souboru **IRIS.TXT** náhodným výběrem (s rovnoměrným rozložením) příkladů (řádků). Pro náhodný výběr a přenos příkladů si napište program v libovolném jazyce. Vzhledem k náhodnému výběru nezáleží na pořadí příkladů v původním souboru **IRIS.TXT**. Množinu všech dat (tj. 150 příkladů) dělte na trénovací a testovací postupně na různé velké poměry (např. **DATA/TEST** může být 140/10, 130/20, ..., 75/75, případně i jinak, aby však dělení dávalo smysl z hlediska testování naučeného výsledku). Sledujte, jak se mění chyba klasifikace pro trénovací a testovací data v závislosti na poměru dat. Existuje-li testovací soubor s příponou **.TEST**, program automaticky po vytvoření stromu provede testování výsledku na testovacích datech. Použijte **See5** jak pro vytvoření rozhodovacího stromu (pomocí trénovacích dat **IRIS.DATA**), tak i pro otestování vzniklého stromu na oddělených datech **IRIS.TEST**. Zaznamenejte si výsledek klasifikace (klasifikační přesnost v procentech) pro oba případy, tj. trénovací i testovací data a údaje srovnajte.

c) Pro trénování a testování použijte pětinasobné křížové ověření (cross-validation): data náhodně rozdělte na 5 částí **A**, **B**, **C**, **D**, **E** tak, aby každá část obsahovala stejný počet příkladů pro každou třídu (tj. 10 příkladů pro C_1 , 10 pro C_2 a 10 pro C_3 , celkem 30, takže $5 \cdot 30 = 150$). Výběr ze základního souboru proveďte náhodně, přičemž pravděpodobnost výběru určitého příkladu (řádku) je stejná pro každý příklad (1/150). Každá podmnožina **A**, ..., **E** musí být disjunktní (vyskytne-li se určitý příklad v **A**, nesmí se objevit v žádné ze zbývajících podmnožin **B**, ..., **E**). Pak proveďte pět fází trénování a testování:

c1) trénovat pomocí **A**, **B**, **C**, **D**, testovat pomocí **E**; c2) trénovat s **A**, **B**, **C**, **E**, testovat s **D**; ... c5) trénovat s **B**, **C**, **D**, **E**, testovat s **A**

(všechny příklady se postupně vystřídají na trénování a testování). Sledujte chybu na trénovací a testovací množině, aby nedošlo k přetrénování. Zaznamenejte si chybu trénování i testování pro každý z pěti případů a výsledné chyby trénování a testování spočítejte jako aritmetický průměr. Potom proveďte cross-validation přímo se zabudovanou funkcí programu **See5** (v menu **Classifier Construction Options** zatrhněte **Cross-validation** a zadejte hodnotu 5 místo implicitních 10). Dosažené výsledky srovnajte, stručně statisticky zhodnoťte a vyvoďte závěry.

Obdobné experimenty proveďte i s dalšími daty (**labour-neg**, **letter**, **vote**...), jejichž význam by měl být jako komentář v příslušných souborech **.NAMES**. Některá data již mají testovací množiny vytvořeny. Výpočty probíhají velmi rychle.

2) UMĚLÁ DATA

Ve strojovém učení se často používá trénování a testování pomocí umělých dat před tím, než se metoda vyzkouší na reálných datech. K prozkoumání vlastností generátoru rozhodovacích stromů `See5` si vygenerujte (v libovolném programovacím jazyku) trénovací a testovací data podle náčrtku na přiloženém obrázku. Obrázek znázorňuje data ve dvourozměrném prostoru, kde jsou známy souřadnice bodů dvou klasifikačních tříd (znázorněné znaky “+” a “o”), přičemž neznámá hranice (lineární i nelineární) oddělující třídy od sebe je znázorněna čárkovaně.

Úkolem rozhodovacích stromů, generovaných pro uvedená data (6 druhů rozmístění dat), je naučit se, kam náležejí členové jednotlivých tříd a pro hodnoty, nepoužité pro trénování (tzv. testovací data), správně určit pomocí klasifikace příslušnost do patřičné třídy.

Definujte si skupiny souborů, např. `KRUH.NAMES`, `KRUH.DATA`, `KRUH.TEST`, dále např. `TROJUHELNÍK.NAMES`, `TROJUHELNÍK.DATA`, `TROJUHELNÍK.TEST`, a podobně pro všech 6 skupin (každý druh dat má tři soubory, přičemž název před tečkou *musí* být stejný u jednoho druhu, rozlišení je pomocí extenze za tečkou).

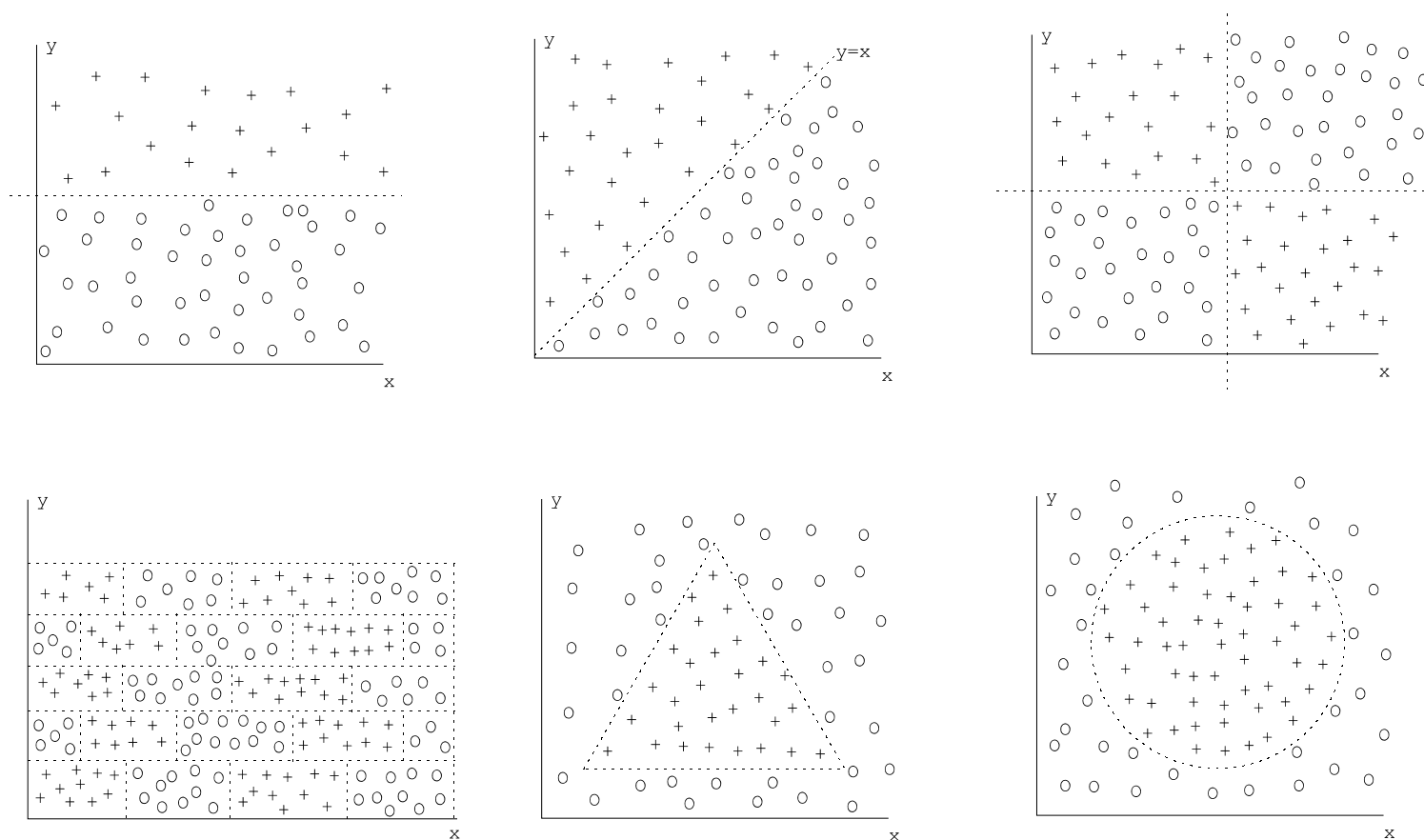
Pro data obou tříd vždy uvažujte např. rozmezí $0 \leq x \leq 100$ a $0 \leq y \leq 100$. Data jsou ve svých oblastech rozmístěna rovnoměrně; počet členů obou tříd je stejný. Data jsou *diskrétního* typu (pro hodnoty souřadnic ovšem použijte náhodně generovaná reálná čísla v daném rozmezí).

Testovací data necht' tvoří postupně cca 5%, 10% a 30% z celkového počtu dat (trénovací a testovací množina jsou navzájem disjunktní; testovací data vytvořte náhodným výběrem z celkové množiny vygenerovaných dat).

Postupně zvolte celkový počet bodů v každé třídě 50, 200, 1000, a pro každý počet aplikujte výše uvedená tři rozdělení na trénovací a testovací data, tj. pro každý ze šesti typů dat je 9 experimentů, tedy celkem 54 experimentů.

Zaznamenejte si klasifikační přesnost pro každý experiment, výsledky porovnejte a udělejte závěr, jak rozhodovací strom založený na entropii klasifikuje různé typy a různé počty dat. Pro který typ dat při každém daném počtu bodů (tj. trénovacích příkladů) je chyba nejmenší a největší? A z jakého důvodu se chyba mění pro různé typy dat? Liší se chyba na trénovacích a testovacích datech? Jak mnoho pro různé typy dat?

Pozn.: Uvedený postup využití různě generovaných umělých trénovacích a testovacích dat je obecně použitelný pro různé typy metod strojového učení (např. neuronové sítě aj.). Lze pozorovat např. schopnost oddělit třídy s lineárními a nelineárními hranicemi atd., dále závislost klasifikační přesnosti na množství trénovacích a testovacích dat, apod.



3) EXPERIMENTY S REÁLNÝMI DATY TYPU HYPOTHYROID

V dalším experimentu si vyzkoušejte stručně modifikaci dat a výsledek poskytovaný programem See5 včetně jednoduché analýzy výsledku:

a) Pomocí **See5** a souboru dat **HYPO.*** (reálná medicínská data týkající se diagnostiky hypotyreózy—zbytnělá štítná žláza; jedná se o záznam vyšetření pacientů z různých hledisek) vygenerujte příslušný rozhodovací strom včetně převodu na pravidla (v menu *Construct Classifier* je nutno zatrhnout parametr *Rulesets*). Testovací data (nepoužitá při trénování) jsou v souboru **HYPO.TEST** a See5 otestuje přesnost klasifikace automaticky, pokud příslušný testovací soubor najde v tomtéž adresáři, jako jsou trénovací data a popis dat. Výsledek si prohlédněte a zaznamenejte (vznikne **HYPO.OUT**).

b) Dále modifikujte data tak, abyste zaměnili doposud závislou proměnnou (poslední sloupec—viz též popis struktury dat v souboru **HYPO.NAMES**) za doposud nezávislou proměnnou (atribut) nazvanou **SEX** (je nutno prohodit datové sloupce v **HYPO.DATA** i v **HYPO.TEST** tak, aby atribut **SEX** byl posledním sloupcem). Na místo atributu **SEX** přesuňte atribut z posledního sloupce—jedná se tedy o prohození druhého a posledního sloupce. (Komentář na konci řádku za znakem "|" lze vypustit.)

Nezapomeňte na to, že pořadí definic atributů v *.NAMES musí odpovídat pořadí sloupců v *.DATA a *.TEST, takže je nutno upravit příslušně i soubor **HYPO.NAMES** vzhledem k záměně obou sloupců.

POZOR: Zkontrolujte v **HYPO.DATA** i v **HYPO.TEST**, které řádky obsahují znak "?" (otazník jako neznámý údaj) pro hodnotu atributu **SEX**. Tyto řádky z **HYPO.DATA** i **HYPO.TEST** vyřaďte; mělo by se jednat např. o řádek č. 8 v **HYPO.DATA**, č. 2 v **HYPO.TEST** a další. **Pokud je atribut použit jako závislý, pak musí mít definovány klasifikace pro všechny kombinace hodnot v datech**, takže **řádky s neznámými klasifikacemi** je nutno při modifikaci **vyloučit**.

Po modifikaci datových souborů opět vygenerujte strom a pravidla, a prostudujte výsledek z hlediska složitosti a správnosti výsledku. Vzhledem k tomu, že byla zvolena proměnná **SEX** jako závislý atribut, je logické očekávat, že v řádcích s hodnotou **PREGNANCY=T** (tj. pozitivní nález těhotenství, *true*) bude vždy zařazení do třídy **F** (female=žena) a nikoliv do třídy **M** (male=muž).

Je tomu tak vždy správně—funguje strom a pravidla?

Dále prostudujte způsob zařazování do výsledných dvou tříd a promyslete si, na čem asi klasifikace závisí (pokud systematicky na něčem závisí nebo nezávisí, kromě atributu **PREGNANCY**) a zda se projevuje nebo neprojevuje vliv irelevance atributů. Všimněte si, zda součástí stromu je někde rozumný test na závislost výsledku klasifikace podle **PREGNANCY**—ano či ne? Pokud ano, kde a v jaké formě?

Dále si všimněte, zda v pravidlech je někde pravidlo, vyjadřující co nejefektivněji výsledek klasifikace podle **PREGNANCY** (což by měl být zcela zjevně relevantní atribut). Je někde ve stromu patrná relevance atributu **PREGNANCY**? Pokud ano, kde a v jaké formě?

4) POKROČILEJŠÍ VLASTNOSTI See5/c5 (pro bližší zájemce)

Pro podrobnější prozkoumání možností a vlastností generátoru rozhodovacích stromů (současné verze) založeného na minimalizaci entropie lze doporučit zejména následující:

- vyzkoušet si, jestli a jak se mění kvalita klasifikátorů, když např. počet trénovacích příkladů v různých třídách je silně nevyvážený (stačí pokusy se dvěma klasifikačními třídami) a zároveň jak kvalita závisí na počtu příkladů ve třídách;
- vyzkoušet si *crossvalidation* poskytovanou programem;
- vyzkoušet si na obtížnějších datech (např. *letters* apod.) klasifikaci “hlasováním” více stromy, tzv. *boosting*;
- vyzkoušet si různý *pruning* (tj. prořezávání stromu, které ovlivňuje schopnosti generalizace klasifikátoru);
- vyzkoušet si vliv překrytí hranic tříd;
- vyzkoušet si *fuzzy thresholds* (tj. zmírnění ostrosti oddělovací hranice tříd);
- prozkoumat vztah mezi stromem a jeho převodem na soubor pravidel (*rules*).

Je vhodné zkoumat See5/c5 jak s umělé vytvořenými daty, tak i s reálnými.

Je nutné si pročíst nápovědu v menu *Help* programu See5/c5 (je přirozeně v angličtině).