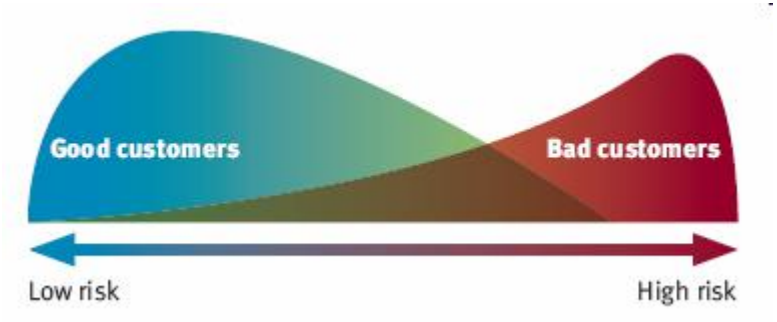
Credit scoring



Úvod

- Credit scoring is a set of predictive models and their underlying techniques that aid financial institutions in granting credits. These techniques decide who will get a credit, how much credit they should get, and what further strategies will enhance the profitability of the borrowers to the lenders. Credit scoring techniques assess the risk in lending to a particular consumer. While it does not identify “good” or “bad” (negative behavior is expected, e.g default) applications on an individual basis, it provides statistical odds, or probability, that an applicant with a given score turns to be “good” or “bad”. These probabilities or scores, along with other business considerations such as expected approval rates, profit, churn, and losses, are then used as basis for decision making.

Úvod

- While the history of credit stretches back 4000 years (the first recorded instance of credit comes from ancient Babylon -2000 BC), the history of credit scoring is only 50-70 years old. The first approach to solving the problem of identifying groups in a population was introduced in statistics by Fisher (1936). In 1941, Durand (1941) was the first to recognize that these techniques could be used to discriminate between good and bad loans. The arrival of credit cards in the late 1960s and the growth in computing power caused huge development and usage of the credit scoring techniques. The event that ensured the complete acceptance of credit scoring was the passage of the Equal Credit Opportunity Acts and its amendments in the U.S. in 1975 and 1976. These outlawed discrimination in the granting of credit unless the discrimination "was empirically derived and statistically valid."

Úvod

- In the 1980s, logistic regression, the main stalwart of today's scoring model builders, and linear programming, were introduced. More recently, artificial intelligence techniques, like expert systems and neural networks, have begun to be used. Furthermore, techniques like nearest-neighbour, splines, wavelet smoothing, kernel smoothing, Bayesian methods, regression and classification trees, support vector machines, association rules, cluster analysis, self-organizing maps and genetic algorithms are involved as well.

Default –definice cílové prom.

- Usually this definition is based on the client's number of days after the due date (days past due, DPD) and the amount past due. We need to set some tolerance level in case of the past due amount. It means what it is considered as the debt and what is not. It may be that the client gets into payment delay innocently (because of technical imperfections of the system). It does not make sense to regard as debt small amount (e.g. less than 3€) past due as well. Furthermore, it is necessary to determine the time horizon in which the previous two characteristics are traced. For example, as a good is marked client which:
- Has less than 60 DPD (with tolerance 3€) in 6 months form the first due date
- Has less than 90 DPD (with tolerance 1€) ever

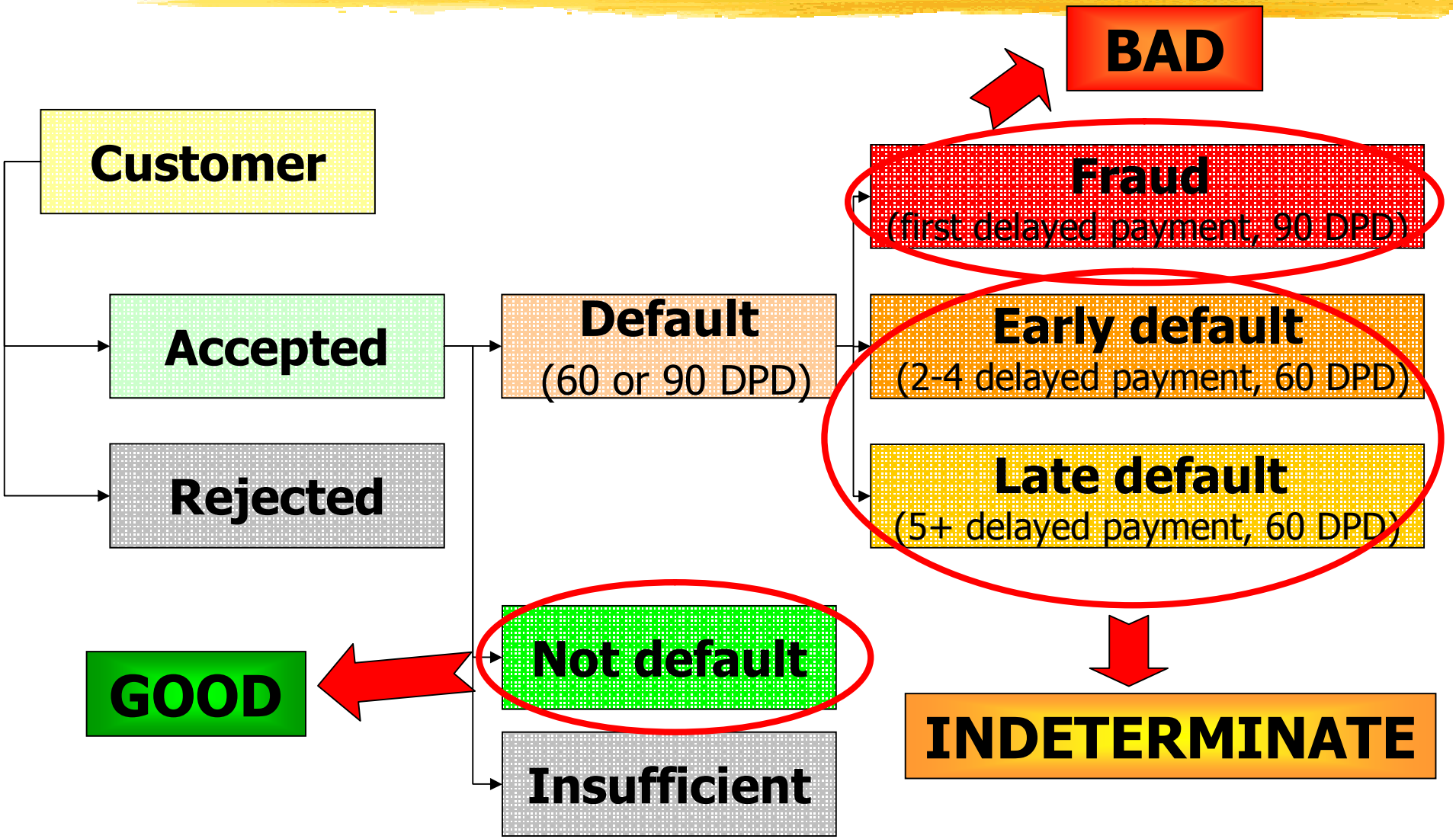
Default –definice cílové prom.

⌘Choice of these parameters depends greatly on the type of financial product (certainly will be different parameters for consumer loans for small amounts with original maturities around one year and for mortgages, which are typically connected to very large amounts and with maturities up to several tens of years) and on further usage of this definition (risk management, marketing, ...). Another practical issue of the definition of good client is the accumulation of several agreements. For example, it may be that the customer is overdue on more contracts, but with different days past due and with different amounts. In this case, all amounts past due connected to the client in one particular point in time are usually added together and it is taken the maximum value from days past due. This approach can be applied only in some cases and especially in a situation where there is a complete accounting data. The situation is considerably more complex in case of aggregated data.

Default –definice cílové prom.

- In connection with the definition of good client we can generally talk about the following types of clients: Good, Bad, Indeterminate, Insufficient, Excluded and Rejected. The first two types were discussed. The third type of client is on the border between good and bad clients, and directly affects their definition. If we are considering only DPD, clients with a high DPD (e.g. 90 +) are typically identified as bad, clients who are not delinquent (their DPD are equal to zero) are identified as good. As indeterminate are then considered delinquent customers who have not exceeded given threshold of DPD. The next type is typically case of the clients with the very short history, which makes impossible the correct definition of dependent variable (good / bad client). The excluded clients are for example clients who have exited the system, or they are so close to the point of no return that their classification is indisputable. They are also marked as "hard bad". The meaning of rejected client is obvious.

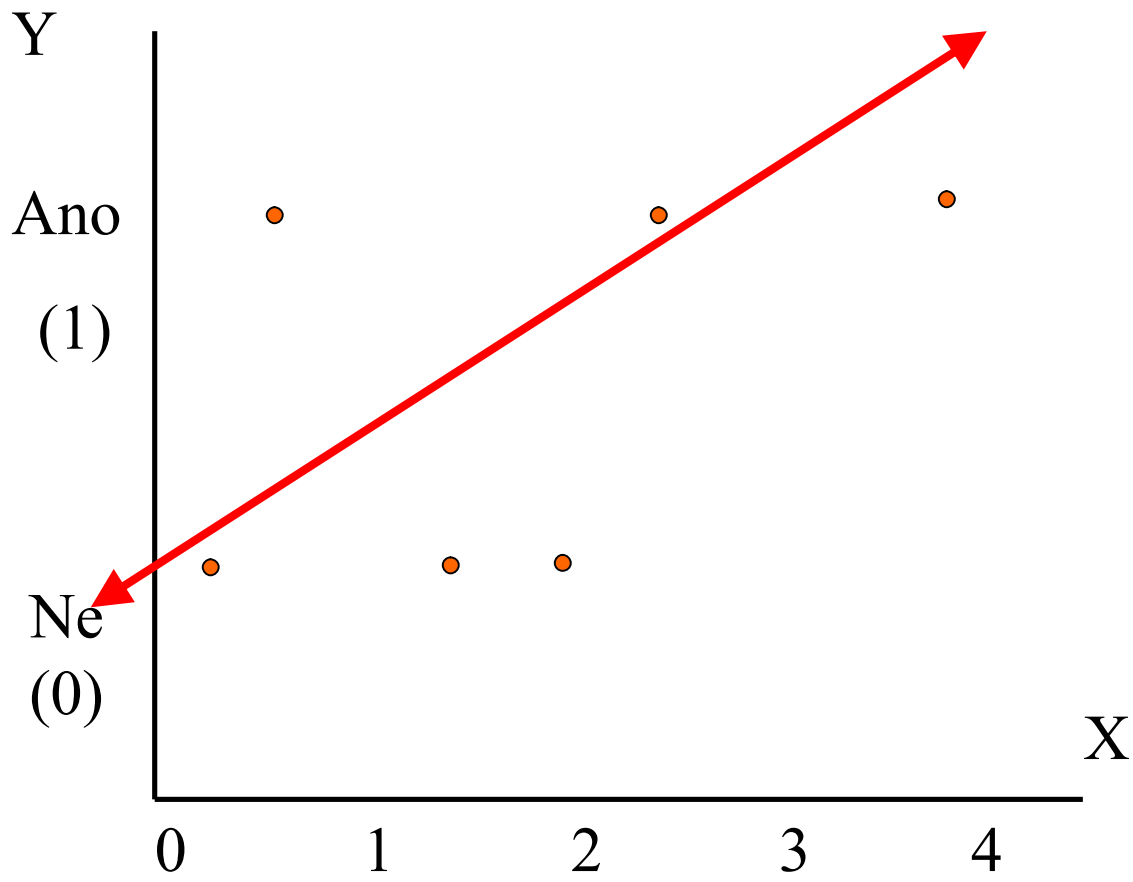
Good/bad client definition



Klasická regrese

- Nastane default?

St.	X	Y
1	2.6	1
2	1.4	0
3	.65	1
4	4.1	1
5	.25	0
6	1.9	0



Požadavky pro logistickou regresy

Je nutné specifikovat:

1) Výstupní proměnnou, která má pouze dvě kategorie (např. 1=úspěch; 0=neúspěch).
2) Způsob odhadu pravděpodobnosti P úspěchu.
3) Způsob propojení výstupní proměnné s vysvětlujícími proměnnými.
4) Způsob odhadu koeficientů a intervalů spolehlivosti.
5) Způsob posouzení validity modelu.

Měření pravděpodobnosti úspěchu

- Pravděpodobnost je měřena pomocí šance úspěchu (události).
- Jestliže P je pravděpodobnost události, pak (1-P) je pravděpodobnost, že nenastane.

- Šance události = P / 1-P

Logistická regrese

Simultánní efekt nezávislých (explanačních) proměnných na šanci

Odds = P/1-P = $e^{\alpha + \beta 1X1 + \beta 2X2 + \ldots + \beta pXp}$

Jestliže logaritmujeme obě strany

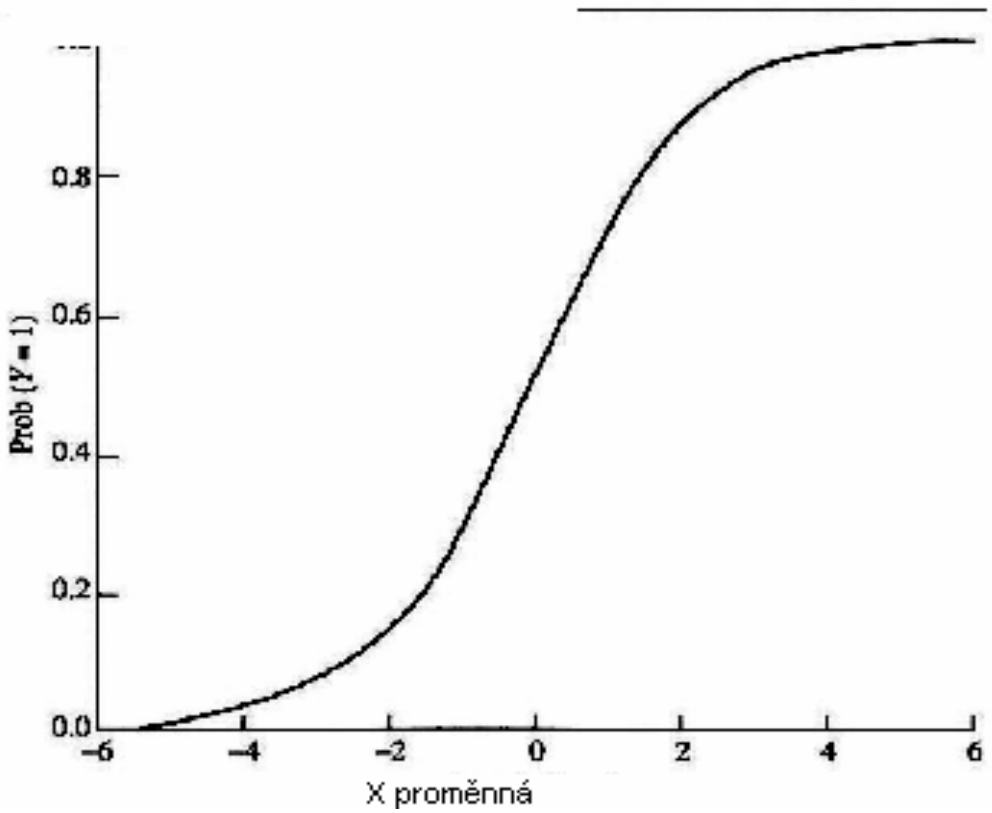
Log{P/1-P} = log $e^{\alpha+\beta 1X1+\beta 2X2+\ldots+\beta pXp}$

Logit P = $\alpha+\beta 1X1+\beta 2X2+..+\beta pXp$

Koeficienty β1, β2, βp jsou takové, že minimalizují funkci věrohodnosti.

Závislost P na X

Logistická křivka



$$P = \frac{e^{a+bX}}{1+e^{a+bX}}$$

$$P = \frac{1}{1+e^{-(a+bX)}}$$

Výhody logistické regrese

- Málo parametrů
- Snadné použití i interpretace
- Lze snadno začlenit i diskrétní prediktory
- Funguje dobře i na datech, která se poměrně značně liší od gaussovských směsí
- A především většinou dobře funguje, pokud věnujeme odpovídající pozornost přípravě dat
  - praktická zkušenost: ve čtyřech případech z pěti je logistická regrese na datech, která analyzuji, buď nejlepší nebo zhruba stejně dobrá jako jiné metody.

Interpretace, rozdíly proti OLS

- Regresní koeficienty b: kladné znamenají, že proměnná svým růstem zvyšuje šanci zařazení do skupiny kódované číslem 1, a naopak záporné indikují pokles této šance
- Často se používá $\exp(b_i)$: je to faktor, kterým se násobí šance $p/(1-p)$ při jednotkovém nárůstu x_i a neměnných ostatních x_k
  - Pozor na různá měřítka, v nichž x_i mohou být měřena;

- Místo F-testu celkové validity nyní máme chí-kvadrátový test pro totéž
- Místo t-testu signifikance proměnných v modelu jsou Waldovy statistiky; je to v podstatě totéž a čteme to stejně
- Místo R^2 jsou jen pseudo-R^2

Multinomiální logistická regrese

- Taktéž polytomická regrese
- Závisle proměnná má *M* kategorií, více než dvě. Např.: kterou stranu respondent volí?
- Základní idea:
  - Prohlásit jednu kategorii za referenční
  - Spočítat *M*–1 obyčejných logistických modelů pro každou ze zbylých kategorií oproti referenční
  - A predikovat tu kategorii, kde vyšla největší pravděpodobnost přes všechny modely
- SPSS má příkaz **NOMREG**

Budování modelu

- Forward
  - začíná se s prázdným modelem
  - postupné přidávání proměnných
- Backward
  - začíná se s plným modelem (všechny proměnné)
  - postupné odebírání proměnných
- Stepwise
  - začíná se s prázdným modelem
  - postupně se přidávají a odebírají proměnné
- Enter
  - je předepsán seznam proměnných v modelu

Princip rozhodovacích stromů

DIVIDE ET IMPERA !

- Rozděl a panuj: vhodně rozdělím zkoumané objekty do skupin...
- a v každé skupině opět postupuji stejně (rekurze)...
- dokud nedojdu k malým skupinkám, na něž stačí zcela jednoduchý model.

Historie metody

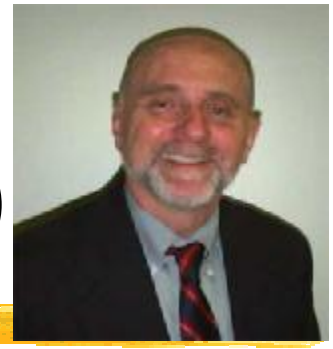
- DIVIDE ET IMPERA je staré římské přísloví, ale…
- jeho použití v analýze dat ve smyslu rozhodovacích stromů bylo navrženo až roku 1959 W. A. Belsonem
  - W. A. Belson: britský sociolog a metodolog, zabýval se především kriminalitou mládeže
- Původní citace: William A. Belson: Matching and Prediction on the Principle of Biological Classification, Applied Stat., VIII:65-75, 1959.
  - *Již předtím (minimálně od 30. let 20. stol.) se však statistici zabývali problémy kategorizace spojitých proměnných a dělením populací, ovšem v jiném kontextu (Yule, Fisher…)*

Historie metody (pokrač.)

- První počítačově implementovaný algoritmus se jmenoval AID – vznikl roku 1963
- Citace: James N. Morgan, John A. Sonquist: Problems in the Analysis of Survey Data, and a Proposal, Journal of the American Statistical Association, 58:415-435, 1963.
- AID byl založen na analýze rozptylu (sumy čtverců) – pomůcka pro přípravu ANOVA => základ statistického směru teorie rozh. stromů (CHAID, SEARCH aj.)

Portrét: James N. Morgan

Historie metody (pokrač.)

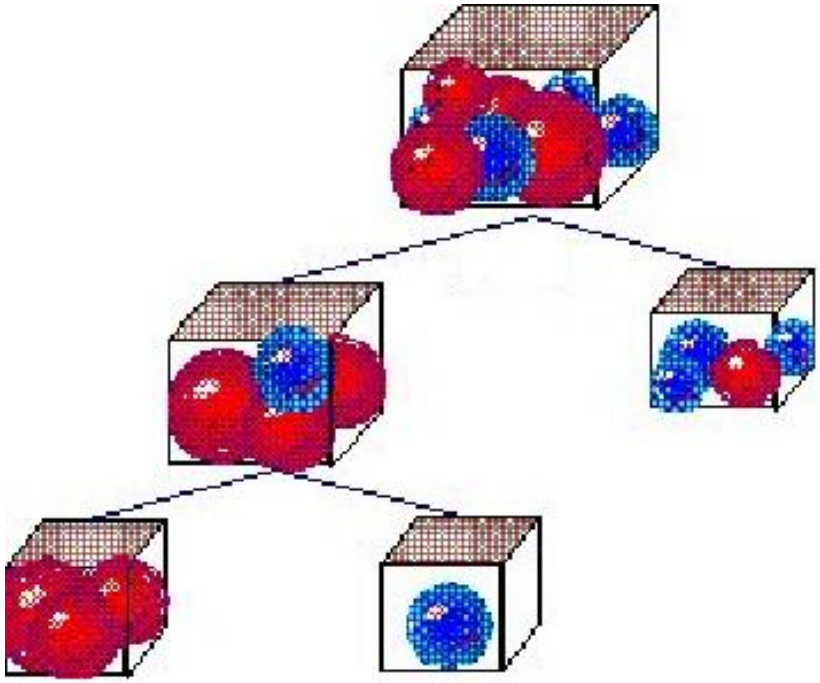


- Označení „rozhodovací strom“ (Decision Tree) je snad z r. 1966 (Experiments in Induction – E. B. Hunt, J. Marinová a P. J. Stone) => směr zakotvený v teorii umělé inteligence.
- Zde vyšli z teorie informace – rozdělení na podskupiny má přinést „informační zisk“, snížit entropii (implementováno např. v dnes užívaných algoritmech ID3, C4.5 a C5).
- Rozvoj aplikací a uplatnění i mimo oblast teoretické vědy přinesl nástup rychlých PC a rozvoj data miningu (cca polovina 90. let 20. století)

Portrét: Earl B. Hunt

Proč se hovoří o stromech?

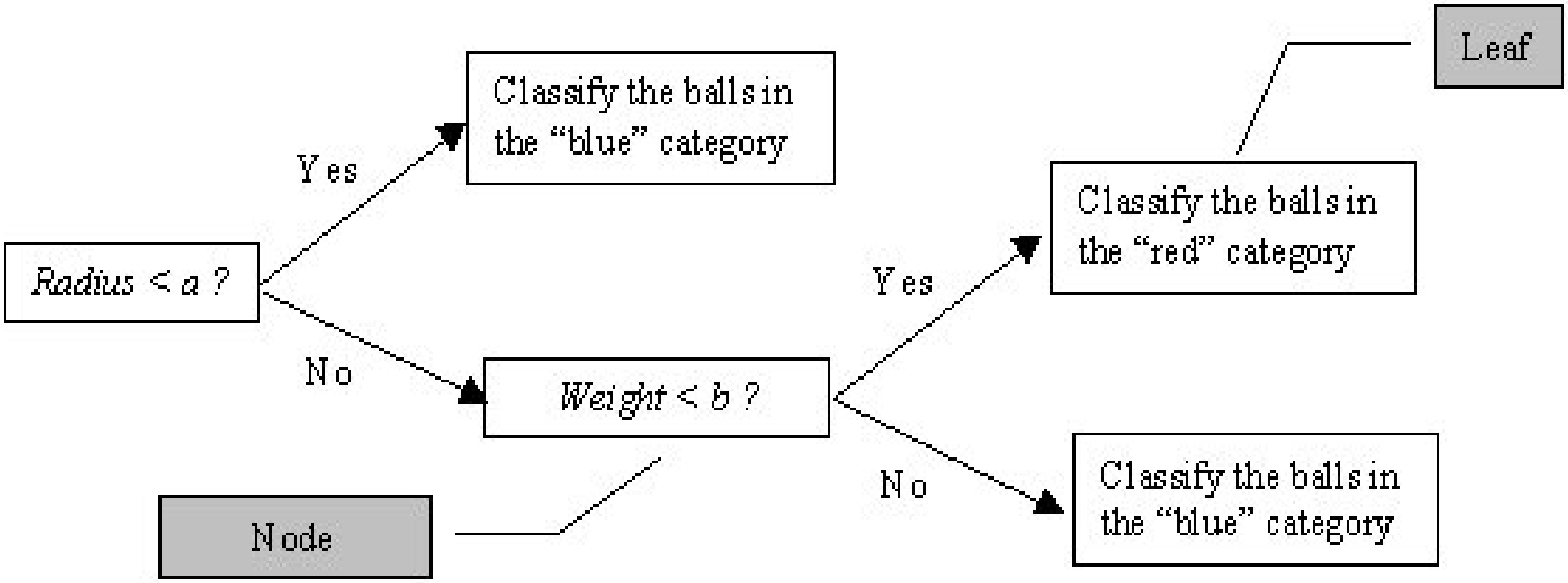
- Postupné dělení skupin zkoumaných případů lze znázornit stromovým schématem
- Kořen – větve – listy: terminologie teorie grafů



www.elseware.fr/storm

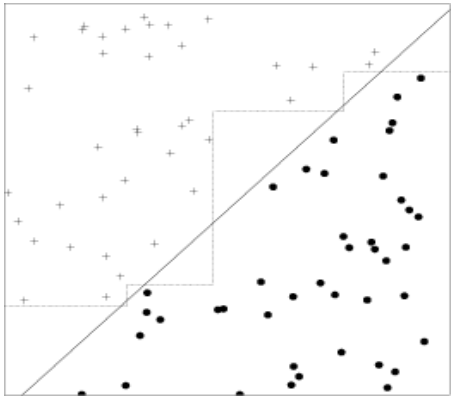
Proč „rozhodovací“?

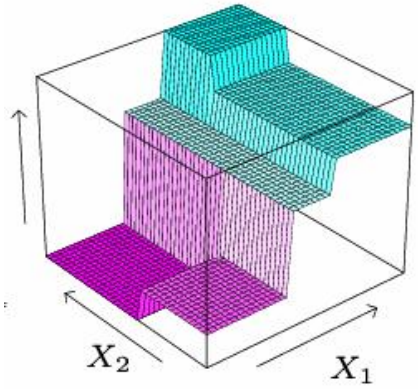
- Strom lze vyjádřit pomocí schémat *jestliže - pak*
- Lze snadno aplikovat do rozhodovacích procesů



Co je lepší: stromy, regrese...?

- Neexistuje obecné pravidlo, kdy volit jaký typ algoritmu – nejlepší bývá vyzkoušet jich několik
- Neexistují data vyloženě vhodná pro jeden typ (a vyloženě nevhodná pro jiný)





- Často však v praxi dosáhnou všechny metody podobnou přesnost => rozhodne interpretovatelnost, snadnost použití, stabilita výsledků, objem potřebných vstupních dat...

Co je lepší? (pokrač.)

- An Empirical Comparison of Decision Trees and Other Classification Methods (Tjen-Sien Lim, Wei-Yin Loh, Yu-Shan Shih, 1998) – srovnání 33 různých metod na 32 datových množinách
- Hlavní závěr: Průměrné chybovosti většiny klasifikátorů se od sebe statisticky významně neliší. Značné rozdíly jsou však ve výpočetním čase, který jednotlivé klasifikátory spotřebují.
- Nejlepší metody s přijatelným časem: polytomická logistická regrese a rozhodovací strom QUEST
  - ☺ *Nutno dodat, že obě programovali autoři článku*

Binární nebo obecné stromy?

Binární stromy

- Např. CART, C5, QUEST
- Z uzlu vždy 2 větve
- Rychlejší výpočet (méně možností)
- Je třeba mít více uzlů
- Zpravidla přesnější

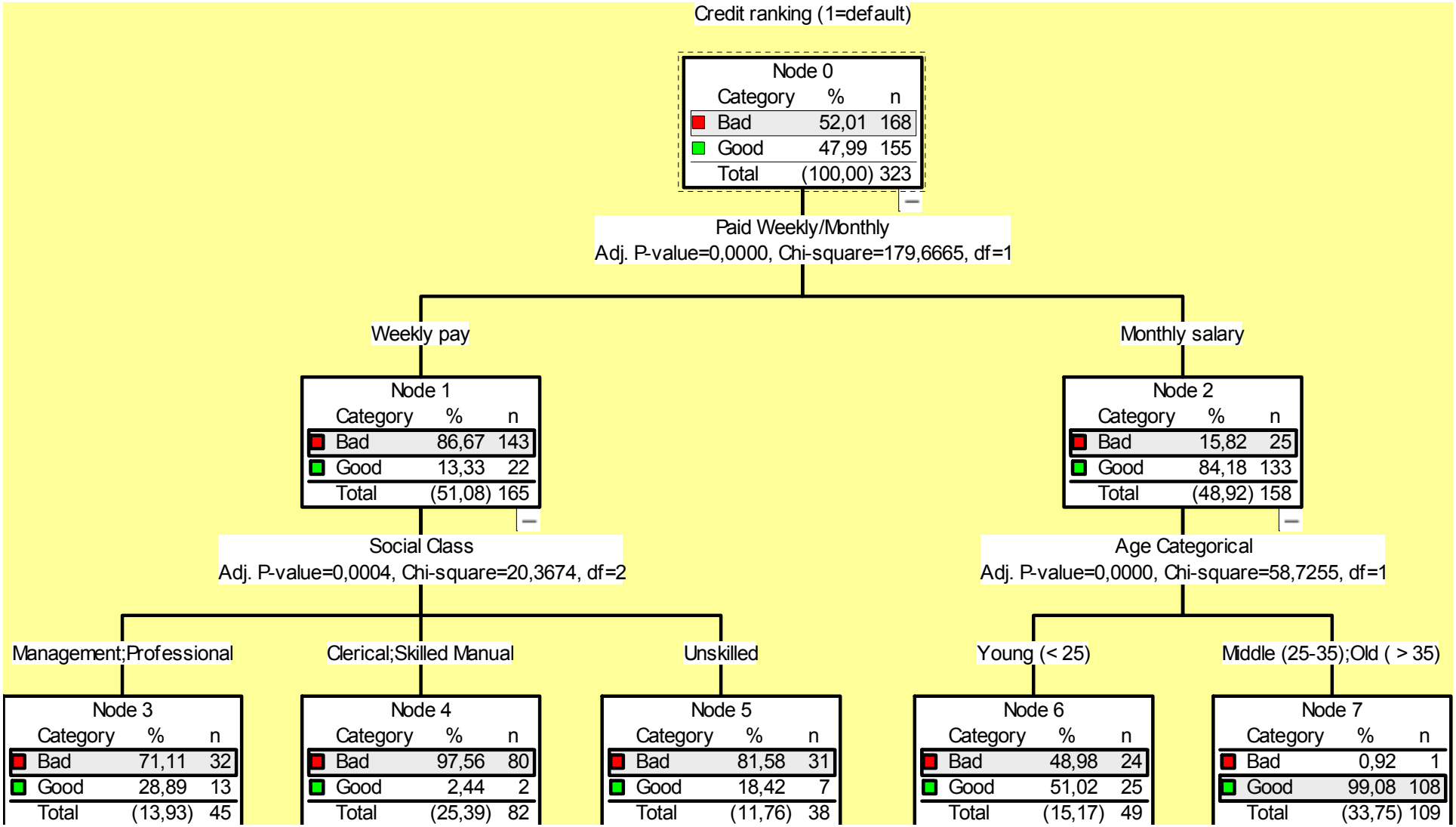
=> Data Mining, skóry

Obecné stromy

- Např. CHAID, Exhaustive CHAID
- Počet větví libovolný
- Interpretovatelnost člověkem je lepší
- Strom je menší
- Zpravidla logičtější

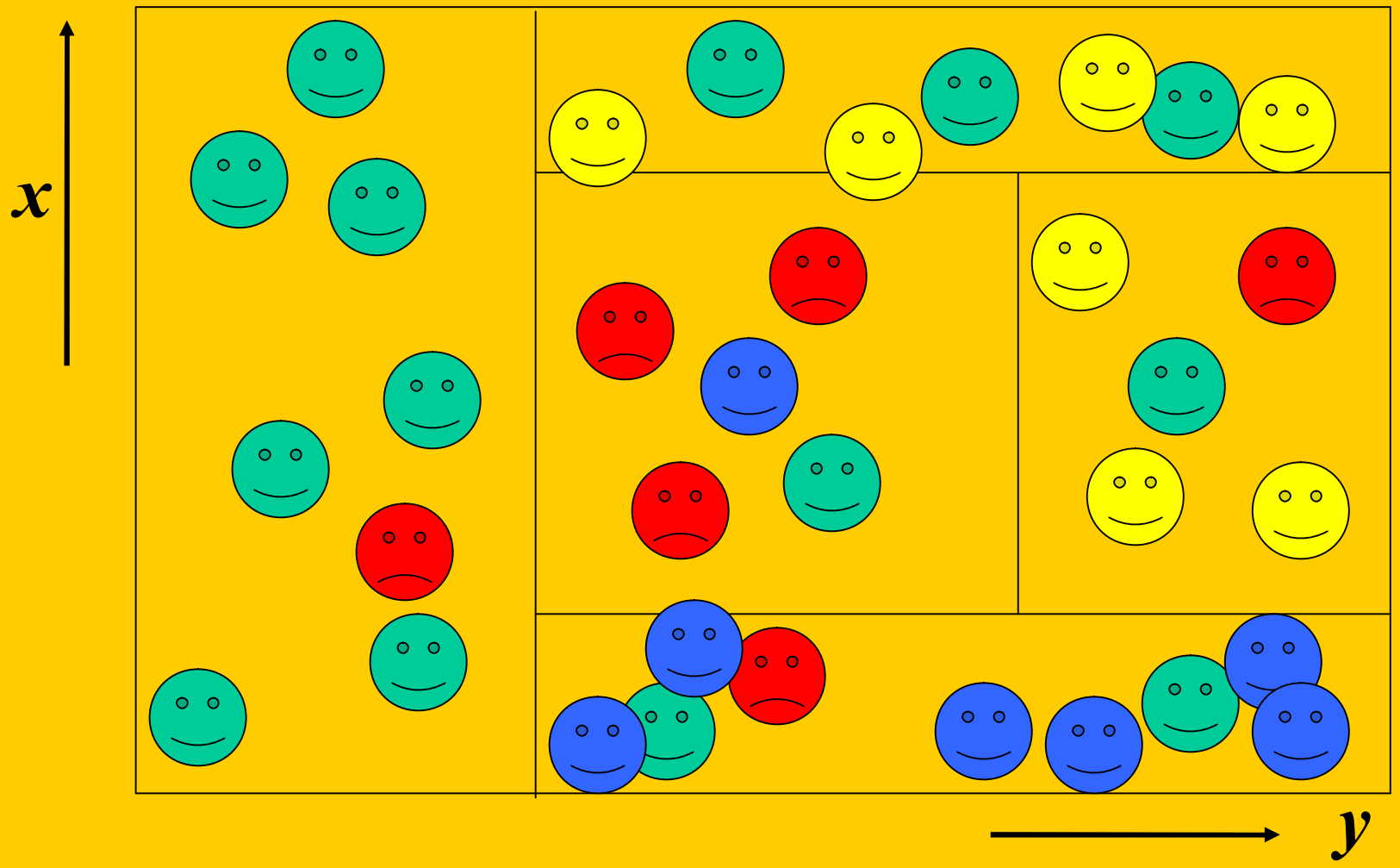
=> segmentace, mrktg.

Klasická prezentace: dendrogram



Alternativní prezentace: box chart

Vhodné pro malý počet použitých prediktorů (zde *x* a *y*)

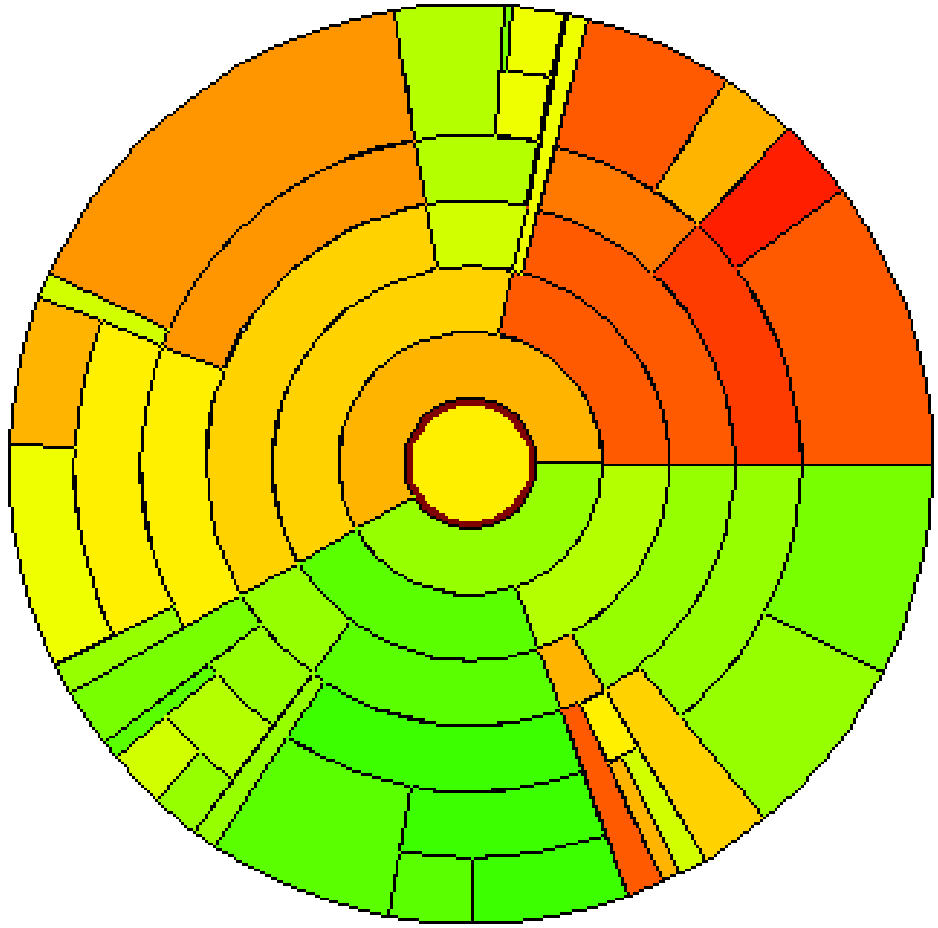


Alternativní prezentace: výseče

Snadno vidíme podíl jednotlivých větví na celém počtu případů.

Barva znázorňuje podíl hledané kategorie nebo míru homogenity uzlu.

Méně vhodné, jde-li nám o rozhodovací pravidla.



Alternativní prezentace: text

Jednoduché, ale hůře čitelné a málo výrazné

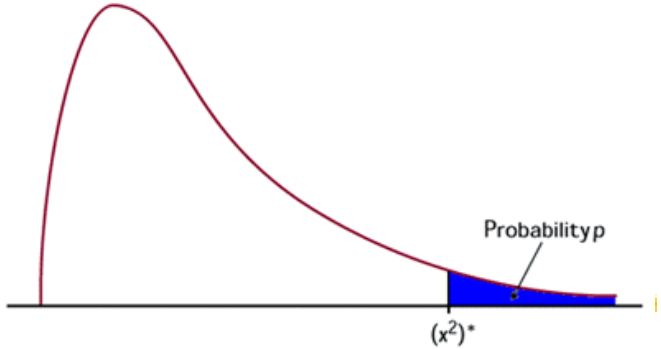
```
Group 1: All Cases N=5235, Mean(Y)=$13.02
  Group 2 EDUCATION 1989 H, NO COLLEGE DEGREE N=4186, Mean(Y)=$10.81
    Group 10 MARRIED MEN N=2535, Mean(Y)=$12.59
      Group 12 EDUCATION 12 GRADES OR LESS N=1420, Mean(Y)=$11.01*
      Group 13 EDUCATION 13+ NO COLL DEGREE N=1115, Mean(Y)=$14.45
        Group 14 AGE 18-34, N=703, Mean(Y)=$12.58*
        Group 15 AGE 35+ N=412 Mean(Y)=$16.24
          Group 18 CITY OF 25,000+NEARBY, N=287, Mean(Y)=$17.83*
          Group 19 NO CITY OF 25,000+ NEARBY N=125, Mean(Y)=$12.82*
    Group 11 SINGLE MAN OR WOMAN N=1651, Mean(Y)=$8.55*
  Group 3 COLLEGE GRAD OR MORE N=1049, Mean(Y)=$19.48
    Group 4 AGE 18-29 N=374, Mean(Y)=$14.17*
    Group 5 AGE 30+ N=675, Mean(Y)=$22.00
      Group 6 MARRIED MEN N=530, Mean(Y)=$24.34
        Group 8 AGE 30-39 N=329, Mean(Y)=$20.77
          Group 16 LIVING IN SAME STATE GREW UP N=174, Mean(Y)=$16.86*
          Group 17 LIVING IN DIFFERENT STATE N=155, Mean(Y)=$25.30*
        Group 9 AGE 40+ N=201, Mean(Y)=$28.04*
      Group 7 SINGLE MEN, WOMEN N=145, Mean(Y)=$16.3825*
```

dle www.isr.umich.edu/src/smp/search/search_paper.html

Algoritmus CHAID – úvod

- **CH**i-squared **A**utomatic **I**nteraction **D**etector
- Jeden z nejrozšířenějších rozhodovacích stromů v komerční oblasti (vedle QUEST a C4.5 / C5)
- Kass, Gordon V. (1980). An exploratory technique for investigating large quantities of categorical data. Applied Statistics, Vol. 29, pp. 119-127.
  - Založeno na autorově disertaci na University of Witwatersrand (Jihoafrická rep.)
  - Předchůdci: AID – Morgan a Sonquist, 1963; THAID – Morgan a Messenger, 1973

Připomenutí: Test nezávislosti χ^2



- Nezávislost testujeme na základě výrazu

$$\chi^2 = \sum_i \sum_j \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

- Jsou-li *x* a *y* nezávislé, má tento výraz Pearsonovo chí-kvadrát rozdělení s df = $(r-1)(s-1)$
- Test: plocha pod grafem „nad" pozorovanou hodnotou (~signifikance *p*) < α => zamítnu hypotézu nezávislosti *x* a y
- Současné testování více hypotéz
  => nutno adjustovat α (Bonferroni)

Algoritmus CHAID: idea

- Začíná se u celého souboru
- Postupné větvení / štěpení souboru (přípustné je rozdělení na libovolný počet větví vycházejících z jednoho uzlu)
- Algoritmus je rekurzivní – každý uzel se dělí podle stejného předpisu
- Zastaví se, pokud neexistuje statisticky signifikantní rozdělení => vzniká list
  - Obvykle je navíc podmínka minimálního počtu případů v uzlu a/nebo v listu, příp. maximální hloubky stromu
  - http://support.spss.com/ProductsExt/SPSS/Documentation/Statistics/algorithms/14.0/TREE-CHAID.pdf

CHAID: postup v uzlu

- Pro všechny prediktory
  - Vytvoř kontingenční tabulku target x prediktor (rozměr *k* x *l*)
  - Pro všechny dvojice hodnot prediktoru spočti chí-kvadrátový test podtabulky (*k* x 2)
  - „Podobné" (=ne signifikantně odlišné) dvojice postupně spojuj (počínaje nejnižšími hodnotami chí-kvardrátu) a přepočítávej výchozí kontingenční tabulku. Zastav se, když signifikance všech zbylých podtabulek je vyšší než stanovená hodnota.
  - Zapamatuj si spojené kategorie a signifikanci chí-kvadrátu výsledné tabulky s redukovanou dimenzionalitou
- Vyber prediktor, kde je tato signifikance nejnižší
- Pokud jsou splněny podmínky štěpení, rozděl případy v uzlu podle již „spojených" kategorií

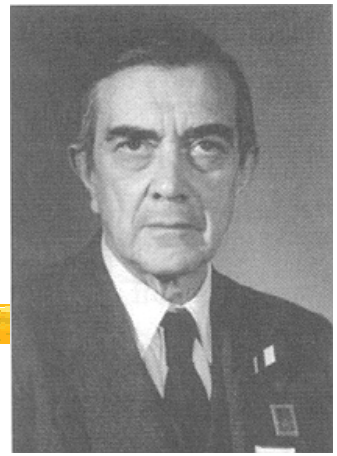
CHAID: zhodnocení

- Pokud je počet kategorií prediktoru *n*, tak je třeba provádět jen řádově n^2 testů
- Kdyby se testovala všechna možná rozdělení, rostl by počet testů exponenciálně s růstem *n*
- CHAID tím šetří výpočetní čas, ale zároveň není zaručeno, že najde optimální řešení uzlu (greedy search v uzlu)
- Ordinální znak: lze spojit jen sousední kategorie
- Spojitý znak: nutná je kategorizace
  - Zde existují lepší i horší implementace

Další algoritmy

- Existují desítky příbuzných algoritmů, často navzájem dost podobných
- Zde pouze naznačíme vlastnosti několika z nich (často používaných a/nebo zajímavých)
  - CART
  - ID3 a C5
  - QUEST
  - TreeNet

CART / C&RT



- **C**lassification **A**nd **R**egression **T**ree
- Algoritmus je založen na počítání míry diverzity („nečistoty“) uzlu: chci maximalizovat

$$div(matka) - (\mathbf{div(dcera\ A) + div(dcera\ B)})$$

konst.

s tím, že sčítance vážíme podílem případů v uzlech

- Giniho míra diverzity (inspirace z ekonomie, kde se podobně měří nerovnosti v distribuci majetku a příjmů)

$$div_{\text{Gini}} = 1 - \sum p_i^2$$

- p_i jsou u nás relativní četnosti v uzlech

Portrét: Corrado Gini

ID3, C4.5, C5 (See5)

- Místo Giniho míry užívají entropii

$$div_{entrop} = -\sum p_i \ln_2 p_i$$

= *střední počet bitů potřebných pro zakódování případu v daném uzlu*

- Binární stromy
- Zabudovaný algoritmus pro zjednodušení množiny odvozených pravidel – lepší interpretovatelnost
- Ross Quinlan: Induction of decision trees (1986); týž: C4.5: Programs for Machine Learning, (1993); týž: C5.0 Decision Tree Software (1999)
- http://www.rulequest.com/see5-info.html

Portrét: Ross Quinlan

QUEST

- **Q**uick, **U**nbiased and **E**fficient **S**tatistical **T**ree
- Loh, W.-Y. and Shih, Y.-S. (1997), Split selection methods for classification trees, Statistica Sinica, vol. 7, pp. 815-840
- Výběr štěpící proměnné na základě statistického testu nezávislosti prediktor x target => mírně suboptimální, ale rychlé, navíc výběr štěpící proměnné je nevychýlený
- Jen nominální target (=závisle proměnná)
- Binární strom, pruning
- Používá se imputace chybějících hodnot

Portrét: Wei-Yin Loh

TreeNet

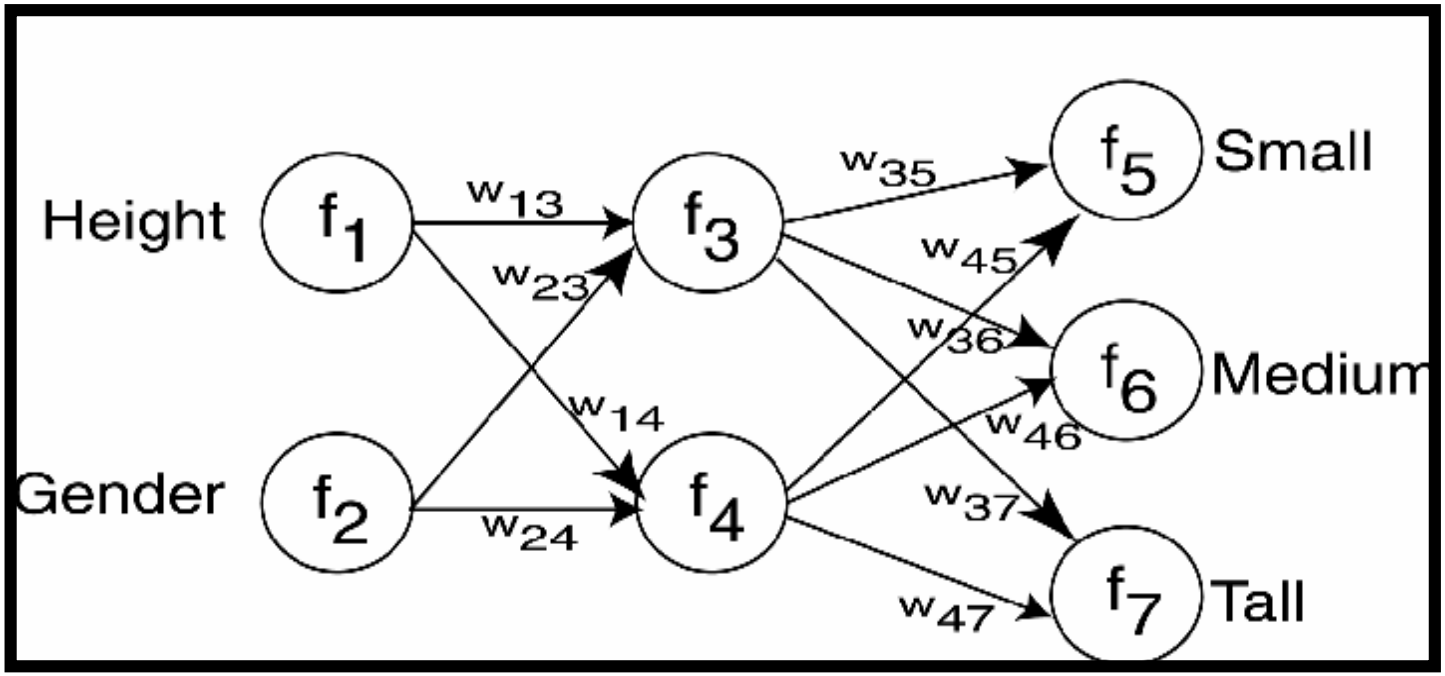
- Friedman, J. H. (1999): Greedy Function Approximation: A Gradient Boosting Machine, Technical report, Dept. of Statistics, Stanford Univ.
- Namísto jednoho velkého stromu „les" malých
- Výsledná predikce vzniká váženým součtem predikcí jednotlivých složek
- Analogie Taylorova rozvoje: rozvoj do stromů
- Špatně interpretovatelné (černá skříňka), ale robustní a přesné; nižší nároky na kvalitu a přípravu dat než neuronová síť nebo boosting běžných stromů
- Komerční, www.salford-systems.com

Portrét: Jerome H. Friedman

Neural Networks

- Based on observed functioning of human brain.
- ***(Artificial Neural Networks (ANN)***
- Our view of neural networks is very simplistic.
- We view a neural network (NN) from a graphical viewpoint.
- Alternatively, a NN may be viewed from the perspective of matrices.
- Used in pattern recognition, speech recognition, computer vision, and classification.

Neural Network Example



NN Advantages

- Learning
- Easy parallelization
- Solves many problems

NN Disadvantages

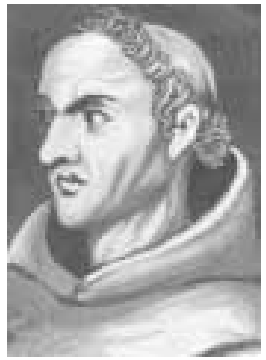
- Difficult to understand
- May suffer from overfitting
- Input values must be numeric.
- Verification difficult.

Mnohorozměrné postrašení

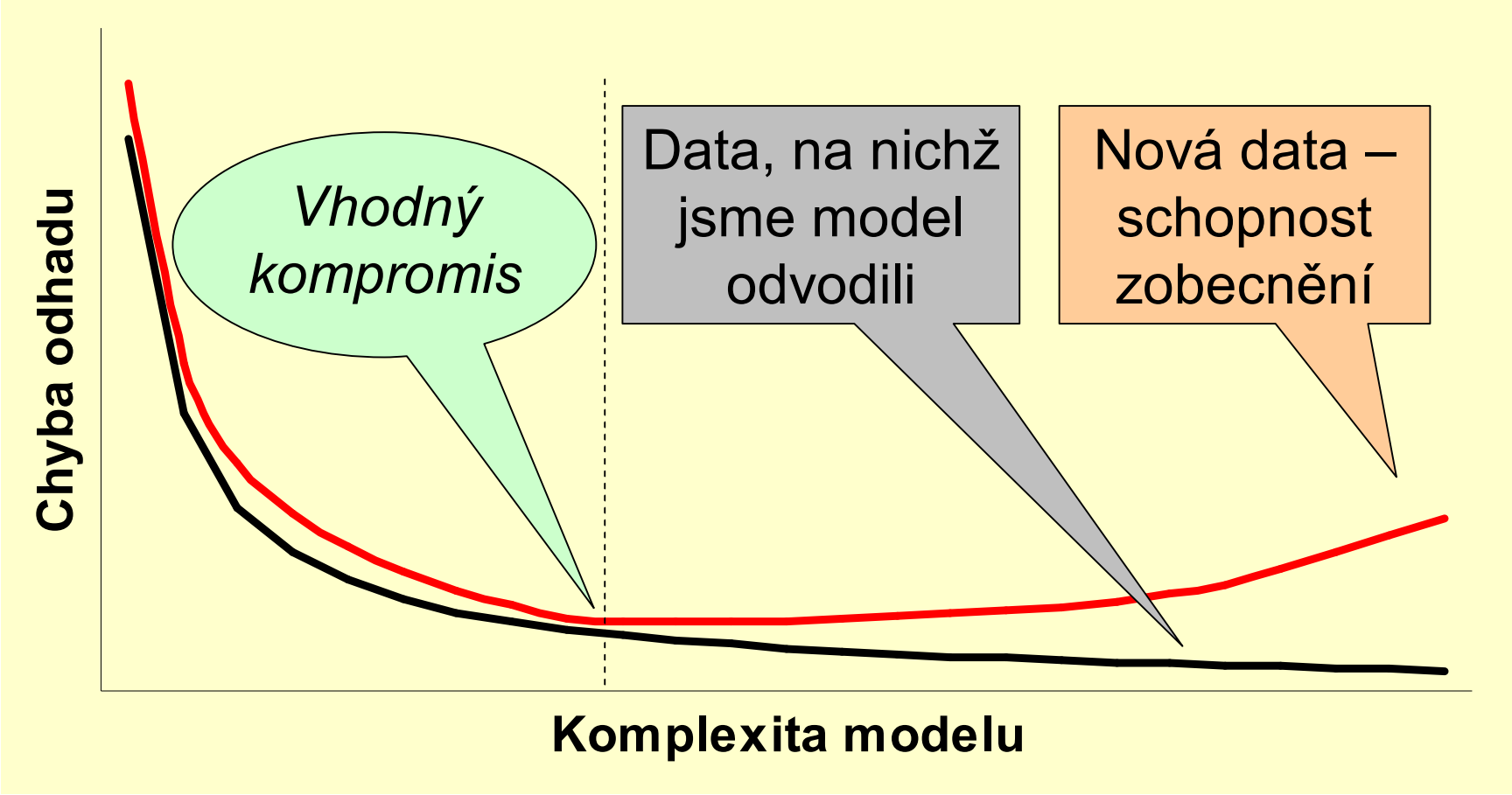
- Přímka délky 10 má „objem" 10. Reprezentativní vzorek „hustě" ji pokrývající může být řádově např. 10 bodů
- Čtverec se stranou 10 má „objem" 100 – při stejné hustotě chci cca 100 bodů
- Krychle s hranou 10 však vyžaduje 1000 bodů. Atd.
- PROKLETÍ DIMENZIONALITY: čím víc dimenzí, tím větší objem prostoru a tím 1) řidší jsou naše data, a tedy 2) tím hrubší a hůře zobecnitelné jsou naše modely
- Bohužel síla prokletí roste s počtem dimenzí **exponenciálně.**
  - Pro jednoduchý problém odhadu průměru std. normální distribuce s přesností ± 10% určil Silverman (1986) potřebný počet *n* datových bodů pro různé dimenze *d*: *d*=1 => *n*=4; *d*=2 => *n*=19; *d*=3 => *n*=67; *d*=6 => *n*=2790; *d*=10 => *n*=842 000
- Řešení: redukce počtu dimenzí vynecháním těch méně významných anebo projekce do prostoru s nižší dimenzí (např. PCA)

Přeurčení – overfitting

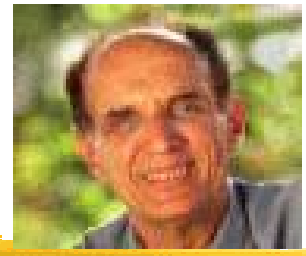
- Zejména (ale nejen) řídká data vedou k „přeu(r)čení“: model nastavuje příliš noho parametrů, tím lépe vystihne data, ale je hůře extrapolovatelný / zobecnitelný na data nová
- Praktické prostředky:
  - kontrolovat konfidenční intervaly odhadů
  - testovat model na čerstvých datech nepoužitých k nastavení parametrů
  - jackknife, bootstrap
- Occamova břitva (*pluralitas non est ponenda sine neccesitate*); zde: modelovat co nejjednodušeji



Komplexita modelu a chyba odhadu



Bootstrap



- Základní postup metody (*n* pozorování):
  - Generování *k* (obvykle 20 a více) nezávislých výběrů *n* prvků **s vracením** z původních dat
    - Tyto výběry z výběru *n* pozorování jsou vlastně přibližnou náhradou za nezávislé výběry z celé populace = idea metody
  - Na každém z *k* výběrů odhadneme model stejně jako na původním základním výběru
  - Populace *k* různých výsledků nám umožní odhadovat stabilitu odhadovaných parametrů
  - Zobecnitelnost lze odhadovat s použitím prvků, které v daném kole nebyly vybrány – OOB odhady (=out-of-bag), – tj. užít je jako testovací množiny
    - Těchto OOB prvků je v průměru 36,8 % z počtu pozorování

Zlepšování kvality predikce

- **Kombinace modelů** – „výbor", „committee" – několik nezávislých modelů „hlasuje" o výsledné predikci nebo se predikce různě průměrují apod. Zvláštní často užívané případy jsou:
  - **Nezávislé modely** – snažíme se o co nejpestřejší složení výboru; např. klasická logistická regrese + neuronová síť + rozhodovací strom + nejbližší soused + SVM
  - **Bagging** – bootstrapová agregace (L. Breiman, 1996) = modeluje se na bootstrapových výběrech, z modelů se tvoří výbor
  - **Boosting** (Y. Freund, R.E.Schapire, 1995). Tvoří se sled modelů na vážených datech, které pak fungují jako výbor. Výchozí váhy jsou postupně pozměňovány tak, aby „chybové" případy dostávaly stále větší váhu, a modely se je učily správně predikovat.

Šest dobrých rad Haira a kol.

- Usiluj o praktickou i statistickou významnost
- Velikost výběru podstatně ovlivní výsledek
- Znej svoje data (*a dobře si je připrav*)
- Hledej ty pravé vysvětlující proměnné – ani víc, ani méně
- Podívej se na své chyby (predikční chyby)
- Výsledky ověřuj (např. resampling apod.)

Hair, J.E. a kol. (1998): Multivariate Data Analysis, 5th ed., Prentice Hall Int., Upper Saddle River , NJ, p. 22-24