České vysoké učení technické v Praze

Fakulta elektrotechnická

Katedra kybernetiky
Katedra počítačů

Vytěžování dat – přednáška 11

RBFN a nové trendy vytěžování

Osnova přednášky

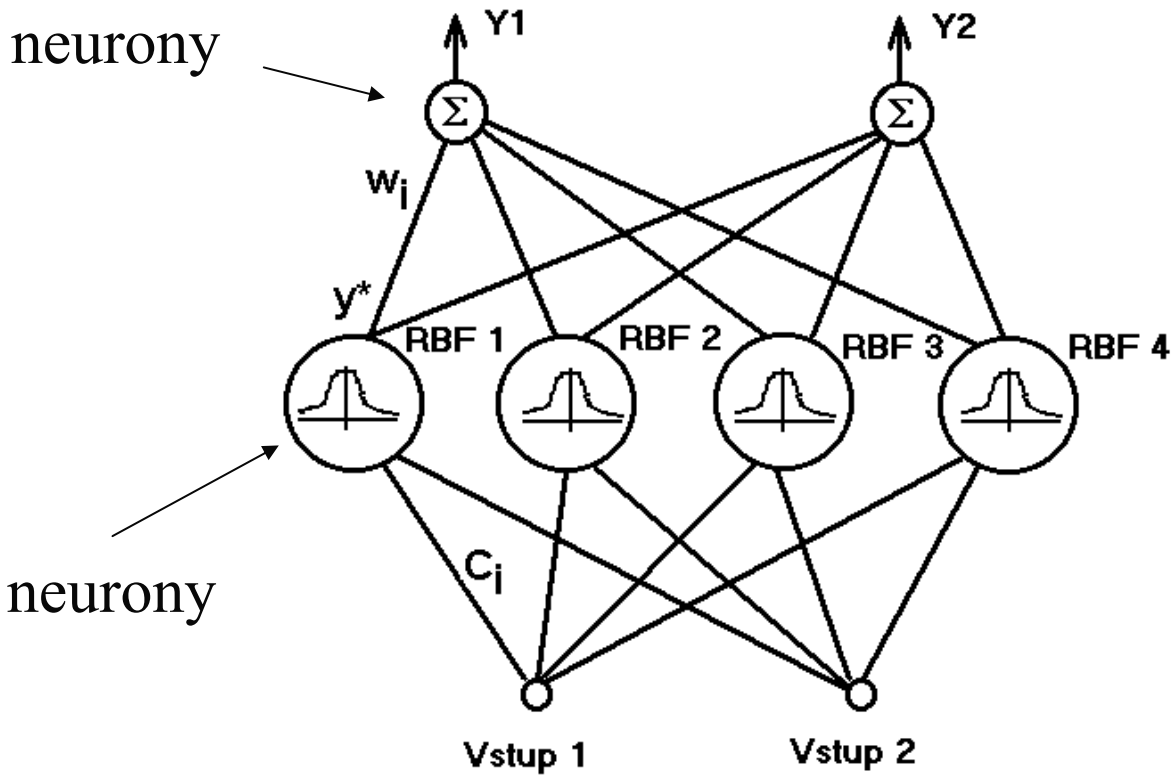
- Radial Basis Function Network (RBFN)
  - Architektura sítě
  - Učení
  - Regrese a klasifikace s RBFN
- Automatické těžení znalostí z dat
  - Trendy
  - Automatizace předzpracování dat
  - Automatizace vytěžování dat
  - Automatizace extrakce znalostí

RBFN

- Radial Basis Function Network
- 1988, Bromhead, Lowe
- Neuronová síť
- Učení s učitelem
- Použití:
  - Klasifikace
  - Regrese

- Hlavní rozdíl oproti MLP – lokální jednotky (vysvětlíme dále)

Architektura RBF sítě



Jak vypadá „sféra vlivu“

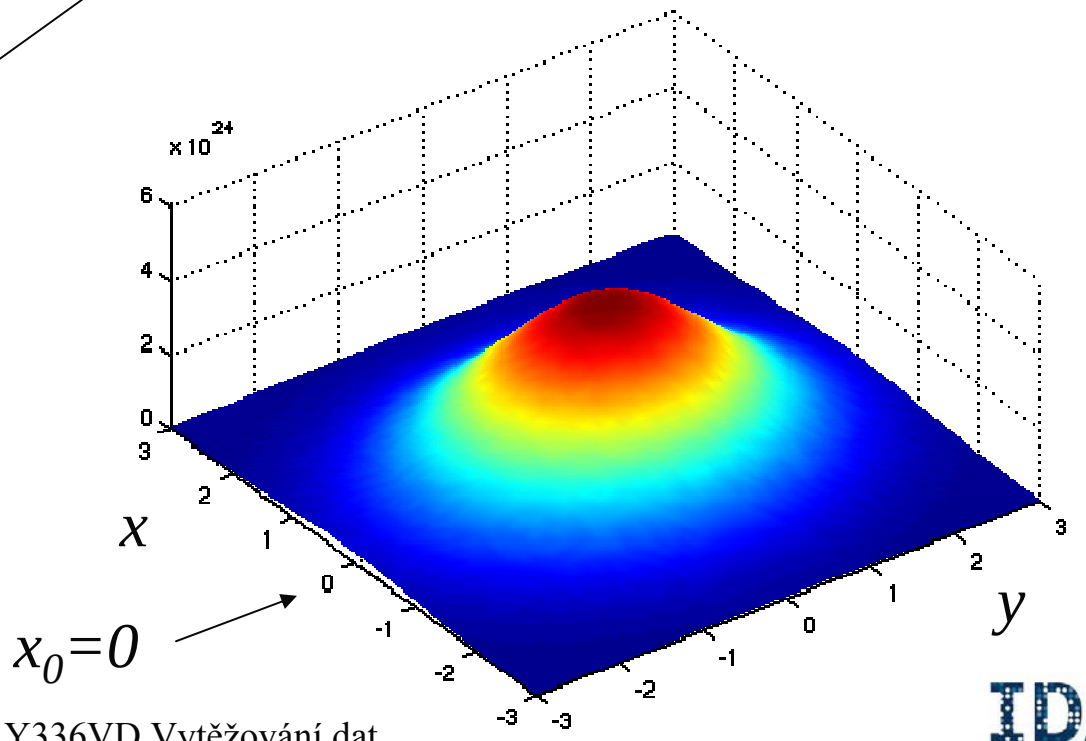
- Většinou gaussovská funkce (kernel)

$$f(x,y) = Ae^{-\left(\frac{(x-x_0)^2}{2\sigma_x^2}+\frac{(y-y_0)^2}{2\sigma_y^2}\right)}$$

posuv

amplituda

rozptyl

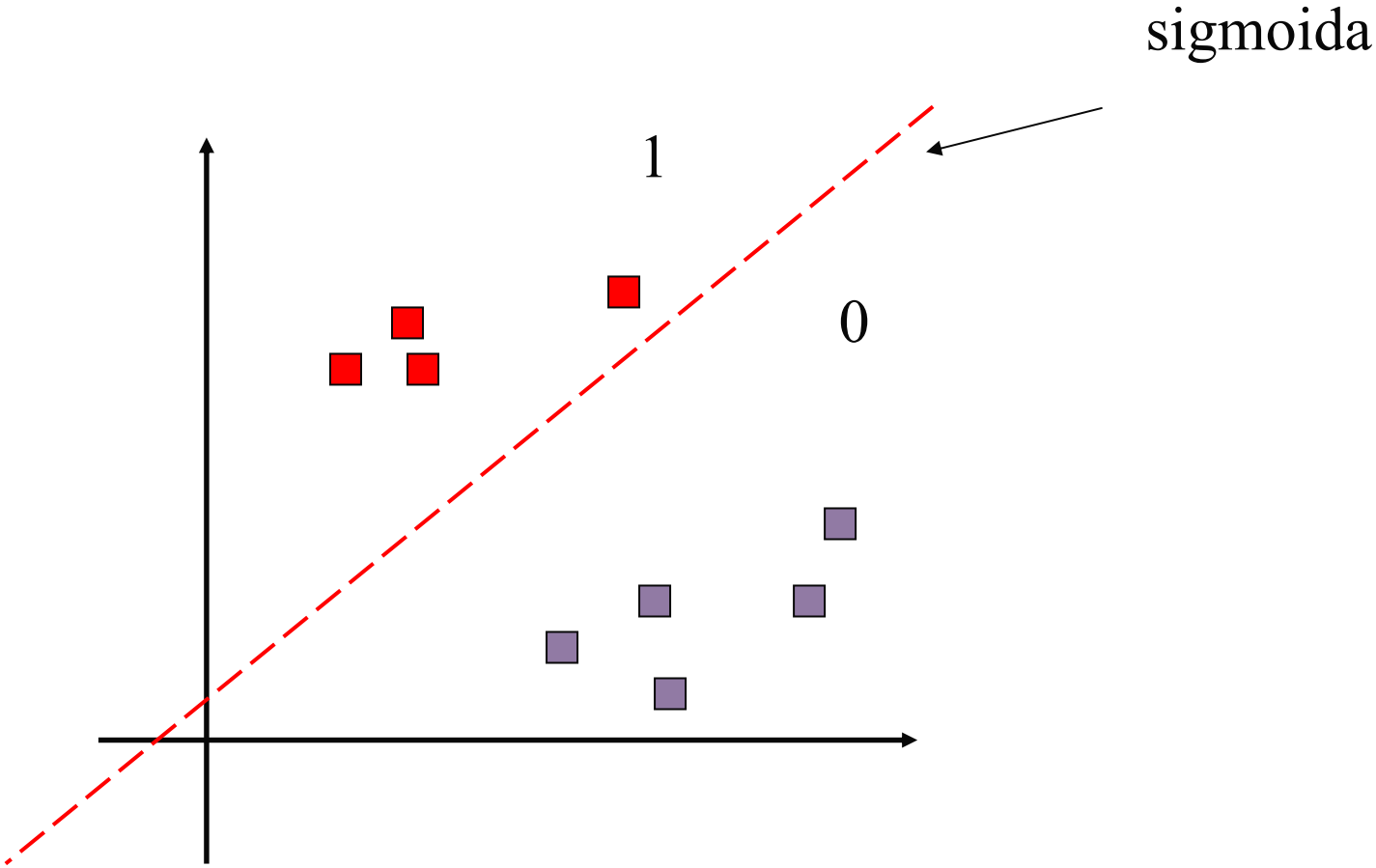


$x_0=0$

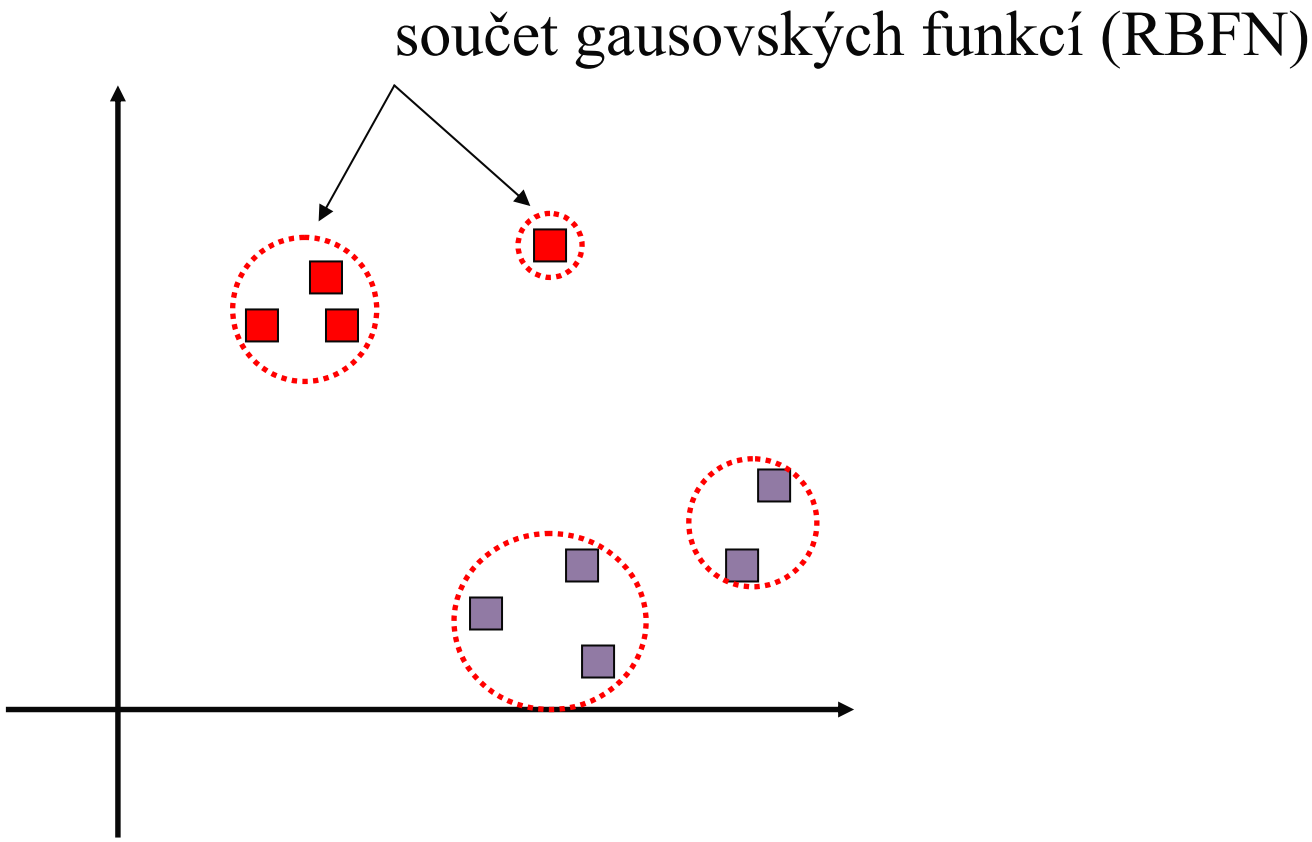
Lokální jednotky

- Co to znamená?
  - pokrývají jen část definičního oboru
  - jsou nenulové jen v jistém úseku
- Globální versus lokální jednotky:
  - gausovská funkce – lokální
  - sigmoida – globální
  - lineární funkce – globální
  - polynom – globální, ale ve speciálních případech může fungovat jako lokální

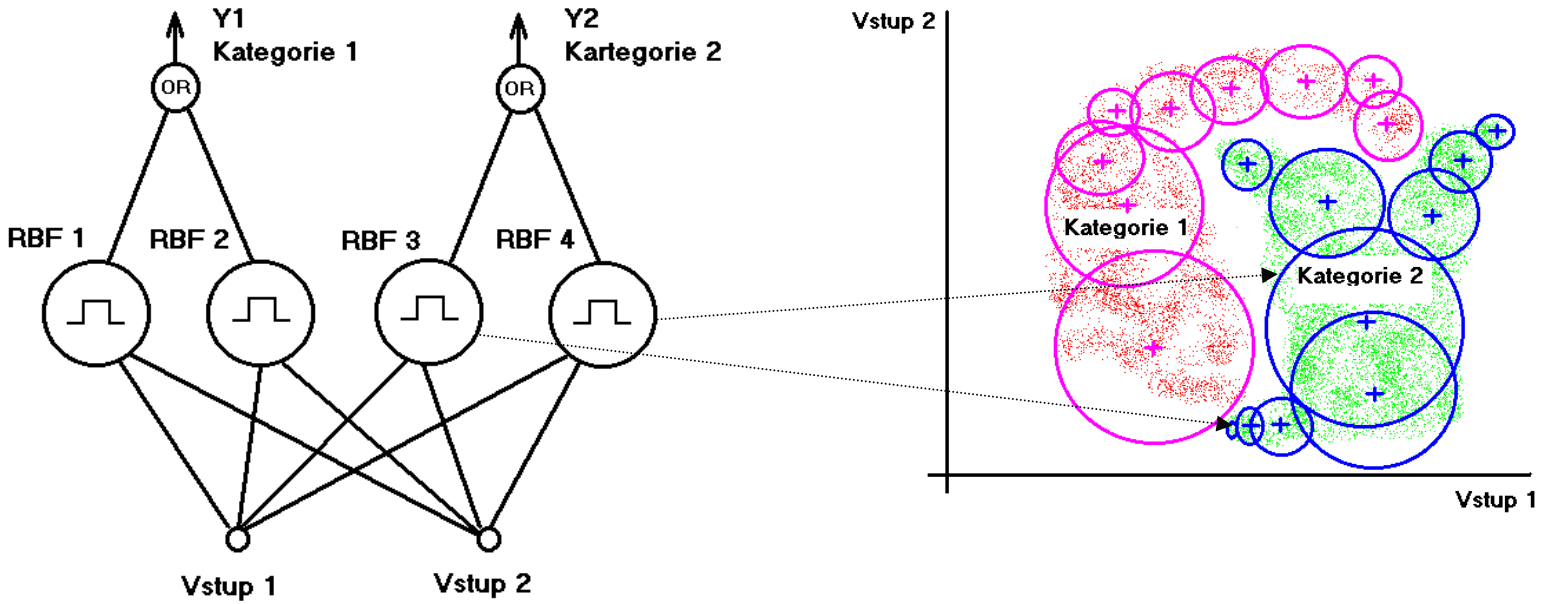
Klasifikace pomocí globálních jednotek



Klasifikace pomocí lokálních jednotek



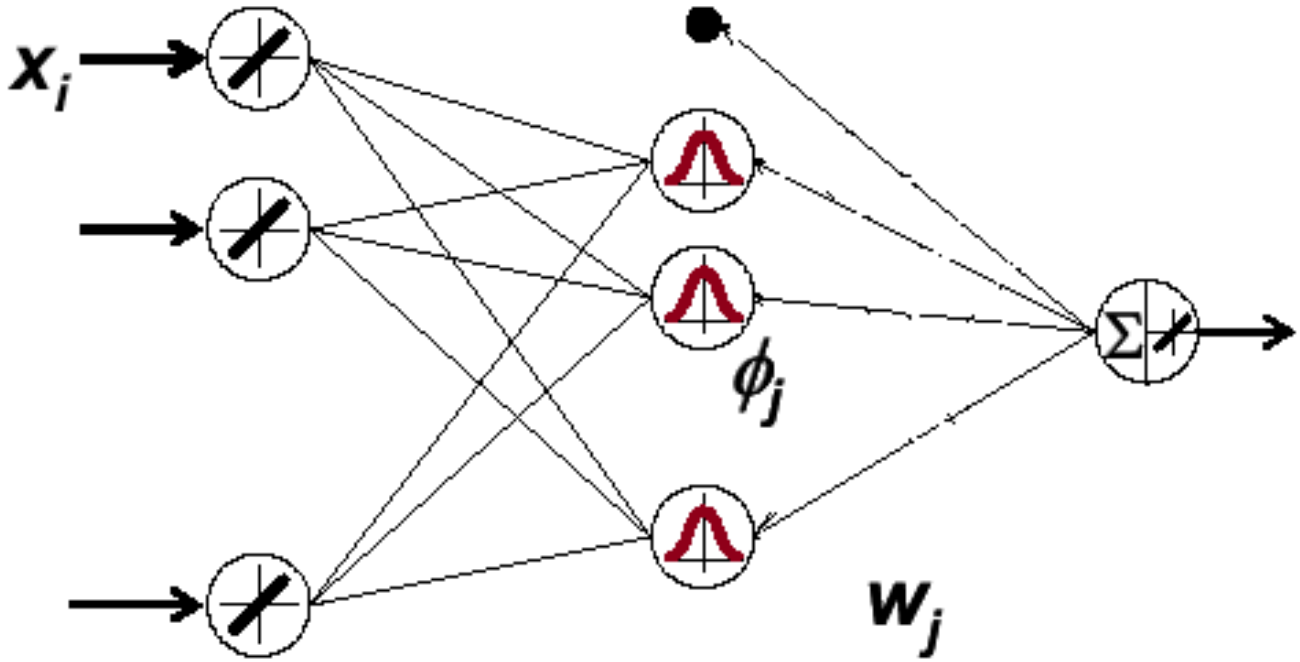
RBFN jako klasifikátor



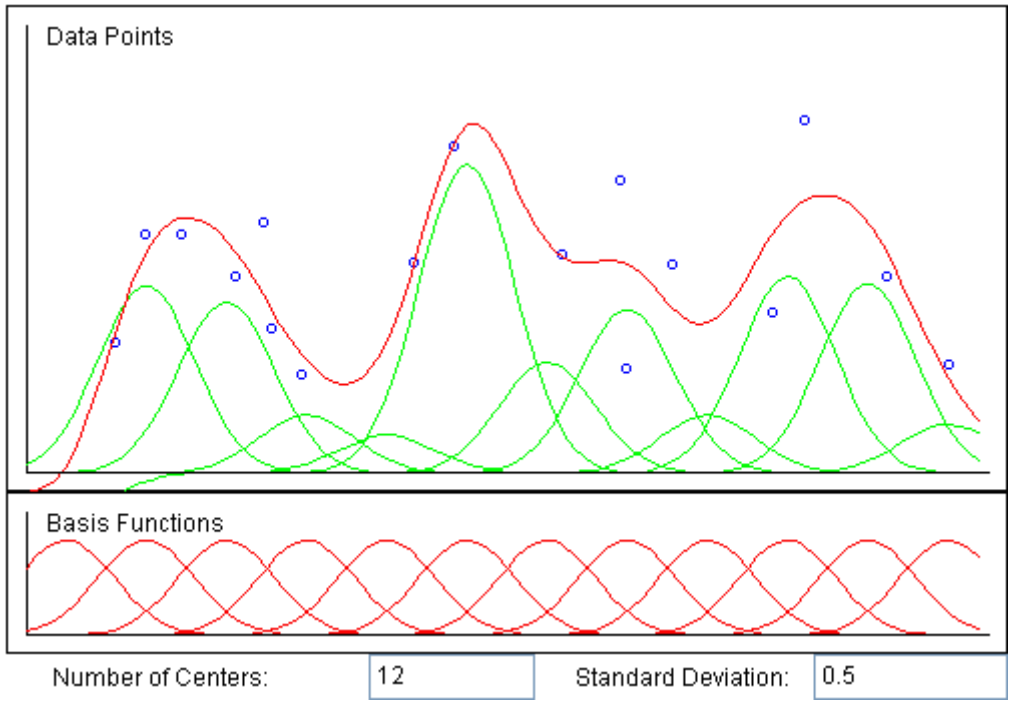
Každý neuron ve vnitřní vrstvě má „sféru vlivu“

Ty se ve výstupní vrstvě vážení sčítají pro každou třídu zvlášť

RBFN pro aproximaci (regrese)



RBF síť jako univerzální aproximátor



http://diwww.epfl.ch/mantra/tutorial/english/rbf/html/

Neurony RBF sítě

- **Skrytá vrstva**,
  - vnitřní potenciál

$$\varphi = \sqrt{\sum_{i=1}^{n} (x_i - c_i)^2}$$

  - lokální nelineární aktivační funkce

$y = f(\phi)$, např. gaussovská

- **výstupní vrstva**,
  - lineární přenosová funkce (vážený součet)

$$y = \sum_{i=1}^{n} w_i y_i^*$$

Diskuse architektury

- RBF neurony:
  - vnitřní potenciál je mírou vzdálenosti vstupního vektoru a **středu** (reprezentovaného vahami neuronu),
  - aktivační funkce vymezuje **sféru vlivu**.
- Výstupní neurony:
  - nasčítávají přírůstky, tak aby požadovaná aproximace byla co nejpřesnější.

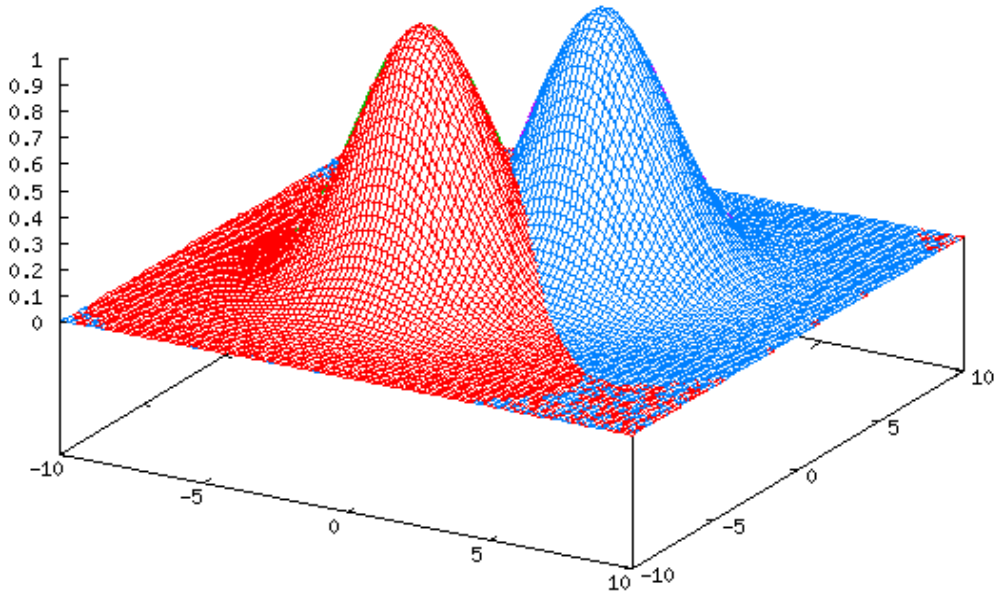
Sféra vlivu

- Hyperkoule se středem *C* a poloměrem *R*,
- RBFN používá pro její určení Eukleidovskou metriku,
- prototyp reprezentuje jistou podmnožinu vstupních dat ve tvaru shluku,

Sféra vlivu - určení

- Nejčastěji se používá Gaussova funkce známá ze statistiky.
  - Pokud je vstupní vektor totožný s prototypem (tj. $\varphi = 0$), nabývá tato funkce maxima, které dosahuje hodnoty jedna. To je také maximální hodnota aktivity neuronu.
- Se zvětšující se vzdáleností od prototypu aktivita neuronu klesá. Parametr σ, jenž je analogií rozptylu normálního rozdělení, určuje strmost aktivační funkce.

Sféra vlivu - geometrická představa



Diskuse

- Gaussova funkce vyjadřuje míru příslušnosti vzoru ke středu.
  - Je-li výstup neuronu blízký jedničce, pak je také vzor velmi podobný středu.

- Podobnost vyhodnocujeme pomocí metrik, které už důvěrně známe

MLP vs RBFN

Globální plocha	Lokální oblasti
Méně neuronů	Většinou více neuronů
Rychlá vybavovací fáze	Pomalejší vybavování
Pomalá učicí fáze	Rychlé učení

Vhodná data pro MLP a RBFN?

Učení RBF neuronových sítí

- Připomenutí:
  - jedná se o učení s učitelem, existují tedy dvojice
    - vzor x kategorie (klasifikátor),
    - argument funkce x funkční hodnota (aproximátor).
- Dvě fáze učení:
  - učení prototypů,
  - učení výstupních neuronů.

Učení středů I

- Předem odhadneme počet shluků ve vstupních datech,
- definujeme funkci příslušnosti *m* vzoru ke shluku,
- odhadneme souřadnice všech *p* vektorů C_p, které jsou středy shluků.

Učení prototypů I - pokračování

Kroky K-means algoritmu:

. Náhodně inicializuj středy RBF neuronů *C*.

. Vypočítej *m*() pro všechny vzory z trénovací množiny.

. Vypočítej nové středy *C* jako průměr všech vzorů, které náležely ke středu *k* podle funkce příslušnosti.

. Ukonči, jestliže se *m*() nemění, jinak pokračuj bodem 2

Učení prototypů II - pokračování

Kroky adaptivního K-means algoritmu:

. Náhodně inicializuj středy RBF neuronů C.

. Přečti vzor X.

. Urči k němu nejbližší nejbližší střed a změň jeho polohu podle pravidla:

$$\overline{C}_k^{(t+1)} = \overline{C}_k^t + \eta(\overline{X}^{(t)} - \overline{C}_k^t)$$

kde η je rychlost adaptace, která se postupně snižuje s počtem iterací.

. Ukonči, pokud $\eta = 0$ nebo po určitém počtu kroků. Jinak pokračuj bodem 2

Učení prototypů III

• Pokud neumíme odhadnout počet shluků v datech, vycházíme z jejich nulového počtu. Postup v tomto případě:

• Přečti vzor

• Vyhledej nejbližší shluk k. Pokud je vzdálenost menší než r, modifikuj střed shluku podle

$$\overline{C}_k^{(t+1)} = \overline{C}_k^t + \eta(\overline{X}^{(t)} - \overline{C}_k^t)$$

• Pokud je vzdálenost větší než r, založ nový střed na pozici vzoru X , tj. .

$$\overline{C}_k^{t+1} = \overline{X}^{(t)}$$

• Ukonči, pokud $\eta = 0$, nebo po určitém počtu kroků. Jinak pokračuj bodem 2.

Určení parametru σ

- Parametr σ je možno určit jako střední kvadratickou vzdálenost vzorů od středu shluku.

$$\sigma_k = \sqrt{\frac{1}{Q}\sum_{i=1}^{Q}\left\|\overline{C}_k - \overline{X}_q\right\|^2}$$

- kde X_q je q-tý vzor náležející ke shluku se středem C_k.

Učení vah výstupních neuronů

- Váhy ve výstupní vrstvě budeme opakovaně upravovat tak, abychom minimalizovali energetickou funkci:

$$\Delta \overline{w}^{(t)} = -\eta \nabla E^{(t)} = \eta (D^{(t)} - Y^{(t)}) Y^{*(t)}$$

- Vzpomínáte si? Co vám to připomíná?

Energetickou funkcí je v tomto případě

$$E = \frac{1}{2}\sum_{t=1}^{m}\sum_{i=1}^{n}\left(d_i^{(t)} - y_i^{(t)}\right)^2$$

Pro odvození vztahu pro úpravu vah jsme použili gradientní algoritmus.

Lze RBFN učit i jinak?

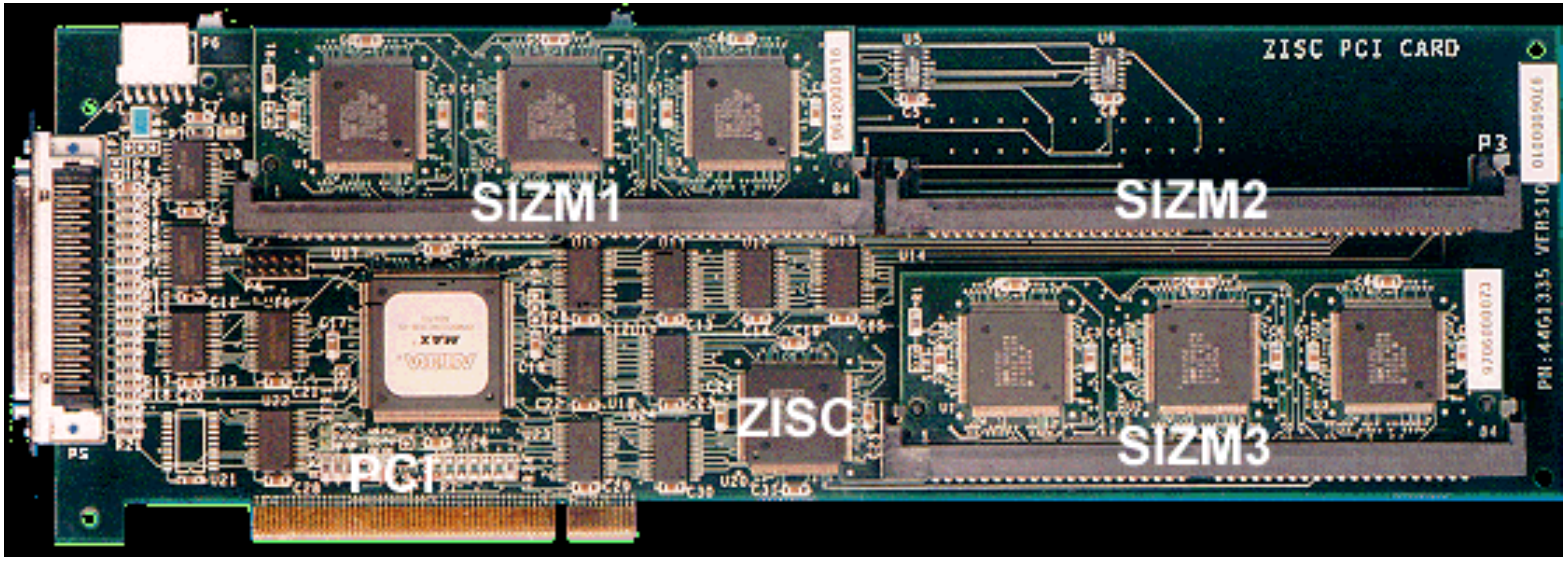
- Genetika!
- Jak na to?

Implementace sítě RBF – neuročip ZISC 36

- Neuročip ZISC (Zero Instruction Set Computer) vyrábí firma IBM.
- Jedná se o jednoúčelový procesor s pevně danou funkcí, který lze omezeně konfigurovat, ale nikoliv programovat.
- Číslo 36 v názvu udává počet neuronů implementovaných v jednom pouzdře.

Neuročip přes WEB

- http://axon.felk.cvut.cz/zisc/zisc.php



Automatické těžení znalostí z dat

Trendy:

- Je těžké se stát DM specialistou (a drahé si takového specialistu najmout)
- Přesto hodně firem potřebuje analyzovat data a vytěžit znalosti
- Řešením je specializovaný software, který uživatele odstíní od miliónů konfiguračních nastavení, kterým nerozumí, a přesto poskytne použitelný výsledek.

- Co musí takový software umět?

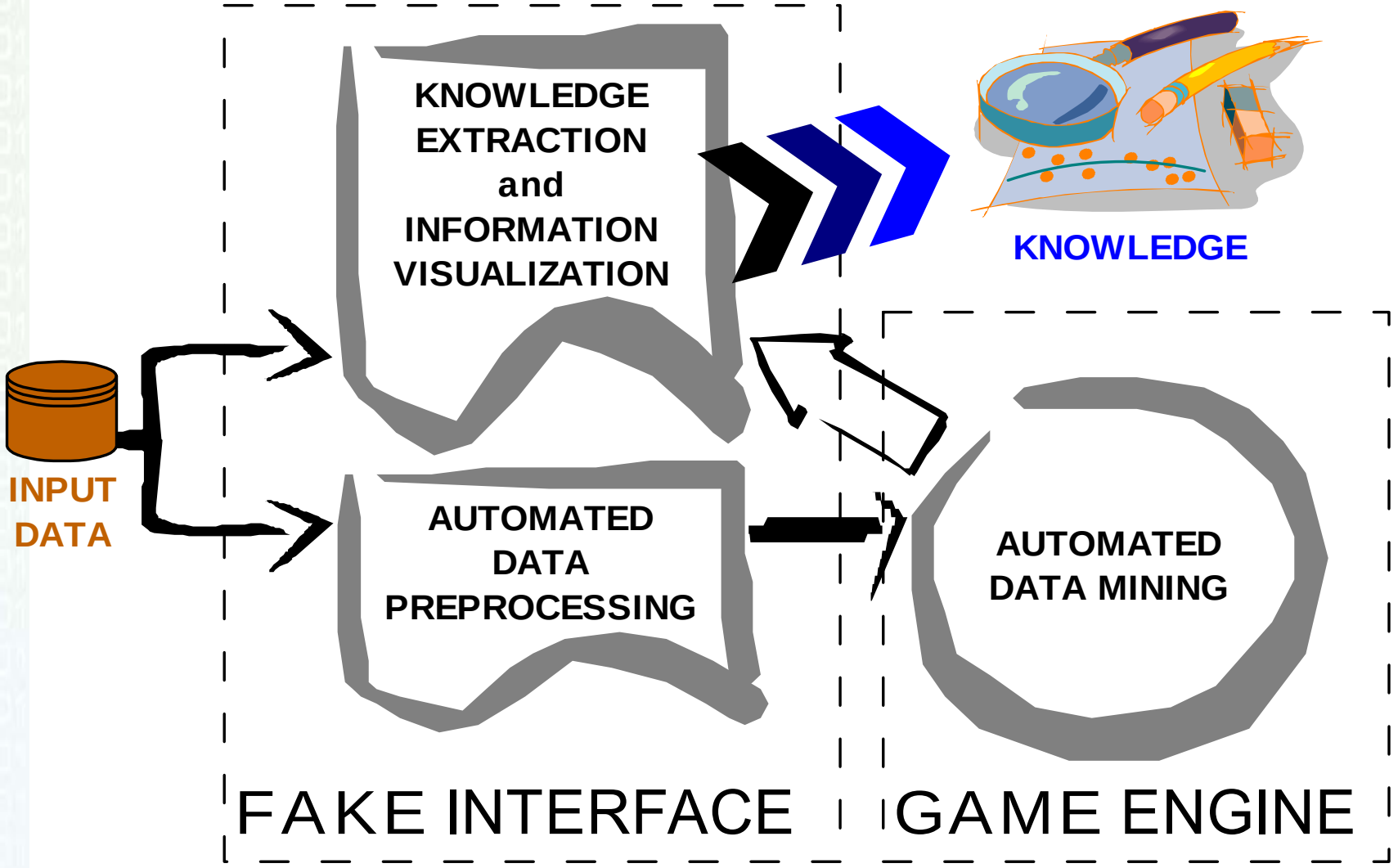
Automatizace předzpracování dat

- **Znalost následujících slajdů nebude předmětem zkoušky**
- Pamatujete si na „žlutý diamant" z minulé přednášky SPSS?

- Předzpracování dat jde obecně automatizovat velmi těžko – velká opatrnost nutná

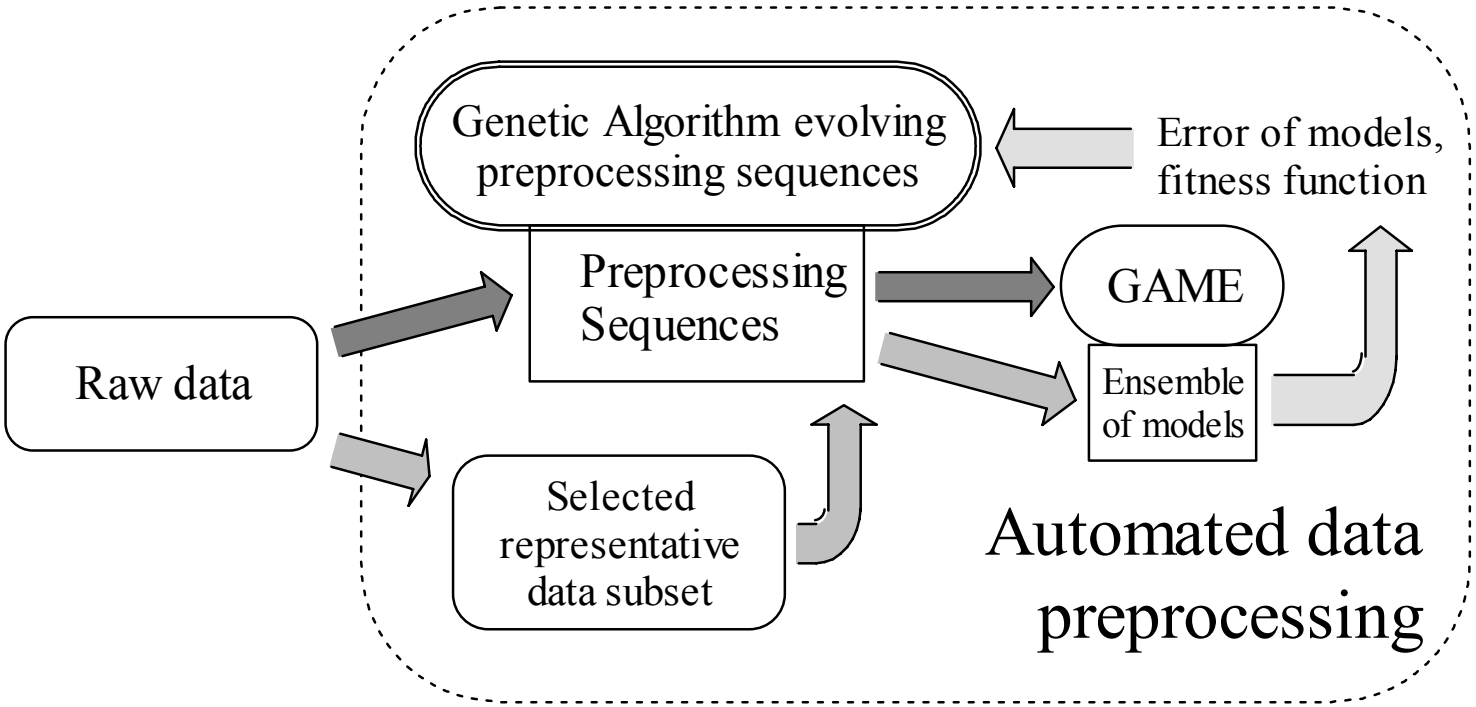
- Ukázka, jak se o to snažíme v naší výzkumné skupině:

FAKE GAME sofware

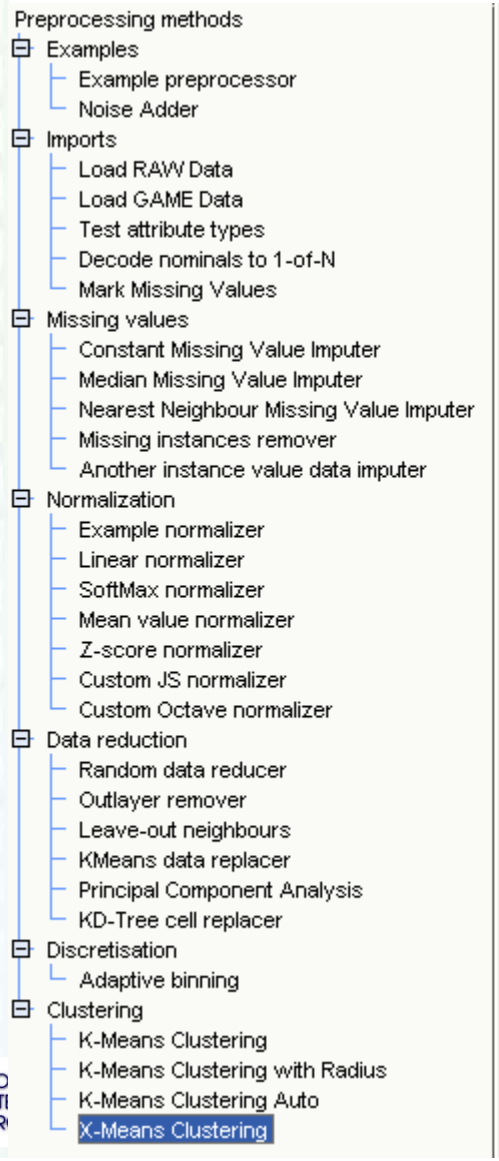


Automated data preprocessing

- Pro každý vstupní atribut vyšlechtíme genetickým algoritmem posloupnost předzpracovacích metod:

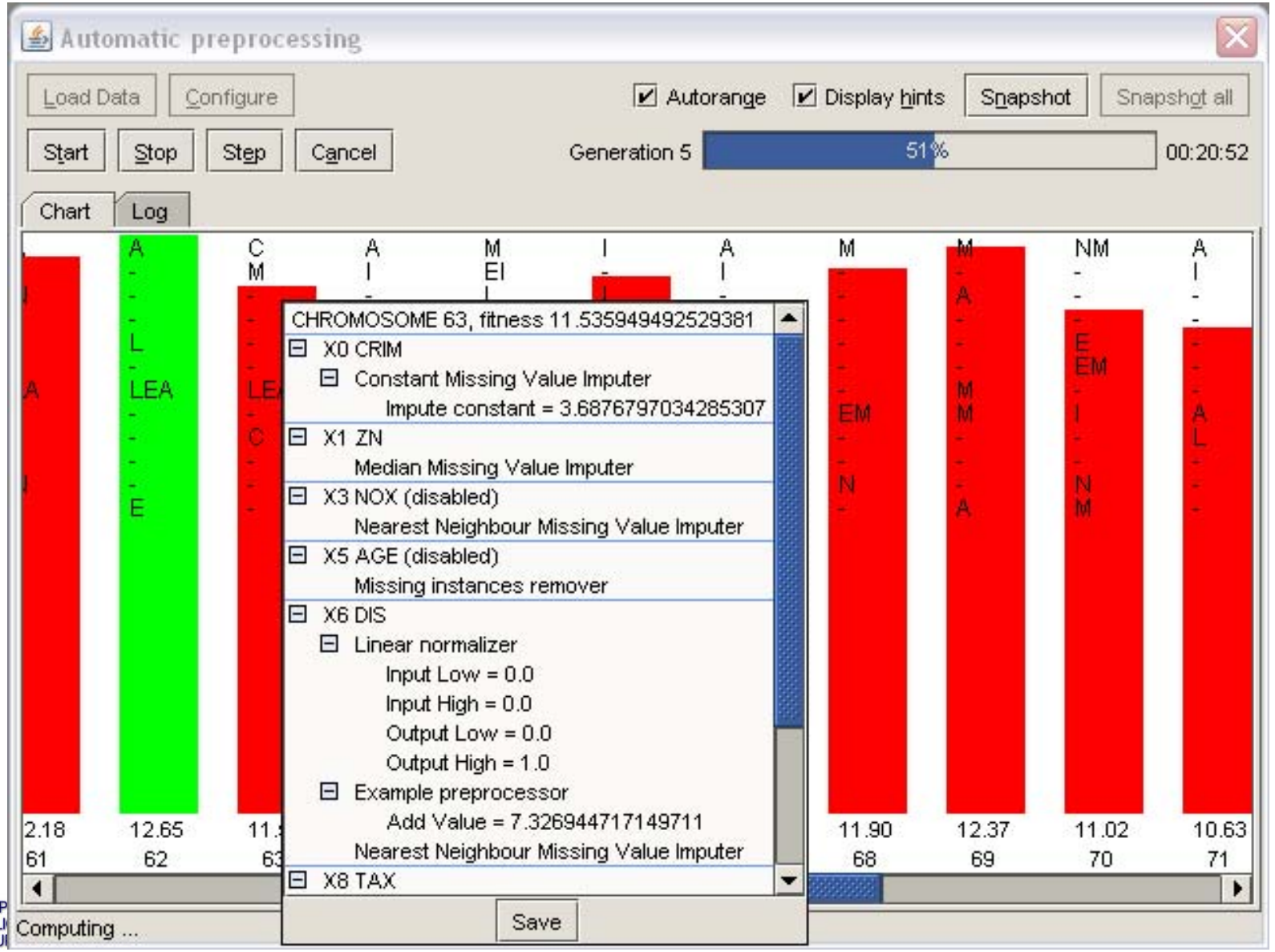


Metody které máme implementovány



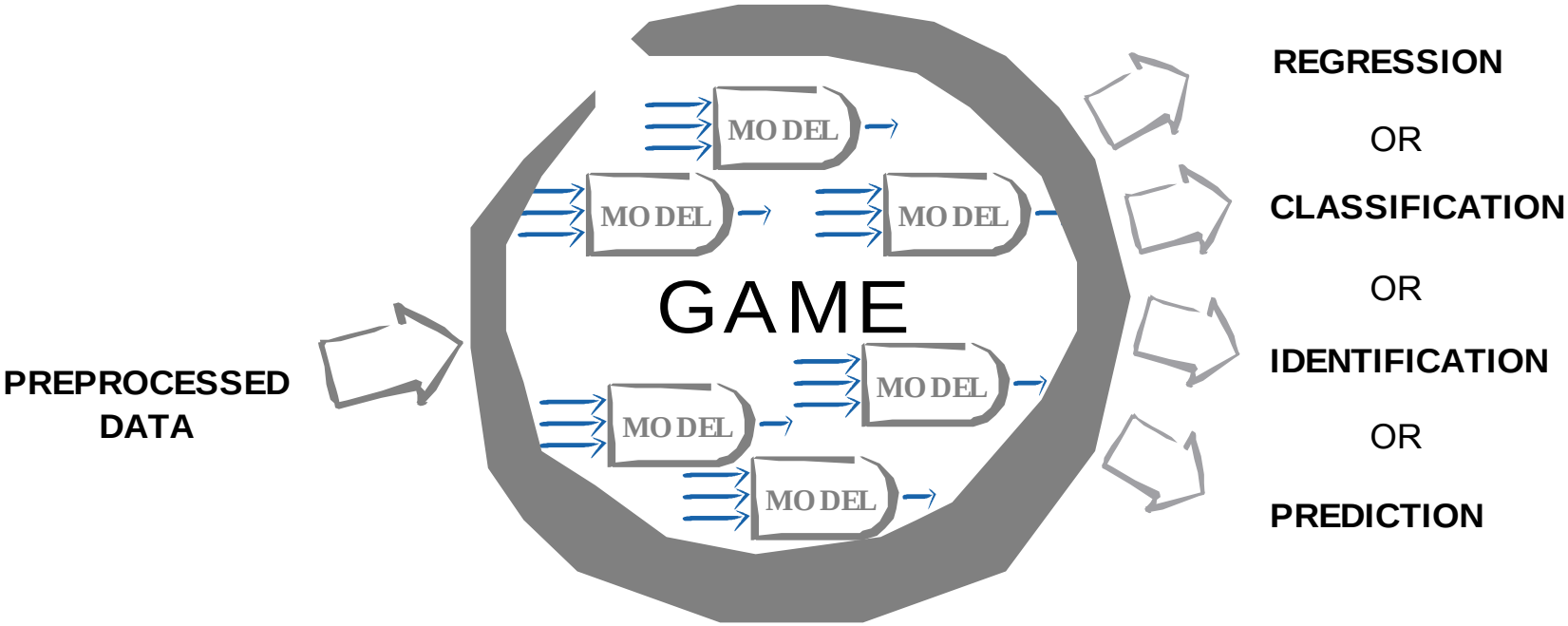
	sepal_length	sepal_width	petal_length	petal_width	Iris-setosa	Iris-versico...	Iris-virginica
Input/Output...	<Input attrib...	<Input attrib...	<Input attrib...	<Input attrib...	<Output attri...	<Output attri...	<Output attri...
Manually sel...	<Numeric ty...	<Numeric ty...	<Numeric ty...	<Numeric ty...	<Numeric ty...	<Numeric ty...	<Numeric ty...
Instance 0	5.1	3.5	1.4	0.2	1.0	0.0	0.0
Instance 1	4.9	3.0	1.4	0.2	1.0	0.0	0.0
Instance 2	4.7	3.2	1.3	0.2	1.0	0.0	0.0
Instance 3	4.6	3.1	1.5	0.2	1.0	0.0	0.0
Instance 4	5.0	3.6	1.4	0.2	1.0	0.0	0.0
Instance 5	5.4	3.9	1.7	0.4	1.0	0.0	0.0
Instance 6	4.6	3.4	1.4	0.3	1.0	0.0	0.0
Instance 7	5.0	3.4	1.5	0.2	1.0	0.0	0.0
Instance 8	4.4	2.9	1.4	0.2	1.0	0.0	0.0
Instance 9	4.9	3.1	1.5	0.1	1.0	0.0	0.0
Instance 10	5.4	3.7	1.5	0.2	1.0	0.0	0.0
Instance 11	4.8	3.4	1.6	0.2	1.0	0.0	0.0
Instance 12	4.8	3.0	1.4	0.1	1.0	0.0	0.0
Instance 13	4.3	3.0	1.1	0.1	1.0	0.0	0.0
Instance 14	5.8	4.0	1.2	0.2	1.0	0.0	0.0
Instance 15	5.7	4.4	1.5	0.4	1.0	0.0	0.0
Instance 16	5.4	3.9	1.3	0.4	1.0	0.0	0.0
Instance 17	5.1	3.5	1.4	0.3	1.0	0.0	0.0
Instance 18	5.7	3.8	1.7	0.3	1.0	0.0	0.0
Instance 19	5.1	3.8	1.5	0.3	1.0	0.0	0.0
Instance 20	5.4	3.4	1.7	0.2	1.0	0.0	0.0
Instance 21	5.1	3.7	1.5	0.4	1.0	0.0	0.0
Instance 22	4.6	3.6	1.0	0.2	1.0	0.0	0.0
Instance 23	5.1	3.3	1.7	0.5	1.0	0.0	0.0
Instance 24	4.8	3.4	1.9	0.2	1.0	0.0	0.0
Instance 25	5.0	3.0	1.6	0.2	1.0	0.0	0.0
Instance 26	5.0	3.4	1.6	0.4	1.0	0.0	0.0
Instance 27	5.2	3.5	1.5	0.2	1.0	0.0	0.0
Instance 28	5.2	3.4	1.4	0.2	1.0	0.0	0.0
Instance 29	4.7	3.2	1.6	0.2	1.0	0.0	0.0
Instance 30	4.8	3.1	1.6	0.2	1.0	0.0	0.0
Instance 31	5.4	3.4	1.5	0.4	1.0	0.0	0.0
Instance 32	5.2	4.1	1.5	0.1	1.0	0.0	0.0

Výstup genetického algoritmu

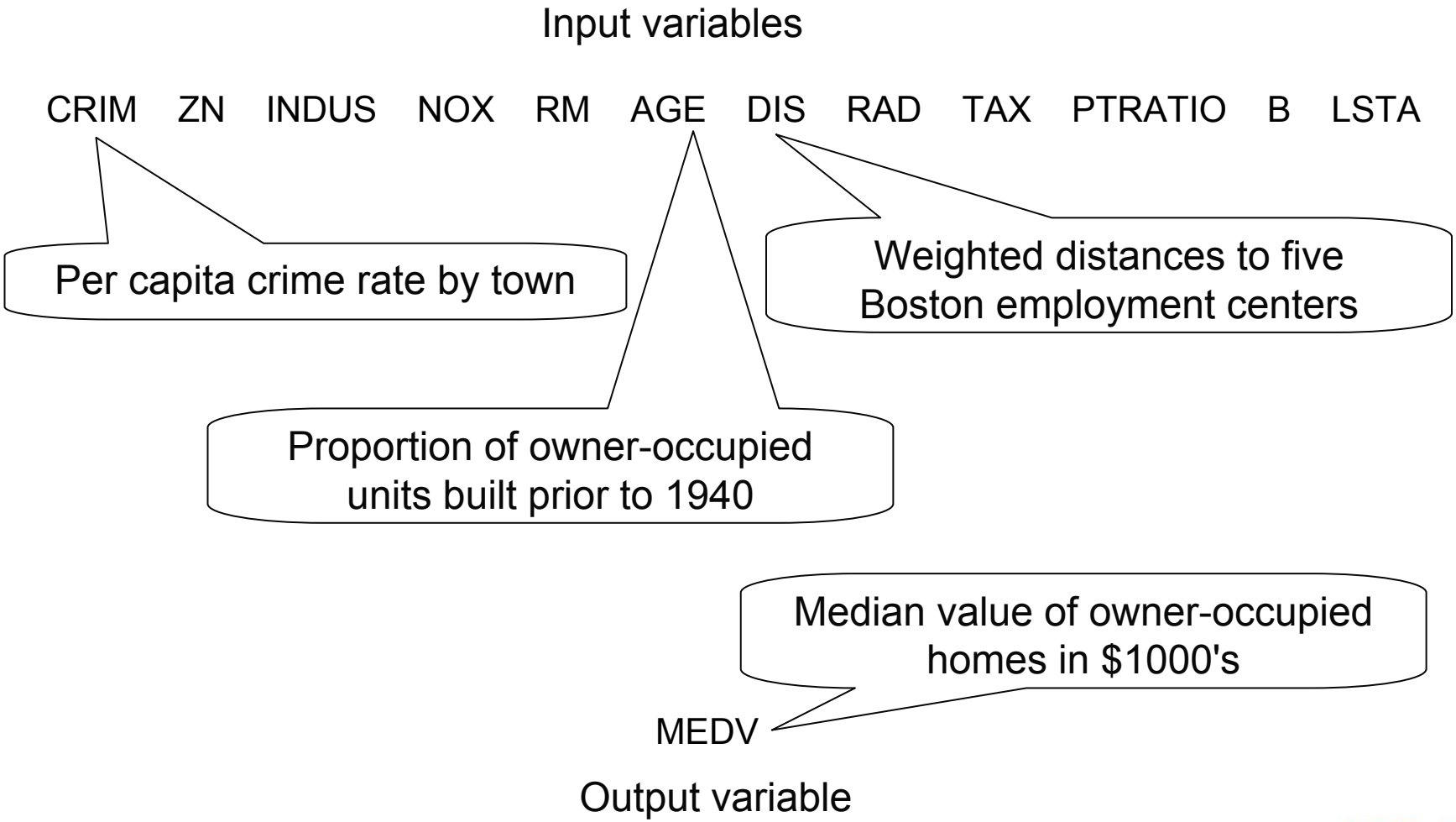


Automatizace vytěžování dat

- Algoritmy se musejí adaptovat na data



Příklad: Housing data



Housing data – records

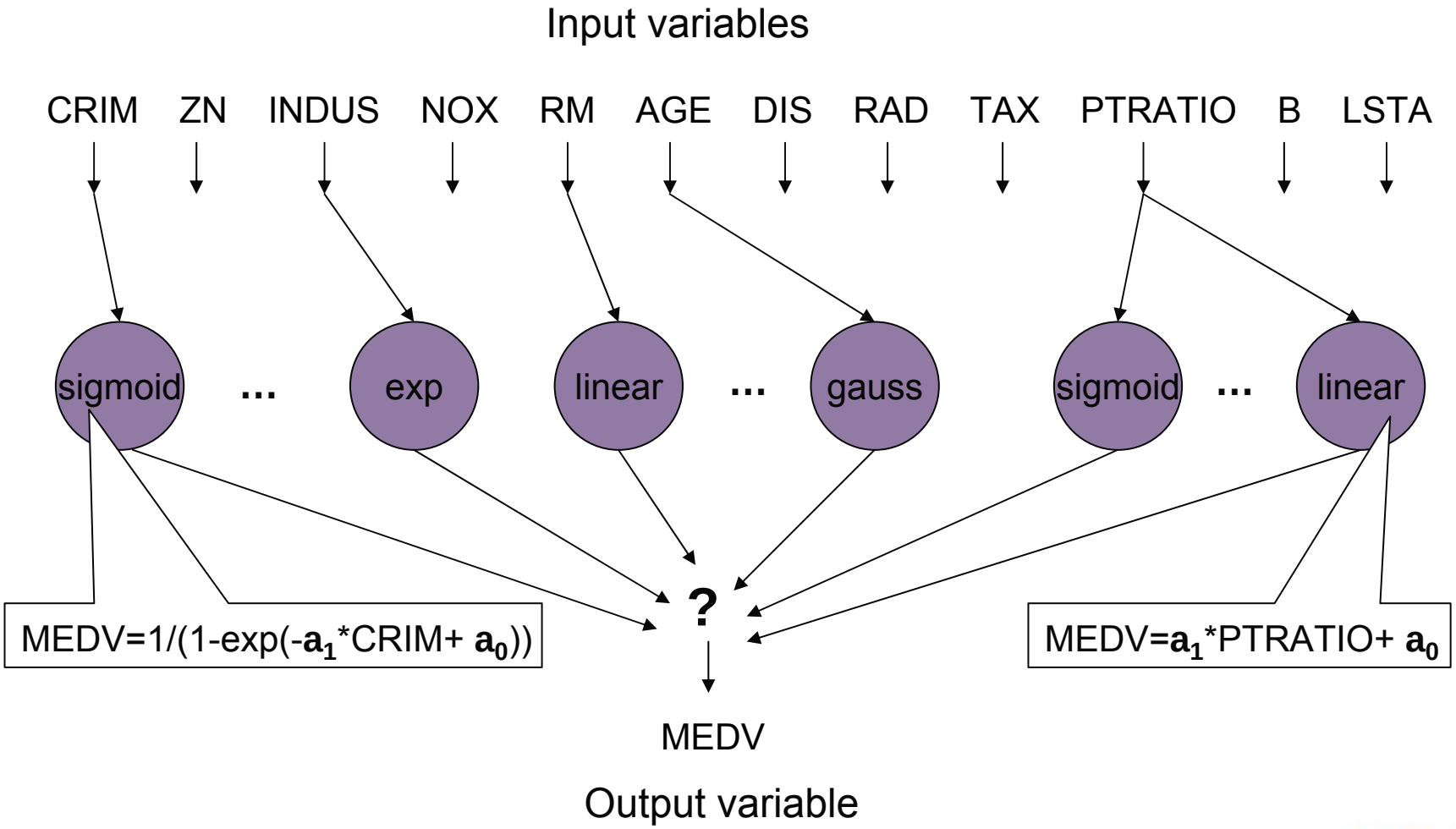
	Input variables										Output variable	
	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTA	MEDV
A	24	0.00632	18	2.31	53.8	6.575	65.2	4.09	1	296	15.3	396.9	4.98
A	21.6	0.02731	0	7.07	46.9	6.421	78.9	4.9671	2	242	17.8	396.9	9.14
A	…	…	…										

B = Training set … to adjust weights and coefficients of neurons

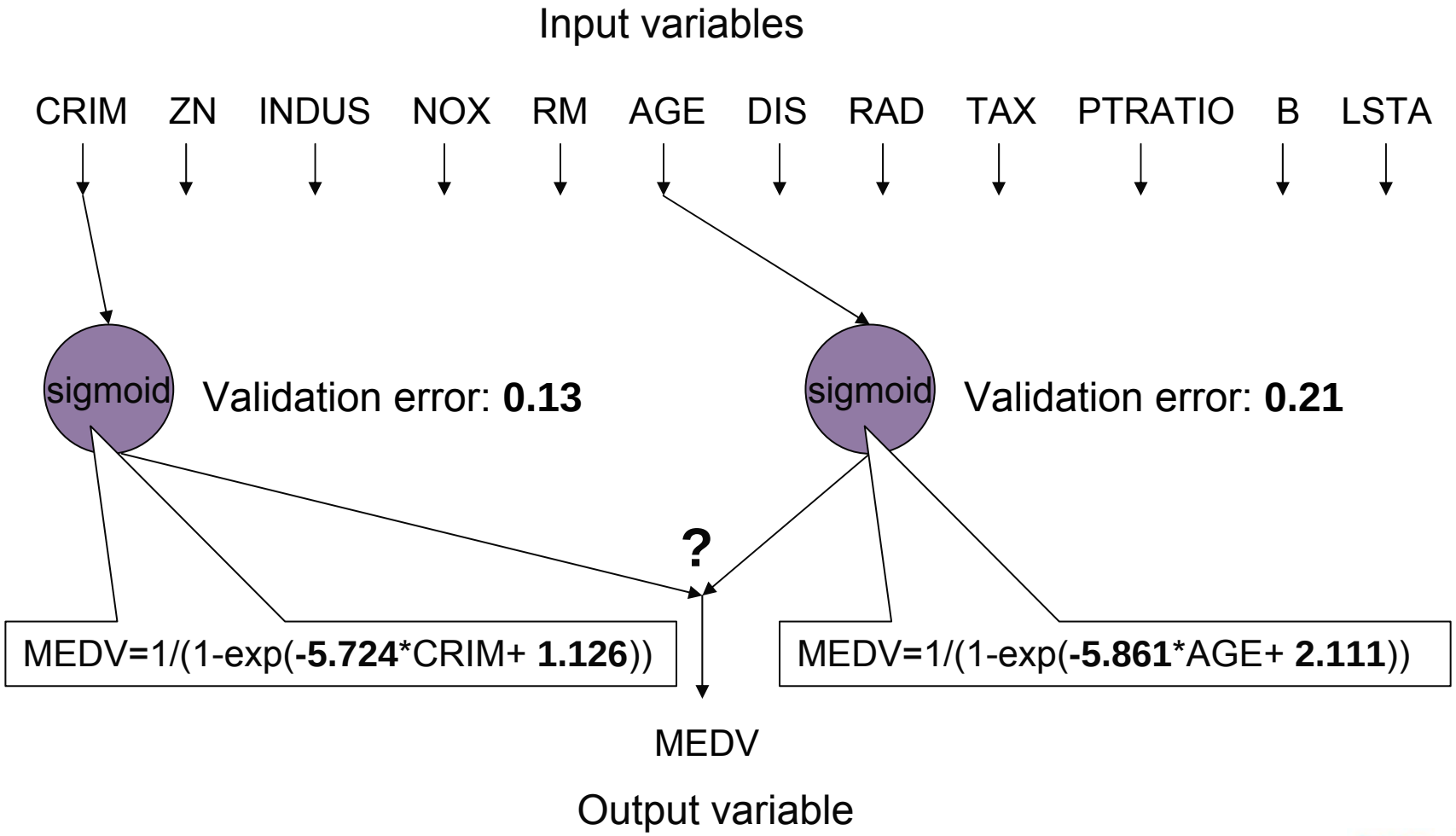
B = Validation set … to select neurons with the best generalization

C = Test set … not used during training

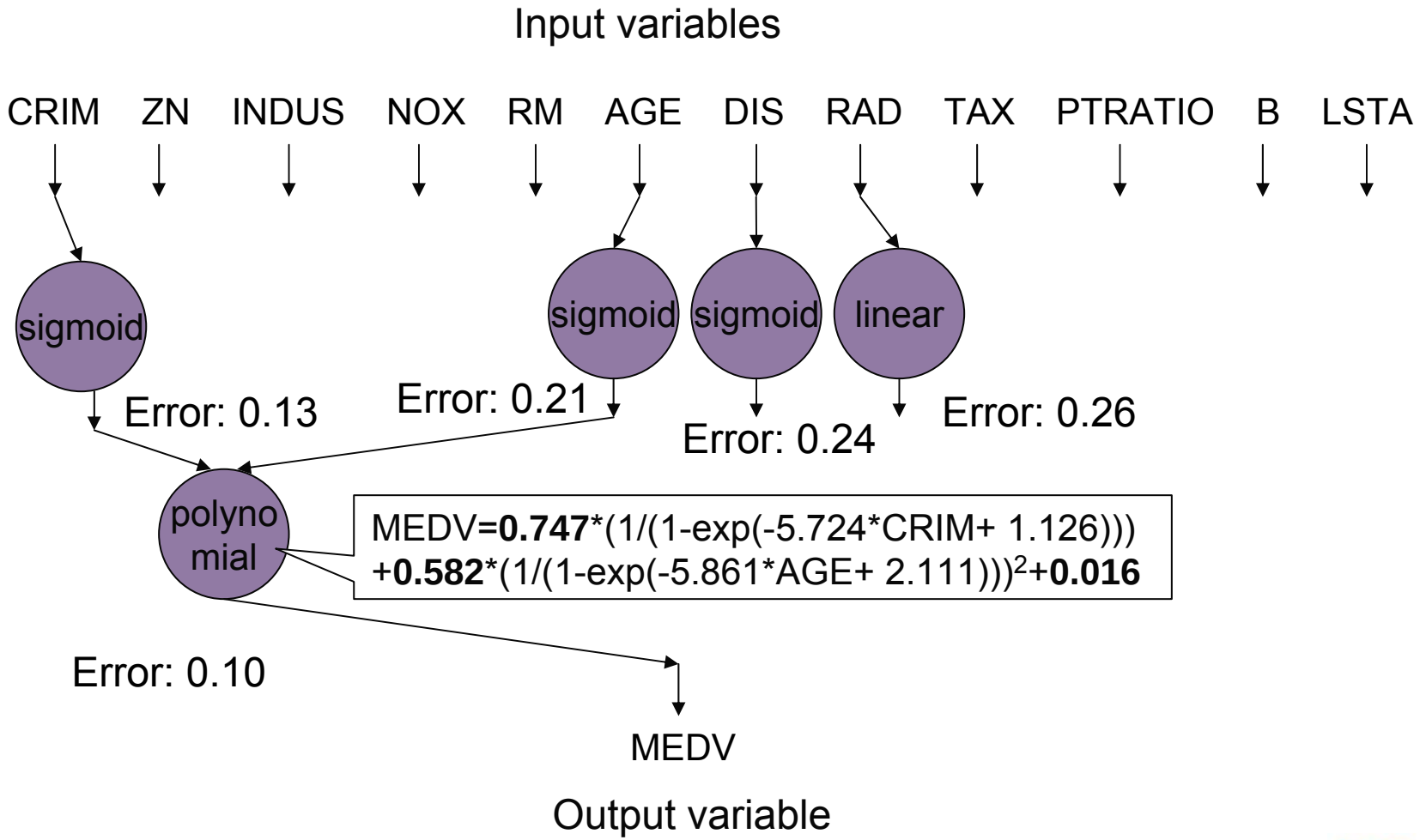
Housing data – inductive model



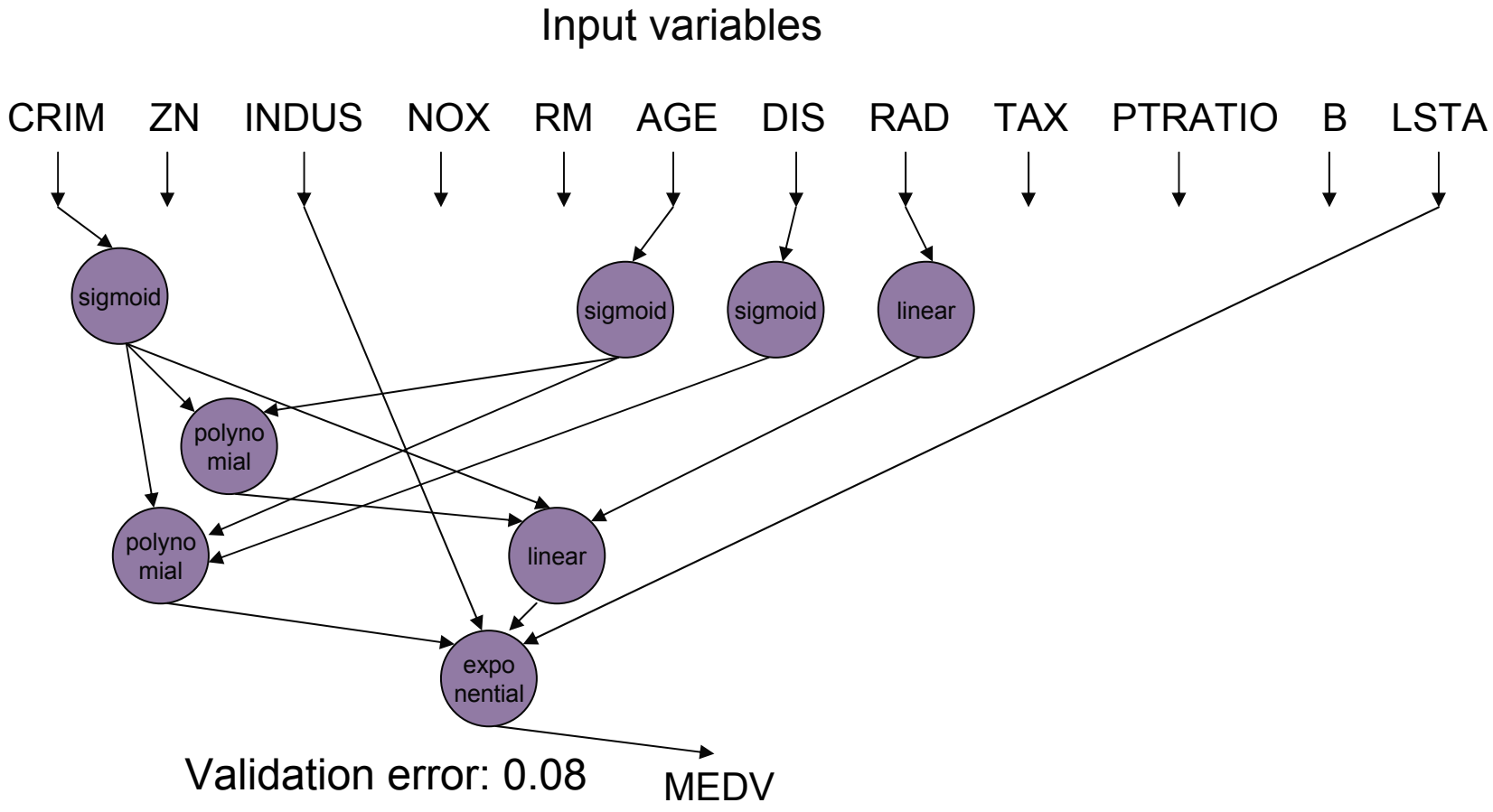
Housing data – inductive model



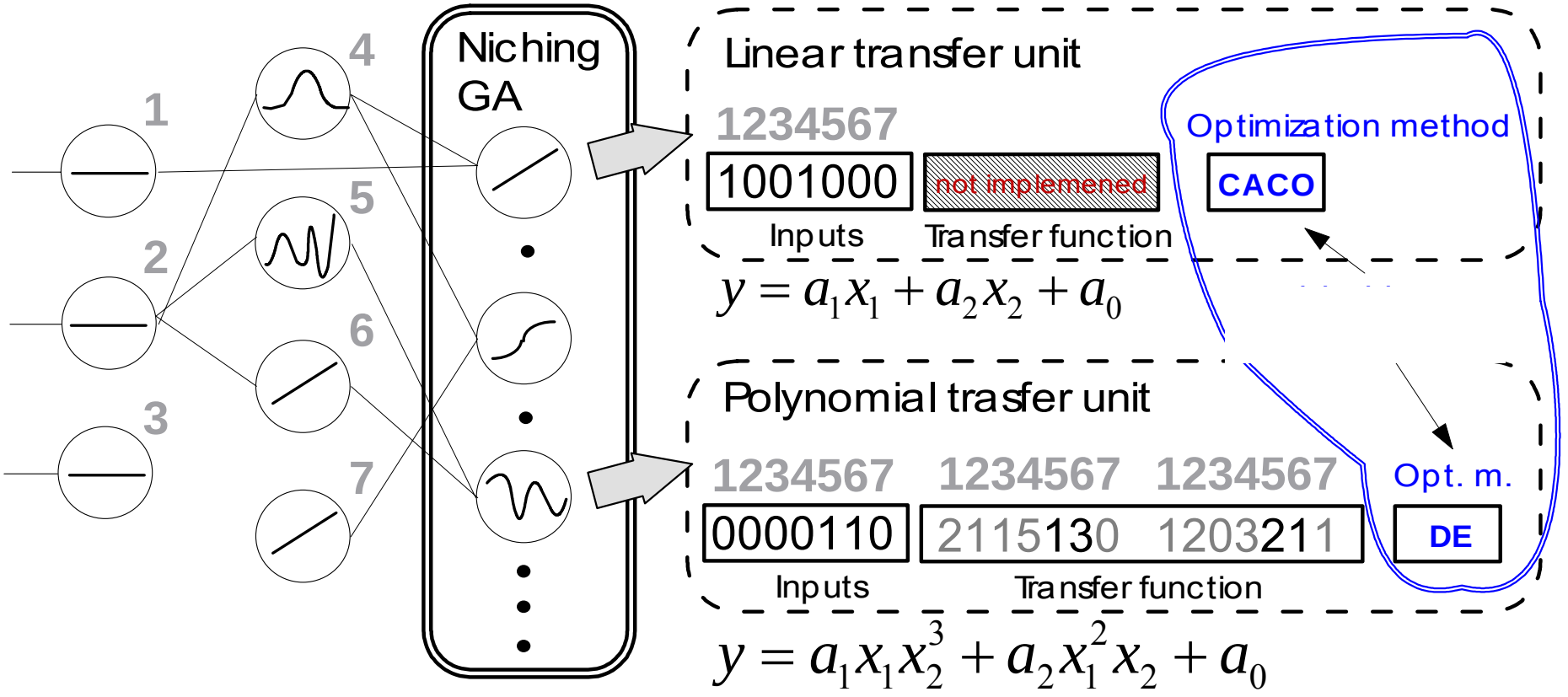
Housing data – inductive model



Housing data – inductive model

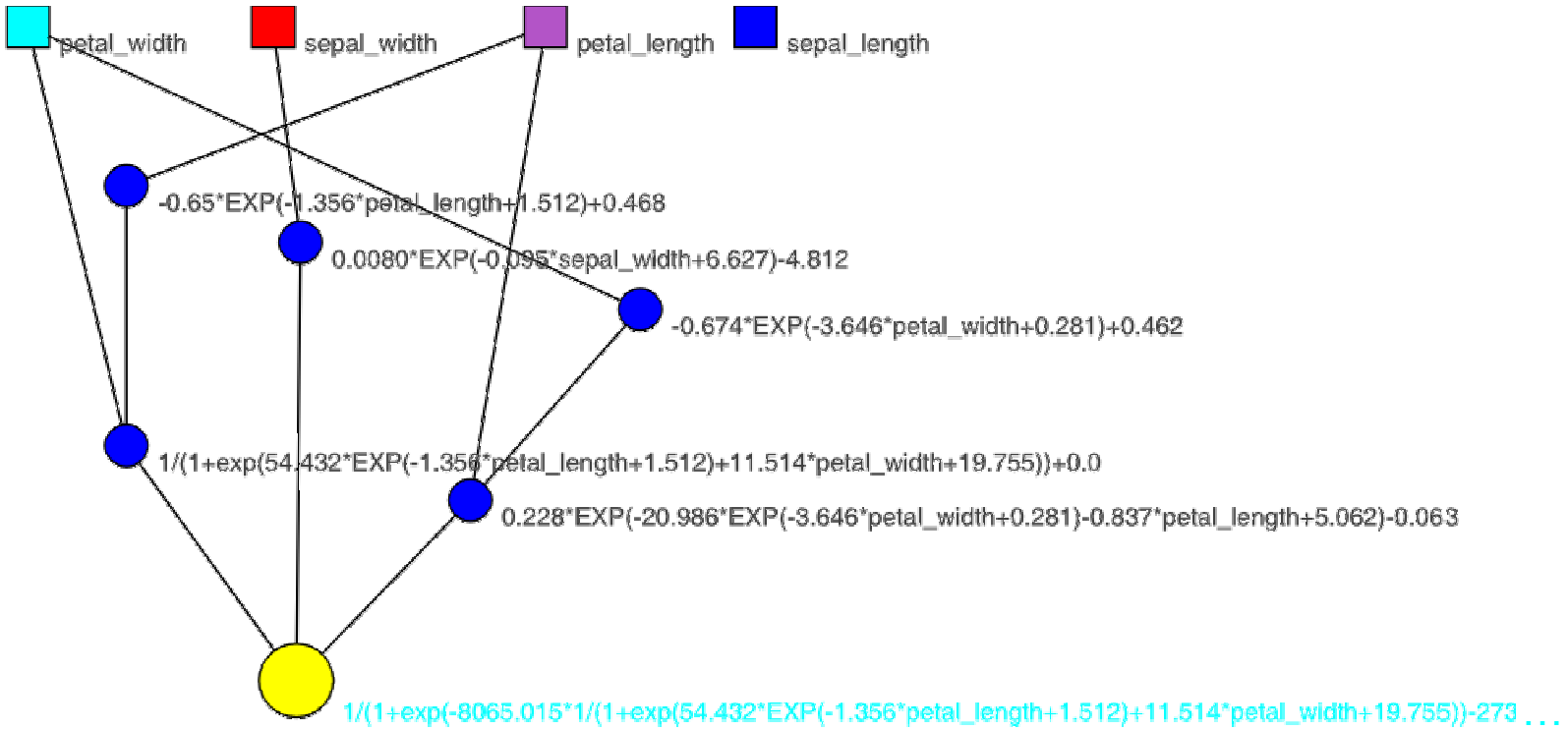


Modely jsou šlechtěny speciálním genetickým algoritmem



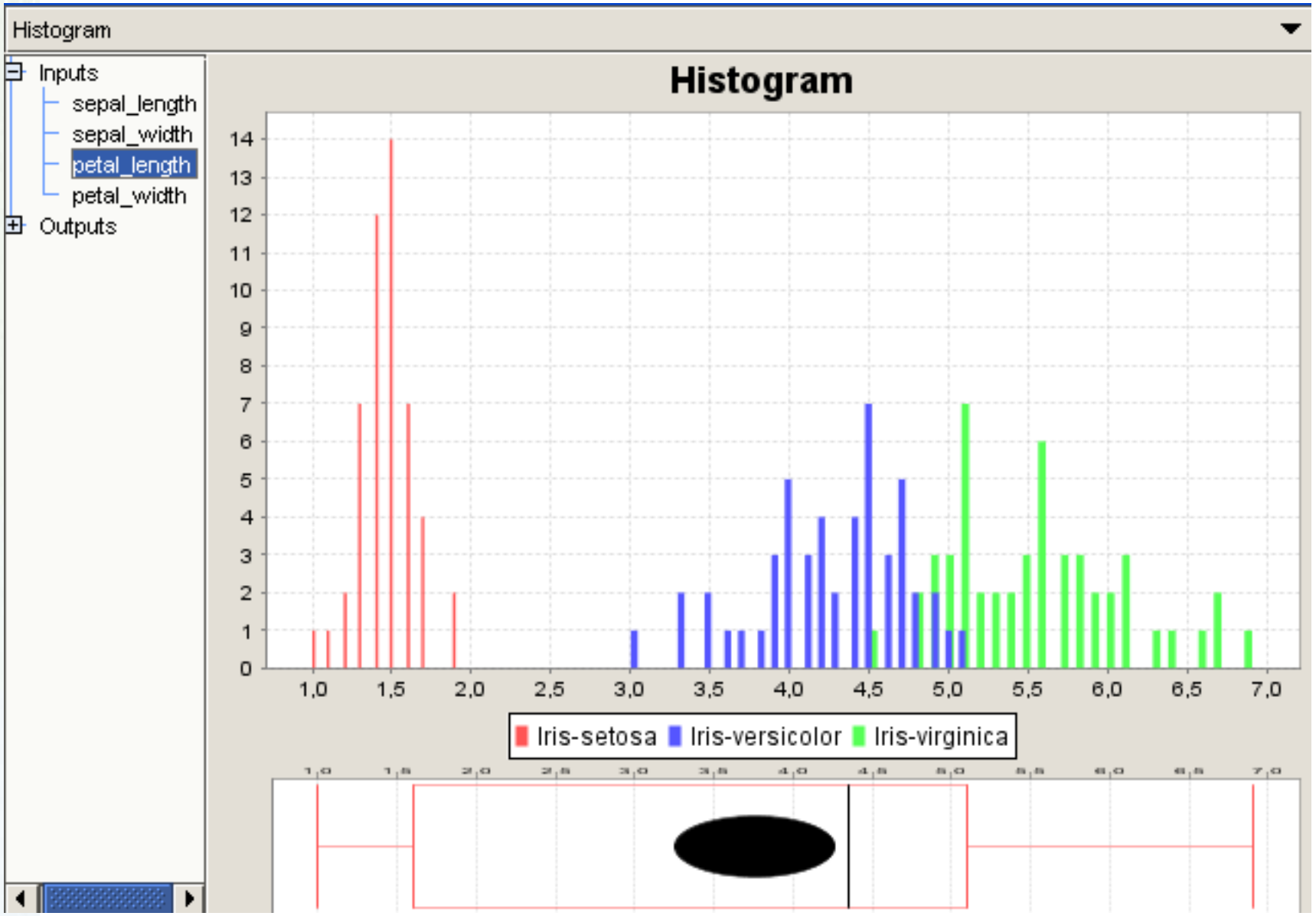
Fitness of unit: inverse of its error on the validation data set

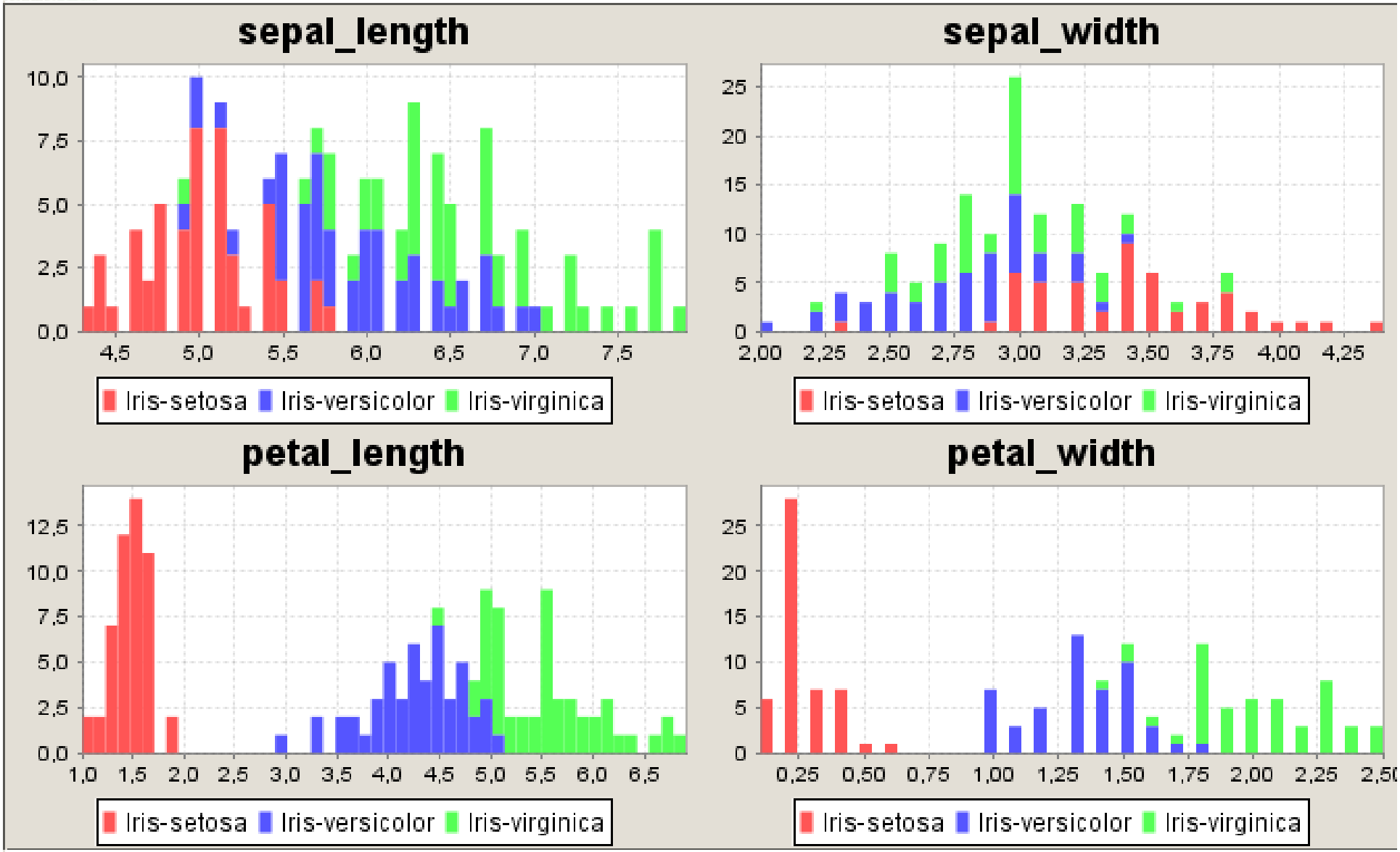
Naroste model a co s nim?



Automaticky extrahované informace

- Co všechno může být užitečné?
- Jak to udělat automaticky?

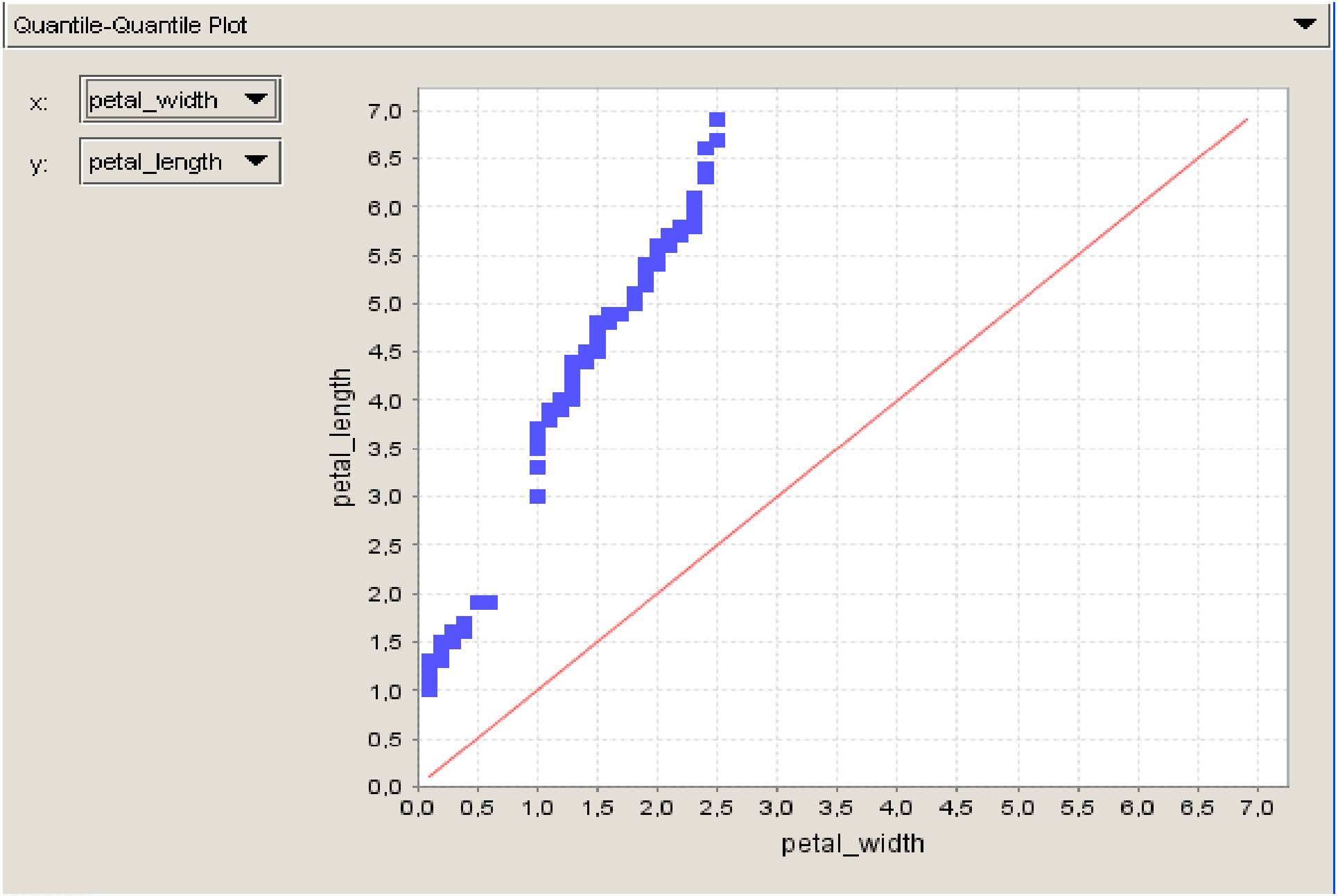

Histogram
Inputs
sepal_length
sepal_width
petal_length
petal_width
Outputs
Histogram
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
1,0
1,5
2,0
2,5
3,0
3,5
4,0
4,5
5,0
5,5
6,0
6,5
7,0
Iris-setosa
Iris-versicolor
Iris-virginica



50

- Inputs
- Outputs

Input	Min	25%	50%	75%	Max	Average	Variance
sepal_length	4.3	5.1	5.8	6.4	7.9	5.84333333...	0.68112222...
sepal_width	2.0	2.8	3.0	3.3	4.4	3.05399999...	0.18675066...
petal_length	1.0	1.5	4.3	5.1	6.9	3.75866666...	3.09242488...
petal_width	0.1	0.3	1.3	1.8	2.5	1.19866666...	0.57853155...


Quantile-Quantile Plot
x:
petal_width
y:
petal_length
7,0
6,5
6,0
5,5
5,0
4,5
4,0
3,5
3,0
2,5
2,0
1,5
1,0
0,5
0,0
petal_length
0,0 0,5 1,0 1,5 2,0 2,5 3,0 3,5 4,0 4,5 5,0 5,5 6,0 6,5 7,0
petal_width

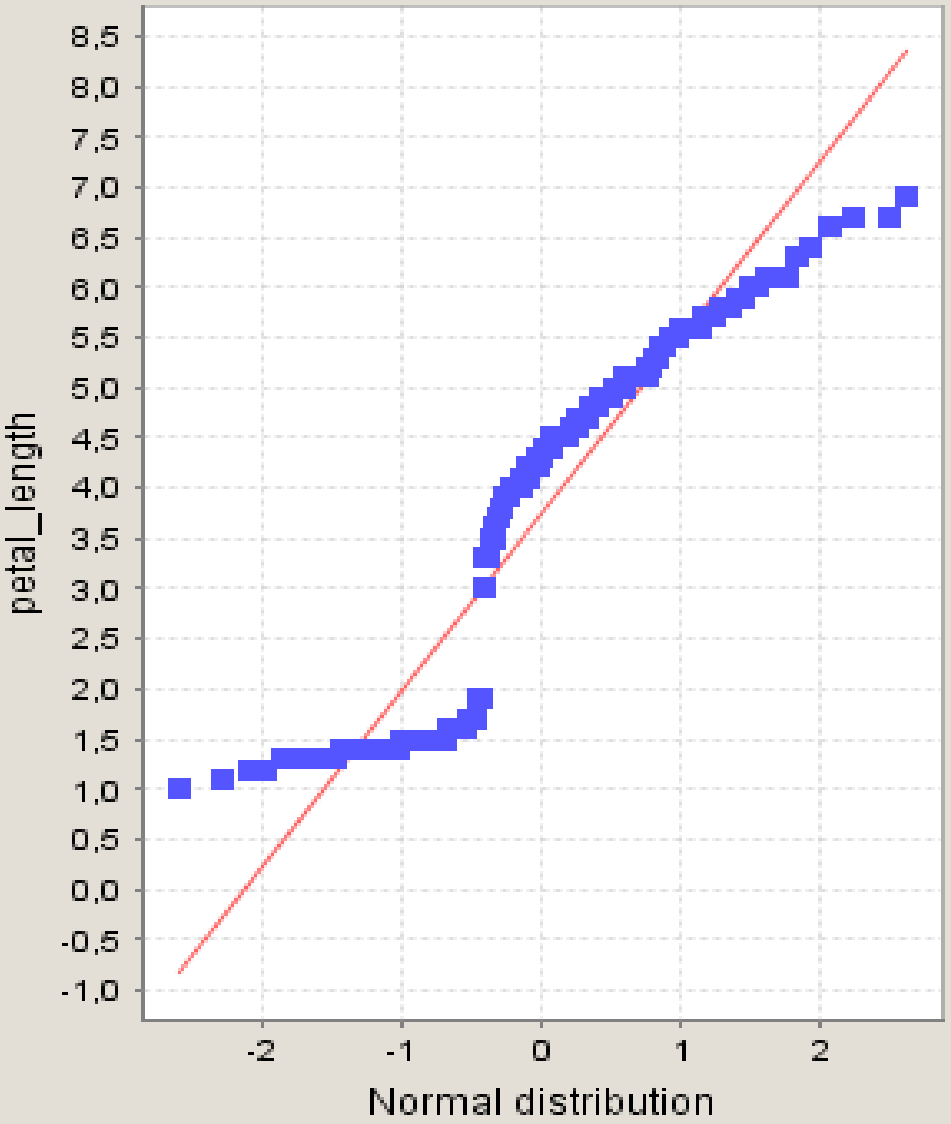
Probability Plot

x: Normal distribution

y: petal_length

EX (3.256383807786332;4.260949525547006)

DX (1.5848096213641532;1.9903089802139846)



Covariance matrix

Covariance:

	sepal_length	sepal_width	petal_length	petal_width
sepal_length	0.6811222222222222	-0.03900666666666667	1.2651911111111114	0.5134577777777779
sepal_width	-0.03900666666666667	0.1867506666666667	-0.31956800000000013	-0.11719466666666661
petal_length	1.2651911111111114	-0.31956800000000013	3.0924248888888854	1.2877448888888892
petal_width	0.5134577777777779	-0.11719466666666661	1.2877448888888892	0.5785315555555559

	Iris-setosa	Iris-versicolor	Iris-virginica
Iris-setosa	0.2222222222222221	-0.11111111111111092	-0.11111111111111098
Iris-versicolor	-0.11111111111111092	0.2222222222222221	-0.11111111111111098
Iris-virginica	-0.11111111111111098	-0.11111111111111098	0.22222222222222168
sepal_length	-0.2791111111111111	0.030888888888888896	0.24822222222222218

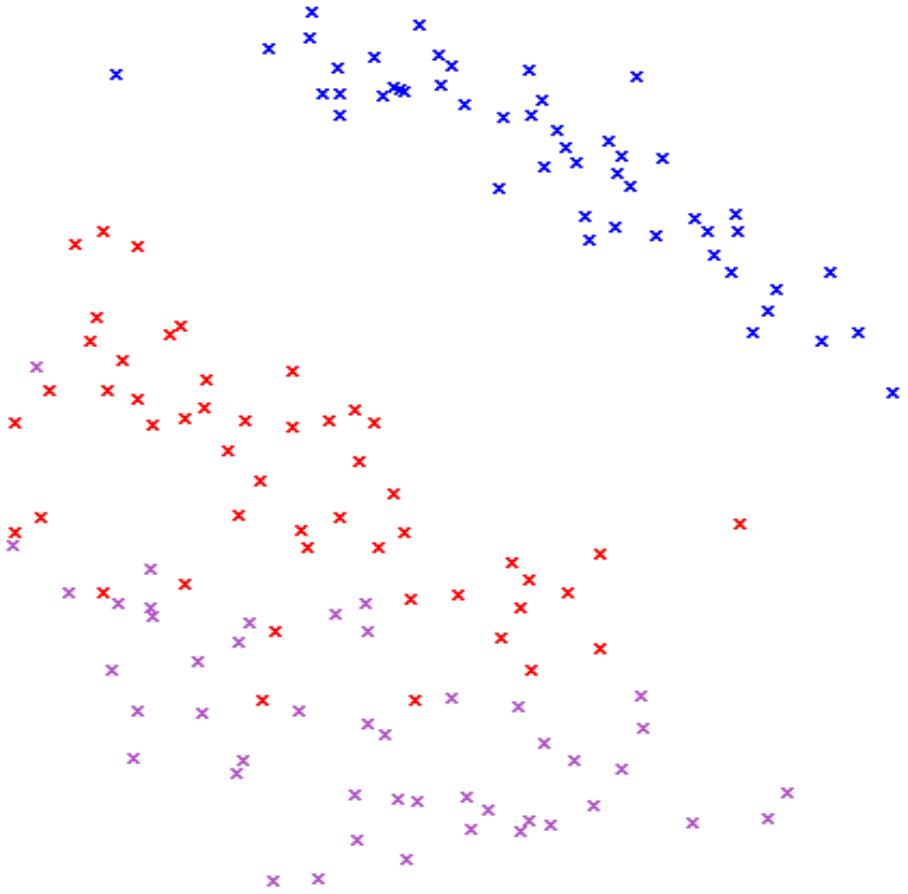
Corelation:

	sepal_length	sepal_width	petal_length	petal_width
sepal_length	0.99333333333338	-0.10864012161765108	0.8659424629228616	0.8125006091467221
sepal_width	-0.10864012161765108	0.9933333333333974	-0.41771265575849487	-0.35416712901639186
petal_length	0.8659424629228616	-0.41771265575849487	0.9933333333333375	0.9563387164039631
petal_width	0.8125006091467221	-0.35416712901639186	0.9563387164039631	0.993333333333335

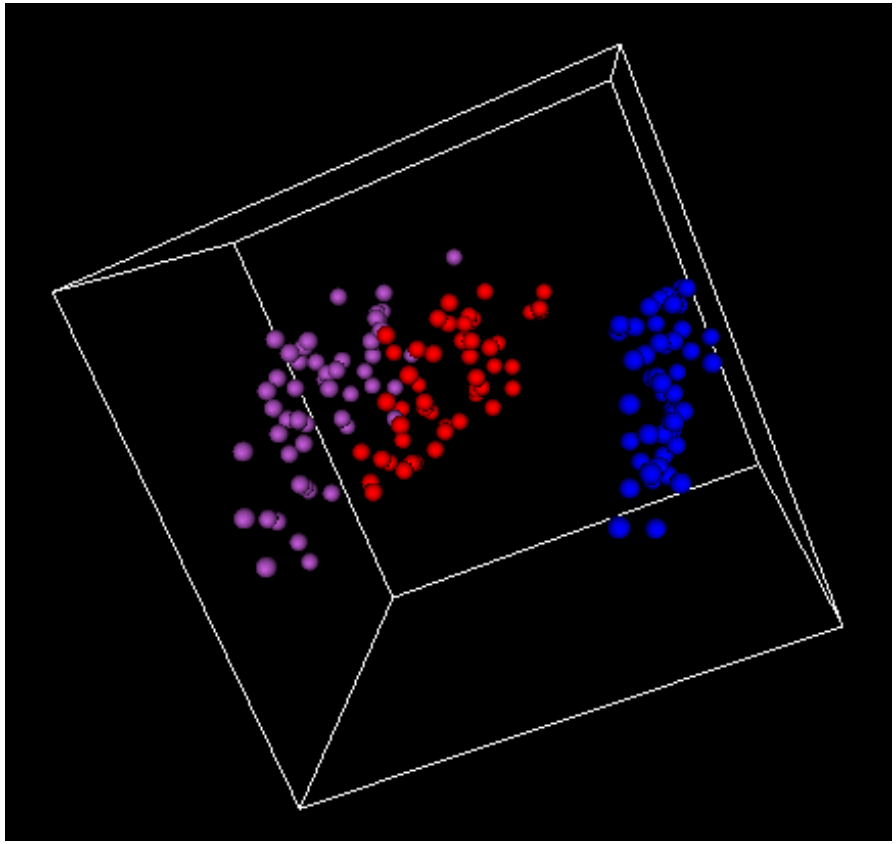
	Iris-setosa	Iris-versicolor	Iris-virginica
Iris-setosa	0.993333333333333	-0.4966666666666659	-0.49666666666666615
Iris-versicolor	-0.4966666666666659	0.993333333333333	-0.49666666666666615
Iris-virginica	-0.49666666666666615	-0.49666666666666615	0.9933333333333311
sepal_length	-0.7126328975615389	0.07886622035115759	0.6337666772103813

Projekce dat

2D



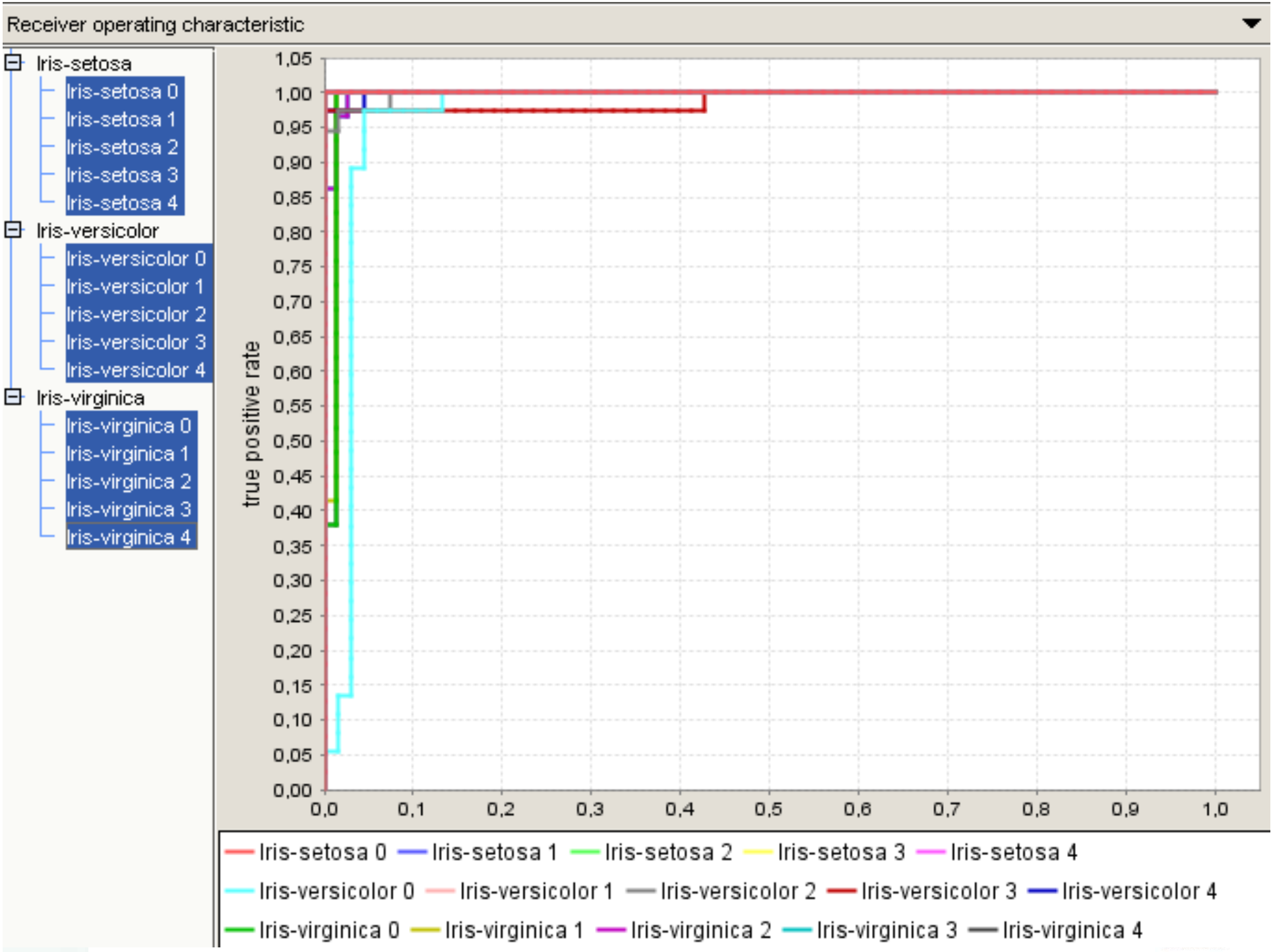
3D

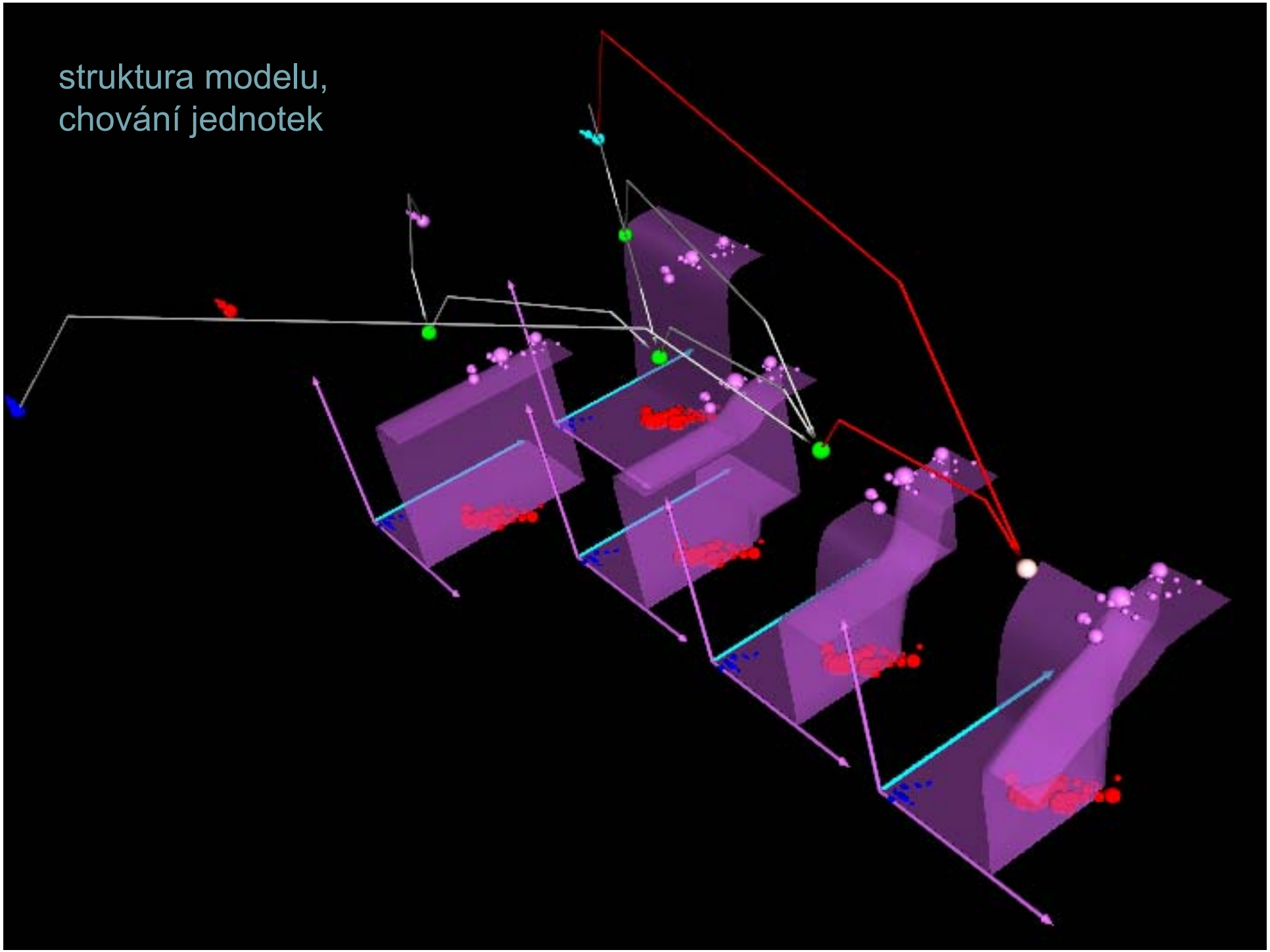


Významnost vstupních atributů

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	-	28	-	33	34	35	-	40	41	42	43
ChiSquare	7	6	9	10	8	2	1	3	5	4	38	47	28	26	21	48	50	33	29	40	49	27		22		42	32	34		30	17	16	20
GainRatio	10	6	5	9	7	1	3	2	4	8	38	50	21	49	27	47	29	36	33	22	37	48		24		39	43	28		34	18	20	19
InfoGain	7	6	8	9	2	10	1	3	5	4	38	47	28	26	29	48	50	40	21	33	46	27		24		22	31	32		43	17	16	20
OneR	7	9	10	8	2	3	1	6	4	5	38	47	48	26	25	40	29	28	50	21	24	33		35		43	30	34		27	16	11	12
ReliefF	6	9	7	4	10	3	5	2	1	8	26	37	24	36	27	50	21	46	30	49	28	31		23		45	41	47		48	18	15	16
SVM	2	4	6	3	7	9	1	10	8	23	47	24	26	39	30	37	42	22	34	21	38	44		5		17	41	14		28	20	45	18
SU	10	6	7	9	5	1	3	2	4	8	38	50	21	47	49	27	29	33	36	48	37	22		24		44	39	40		30	18	20	19
GAME 1	7	8	2	6	26	23	9	35	28	4	1	32	25	33	38	40	43	20	42	47	3	5		15		21	22	24		34	36	37	39
GAME 2	10	7	3	9	37	27	24	50	8	5	23	26	32	46	44	1	2	4	6	11	12	13		19		28	29	30		36	38	39	40
GAME 3	10	2	5	8	7	6	3	29	26	4	37	42	44	47	36	46	16	34	43	1	9	11		18		23	24	25		32	33	35	38
GAME 4	3	9	10	41	6	7	5	8	27	25	31	38	49	48	16	39	20	34	44	28	35	1		14		21	22	23		32	33	36	37
GAME 5	9	6	3	10	28	50	4	38	29	8	13	32	35	36	48	27	20	15	18	23	42	1		14		22	24	25		34	37	39	40

Nr.Ggroups	Most – least Ranked Features																															
All	1	2	3	4	5	6	46	15	24	37	48	13	17	21	28	29	30	41	43	45	49	50	7	8	9	10	11	12	14	16
1 / 2	1	2	3	4	26	12	35	46	5	6	7	8	9	10	11	13	14	15	16	17	18	19	20	21	22	23	24	25	27	28
1 / 3	1	2	37	21	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	22	23	24	25	26	27	28	29
1 / 4	1	2	41	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
3	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30


Receiver operating characteristic
Iris-setosa
Iris-setosa 0
Iris-setosa 1
Iris-setosa 2
Iris-setosa 3
Iris-setosa 4
Iris-versicolor
Iris-versicolor 0
Iris-versicolor 1
Iris-versicolor 2
Iris-versicolor 3
Iris-versicolor 4
Iris-virginica
Iris-virginica 0
Iris-virginica 1
Iris-virginica 2
Iris-virginica 3
Iris-virginica 4
true positive rate
1,05
1,00
0,95
0,90
0,85
0,80
0,75
0,70
0,65
0,60
0,55
0,50
0,45
0,40
0,35
0,30
0,25
0,20
0,15
0,10
0,05
0,00
0,0
0,1
0,2
0,3
0,4
0,5
0,6
0,7
0,8
0,9
1,0
Iris-setosa 0
Iris-setosa 1
Iris-setosa 2
Iris-setosa 3
Iris-setosa 4
Iris-versicolor 0
Iris-versicolor 1
Iris-versicolor 2
Iris-versicolor 3
Iris-versicolor 4
Iris-virginica 0
Iris-virginica 1
Iris-virginica 2
Iris-virginica 3
Iris-virginica 4


struktura modelu,
chování jednotek

Play FAKE GAME with your data

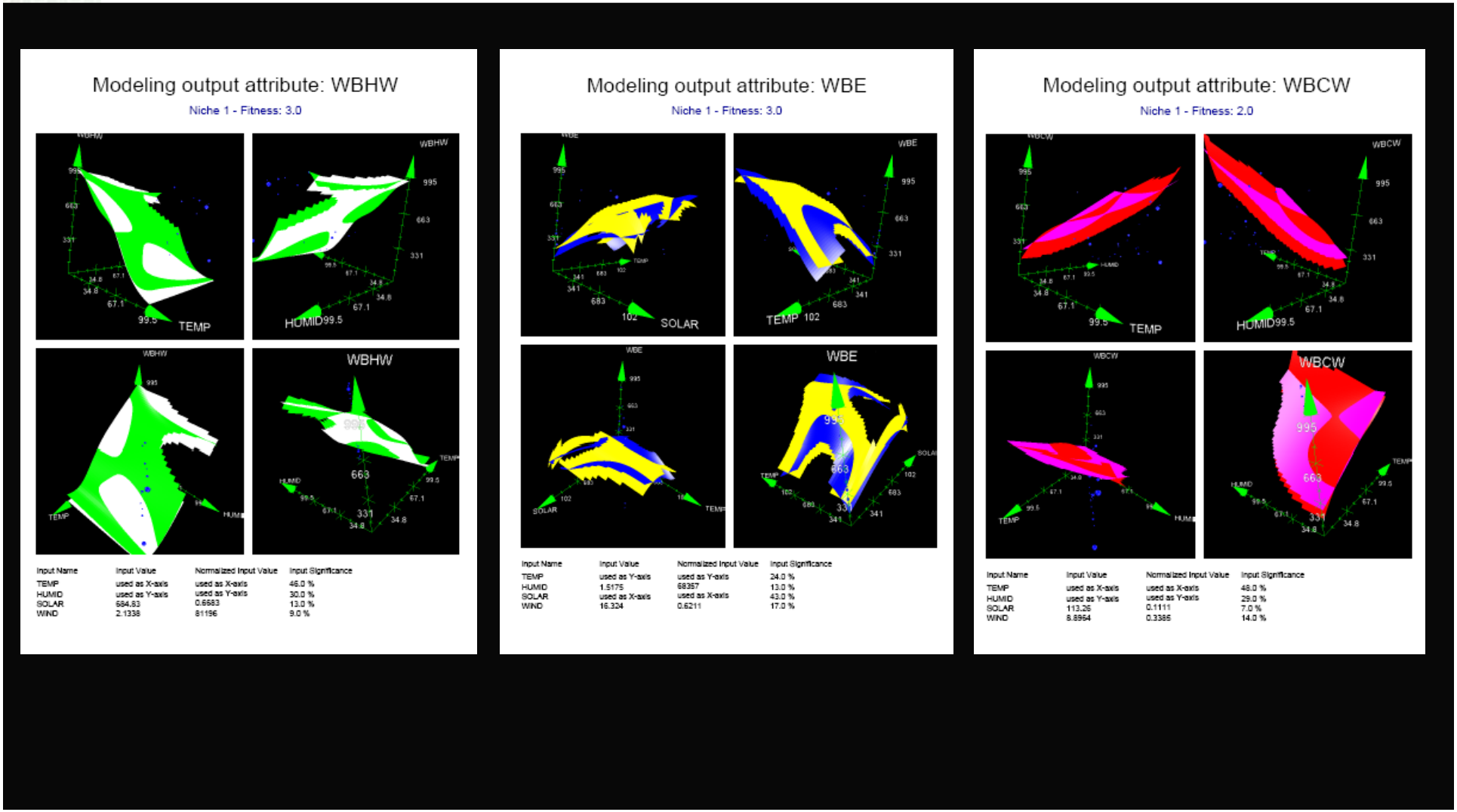
Log messages

- Evolving preprocessing sequences …
- Evolving ensemble of inductive models …
- Evolving "interesting" visualizations …

- Generating report …

- Done … in 2009 ☺

Toto už umíme generovat automaticky:



Co to znamená?

- Part of the FAKE GAME project (fully automated knowledge extraction from data)
- Ensemble of models is generated on a data set – in this case Building data [proben1]
- "Interesting and credible" areas of model behavior are located in multidimensional input space by means of the niching genetic algorithm.
- These areas are visualized in the 3D graph and the report is produced.
- More:

http://neuron.felk.cvut.cz/game/doc/fake-game.pdf

Chcete se na projektu podílet?

- Computational Intelligence Group, Dept. of Computer Science, FEE, Czech Technical University in Prague, Czech Republic
- logo: 
- website: http://cig.felk.cvut.cz/
- FAKE GAME project: http://sourceforge.net/projects/fakegame
- Contact:
- Pavel Kordik, kordikp@fel.cvut.cz