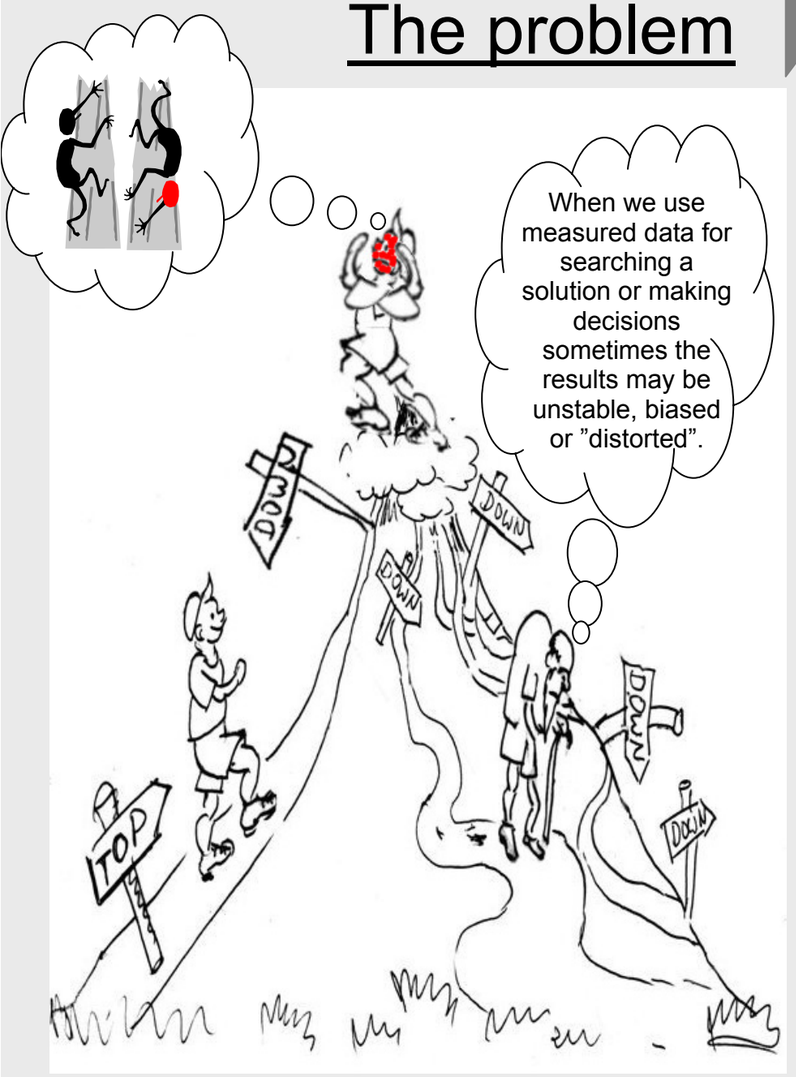
Tatjana Lange

Automatic & Feedback Control
University of Applied Sciences Merseburg, Germany

COMPSTAT'2010

Solution Tuning - an attempt to bridge existing methods and to open new ways

The problem



When we look at applied engineering tasks we often meet the situation where **measured data** must be used to find a certain result.
These tasks can be described in a very generic way as follows:

$$y = A\nu$$

where:
- A – is a linear operator, e.g. a matrix
- y – is the measured vector
- ν – is the unknown vector that is searched

A - may be a Fredholm operator. In this case $y(\tau) = \int_a^b k(\tau, t) \cdot \nu(t) dt = \mathbf{A}\mathbf{v}$

$\nu(t)$ is searched within the space F and
$y(\tau) \in \nu$ – is the space where the measured data are derived from.

Here, we have to solve an **inverse task** and the usual solution approach $y = A^{-1}n$ can lead to unstable (w.r.t. Hadamard) results.

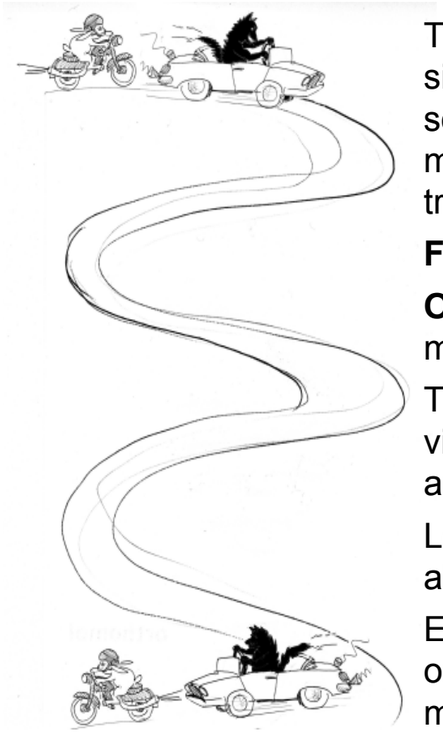
We face this problem in many cases, e.g. when we have to deal with

- Empirical Risk Minimization,
- Modeling and Forecasting,
- Pattern Recognition,
- Reconstruction of dependencies by measured data,
- the solution of the Fredholm Integral Equation of the first kind (e.g. convolution, Wiener Filter)
- the handling of Matrices which are sensitive to inversion.

Therefore these tasks can be solved by means of **optimization** instead of using the rules of inversion.

The elaboration (80 % of success depends on it) and the solution of optimization tasks is connected with many characteristics that are common for a lot of different applications.

This poster-presentation describes and partly generalizes these common characteristics with the help of examples and using **geometric representations**.



The picture with the hare and the wolf just simplifies the quantitative tasks which have to be solved in a very fast way by animals and humans more than 1000 times a day, e.g. when the trajectory of the race is not a road and is unknown.

Feedback control tries to imitate the pursuer.

Optimal control tries to imitate the victim who must be more clever to survive.

The decision spaces for both the pursuer and the victim are different, but the solution approaches are similar.

Let us just simplify the problem a bit and forget about the two decision spaces.

Even more, let us take from the continuous time only <u>one moment</u> ***i*** and consider the decision making only for this one moment ***i***.

For that case the task of **minimizing** (if you are the pursuer) the distance within the decision space Q ($a_1,a_2,....,a_m$) can be formulated as fo

The best decision is the optimum value

$$Q^*(a^*_1,a^*_2,....,a^*_h) = \min Q\ (a_1,a_2,....,a_h)$$

with

$a_1 ... a_h$ – as decision coordinates, e.g.

$a^*_1 = \gamma^*_{1i}$ – the angle by which the driver should turn steering wheel in moment ***i***

a^*_2 – the optimum acceleration in moment ***i***

Why can little animals so easy solve such a problem where we spend so much effort with mathematical abstractions ? Maybe, we can learn from the nature ? Maybe, the magic is not only in mathematics but also in **Optimization** and **Feedback principles** ?

Example:

$$\min Q\ (a_1,a_2,....,a_h) = \min \rho(object_1, object_2) = \min \rho(y, F(x,a))$$

The difficulties are cause by the fact that

F(x,a) – is **unknown**, that means

- the form of the function is unknown
- the number h of the parameters $a_1 ... a_h$ is unknown
- the values of the parameters $a_1 ... a_h$ are unknown

x – is measured (best case) without but more often with disturbances

y – is measured with disturbances

The task is to find this function F(x,a*) – it's structure and it's parameter values - in a way that the solution will be **stable**.

1. The task of finding the object F(x,a*) is an **<u>inverse task</u>** and all inverse tasks can usually be solved with the help of optimization, what lead us to the minimization of the distance between two objects.
2. The metric, that means the interpretation of the distance between the objects, can be chosen depending on the sense of the task. But the used metric has influence, sometimes even significant influence, on the skyline of the "mountains" built by the optimization functional within the decision space.

   <u>Example:</u>

   $$Q(a_1,...,a_h) = \sum_i [y_i - F(x_i,a)]^2 \qquad Q(a_1,...,a_h) \text{ is a paraboloid}$$

   $$Q(a_1,...,a_h) = \sum_i |y_i - F(x_i,a)|^k;\ 0 < k < 2;\ \ Q(a_1,...,a_h) \text{ is } \underline{\text{not}} \text{ a paraboloid}$$

3. The absolute minimum of the functional Q sometimes does not deliver a satisfying solution (F(x,a*)) and can only be used as orientation.

   The reason for that phenomenon lies in the following contradiction:

   The deeper the minimum of the functional the more exact is the solution but only with respect to the given data sample. But at the same time the model can be very complex (e.g. polynomial of high degree) and fits only to this one (given) data sample. That means the solution (e.g. the found model) does not properly work for a new data sample derived from the same object. The solution (the model) is unstable and does not give any useful prediction.

   What is the way out ? The way out is **polyoptimization** that means the introduction of penalty functionals that "punish" for complexity. In this way we get **new "mountains" within the decision space**.

4. The change of the geometry of the "mountains" within the decision space or the change of the dimensioning of the decision space (compare, for example, with α-procedure) following the goal of making the solution stable is called here **Solution Tuning.**

   For these changes different mathematical tools can be used.

Task of Mean Square Risk (Q_m) and Task of Empirical Risk (Q_e) in case of **Pattern Recognition and Neuronal Networks**

In case of Empirical Risk we do not use

$$Q_m^* = \iint \left(\omega - F(x,a)\right)^2 p(x,\omega)\,dx\,d\omega \rightarrow \min$$

but

$$Q_e^* = \frac{1}{l}\sum_{i=1}^{l}\left(\omega_i - F(x_i,a)\right)^2 \rightarrow \min$$

where ω_i and x_i – are empirical data and
l - is the length of the data sample of the pairs ω_i and x_i

Here we search F*(x,a) as optimum decision rule for <u>classification</u>.

Speaking about <u>pattern recognition</u> the physical sense is as follows:

The experimental observations are given as vector **x** but it's classification with the help of the number ω. We can have up to p-1 classes.

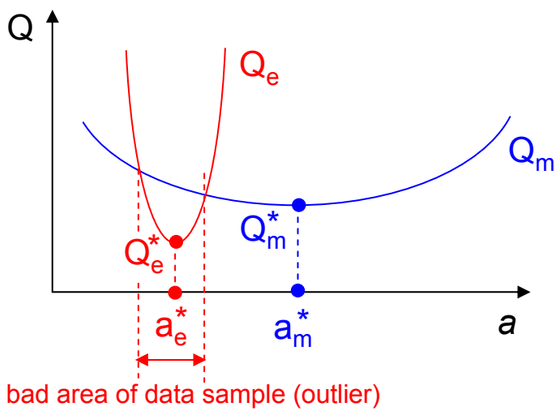
The task is to construct the decision rule ω=F*(**x**,a) with the help of the available sequence of ***l*** observations and classification $\mathbf{x}_1,\omega_1;\ldots;\mathbf{x}_l,\omega_l$ in such a way that this rule will classify new observations with a minimum of errors.

To make the things simple – let us just consider <u>two</u> classes V_1 and V_2.

Now, let us consider 2 typical problems:

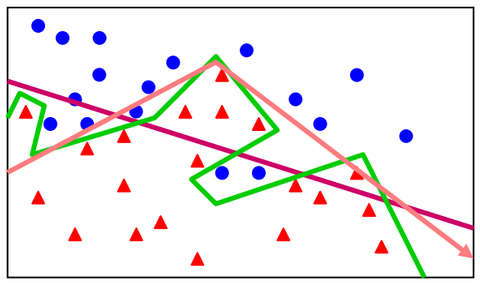
<u>Problem 1</u>

The empirical functional deviates significantly from the mean square functional. This is typical for outliers within the measured data..



<u>Problem 2</u>

How can we find the best compromise between fit and complexity ?



● - objects $v_1 \in V_1$
▲ - objects $v_2 \in V_2$
} V_1, V_2 – two classes

(Source: O. Bousquet, S. Boucheron, G. Lgosi: Introduction to Statistical Learning Theory.
www.kyb.mpg.de/publications/pdfs/pdf2819.pdf

The **<u>problem 1</u>** was solved by **Vapnik** with the help of a generalization of the Glivenko-Cantelli Theorem of Uniform Convergence.
Vapnik used the C-metric instead of the L_p^2-metric:

$$\rho(f_1,f_2) = \sup\left|f(x,a_1) - f(x,a_2)\right|$$

The use of the C-metric limits the effect of the outlier sample up to a certain width of the corridor.

Vapnik's Theorem:

When we select from a set of ***N*** decision rules that one rule that gives us a error frequency on the training sequence which is equal ν, then we can claim with a probability of **1-**η, that the probability of making an incorrect classification with the help of the selected rule equals a value which is less than ν **+** ε if the length of the training sequence is not less than

$$l = \frac{\ln N - \ln \eta}{2\varepsilon^2}$$

For the **<u>problem 2</u>** there exists a solution approach since the beginning of the eighties – especially for "short data samples". It is the so-called "**reduction theory**" or the "α**-procedure**".

With respect to Vapnik one can say: The less the number of decision rules ***N*** the less can be the length of the training sample.

Further on we define ***k*** as number of hypersurfaces within the decision space.

If ***k*** = 1 then we have a linear discrimination.

The number of the decision rules can be defined as:

$$\ln N \approx (k \cdot m)^2$$

where ***m*** – is the primary dimension of the decision space (number of primary characteristics of object classes).

Out of these number of primary (or original) characteristics ***m*** a subset $\boldsymbol{n_0}$ of the most powerful characteristics should be selected.

The reduction theory provides a procedure how to synthesize a (reduced) space of dimension $\boldsymbol{n_0}$ in which a correct <u>linear</u> discrimination of the object classes - given by the training sample of length ***l*** – will be possible.

$$n_0 = \frac{\varepsilon \cdot l + \ln \eta}{\ln m}$$

Any exceeding of n_0 leads to the loss of the guarantee that the given ε and η can be reached.
To avoid a dimensioning of the decision space that exceeds n_0 we define the "power of discrimination"

$$F(x_i) = \frac{\omega_i - \omega_{i-1}}{l}$$

where ω_{i-1} and ω_i are the number of objects of the training sample which have been correctly classified before and after the characteristic x_i was counted.
For the synthesized decision space we will only use only the characteristics with the greatest values of "power of discrimination" but in any case it should be greater than the minimum value

$$F_{\min}(x_i) = \frac{l}{n_0}$$

Task of Mean Square Risk (Q_m) and Task of Empirical Risk (Q_e)
in case of **Empirical Regression for Modeling and Forecasting, Neuronal Networks and Group Method of Data Handling (GMDH)**

Here again we substitute the Mean Square Risk (Q_m)

$$Q_m^* = \iint \left(y - F(x,a)\right)^2 p(x,y)\,dx\,dy \to \min$$

by the **Empirical Risk (Q_e)**

$$Q_e^* = \frac{1}{l}\sum_{i=1}^{l}\left(y_i - F(x_i,a)\right)^2 \to \min$$

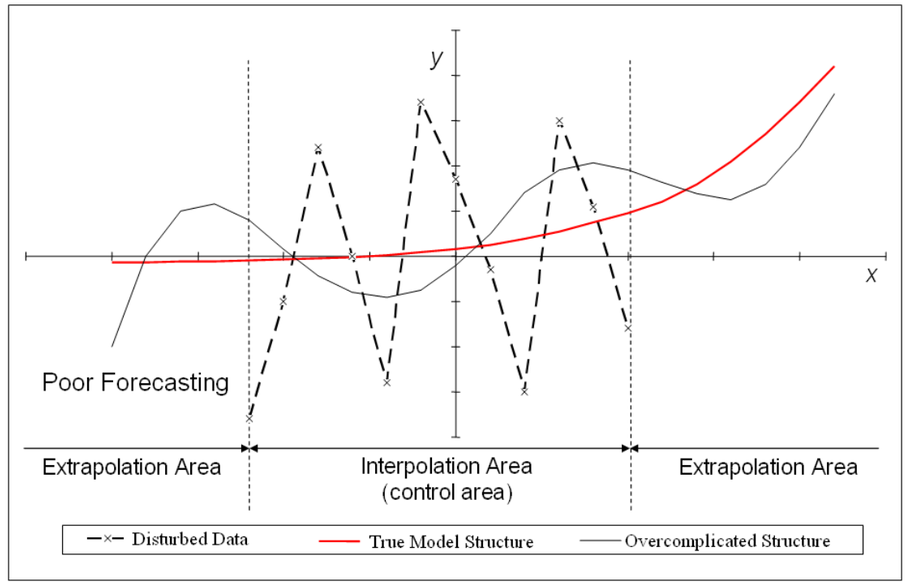
but the **content of $F(x_i,a)$ is different**.

$F(x_i,a)$ is an **unknown polynomial** which approximates linear or nonlinear regressions or unknown dependencies *y* on *x*.

x_1,y_1; .... ; x_i,yi_i;...; x_l,y_l – are the measured data which "generate" that dependency.

Here again we look at the compromise between fit and complexity.

Comparison of a True and an Overcomplicated Model Structure



This problem of finding the best compromise between fit and complexity was differently solved by at least 20 authors (e.g. Vapnik, Akaike, Schwarz, Lange – see appendix 4).Vapnik (Structure Minimization) and Lange (Model Tuning) follow Tichonoff's ideas.

Let us shortly describe the idea of "**Model Tuning**" for that case:

First, there were introduced 3 new terms for the **proximity of the models** $F_I(x,a_I)$ and $F_{II}(x,a_{II})$ which are based on the general definition of "Solution Stability" given by Hadamard.

(1) With regard to the output *y*, two models $\mathbf{M_I}$ and $\mathbf{M_{II}}$ are near to each other (that means $\rho_{R^l}(M_I, M_{II}) \le \varepsilon$, where ε is small). This corresponds to the convergence according to the output, when $\rho_l(y_I, y_{II}) \le \gamma$ and γ are small,
where y_I and y_{II} are output vectors of the models $\mathbf{M_I}$ and $\mathbf{M_{II}}$.

If ρ_l represents an *l*-dimensional Euclid Metric, then $\rho_l(y_I, y_{II}) \le \gamma$ can be replaced by the equivalent postulation

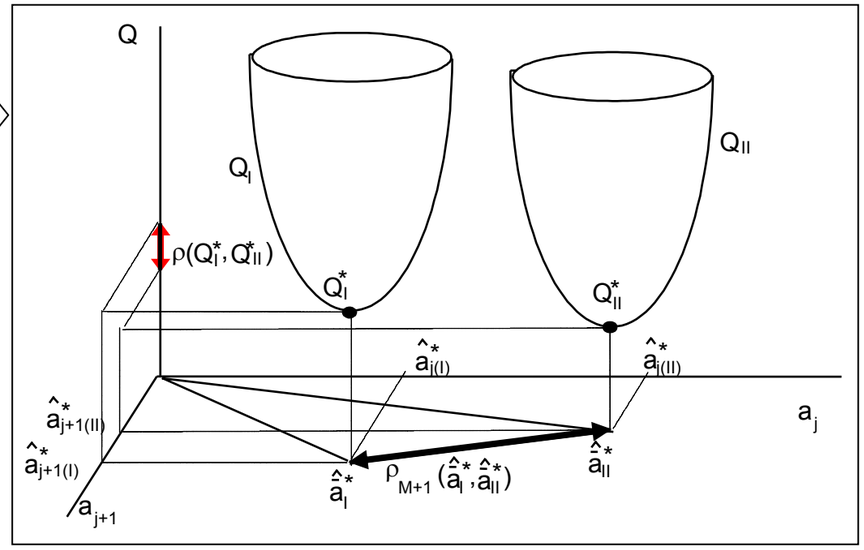
$$\rho_l(Q_I^*, Q_{II}^*) \le \gamma$$

where

Q_I and Q_{II} are functionals (criteria) for estimating parameters by means of Least Square Methods

Q_I^* and Q_{II}^* are the values of *RSS* of the first and second model.

Case 1:
Proximity of Models **only** according to the output *y*

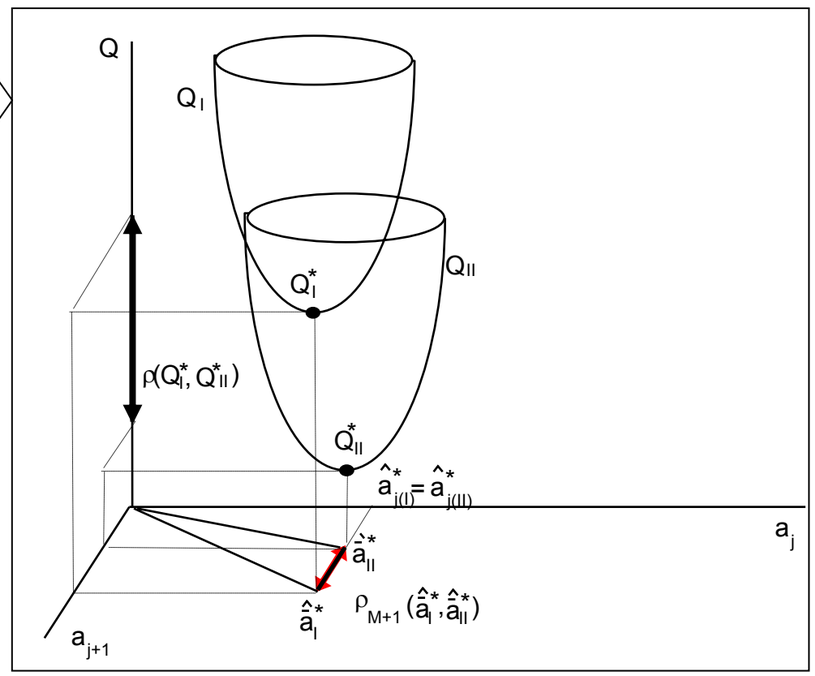


Distance between models estimated on different data sets

(2) Two models $\mathbf{M_I}$ and $\mathbf{M_{II}}$ are near to each other with regard to the <u>coefficient vector</u> $\vec{a}$ (that means $\rho_{R^l}(M_I, M_{II}) \le \varepsilon$, where ε is small). This corresponds to the convergence according to the parameters, when $\rho_{M+1}\left(\hat{\vec{a}}^*_{(I)}, \hat{\vec{a}}^*_{(II)}\right) \le \xi$ and ξ are small,

where $\hat{\vec{a}}^*_I$ and $\hat{\vec{a}}^*_{II}$ are the vectors of the estimated parameters of the models $\mathbf{M_I}$ and $\mathbf{M_{II}}$. correspond to the estimated values of the parameter functional.

Case 2:
Proximity of Models **only** according to the coefficient vector

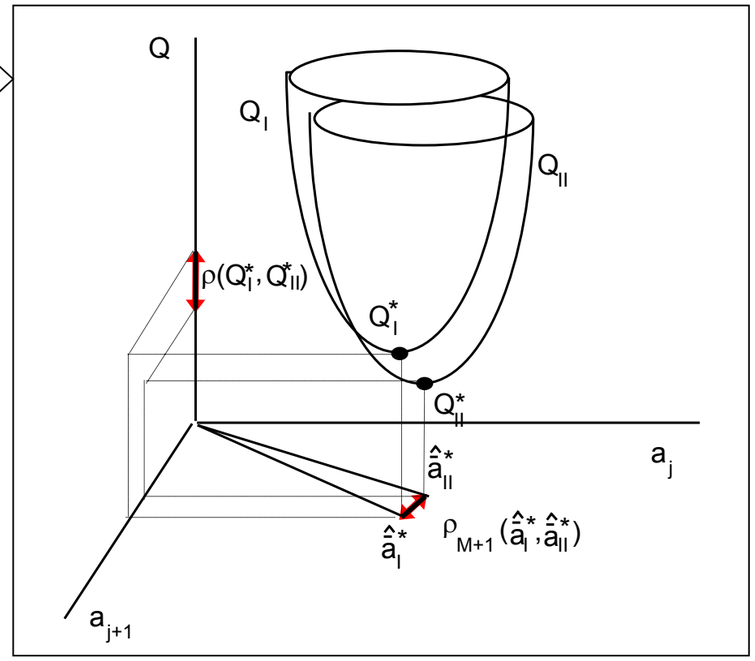


Distance between Models estimated on different data sets

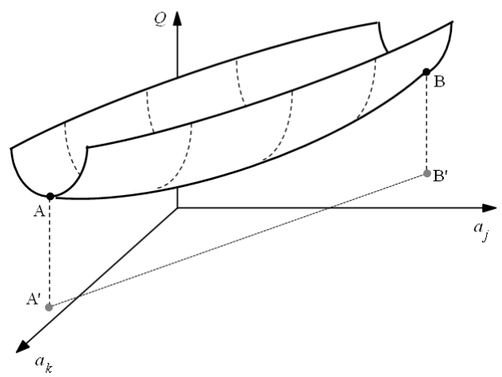
(3) Two models are really near to each other if the proximity of outputs *y* and parameters $\vec{a}$ is guaranteed. The distance between two models is defined by the sum of distances between outputs and parameters:

$$\rho_l(y_I, y_{II}) < \gamma \text{ and } \rho_{M+1}(\hat{\vec{a}}^*_I, \hat{\vec{a}}^*_{II}) < \xi \;\Leftrightarrow\; \rho_{R^l}(M_I, M_{II}) < \varepsilon$$

Case 3:
Complete Proximity of Models



Distance between models estimated on different data sets



We consider here the **case 1** where we do not have complete complexity.

Thus the Gauss's method and consequently the Residuum Sum if Squares (RSS) is not always applicable because they do not exclude unstable models.

The problem is the harder the more the functional has a "trough"-like form.

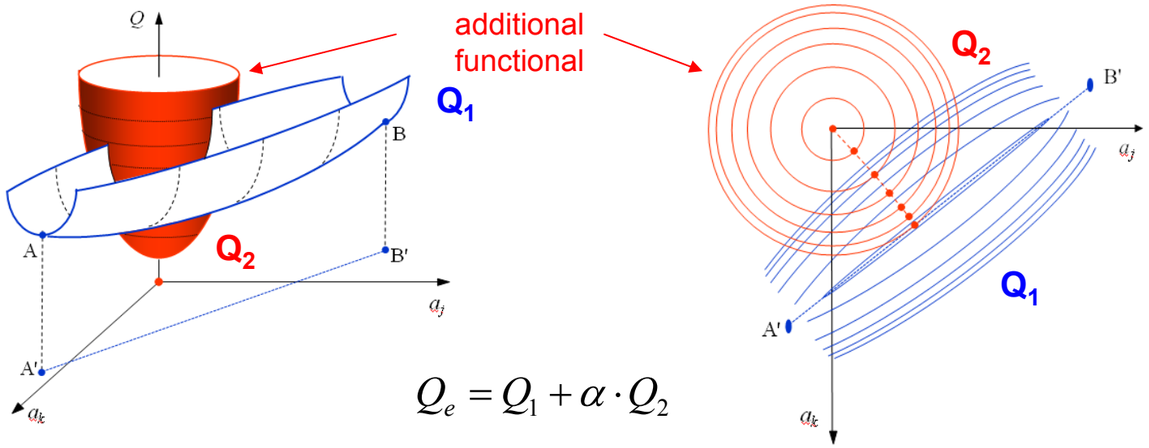
The bottom of that "trough" builds the "zone of insensitivity".

For this special case (the "trough"-like functional) the "Model Tuning" offers <u>two possibilities</u> for the simultaneous calculation of both the structure and the parameter values of F(x,a):

The **first idea** is to add stepwise new terms to the polynomial in the order of their "power of discrimination" using the rules of the **α-procedure** and the so-called "compromising clearance" ξ as corridor of permissible tolerance.

We use here in a formal-mathematical way the α-procedure originally developed for pattern recognition – formally considering each term of the polynomial like a "characteristic" in case of pattern recognition.
Thus the well-proven algorithms of the α-procedure can be used for the finding of the functional that makes a good compromise between fit and complexity.

The **second idea** consists in the introduction of the new **Local Data Uncertainty Criterion (LDUC)** as an evolution of Tichonov's idea of poly-optimization, i.e. the use of an additional "penalty"-functional for finding a compromising solution.



$$Q_e = Q_1 + \alpha \cdot Q_2$$

The figures above illustrate the Ridge-estimation:

$$Q_e(x,a,\alpha) = \frac{1}{l}\sum_{i=1}^{l}\left(y_i - F(x_i a)\right)^2 + \alpha\|a\|^2$$

with α – as Tichonov's regularization Parameter.

The value α_i corresponds to the Lagrange Multiplier and the value c_j defines the restriction of the contour lines of Q_2.

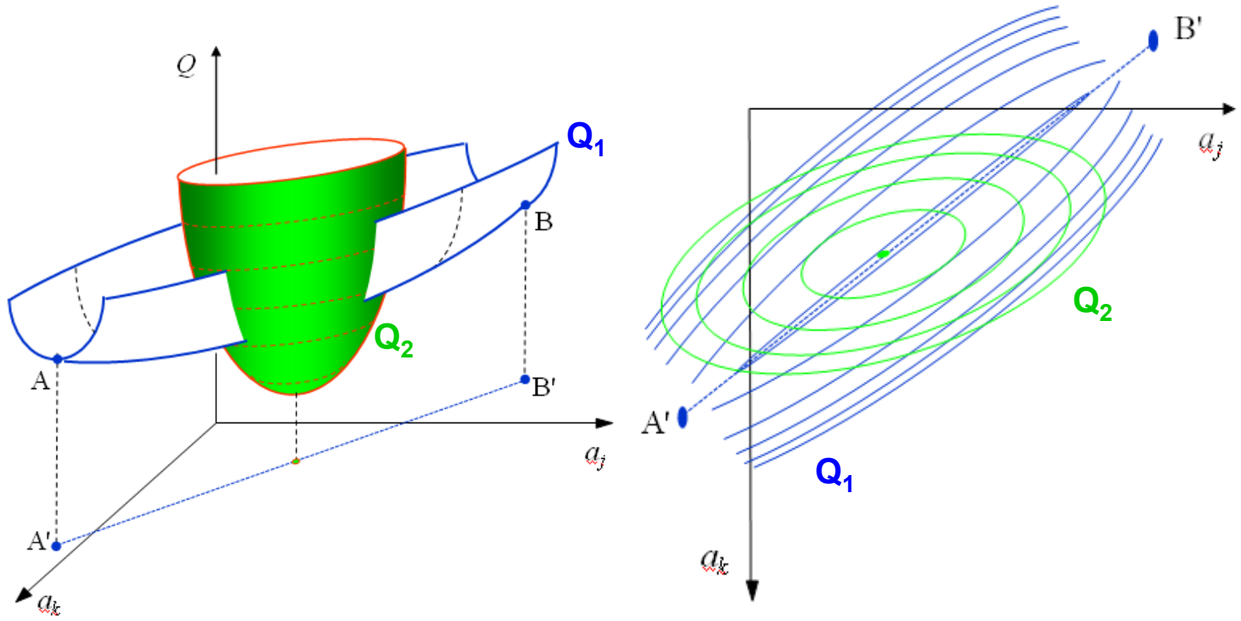
$$\sum a_i^2 \le c_i$$

$0 = \alpha_0 < \alpha_1 < ..... < \alpha_n$ corresponds to $c_\infty > .... > c_1 > c_o = 0$

The next figures showing us a quadric metric should be understood just as a partial help for the explanation of the LDUC.

The figures show a continuous space representing the complete decision space with **all** parameters of the Kolmogorov-Gabor polynomials.

The functional Q_1 reflects – with the help of the square sum of the axis's of the ellipsoid – the sensitivity of the coefficients and with it the structure that is caused by the data sample *y* but transformed into the space F, a ∈ F.
The functional Q_2 reflects the sensitivity of the coefficients that is caused by the given structure of the polynomial.



In a common and analytical form the **Local Data Uncertainty Criterion** (LDUC can be represented as follows:

$$LDUC = Q = Q_1 + Q_2 = tr\left[\rho_Y\left(\hat{\vec{y}}, \vec{y}\right)\left(x^T x\right)^{-1}\right] + \rho_F\left(\vec{a}_I, \vec{a}_{II}\right)$$

Q_1 - is the main criterion, $Q_1 = \sum_i^h \hat{\sigma}_{a_i}^2$ Q_2 - is the penalty criterion

$\left(x^T x\right)^{-1}$ - is a correlation matrix

$\rho_Y\left(\hat{\vec{y}}, \vec{y}\right)$ - is a metric in space Y but it transforms RRS into the space of coefficient F

$\rho_F\left(\vec{a}_I, \vec{a}_{II}\right)$ - is a measure of the stability of modelling in the space F

The used metric may be different (see appendix 2), e.g.

$$LDUC_{simple} = Q = \underbrace{tr\left[\frac{\sum_{i=1}^{l}\left(\hat{y}_i - y_i\right)^2}{l-h}\cdot\left(X^T X + \alpha \mathbf{I}\right)^{-1}\right]}_{Q_1} + \underbrace{\sum_{j=0}^{h}\left(\hat{E}_{boot}\left(\hat{a}_j\right) - \hat{a}_j\right)^2}_{Q_2}$$

The estimation of the coefficients is performed NB times where B is the number of he Bootstrap tables.

$$\hat{E}_{boot}\left(\hat{a}_j\right) = \frac{\sum_{nb=1}^{NB}\hat{a}_{j\,nb}}{NB}$$

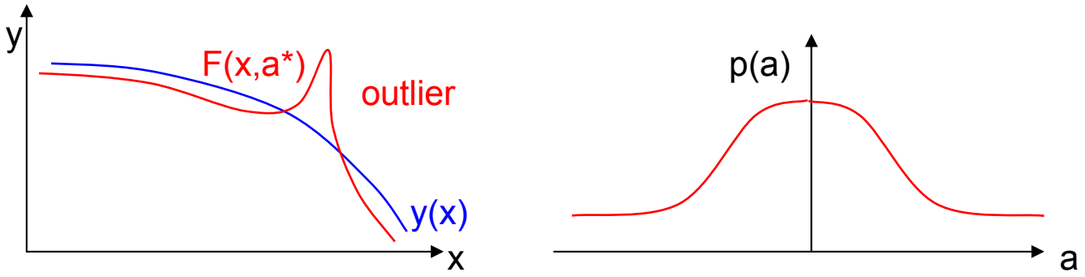
Robust Estimation and Robust Metrics for Structure Selection Criteria

40 years after the Glivenko-Cantelli theorem **Tukey** initiated with his two ideas a new direction of development of mathematical statistics.

Both ideas are connected to the terms “distance” and “metric”.

The first idea that can be considered as “foundation stone” of robust estimation will be described below. The second idea – the introduction and use of the term “data depth” instead of mathematical expectation – will be shortly discussed in the **outlook** of the given paper.

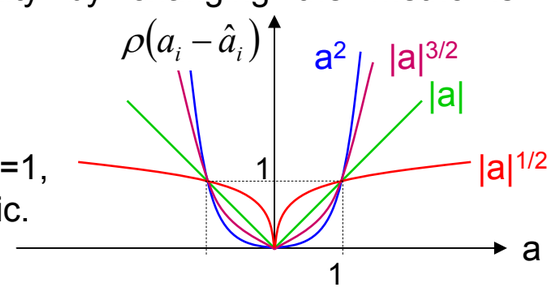
Now, let us shortly describe that “foundation stone” of robust estimation under the angle of modeling.



During the measurement of the quantity *a* for the estimation of $\hat{E}(a)$ any **outlier** can significantly change the result. It is unknown whether that outlier belongs to the parent population or whether it is caused by a short non-representative data sample (see figure above). We have to do with a **heavy-tailed distribution**.

The target is not to eliminate the outlier completely but to reduce its influence. For that purpose a penalty by changing the metric is introduced:

$$\rho(a_i - \hat{a}_i) = \sum_i |a_i - \hat{a}_i|^k \qquad 0 < k < 2$$



The “slightest” penalty is on the left to a=1, the “highest” penalty is the square metric.
k=1 means Huber estimation.

For “short” data samples we can use the **Gnostic Theory** which works with the “Gnostic” distance that is also a robust distance.

For the **LDUC** proposed by the author of this paper there also exists a robust version that uses a robust metric.

Unfortunately the use of robust metrics affects the computing complexity of the optimization task because it may lead to complex landscapes of Q.

Summery

About Distances and Metrics

The situation when instability appears during the handling of inverse tasks leading to “multiple” solutions (as shown with the help of the “trough”) requires additional information for “fixing” the solution.

Two big directions of the solution of the problem have been emerged – reflecting the application area but also the affinity of the scientists.

The **first** and more generic direction was born during the work with continuous objects, e.g. with the convolution equation. Together with it Tichonov developed his “Regularisation” Theory. In the contexts of compiling algorithms this theory can be shortly represented as a polyoptimization task:

$$Q^*(\nu) = [Q_1(\nu) + \alpha \cdot Q_2(\nu)] \to \min$$

with ν - as searched solution vector, i.e. the coordinates of the optimum.

... compare with the angle of the wolf’s steering wheel ;-)

The definition of the regularization parameter α may be very difficult. The Ridge Regression is just special and easy case.

Behind Q_1 and Q_2 there may be hidden the distances with different metrics:

$Q_1(\nu) = Q_y(\nu) = \rho_y^2(y, A\nu)$ - distance in the space of measured data y, y ∈Y

$Q_2(\nu) = Q_F(\nu) = \Omega(\nu) = \rho_F(\nu_1, \nu_2)$ - convex functional in the space F, ν ∈F

It can also be that $\rho(\nu_1, \nu_2) = \sup|z_1(s) - z_2(s)|$ (*refer to Glivenko / Vapnik*)

where *s* – element of the definition interval of the linear operator A.

For example, $\Omega(\nu) = \|a\|^2$ gives us a Ridge-estimation.

A different but similar functional Ω for estimations was proposed by LAN 83.

In general, the biggest problem in modeling the regression of polynomials is caused by the fact that the estimation of the parameters and of the structure are different – the estimation of the parameters is done in a continuous space but the estimation of the structure - in a discrete space.

The **second** direction is connected to such applications like pattern recognitions and structure minimization. It has many faces and proposes a lot of approaches. We can joke and say that the direction of investigations itself is nothing else than a great variety of solutions. The first proposals were done by Fisher, Neyman and Pearson and close to the task of polyoptimazation. But they were not really usable for automatic algorithms for the computing of F(x,a)

But with the following proposals that disadvantage was eliminated and it became possible to find a compromise between the complexity of the model (or the complexity of decision rules) and the fit for a given lengths of the data sample with the help of automatic computer algorithms.

(1) Mallows, Vapnik and others proposed a “Scaling Criterion” for the step of structure selection:

$$Q = \frac{Q_1}{Q_2}$$

(2) Akaike, Broersen, Schwarz, Hampel and others introduced a penalty criterion:

$$Q = Q_1 + Q_2$$

where Q_1 - is the main criterion and $Q_2(l,h)$ - an additional criterion.

Ivachnenko, Tamura, Kondo, Sawaragi, Lange and others also proposed a modification for the sub-direction (2) – “penalty by validation”.

The reason for the big number of proposals which work more or less fair can be explained by the phenomena that it is difficult to implement the best and most elegant theoretical ideas as computer algorithms.

Outlook

Let us remember the second famous idea of Tukey (1974) – the introduction of the term **data depth** – which has disburdened us from the binding to a concrete distribution function. (From where the poor animals – our hair and wolf - should have information about the distribution ?)

After **Tukey** (see overview Zuo 2000) different methods for computing the data depths were proposed. The term “convex hull” was introduced. With it the goal to use these methods for many applied task, e.g. for pattern recognition and modeling, moved closer.

But all corresponding tasks were restricted to a dimension of m=2.

But the problem of extending the dimensioning to m > 2 was solved by Mosler/Lange/Bazovkin [MoLaBa 2009] and new opportunities for pattern recognition have been opened.

The overcoming of the dimensioning restrictions also helps to improve the use of the α-procedure (original as method for pattern recognition) for modeling better solving the “clearance problem”.

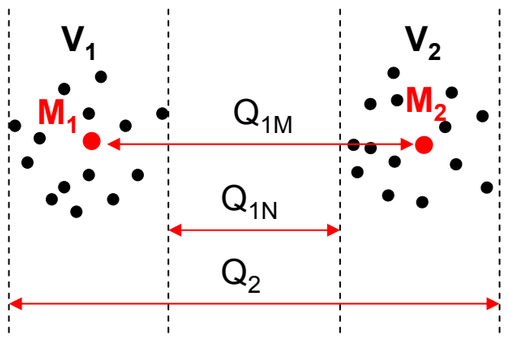
It opens opportunities for the direct calculation of the scaling distance from the measured data and it’s use for the discrimination of object classes. With it the following scaling distances are especially important:

- Machalanobis distance which is the lower estimation of the quality of the decision rule when the length of the data sample is fixed

$$Q_M = \frac{Q_{1M}}{Q_2}$$

- Novikov distance which is the upper border of the quality of the decision rule

$$Q_N = \frac{Q_{1N}}{Q_2}$$



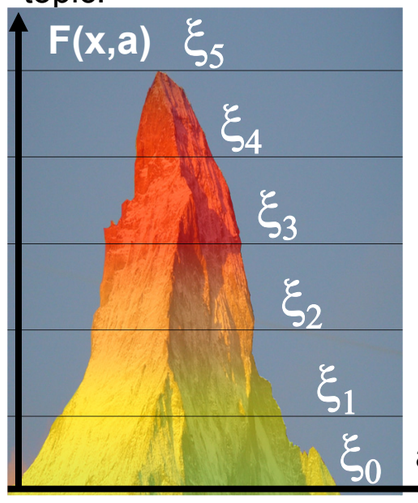
V_1,V_2 – object classes
M_1,V_2 – data depth
Q_{1N},Q_2 – distances between the internal and external convex hulls

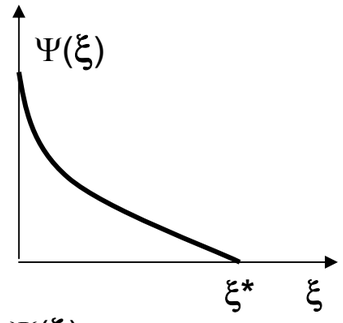
If we can finally solve the problem of the data function depth we could solve the modeling task in a **direct** way: First select the structure and then estimate the coefficient values.

Maybe the **psi-transformation** can help to elaborate an effective method for the estimation of the data function depth.

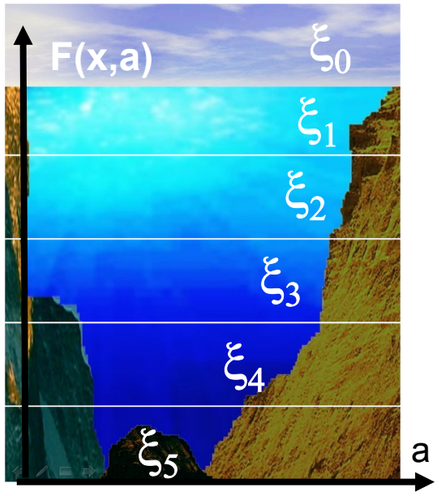
The psi-transformation was proposed by **Chichinadze** for the optimization in case of discontinuous multimodal “mountains”. It is based on the term **Lebesgue Measure.**

But today there is no room for further discussions with respect to that topic.





Ψ(ξ) - Lebesgue Measure for each level



This presentation and the attachments will be available on
http://www.iks.hs-merseburg.de/~tlange/
Contact: tatjana.lange@hs-merseburg.de