



Testování modelů a jejich výsledků

Jak moc můžeme věřit tomu, co jsme se naučili?



Nature Inspired
Technologies Group

† Osnova



- ❖ Úvod – různé klasifikační modely a jejich kvalita
- ❖ Hodnotící míry (kriteria kvality) pro zvolený model.
- ❖ Postup vyhodnocování modelu pro daná data
 - ◆ ladění parametrů pomocí validačních dat
- ❖ Tvorba trénovacích, validačních a testovacích dat. souborů
 - ◆ např. pomocí křížová validace, bootstrapping, ...
 - ◆ Problém různého zastoupení klasifikačních tříd, práce s nebalancovanými daty
- ❖ Porovnání různých modelů pro daná data
- ❖ Odhad budoucího výkonu modelu
- ❖ Křivka učení a problém přeučení

Úvod – jaké klasifikační modely máme k dispozici?



- ❖ I když jsme zatím probrali konstrukci jen 2 typy klasifikačních algoritmů, můžeme pro daná data zkonstruovat řadu různých modelů. Tyto mohou vzniknout volbou různých parametrů, kterými mohou být pro
 - ◆ **rozhodovací stromy** např. volba atributů pro reprezentaci dat, volba složitosti stromu, výběr kritéria pro volbu atributu do následujícího uzlu (minimální entropie, maximální zisk, cena, ..)
 - ◆ **n -nejbližších sousedů** volba parametru n , volba metriky vzdálenosti, volba váhy některých atributů ...
- ❖ Časem se seznámíme i s dalšími klasifikačními metodami, např. neuronové sítě, ..

Úvod – kvalita modelu pro daná data



- ❖ Jak dobře předpovídá klasifikaci neznámého objektu model, který jsme navrhli?
- ❖ Chyba, s jakou model klasifikuje na výchozích (trénovacích) datech *není* dobrým odhadem pro chování modelu na dosud neznámých datech
 - ◆ **Q: Proč?**
 - ◆ A: Nová data **nebudou přesně stejná** jako ta použitá pro učení!
- ❖ Na trénovacích datech můžeme vytvořit model s libovolně malou chybou! Ale testování takového modelu na nových datech obvykle dává špatné výsledky! Jedná se o problém **přeučení** (overfitting), viz konec dnešní přednášky.

Hodnocení klasifikátoru pomocí jeho relativní chyby



- ❖ Velmi přirozenou mírou je relativní chyba (*error rate*) vypočtená přes všechny uvažované instance :
 - ◆ *Úspěch (success)* : model pro danou instanci určí správnou třídu
 - ◆ *Chyba* : model pro danou instanci určí třídu špatně
 - ◆ **Relativní chyba**: procentuální podíl chybných instancí vůči mohutnosti celé uvažované množiny instancí
- ❖ *Chyba na trénovacích datech* je příliš optimistický odhad!
 - ◆ I náhodně vygenerovaný konečný soubor dat lze totiž popsat nějakým modelem, který nedělá chybu (třeba samotná výchozí tabulka)
 - ◆ Jak moc je takový model užitečný pro další rozhodování?

Nejčastěji používané míry pro klasifikátory



	Klasifikace modelem	
Správná klasifikace (daná uživatelem)	+	-
+	<i>TP</i>	<i>FN</i>
-	<i>FP</i>	<i>TN</i>

Celková správnost (*overall accuracy*) nebo **klasifikační přesnost**

$$Acc = (TP + TN) / (TP + TN + FP + FN)$$

Acc by se měla pohybovat v intervalu $\langle Acc_{def}, Acc_{max} \rangle$, kde

Acc_{max} je 1 (100%) pro model zcela konzistentní s daty

Acc_{def} je správnost modelu, který všechny instance řadí k majoritní třídě (= frekvence majoritní třídy)

Celková chyba (*overall accuracy*)

$$Err = (FP + FN) / (TP + TN + FP + FN)$$

$$Err + Acc = 1$$

Bereme-li v úvahu i **cenu chyby**, pak se pracuje s celkovou cenou chyby

$$T_Err_v = FP * c(P,n) + FN * c(N,p),$$

kde $c(P,n)$ je cena zařazení pozitivního příkladu mezi negativní

kde $c(N,p)$ je cena zařazení negativního příkladu mezi pozitivní

Další často používané míry



	Klasifikace modelem	
Správná klasifikace (daná uživatelem)	+	-
+	<i>TP</i>	<i>FN</i>
-	<i>FP</i>	<i>TN</i>

← *senzitivita*

← *specifická*

Správnost pro jednotlivé třídy se používá pro data, kde je zastoupení tříd silně nevyvážené $Acc_+ = TP / (TP + FP)$, $Acc_- = TN / (TN + FN)$,

Přesnost (*precision*) a úplnost (*recall*)

$$Přesnost = TP / (TP + FP)$$

$$Úplnost = TP / (TP + FN)$$

Senzitivita či **specifická** klasifikačního algoritmu - termíny převzaté z medicíny:

U nově nasazeného léku (léčebné metody) na nějakou chorobu se hodnotí

- **senzitivita** „u kolika opravdu nemocných lék zaberé“ a
- **specifická** „zda lék zabírá jen na tuto chorobu“.

$$Senzitivita = TP / (TP + FN)$$

$$Specifická = TN / (TN + FP)$$

Celková chyba (odchylka) v případě predikce numerických hodnot (při regresi) – např. součet čtverců (nebo abs. hodnot) odchylek od skutečné hodnoty

Postup hodnocení modelu



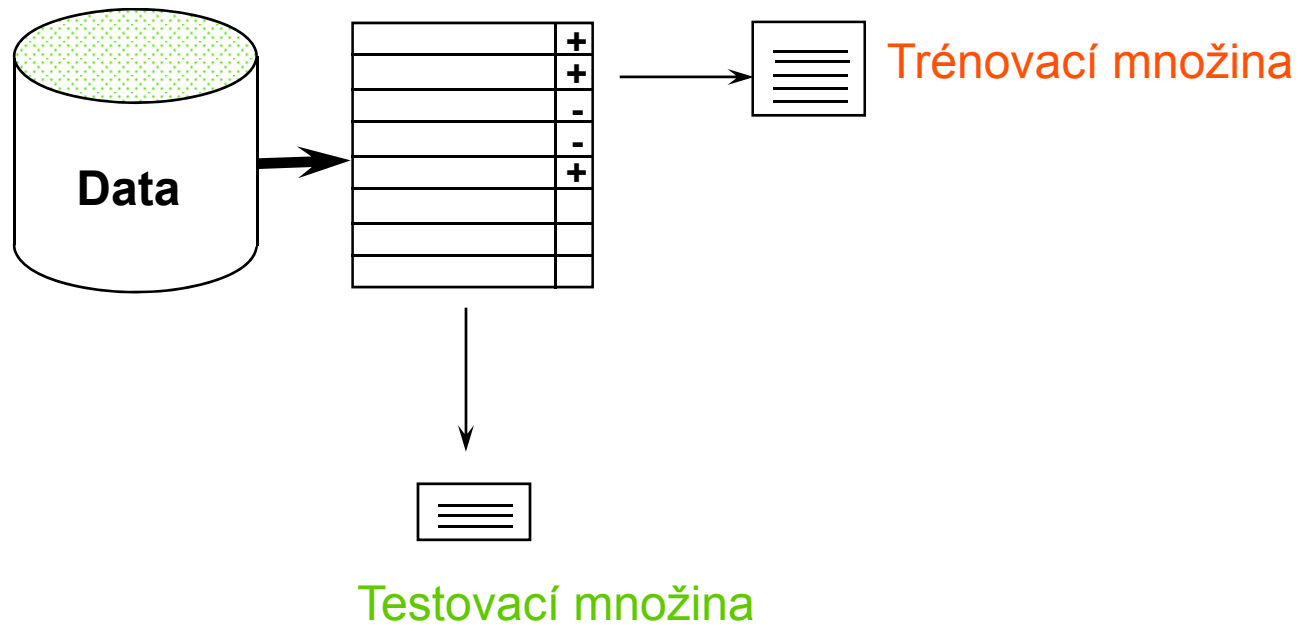
pro "ROZSÁHLÁ" data

- ❖ Máme-li hodně dat (tisíce instancí), které obsahují pro každou třídu dostatek vzorků (stovky instancí), pak stačí provést jednoduché testování:
 - ◆ Rozděl výchozí data náhodně do 2 množin: **trénovací** (asi 2/3 dat) a **testovací** (zbytek, tedy asi 1/3 dat)
 - ◆ Vytvoř klasifikační model nad *trénovací množinou* a proved' hodnocení (např. pomocí relativní chyby) na *testovací množině*.

Klasifikace - krok 1: Rozděl data na trénovací a testovací množinu



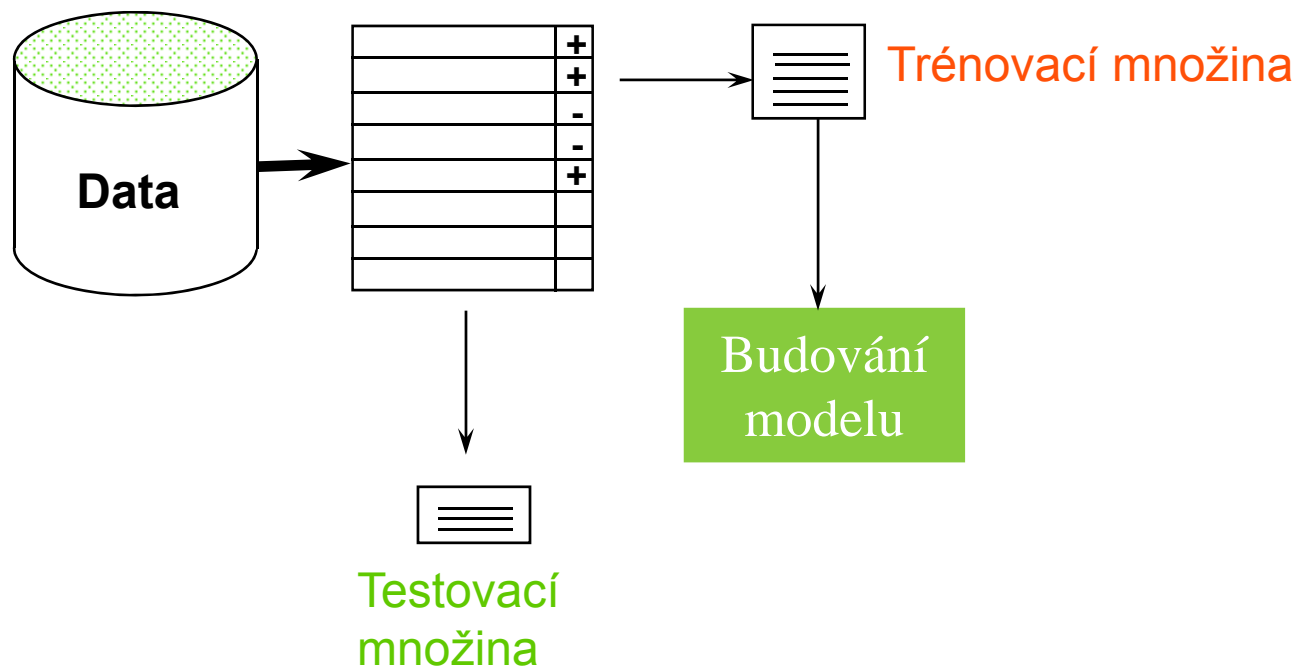
DATA se známými
výsledky klasifikace



Klasifikace - krok 2: Vytvoř model na trénovacích datech



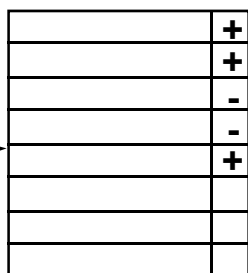
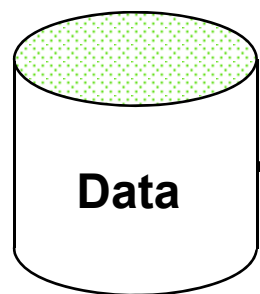
DATA se známými výsledky klasifikace



Klasifikace - krok 3: Otestuj model na test. datech (a případně zkus vytvořit jiný)



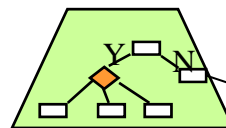
DATA se známými výsledky klasifikace



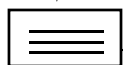
Trénovací množina



Při nedostatečné přesnosti nová volba algoritmu/jeho parametrů



Testovací množina



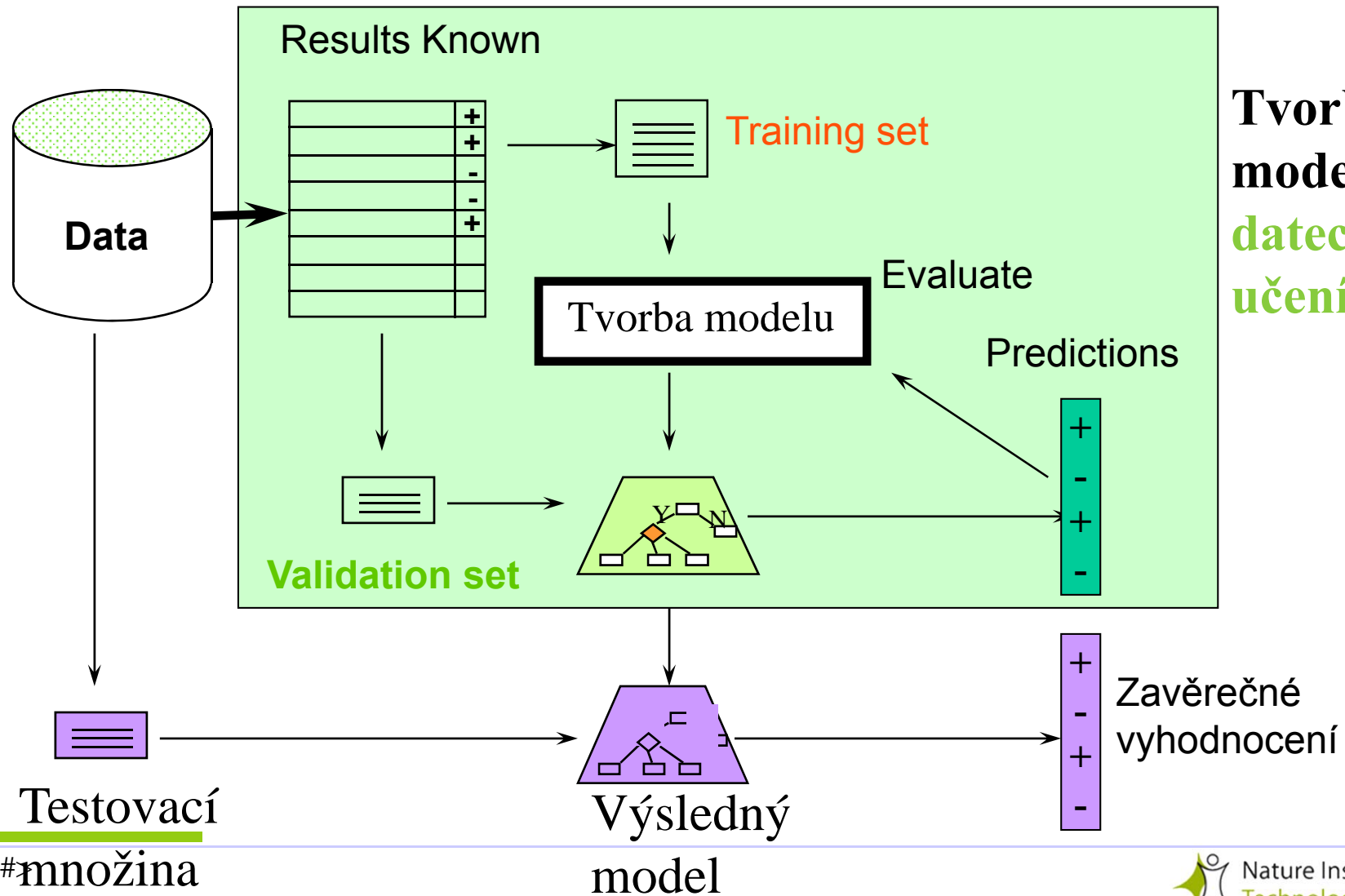
Hodnocení kvality modelu

† Poznámka k ladění parametrů



- ❖ Někdy učení modelu postupuje ve 2 krocích:
 - ◆ **Krok 1:** navrhne základní strukturu (např. rozhodovací strom)
 - ◆ **krok 2:** optimalizuje parametry zvolené struktury na validačních datech (která rozhodnou, jak moc se má vzniklý strom prořezat)
- ❖ Testovací model musí vzniknout tak, že *nijak nejsou použita* trénovací data! A to ani pro ladění parametrů!
- ❖ V tomto případě by korektní procedura měla používat 3 nezávislé množiny dat: **data pro učení** (rozdělená na **trénovací, validační**) a **data pro testování**

Klasifikace: dělení na množiny trénovací, validační a testovací



Metoda opakované zadržé (repeated holdout)



- ❖ Odhad pomocí zadržé může být upřesněn tím, že se proces vícekrát opakuje s různými vzorky
- ❖ Metoda opakované zadržé:
 - ◆ V každé iteraci je určitá část dat náhodně vybrána jako trénovací (s využitím stratifikace, je-li třeba)
 - ◆ Relativní chyby všech iterací se zprůměrují - výsledek je **celková relativní chyba**
- ❖ Ale pozor: různé testovací množiny se mohou překrývat.
Lze se tomu vyhnout?

Tvorba trénovacích a testovacích dat pomocí křížové validace



❖ ***k-ární křížová validace*** (*cross-validation*) zamezuje překrývání testovacích množin

- ◆ Krok 1: data jsou rozdělena do k disjunktních podmnožin stejné velikosti
- ◆ Krok 2: Každá podmnožina je použita právě jednou pro testování modelu vzniklého ze zbylých dat

❖ Často se ještě předem jednotlivé podmnožiny stratifikují

❖ Odhad chyby zvoleného modelu se pak získá jako průměr chyb pro jednotlivé testovací množiny

† Příklad na křížovou validaci:

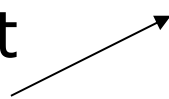


— Rozděli data data do skupin (folds) stejné velikosti



— Zadrž jednu skupinu na testování a zbytek použij pro trénování

Test



— Opakuj



Ještě o křížové validaci



- ❖ Standardní postup vyhodnocení: **stratifikovaná 10-násobná** (ten-fold) **křížová validace**
- ❖ **Proč 10?** Empirická zkušenost ověřená na řadě experimentů: odhady z této volby bývají velmi dobré!
- ❖ Stratifikace pak ještě zmenšuje variabilitu odhadu
- ❖ **Další vylepšení:** opakovaná stratifikovaná křížová validace
 - ◆ Např. opakuj 10x křížovou validaci se základem 10 (10-násobná KV) a zprůměruj výsledky

Křížová validace „vynech 1“



- ❖ „vynech 1“:
(zvláštní případ křížové validace):
 - ◆ Necht' počet skupin = počet výchozích dat
 - ◆ T.j., pro n výchozích instancí, vytvoř $n \times$ klasifikátor (z trénovacích dat o rozsahu $n-1$)

- ❖ **Vlastnosti:**
 - ◆ Optimální využití dat (důležité pro malé soubory)
 - ◆ Nepoužívá náhodné vzorkování
 - ◆ **Nevýhody:**
 - ❖ **výpočetně náročné** (výjimkou jsou některé neuronové sítě)
 - ❖ **stratifikace:** nelze žádným způsobem zajistit!

Metoda „bootstrapping“



- ❖ *Křížová validace* pracuje se vzorkováním *bez navracení* (*without replacement*)
 - ◆ Je-li určitá instance jednou vybrána do jedné skupiny, nemůže být vybrána podruhé do jiné
- ❖ *Bootstrapping* vytváří z výchozí množiny, která obsahuje právě n instancí dat, trénovací množinu takto:
 - Vytvoř **skupinu n instancí** tak, že budeš n krát vybírat z výchozí množiny dat (výběr *s navracením*)
 - Právě vybraná skupina se stane trénovací množinou.
 - Data z původní množiny, která se nedostala do trénovací množiny, tvoří **testovací** množinu.



† Jiný název: „0.632 bootstrap“



- ❖ Zdůvodnění pro výchozí množinu s n instancemi
 - ◆ **Konkrétní instance** má pravděpodobnost $(1-1/n)$, že při jednom výběru *nebude* vybrána do trénovací množiny
 - ◆ Pravděpodobnost, že tato konkrétní instance *nebude* vybrána ani v jednom z realizovaných n výběrů (a tedy se dostane do testovací množiny), je:

$$\left(1 - \frac{1}{n}\right)^n \approx e^{-1} = 0.368$$

- ◆ Z toho vyplývá, že testovací data budou obsahovat asi 36.8% instancí výchozích dat a trénovací data asi 63.2% .

Odhad chyby a bootstrapping



- ❖ Odhad chyby z testovacích dat, kterých je jen 36,8 %, je velmi pesimistický!
- ❖ Proto se doporučuje tento odhad upřesnit tím, že se kombinuje s odhadem z trénování

$$err = 0.632 \cdot e_{\text{test instances}} + 0.368 \cdot e_{\text{training instances}}$$

s tím, že resubstituční chyba (z trénovacích dat) má nižší váhu než chyba na testovacích datech!

- ❖ **Další upřesnění:**

Celý proces „bootstrapping“ se několikrát opakuje a výsledky zprůměrní !

Co s nerovnoměrným zastoupením tříd (*unbalanced data*)?



- ❖ Často máme data s nerovnoměrným zastoupením tříd
 - ◆ Únik zákazníků: 97% zůstane, 3% odchází (za měsíc)
 - ◆ Léč.diagnóza: 90% zdravých, 10% nemocných
 - ◆ eCommerce: 99% nekoupí, 1% koupí
 - ◆ Bezpečnost: >99.99% cestujících nejsou teroristé
- ❖ Model klasifikující do **majoritní třídy** bude dávat nízkou relativní chybu. **Ale není vůbec užitečný!**

Obdobná je situace i při klasifikaci do více tříd !

Vyvážení nevyvážených dat



- ❖ Klasifikace do 2 tříd: vytvoř **vyvážené** (*BALANCED*) **soubory dat na trénování** (vytvoření modelu) i na **testování**.
 - ◆ Vyber náhodně potřebný počet instancí klasifikovaných do minoritní třídy
 - ◆ a doplň je stejným množstvím náhodně vybraných instancí z majoritní třídy
- ❖ Zobecnění postupu "vyvážení" pro více tříd
 - ◆ Je nutné zajistit, aby v trénovací i testovací množině byl **počet instancí** pro každou třídu **zhruba vyrovnaný**

Řešení pro "malé soubory" dat



- ❖ Metoda zádrže (*holdout*) si ponechá určitou část dat pro testování a zbytek použije na trénování
 - ◆ Obvykle: 1/3 na testování, zbytek tvoří trénovací množinu
- ❖ Výsledné trénovací a testovací množiny nemusí být dostatečně reprezentativní v případě malých či nevyvážených souboru data.
 - ◆ Např. máme-li jen málo (nebo žádné) instance některé třídy – řeší to vytvoření **vyváženého** (*ballanced*) **vzorku**



Vyhodnocení na "stratifikovaném vzorku"



- ❖ **Stratifikovaný náhodný vzorek** musí respektovat i zastoupení vrstev, které jsou relevantní pro studovanou úlohu. *Např.* při studiu tělocvičných aktivit je třeba zohlednit např. věk, pohlaví a sociální status. Náhodný vzorek pak musí vzniknout v každé příslušné vrstvě zvlášť !
- ❖ **Stratifikovaný vzorek:** *pokročilá verze vyvažování dat*



KV „Vynech 1“



Pozor na to, že výsledek této KV dává dost hrubý odhad chyby - viz následující *extrémní příklad*:

Mějme výchozí soubor dat D se sudým počtem instancí 150, pro které je zvolena klasifikace NÁHODNĚ (ale pevně), a to tak, že $|D^+| = |D^-|$.

Na každé trénovací skupině necht' je vybrán ten model, který predikuje majoritní třídu. Jaká bude jeho úspěšnost?

- ❖ Při 10-násobné křížové validaci bude úspěšnost asi 50%.
- ❖ Jaký bude výsledek křížové validace „Vynech 1“? Úspěšnost je 0%! Proč?

† Odhad budoucího výkonu modelu



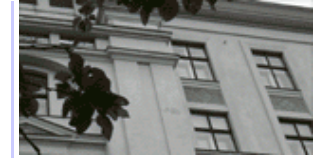
- ❖ Necht' je relativní chyba (počet chyb, které model udělá na testovacích datech) 25%. Jak to bude při reálném testování?
 - ◆ Záleží na množství testovacích dat.
- ❖ Predikce modelu na test.datech je podobná házení „cinknutou“ mincí (jedná se o příklad **Bernoulliho procesu** !):
 - ◆ „**Hlava**“ je „shoda mezi třídou skutečnou a tou, která je předpovězena modelem“, „**znak**“ je „neshoda ...“
 - ◆ Označme neznámou **p** pravděpodobnost správné predikce (padne „hlava“) – relativní správnost na testovacích datech ji jen odhaduje.
- ❖ Pro **Bernoulliho proces** nabízejí statistické tabulky **konfidenční intervaly**, které odhadnou odpovídající skutečnou hodnotu chyby (tedy interval, kam padne hodnocení **p** toho, jak je mince *cinknutá*) !

Konfidenční intervaly



- ❖ **Význam:** Úspěšnost klasifikace p leží uvnitř nějakého specifického intervalu s určitou mírou důvěry.
- ❖ **Příklad 1:** $S=750$ správně klasifikovaných příkladů pro $N=1000$ instancí
 - ◆ Odhad relativní úspěšnosti klasifikace: 75%
 - ◆ Jak spolehlivý je tento odhad relativní úspěšnosti klasifikace p ?
 - ❖ Odpověď: S pravděpodobností 80% je $p \in [73.2, 76.7]$
- ❖ **Příklad 2:** $S=75$ a $N=100$
 - ◆ Odhad relativní úspěšnosti klasifikace: 75%
 - ◆ S pravděpodobností 80% je $p \in [69.1, 80.1]$

Která ze 2 metod DM je lepší?



❖ Častá otázka!

Odpověď: obecně nelze posoudit! Záleží na úloze!

❖ **Přirozený postup** pro jednoduché srovnání úspěšnosti 2 různých metod: srovnej odhady chyb obou modelů při 10-nás. KV nad týmiž daty!

❖ **Problém?** Rozptyl v získaných hodnotách. Ten lze snížit při použití opakované křížové validace, např. při 10 opakováních KV:

- ◆ Výsledky prvního klasifikátoru jsou $\mathbf{x}_1, \dots, \mathbf{x}_{10}$
- ◆ Výsledky druhého klasifikátoru jsou $\mathbf{y}_1, \dots, \mathbf{y}_{10}$

❖ Zajímá nás rozdíl mezi oběma průměry. **Je zjištěný rozdíl statisticky signifikantní?**

Testy významnosti (*significance*)



- ❖ **Test významnosti** podává informaci o tom, jak moc můžeme důvěřovat tomu, že mezi oběma veličinami je skutečně rozdíl!
- ❖ *Nulová hypotéza*: mezi oběma veličinami **NENÍ** rozdíl
- ❖ *Alternativní hypotéza* : rozdíl mezi oběma veličinami **JE**
- ❖ Test významnosti kvantifikuje to, jak moc naše experimenty potvrzují to, že máme zamítnout nulovou hypotézu

- ❖ ***Tento problém řeší Studentův párový t-test !***

* Studentův párový t-test



Princip vychází z následující úvahy pro výsledky $x_1 x_2 \dots x_k$ a $y_1 y_2 \dots y_k$ získané při testování dvou modelů pomocí k násobné KV

❖ Máme-li dostatek vzorků, pak by průměr měl mít normální distribuci

- ♦ m_x a m_y jsou příslušné průměry
- ♦ Odhad pro rozptyl průměrů je σ_x^2/k a σ_y^2/k

$$\frac{m_x - \mu}{\sqrt{\sigma_x^2 / k}}$$

❖ Kdyby μ_x a μ_y byly skutečné hodnoty průměrů nad oběma modely, mělo by jít o **normální rozdělení** s průměrem 0 a rozptylem 1

$$\frac{m_x - \mu_x}{\sqrt{\sigma_x^2 / k}}$$

$$\frac{m_y - \mu_y}{\sqrt{\sigma_y^2 / k}}$$

William Gosset, Born:1876 in Canterbury; Died: 1937 in Beaconsfield, England

Obtained a post as a chemist in the Guinness brewery in Dublin in 1899. Invented the t-test to handle small samples for quality control in brewing. Wrote under the name "Student".



* Distribuce rozdílů



- ❖ Necht' $m_d = m_x - m_y$
- ❖ Rozdíl průměrů (m_d) má rovněž Studentovo rozložení s $(k-1)$ stupni volnosti
- ❖ Necht' σ_d^2 je rozptyl rozdílů
- ❖ Standardizovaná verze m_d se nazývá t -statistika:

$$t = \frac{m_d}{\sqrt{\sigma_d^2 / k}}$$

- ❖ Veličina t se používá pro realizaci t -testu

*Průběh testu



1. Zvol hladinu významnosti α
 - ◆ Je-li rozdíl signifikantní na hladině $\alpha\%$, pak s pravděpodobností $(100-\alpha)\%$ lze rozdíl prohlásit za významný
2. Sniž hladinu významnosti na polovinu (protože test je párový „2-tailed“)
 - ◆ Tj. skutečný rozdíl může být **+ve** nebo **-ve**
3. Najdi hodnotu z odpovídající $\alpha/2$
4. Když $t \leq -z$ nebo $t \geq z$, můžeme rozdíl prohlásit za významný
 - ◆ Tj. Nulovou hypotézu lze zamítnout!

* Nepárová pozorování



- ❖ Pokud odhady KV jsou získány z různých randomizací, nepovažují se za párové!

(stačí např. když se pro jeden model používá ***k***-násobná KV a pro druhý ***j***-násobná KV)

- ❖ V takovém případě používáme *nepárový* t-test s **$\min(k, j) - 1$** stupni volnosti
- ❖ Výsledná *t*-statistika

$$t = \frac{m_d}{\sqrt{\sigma_d^2 / k}} \quad \longrightarrow \quad t = \frac{m_x - m_y}{\sqrt{\frac{\sigma_x^2}{k} + \frac{\sigma_y^2}{j}}}$$

Interpretace výsledků



- ❖ Všechny naše odhady z KV vycházejí z výsledků získaných na témže souboru dat
- ❖ Tedy test říká pouze, jestli existuje rozdíl pro *úplnou* (*complete*) *k-násobnou* KV na tomto souboru dat
 - ◆ Úplná *k-násobná* KV generuje všechna možná disjunktní pokrytí dat vedoucí ke *k* skupinám a průměruje získané výsledky
- ❖ Ideálně by bylo nejlépe používat různá data pro získání každého *k-násobného* KV odhadu v testu

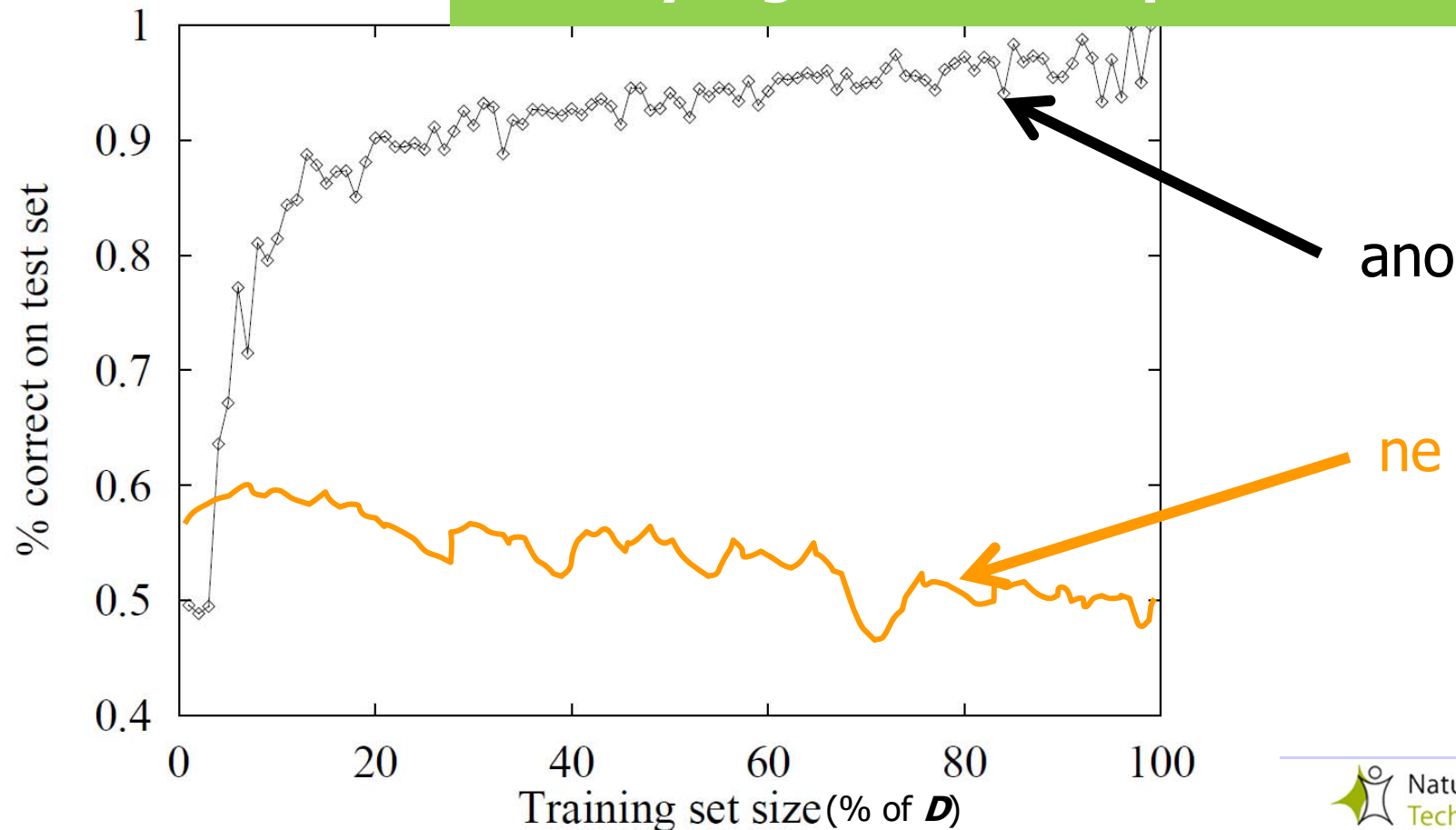
- ❖ **t-statistika je pro DM velmi užitečná!!!**

Křivka učení



❖ **Experiment:** Z dostupných klasifikovaných dat D vybereme podmnožinu pro trénování $Training \subset D$ a zbytek použijeme pro testování $Test = D - Training$

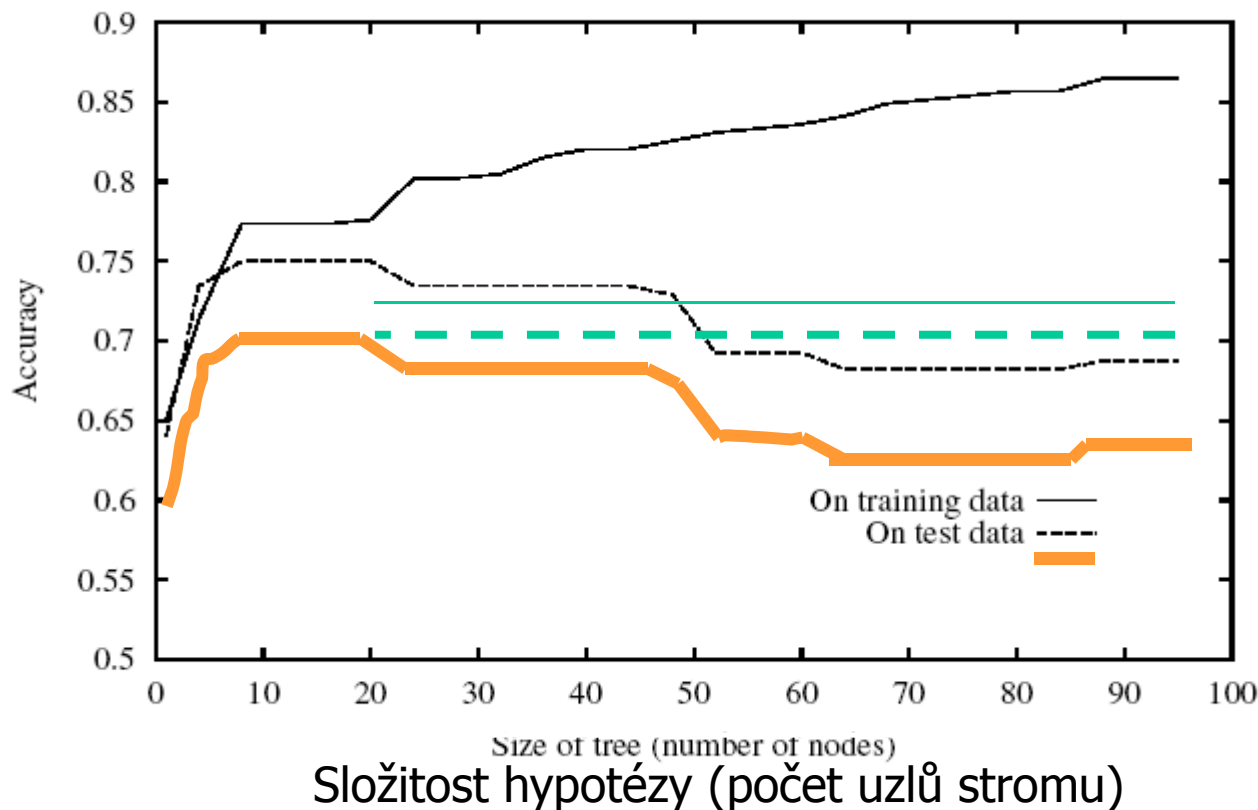
Je daný algoritmus ML pro data vhodný?



† Přeučení (*overfitting*).



❖ Necht' \mathbf{H} je prostor hypotéz. Hypotéza $h \in \mathbf{H}$ je **přeučená**, pokud existuje jiná hypotéza $h_1 \in \mathbf{H}$ taková, že na trénovacích datech má sice h přesnost vyšší než h_1 , avšak na celém prostoru instancí (nebo na testovacích datech) je h_1 přesnější (úspěšnější) než h . *Např. hypotéza s počtem uzlů 20 má přesnost 0,77 na trénovacích data (je horší než hypotézy s více uzly), ale na testovacích datech má přesnost 0,75 a je lepší než hypotézy s více uzly.*



Toto je často pozorovaná vlastnost zkonstruovaných stromů.

Jak složité stromy je rozumné zde konstruovat?

† Jak se vyhnout přeučení?

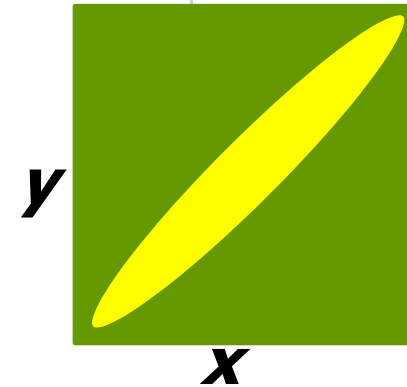


- ❖ **Jak zvolit správnou velikost stromu?** Existuje teoreticky odvozený vztah mezi velikostí trénovací množiny, počtem atributů a hloubkou stromu.
- ❖ **Jak strom správné velikosti získat?**
 1. **Včasné zastavení růstu stromu** (dřív než jsou vyčerpána všechna trénovací data)
 2. **Prořezávání hotového stromu** – ukazuje se jako zvlášť užitečné! Volba vhodného prořezání pomocí **validační množiny dat**. Používá se např. algoritmus **pro následné prořezávání podle „redukce chyby“**:
 - ◆ Vyberte uzel, odstraňte podstrom, v něm začínající a přiřadte většinovou klasifikaci.
 - ◆ Pokud se chyba na validačních datech zmenšila, proveďte uvedené proříznutí (ze všech možností vyberte tu s největším zlepšením).

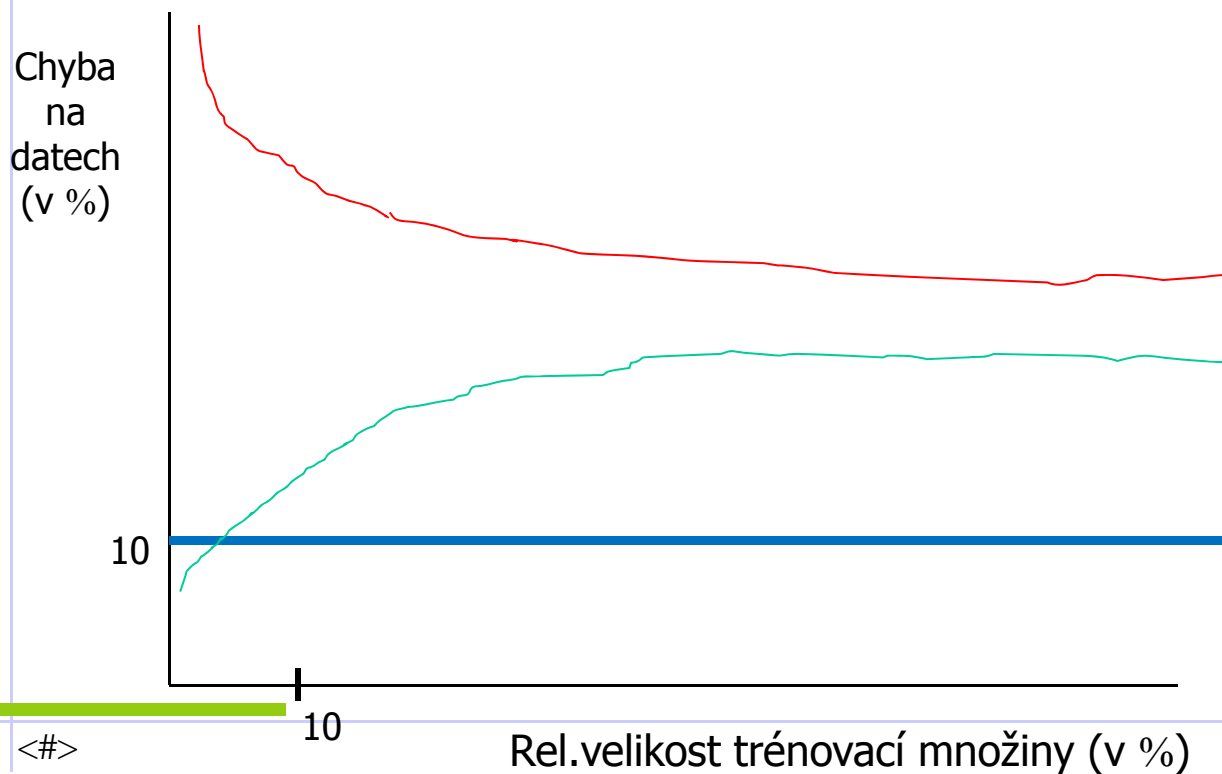
Co prozradí křivka učení?



Experiment: Uvažujme data na obr. popsaná v souř. x a y a **málo složité modely** (např. stromy s pevně daným malým počtem uzlů). Podobná situace nastává např. pokud model je reprezentován lineární hranicí).



Jak bude vypadat křivka učení?



Trénovací data

Testovací data

Požadovaná přesnost

Model má vysoký bias (je moc „jednoduchý“) – pomohlo by použití více či jiných atributů

Co prozradí křivka učení?

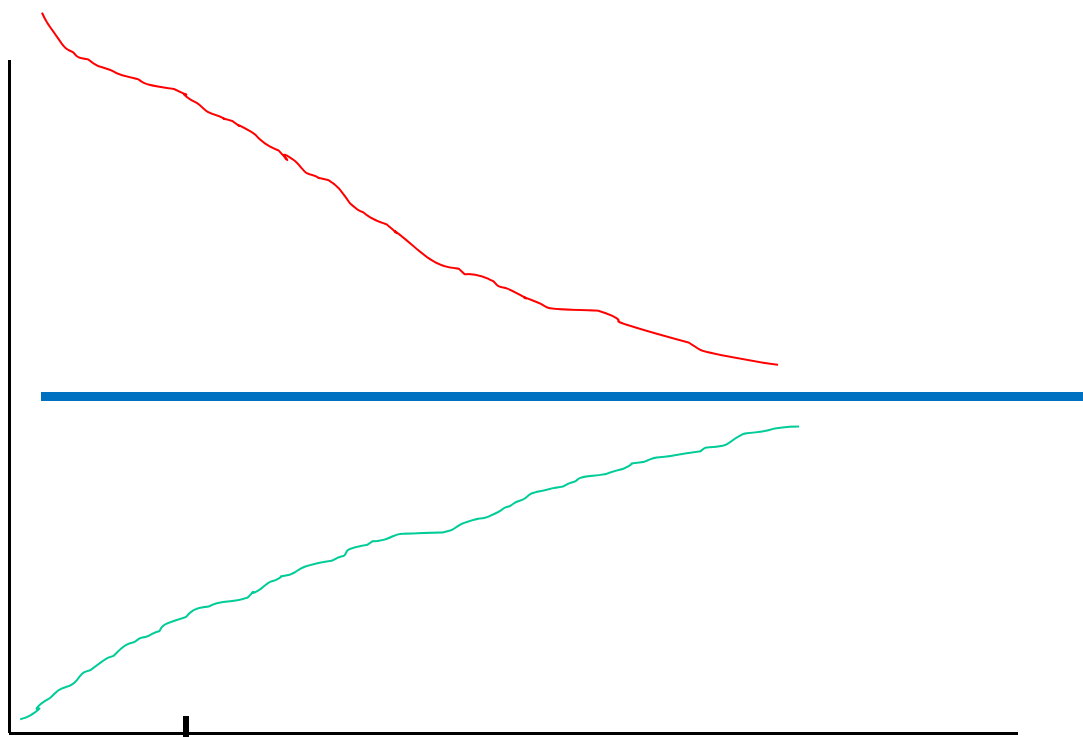


Pokud za stejných podmínek naopak **křivka učení na testovacích datech stále klesá**, svědčí to tom, že ke zlepšení výkonnosti by přispělo **získání nových dat** (tato situace nastává, např. pro složitý strom či pokud je hranice modelována polynomem vysokého stupně).



Chyba
na
datech
(v %)

20



Testovací data

Trénovací data

Požadovaná přesnost

Vybraný model má **vysokou varianci**, má potenciál být dobrým klasifikátorem, ale nutno doladit parametry.

<#>

10

Rel. velikost trénovací množiny (v %)

Shrnutí:



- ❖ Jsou-li k dispozici ROZSÁHLÁ data, rozdělí se na disjunktní **trénovací, testovací a validační podmnožiny**
- ❖ Nevyvážená data je nutné vhodně upravit
- ❖ Křížová validace je zvláště vhodná pro MALÉ objemy dat
- ❖ Je nutné dbát, aby testovací data NEBYLA použita pro ladění parametrů metody – k tomu slouží data validační
- ❖ **Především je třeba se vyhnout přeučení (*overfitting*) !**



Jak co nejlépe využít dostupná data ?

❖ Po ukončení evaluace je možné použít *VŠECHNA DATA* pro budování výsledného klasifikátoru

❖ **Obecně:**

- ◆ Čím větší je trénovací množina, tím je lepší klasifikátor (úměra však není lineární)
- ◆ Čím větší je testovací množina, tím kvalitnější je odhad průměrné chyby.

+ Další informace



- ❖ Petr Berka: ***Dobývání znalostí z databází***, Academia, Praha 2003
- ❖ Kap.11 *Strojové učení v dobývání znalostí* (F. Železný, J. Kléma, O. Štěpánková) v ***UI*** (4)
- ❖ V. Mařík, O. Štěpánková, J. Lažanský: ***Umělá inteligence*** (4), Academia, Praha 2003
- ❖ I.H. Witten, E. Frank, M.A. Hall: ***Data Mining: Practical Machine Learning Tools and Techniques*** (Third Edition), 3rd edition, Morgan Kaufmann 2011

Konfidenční intervaly



- ❖ **Význam:** Úspěšnost klasifikace p leží uvnitř nějakého specifického intervalu s určitou mírou důvěry.
- ❖ **Příklad 1:** $S=750$ správně klasifikovaných příkladů pro $N=1000$ instancí
 - ◆ Odhad relativní úspěšnosti klasifikace: 75%
 - ◆ Jak spolehlivý je tento odhad relativní úspěšnosti klasifikace p ?
 - ❖ Odpověď: S pravděpodobností 80% je $p \in [73.2, 76.7]$
- ❖ **Příklad 2:** $S=75$ a $N=100$
 - ◆ Odhad relativní úspěšnosti klasifikace: 75%
 - ◆ S pravděpodobností 80% je $p \in [69.1, 80.1]$