Neural Networks

doc. RNDr. Iveta Mrázová, CSc.

DEPARTMENT OF THEORETICAL COMPUTER SCIENCE AND MATHEMATICAL LOGIC

FACULTY OF MATHEMATICS AND PHYSICS, CHARLES UNIVERSITY IN PRAGUE

Neural Networks:

Multi-layered Neural Networks: an Application Example

doc. RNDr. Iveta Mrázová, CSc.

DEPARTMENT OF THEORETICAL COMPUTER SCIENCE AND MATHEMATICAL LOGIC

FACULTY OF MATHEMATICS AND PHYSICS, CHARLES UNIVERSITY IN PRAGUE

Neural Networks:

Contents:

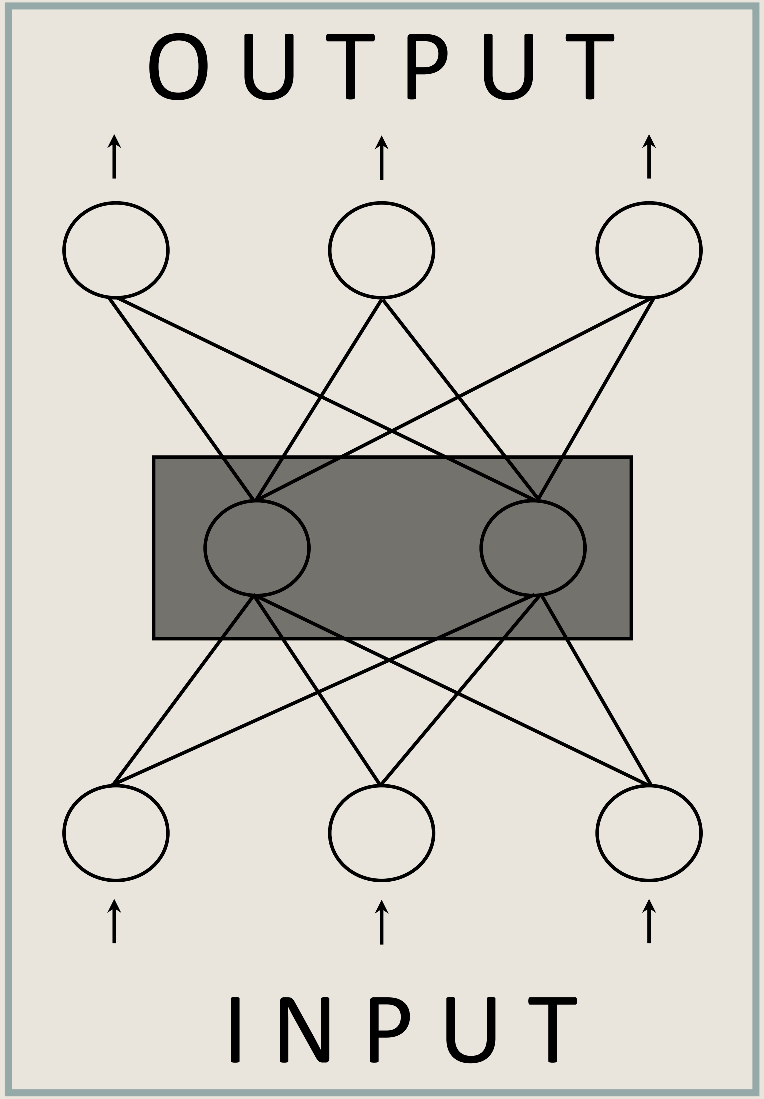
- **Multi-layered Neural Networks: Analysis of Their Properties**
- **Multi-layered Neural Networks: an Application Example**

Contents:

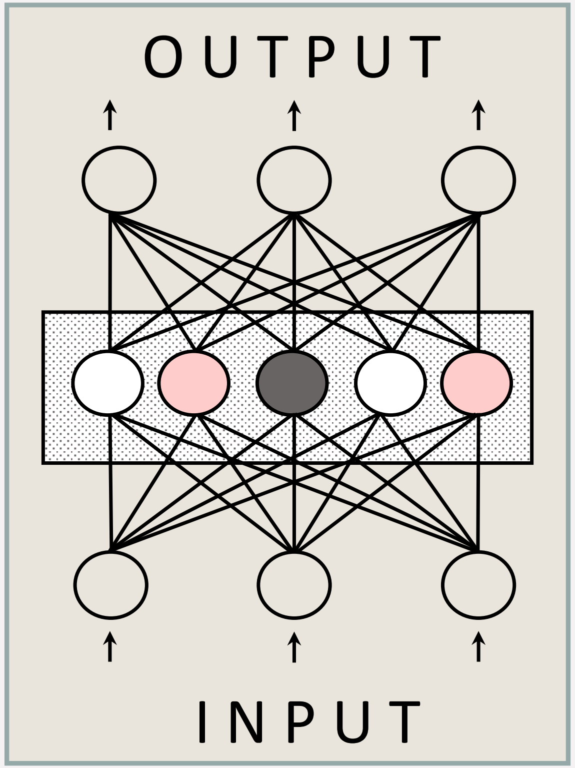
Internal Knowledge Representation and Pruning

- the number of neurons and generalization capa-bilities of the network

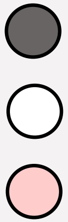
  → pruning and retraining



Condensed Internal Representation



- interpret the activity of hidden neurons:



	1	$\longleftrightarrow$	active	$\longleftrightarrow$	YES
	0	$\longleftrightarrow$	passive	$\longleftrightarrow$	NO
	$\frac{1}{2}$	$\longleftrightarrow$	silent	$\longleftrightarrow$	
		$\longleftrightarrow$	„impossible to decide“		

- transparent structure
- detection of redundant neurons and pruning
- **improved generalization**

Condensed Internal Representation

Definition:

For a BP-network B processing an input pattern $\vec{x}$:

- A hidden neuron with the weights $(w_1, \ldots, w_n)$, threshold ϑ, input pattern $\vec{z}$ and transfer function $f[\vec{w}, \vartheta](\vec{z})$ forms a *representation* *r* :

$$r = y = f[\vec{w}, \vartheta](\vec{z})$$

- The vector $\vec{r}$ of representations formed by a layer of hidden neurons is called an *internal representation* of $\vec{x}$

Condensed Internal Representation

Definition:

For a BP-network B, the internal representation $\vec{r} = (r_1, \dots, r_m)$ is:

- *binary*, if $r_i \in \{0, 1\};\ 1 \leq i \leq m$
- *condensed*, if $r_i \in \{0, 0.5, 1\};\ 1 \leq i \leq m$

Requirements on Forcing the Condensed Internal Representation

- formulate the „desired properties“ in the form of an objective function:

standard error function

$$G = E + c_s\, F$$

representation error function

the amount of influence of F on G

- local minima of the representation error function correspond to active, passive and silent states:

$$F = \sum_p \sum_h y_{h,p}^s \left(1 - y_{h,p}\right)^s \left(y_{h,p} - 0.5\right)^2$$

patterns

hidden neurons

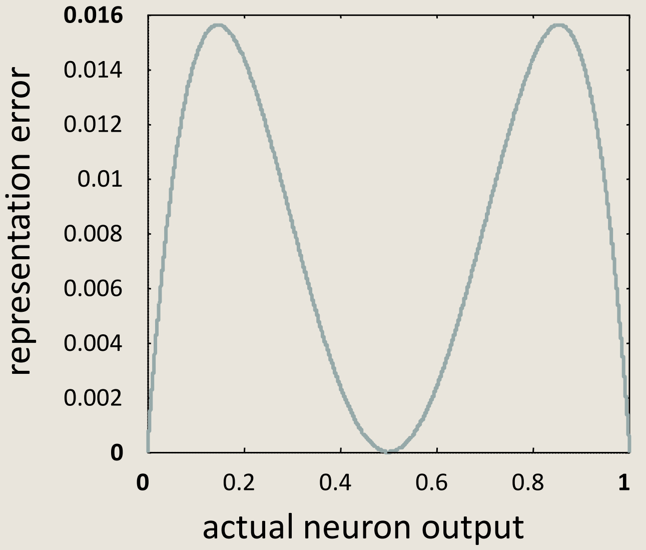
passive state

active state

the shape of F

silent state

The Influence of the Parameters on the Condensed Internal Representation



$$w_{ij}(t+1) = w_{ij}(t) + \alpha\delta_j y_i + \alpha_r \rho_j y_i +$$
$$+\alpha_m \left(w_{ij}(t) - w_{ij}(t-1)\right)$$

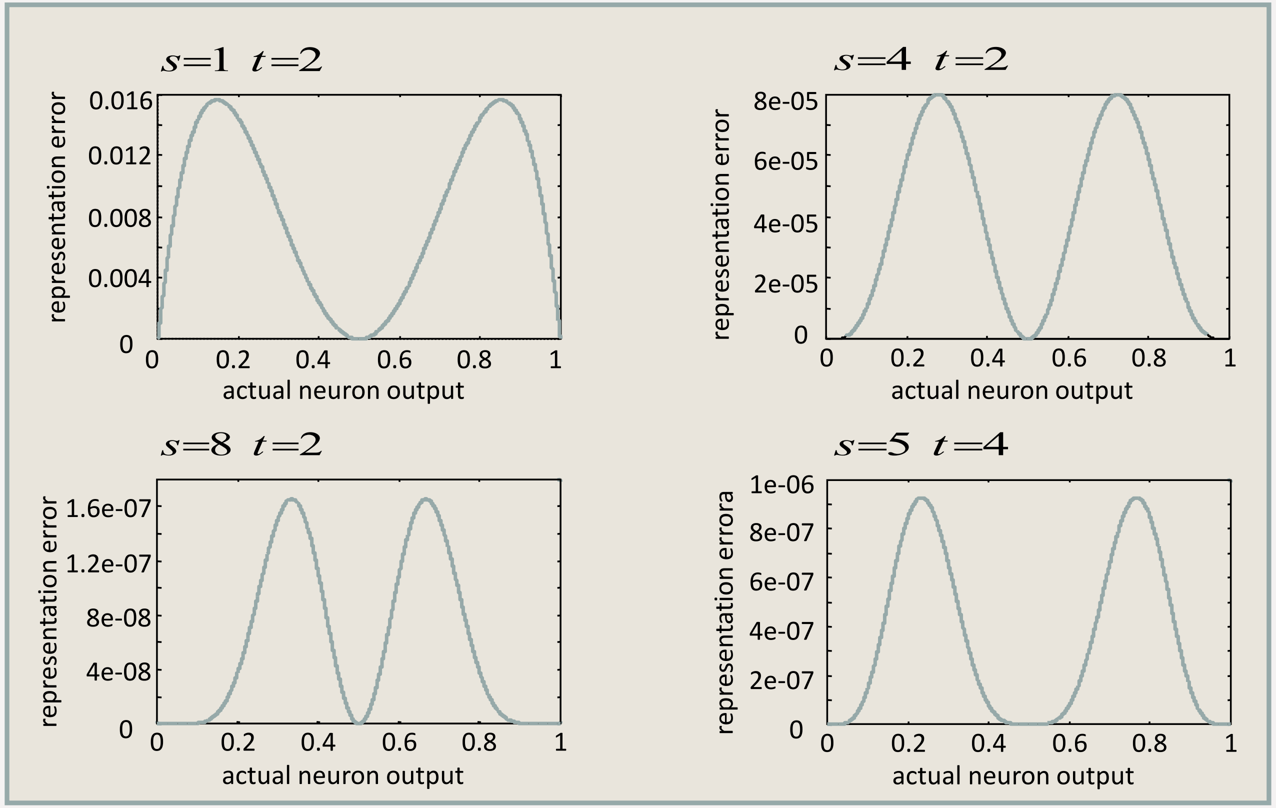
- slower enforcement of the internal representation and required network function
- stability of the formed internal representation and optimal network architecture
- form of the representation error function, speed and form of the enforced internal representation
- time complexity of weight adjustment

Error Term Enforcing the Condensed Internal Representation

Condensed internal representation $\left(y_j^s(1-y_i)^s(y_i-0.5)^2\right)$:

$$\rho_j = \begin{cases} 0 & \text{for the output neurons} \\ -\left[2(s+1)y_j\,(1-y_j)-\frac{s}{2}\right]\cdot\; y_j^s\,(1-y_j)^s\left(y_j-0.5\right) & \text{for neurons from the last hidden layer} \\ \left(\sum_k \rho_k w_{jk}\right) y_j\left(1-y_j\right) & \text{for other hidden neurons} \end{cases}$$

The Shape of the Representation Error Function $F = y^s(1-y)^s(y-0.5)^t$



Unambiguous Internal Representation

- Varying outputs should be represented by varying internal representations
- Formulation of the requirements in the form of a modified objective function: $G = E + F + H$
- The criterion for unambiguity of the IR:

$$H = -\frac{1}{2} \sum_{p} \sum_{q \neq p} \sum_{j} \sum_{o} \left(d_{o,p} - d_{o,q}\right)^2 \left(y_{j,p} - y_{j,q}\right)^2$$

patterns

hidden neurons

output neurons

= const. for a fixed p

= const. for a fixed q

= const. for a fixed q

Pruning According to the Internal Representation (1)

<u>Definition:</u>

For a given BP-network B and a set S of input patterns yielding the input vectors $\vec{z}$:

- A hidden neuron with the weights $(w_1, \dots\ \ w_n)$, the threshold ϑ and a transfer function $f[\vec{w}, \vartheta](\vec{z})$ forms a *uniform representation* r, if:

$$r = f[\vec{w}, \vartheta](\vec{z}) = const \qquad \text{for all input patterns } \vec{x} \in S$$

Pruning According to the Internal Representation (2)

<u>Definition:</u>

For a given BP-network B and a set S of input patterns yielding the input vectors $\vec{z}$:

- A hidden neuron $i \in N$ with the weights $(w_{i1}, \dots, w_{in})$, the threshold ϑ_i and a transfer function $f_i[\vec{w}_i, \vartheta_i](\vec{z})$ forms a *representation* r_i, *identical to the representation* r_j formed by the hidden neuron $j \in N$ with the weights $(w_{j1}, \dots, w_{jn})$, the threshold ϑ_j and a transfer function $f_j[\vec{w}_j, \vartheta_j](\vec{z})$, if:

$$f_i[\vec{w}_i, \vartheta_i](\vec{z}) = f_j[\vec{w}_j, \vartheta_j](\vec{z}) \quad \text{for all input patterns } \vec{x} \in S$$

Pruning According to the Internal Representation (3)

<u>Definition:</u>

For a given BP-network B and a set S of input patterns yielding the input vectors $\vec{z}$:

- A hidden neuron $i \in N$ with the weights $(w_{i1}, \dots, w_{in})$, the threshold ϑ_i and a transfer function $f_i[\vec{w}_i, \vartheta_i](\vec{z})$ forms a *representation* r_i, *inverse to the representation* r_j formed by the hidden neuron $j \in N$ with the weights $(w_{j1}, \dots, w_{jn})$, the threshold ϑ_j and a transfer function $f_j[\vec{w}_j, \vartheta_j](\vec{z})$, if:

$$f_i[\vec{w}_i, \vartheta_i](\vec{z}) = 1 - f_j[\vec{w}_j, \vartheta_j](\vec{z}) \text{ for all input patterns } \vec{x} \in S$$

Pruning According to the Internal Representation (4)

<u>Definition:</u>

For a BP-network B and a set S of input patterns, *a reduced layer* is a layer, for which it holds that:

- no neuron forms a uniform representation,
- no neuron i forms a representation identical to the representation formed by another neuron j and
- no neuron i forms a representation inverse to the representation formed by another neuron j.

Internal representation formed by a reduced layer is called *reduced*.

Pruning According to the Internal Representation (5)

<u>Definition:</u>

For a given set of input patterns S:

- a BP-network B is *reduced*, if all its hidden layers are reduced.
- a BP-network B *is equivalent to* a BP-network B', if for any input pattern $\vec{x} \in S$, the actual output $\vec{y}_B$ of the network B is equal to the actual output $\vec{y}_{B'}$ of the network B': $\vec{y}_B = \vec{y}_{B'}$

Pruning According to the Internal Representation (6)

Theorem:

To each BP-network B and a set of input patterns S, there exists an equivalent reduced BP-network B'.

Proof (idea) - construction of a reduced BP-network B':

Let $B = (N, C, I, O, w, t)$ is the original BP-network.

1. Sequential elimination of all those neurons i, that form a uniform representation r_i^u and addition of the product $w_{ij} r_i^u$ to all the thresholds ϑ_j in the following layer.

Pruning According to the Internal Representation (7)

<u>Proof (continue):</u>

2. Sequential elimination of all those neurons i, that form a representation r_i^{id} identical to the representation r_k formed by another neuron k and addition of the weights w_{ij} to all the weights w_{kj}, where j denotes the neurons in the following layer.
3. Sequential elimination of all those neurons i, that form a representation r_i^{in} inverse to the representation r_k formed by another neuron k and replacement of all the weights w_{kj}, where j denotes the neurons from the following layer, by the difference $w_{kj} - w_{ij}$ and addition of the weights w_{ij} to the threshold ϑ_j of each neuron j.

Pruning According to the Internal Representation (8)

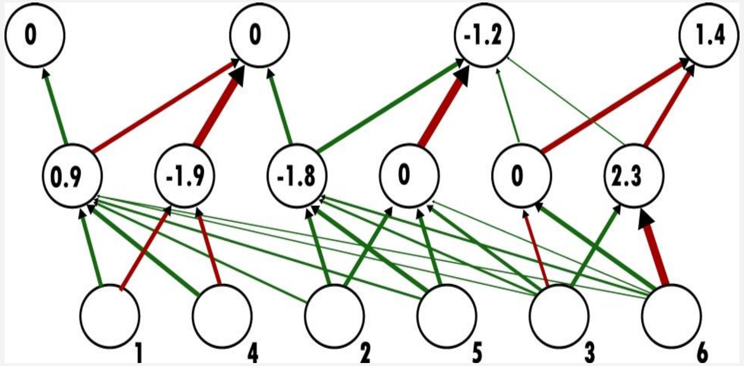
<u>Proof (continue):</u>

- Then, the actual output $\vec{y}_{B'}$ of the BP-network B' will be equal to the actual output $\vec{y}_B$ of a BP-network B for any input pattern $\vec{x}$.
- The BP-network B' constructed from the BP-network B in the above-discussed way is reduced and equivalent to B.

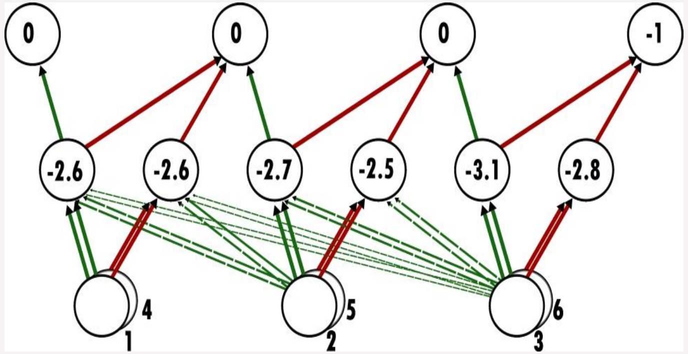
QED

Experimental Results: Binary Addition Networks [5(≈(1,-1,1)) + 3(≈(-1,1,1)) = 8(≈(1,-1,-1,-1))]

[2011, Reitermanová, Mrázová: A New Sensitivity-Based Pruning Technique …]



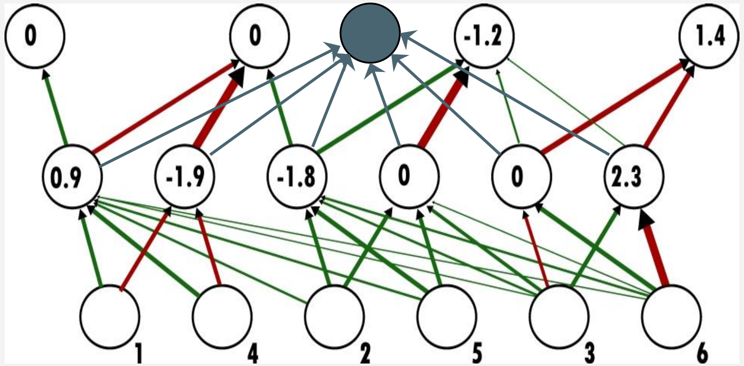
- SCG with hints (carry to the 2^{nd} output neuron)
  - ‘carry’ of the first and second output bit – hidden neurons 1 and 3
  - the function of other hidden neurons not clear

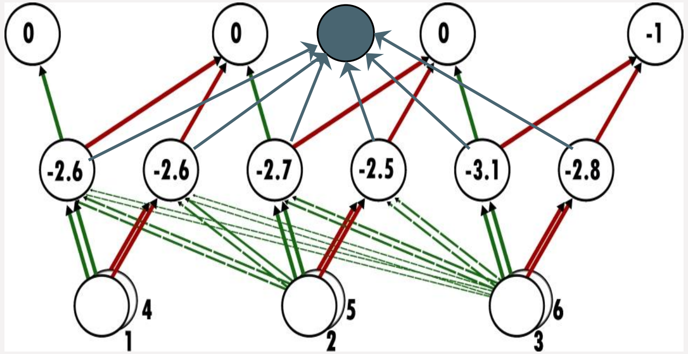


- SCGIR with hints (carry to the 2nd output neuron)
  - ‘carry’ to higher output bits – hidden neurons 1, 3, 5
  - a similar function is apparent for the respective output neurons

Experimental results: binary addition [5(≈(1,-1,1)) + 3(≈(-1,1,1)) = 8(≈(1,-1,-1,-1))]

[2011, Reitermanová, Mrázová: A New Sensitivity-Based Pruning Technique …]





- SCG *with hints* (carry to the 2nd output neuron)
  - 'carry' of the first and second output bit – hidden neurons 1 and 3
  - the function of other hidden neurons not clear
- SCGIR *with hints* (carry to the 2nd output neuron)
  - 'carry' to higher output bits – hidden neurons 1, 3, 5
  - a similar function is apparent for the respective output neurons

Neural Networks:

Contents:

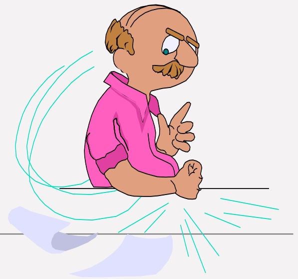
- **Multi-layered Neural Networks: Analysis of Their Properties**
- **Multi-layered Neural Networks: an Application Example**

Contents:

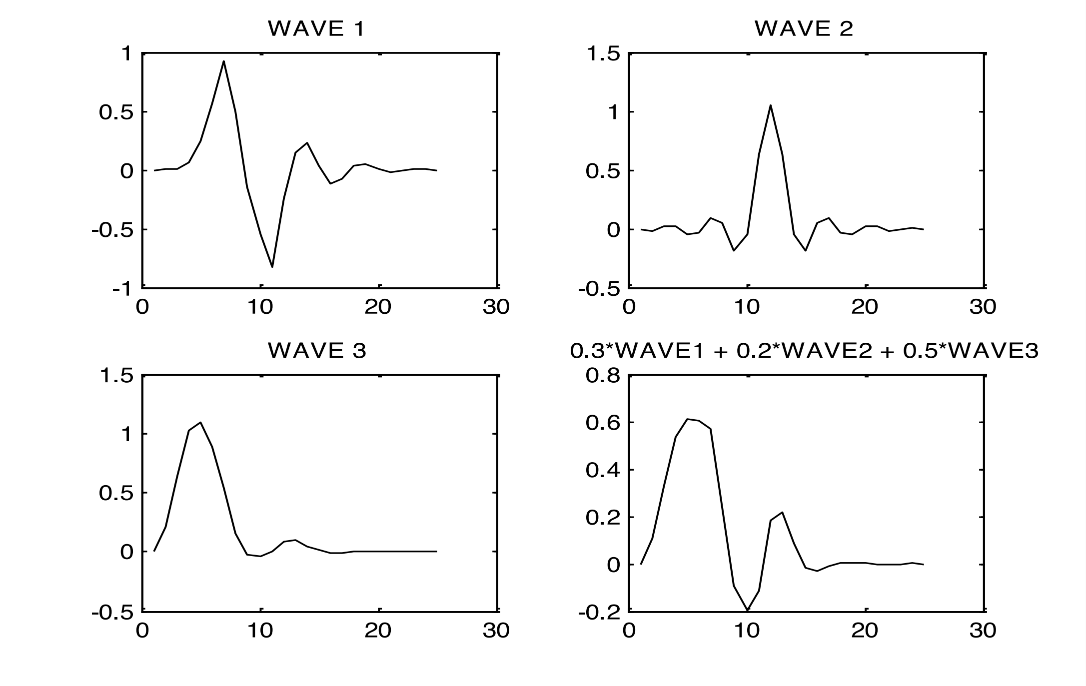
- Multi-layered Neural Networks: Analysis of Their Properties
  - Kolmogorov´s Theorem
  - Function Approximation
  - The Complexity of Learning
  - The Number of Regions in the Feature Space
  - Vapnik-Chervonenkis Dimension
- Multi-layered Neural Networks: an Application Example
  - Internal Knowledge Representation and Pruning
  - Sensitivity Analysis and Feature Selection
  - Analysis of the World Bank Data

Acoustic Emission: Simulation

(I. Mrázová, M. Chlada, and Z. Převorovský)

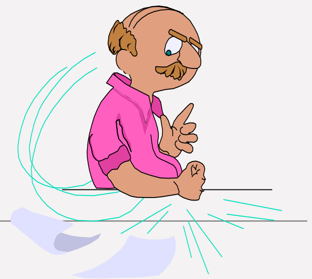


MODELED SIGNAL

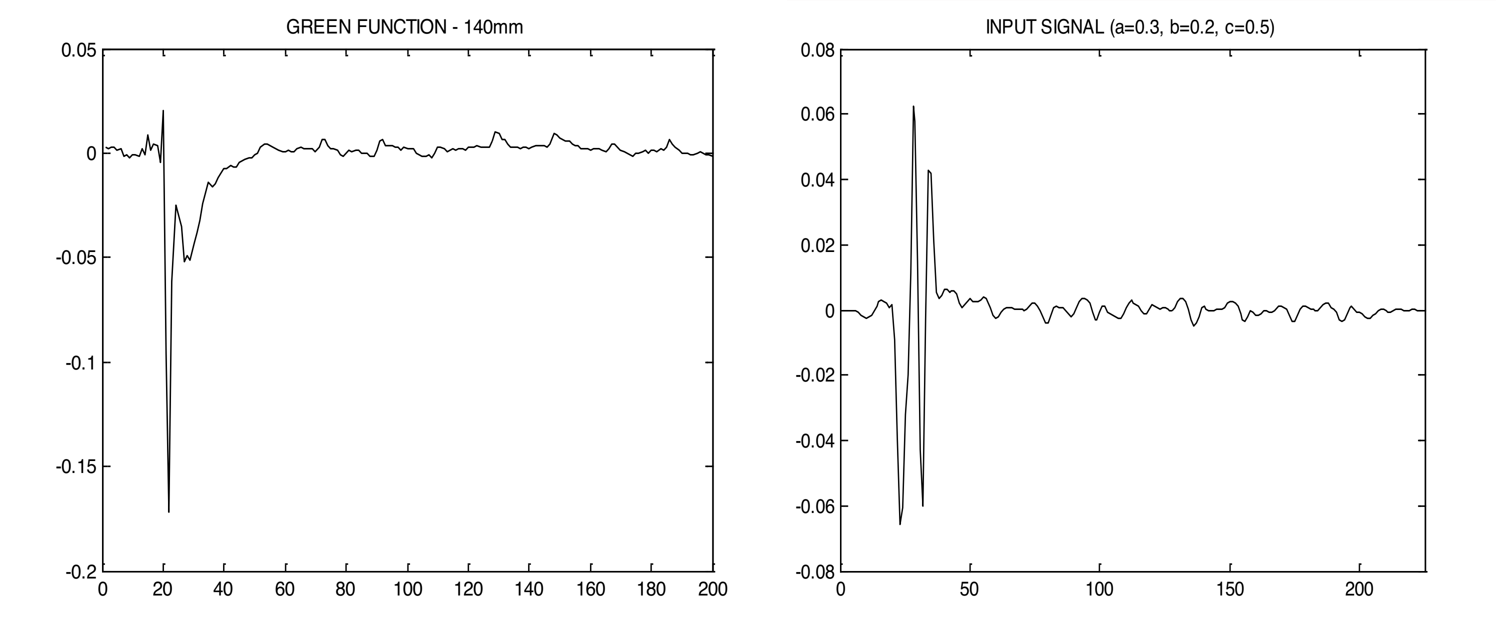


Simulated AE-data

(I. Mrázová, M. Chlada, and Z. Převorovský)

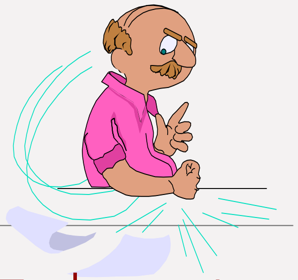


CONVOLUTION WITH THE GREEN FUNCTION

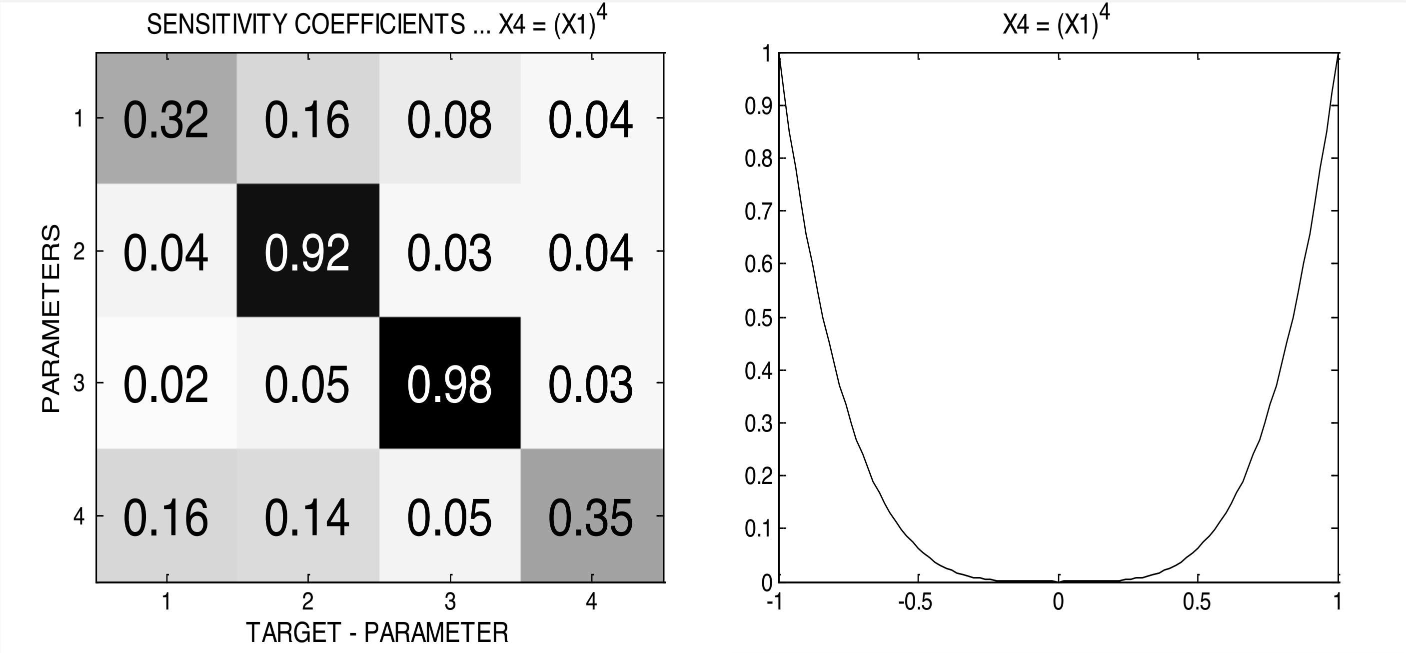


Dependency Model

(I. Mrázová, M. Chlada, and Z. Převorovský)



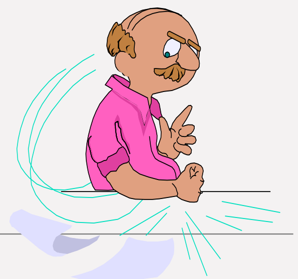
Overall *network sensitivity* to the *r-th input* (over Q patterns): $sens_r = 1/Q \ \sum_q^Q \sum_s \left| \partial y_{q,s} / \partial y_{q,r} \right|$

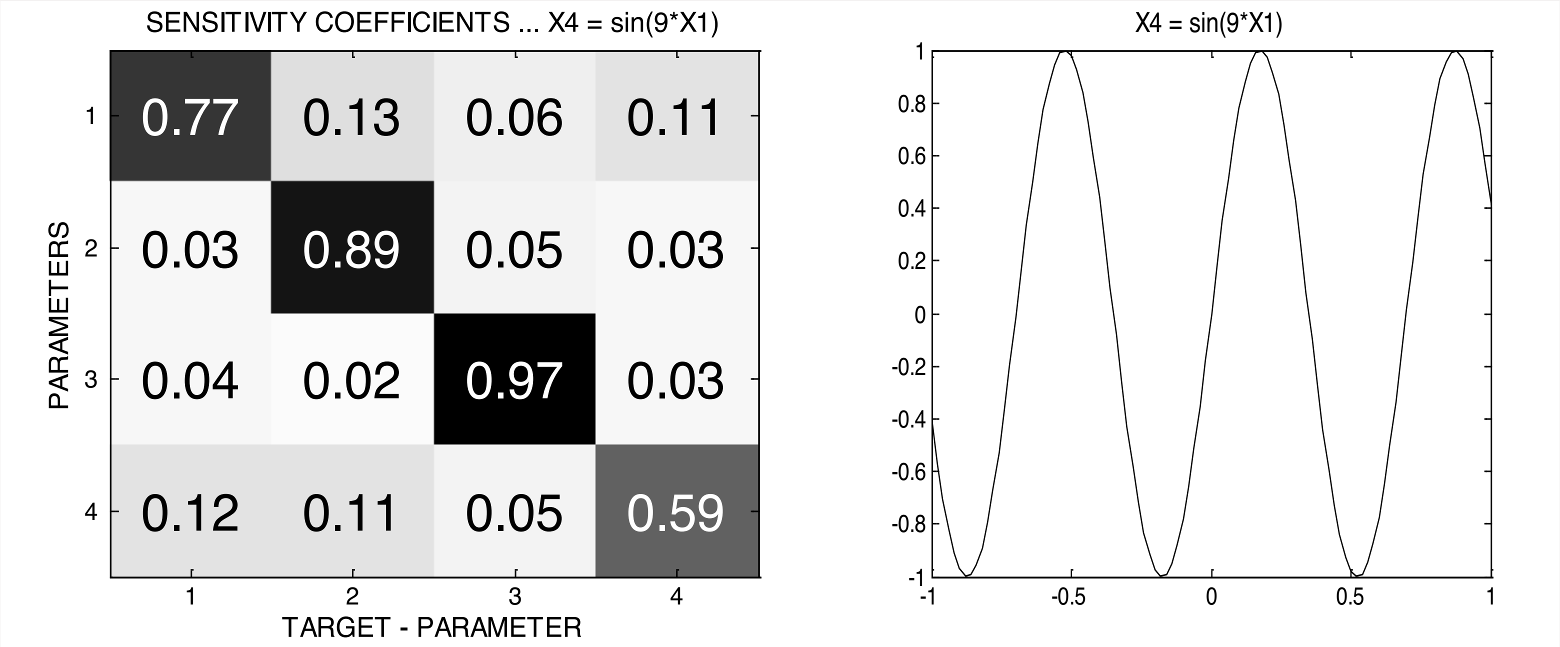


the sensitivity of the s-th output to the r-th input

Dependency Model

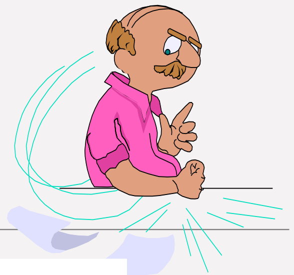
(I. Mrázová, M. Chlada, and Z. Převorovský)





Factor vs. Sensitivity Analysis of Input Parameters

(I. Mrázová, M. Chlada, and Z. Převorovský)



input parametrs

	1	2	3	4	5	6	7	8	9
1	0.04	0.91	0.16	0.10	0.01	0.07	0.06	0.01	0.23
2	0.09	0.02	0.01	0.19	0.03	0.03	0.95	0.03	0.16
3	0.10	0.96	0.15	0.00	0.03	0.00	0.02	0.05	0.07
4	0.13	0.91	0.02	0.03	0.05	0.06	0.04	0.06	0.20
5	0.30	0.05	0.04	0.41	0.06	0.49	0.17	0.07	0.66
6	0.26	0.00	0.04	0.03	0.08	0.93	0.02	0.10	0.20
7	0.29	0.06	0.03	0.27	0.03	0.17	0.16	0.04	0.86
8	0.12	0.06	0.01	0.88	0.02	0.02	0.24	0.03	0.36
9	0.90	0.15	0.09	0.08	0.15	0.17	0.06	0.21	0.18
10	0.93	0.14	0.06	0.08	0.10	0.17	0.06	0.09	0.19
11	0.25	0.10	0.12	0.03	0.25	0.11	0.03	0.90	0.05
12	0.20	0.07	0.14	0.02	0.93	0.09	0.03	0.23	0.04
13	0.04	0.09	0.97	0.00	0.05	0.00	0.00	0.06	0.02
14	0.08	0.18	0.95	0.02	0.09	0.04	0.01	0.06	0.02

selected factors

SENSITIVITY COEFFICIENTS

INPUTS \ OUTPUTS	1	2	3
1	0.173	0.266	0.149
2	0.093	0.068	0.047
3	0.320	0.193	0.184
4	0.301	0.178	0.196
5	0.564	0.250	0.206
6	0.196	0.322	0.158
7	0.099	0.063	0.043
8	0.065	0.015	0.030
9	0.022	0.014	0.016
10	0.053	0.020	0.012
11	0.035	0.012	0.032
12	0.039	0.050	0.022
13	0.081	0.134	0.082
14	0.260	0.172	0.109

- 9 factors selected ("explain" 98.4% of variables)
- elimination of linearly dependent input parameters

- 7 features selected
- detection of non-linear dependencies among input parameters (1, 3, 4, 5, 6, 13, 14)

Neural Networks:

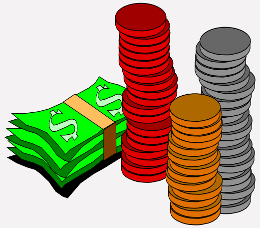
Contents:

- **Multi-layered Neural Networks: Analysis of Their Properties**
- **Multi-layered Neural Networks: an Application Example**

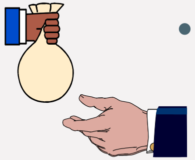
Contents:

- Multi-layered Neural Networks: Analysis of Their Properties
  - Kolmogorov´s Theorem
  - Function Approximation
  - The Complexity of Learning
  - The Number of Regions in the Feature Space
  - Vapnik-Chervonenkis Dimension
- Multi-layered Neural Networks: an Application Example
  - Internal Knowledge Representation and Pruning
  - Sensitivity Analysis and Feature Selection
  - Analysis of the World Bank Data

Analysis of the World Bank Data

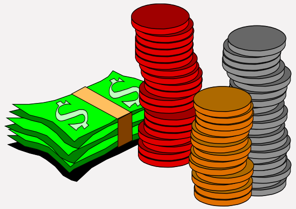


- WDI-indicators (indicators of world development)
  - published every year by the World Bank
    - support developing countries – loans / investments
    - assess the state of economies and their development
  - data origin – incomplete and not accurate



- used techniques
  - regression analysis – linear dependencies
  - categorization of economies used in developed countries (G. Ip, Wall Street Journal)
  - categorization of economies according to GDP (World Bank)
  - Kohonen maps (T. Kohonen, S. Kaski, G. Deboeck)

Analysis of the World Bank Data: the Used WDI-indicators



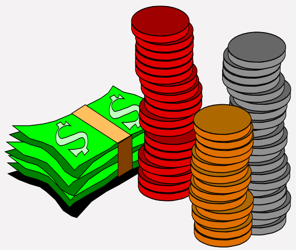
- GDP implicit deflator
- External debt (% GNP)
- Total debt service (% of export of goods and services)
- High-technology exports (% of manufactured exports)
- Military expenditures (% GNP)
- Expenditures for research and development (% GNP)
- Total expenditures on health (% GDP)
- Public expenditure on education (% GNP)
- Male life expectancy at birth
- Fertility rates
- GINI-index (the distribution of income / consumption)
- Internet hosts per 10000 people
- Mobile phones per 1000 people

- Purchasing power parity (PPP)

- - - - - - - - - - - - - - - - - - - -

- GNP per capita (in USD)
- Average annual growth rate of GDP (% per capita)

Analysis of the World Bank Data: Preprocessing



- 99 states with 16 WDI-indicators
- elementwise transformation of patterns to the interval (0,1) by means of:

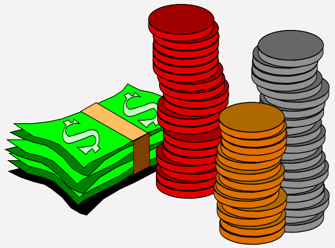
$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad \text{and} \quad x'' = \frac{1}{1 + e^{-4\left(x' - \frac{1}{2}\right)}}$$

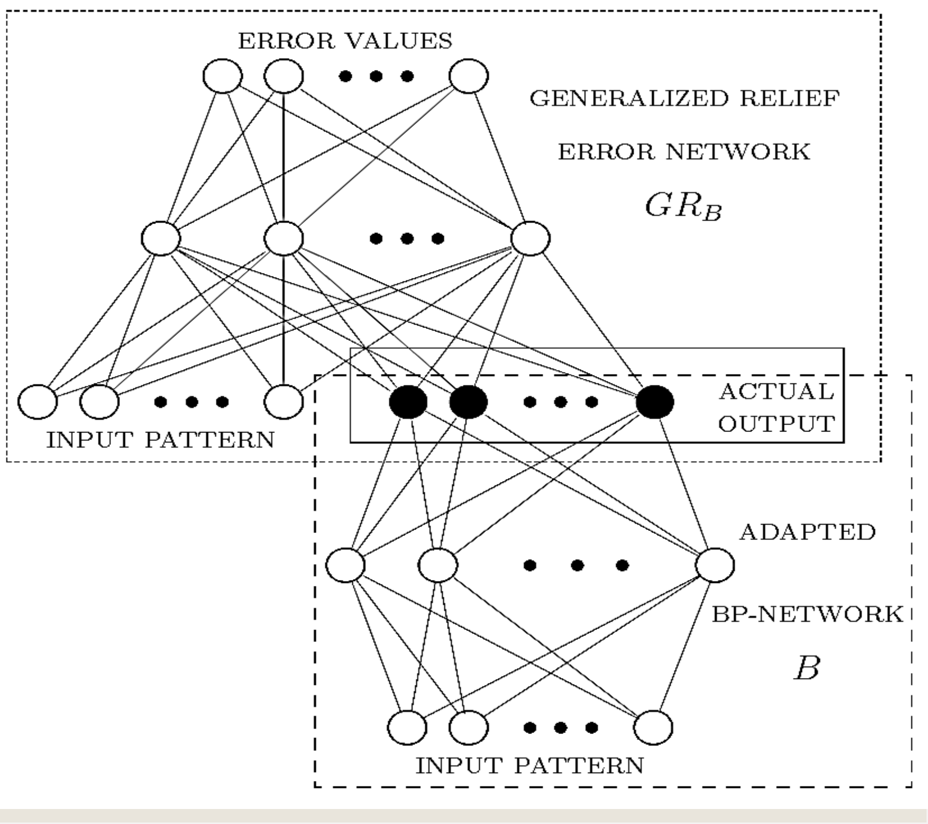
maximum over all patterns

minimum over all patterns

- FCM-clustering: 7 clusters, fuzziness parameter $s = 1.4$
- controlled learning and iterative recall:
  - **99 (90+9)** states with **14 (13+1)** WDI-indicators
  - GREN-net **14-12-1**, BP-net **13-10-1**; **500-600** training cycles

Analysis of the World Bank Data: Preprocessing



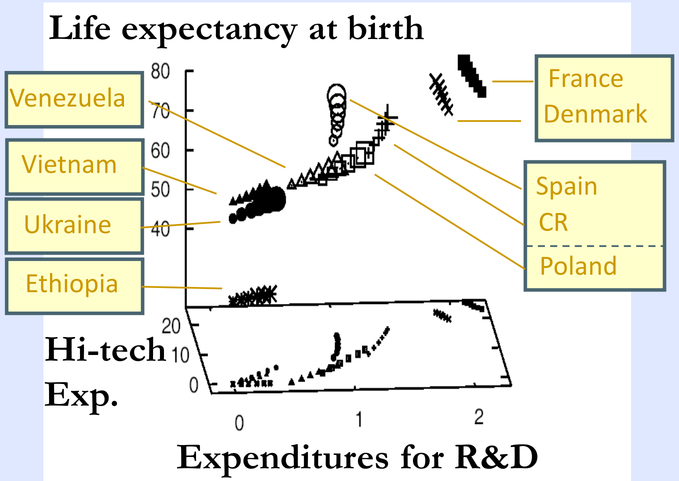


- controlled learning and iterative recall:
  - **99 (90+9)** states with **14 (13+1)** WDI-indicators
  - GREN-net **14-12-1**, BP-net **13-10-1**; **500-600** training cycles

Analysis of the World Bank Data:
Impact of the Indicators on the Economy

Indicator	Net 1	Net 2
GDP defl.	0.0	0.0
External debt	5.6	10.9
Total debt service	5.5	8.1
High-tech export	12.2	6.6
Military expenditures	5.4	6.1
Expenditures fot R&D.	16.0	12.0
Internet users	11.1	12.4
Mobile phones	8.3	10.0
GINI-index	7.1	3.9
Life expectancy	12.3	7.6
Fertility	4.4	5.0
Expend. on health .	6.1	10.9
Expend. on education	6.1	6.1

Sensitivity of GREN-networks

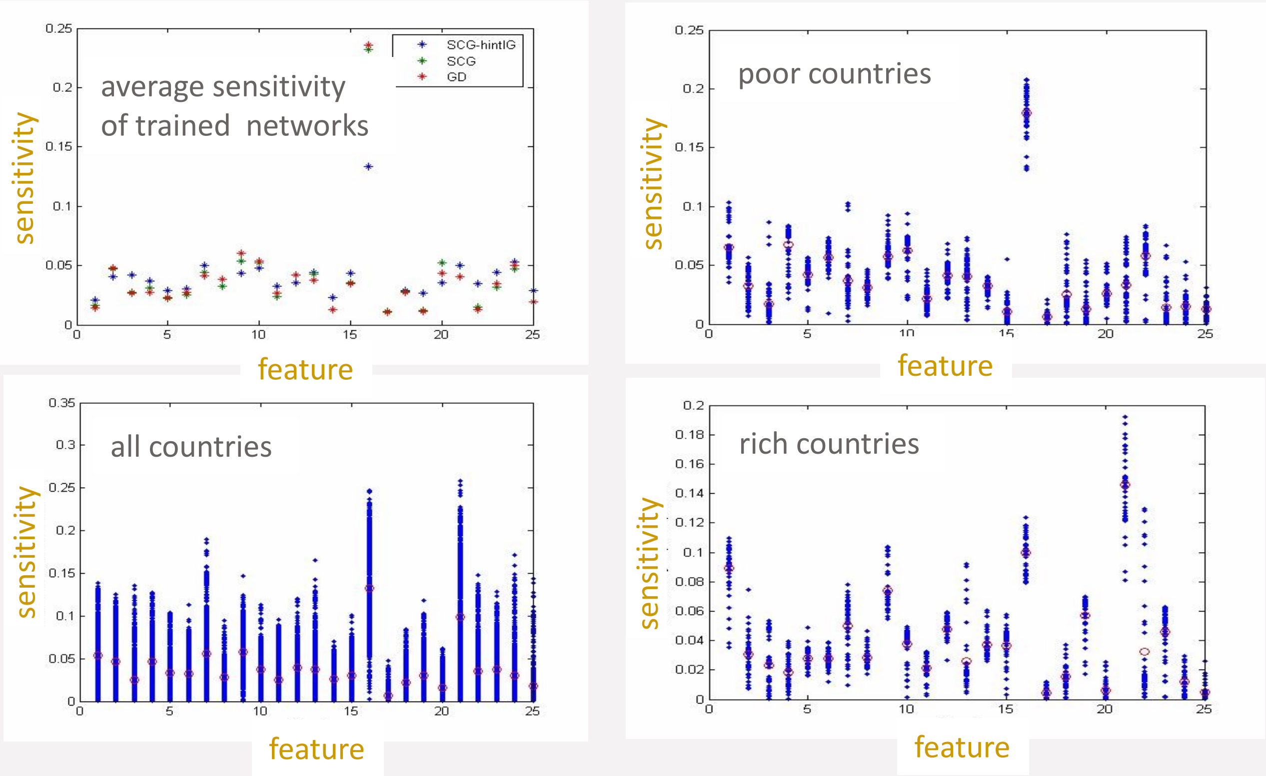


Iterative recognition – higher GNP / PPP (Net 1)

Sensitivity to Input Features

(I. Mrázová and Z. Reitermanová)

FIXED LINE AND MOBIE PHONE SUBSCRIBERS



FIXED LINE AND MOBIE PHONE SUBSCRIBERS

LIFE EXPECTANCY AT BIRTH

PPP

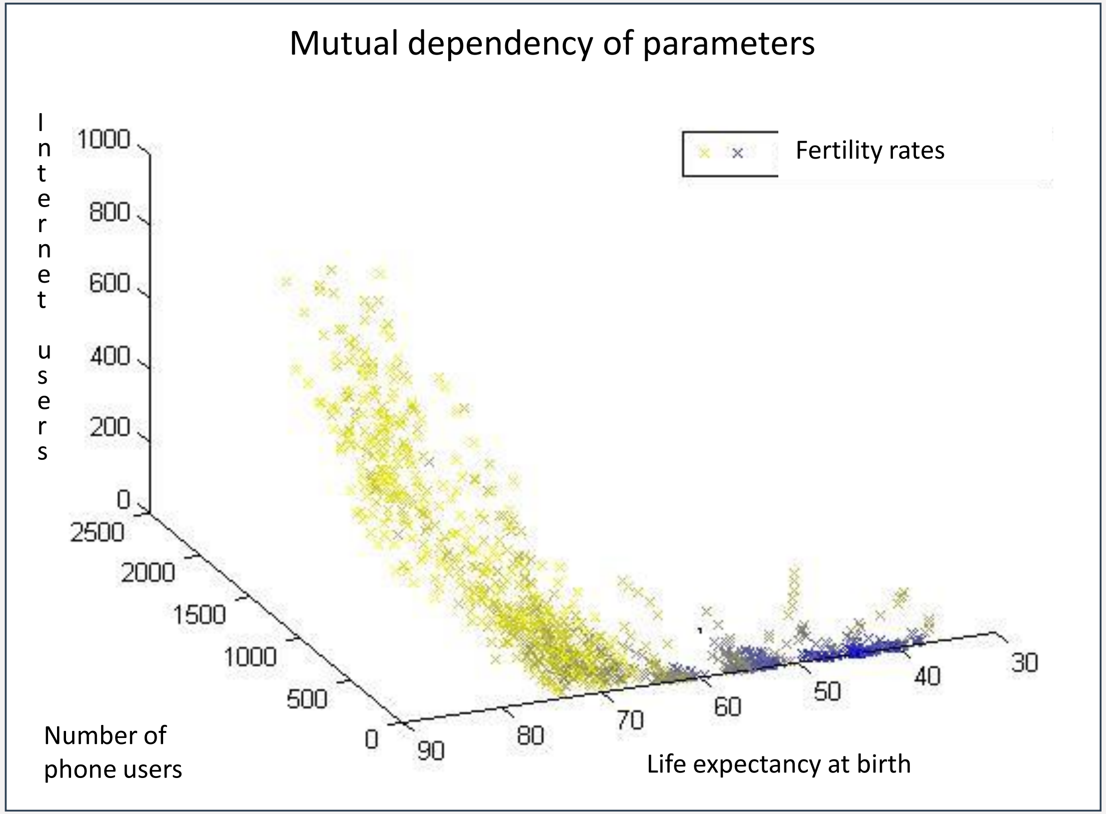
FIXED LINE AND MOBIE PHONE SUBSCRIBERS

LIFE EXPECTANCY AT BIRTH

TAXES ON INCOME, ETC.

Mutual Dependency of Parameters

(I. Mrázová and Z. Reitermanová)



Neural Networks:

Contents:

- **Multi-layered Neural Networks: Analysis of Their Properties**
- **Multi-layered Neural Networks: an Application Example**

Contents:

- Multi-layered Neural Networks: Analysis of Their Properties
  - Kolmogorov´s Theorem
  - Function Approximation
  - The Complexity of Learning
  - The Number of Regions in the Feature Space
  - Vapnik-Chervonenkis Dimension
- Multi-layered Neural Networks: an Application Example
  - Internal Knowledge Representation and Pruning
  - Sensitivity Analysis and Feature Selection
  - Analysis of the World Bank Data