Urychlení evolučních algoritmů pomocí regresních stromů a jejich zobecnění

Jan Klíma

Obsah

- Motivace & cíle práce
- Evoluční algoritmy
- Náhradní modelování
- Stromové regresní metody
- Implementace a výsledky testů

Motivace

- Empirické cílové funkce a evoluční optimalizace
- Vyhodnocení cílové funkce bývá
  - časově náročné
  - drahé
- Příklady – Aerodynamika, katalýza
- Řešení = použití náhradního modelu místo cílové funkce

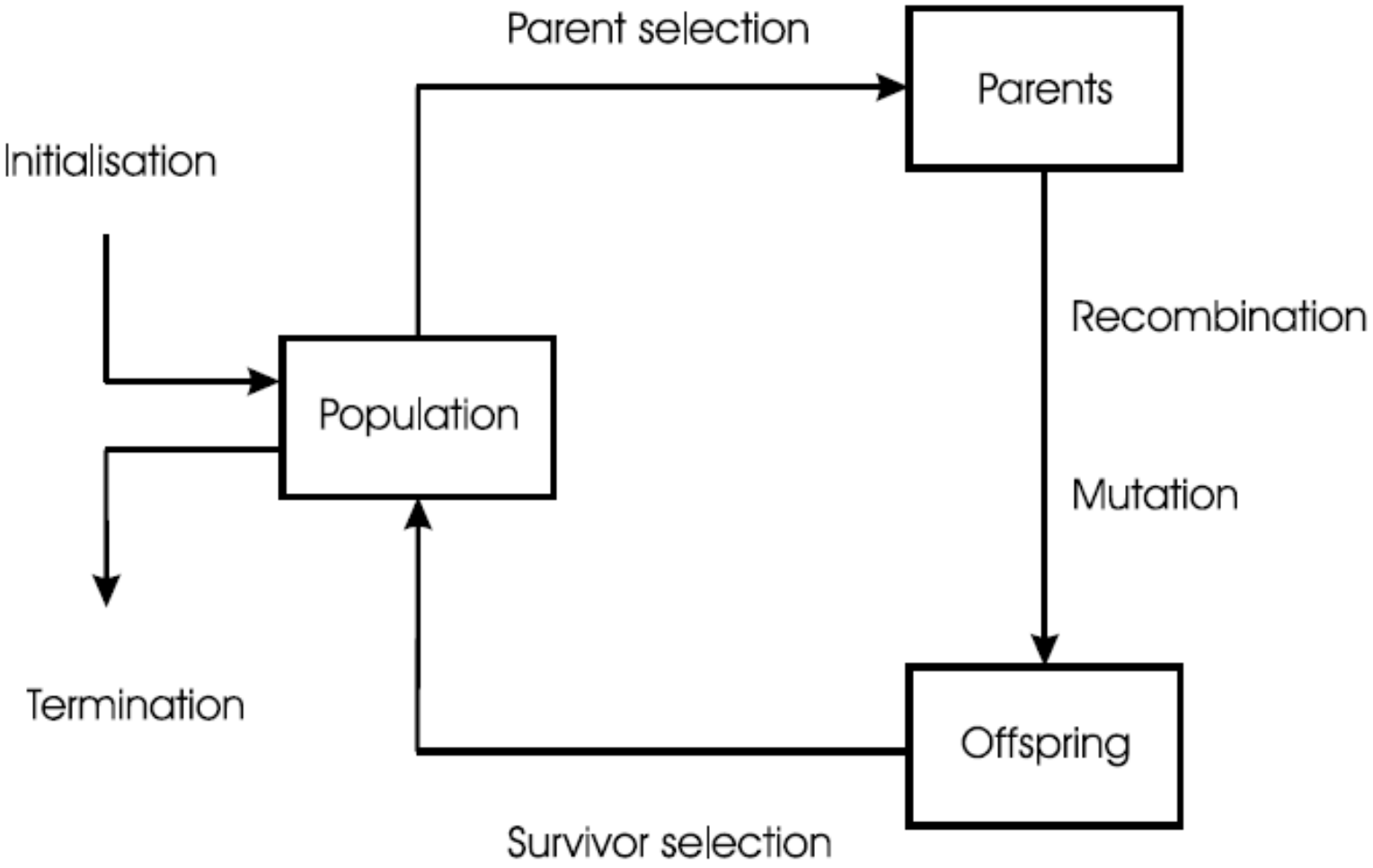
Cíle práce

- Návrh evolučního algoritmu
  - Náhradní model = regresní stromy a odvozené metody
    - Umí pracovat s diskrétními proměnnými
  - Vstup algoritmu ve speciálním tvaru vhodném pro řešení optimalizačních problémů v katalýze
- Otestování prototypové implementace navržených algoritmů

Evoluční algoritmy

- Stochastická optimalizační metoda
- Jedinec = řešení problému
  - Volba vhodného kódování
- Fitness = kvalita jedince
- Populace = soubor jedinců
- Selekce
- Operátory křížení a mutace
- Elitismus

Evoluční algoritmy - schéma

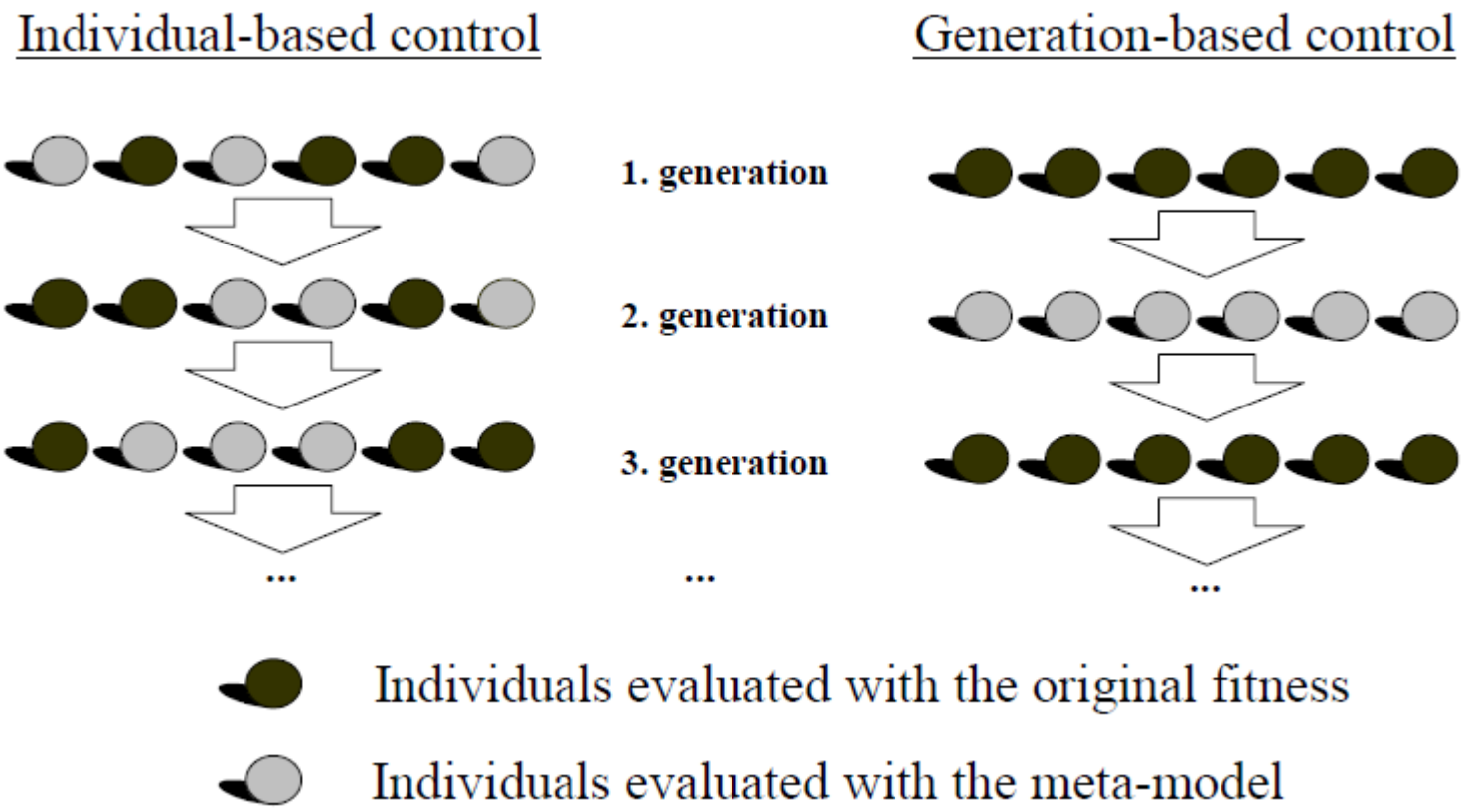


Náhradní modelování

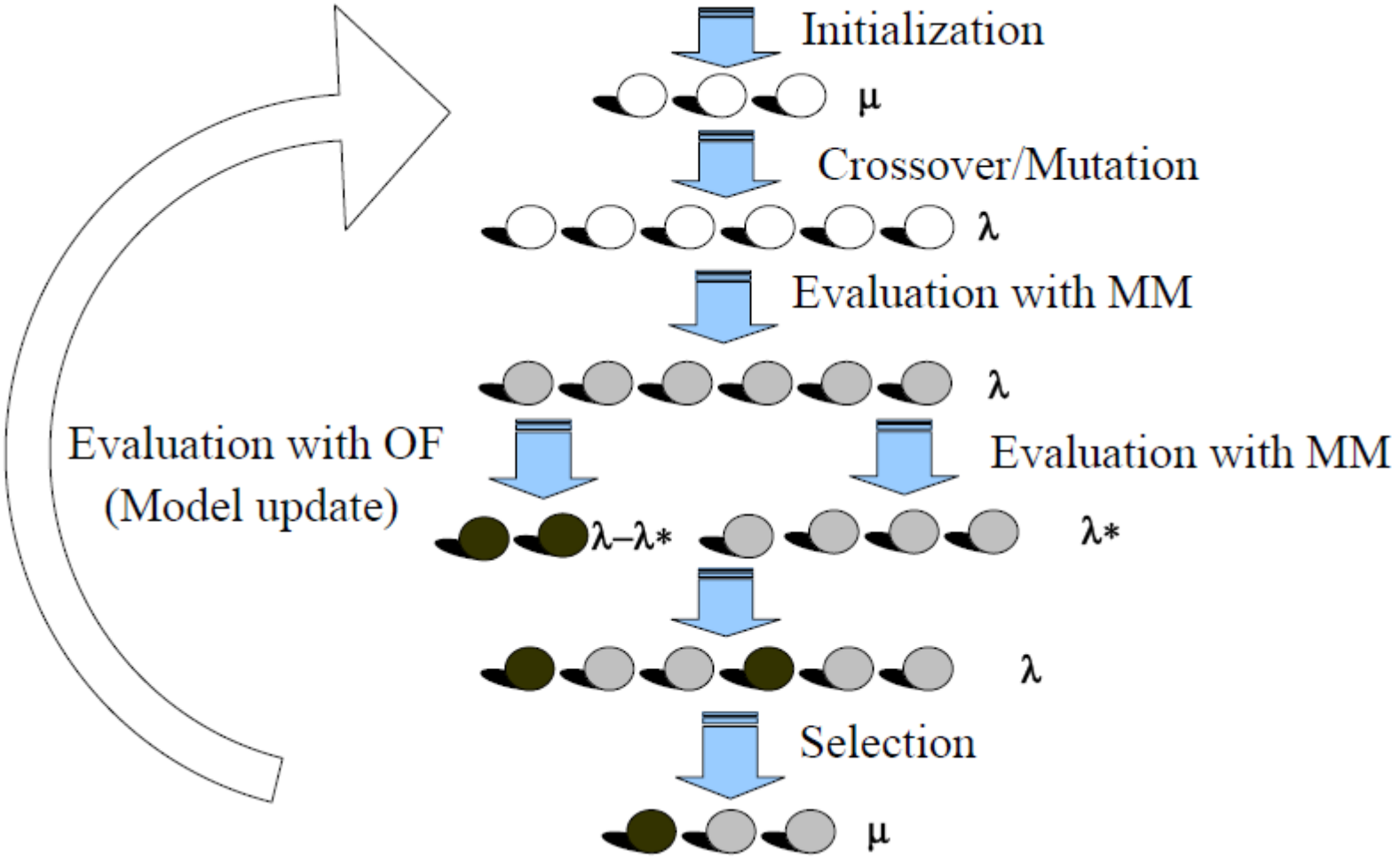
- Při výpočtu fitness model místo cílové funkce
- Cíl = rychlejší konvergence v závislosti na počtu vyhodnocení cílové funkce
- V literatuře především spojité modely
  - Neuronové sítě
  - Gaussovské procesy
  - Polynomiální modely
- Databáze hodnot cílové funkce
- Evoluční řízení

Evoluční řízení

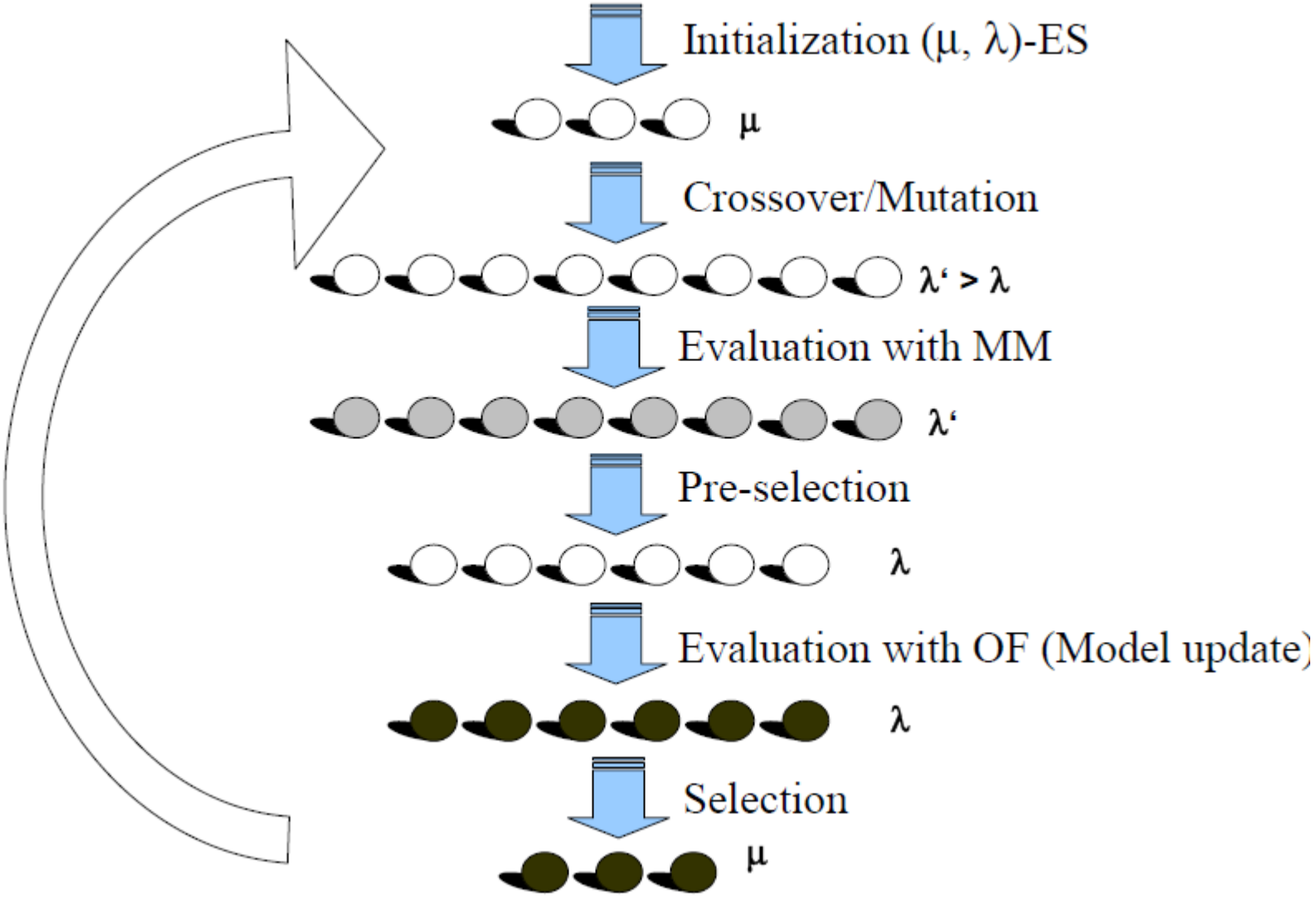
- Individuální
- Generační



Individuální evoluční řízení



Preselekce


Initialization (μ, λ)-ES
μ
Crossover/Mutation
λ‘ > λ
Evaluation with MM
λ‘
Pre-selection
λ
Evaluation with OF (Model update)
λ
Selection
μ

Individuální evoluční řízení

- Best strategie
- Random strategie
- Shlukovací strategie
  - Jak vybírat reprezentanty shluků?
- Strategie založená na odhadu chyby modelu

Generační evoluční řízení

- Cílová funkce se použije pro ohodnocení λ populací v cyklu délky $\mu > \lambda$ populací
- Volba λ, μ:
  - Pevná (heuristika)
  - Adaptivní
    - V závislosti na kvalitě modelu

Stromové regresní metody

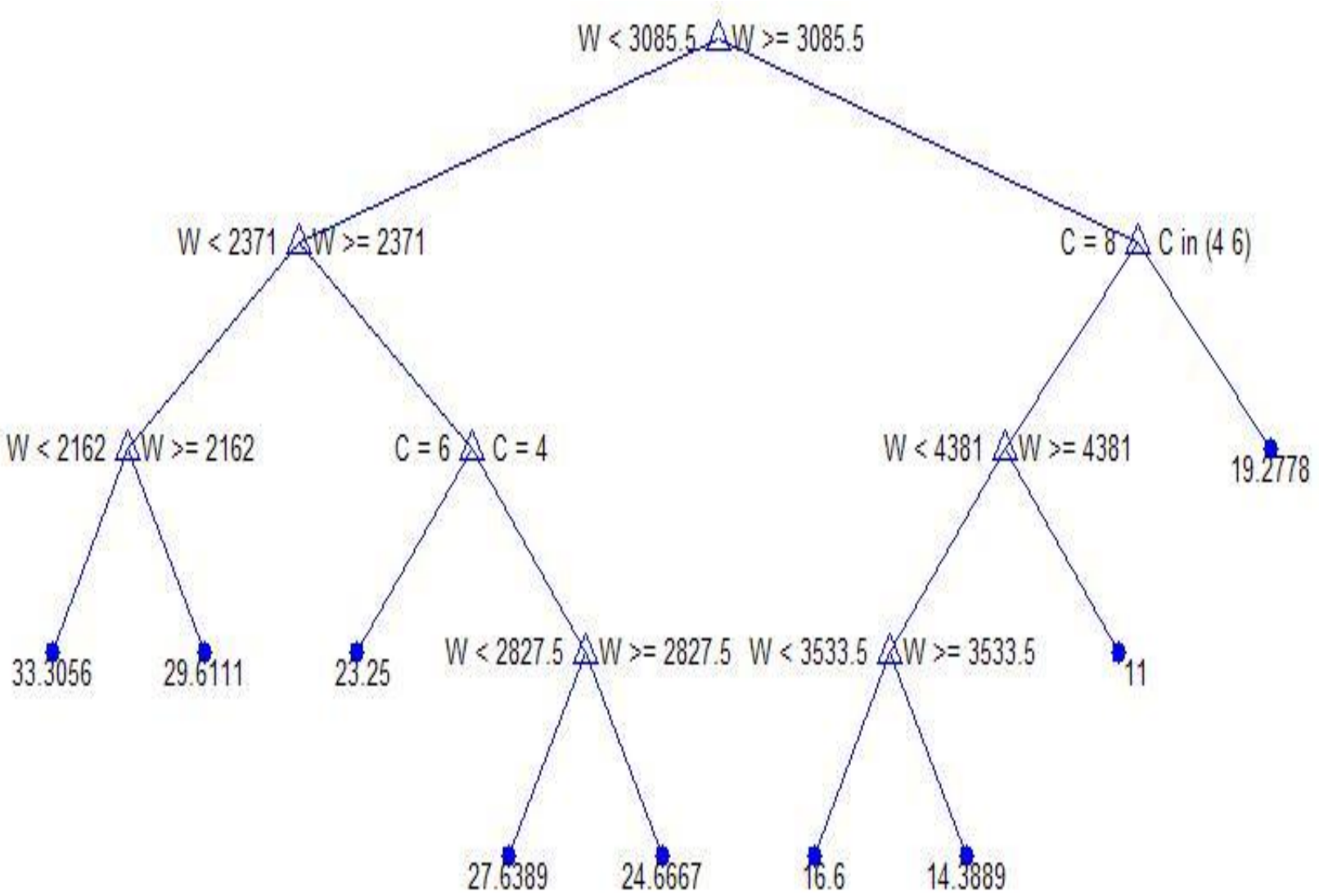
- Regresní stromy
- Bagging
- Náhodné lesy
- AdaBoost R2
- Gradientní boosting

Regresní stromy

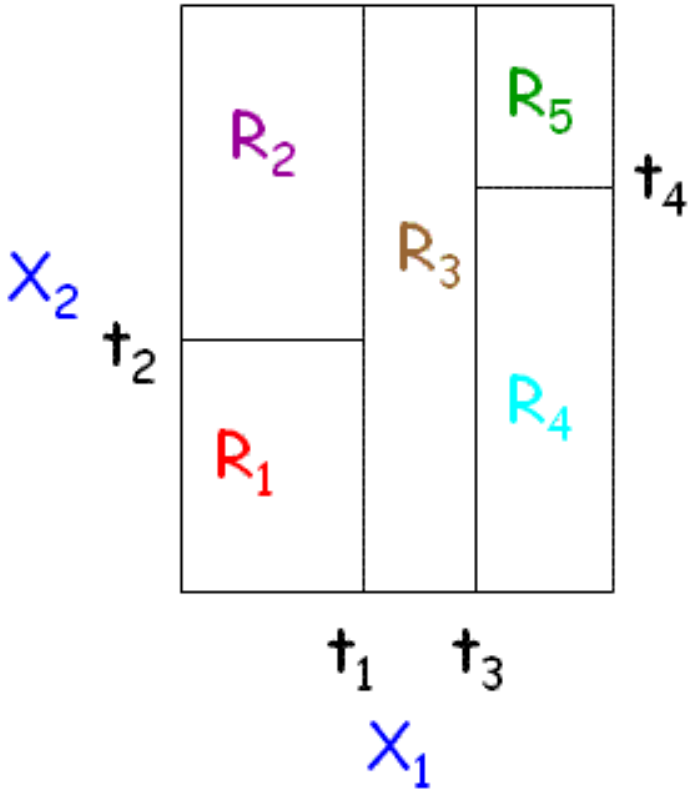
- Základní publikace je Breiman a kol. (1984)
  - Binární stromy
  - Vnitřním vrcholům stromu přiřazena pravidla tvaru $X_n \leq t_n$ pro spojité proměnné nebo $X_n \in subset(dom(X_n))$ pro diskrétní proměnné
  - Listům jsou přiřazeny konstanty $c_m \in R$
  - Každý list odpovídá části def. oboru R_m
  - Odezva v bodě x je dána předpisem

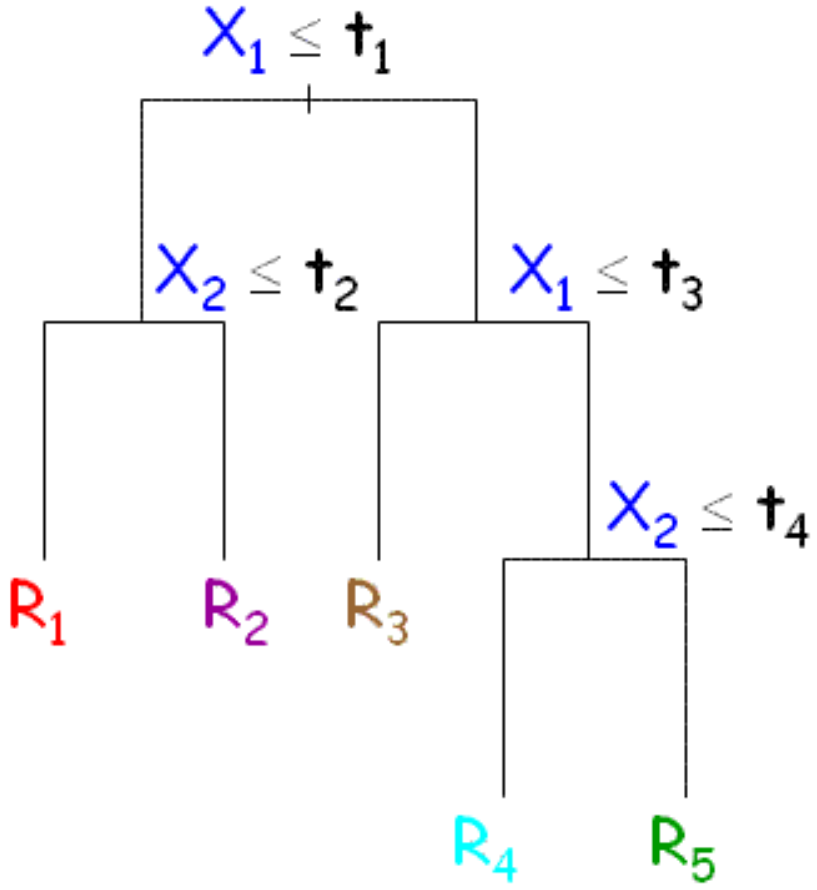
$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m)$$

Regresní stromy



Regresní stromy





Regresní stromy - konstrukce

- Počáteční strom obsahuje jediný list a

$$c_1 = mean(y \in Data)_j$$

- Iterativně:
  - Vyber vrchol a dělení s největším ziskem

$$\sum_{(x,y)\in Parent}(y - y_{Parent})^2 - \sum_{(x,y)\in Left}(y - y_L)^2 - \sum_{(x,y)\in Right}(y - y_R)^2$$

  - Vrcholu přiřaď rozhodovací pravidlo odpovídající dělení a nastav pro oba potomky

$$c_{Leaf} = mean(y \in Leaf)$$

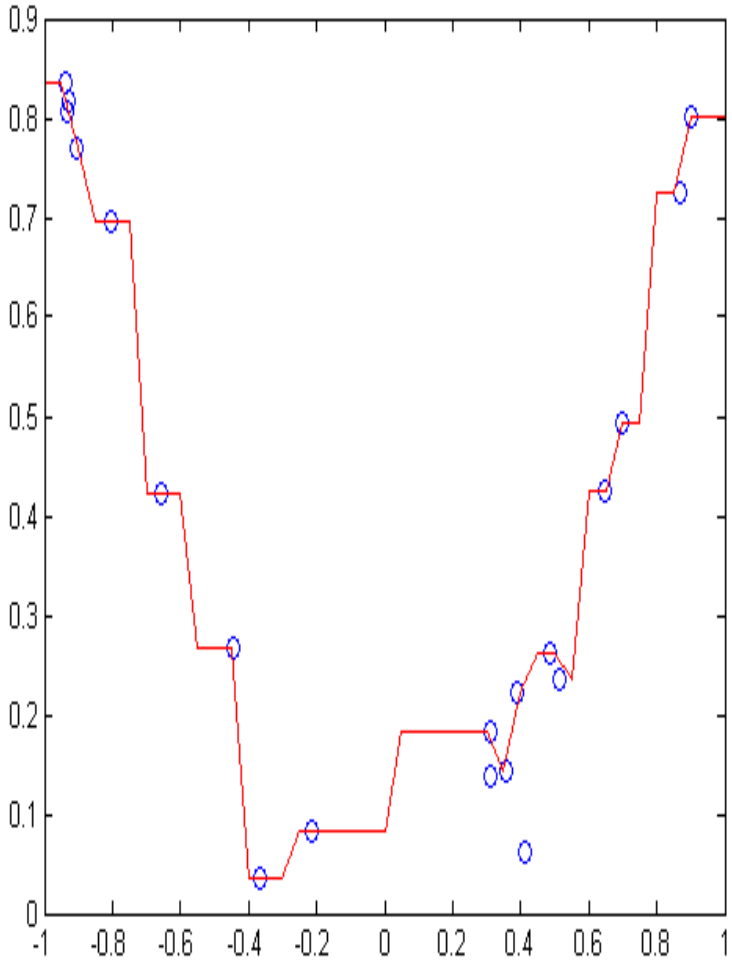
Regresní stromy - konstrukce

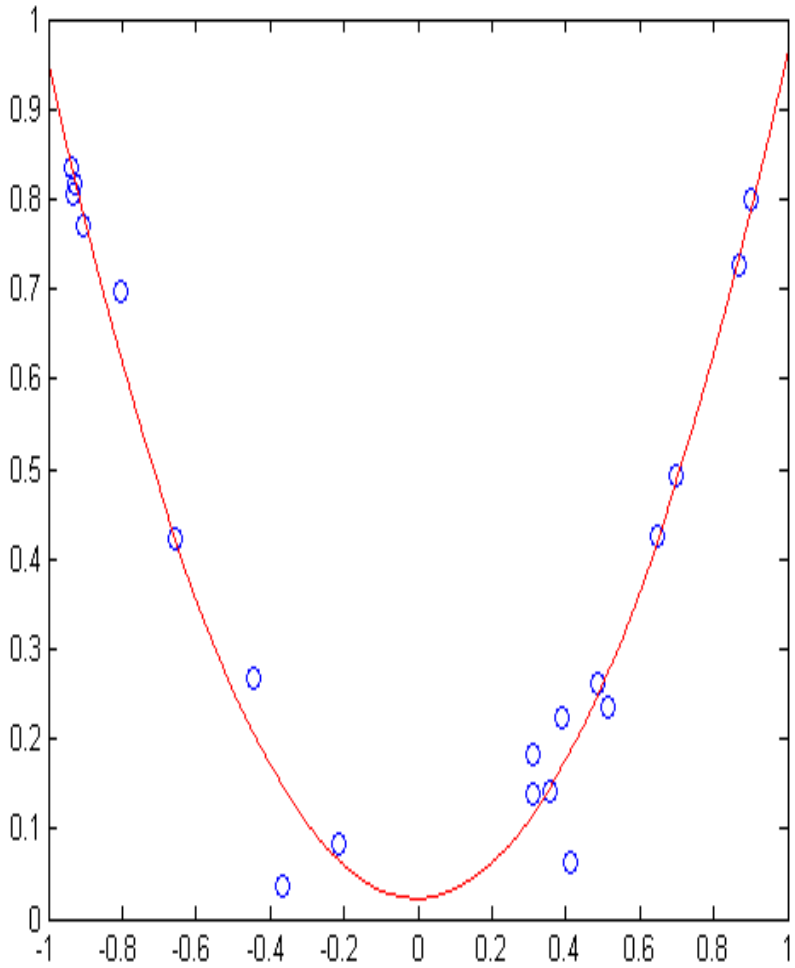
- Volba ukončovacího kriteria
  - Problém přeučení
- Prořezávání
  - Optimální prořezávací sekvence vzhledem k $MSE + \alpha \cdot \#listů$
  - 0-SE, 1-SE pravidla
  - Možnost využití křížové validace

Regresní stromy - vlastnosti

- Výhody
  - Jednoduchá konstrukce i interpretace
  - Rychlost
  - Kombinace spojitých i diskrétních proměnných
- Nevýhody
  - Po částech konstantní model
  - Nestabilní

Regresní stromy vs. polynom. model





Stromový bagging

- Bagging = Bootstrap Aggregation
- N regresních stromů natrénováno bez prořezávání
  - Trénovací množina pro i-tý strom je vybrána z původní trénovací množiny s opakováním
  - Odezva modelu v bodě x se pak vypočítá jako průměr odezvy jednotlivých stromů:

$$f(x) = \frac{\sum_{i=1}^{N} T_n(x)}{N}$$

Stromový bagging

- Průměrování
  - Odstraní nestabilitu
  - Brání přeučení
- Každý strom popisuje jinou část dat
  - Jeden vzorek má pst., že nebude vybrán

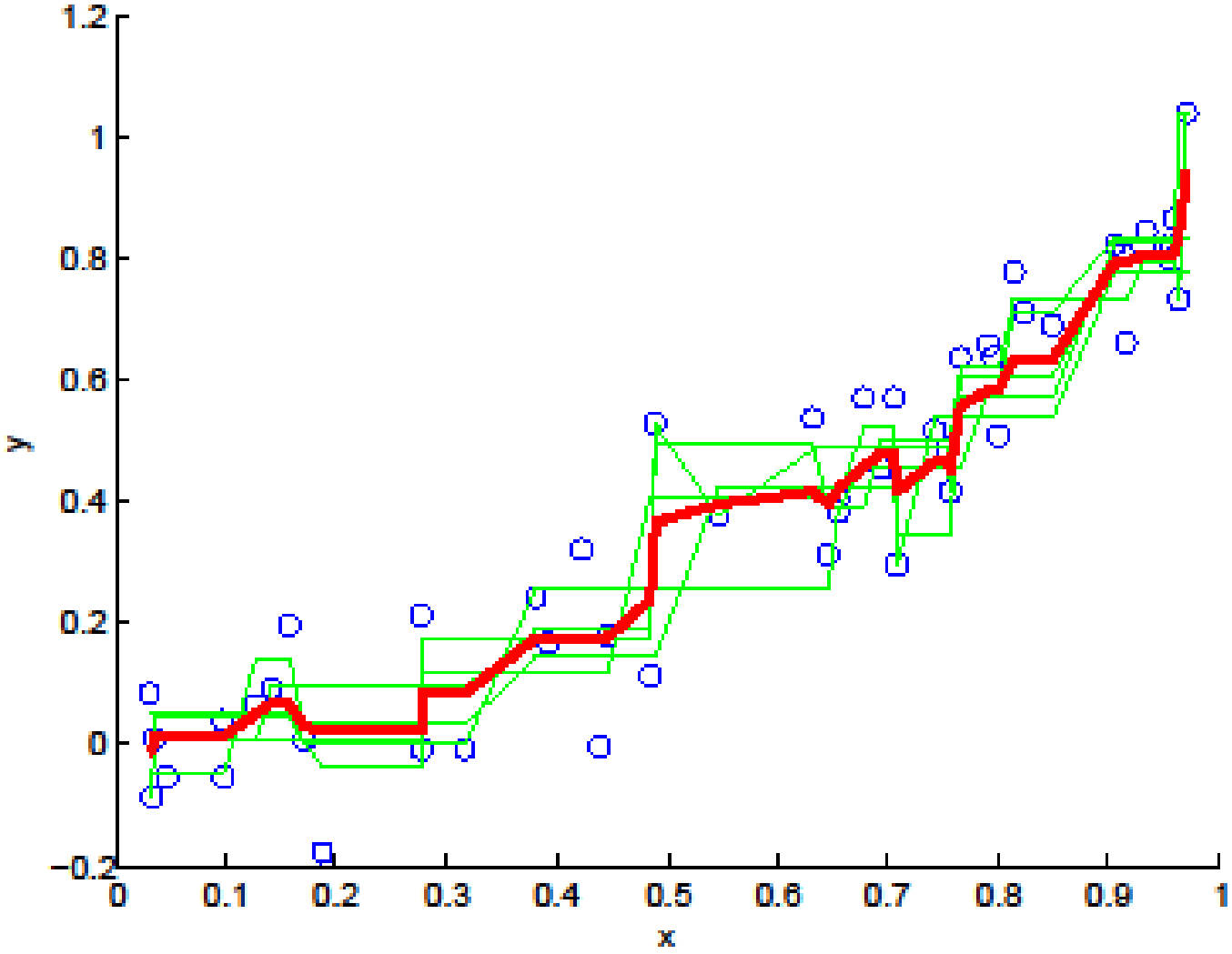
$$(1-\frac{1}{N})^N \approx e^{-1}$$

  - Přibližně 37% vzorků se pro jeden strom nepoužije
  - *Tzv. out-of-bag data*

Využití out-of-bag dat

- Odhad aproximační chyby modelu
  - Chyba se počítá pouze na vzorcích, které se nepoužily při trénování stromu
- Výpočet matice příbuznosti mezi jednotlivými vzorky $M = [\delta_{ij}]$
  - δ_{ij} je procento stromů, ve kterých i-tý a j-tý vzorek skončí ve stejném listu
  - Detekce outliers

Stromový bagging - příklad



Náhodné lesy

- Algoritmus téměř totožný s baggingem
  - Při dělení vrcholu se uvažují dělení jen podle náhodně vybrané podmnožiny všech proměnných
  - Počet uvažovaných proměnných je pro všechny vrcholy stejný (pro regresní stromy se doporučuje podmnožina třetinové velikosti)
- Každá proměnná má větší šanci přispět

AdaBoost R2

- Inspirace klasifikačním algoritmem AdaBoost
- Výběr trénovací množiny pro jeden strom s opakováním
  - Každý vzorek má přiřazenu pravděpodobnost vybrání
  - Po přidání nového stromu se pravděpodobnost vybrání vzorku mění exponenciálně vzhledem k velikosti chyby
- Odezva se počítá jako vážený medián odezev jednotlivých stromů vzhledem k míře spolehlivosti

Gradientní boosting

- Aditivní model
- Využívá stromy jako tzv. weak learner
  - Typicky stromy s pevným malým počtem listů (4-10)
- Nový strom se učí místo na hodnotách cílové funkce na negativním gradientu ztrátové funkce

$$gr_i = -[y_i - f_{m-1}(x_i)]$$

- Iterativní konstrukce modelu:

$$\hat{f}_0 = mean(y \in Data)$$

$$\hat{f}_m(x) = \hat{f}_{m-1}(x) + \upsilon \cdot Tree_m(x)$$

Gradientní boosting

- Volba υ
  - Doporučuje se malé, ~0.01
  - Kvalita modelu vs. rychlost konvergence
- Volba trénovací množiny
  - Celá trénovací data
  - Podmnožina trénovacích dat vybraná bez opakování (stochastický gradientní boosting)

Učení lesových modelů

- Pevný počet stromů
- Early stopping
  - Out-of-bag data
  - Validační množina
    - Klady vs. zápory

Implementace algoritmů

- v MATLABu
- Zmíněné stromové metody
- Genetický algoritmus
- Evoluční řízení
  - Individuální
  - Generační
- Trénování náhradního modelu
- Speciální genetické operátory pro křížení a mutace

Testování stromových metod

- Testovací data
  - Umělé testovací funkce
  - Standardní datasety z repozitářů
  - Dataset z reálného experimentu v katalýze
- Trénovací metody
  - Pevná velikost lesa
  - Early stopping
- 10 pokusů
- Lesové metody obecně podstatně lepší než samotný regresní strom

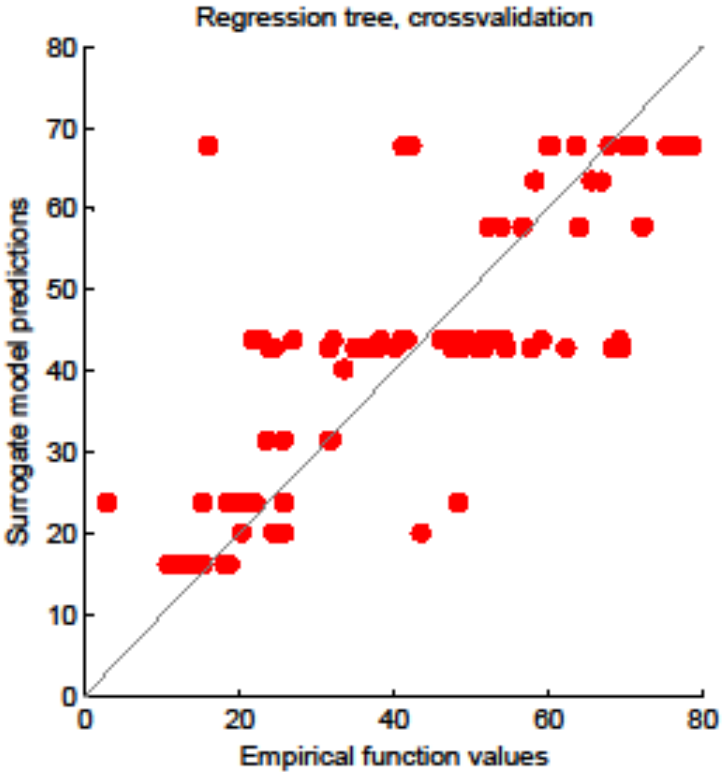
Testování stromových metod

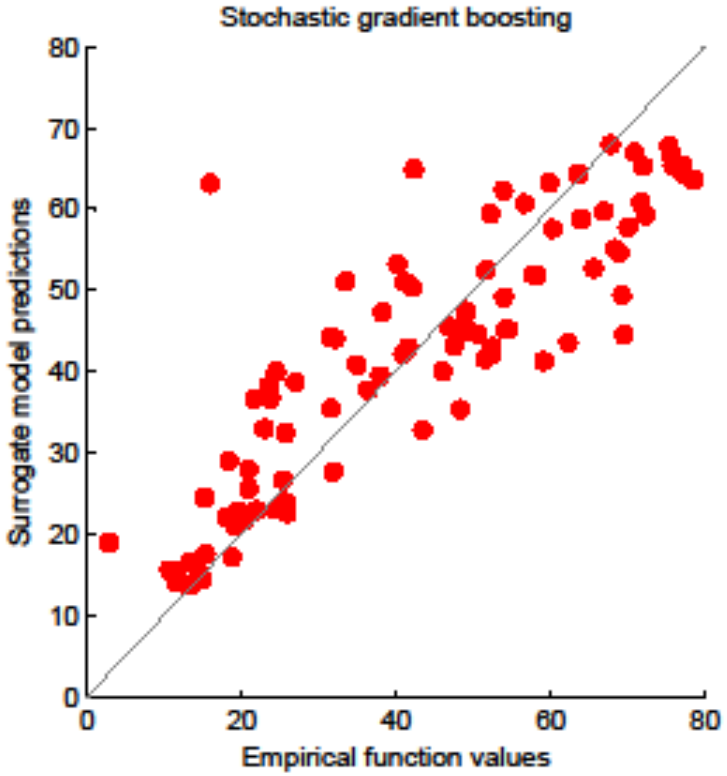
MSE	**RT**		**B**	**RF**	**SGB**	**ABR2**
Friedman #1	8.77 ± 0.77	(500)	3.59 ± 0.30	3.60 ± 0.22	1.99 ± 0.16	3.00 ± 0.18
		(VS)	3.83 ± 0.43	3.89 ± 0.43	2.07 ± 0.29	3.39 ± 0.52
		(OOB)	3.63 ± 0.32	3.67 ± 0.23		
Friedman #2	97.99 ± 15.83	(500)	34.57 ± 4.16	39.03 ± 7.54	19.11 ± 3.98	65.52 ± 6.66
		(VS)	35.73 ± 5.62	43.11 ± 7.90	17.46 ± 3.11	66.14 ± 11.00
		(OOB)	35.80 ± 4.90	41.62 ± 8.40		
Friedman #3	22.15 ± 4.37	(500)	6.81 ± 1.97	7.30 ± 1.45	4.39 ± 1.04	8.98 ± 2.56
		(VS)	8.37 ± 1.88	9.04 ± 2.15	4.83 ± 1.14	10.32 ± 3.13
		(OOB)	7.20 ± 2.10	7.60 ± 1.51		

Testování stromových metod

Nodes	RT		B	RF	SGB	ABR2
Friedman #1	36	(500)	312,826	313,755	5,500	200,949
		(VS)	46,669	42,092	5,497	31,048
		(OOB)	84,236	77,389		
Friedman #2	34	(500)	312,510	313,850	5,500	193,710
		(VS)	33,099	36,319	4,828	14,236
		(OOB)	78,480	64,667		
Friedman #3	33	(500)	312,880	313,901	5,500	148,030
		(VS)	18,369	27,676	5,432	11,349
		(OOB)	44,858	73,865		

Testování stromových metod





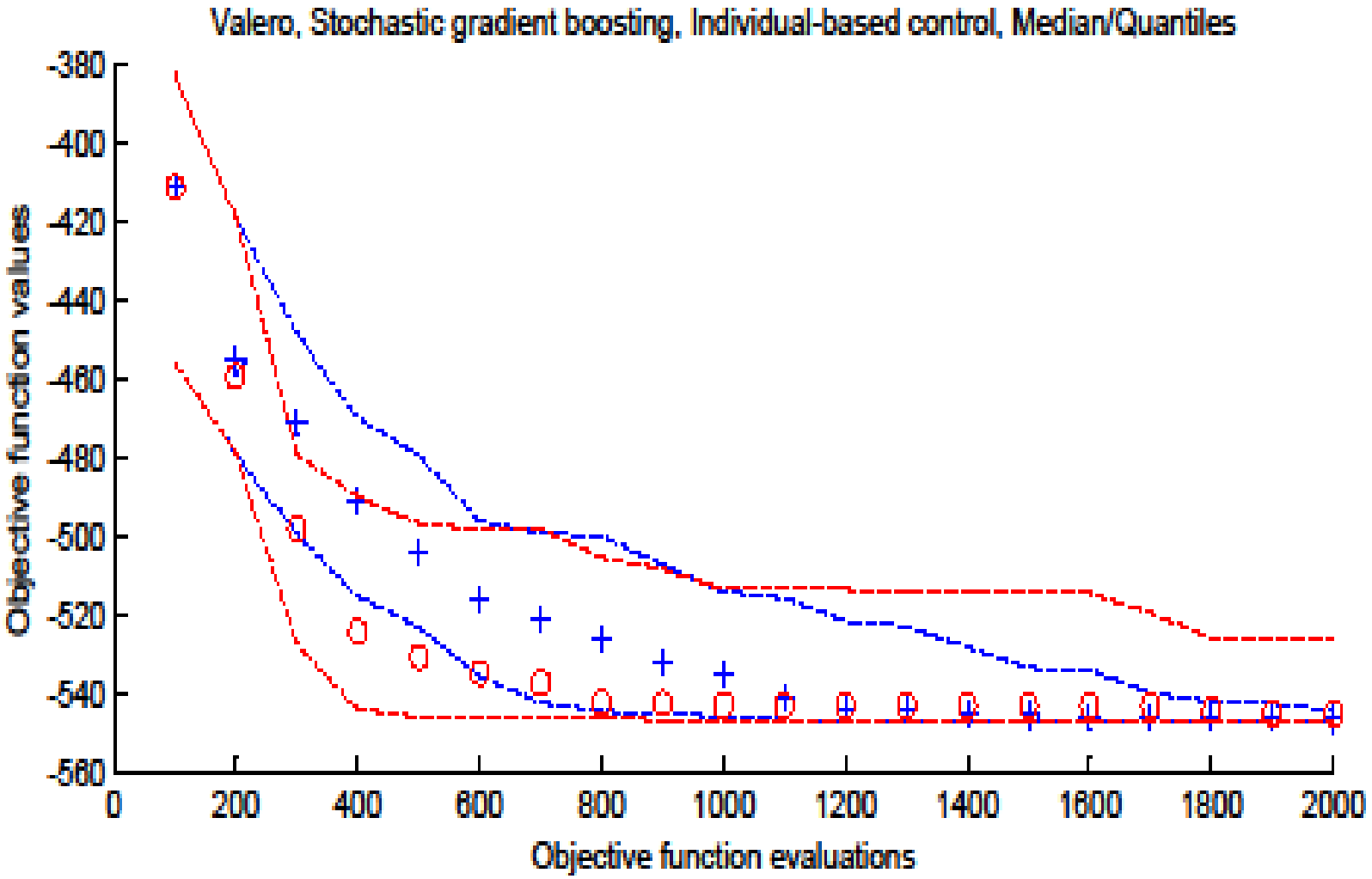
Testování genetického algoritmu

- 2 různé benchmarkové funkce:
  - První pouze spojitých proměnných
  - Druhá spojitých i diskrétních proměnných
- Individuální i generační strategie
- Vybrané typy náhradních modelů
- 50 běhů algoritmu s velikostí populace 100
- Náhradní model vždy od 3. generace
- Maximálně 2000 vyhodnocení cílové funkce

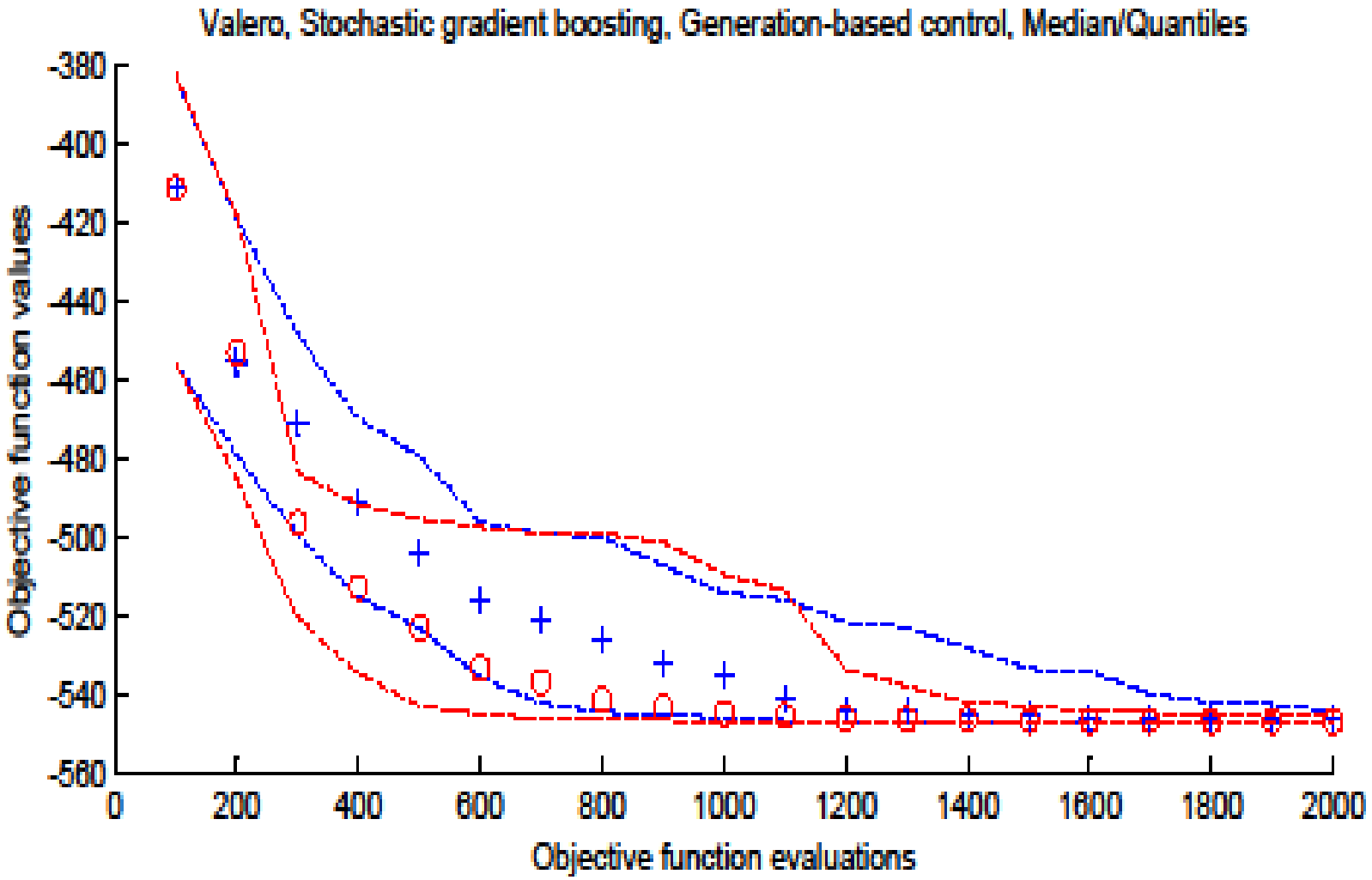
Testování genetického algoritmu

- Ve většině případů urychlení konvergence
- Generační strategie pro obě funkce lepší než individuální
- Přínos modelu s rostoucím počtem generací klesá
  - Někdy až zhoršení chování s modelem
- Nejlepší AdaBoost R2 a stochastický gradientní boosting

Testování genetického algoritmu



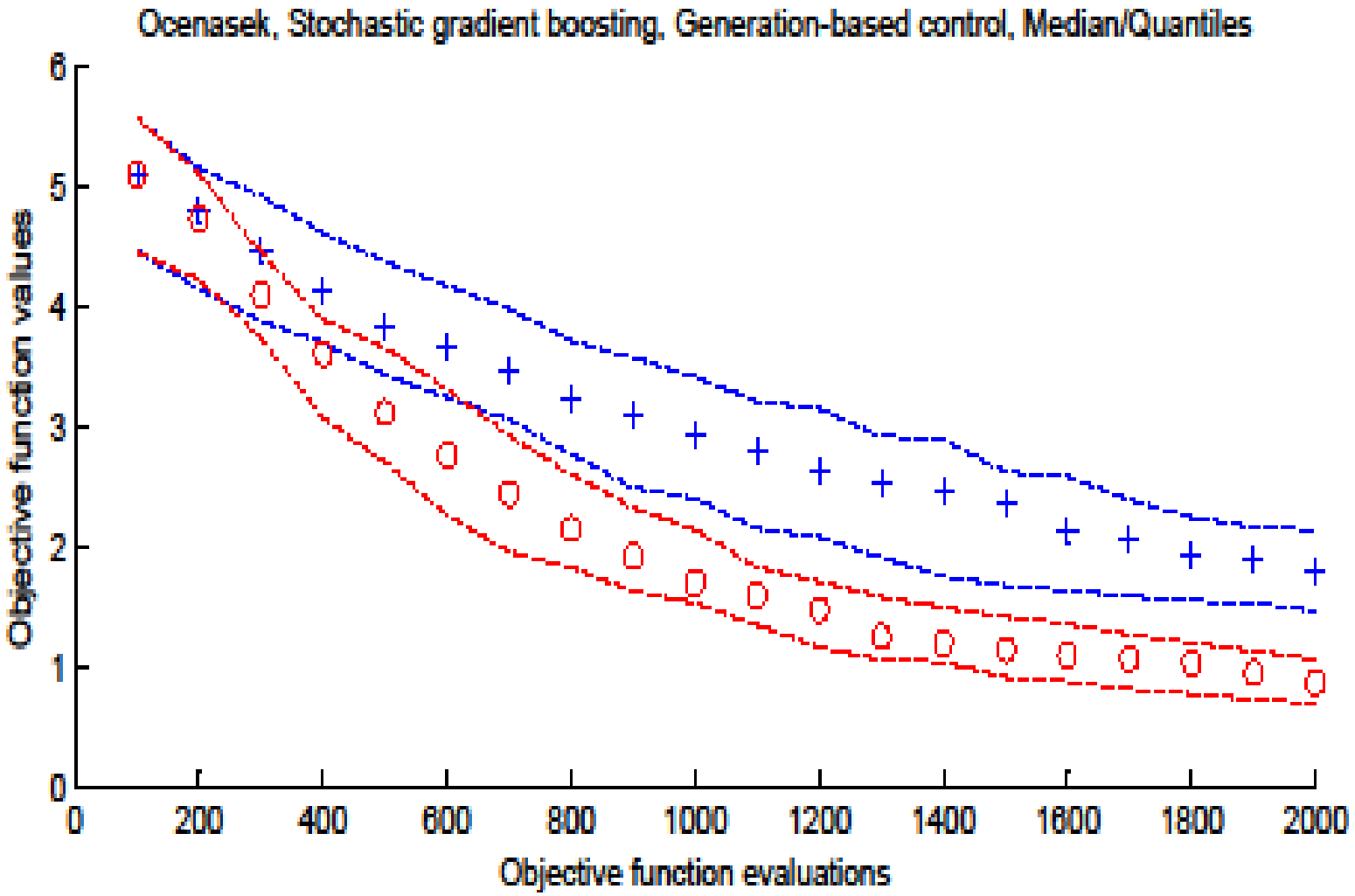
Testování genetického algoritmu



Testování genetického algoritmu

Valero/Gen	**WEC**	**RT**	**B**	**RF**	**SGB**	**ABR2**
Mean	-544.26	-545.06	-543.68	-541.42	-546.25	-546.89
Deviation	8.21	9.48	11.47	14.37	2.56	0.79
Median	-546.64	-547.31	-546.95	-546.47	-546.91	-547.09
0.159 quantile	-547.43	-547.55	-547.54	-547.43	-547.50	-547.58
0.841 quantile	-544.40	-546.06	-545.06	-543.23	-545.54	-546.03

Testování genetického algoritmu



Testování genetického algoritmu

Ocenasek/Gen	**WEC**	**RT**	**B**	**RF**	**SGB**	**ABR2**
Mean	1.78	1.34	0.92	0.96	0.87	0.91
Deviation	0.33	0.31	0.25	0.24	0.18	0.23
Median	1.79	1.34	0.91	0.92	0.87	0.89
0.159 quantile	1.47	1.02	0.69	0.73	0.70	0.70
0.841 quantile	2.14	1.64	1.21	1.26	1.06	1.18

Závěr

- Dotazy
- Připomínky