Pharmaceutical Process Optimization With Artificial Neural Networks (ANN)

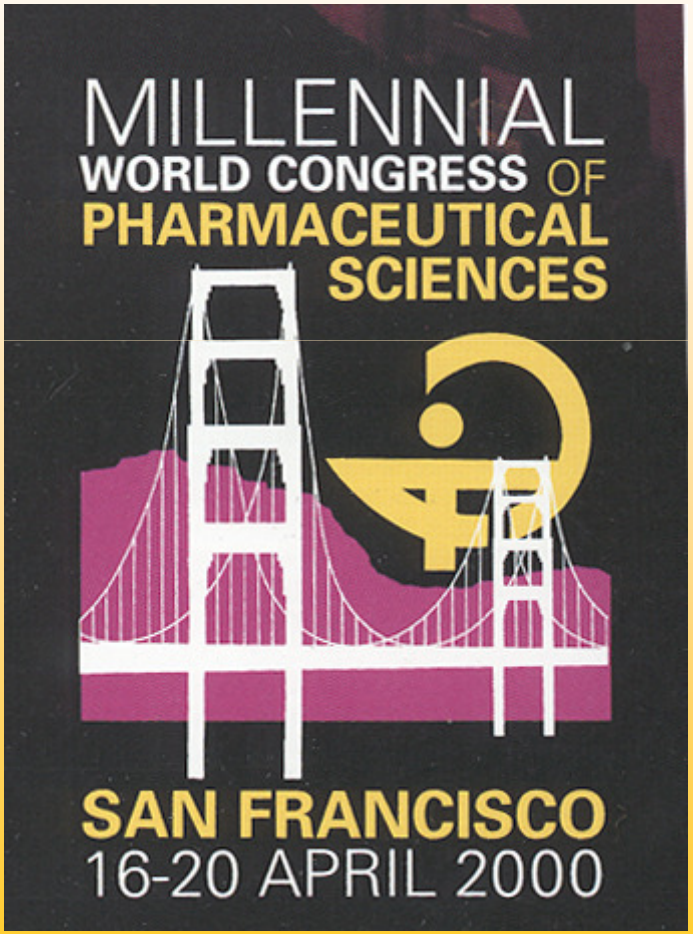
Hans Leuenberger

Pharmacenter of the University

Klingelbergstrasse 50

CH-4056 Basel, Switzerland

Hans.Leuenberger@unibas.ch



1. Optimization with artificial neural networks

1.1 basic concept of ANN

1.2 comparison with other methods used for process optimization

2. Advantages and limitations of ANN

3. Conclusions

CONCEPT OF A *BIOLOGICAL* NEURON CELL

- *Broad network*

- *Cell body*
- *Dendrites* and
- One *Axon*

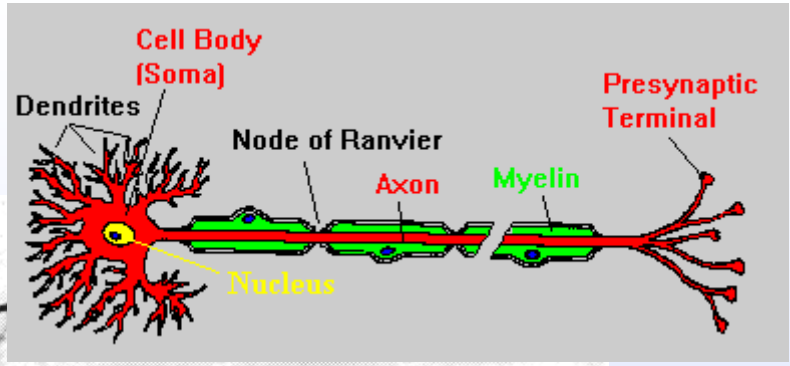
- *Synapse*

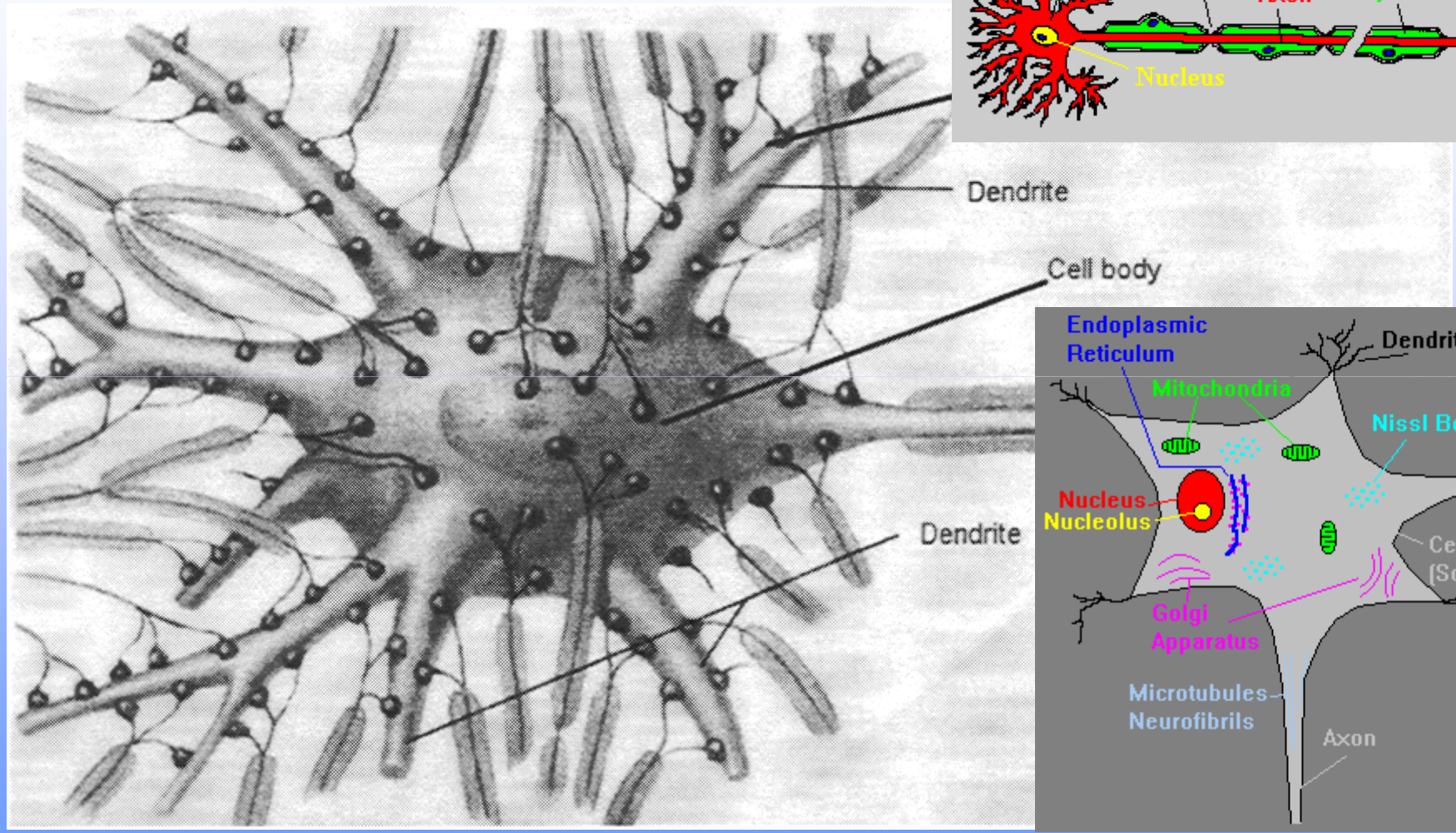
CONCEPT OF A *BIOLOGICAL* NEURON CELL

When the total sum of impulses surpasses a certain threshold, the neuron „fires“.

The effect of an impulse reaching a neuron can be excitatory *(„positive“)* or inhibitory *(„negative“).*

Fig.1. Neuron Cell





An artificial neuron is described as Processing Element (PE)

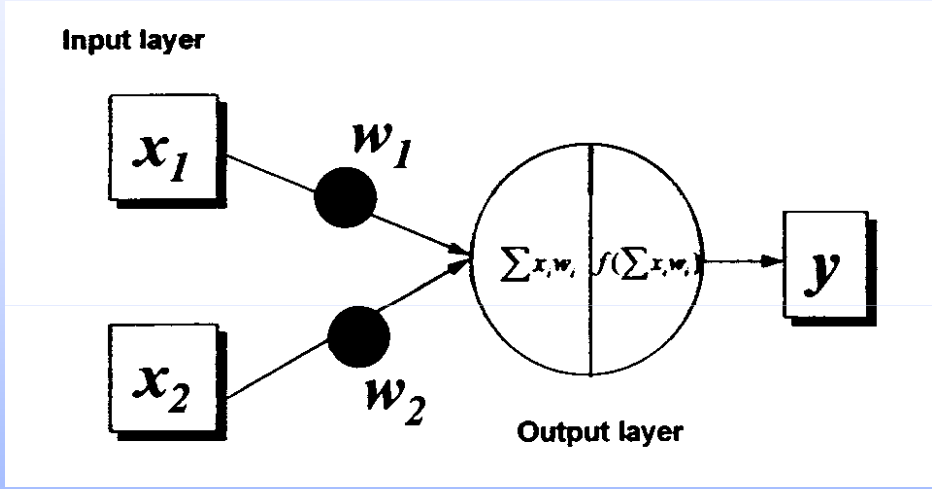


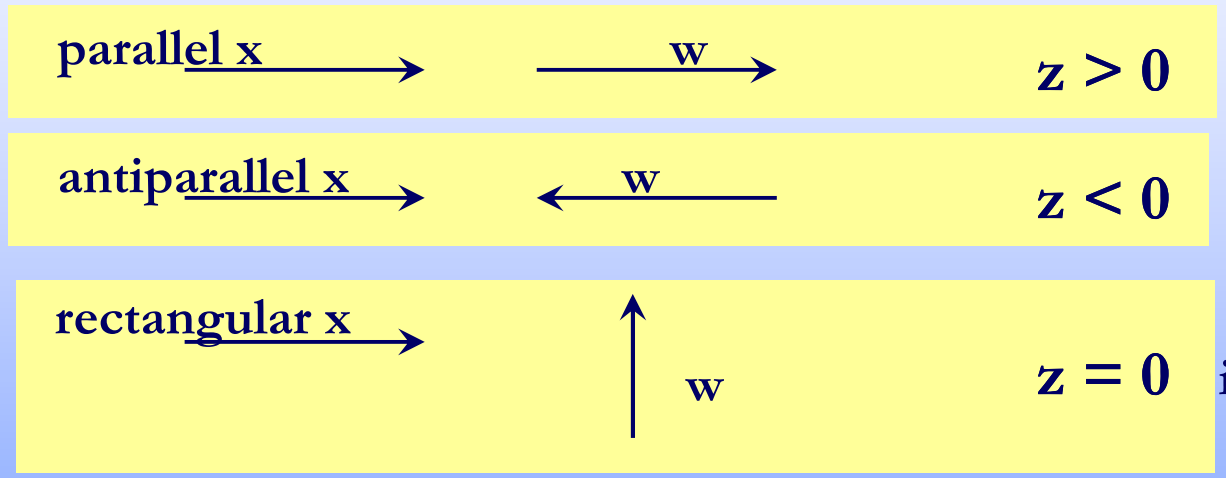
Fig.2

Perceptron with a single Processing Element (PE)

These pulses are summed

Input and weight values can be imagined as vectors:

The sum $z = \Sigma\, x_i\, w_i$ is the *inner product* of the vectors x and w!



if the directions of the vectors x and w are rectangular!

Thus the result of the transfer function (i.e. output) depends on the value of the sum z!

REMARK:

The *Input summation*

triggers

the *activation function*

in order to *fire* or *not to fire*

like the neuron in the brain.

SIMPLIFIED NETWORK PROCEDURE:

Activation/Transfer function

- sigmoidal function
- hyperbolic tangent function

Values between

- +1 exitatory
- -1 inhibitory

Output-signal

- 0 and 1
  or
- -1 and +1

The first derivative of the transfer function has to exist.

ACTIVATION- RESP. TRANSFERFUNCTION

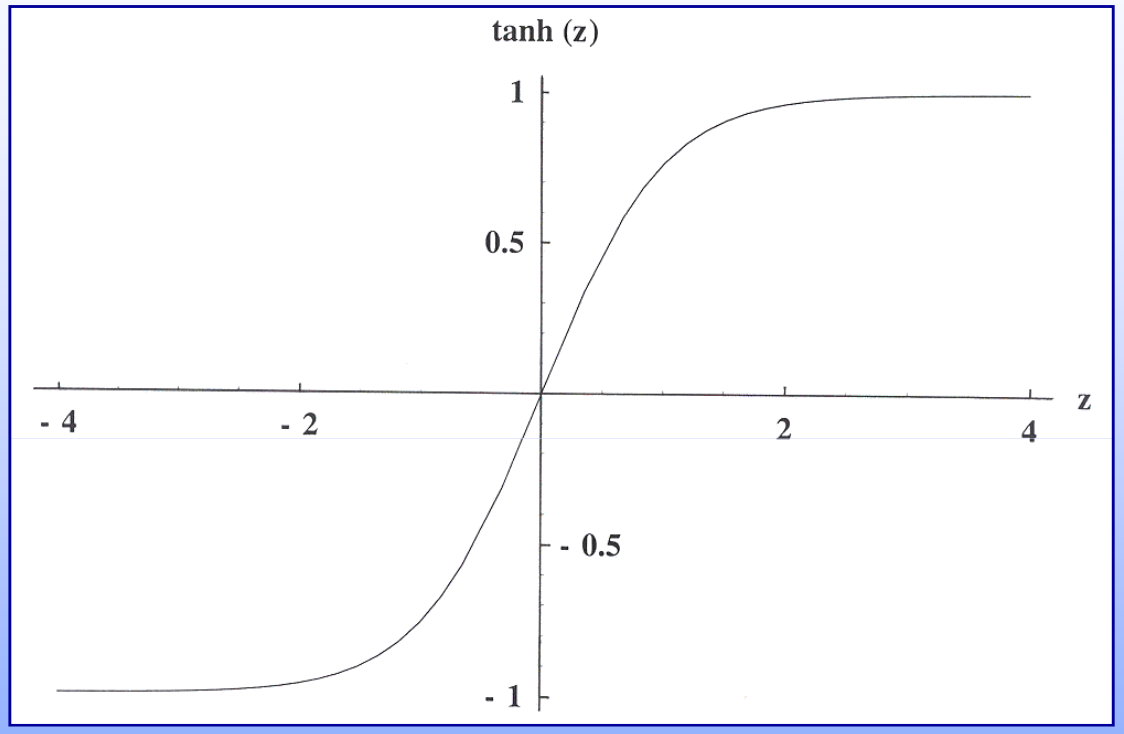
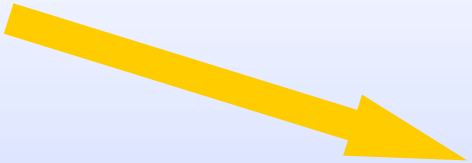


Fig.3

$$y = [\tanh (\Sigma x_i w_{si})]$$

RESULT:

$PE_s\ f(\Sigma\ x_i w_i)$ 

the final (output) value y_s

Is this result close to the target value t_s?

DELTA RULE BACK PROPAGATION OF ERRORS

Back-propagation procedure **adjusts** the weight w_s

starting from the final output layer backwards!

In order to **minimize** $(y_s - t_s)$ i.e.

the partial error $$E_s(at\ a\ PE_s) = \frac{1}{2}(y_s - t_s)^2$$

The difference between y_s and t_s is smaller if more layers are used.
Too many layers lead however to an overfitting!

$$E_s = \frac{1}{2}\left[f\left(\Sigma x_i w_{si}\right) - t_s\right]^2 \Downarrow \textit{Minimum!}$$

For this optimization the gradient descent method is normally used.

Thus the partial derivative of E_s has to be calculated for each weight w_k attached to PE_i.

Known from the last layer

$$\frac{\partial E_s}{\partial w_k} = \frac{\partial E_s}{\partial y_s} \cdot \frac{\partial y_s}{\partial z_s} \cdot \frac{\partial z_s}{\partial w_k} \qquad \text{i.e.} \qquad \frac{\partial E_s}{\partial w_k} = \left[f\left(z_s\right) - t_s\right] f'\left(z_s\right) x_{sk}$$

$$f\left(z_s\right) = \textit{Activation function} \qquad z_s = \Sigma x_i w_{si} \qquad y_s = f\left(z_s\right)$$

$$f'\left(z_s\right) = \textit{First derivative of the activation function}$$

In the gradient descent method the weight is changed as follows:

$$w_k = \frac{\eta \partial E_s}{\partial w_k}$$

η = *proportionality constant (learning rate)*

Step 1: Comparison between the target values t_s and the *output* values y_s of the *last* layer:
Minimization of E of the last layer: Thus *new weights* are assigned as *input* t o the last layer!

The procedure of the Delta Rule Back Propagation of Errors includes the following steps:

Step 2: The transfer-function of the sum of these new weights w_{sk} multiplied by x_{ik} correspond to the *„new" output* values of the PE_k of the *preceding layer*. Now the preceding layer can be treated in the same way! Thus „new" weights w_{sk} of the preceding layer can be calculated.

Step 3: etc: in analogy to step 2!

Final Step: After having adjusted all the weights one cycle of the back-propogation procedure has been completed and the second iteration with the new weights can be started!

Today excellent software exists using specialized methods to minimize the error E_s.

Which values are of interest?
The values w_i or the goodness of fit between y_s and t_s ?
BOTH!

In case of an *excellent fit*, the *ANN model* can be established for *predicting* the *result* y_k of another set of input variables x_{ik}.

The knowledge of the *weight* w_i attributed to the *variable* x_i is of special interest *analyzing* a process or formulation!

ANN

„black box“

Input layer

a number of *hidden layers of processing elements*

Output layer

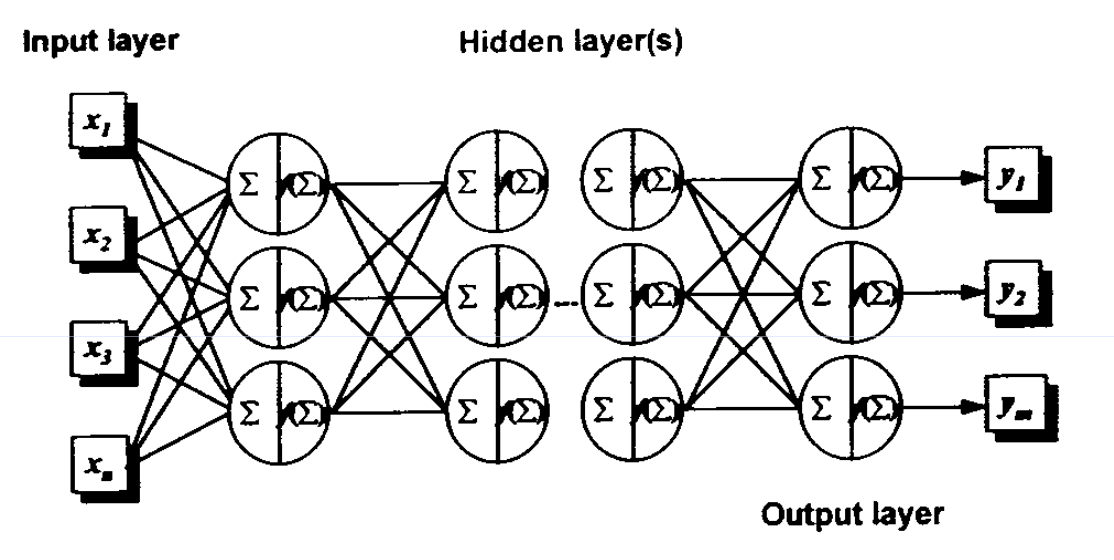


Fig.4

MULTI-LAYER PERCEPTRON

The Input Layer describes ...

... the independent values of the variables x_{sm} (m = 1...p) fixed for a certain experiment s, which yield the output y_s (= result) in the output layer.

In each processing element the sum $z = \Sigma x_{sm} w_m$ is calculated to be used in the activation function and the weights are determined in order to receive an output y_s close to the target value t_s.

To summarize the ANN computing procedure:

For each Processing Element PE in a layer a partial error E_s can be defined and minimized by an approriate mathematical tool such as the gradient descent method. This procedure has to be repeated for successively preceding layers.

This method is generally known as Delta Rule Back-Propagation of Errors! The back propagation procedure repeatedly adjusts the weights of connections in the network!

For the determination of the individual weights w_m a sufficiently large data set is needed, i.e. a learning set:

Training Data Set					Network's Output
Input Data				Target Data	
x_1	x_2	...	x_p	t	y
x_{11}	x_{12}	...	x_{1p}	t_1	y_1
.				.	
.				.	
x_{s1}	x_{s2}	...	x_{sp}	t_s	y_s
.				.	
.				.	
x_{n1}	x_{n2}	...	x_{np}	t_n	y_n

Fig. 5

DATA SET INDICES

1.2 Comparison with other methods used for process optimization

1.2.1 classical experimental design

1.2.2 factorial design (statistical designs)

1.2.3 central composite design (higher order designs)

1.2.4 simplex design

1.2.5 true physical (mechanistic) models…

1.2.6 empirical models, overfitting

1.2.7 application of percolation theory

1.2.8 the black box model (convolution/deconvolution model)

1.2.1 CLASSICAL EXPERIMENTAL DESIGN

- Only one Factor a time is studied.
- Interactions between factors cannot be detected, thus ...
- The results depend on the location of the experiment in the space of the variables.
- Thus the results can advance misleading conclusions leading to a time consuming optimization process!

1.2.2 FACTORIAL DESIGN (Statistical Designs)

- More than one Factor a time is studied.
- Interactions between factors can be quantified!
- The results depend less on the location of the experiment in the space of the variables.
- Thus the results are in general more reliable.

Example: Comparison between a classical and a statistical, i.e. 2^2 Design

% Drug dissolved of a tablet as a function of the amount of cornstarch (Factor A) and compressional pressure (Factor B):

Classical Experiments:

A) Effect of a higher amount of cornstarch, keeping the pressure at the lower level: No change in % Drug dissolved!

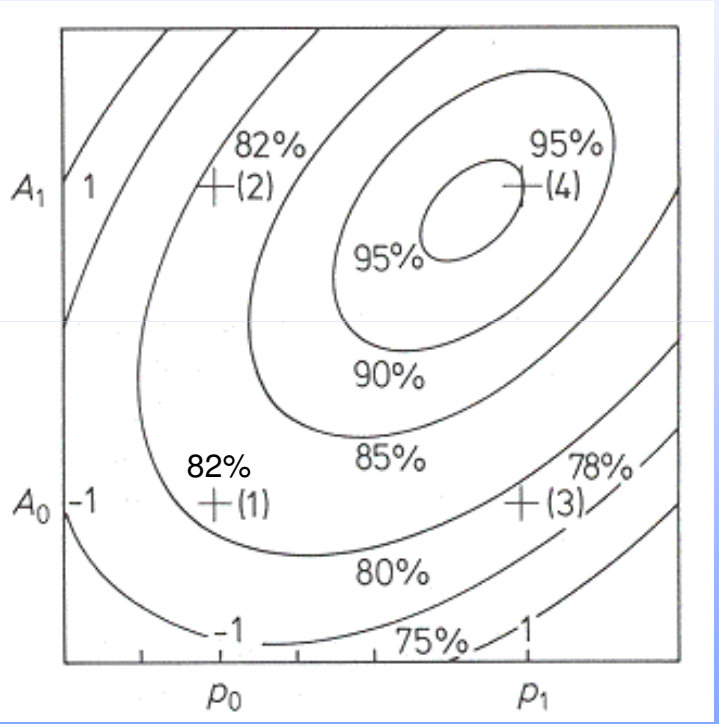
B) Effect of a higher pressure: % Drug dissolved is lower!

CONCLUSION 1: **Reduction of the compressional pressure to avoid a possible capping tendency keeping the amount of cornstarch at the lower level!**

RESULT OF CONCLUSION 1: **The % of Drug dissolved is again lowered! What happened? The classical design of experiments lead to a wrong conclusion!**

THE REAL SITUATION: **% Drug dissolved as a function of the factors A (cornstarch) and B (pressure) according to the following contour plot:**

THE REAL SITUATION: **% Drug dissolved as a function of the factors A (cornstarch) and B (pressure) according to the following contour plot:**



Contour Plot

Fig. 6

CONCLUSIONS: **The amount of cornstarch and the pressure need to be increased for the optimization of the dissolution rate!**

THE STATISTICAL 2^2 DESIGN

(based on the same example as before)

FACTOR X_1 = A
(cornstarch)
lower level $= -1$
upper level $= +1$

FACTOR X_2 = B
(pressure)
lower level $= -1$
upper level $= +1$

A = Effect of X_1
B = Effect of X_2
AB = Interaction
Y_i = Experim. Result
T = Total
T/4 = Mean

EXPERIMENTAL PLAN (YATES):	A	B	AB	Y_i	%
1) both factors at the lower level	-	-	+	+	82
a) Factor A at the upper level	+	-	-	+	82
b) Factor B at the upper level	-	+	-	+	78
ab) both factors at the upper level	+	+	+	+	95
Sum (+ -)	2A	2B	2AB	T	

RESULTS of the $2^2 = 4$ Experiments:

i = 1, ... , 4

$$y_i = \frac{T}{4} + \left(\frac{A}{2}\right) x_{1i} + \left(\frac{B}{2}\right) x_{2i} + \left(\frac{AB}{2}\right) x_{1i} x_{2i}$$

$$x_{1i} = +1 \; oder \; -1$$

$$x_{2i} = +1 \; oder \; -1$$

COMMENTS on the Result of the 2^2 Design:

The Summarizing Equation (SE) is an approximation comparable to the *Taylor expansion* of the true (mechanistic) function $y = f(x_1, x_2)$:

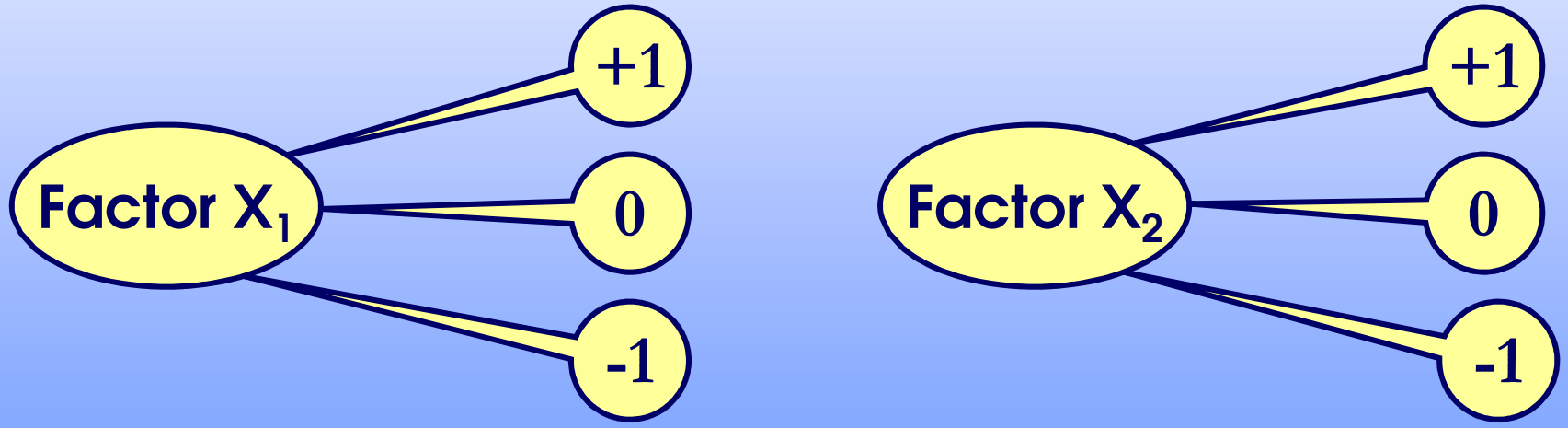
$$y(x_1, x_2) = y(x_1, x_2 = 0) + \left(\frac{\partial y}{\partial x_1}\right) x_1 + \left(\frac{\partial y}{\partial x_2}\right) x_2 + 2\left(\frac{\partial y^2}{\partial x_1 \partial x_2}\right) x_1 x_2$$

$$SE: y_i = \frac{T}{4} + \frac{A}{2} x_{1i} + \frac{B}{2} x_{2i} + \frac{AB}{2} x_{1i} x_{2i}$$

The true function is approximated by the tangents, resp. tangential plane (first order derivative), resp. by a tangential curved surface (degenerate second order derivative).

1.2.3 CENTRAL COMPOSITE DESIGN

A better approximation can be obtained with a higher order experimental design such as a Central Composite Design with three levels:



Comments on the Result of the Central Composite Design:

The Model Equation (ME) of the Central Composite Design Experiments:

$$ME: y(x_1, x_2) = a_0 + a_1 x_1 + a_2 x_2 + a_{12} x_1 x_2 + a_{11} x_1^2 + a_{22} x_2^2$$

is again an approximation similar to the following *Taylor expansion* of the (unknown) true function $y = f(x_1, x_2)$:

$$y(x_1, x_2) = y(x_1, x_2 = 0) + \left(\frac{\partial y}{\partial x_1}\right) x_1 + \left(\frac{\partial y}{\partial x_2}\right) x_2 + 2\left(\frac{\partial y^2}{\partial x_1 \partial x_2}\right) x_1 x_2 + \frac{1}{2}\left(\frac{\partial y^2}{\partial x_1^2}\right) x_1^2 + \frac{1}{2}\left(\frac{\partial y^2}{\partial x_2^2}\right) x_2^2$$

This approach corresponds to

the RSM Technique,

i.e. RESPONSE SURFACE METHODOLOGY,

often used in place of the ANN- Methodology!

Comments and Comparison to the Application of Artificial Neural Networks (ANN):

The RSM - Technique is a suitable alternative to the ANN - methodology.

However a number of preliminary experiments is necessary to establish a correct central composite design, which does not enter directly into the final evaluation contrary to the ANN-methodology, where in general the results of all experiments are taken into account.

General comment: In the majority of the experiments the RSM - Technique yields more or less the same result as the ANN - Methodology.

1.2.4 SIMPLEX DESIGN

The Simplex Design

is an extremely efficient method

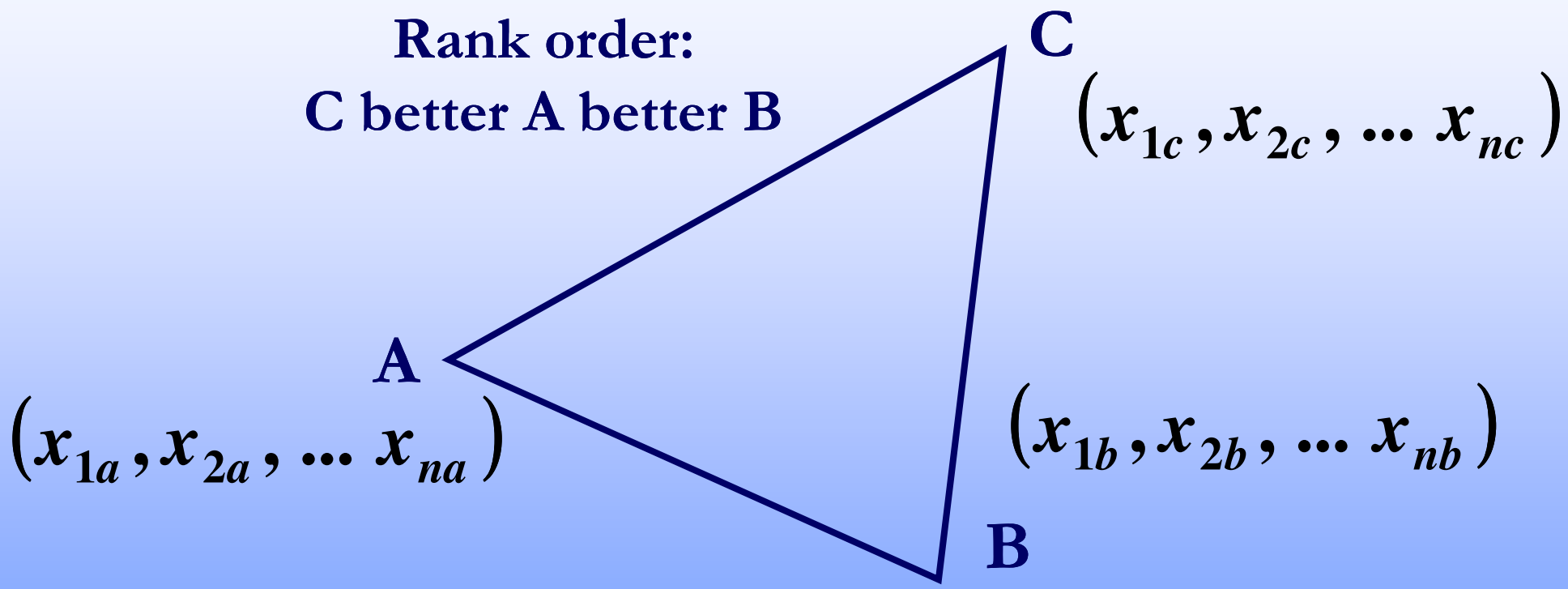
in the area of optimization

of processes and/or formulations.

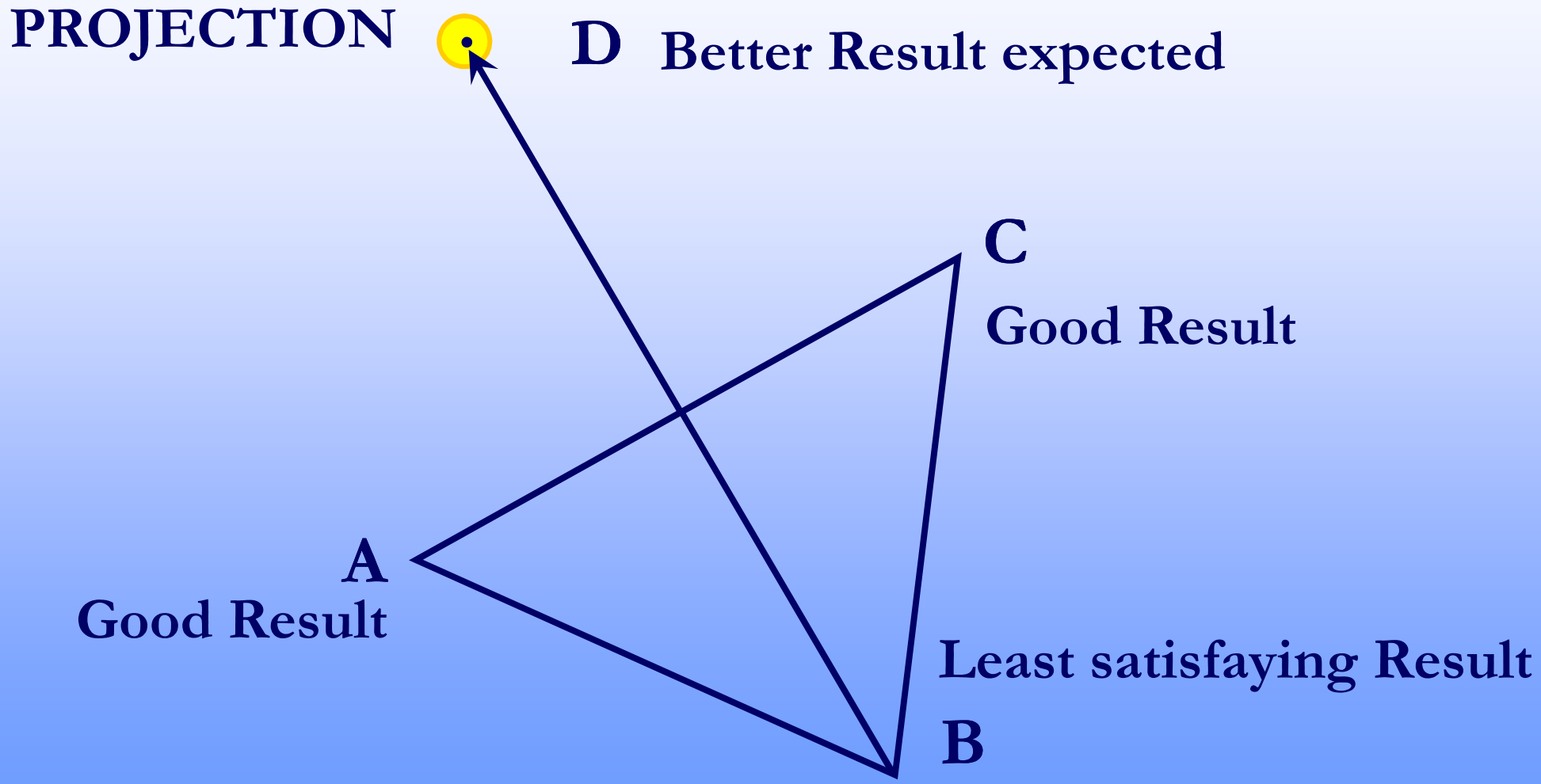
SIMPLEX DESIGN (Triangle)

Results:

Rank order:
C better A better B



SIMPLEX DESIGN:
Start Triangle and new experimental point D



Comparison with ANN Methodology

- The results of the experiments performed with a Simplex Design can be compiled in a list (e.g. EXCEL List) for further Evaluation with the ANN Methodology.
- Both approaches can be used simultaneously.
- The use of a Simplex Design cannot be recommended if the necessary experiments per triangle are very time consuming (e.g. a stability test). In such a case a factorial design has to be preferred.

1.2.5 TRUE PHYSICAL (MECHANISTIC) MODELS...

would have the advantage

to be valid in a broad range of the variables.

Thus an *extrapolation* would be *less problematic*.

Unfortunately mechanistic models are rare in the area of pharmaceutical technology. If a true physical model is known, it should be preferentially used for an optimization.

1.2.6 EMPIRICAL MODELS, OVERFITTING

Experimental results $Y(x_1,x_2)$
can be fitted by a mechanistic or an empirical model.
If the exact (mechanistic) model is not known,
a power series is often applied:

$$y(x_1,x_2)=a_0+a_1x_1+a_2x_2+a_{12}x_1x_2+a_{11}x_1^2+a_{22}x_2^2$$

Comments: Models and number of parameters

If more parameters are included in a model.

fit of the experimental data is obtained

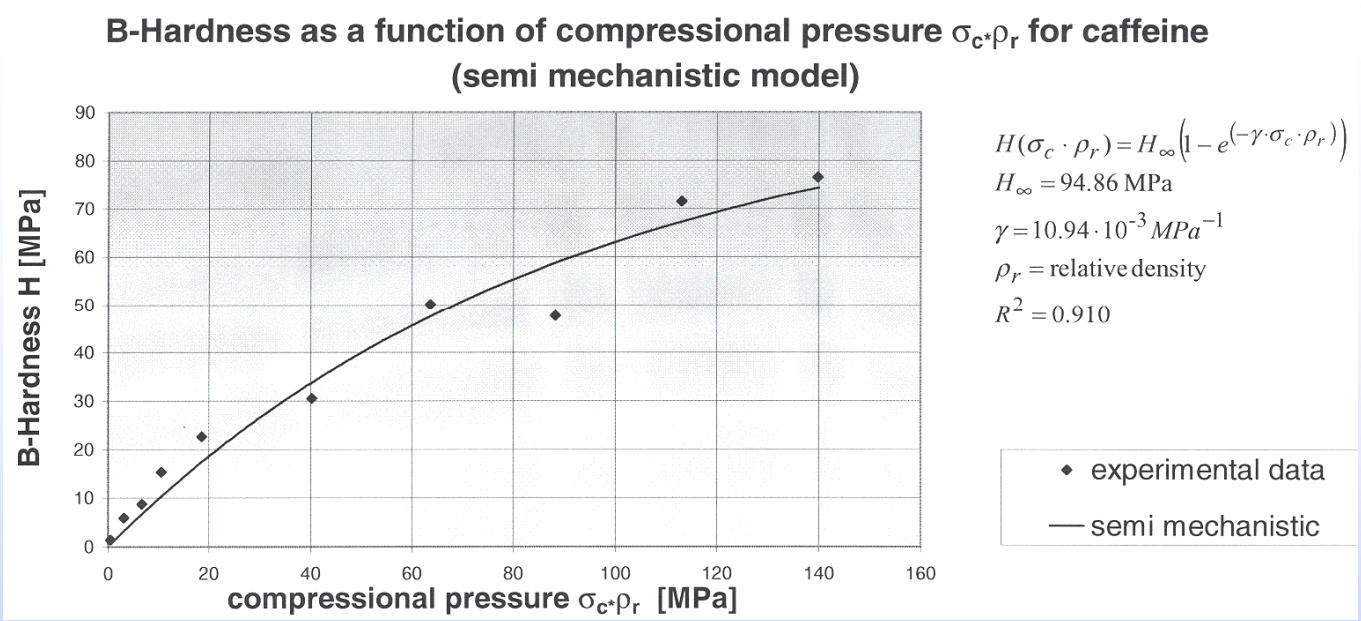
However such an approach is completely wrong

as the experimental data exhibit a normal scatter due to the statistical error!

The inclusion of too many parameters lead to an overfitting!

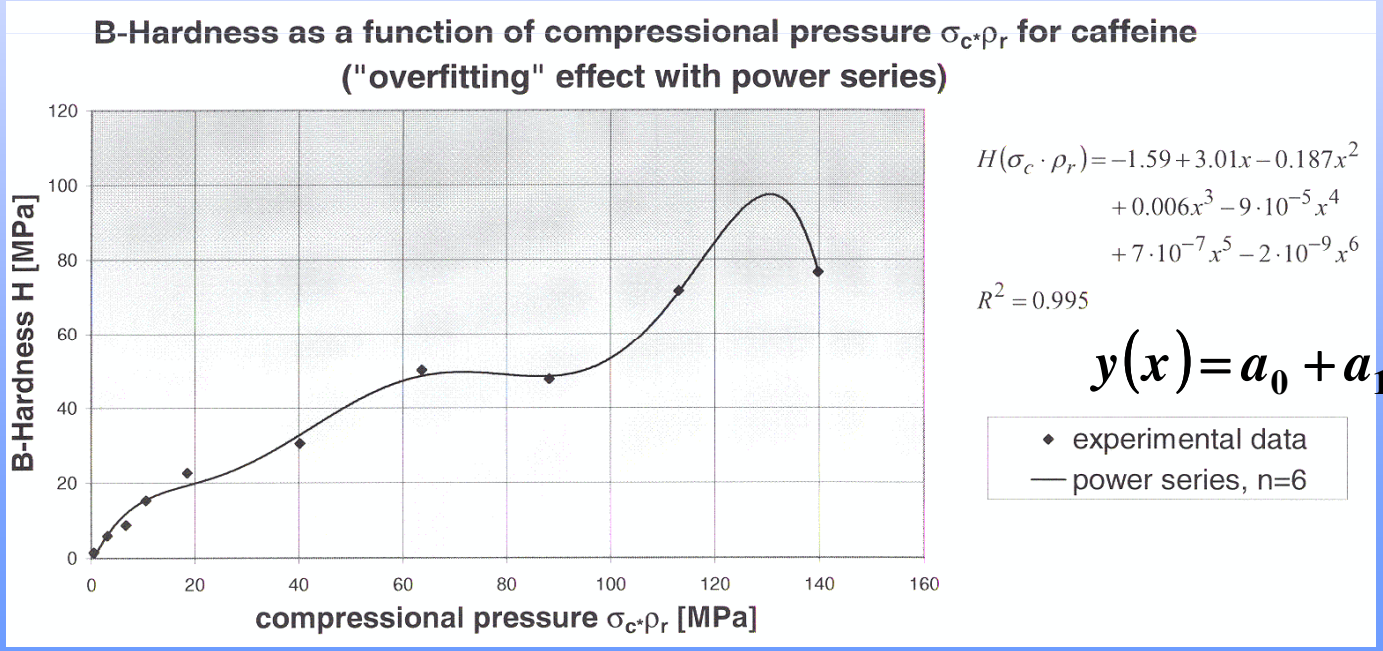
Plot: Hardness

H= y(x) as a function of compressional stress x



$$y(x) = H_\infty\left(1 - \exp(-\gamma x)\right)$$

Fig. 7



$$y(x) = a_0 + a_1 x^1 + a_2 x^2 + a_3 x^3 \ldots + a_n x^n$$

Fig. 8

General Comments on Classical Modeling/ANN:

the result
$y = f(x)$ as an
effect of varying
the factor x

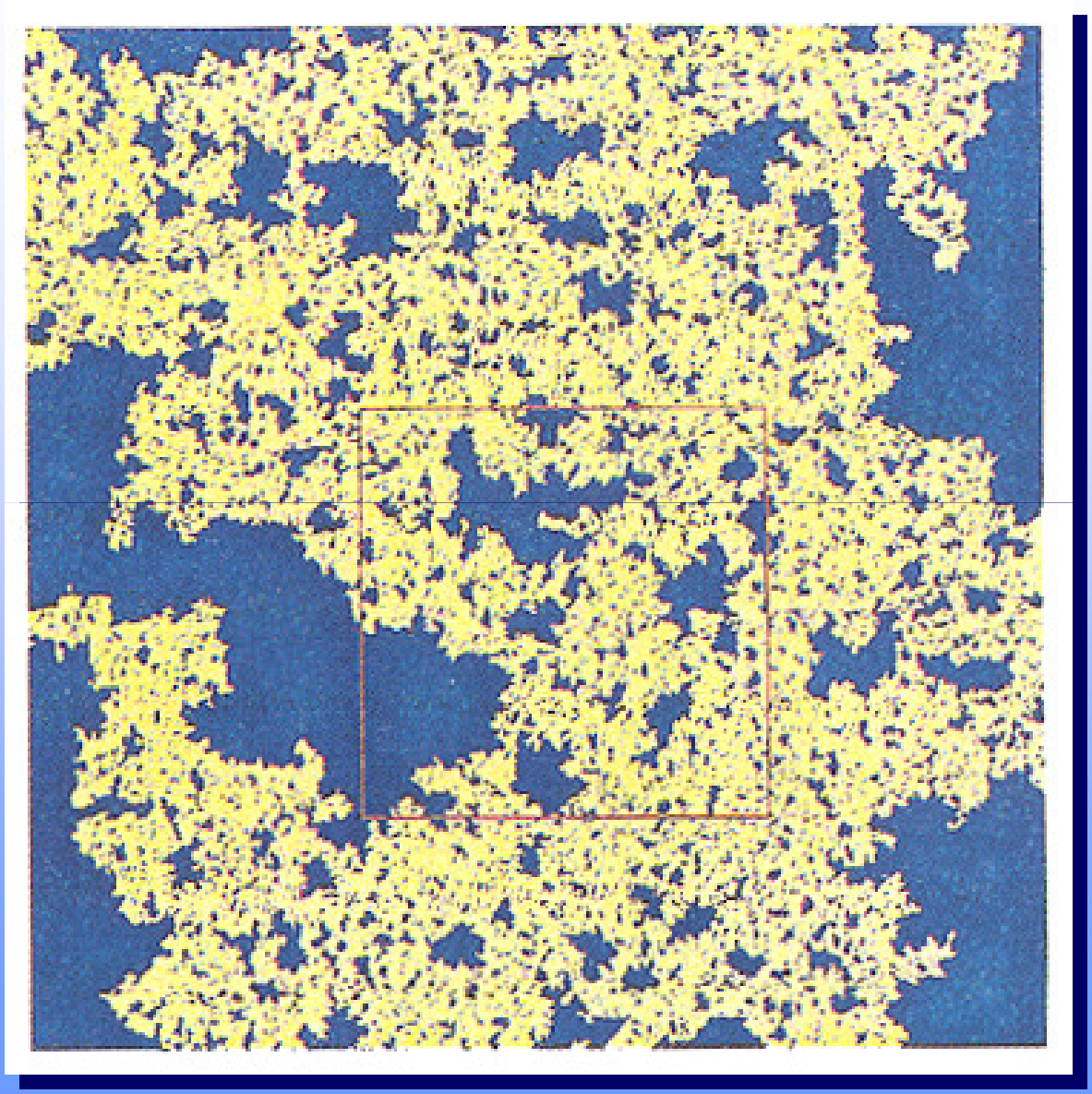
can be described as

- a linear
- a quadratic
- a ln-linear
- a ln-ln

MODEL

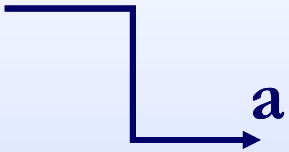
- Do we have an overfitting of the data?
- Question of experimental error and lack of fit!
  Problem of extrapolation!
- Analogues questions are necessary in case of ANN:
  Input data: x, ln x Output data: y, ln y

1.2.7 APPLICATION OF PERCOLATION THEORY



Percolation theory deals among other items with geometrical phase transitions.

The continuous addition of water to an appropriate W/O emulsion induces as an example



a geometrical phase transition which results in a O/W emulsion.

It is evident that certain properties of the system

e.g. electrical conductivity of the O/W-emulsion

suffer a dramatic change at the critical concentration,

i.e. at the so called percolation threshold p_c,

where the geometrical phase transition occurs.

Thus close to the percolation threshold p_c the following model equation of percolation theory holds for the conductivity:

$$Y(p) = S\,(p - p_c)^q$$

S = Scaling factor
p = V/V concentration of water
p_c = critical concentration of water
q = critical exponent

1.2.8 THE BLACK BOX MODEL
(Convolution/Deconvolution Model)

needs input and output data

similar to the Artificial Neural Network.

The „Black Box“ is however different from an ANN.

In principle the answer of a special *input function* is compared to the answer of the input function of interest.

A typical example is the determination of the in-vivo dissolution rate of a orally administered controlled release dosage form.

The black box is represented by  the test person

The special input function

is a drug solution,

i.e. a Deltafunction

which describes the situation that

100% of the drug

has been dissolved

in an infinite short time.

A drug solution

creates as an answer

the plasma drug concentration profile of the test person.

After the administration of a controlled release dosage form

an other plasi as an output function. profile results

This outpunt function

can be explained as the result of a sequential administration of Deltafunctions with different weights.

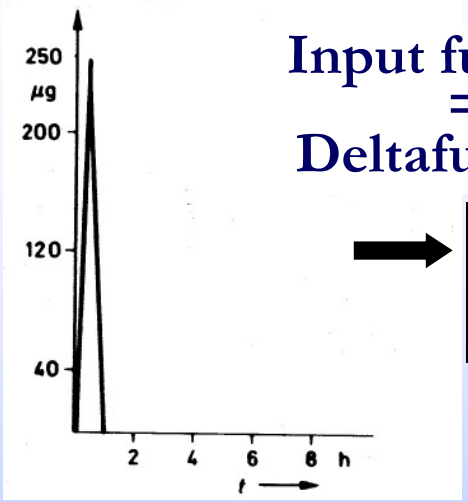
The superposition of the answers of the individual Deltafunctions

lead to

the outputfunction of the controlled release dosage form.

Black Box Model (Diagrammatic Representation)

Dissolution rate



Input function
=
Deltafunction

The black box

Output function
=
Transfer function

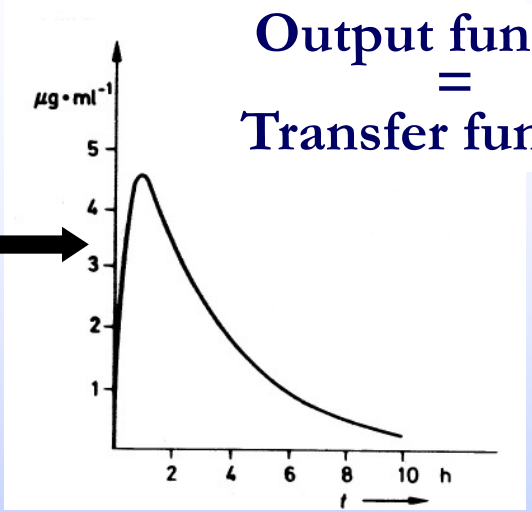
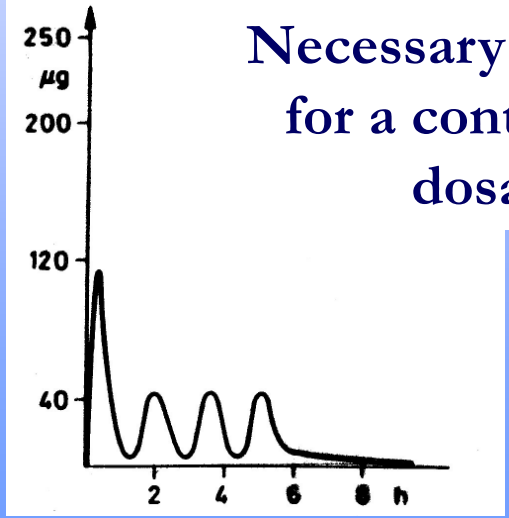


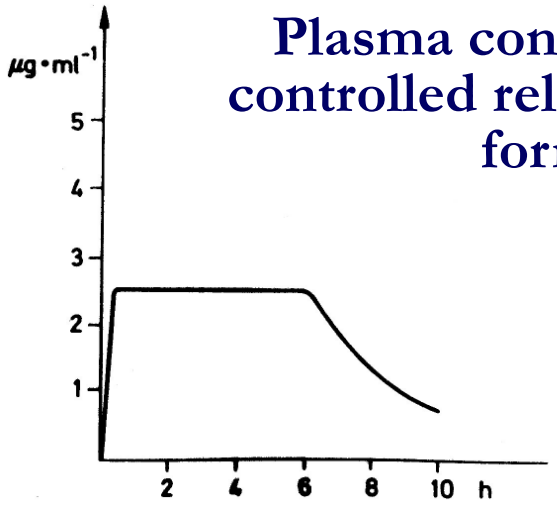
Fig. 9

Fig. 10



Necessary Input-function for a controlled release dosage form

Fig. 11



Plasma concentration controlled release dosage form

Fig. 12

Comments and comparison to the ANN -Methodology

A comparison is difficult and somehow stressed:

What is in common?

1) The system, i.e. the „Black Box“ needs one „learning step“, which consists in knowing the result, i.e the output of a Deltafunction as input.

Comments and comparison to the ANN -Methodology

Convolution / Deconvolution

2) Together with the output data of interest (i.e. output function a(t)) it is possible to calculate the unknown input function, which has some similarities to the weights which are calculated with ANN to define the importance of the input factors.

Convolution Integral $$a(t) = \int_0^t e(\tau)\, a_\delta(t-\tau)\, d\tau$$

2. APPLICATION OF ARTIFICIAL NEURAL NETWORKS (ANN) IN PHARMACEUTICAL PROCESS OPTIMIZATION

2.1 Design of Networks [1]

The Design of ANN is considered as an art and depends on the goals to be achieved.

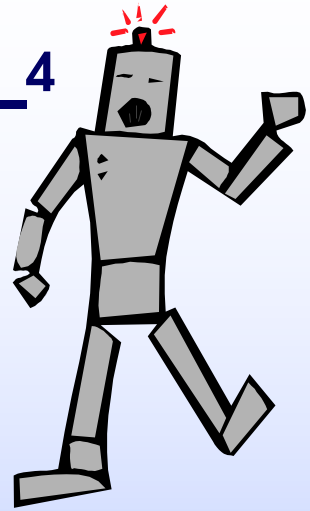
2.1 Design of Networks 2

- In case of a *supervised* learning step often a generalized feed forward multilayer perceptron is used GFF-MLP
- In case of an *unsupervised or only partially supervised* learning step (used for feature extracting networks) a hybrid network composed of a self-organizing feature map joined to a multilayer perceptron is generally used.

2.1 Design of Networks 3

- The *supervised* learning step needs a set of training data (input and output).
- In case of the *unsupervised or adaptive* training the network is provided only with input data but not with desired output values.

2.1 Design of Networks 4



- Totally unsupervised learning
  is not yet well understood and
  is studied in special labs to create robots
  capable of learning from a changing environment.

- In unsupervised learning Processing Elements
  can cooperate or work in
  a competitive manner
  (Kohonen network).

2.1 Design of Networks 5

Partially unsupervised learning is often used for

- data association
- data classification
- data conceptualization

and can be best compared to the statistical Principal Component Analysis (PCA).

2.1 Design of Networks 6

Too many hidden layers in a MLP lead to an overfitting,

i.e. to a memorizing effect,

which can be compared to

the well known learning technique of students,

who learn by memorizing

but do not understand the problem.

The Problem of Overfitting and the number of PEs
Kolgomorov's Rule

In case that the number of PEs exceeds or is equal to (2n +1)

with n =
number of input variables
xi (i = 1...n)

ANY FUNCTION $F(x_1, x_j, \ldots x_n)$ CAN BE DESCRIBED.

To avoid an „overfitting"
it is recommendable to start with (2n +1) PEs
and to reduce subsequently the number of PEs
to be on the save side.

The Problem of Overfitting and the number of PEs

General Rule

If the number of input data (samples)

is equal to the number of PEs

the target values

will be overfitted

and the „generalization“ power will be inadequate.

2.2 An example: Tablet compression study using two Artificial Neuron Networks and the RSM-technique

2.2.1 The Generalized Feed Forward Multilayer Perceptron (GFF-MLP):

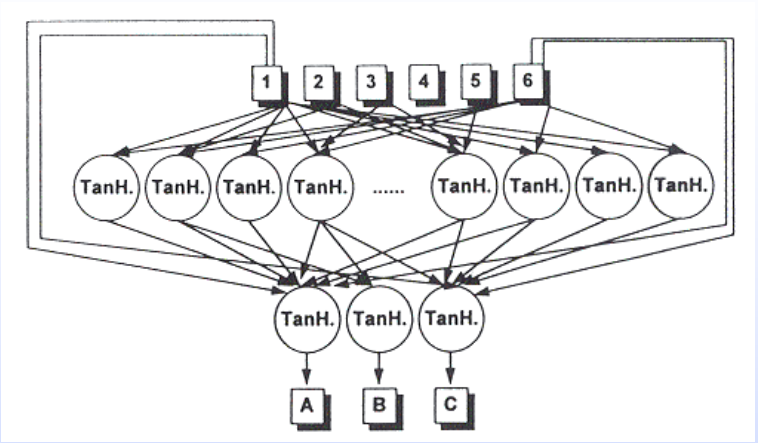


Fig. 13 GFF-MLP
simplified

The input layer consists of 6 PEs, which correspond to 4 compression variables (matrix filling speed, precompression force, compression force, rotation speed)
the formulation and the batch . Not used for prediction

The output layer consists of three results, i.e.

A: the hardness of the tablet

B: % drug diss. after 15min. and

C: $t_{50\%}$ (time, when 50% of drug dissolved)

The transfer function tanh (z) is used. Between the input and the output layer there is a hidden layer with 11 Processing Elements (PEs) with direct connections between the in- and output layers.

2.2.2 Self organizing feature map (SOFM) - MLP

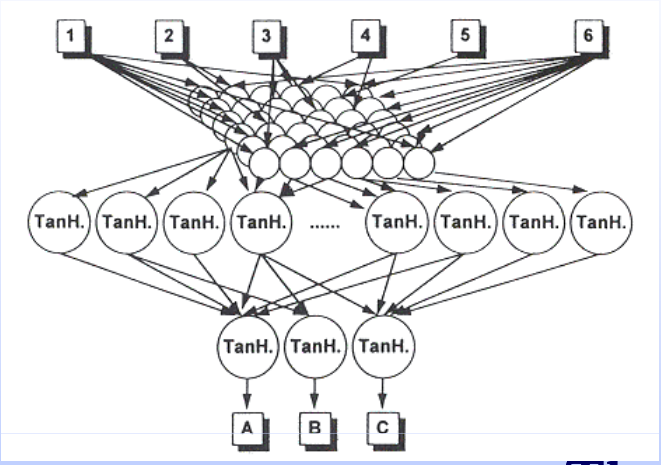


Fig. 14 The hybrid network consists of a SOFM (Kohonen network) combined with a normal MLP

The input layer is identical to the inputlayer of the GFF-MLP. The following Kohonen Layer (SOFM) consists of 6x6 Processing Elements (PE) and the hidden layer of 11 PEs of the subsequent „MLP“.

The output layer is identical to the GFF-MLP network.

In both networks the input variables consist of totally 5 factors with 3 levels and 1 factor (formulation) with 2 levels. The input data are processed in the 6x6 PEs Kohonen layer working as a PCA system and reducing the input variables to the prinicipal components.

The resulting output of the SOFM is then used as a new input to the MLP. The factor „Batch“ was used as a further variable.

2.2.3 Results$_1$ of the 2 networks and of the RSM - technique

A) Replication of an arbitary function

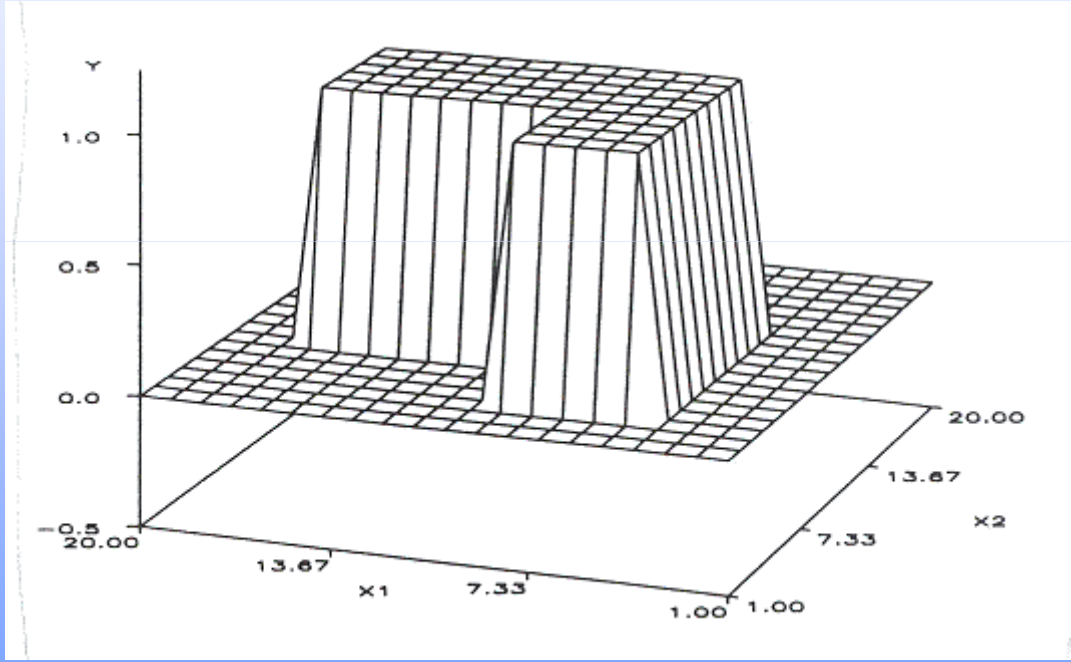


Fig.15: Arbitary function

2.2.3 Results$_2$ of the 2 networks and of the RSM - technique

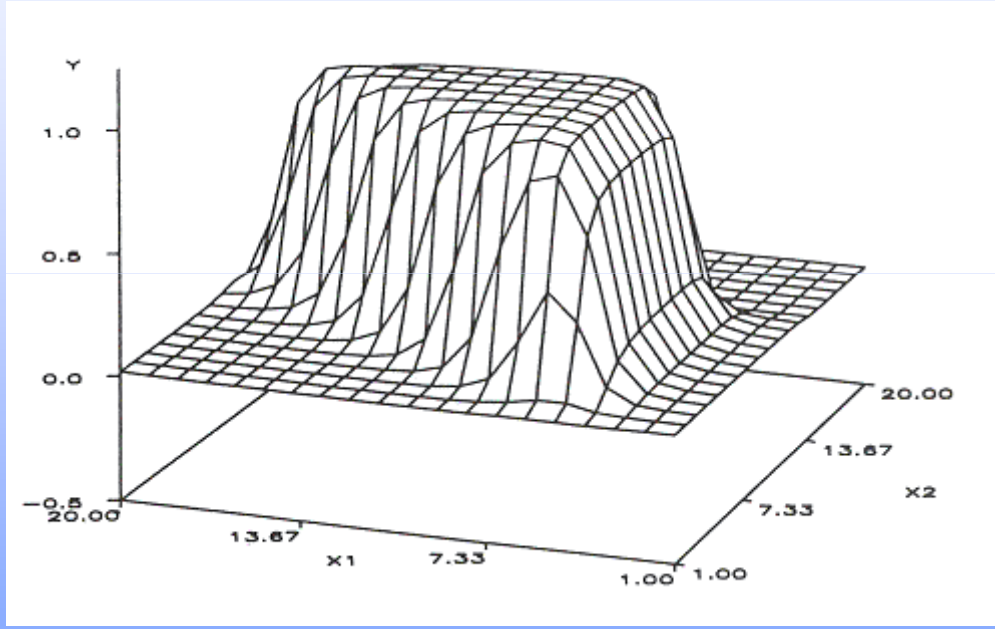


Fig.16: ANN

2.2.3 Results$_3$ of the 2 networks and of the RSM - technique

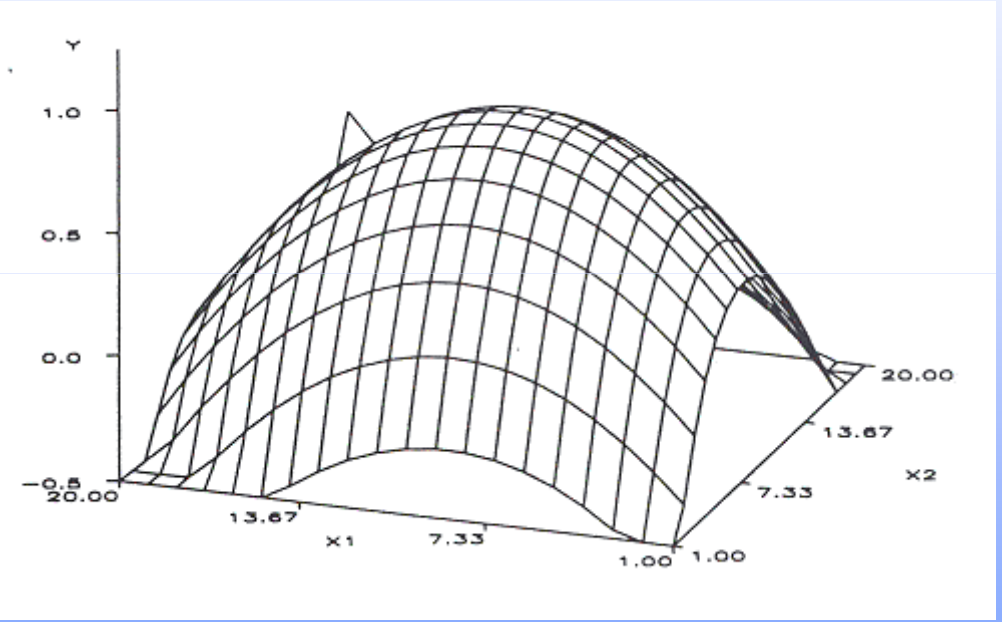


Fig.17: RSM

2.2.3 Results$_4$ of the 2 networks and of the RSM - technique

B) Hardness (Crushing strength) values and dissolution rate data

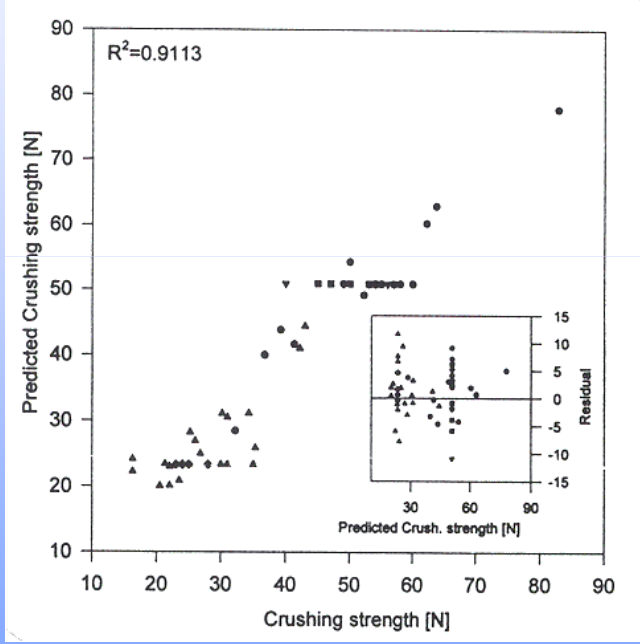


Fig.18: ANN (GFF-MLP)

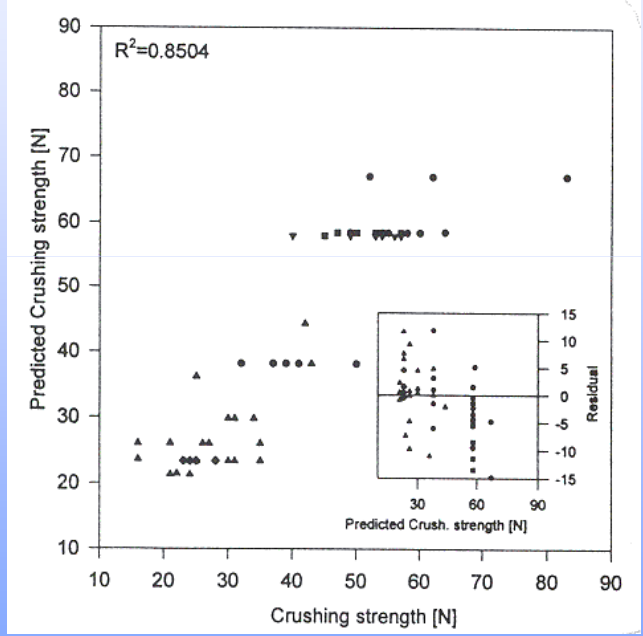


Fig.19 ANN (SOFM-MLP)

2.2.3 Results$_5$ of the 2 networks and of the RSM - technique

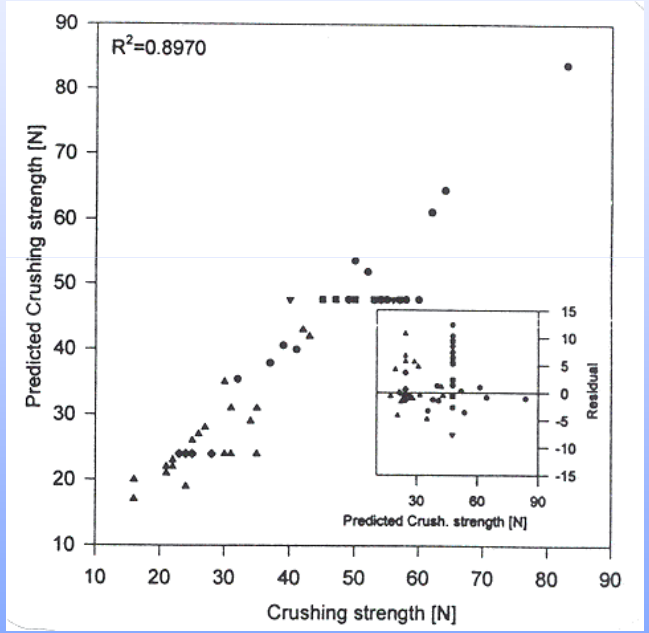


Fig.20: RSM - technique

Percentage of Drug Dissolved After 15 min (%)
R-Square Results for the Tablet Compression Study

	GFF-MLP	SOFM-MLP	RSM
R^2 without factor "Batch"	0.2589	0.1040	0.1366
R^2 with factor "Batch"	0.8809	0.8775	0.8679

Time to 50% Drug Dissolution (min)
R-Square Results for the Tablet Compression Study

	GFF-MLP	SOFM-MLP	RSM
R^2 without factor "Batch"	0.3411	0.2942	0.2739
R^2 with factor "Batch"	0.8709	0.8536	0.8449

Fig.21 R^2 - Results „Dissolution Rate"

2.2.3 Results$_6$ of the 2 networks and of the RSM - technique

Dissolution rate data

The factor „batch“ can not be used for prediction purposes!

BUT...

- The factor „batch“ can be used in the case of unknown , hidden factors, which have to be analysed in a subsequent, separate study!
- No wonders can be expected from ANN or other evaluation/modeling techniques if the experimental data are not sufficient to explain the behavior of a system!

3. Conclusions and Experience with ANN 1

- ANN can be very helpful in the design and in the optimization of pharmaceutical (and other) dosage forms.

- ANN is a valuable alternative to experimental statistical design (RSM) and evaluation.

3. Conclusions and Experience with ANN 2

- The advantage of ANN consists in the fact that all experimental data can be used and that the evaluation is less sensitive to missing data than in the case of RSM.
- Outlayers or erroneous data can disturb ANN and/or other evaluations.
- If appropriate RSM and ANN approches should be used for a comparative evaluation.

3. Conclusions and Experience with ANN 3

- Major pharmaceutical and chemical companies use ANN - methodology to optimize the performance of formulations and of processes leading to new patents and to the reduction of the development time.
- It is reported by a German Company* that thanks to ANN the development time for a special product could be reduced by three years!

* ANN is used in Germany by BASF, Bayer, Henkel, Merck and other companies.

3. Conclusions and Experience with ANN 4

- In a modern lab all experimental data should be recorded in an appropriate way for a subsequent analysis and evaluation by ANN!
- It may be useful to analyse existing data of complex systems such as formulations to get a new insight and understanding of the behavior of such systems.
- „Older“ Formulations consist in general of many components and it is often doubtful which of the partly expensive components is really necessary!

Thus an ANN - analysis may lead to new conclusions and to new patents for a line extension of a product!

4. Acknowledgements

The following persons are acknowledged
for their contribution to the application of ANN
in Pharmaceutical process optimization

Jaques Bourquin, PhD
Novartis Pharma Ltd.
CH-4002 Basel

Heinz Schmidli, PhD
Novartis Services CP Ltd.
CH-4002 Basel

Peter van Hoogevest, PhD
ADD Technologies Ltd.
CH-4132 Muttenz

Michael Lanz
Pharmacenter
University of Basel
CH-4056 Basel

Christina Erb
Pharmacenter
University of Basel
CH-4056 Basel

5. References

- A.S. Hussein et al. Pharm. Res. 8: 1248-1252 (1991)
- E. Murtomemi et al. Lab. Microcomput. *12*, 69-76 (1994)
- J. Bourquin et al. Pharm. Dev. And Tech. *2*, 95-109 (1997)
- J. Bourquin et al. Pharm. Dev. And Tech. *2*, 111-121(1997)
- J. Bourquin et al. Eur. J. Pharm. Sci. *6*, 287-300 (1998)
- J. Bourquin et al. Eur. J. Pharm. Sci. *7*, 17-28 (1998)

ARTIFICIAL NEURAL NETWORKS

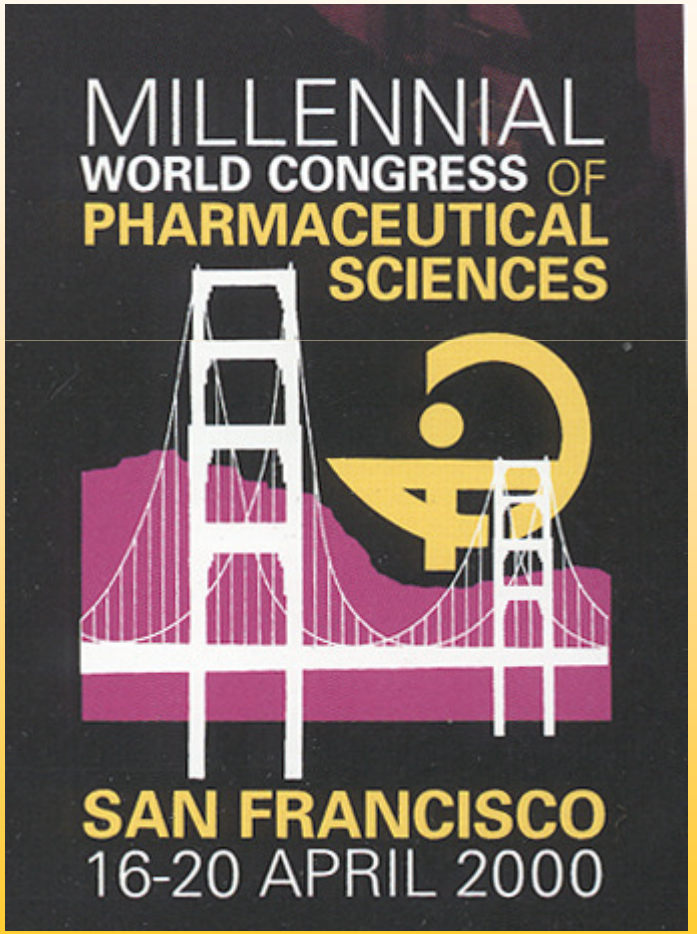
was presented by

Prof. Dr. Hans Leuenberger

Pharmaceutical Technology

of the University of Basel

Switzerland



It was a great pleasure to have your attention.

Thank you.