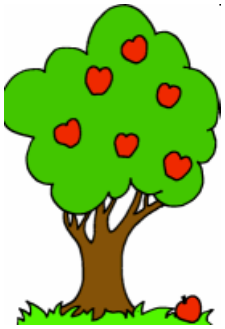
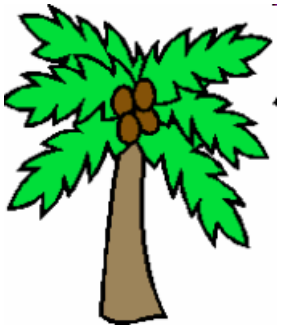
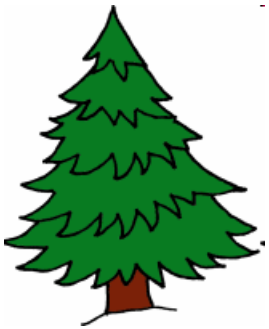
Pokročilé neparametrické metody

Klára Komprdová

evropský sociální fond v ČR
EVROPSKÁ UNIE
MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY
OP Vzdělávání pro konkurenceschopnost
UNIVERSITAS MASARYKIANA BRUNENSIS
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ





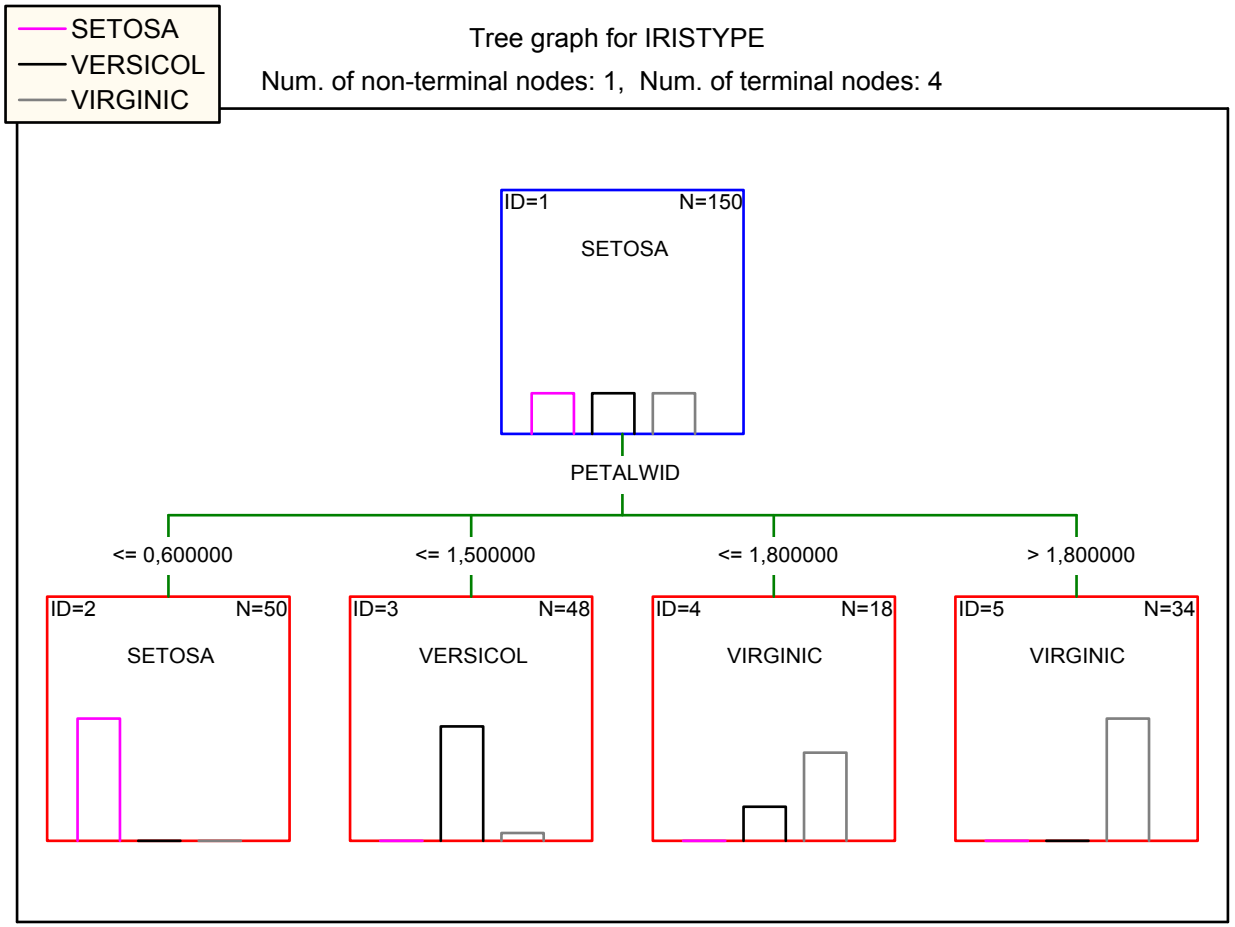


Další typy stromů
CHAID, PRIM, MARS

CHAID - Chi-squared Automatic Interaction Detector

- G.V.Kass (1980)
- Strom pro kategoriální proměnné→ převod spojitých proměnných na ordinální
- Je často využíván v komerčních sférách, především v marketingu a průzkumech veřejného mínění, má ale použití i v přírodovědných oborech.
- nebinárního typu
  - Po prvním dělení nemusí zbývat dostatek pozorování na vytvoření dalších „pater“ stromu →vhodnější pro větší datové soubory.
- Jako kriteriální statistika pro větvení se používá χ^2 –test.

Příklad - kosatce



χ^2 –test - opakování

- χ^2 –test je použit pro zjištění nezávislosti v kontingenční tabulce, která je tvořena kombinací kategorií závisle proměnné a prediktoru
- Jsou-li *Y* a *X* nezávislé, má testová statistika přibližně Pearsonovo χ^2 rozdělení s $u = (r-1)(s-1)$ stupni volnosti, kde *r* je počet řádků a *s* je počet sloupců v kontingenční tabulce.
- Nezávislost v kontingenční tabulce znamená, že se obě proměnné navzájem neovlivňují v hodnotách, které nabývají.
- Hypotéza nezávislosti jevů je zde nulovou hypotézou H_0.
- Pearsonův χ^2 –test je často označován jako test dobré shody.

Kontingenční tabulka

		kategorie prediktoru X				
kategorie Y	*j* / *i*	**1**	**2**	**…**	***s***	**Celkem**
	1	p_{11}	p_{12}	…	p_{1s}	R_1
	2	p_{21}	p_{22}	…	p_{2s}	R_2
	…	…	…	…	…	…
	r	p_{r1}	p_{r2}	…	p_{rs}	R_r
	Celkem	S_1	S_2	…	S_s	***n***

porovnáváme očekávané četnosti v kontingenční tabulce s jejich skutečnými četnostmi

χ^2 –test

$$\chi^2 = \sum_{i=1}^{r}\sum_{j=1}^{s}\frac{\left(p_{ij} - o_{ij}\right)^2}{o_{ij}}$$

$$o_{ij} = \frac{R_i S_j}{n}$$

- kde *i* a *j* je označení řádků (resp. sloupců) v kontingenční tabulce, p_{ij} je pozorovaná frekvence, o_{ij} očekávaná frekvence, *n* je celkový počet pozorování, R_i je počet pozorování v řádku *i*, S_j je počet pozorování ve sloupci *j*.

Příklad - Rozdělení semen dvou příbuzných rostlin podle barvy a tvaru

- Bylo zkoumáno celkem 160 semen dvou druhů příbuzných rostlin. Semena byla roztříděna do následujících kategorií: žluté/hladké; žluté/vrásčité; zelené/hladké; zelené/vrásčité

	žluté/hladké	žluté/vrásčité	zelené/hladké	zelené/vrásčité	Celkový součet
Druh1	10	25	10	15	60
Očekávaný počet					
Druh2	20	30	20	30	100
Očekávaný počet					
Celkový součet	30	55	30	45	160

Algoritmus růstu stromu CHAID

- **Krok1:** pro každý prediktor X_i: Vytvoř kontingenční tabulku kategorií závisle proměnné a prediktoru.

- **Krok 2:** mohou nastat tři případy:
  - Pokud je počet kategorií prediktoru > 2, utvoří se dvojice z kategorií prediktoru→ kategoriální x ordinální. Najde se taková dvojice, která si je co do hodnot závisle proměnné *Y* nejvíce podobná →dvojice, jejíž χ^2 - test má nejvyšší *p* hodnotu.
  - Pokud má prediktor 2 kategorie → algoritmus pokračuje krokem 5
  - Pokud má prediktor *X* pouze jednu kategorii → *p* hodnota je nastavena na 1

- **Krok 3:** Dvojice s nejvyšší *p* hodnotou, která není statisticky významná nebo větší než *alpha2*, se sloučí do jedné skupiny.
  - u ordinálního prediktoru se spojují pouze sousední kategorie
  - u kategoriálního jsou dvojice vytvořeny kombinací všech kategorií.

  Prediktor *X* je dále používán s novými již sloučenými kategoriemi
  - Pokud je i po sloučení počet kategorií > 2, algoritmus se vrátí do kroku 2. Pokud ne, algoritmus pokračuje krokem 4 nebo 5.

- Pozn: *alpha2,* 3 a 4 jsou hodnoty zadané uživatelem

Algoritmus růstu stromu CHAID

- **Krok 4:** Sloučené kategorie mohou být zpětně rozděleny. Jestliže se nově vytvořené skupiny kategorií skládají ze tří nebo více původních kategorií, najde se nejlepší binární rozdělení mezi sloučenými kategoriemi (s nejnižší *p* hodnotou). Pokud je *p* hodnota významná nebo větší než *alpha3*, dojde k rozdělení.

- **Krok 5:** Každá kategorie, která má velmi málo pozorování (minimum je definováno uživatelem), je spojena s nejpodobnější kategorií (opět určeno na základě největší *p* hodnoty)

  pozn: toto nastavení je volitelné a bývá dostupné jen v některých softwarech.

Výše popsaným postupem jsme získali optimální sloučení pro každý prediktor.

- **Krok 6:** V posledním kroku je spočítána adjustovaná *p* hodnota $\chi 2$ testu pro sloučené kategorie každého z prediktorů pomocí Bonferroniho korekce. Vybere se prediktor s nejmenší adjustovanou *p* hodnotou nebo hodnotou větší než *alpha4*. Tento prediktor s optimálně sloučenými kategoriemi je použit k rozdělení uzlu. Pokud významný prediktor nelze nalézt, uzel se již dále nedělí a je považován za terminální.

Algoritmus růstu stromu CHAID – ilustrační příklad

- Zajímá nás klasifikace potravních strategií druhů makrozoobentosu podle různých kategorií nadmořské výšky. Pro jednoduchost se budeme zabývat pouze jedním prediktorem.

Krok 1

Kontingenční tabulka -v buňkách by byly počty jednotlivých druhů

	N-nížinné	S - střední	P - podhorské	H - horské
sběrači				
spásači				
filtrátoři				
dravci				

Algoritmus růstu stromu CHAID – ilustrační příklad

- Pro každou podtabulku je spočítán Pearsonův χ^2 -test nezávislosti. Najdeme největší *p* hodnotu testu, pokud není signifikantní (menší než zvolené α), kategorie spojíme. Protože je nadmořská výška ordinální parametr, můžeme sloučit pouze vedlejší kategorie.

Krok 2 a 3

	N	S
sběrači		
spásači		
filtrátoři		
dravci		

χ_1^2 $p = 0{,}01$

	S	P
sběrači		
spásači		
filtrátoři		
dravci		

χ_1^2 $p = 0{,}05$

	P	H
sběrači		
spásači		
filtrátoři		
dravci		

χ_1^2 $p = 0{,}1$

	N	S	P + H
sběrači			
spásači			
filtrátoři			
dravci			

Algoritmus růstu stromu CHAID – ilustrační příklad

Test sloučených kategorií:

Opět spočítáme Pearsonův χ^2-test nezávislosti pro každou podtabulku, nyní již sloučených kategorií. Obě *p* hodnoty byly statisticky významné pro zvolené α=0,05 a k dalšímu sloučení již nedochází. Přecházíme rovnou do kroku 6, neboť jsme získali optimální sloučení prediktoru → krok 4 a 5 není v našem příkladu potřeba.

Krok 2 a 3

	N	S
sběrači		
spásači		
filtrátoři		
dravci		

χ_1^2 p = 0,01

	S	P + H
sběrači		
spásači		
filtrátoři		
dravci		

χ_1^2 p = 0,001

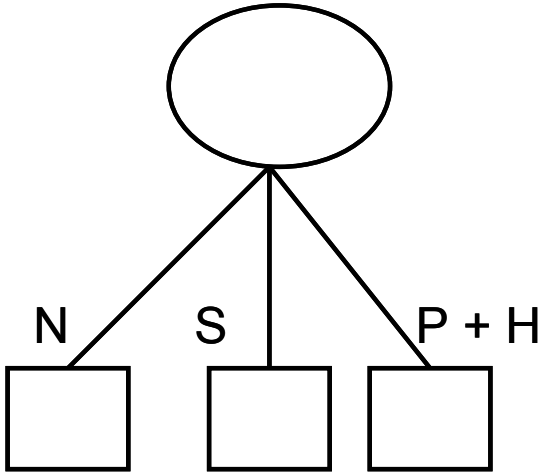
	N	S	P + H
sběrači			
spásači			
filtrátoři			
dravci			

pB*

Algoritmus růstu stromu CHAID – ilustrační příklad

- Finální rozdělení uzlu:
  - Za předpokladu, že je nadmořská výška prediktorem s nejnižší adjustovanou *p* hodnotou, původní uzel obsahující celý datový soubor bude rozdělen na tři dceřiné uzly, podle sloučených kategorií nadmořské výšky.

Krok 6



Bonferroniho korekce

- V algoritmu dochází k současnému testování více hypotéz →v našem příkladu bylo třeba učinit celkem čtyři testy pro možné sloučení kategorií.
- Při mnohonásobném testování však vzrůstá pravděpodobnost, že zamítneme nulovou hypotézu H_0, přestože platí.
- Počet prováděných testů u metody CHAID roste s počtem kategorií závisle proměnné a prediktorů.
- Použitím Bonferroniho korekce je možné zmírnit vliv mnohonásobného testování a získat porovnatelné *p* hodnoty pro jednotlivé prediktory s různým počtem kategorií.
- Výsledná *p* hodnota pro kontingenční tabulku kategorií závisle proměnné a optimálně sloučeného prediktoru je vynásobena *B* koeficientem, čímž získáme adjustovanou *p* hodnotu pro daný prediktor.

Bonferroniho korekce - Koeficient *B*

- ordinální proměnná → slučování sousedních kategorií
- kategoriální proměnná→ slučování všech možných kombinací

$$B_{ordinal} = \binom{s-1}{r-1}$$

$$B_{kategorial} = \sum_{i=0}^{r} (-1)^i \frac{(r-i)^s}{i!(r-i)!}$$

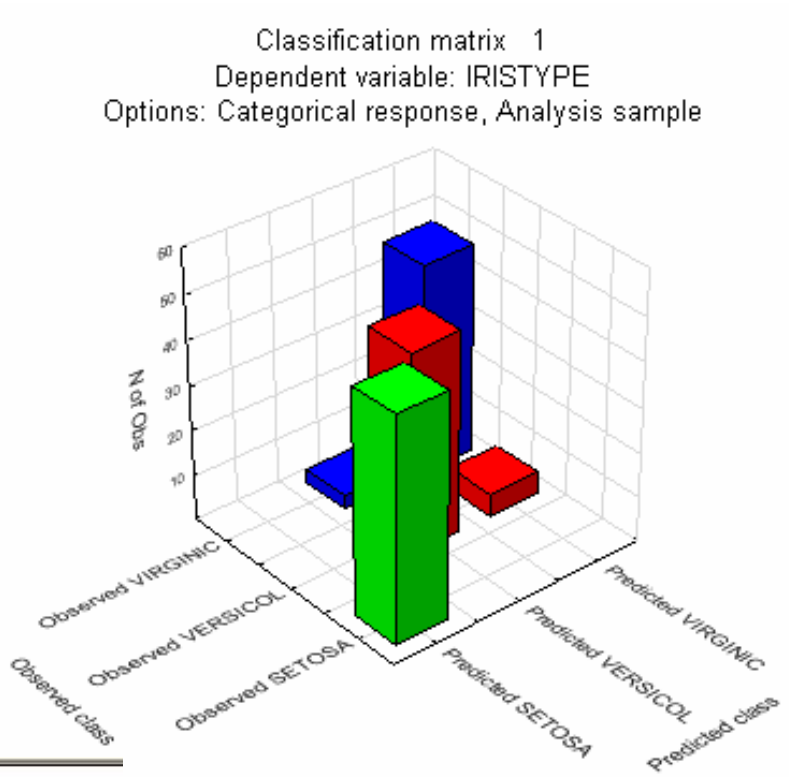
- kde *r* je počet řádků a *s* je počet sloupců kontingenční tabulky kategorií závisle proměnné a prediktoru.

Strom CHAID

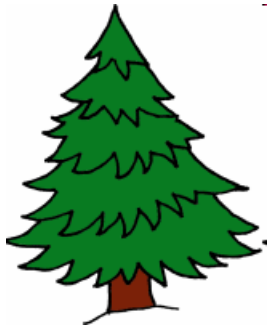
- Růst stromu se zastaví, pokud je dosaženo následujících pravidel:
  - není možné nalézt žádné významné rozdělení.
  - Všechna pozorování závisle proměnné v uzlu mají stejnou hodnotu nebo identickou hodnotu pro každý prediktor.
  - Pokud je dosaženo uživatelem definovaných nastavení, která se týkají:
    - parametrů velikosti stromu jako je nastavení počtu terminálních uzlů nebo větví;
    - počtu pozorování v uzlu, které je menší než minimum stanovené uživatelem nebo počtu pozorování, které by po rozdělení vedlo k dceřiným uzlům s menším počtem pozorování, než je definováno uživatelem.

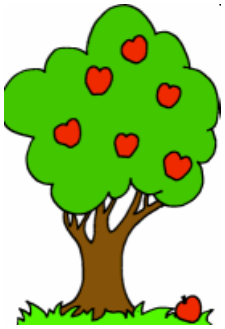
- Celkovou správnost stromu OA_{kateg} určujeme stejně jako v případě stromu CART. K odhadu obecné chyby $e(t)$ je možné opět použít k-testovacích souborů z krosvalidace.

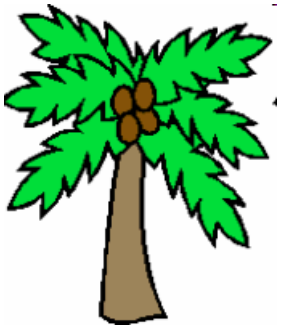
Příklad- kosatce



	Classification matrix 1 (Irisdat) Dependent variable: IRISTYPE Options: Categorical response, Analysis sample				
	Observed	Predicted SETOSA	Predicted VERSICOL	Predicted VIRGINIC	Row Total
Number	SETOSA	50			50
Column Percentage		100.00%	0.00%	0.00%	
Row Percentage		100.00%	0.00%	0.00%	
Total Percentage		33.33%	0.00%	0.00%	33.33%
Number	VERSICOL		45	5	50
Column Percentage		0.00%	93.75%	9.62%	
Row Percentage		0.00%	90.00%	10.00%	
Total Percentage		0.00%	30.00%	3.33%	33.33%
Number	VIRGINIC		3	47	50
Column Percentage		0.00%	6.25%	90.38%	
Row Percentage		0.00%	6.00%	94.00%	
Total Percentage		0.00%	2.00%	31.33%	33.33%
Count	All Groups	50	48	52	150
Total Percent		33.33%	32.00%	34.67%	





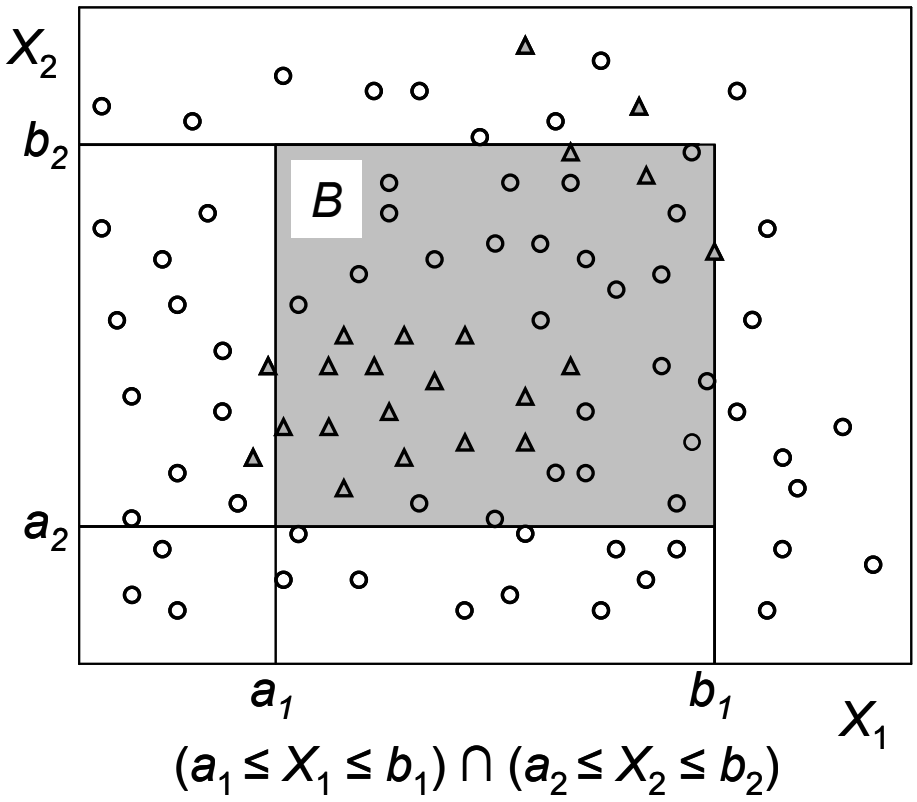


PRIM - Patient Rule Induction Method

PRIM - Patient Rule Induction Method

- **PRIM** (Friedman & Fisher, 1999) - metoda primárně určena pro regresi.
- PRIM podobně jako ostatní rozhodovací stromy rozděluje pozorování závisle proměnné Y pomocí hodnot prediktorů do uzlů $t_1,..,t_N$, → označovaných jako okna $B_1,\ldots, B_K$
- Graficky můžeme okna znázornit jako jednotlivé regiony v prostoru prediktorů $X_1,\ldots, X_M$.
- V případě metody PRIM se však vyhledávají takové regiony, ve kterých je průměr hodnot závisle proměnné Y nejvyšší (nebo nejnižší).
- Výsledkem je sada jednoduchých pravidel, která definují jednotlivá okna a rozdělují pozorování závisle proměnné

PRIM



Mějme 100 pozorování. Závisle proměnná Y označuje presenci $y_i = 1$ (trojúhelníky) nebo absenci $y_i = 0$ (kolečka) určitého druhu rostliny. Pro jednoduchost uvažujme pouze dva spojité prediktory: teplotu X_1 a srážky X_2. Rostlina bude přítomna s větší pravděpodobností v podmínkách daných rozsahem prediktorů $(a_1 \leq X_1 \leq b_1) \cap (a_2 \leq X_2 \leq b_2)$, které jsou zde znázorněny pomocí okna B.

PRIM - algoritmus

- 1. Soubor se rozdělí na testovací a trénovací (v poměru zadaném uživatelem). Seřadí se hodnoty prediktorů od nejmenší po největší. Okno obsahuje celý datový soubor (trénovací)

- 2. Okno se zmenšuje podél jedné hrany o malé množství pozorování (často α=0.1 nebo α=0.05) – tak aby výsledný průměr ve zmenšeném okně byl co největší (nejmenší)

Krok 1 a 2 se opakuje dokud okno neobsahuje předem stanovené minimum pozorování (např. 10)

- 3. dochází k reverznímu procesu-okno je zpětně rozšiřováno do všech směrů, ale jen pokud se zvýší průměr v okně

Z kroku 1-3 se získá sekvence oken o různém počtu pozorování

- 4. použije se krosvalidace k vybrání optimálního okna B_1- testovací soubor!
- 5. Odstraní se vzorky z okna B_1 -pozorování která jsou odstraněna z okna mají nejvyšší (nejnižší) hodnoty prediktoru X_j

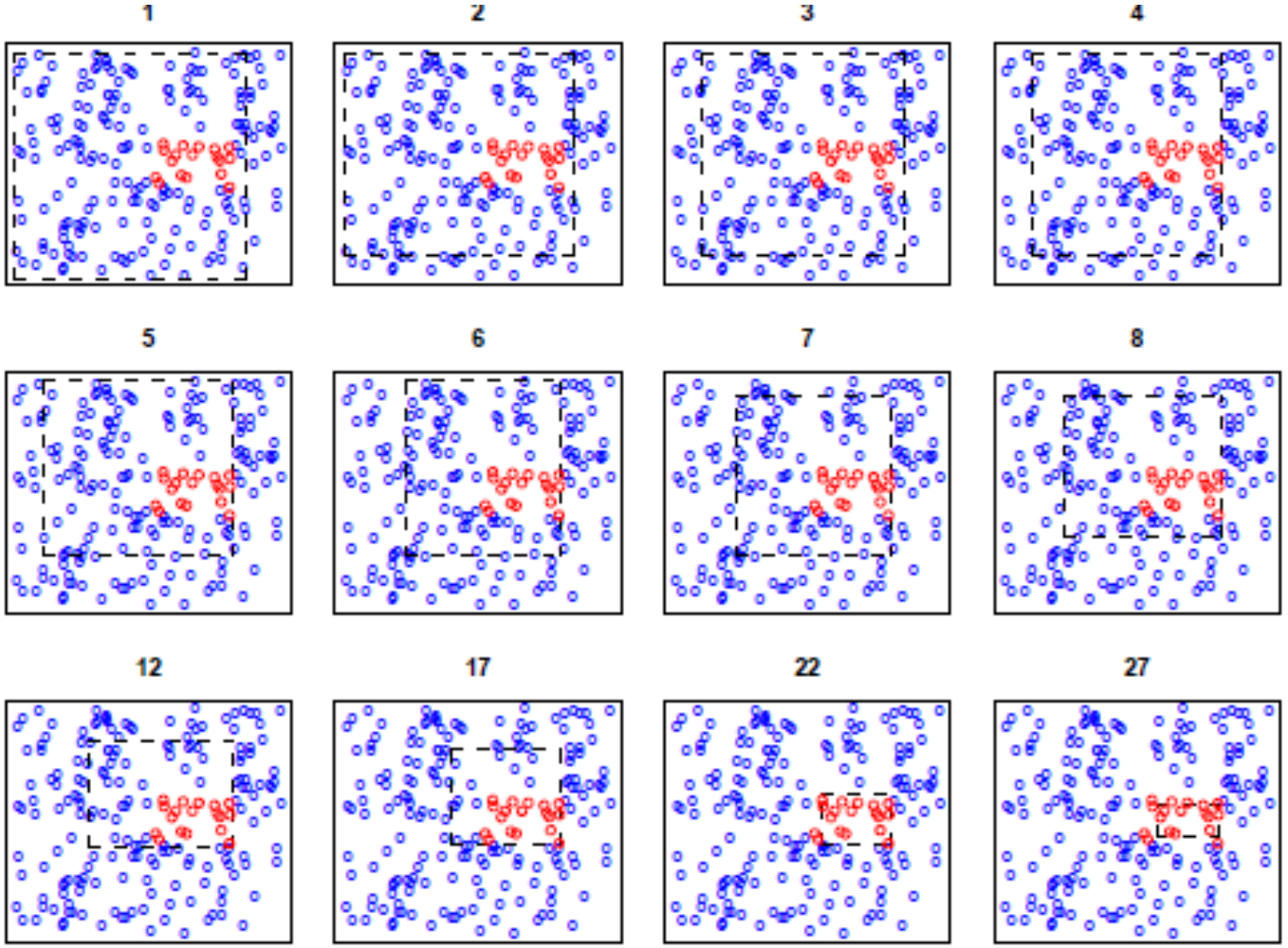
Krok 2-5 se opakuje, dokud není dosaženo konečného počtu oken $B_1, B_2, \ldots B_K$

- Okna jsou dána rozhodovacími pravidly
- Stejně jako v CART lze použít kategoriální prediktor

PRIM - algoritmus

- 200 bodů, rovnoměrně rozdělených do jednotkového čtverce
- Závisle proměnná Y má hodnotu 1 (červená barva) pokud je $0.5 < X_1 < 0.8$ a $0.4 < X_2 < 0.6$
- Závisle proměnná Y má hodnotu 0, modrá barva
- Proporce bodů o které se okno posune $\alpha = 0.1$

PRIM - algoritmus



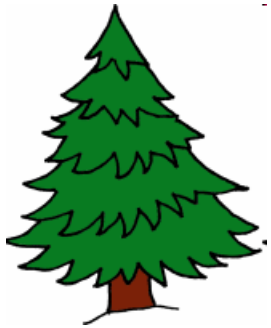
(Hastie et. al, 2009)

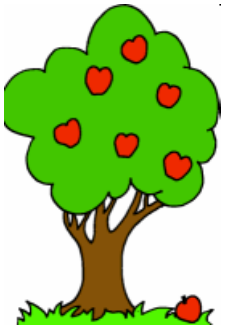
Algoritmus je hierarchický a používáme krosvalidaci

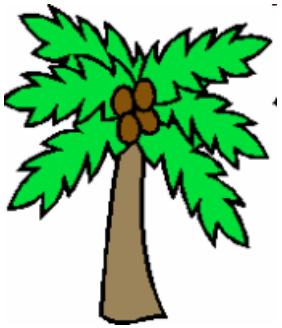
PRIM

- Stejně jako v CART lze použít kategoriální prediktor
- Oproti CART je výhodou, že se probere větší škála pravidel a můžeme najít optimální řešení
- Nevýhoda- není k dispozici stromová struktura → okna jsou dána rozhodovacími pravidly

- PRIM je velmi vhodný pro případy, kdy nás zajímá nalezení skupin v datech s nejvyšší nebo nejnižší hodnotou závisle proměnné – např. při různých ochranných opatření, kdy výsledky mohou sloužit ke stanovení vhodné velikosti území podle pravděpodobnosti výskytu druhu nebo ke zjištění klimatických podmínek, při kterých dochází k největšímu znečištění ovzduší







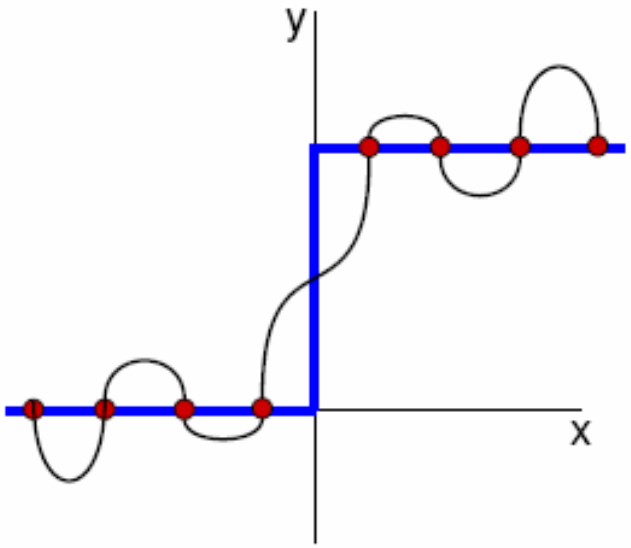
MARS - Multivariate Adaptive Regression Splines

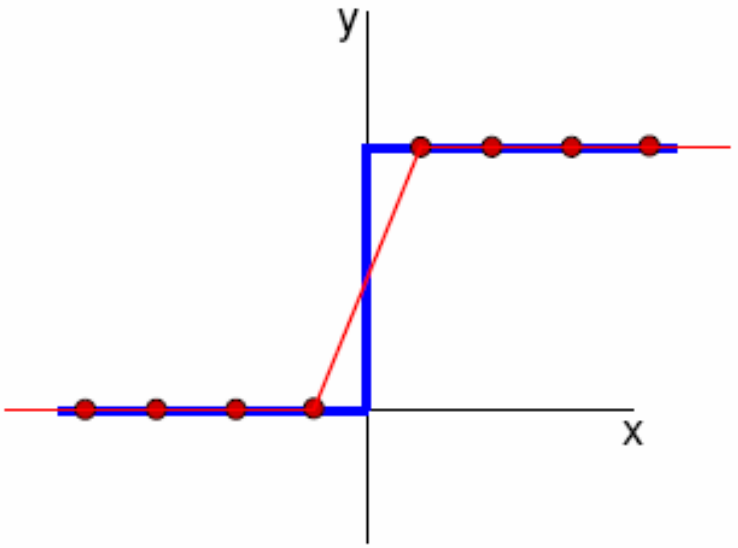
MARS - Multivariate Adaptive Regression Splines

- Friedman (1991)
- technika pro regresní problémy
- na rozhraní mezi stromovou technikou a parametrickou regresí→zobecnění postupné (*stepwise*) lineární regrese
- odstraňuje určité nedostatky binárních regresních stromů, především nespojitosti odhadnutých hodnot závisle proměnné
- prediktory mohou být spojité i kategoriální
- výsledkem metody je regresní rovnice → chybí stromová struktura a interpretace výsledků při velkém počtu proměnných může být obtížnější
- k rozdělení pozorování závisle proměnné se nepoužívá konstanta, ale lineární aproximace

Spline- interpolace

Interpolace - *n* body mohu proložit polynom *(n -1) řádu*
- větší stupeň polynomu - oscilace mezi body
daná množina bodů se aproximuje po částech = ***spline křivky***





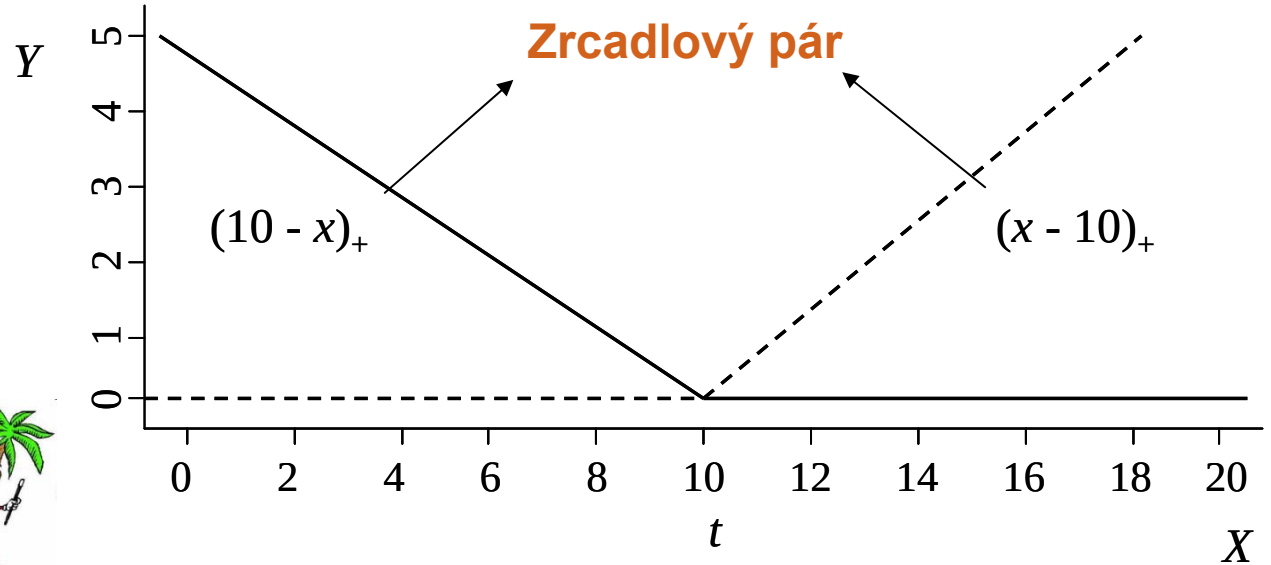
MARS

- lineární spliny - po částech lineárních funkce $(x - t)_+$ a $(t - x)_+$, kde + je kladná část

$$(x - t)_+ = \begin{cases} (x - t), & \text{pokud } x > t \\ 0, & \text{jinak} \end{cases} \qquad (t - x)_+ = \begin{cases} (t - x), & \text{pokud } x < t \\ 0, & \text{jinak} \end{cases}$$

- se svým středem (uzel) v každé hodnotě x_{ij}, pro každý prediktor X_j.

Příklad funkce $(x - 10)_+$ a $(10 - x)_+$ Alternativní zápis: $max(0, x - t)$ a $max(0, t - x)$

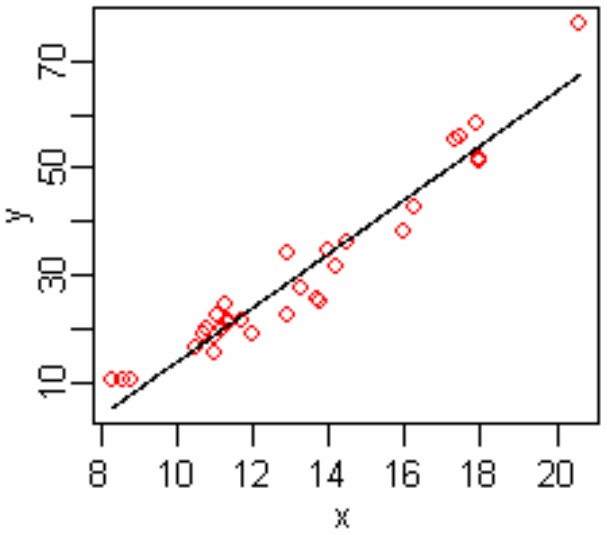


$t = 10$

MARS - příklad

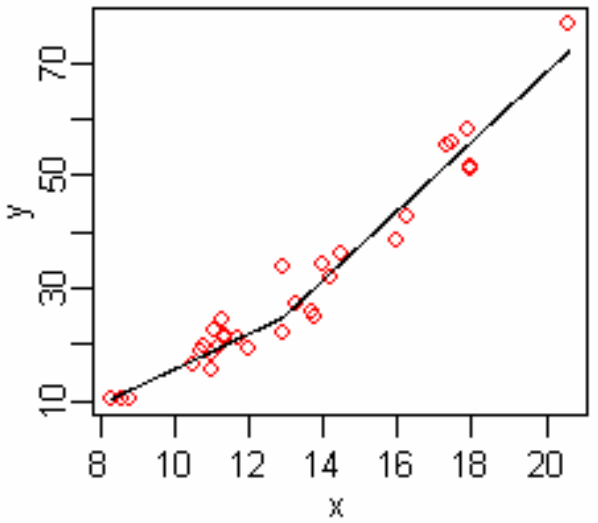
$\hat{y} = -37 + 5.1x$

Lineární regrese



$\hat{y} = 25 + 6.1\max(0, x - 13) - 3.1\max(0, 13 - x)$

MARS



MARS x lineární regrese

Mějme regresní rovnici:

$$Y = \beta_0 + \sum_{m=1}^{M} \beta_m \left(X_m \right) + \varepsilon$$

- kde Y je závisle proměnná, $X_1,\ldots, X_M$ jsou prediktory
- β_0 je intercept a $\beta_1,\ldots, \beta_M$ regresní koeficienty
- u jednorozměrné lineární regrese je k vyjádření závislosti Y na X použita přímka a koeficienty jsou odhadnuty metodou nejmenších čtverců

MARS

- předpokládejme model s jedním prediktorem a hodnotou uzlu $t = 10$, který můžeme zapsat pomocí dvou regresních rovnic:

$$Y = \beta_0 + \beta_1(X_1) + \varepsilon \qquad \text{pro } x > 10$$

$$Y = \beta_0 + \beta_2(X_1) + \varepsilon \qquad \text{pro } x < 10$$

Rovnice můžeme vyjádřit ve tvaru:

$$Y = b_0 + b_1(X_1 - t)_+ + b_2(t - X_1)_+ + \varepsilon$$

kde $b_0 \equiv \beta_0$, $b_1 \equiv \beta_1$ a $b_2 \equiv \beta_2$

MARS – interakce proměnných

- Stejně jako u lineární regrese lze i u metody MARS použít interakce proměnných
- pro dva prediktory X_1, X_2:

$$Y = b_0 + b_1(X_1 - t_1)_+ + b_2(t_1 - X_1)_+ + b_3(X_1 - t_1)_+(X_2 - t_2)_+ + \varepsilon$$

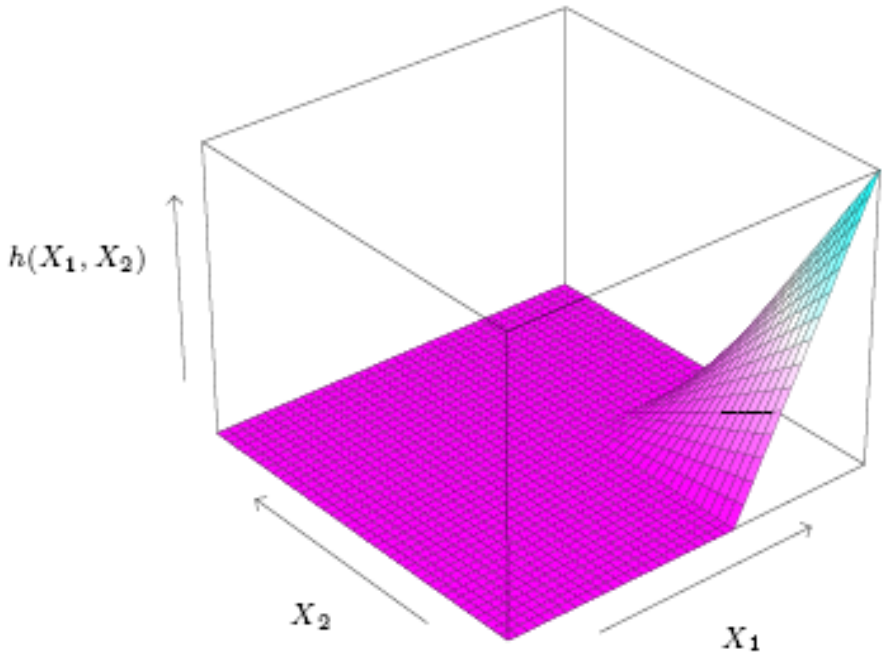
z čehož plyne:

$$Y = b_0 + b_1 X_1 - b_1 t_1 + \varepsilon \qquad pro \quad X_1 > t_1 \; a \quad X_2 < t_2$$

$$Y = b_0 - b_2 X_1 + b_2 t_1 + \varepsilon \qquad pro \quad X_1 < t_1$$

$$Y = b_0 + b_1 X_1 - b_1 t_1 + t_3(X_1 X_2 - t_1 X_1 - t_2 X_1 + t_1 t_2) + \varepsilon \qquad pro \quad X_1 > t_1 \; a \quad X_2 > t_2$$

MARS - interakce

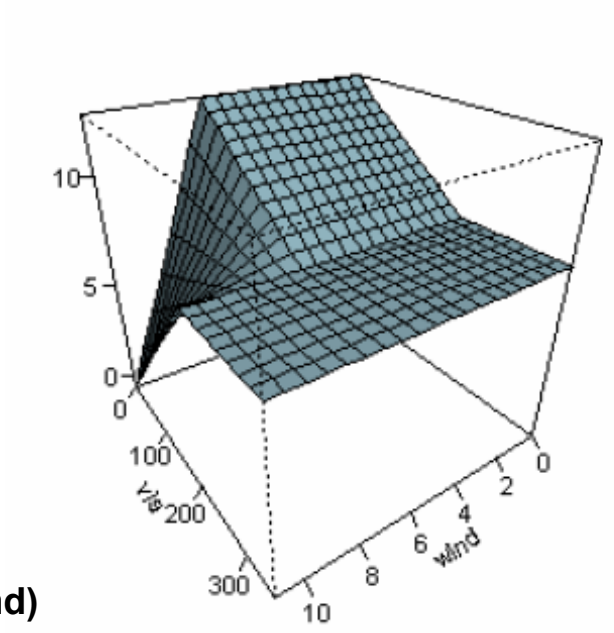


$$h(X_1,X_2) = (X_1\text{-}x_{51})_+ * (x_{72} - X_2)_+$$

(Hastie et. al, 2009)

MARS - příklad

% denní měření koncentrace ozonu, rychlosti větru, teploty vzduchu a intenzita slunečního záření v New Yorku



ozone = **25**
+ 3.1*max(0; temperature-85)
-1.28*max(0; 85-temperature)
-4.9*max(0; 13-wind)
- 0.09*max(0; radiation-139)
-0.049*max(0; radiation-112)*max(0; 13.21-wind)

MARS

Regresní funkci pro MARS můžeme tedy vyjádřit jako:

$$f(X) = \beta_0 + \sum_{m=1}^{M} \beta_m h_m(X)$$

- kde h_m jsou bázové funkce nebo jejich interakce a koeficienty b_m pro dané h_m jsou odhadovány stejně jako u lineární regrese metodou nejmenších čtverců.
- Algoritmus MARS je velmi podobný postupnému dopřednému výběru (*forward stepwise selection*) vysvětlujících proměnných v regresním modelu → namísto proměnných se vybírají lineární splajny.
- Začínáme s nulovým modelem (bez prediktorů).
- Postupně se přidávají jednotlivé členy do rovnice (bázové funkce) → pouze takové, jejichž příspěvek k variabilitě vysvětlené modelem je statisticky významný.
- Tento příspěvek se určuje na základě snížení residuálního součtu čtverců modelu.

MARS- krovalidace

- krosvalidační kritérium *GCV* (*generalized cross-validation)* → vybere se model s optimálním počtem členů v rovnici.
- *GCV* lze použít i pro odhady relativních významností jednotlivých prediktorů.

$$GCV(\lambda) = \frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{(1 - M(\lambda)/N)^2}$$

kde N je počet pozorování, $\hat{y}_i$ je hodnota závisle proměnné odhadnutá modelem a $M(\lambda)$ je parametr složitosti modelu, který má tvar:

$$M(\lambda) = r + cK$$

kde r je počet nekonstantních bázových funkcí v modelu a K je počet uzlů t v modelu, kde již proběhl výběr parametrů pomocí dopředného výběru

Konstanta c je určena experimentálně:

c = 3 pokud nejsou zahrnuty interakce

c = 2 pro rovnici s interakcemi

MARS - krovalidace

- Datový soubor je rozdělen na testovací a trénovací v poměru zadaném uživatelem (často 70% trénovací a 30% testovací)
- Na trénovacím souboru je vytvořen model a je spočítána jeho přesnost (R^2) na testovacím souboru.
- Hodnota *GCV* je spočítána pro různé podmodely, mající různý počet členů v rovnici, který označuje parametr M(λ).
- Je vybrán podmodel s nejmenší hodnotou *GCV*.
- Analogie s CART a CHAID → optimální počet terminálních uzlů stromu a PRIM → okna optimální velikosti.

Algoritmus metody MARS

- Krok1: Algoritmus začíná s konstantní funkcí $h_m(X) = 1$

- Krok2: Vytvoří se splajny (zrcadlové páry) se svým středem (uzlem t) v každé hodnotě x_{ij}, pro každý prediktor $X_j \rightarrow$ získáme množinu všech „kandidátských“ bázových funkcí $C \rightarrow$ model je tvořen prvky z této množiny nebo jejich kombinací.

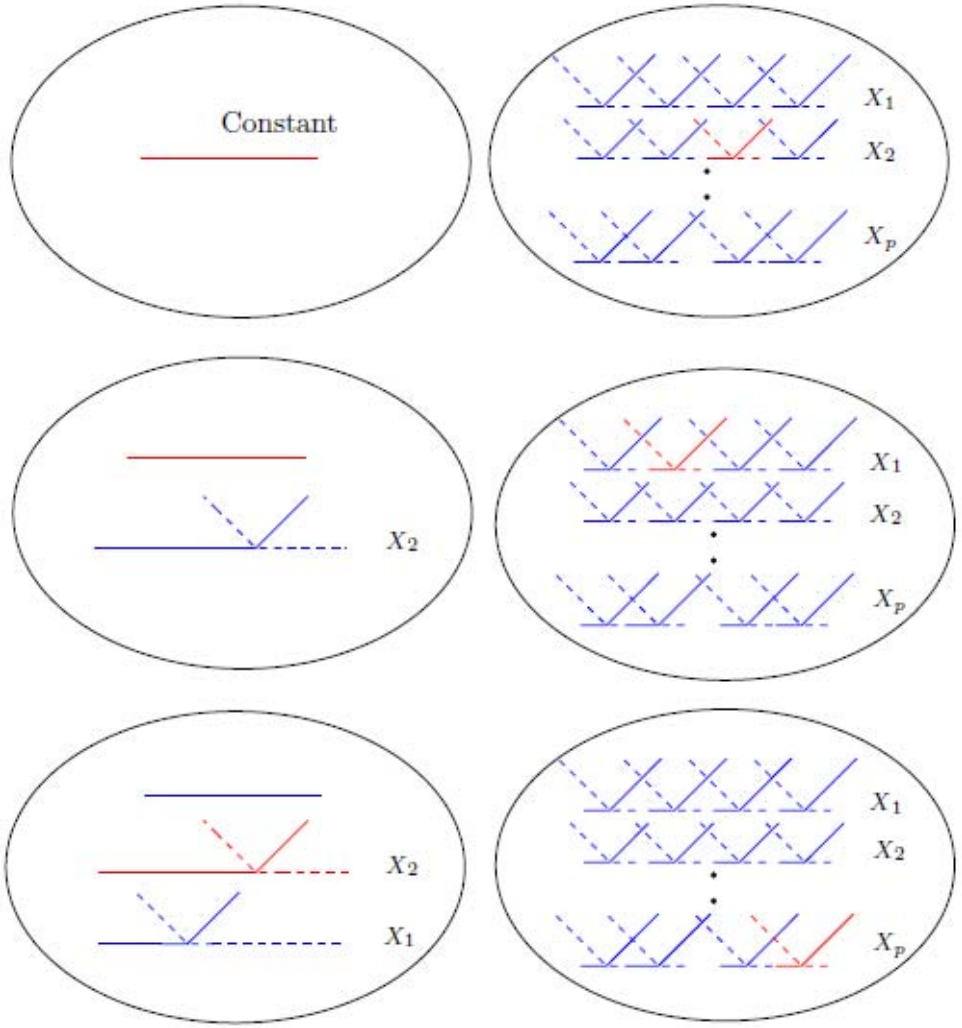
- Krok3: Z množiny C jsou do modelu přidávány pomocí postupného výběru významné bázové funkce, které snižují reziduální chybu modelu.

  !Proces postupuje hierarchicky, významné interakce jsou přidávány do modelu pouze z kombinace bázových funkcí, které již byly do modelu vybrány!

- Z kroku 1 - 3 jsme získali rovnici s vybranými členy → počet členů však bývá většinou velmi velký

- Krok4: procedura zpětného odstraňování.
  - Z rovnice jsou odstraněny ty členy, u kterých po jejich odstranění dojde k nejmenšímu zvýšení chyby modelu.
  - Zpětné odstraňování je učiněno pomocí krosvalidace. Hodnota *GCV* je spočítána pro různé velikosti modelu (s různým počtem členů v rovnici) a je vybrán model, pro který je hodnota *GCV* minimální.

MARS - algoritmus



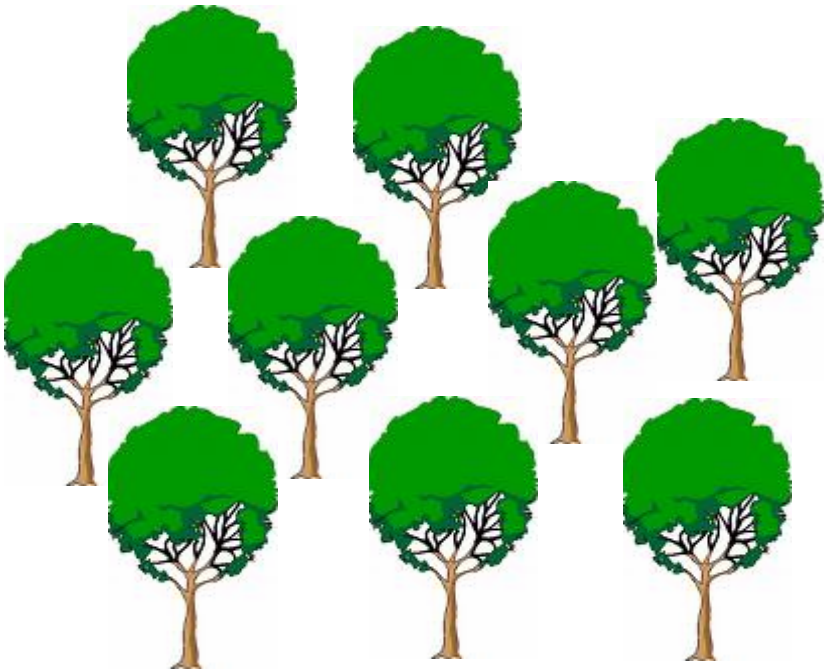
(Hastie et. al, 2009)

MARS

☺ modelovaná plocha je spojitá
☺ zahrnuje aditivitu proměnných
☺ zahrnuje interakci proměnných
☺ vhodná i pro větší počet prediktorů

☹ nevýhodou je méně názorná interpretace → chybí stromová struktura
☹ dopředný výběr proměnných je hierarchický
☹ každý vstup se může v modelu objevit pouze jednou

- PolyMARS (Stone et al., 1997) – pro klasifikaci



Skupinové modely

Klasifikační a regresní lesy

Moudrost davu (*Wisdom of Crowds*)

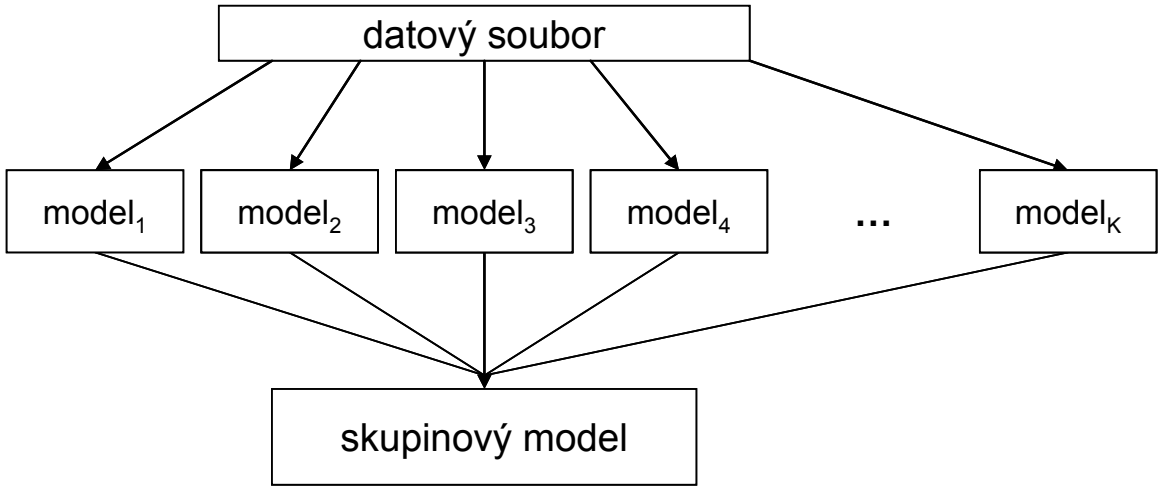
- James Surowiecki, 2004

„skupinový úsudek je daleko inteligentnější a přesnější než úsudek jednotlivce, v případech, kdy jde o hodnocení faktů“

- každý příslušník davu musí činit svůj úsudek na základě vlastních, nezávislých informací
- Výsledek je dán hlasováním

Skupinové modely (*ensamble models*)

- skupině modelů zadáme stejný problém, na kterém se naučí
- výstupy naučených modelů se kombinují
- výsledkem skupinového modelu je
  - v případě regrese → zprůměrování všech výsledků jednotlivých modelů
  - u klasifikace → většinové hlasování jednotlivých modelů (lze však použít průměrování)



Skupinové modely (*ensamble models*)

- Můžeme však kombinací modelů získat přesnější model?

- Podmínka → jednotlivé modely musejí být různé například použitím různých souborů pro učení modelu, které získáme náhodný výběrem z trénovací množiny dat.
- Modely tak budou vykazovat „odlišné“ chyby.
- Přesnost a stabilita těchto modelů se následně ověřuje na testovacích souborech.

- Označení skupinové modely se občas používá také pro kombinaci výsledků z různých modelů (např. neuronových sítí, rozhodovacích stromů a regrese) na stejném souboru.

Čím je způsobena chyba modelu…?

- Př: měříme náhodnou veličinu *Y* v populaci (např. váha člověka) a chceme vyjádřit její reprezentativní hodnotu pro celou populaci.

- Hledáme takový odhad $\hat{y}$, který minimalizuje střední hodnotu chyby $Ey(y-\hat{y})^2$ přes celou populaci.

- V ideálním případě bychom změřili všechny vzorky v populaci (zvážili všechny lidi) a zjistili jejich střední hodnotu $Ey(y)$ (např. průměr, medián), kterou bychom prohlásili za optimální odhad.

- V praxi však tento přístup není možný a pomůžeme si výběrem pouze určité skupiny pozorování z populace, který však musí mít stejné vlastnosti jako celá populace. Takovýto výběr vytvoříme náhodným výběrem.

Skupinové modely -Rozklad chyby

- analogie u modelů, kdy vybíráme pozorování pro trénovací soubor z množiny všech pozorování
- Odchylky pozorovaných od predikovaných hodnot (chybovost modelu) nebudou způsobeny pouze „přírodní“ variabilitou, kterou jsme modelem nevysvětlili, ale také rozdílem ve výsledcích pro různé náhodné výběry a celou populaci.

- Mějme soubor trénovacích dat:
  - $L = (\boldsymbol{y}_i, \boldsymbol{x}_i)$, $i = 1,\ldots,n.$

→ hledáme takovou funkci v prostoru všech prediktorů a hodnot závisle proměnné, aby predikční chyba byla malá.

Skupinové modely -Rozklad chyby

- Pokud mají (*Y*,*X*) stejné rozdělení a daná funkce *R* udává rozdíl mezi pozorovanou hodnotou y_i a predikovanou hodnotou $\hat{y}_i$ závisle proměnné *Y*, pak můžeme predikční chybu (*prediction error*) obecně vyjádřit jako:

$$PE(f, L) = E_{Y,X} R(Y, f(X, L))^2$$

- kde *f*(*X*,*L*) jsou predikované hodnoty $\hat{y}_i$ pro trénovací soubor *L*

Skupinové modely -Rozklad chyby

- Průměrná obecná chyba (*mean-squared generalization error*) na trénovacím souboru *L* je rovna:

$$PE(f, L) = E_{Y,X}\left(Y - f(X, L)\right)^2$$

- Optimální model by měl mít minimální průměrnou chybu pro různé výběry *L* → výsledky modelu pro jednotlivé výběry trénovacích souborů by se neměly příliš lišit.

- Vyjádříme průměr trénovacích souborů stejné velikosti ze stejného rozložení:

$$\bar{f}(x) = E_L\, f(x, L)$$

- kde $E_L\, f(x, L)$ je průměr přes všechny trénovací soubory *L* predikované hodnoty y_i v hodnotě x_i.

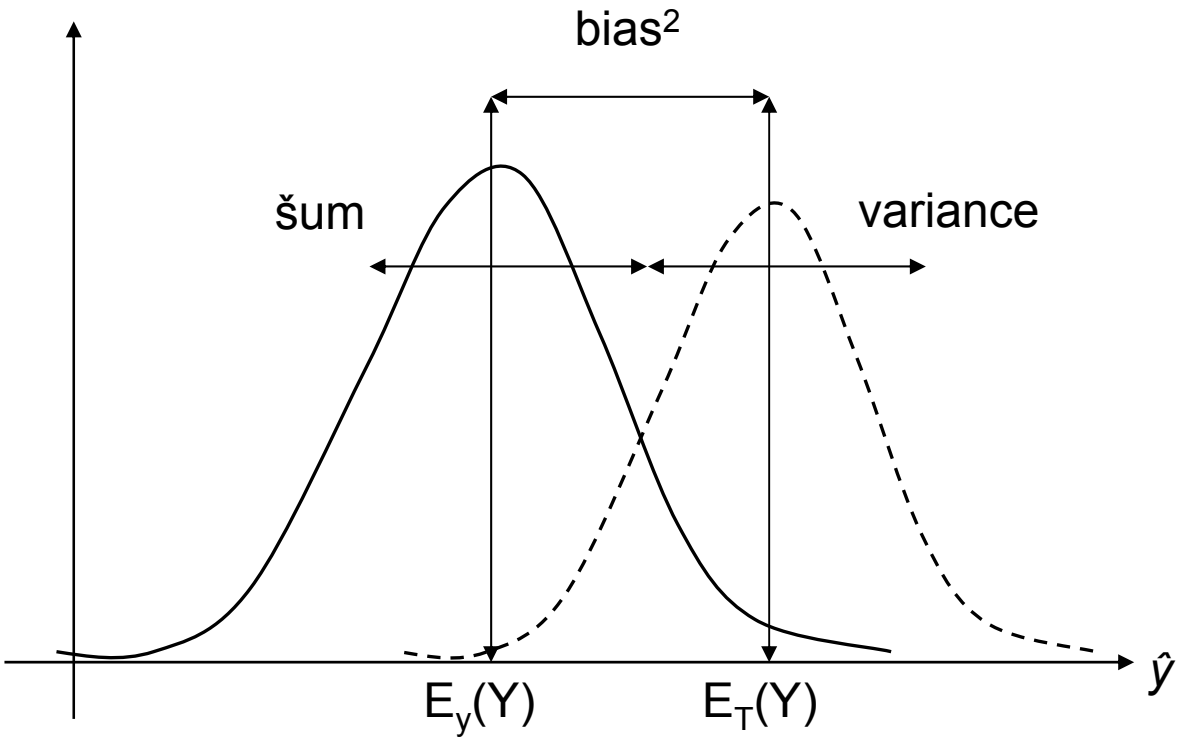
Rozklad na systematickou chybu a varianci (*Bias-Variance Decomposition*)

$$PE = E\varepsilon^2 + E_{Y,X}\left(f(X) - E_L f(X,L)\right)^2 + E_{X,L}\left(f(X,L) - E_L f(X,L)\right)^2$$

šum **zkreslení²** **variance**

- **Šum** – je reziduální chyba neboli minimální dosažitelná chyba modelu, kterou nejsme schopni modelem vysvětlit.
- **Zkreslení²**- určuje systematickou chybu modelu. Je to rozdíl optimálního modelu od průměrného modelu.
- **Variance –** je variabilita výsledků jednotlivých výběrů, jinými slovy, jak moc se predikované hodnoty $\hat{y}_i$ liší v rámci trénovacích podsouborů $L \rightarrow$ vysoká variance značí přeučený model.

Rozklad na systematickou chybu a varianci (*Bias-Variance Decomposition*)

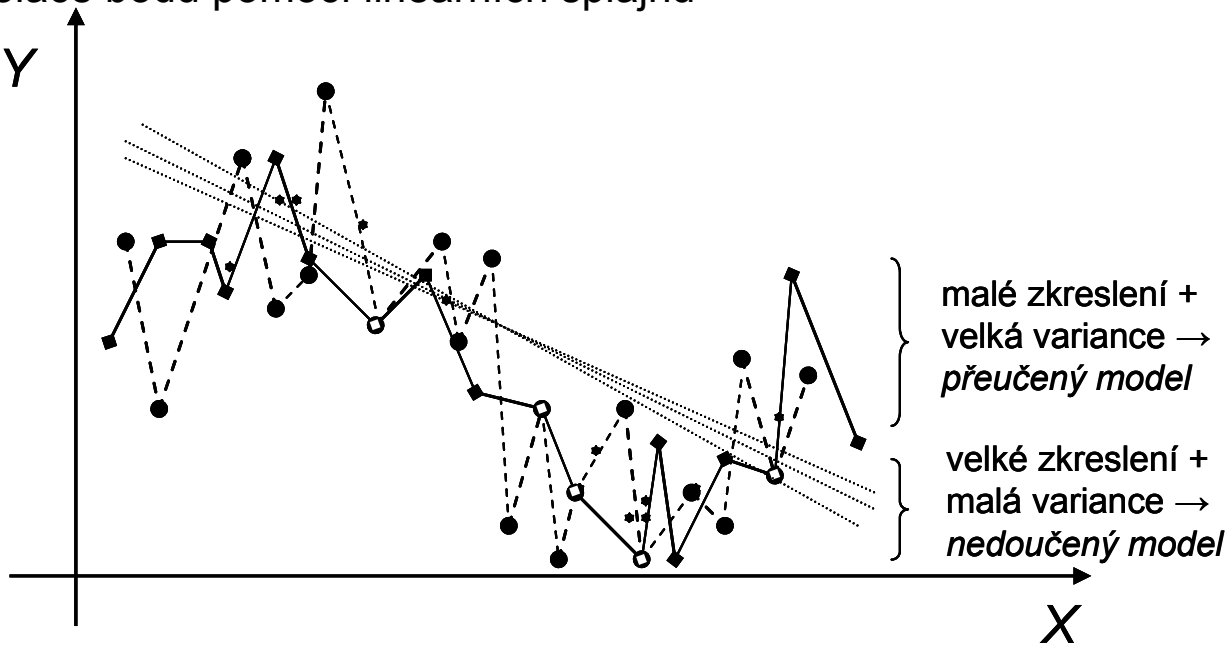


Šum – chyba modelu
Zkreslení²- systematická chyba modelu → optimální x průměrný
Variance –variabilita výsledků jednotlivých výběrů

Slabé modely

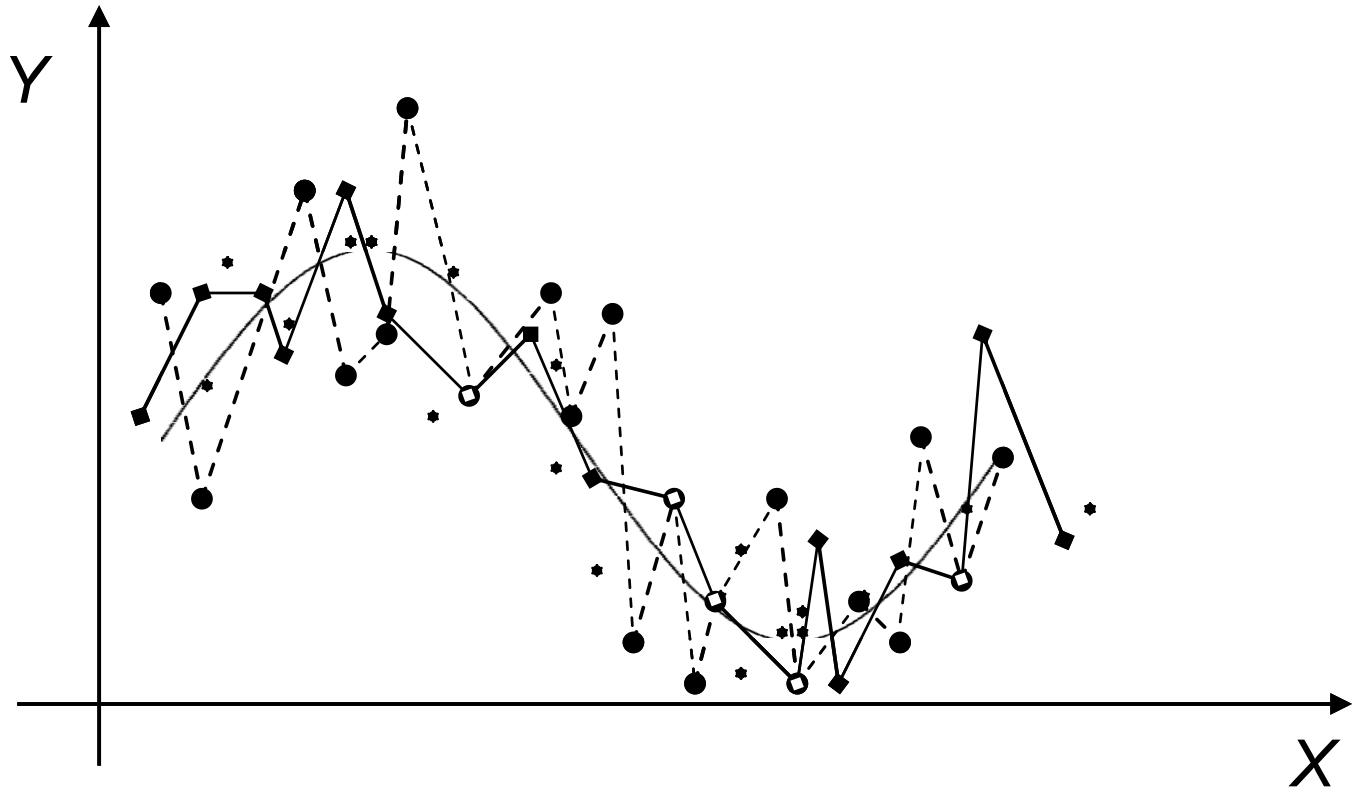
- Modely, které se používají ve skupinových modelech, se označují jako slabé modely neboli *weak learners* (slabý žák, u klasifikace také slabý klasifikátor).

- **Slabý model** je definován obecně jako model, který má malé zkreslení, ale vysokou varianci → mají velmi vysokou přesnost, ale pouze pro pozorování z trénovacího souboru
- Příkladem slabých modelů s velkým zkreslením, ale nízkou variancí může být interpolace bodů pomocí lineárních splajnů



Slabé modely – vytvoření skupinového modelu

- Hledáme tedy model, který by měl nízkou varianci i zkreslení. Kombinováním několika slabých modelů můžeme snížit obě tyto složky.
- Jak na to?



A co na to stromy?

- Rozhodovací stromy jsou dobrými kandidáty pro použití ve skupinových modelech.
- Neprořezané stromy mají totiž vysokou přesnost pro trénovací soubor (tedy nízký bias), ale vysokou varianci (výsledky mezi testovacím a trénovacím souborem se liší).
- Rozhodovací stromy, na které nejsou aplikovány metody pro hledání optimální velikosti stromu, jsou tedy podle výše uvedené definice slabými modely.
- u rozhodovacích stromů jsme pro určení jeho optimální velikosti museli rovněž najít kompromis mezi variancí a zkreslením!