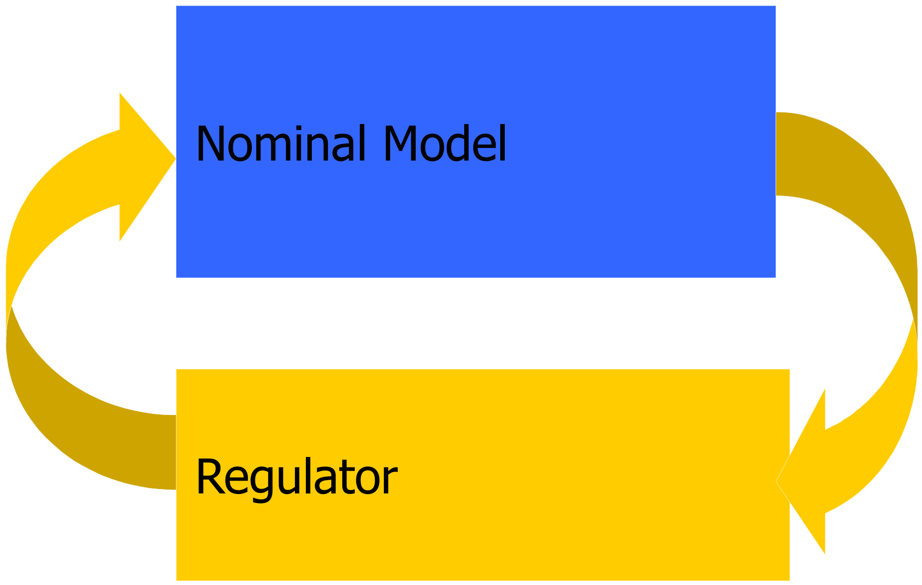
1918
TALLINNA
TEHNIKAÜLIKOOL

Dünaamiliste süsteemide modelleerimine

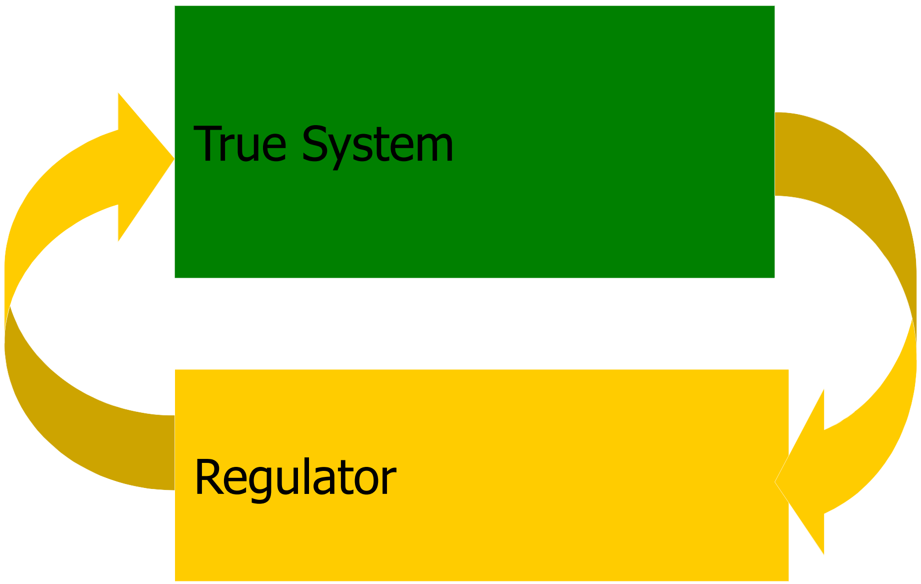
Identification for control in a non-linear system world

Eduard Petlenkov, Automaatikainstituut, 2015

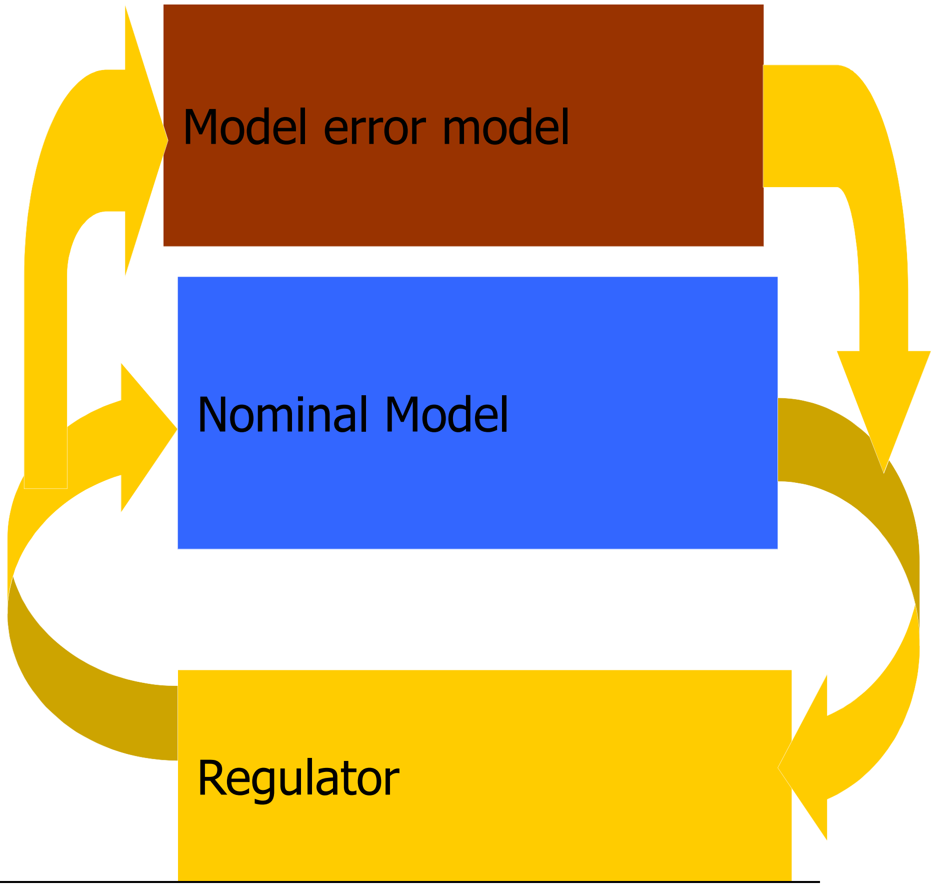
Control Design



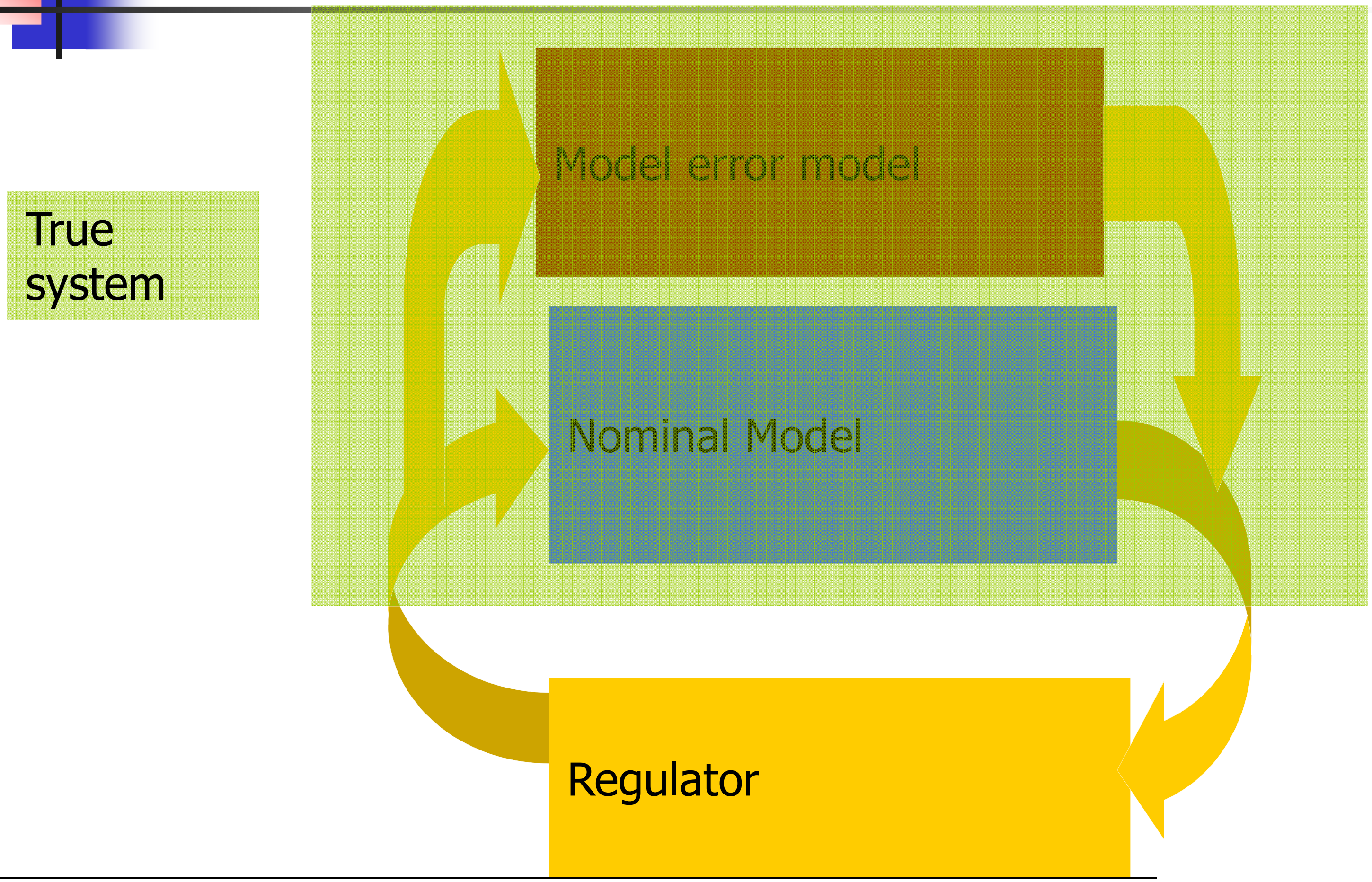
Control Design



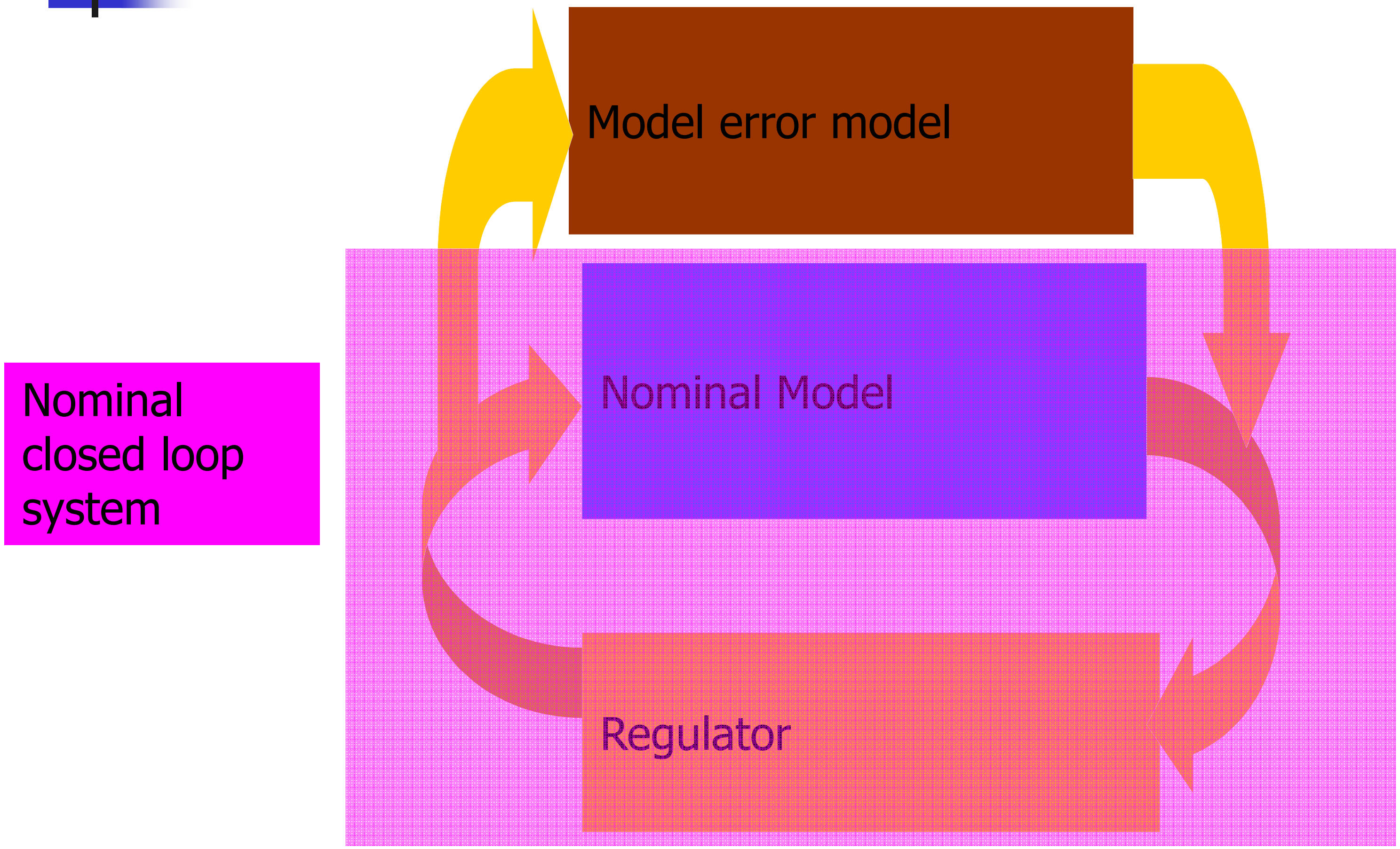
Control Design



Control Design



Control Design



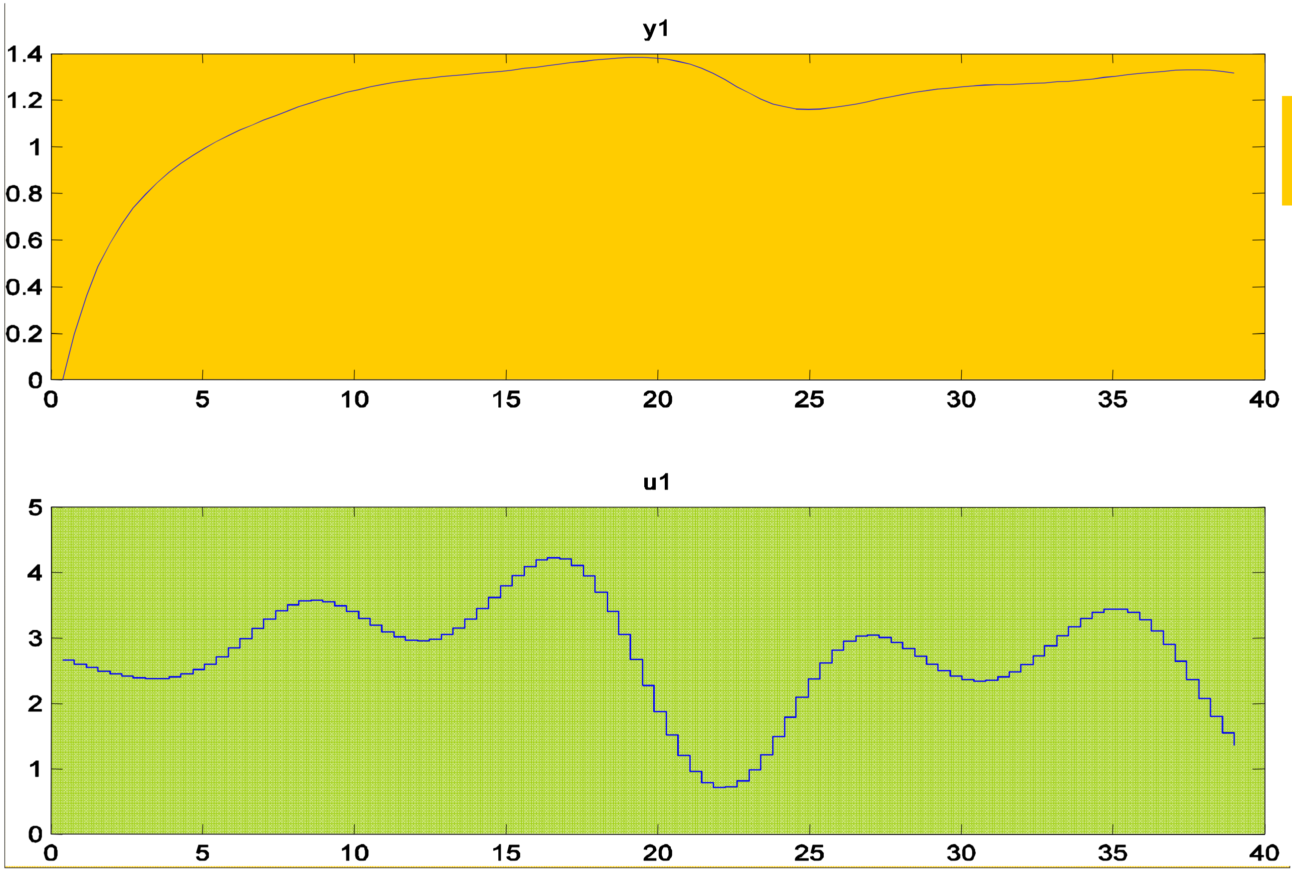
Model Error Models

$$\varepsilon = y - y_{model}$$

Identification for Control

- Identifiction for control is the art and technique to design identification experiments and regulator design methods so that the model error model matches the nominal closed loop system in a suitable way

A Data Set/ nonlinear process

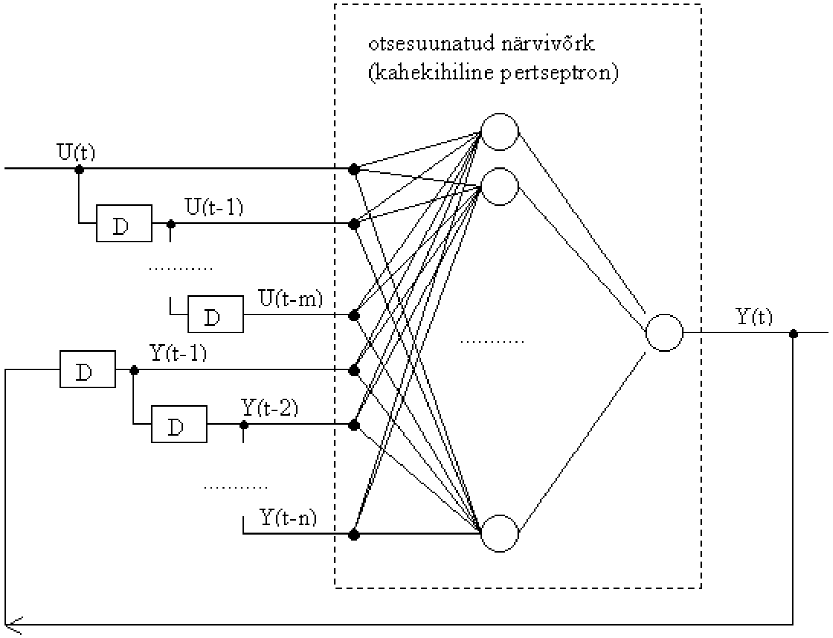


Output

Input

Feedback Neural Networks. Identification of dynamic systems (1)

External feedback:

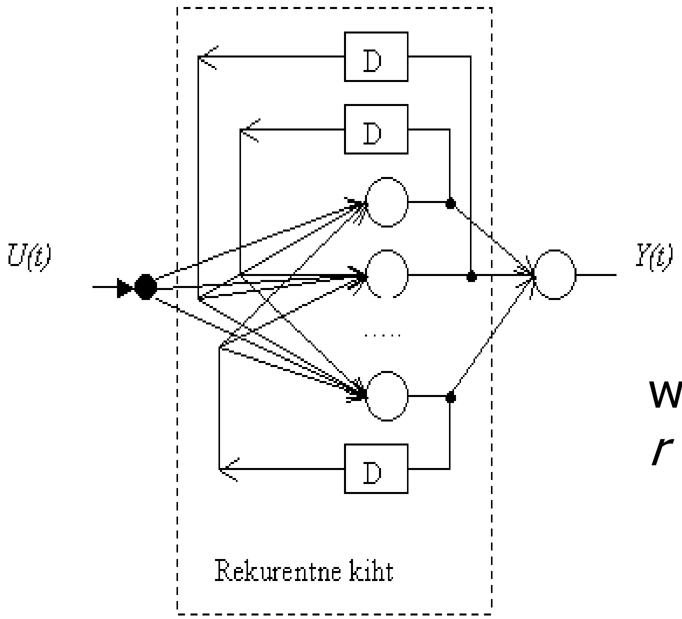


$$y(t) = F_2 \cdot (W_2 \cdot (F_1 \cdot (W_1 \cdot \begin{bmatrix} u(t) \\ \vdots \\ u(t-m) \\ y(t-1) \\ \vdots \\ y(t-n) \end{bmatrix} + \Theta_1) + \Theta_2) =$$

$$= f_{nn}(u(t), \ldots, u(t-m), y(t-1), \ldots, y(t-n))$$

1918 TALLINNA TEHNIKAÜLIKOOL

Feedback Neural Networks. Identification of dynamic systems (2)



Elman network

Internal feedback:

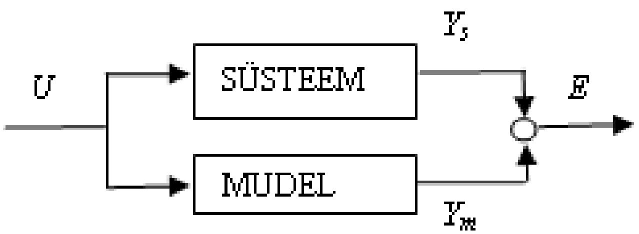
$$W_r = \begin{bmatrix} w_{11} & \cdots & w_{1r} \\ \vdots & \ddots & \vdots \\ w_{r1} & \cdots & w_{rr} \end{bmatrix}$$

where

r is the number of neurons on the recurrent layer;

$$\begin{cases} Y(t) = f_{nn}(U(t), X(t), W, B) \\ X(t) = Y_h(t-1) \end{cases}$$

Identification with ANNs (1)



$$E = Y_s - Y_m$$

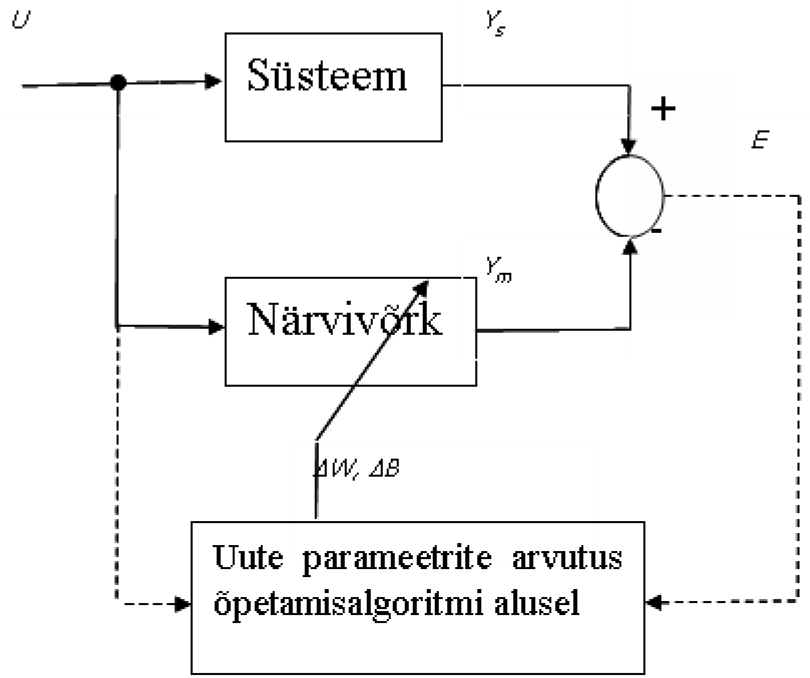
$$E \to 0$$

U is the input of the system

Y_s is the output of the system

Y_m is the output of the model

Identification with ANNs (2)



Identification with ANNs (3)

1. Getting experimental training data: Input signal is given to the system and the corresponding output reaction is measured. Obtained input-output data is used to train a NN-based model;
2. Choosing NN's structure: number of inputs, outputs, hidden layers, neurons; activation functions;
3. Choosing initial weighting coefficients (usually random).

Identification with ANNs (3)

4. Calculation of NN outputs corresponding to the training input set;
5. Comparison of outputs with reference outputs from the training set – calculation of model error;
6. Calculation of uptadeted weighting coefficients using selected training algorithm;

Repeat steps 4-6 untill a predefined number of iterations (epochs) is reached or until the desired accuracy is achieved.

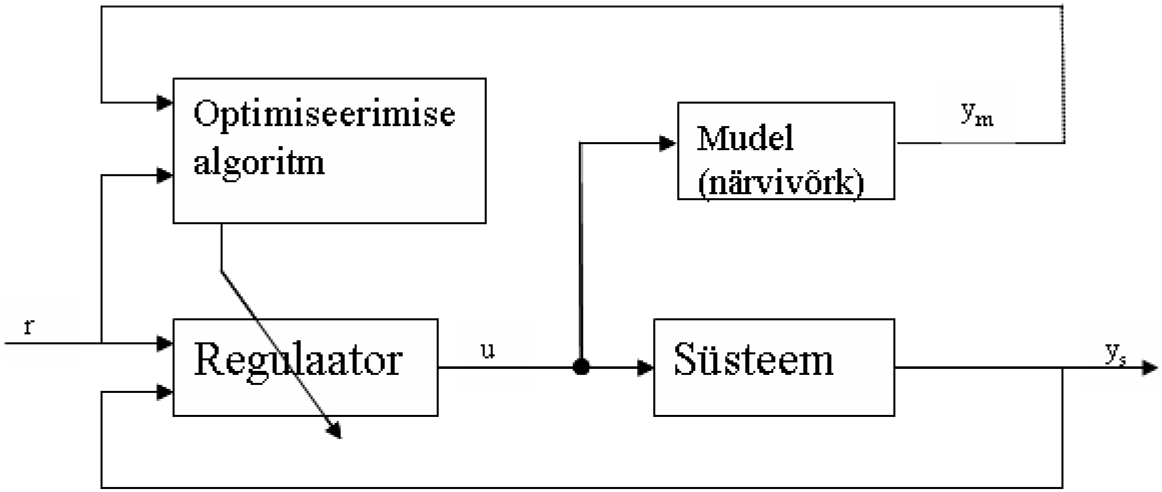
Dünaamiliste süsteemide juhtimine tehisnärvivõrkudega

NN-based control of dynamic systems

Eduard Petlenkov, Automaatikainstituut, 2015

1918 TALLINNA TEHNIKAÜLIKOOL

Predictive Control

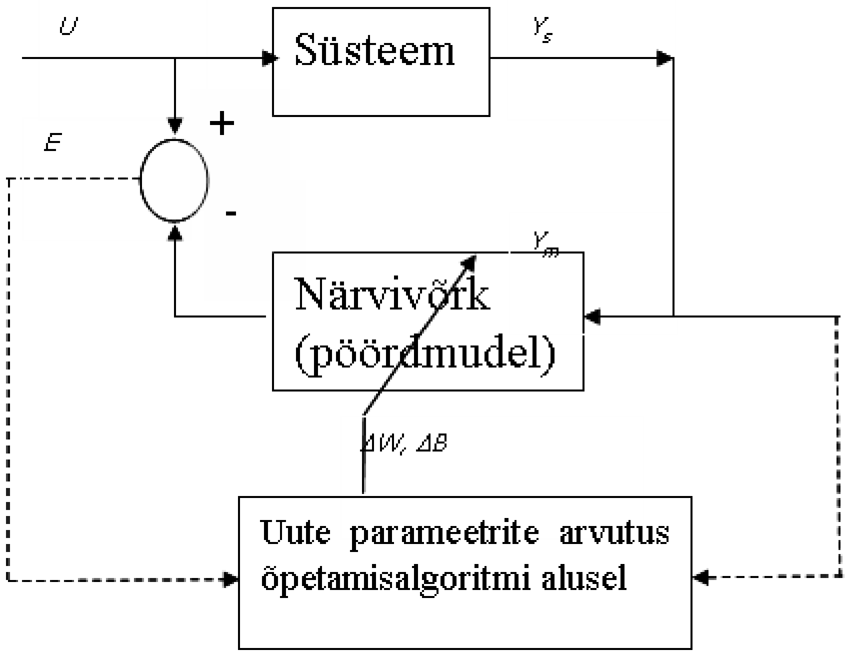


$$J(t)=\sum_{j=N_1}^{N_2}(r(t+j)-y^m(t+j))^2+\sum_{j=1}^{N_u}\lambda_j(\hat{u}(t+j-1)-\hat{u}(t+j-2))^2\rightarrow \min$$

$$u(t)=u(t-1)+\begin{bmatrix}q_1 & \dots & q_{N_u}\end{bmatrix}\cdot\begin{bmatrix}r(t)-y^m(t+N_1)\\ \vdots \\ r(t)-y^m(t+N_2)\end{bmatrix}$$

TALLINNA TEHNIKAÜLIKOOL 1918

Inverse model

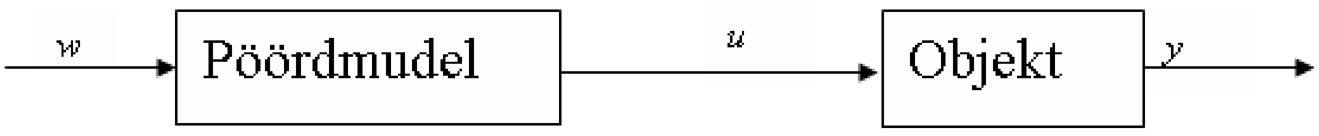


Inverse model

$$f^{-1} : Y(t) \rightarrow U(t-1)$$

$$U(t-1) = f^{-1}(Y(t),\ldots,Y(t-q),U(t-2),\ldots,U(t-r))$$

Inverse model based control



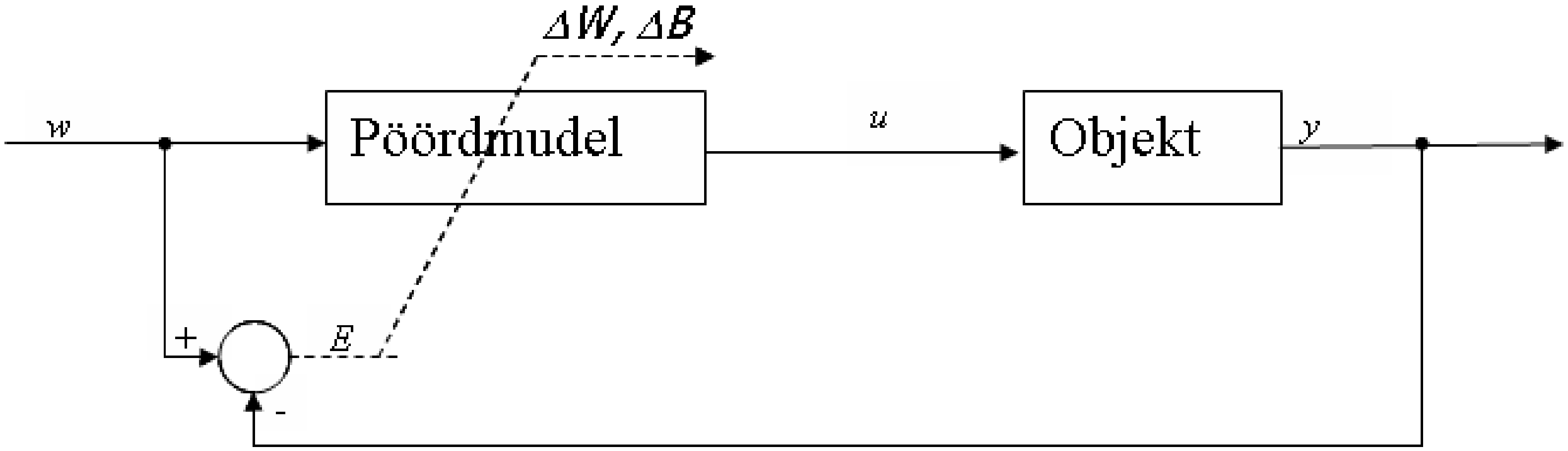
$$y(t) = f(y(t-1), y(t-2), u(t-1)u(t-2))$$

$$u(t-1) = f^{-1}(y(t), y(t-1), y(t-2)u(t-2))$$

Here $y(t) = w(t)$

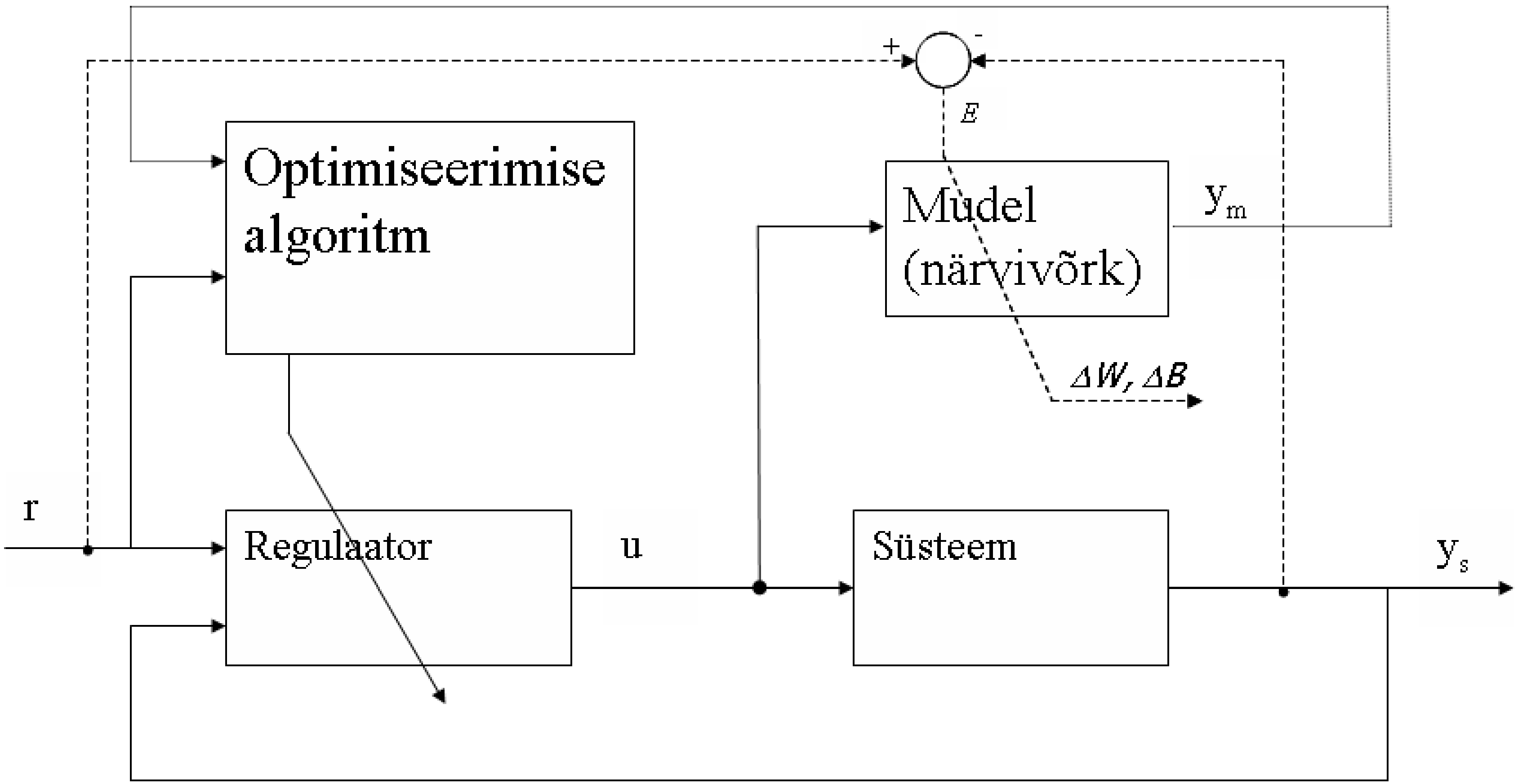
TALLINNA TEHNIKAÜLIKOOL 1918

Adaptive inverse model based control



$$E \to 0$$

Adaptive Predictive Control



1918
TALLINNA
TEHNIKAÜLIKOOL

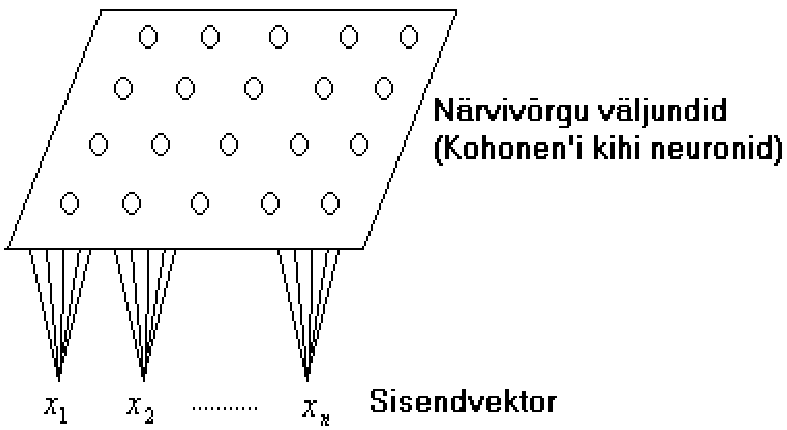
Mustrite tuvastamine ja klassifitseerimine tehisnärvivõrkudega

NN-based pattern recognition and classification

Eduard Petlenkov, Automaatikainstituut, 2015

Self-organizing networks

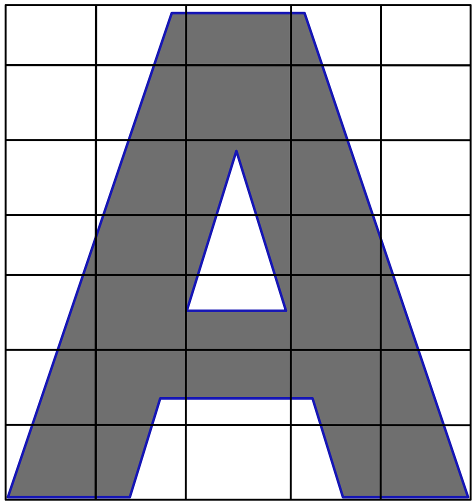
Self-orgamizing network is capable of adjusting its parameters on the basis of inputs and chosen optimization criteria.
It doesn't have any reference outputs



Kohonen map

$$d_j = \sum_{i=1}^{N} (x_i - W_{ij}(t))^2, \quad j = 1, \ldots, M$$

Näide 1: Character recognition(1)



$$A=\begin{bmatrix}0&0&1&0&0\\0&1&0&1&0\\0&1&0&1&0\\1&0&0&0&1\\1&1&1&1&1\\1&0&0&0&1\\1&0&0&0&1\end{bmatrix}$$

$$LetterA=\begin{bmatrix}0\\0\\1\\0\\0\\0\\1\\0\\1\\0\\\vdots\\1\end{bmatrix}$$

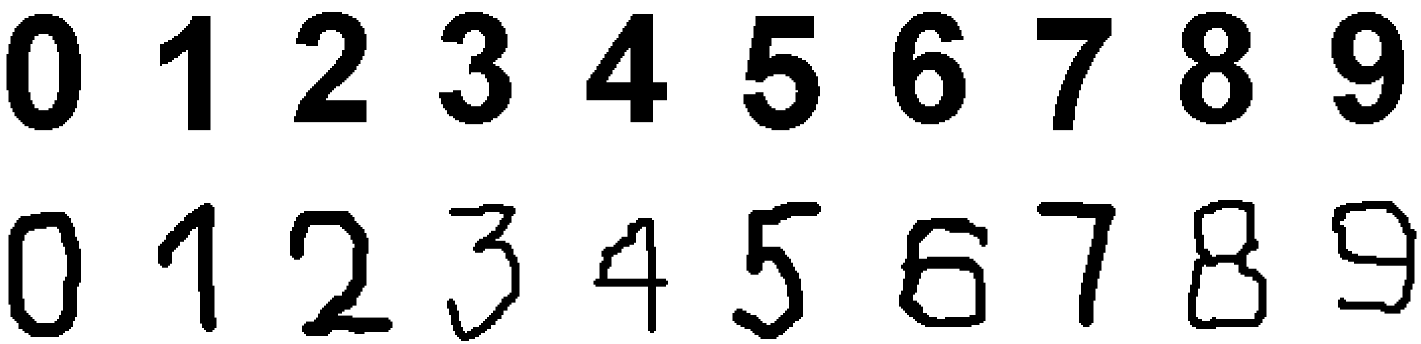
$$B=\begin{bmatrix}1&1&1&1&0\\1&0&0&0&1\\1&0&0&0&1\\1&1&1&1&0\\1&0&0&0&1\\1&0&0&0&1\\1&1&1&1&0\end{bmatrix}$$

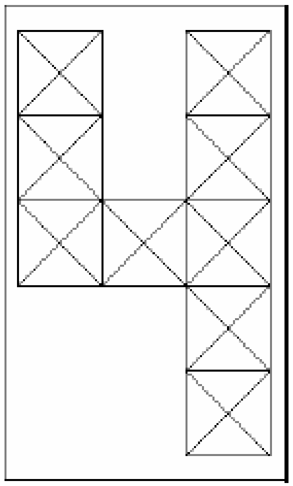
$$LetterB=\begin{bmatrix}1\\1\\1\\1\\0\\1\\0\\0\\0\\1\\\vdots\\0\end{bmatrix}$$

Näide 1: Character recognition (2)

$$
Input = \begin{bmatrix} 0 & 1 & \\ 0 & 1 & \\ 1 & 1 & \\ 0 & 1 & \cdots \\ 0 & 0 & \\ 0 & 1 & \\ 1 & 0 & \\ 0 & 0 & \\ 1 & 0 & \cdots \\ 0 & 1 & \\ \vdots & \vdots & \\ 1 & 0 & \end{bmatrix} \Longleftrightarrow Etalon = \left.\begin{bmatrix} 1 & 0 & & 0 \\ 0 & 1 & & 0 \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & & 0 \\ 0 & 0 & & 0 \\ 0 & 0 & & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & & 1 \end{bmatrix}\right\} \begin{array}{c} \text{väljundite arv} = \\ \text{erinevate kujundite (klasside) arv} \end{array}
$$

Näide 2: number recognition



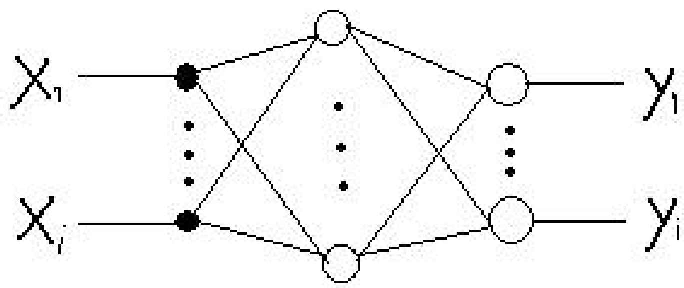


1	0	1
1	0	1
1	1	1
0	0	1
0	0	1

$$Number4 = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Näide 2: Number recognition– NN structure

```
net = newff(minmax(P),[25 10],{'logsig' 'logsig'},'traingda')
```




```
Etalon_1=[1 0 0 0 0 0 0 0 0 0]';
Etalon_2=[0 1 0 0 0 0 0 0 0 0]';
Etalon_3=[0 0 1 0 0 0 0 0 0 0]';
Etalon_4=[0 0 0 1 0 0 0 0 0 0]';
Etalon_5=[0 0 0 0 1 0 0 0 0 0]';
Etalon_6=[0 0 0 0 0 1 0 0 0 0]';
Etalon_7=[0 0 0 0 0 0 1 0 0 0]';
Etalon_8=[0 0 0 0 0 0 0 1 0 0]';
Etalon_9=[0 0 0 0 0 0 0 0 1 0]';
Etalon_0=[0 0 0 0 0 0 0 0 0 1]';
```

- net.trainParam.goal = 0
- net.trainParam.epochs=10000

TALLINNA TEHNIKAÜLIKOOL 1918

Näide 2: Number recognition– testing on the training set

	1	2	3	4	5	6	7	8	9	0
1	**0.9999**	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	0.0000	**0.9999**	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.0000	0.0000	**0.9999**	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	0.0000	**0.9999**	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	0.0000	**0.9999**	0.0000	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000	0.0000	**0.9999**	0.0000	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	**0.9999**	0.0000	0.0000	0.0000
8	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	**0.9999**	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	**0.9999**	0.0000
10	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	**0.9999**

Näide 2: Number recognition– testing on the validation set

	1	2	3	4	5	6	7	8	9	0
1	***0,0668***	0,0006	0,0066	0,0026	0,0089	0,0152	0,0034	0,0000	0,0050	0,0001
2	0,0019	**0,1548**	0,0014	0,0005	0,0108	0,0005	0,0044	0,0001	0,0000	0,0013
3	0,0145	0,0002	***0,1495***	0,0001	0,0032	0,0343	0,0081	0,0001	0,0039	0,0001
4	0,0003	0,0026	0,0001	***0,0102***	0,0004	0,0000	0,0014	0,0047	0,0008	0,0092
5	0,0000	0,0003	0,0000	0,0000	**0,0177**	0,0000	0,0000	0,0000	0,0000	0,0000
6	0,0001	0,0013	0,0020	0,0002	0,0005	**0,0611**	0,0002	0,0000	0,0065	0,0081
7	0,0002	0,0000	0,0002	0,0003	0,0001	0,0000	**0,0154**	0,0009	0,0000	0,0000
8	0,0000	0,1030	0,0000	0,0000	0,0000	0,0000	0,0000	***0,5868***	0,0000	0,0000
9	0,0002	0,0002	0,0003	0,0008	0,0580	0,0056	0,0006	0,0010	**0,2799**	0,0051
10	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	**0,0186**

Databases for benchmark problems

- **MNIST** database of handwritten digits
  - 26x26px
  - 60 000 labeled images and 10 000 test images

- **Semeion** - 1593 handwritten digits from around 80 persons
  - 16x16px