DRAFT COPY – PLEASE DO NOT DISTRIBUTE

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY
and
CENTER FOR BIOLOGICAL AND COMPUTATIONAL LEARNING
DEPARTMENT OF BRAIN AND COGNITIVE SCIENCES

A.I. Memo No. 1598 December 1996
C.B.C.L Paper No. 141

Prior Information and Generalized Questions

Jörg C. Lemm

This publication can be retrieved by anonymous ftp to publications.ai.mit.edu.
The pathname for this publication is: ai-publications/1500-1999/AIM-1598.ps.Z

Abstract

In learning problems available information is usually divided into two categories: examples of function values (or training data) and prior information (e.g. a smoothness constraint).

This paper 1.) studies aspects on which these two categories usually differ, like their relevance for generalization and their role in the loss function, 2.) presents a unifying formalism, where both types of information are identified with answers to generalized questions, 3.) shows what kind of generalized information is necessary to enable learning, 4.) aims to put usual training data and prior information on a more equal footing by discussing possibilities and variants of measurement and control for generalized questions, including the examples of smoothness and symmetries, 5.) reviews shortly the measurement of linguistic concepts based on fuzzy priors, and principles to combine preprocessors, 6.) uses a Bayesion decision theoretic framework, contrasting parallel and inverse decision problems, 7.) proposes, for problems with nonapproximation aspects, a Bayesian two step approximation consisting of posterior maximization and a subsequent risk minimization, 8.) analyses empirical risk minimization under the aspect of nonlocal information 9.) compares the Bayesian two step approximation with empirical risk minimization, including their interpretations of Occam's razor, 10.) formulates examples of stationarity conditions for the maximum posterior approximation with nonlocal and nonlinear priors, leading to inhomogeneous nonlinear equations, similar for example to equations in scattering theory in physics.

In summary, this paper focusses on the dependencies between answers to different questions, and because not training examples but such dependencies enable generalization, it emphasizes the need of their empirical measurement and control and of a more explicit treatment in theory.

This report describes research done within the Center for Biological and Computational Learning in the Department of Brain and Cognitive Sciences at the Massachusetts Institute of Technology. The author holds a postdoctoral fellowship (Le 1014/1–1) from the Deutsche Forschungsgemeinschaft.

1 Introduction

To clarify the aim of the paper, our use of the term "prior information" and to define vocabulary and notation we analyse the fundamental problem of generalization for a simple toy example:

Let us assume that we are interested in the answers $y_1(x_1)$ and $y_2(x_2)$ to two questions x_1 and x_2. We will call these questions *relevant* to our *application*. Let us assume we have found for y_1 the value $y_1 = 2$ in a first measurement. We are interested in the future output when measuring x_1 and x_2. Clearly this is not possible without further assumptions. These assumptions are used in choosing a set F^0 of 'possible states of nature' f^0. If we assume that the problem is deterministic, then the problem is already solved for x_1. We call this assumption *local* as they refer to a single question. If the problem is probabilistic and we want to determine the probability of future answers y_1 to x_1 this requires the assumption of a set of stationary distributions $p(y_1|x_1, f^0)$ corresponding to possible states f^0 for y_1 so we can use the data to reweight the probabilities $p(f^0)$. The collection of $p(f^0)$ is what we call a 'state of knowledge' and can be used to predict future outcomes y_1.

But in most real world problems there are new questions x_2 for which no training examples are available. This is the *generalization* problem. It seems even harder than the learning problem for x_1 for which at least local data are available. To remedy the situation we clearly need answers which depend on y_2 at least indirectly. We may directly use a *nonlocal* assumption like $y_2 = y + y_1$. This is trivial if we know y. But this may not be the case in the beginning and this paper analyses how information about y may be obtained. Therefore, we want to start with independent relevant outcome y_1 and y_2 (what we will call a factorial state) and try to relate all dependencies to additional information. That means we see y as an answer to a question q with $y(q) = y_2 - y_1$. This *nonlocal* q question represents a measurement device for differences. We call questions depending on more than one y_i *generalized* questions. Thus nonlocal questions like $y_2 - y_1$ or questions refering to repeated measurements of the same questions like $y_1 + y_1'$ are examples therefore. We can now separate the nonlocal information in two parts: the structural information, that is the definition of the nonlocal question q and an empirical or controlable part, that is the result of measuring q or controlling its answer. The definition of q alone does not yet determine the value or probability of y_2. Only an actual measurement of q relates those two questions.

This is a conceptually clear starting point to analyse generalization. We start with an unrelated set of relevant questions x_i in which we are interested and formulate all nonlocal information as answers to nonlocal questions. How can we know what q measures? Clearly the definition of q,that is the structural information, will contain somey assumptions, specifically some kind of stationarity. The definition of q is usually also based on previous empirical information. For example, we could have tested a difference device many times before using it and found it working correctly (or at least indicating correctly its failure). Then the stationarity assumption that it will work correctly also for our task, does not seem very strong. (In a Bayesian approach we can give it a high subjective probability.) Thus, structural information is based on *transfer* of knowlegde between tasks. Transfer means generalization with respect to a task variable. In statistics a nonlocal question corresponds for example to the choice of a smoothness functional to be used as prior. Likewise, *rules* in an expert system with dependencies modeled by *logic, fuzzy logic* or *Bayesian belief networks* and *macroscopic variables* (e.g. energy) in physics are special cases of nonlocal questions.

Data are pairs of question and answer and we refer to the data related to relevant questions x_1, x_2 as *test data* and to data related to questions x_1, q with answers available as *training data*. Then for independent test questions the training data necessarily have to contain answers to nonlocal questions to allow generalization. It is from this point of view that we want to analyse prior information: we start with independent relevant questions and try to give explicit explanations, namely measurable prior information, for their dependencies. We summarize three aspects we have been mentioning

1. Generalization aspect: not all relevant questions are available for training,
2. Application aspect: not all available data correspond directly to relevant questions,
3. Transfer aspect: Some aspects of the setting, like the nature of the measuring devices, have to be known.

Now let us switch to more realistic statistical problems and consider a given function $f : y = f(x)$ with $x \in X$ and $y \in Y$ for some set X and some set Y. We allow $f(x)$ to be the realization of a random variable which takes values y with probability $p(y|x, f)$. Let us consider the following two kinds of information: A typical training example, that is a pair (x, y), and some smoothness constraint. Let the latter be given by a bound on a smoothness functional, like, for example, the discretized version $\sum_i (\frac{E(f(x_i)) - E(f(x_i+\Delta x_i))}{\Delta x_i})^2 < \theta$, where $E(f(x))$ denotes the regression function or expectation of $y = f(x)$ at point x.

The first one can clearly be interpreted as a pair of question and answer. We see $x = q_x$ as a question for function f about its value at x and denote the answer of function f by $q_x(f) = f(x) = y$, generated by a probability distribution $p(y|x, f)$. We write such a pair as $(x, y) = (q_x, q_x(f))$. Information about smoothness can be seen as answer to a more generalized 'smoothness' question $q_{s,\theta} = \text{sign}\left(\sum_i (\frac{E(f(x_i)) - E(f(x_i+\Delta x_i))}{\Delta x_i})^2 - \theta\right)$. We write this pair as $(q_{s,\theta}, q_{s,\theta}(f))$ with $q_{s,\theta}(f)$ equal to ± 1 and remark that replacing the fixed θ by a random variable Θ with mean θ is one easy way to generalize to the case of a probabilistic answer. Now note that in contrast to a simple example data question the smoothness question depends on more than one x–value of the function f. In this sense the smoothness question may be seen as nonlocal. Nonlocality is also present in symmetry questions. e.g. like $q_{p,\theta} = \text{sign}\left(\sum_i (E(f(x_i)) - E(f(-x_i)))^2 - \theta\right)$ or more general

$q_{s,\theta} = \text{sign}\left(\sum_i (E(f(x_i)) - E(f(sx_i)))^2 - \theta\right)$ where s represents any symmetry operator under which invariance of f is tested. Also bounds for the nonlocal maximum function $\max_x E(f(x))$ are widely used. Those three types of bounds, on smoothness, symmetry, and maximum functionals, are the the most frequently used forms of prior information in practice.

There is one practical distinction between nonlocal prior information like smoothness and standard training examples: the latter are assumed to be empirically measurable while the 'measurability' of nonlocal data like smoothness is often quite unclear. The difficulty lies in the fact that these nonlocal questions depend on the whole function $f(x)$, and for continuous x there is an infinite number of function values. But, if we want to treat nonlocal information in a similar way as standard training examples we also have to discuss the measurement of such data. This paper shows that, besides cases where highly parallel measurement devices are available, also restrictions of measurement devices and the definition of situations of interest and their control can be interpreted as measurements of such questions.

There is another group of nonlocal questions, for which data are more easily available in practice. These are nonlocal questions depending on only a finite (and in practice 'small enough') number of $f(x_i)$. Empirical answers to these questions can be found by measuring standard training examples $f(x)$ and applying some operations to the results which define the questions. Examples include differences $f(x_i) - f(x_j)$, weighted averages $\sum_i w_i f(x_i)$ like discrete wavelet components, symmetries s for specific points $f(x_i) - f(sx_i)$, and measurements of $f(x)$ with input noise. When the expectation $E(f(x_i))$ can be approximated by $1/n \sum_i x_i$, that is by using repeated measurements, we can, at least approximately, also measure examples depending on $E(f(x_i))$. Sometimes only nonlocal data are available, as in models where the variables of interest are hidden but related by structural assumptions to several observable variables, like for example in hidden Markov models. However, even when the variables of interest are observable there can be measurement devices which measure finite nonlocal questions with greater accuracy than by using the measuring devices for $f(x)$. Specifically, differences are often measured better directly than by first looking for $f(x_i)$ then for $f(x_j)$ and then taking the difference. This is always the case if part of the measurement error appears in a final step common to all measuring devices, like an output scale with fixed finite resolution or memory errors. We also remarked that the definition of a generalized question is called a rule in expert systems. Thus, available rules collected from experts may be incorporated as nonlocal question. Also, many examples of measuring nonlocal questions can be found in physics. The energy function of a macroscopic system is highly nonlocal with respect to the microscopic constituents. It seems that nonlocal data are often available, however in many cases their practical combination with priors of infinite nonlocality is probably somewhat hindered due to the difficulties of the Frequentist approach in statistics in dealing with nonlocal questions. That means that for data with infinite nonlocality the measurement problem represents an empirical difficulty, while including complex forms of nonlocal data, even with finite nonlocality, yields easily to mathematical difficulties related to solving nonlinear equations.

The paper formulates a theoretical approach in which answers to generalized questions are treated similar to standard training examples. In Section 2 the theoretical framework is presented, paying particular attention to the generalization problem. It is clarified that not the (standard training) data, but the nonlocal (prior) data enable learning.

Section 3 presents generalized questions in more detail. The importance of nonlocal informtion for learning makes it necessary to discuss the fundamental measurement problem for nonlocal questions. For example, restrictions of measurement devices correspond to nonlocal measurements.

Section 4 relates the classical approximate symmetry priors (e.g. smoothness) to nonideal measurement devices, the specific restriction being input noise.

Usually learning problems are defined and controlled by men. Consequently, nonlocal dependencies may be caused by internal human concepts. In addition experts may contribute knowledge in form of verbal statements. Section 5 discusses principal variants of including subjective priors (which is meant to include priors caused by subjects), like an interface with fuzzy priors. The Section does not aim to give an overview over this active and growing area of research, but presents some principles and aims in explaining the origin of nonlinearities which appear as technical difficulties in Section 9.

Section 6 extends the theoretical framework of Section 2 to decision problems using the language of generalized questions. Parallel and inverse settings are contrasted.

Section 7 contains a discussion of the Bayesian approach, concentrating an the saddle point approximation, leading to the two step MaP–MiR procedure.

Section 8 discusses the Frequentist approach of empirical risk minimization from the Bayesian decision theoretic point of view. Especially the relation to the maximum posterior approximation in approximation and nonapproximation situations is discussed in detail.

Finally, Section 9 gives examples of variational equations in a maximum posterior approximation including generalized information. In general those have the form of inhomogeneous integro–differential equations, similar to those in a time independent formulation of quantum mechanical scattering theory.

2 A constructive Bayesian approach

2.1 Model and vocabulary

In this section we choose a Bayesian approach (See for example Berger, 1985; Haussler, 1995; Bishop, 1995a, Wolpert, 1996a) and attempt to separate in a clear manner the local from the nonlocal parts in our model. Therefore, we begin with the construction of the local components: We enumerate a set of basis questions x necessary to determine the possible states we are interested in. The term *constructive* means that we do not start with a model given by some chosen parametrization (see Wolpert, 1994a), however, we use the relevant questions we are interested in to construct the related model. For every question we give a set of possible answers together with a set of question specific answer distributions $p(y|x, f_x^0)$. Each such answer distribution represents one possible pure local state f_x^0. Its index x allows the local states to be defined independently and individually for every basis question x. In this paper a pure state is indicated by an superscript 0. We assume the nature to be in exactly one pure state. Which one, is usually unknown to us. Thus, in contrast an f_x without the superscript 0 denotes not a real pure state but a state of knowledge being equivalent to an assignment of a probability $p(f_x^0|f_x)$ to every pure state, where we often skip f_x in the notation. In a learning situation we assume the actual, usually unknown, pure state (of nature) to be constant[1] for the time under study, while the state of knowledge, which reflects the learning process, changes with our information.

The independently defined local components are related by nonlocal states. A pure nonlocal state f^0 is defined through an assignment of one local state f_x^0 to every basis question x and a nonlocal state of knowledge f through an assingment of a probability $p(f^0)$ to every pure nonlocal state, with $p(f^0)$ implicitely understood as $p(f^0|f)$. The probabilities $p(f^0)$ contain all the relations between local components and therefore the nonlocal information.

We now define the ingredients of the theoretical approach more formally:

1. A *local basic model* consisting of
   a. a (finite[2]) set X of *basis questions* x,
   b. a (measurable[3]) space Y_x of possible answers y_x for every question[4],
   c. a (finite[5]) set of probability distributions (local elementary functions) over answers $F_x^0 = \{p(y|x, f_x^0)\}$ for every question, the *pure local states* ,
   d. a set F_x of local states of knowledge f_x defined by assigning a probability $p(f_x^0)$ to every local pure state f_x^0.

A local state f_x^0 can be considered indexing a vector $p(y|x, f_x^0)$ in the linear space $\mathcal{F}(Y, x)$ of functions defined on $Y = Y_x$, i.e. $f_x^0 \in \mathcal{F}(Y, x)$, with norm $|f_x^0| = \sum_y p(y|x, f^0) = 1$. A state of knowledge f_x is in the convex hull of F_x^0, with $\sum_{f_x^0} p(f_x^0) = 1$. We denote by $\mathcal{C}(V)$ the convex hull of a set V of vectors v_i generated by linear combinations $\sum_i a_i v_i$ with coefficients fulfilling $\sum_i a_i = 1$ (e.g. $i = f_x^0$, $a_i = p(f_x^0|f_x)$), and by $\mathcal{L}$ the linear span, generated by unrestricted linear combinations. So we have

$$F_x^0 \subseteq F_x = \mathcal{C}(F_x^0) \subset \mathcal{L}(F_x^0) \subseteq \mathcal{F}(Y, x).$$

Now we construct the nonlocal parts.

2. *Pure nonlocal states* f^0, or, more shortly, pure states, are defined by a set of pairs $(p(y|x, f_x^0), x)$ containing exactly one local state for each question: $\{(p(y|x, f_x^0), x) : f_x^0 \in F_x^0, x \in X\}$ to give $p(y|x, f^0) = p(y|x, f_x^0)$. They form the set F^0 of pure states or elementary functions, which is a subset of the linear space of functions defined on Y_x, X, $f^0 \in \mathcal{F}(Y_x, X)$. We implicitly understand states (functions) as equivalence classes defined with respect to X by $[f^0]_X = [f'^0]_X \Leftrightarrow \forall x \in X, \forall y \in Y_x : p(y|x, f^0) = p(y|x, f'^0)$.

3. A *nonlocal state of knowledge* f, or, more shortly, a state of knowledge is defined through assigning probabilities $p(f^0|f)$, or often shortly $p(f^0)$, to the pure states $f^0 \in F^0$ of the model. Thus, f denote convex combinations of pure states f^0, i.e. $p(y|x, f) = \sum_{f^0 \in F^0} p(f^0)p(y|x, f^0)$, with $\sum_{f^0 \in F^0} p(f^0) = 1$,. The set F of possible states of knowledge (also called mixture states) f of the model form the convex hull $F = \mathcal{C}(F^0)$ of $F^0 \subset \bigotimes_x \mathcal{L}(F_x^0)$. By construction we have $p(f^0|x) = p(f^0)$ stating independence between states f^0 and basic questions x. A state of knowledge is called *factorial* with respect to X iff $p(f^0) = \prod_x p(f_x^0); \forall f^0 \in F^0$. It is fully determined by its local probabilities $p(f_x^0)$ assigned to each local state for question x.[6]

This paper discusses predefined dependencies between answers to different questions. The role of such *structural information* is discussed in section 2.3. Structural information is implemented by defining generalized questions.

4. A *generalized question* is any probabilistic functional $q(f)$ of f. A probabilistic functional q is a probabilistic mapping from the space F of functions f to a space y^q of answers $y = q(f)$ given by probability distributions $p(y|q, f) = \sum_{f^0 \in F^0} p(f^0)p(y|q, f^0)$ defined by $p(y|q, f^0)$. We require the f^0–dependency of $p(y|q, f^0)$ to be ex-

[1] Constant is meant relative to a chosen reference system, which might also, for example, be varying in time.

[2] For infinite X the corresponding functional integration represents a stochastic process in the language of mathematics or a field theory in the language of physics and must be consistently defined for every finite subset (Gardiner, 1990; van Kampen, 1992; Schervish, 1995; Bialek, Callan, Strong, 1996, Balasubramanian, 1996). Gaussian processes have been used in Bayesian statistics (see e.g. Williams, Rasmussen, 1996; Barber, Williams, 1997; Neal, 1997).

[3] In the mathematical sense.

[4] The set Y_x can be assumed x–independent without loss of generality, so we will usually write $Y_x = Y$.

[5] The case of an infinite space F_x^0 requires the definition of a measure df_x^0.

[6] $\sum_{f_x^0 \in F_x^0} p(f_x^0) = 1, \forall x \in X$ ensures $\sum_{f^0 \in F^0} p(f^0) = 1$.

pressable in terms of only the $p(y|x, f^0)$.[7] [8] [9] We define some set Q of generalized questions to be *measurable* (or observable) by assuming, for every question in Q, the existence of a measuring device producing the corresponding answer. As consistency condition we require any question depending on a finite number of answers to measurable questions also to be measurable. We call the set of $x \in X$ on which the $q \in Q$ depend the *basis*[10] of Q and denote it by X^Q. Skipping the unobservable elements $x \notin X^Q$ we write $X = X^Q$. Data D are pairs of questions and corresponding answers (q, y^q). In section 3.1 we will show how to write generalized questions explicitly, and section 3.2 discusses the measurement processes.

For applications we use a decision theoretic framework (see also Section 6) and separate the two subsets

5. $Q^D \subseteq Q$ of *available* or *training questions* for which answers are available.[11] The basis of Q^D will be denoted by $X^{Q^D} = X^D \subseteq X$. D_{Q^D} denotes the set of all possible data which only depend on questions $q \in Q^D$. We denote by q^D, and analogous for other sets of questions, vectors with components $q_i^D \in Q^D$. Here, $q_i^D = q_j^D \in Q^D, i \neq j$ is also allowed.[12] By $q^D \in Q^D$ we mean that $q_i^D \in Q^D$ for every component of the vector. If a vector appears in Q^D we understand not the vector but its components to be elements. An data vector D is a question–answer pair (q^D, y^D). We may decompose D into lower dimensional vectors $D_i = (q_i^D, y_i^D) \subseteq D_{Q^D}$, $i \in I$.

6. $Q^l \subseteq Q$ of *relevant* or *test questions* defining the possible application situations. The basis of Q^l is denoted by $X^{Q^l} = X^l \subseteq X$, D_{Q^l} the set of all possible data depending on Q^l, and we use D^l to denote a test data vector[13] i.e. a set of pairs $(q^l, y_{q^l}) \in D_{Q^l}$.

The set Q^l is called relevant, because

7. we assume a given *loss function* $l(q^l, y_{q^l}, z)$, defined for all $q^l \in Q^l$, i.e. application situations. The loss function depends on the question q^l, the answer y_{q^l} and potentially also on additional variables $z \in Z$. We will call z the *action variables* as we usually allow them to be controlled.

We include the possibility to control the action variables $z \in Z$ within the loss function by an active choice of an action state $\hat{f} \in \hat{F}$. Thus, we define

8. a family $\hat{f} \in \hat{F}$ of possible *action states* producing the probabilistic action $z \in Z$ according to $p(z|q^l, y, \hat{f})$ for $q^l \in Q^l$.

We write $\tilde{l}(q^l, y, \hat{f})$ for the effective loss if the z variables in $l(q^l, y, z)$ can be integrated out.

The possible application situations, i.e. the relevant or test questions q^l, are generated by

9. a *test question* q^l–producing device $p(q|y_c, z_c)$ which conditioned on a subset c of 'past' values of $y_c \in Y$ and $z_c \in Z$ does not depend on f^0, f or $\hat{f}$.

The probability distributions $p(y|q, f^0)$, $p(q|y_c, z_c)$, and $p(z|q^l, y, \hat{f})$ define a $\hat{f}$–dependent loss distribution $p(l|f, \hat{f})$:[14]

$$p(l|f,\hat{f}) = \int dq \int dy \int dz\, p(q|y_c, z_c, \hat{f})$$

$$\times p(y|q, f) p(z|q, y, \hat{f}) \delta(l(q, y, z, \hat{f}) - l).$$

The normative component is represented by the requirement to minimize

10. a *risk functional* $r[p(l|f, \hat{f})]$, which is a mapping from the loss distribution $p(l|f, \hat{f})$ into a subset of the real numbers, bounded from below. A common risk functional is the expectation of l or *expected risk* $r(f, \hat{f}) = \int dl\, p(l|f, \hat{f}) l$. [15]

New available data D require updating of an initial state of knowledge f^I to obtain a new f.

11. A *learning* model is a mapping $f = f(D, f^I)$ from F to F parametrized by D. The initial state will be called *prior* state. The Bayesian learning model is defined by

$$p(f^0|f(D, f^I)) = \frac{p(y^D|q^D, f^0) p(f^0|f^I)}{p(y^D|q^D)}.$$

[7] Aside from the possibility that a functional is not defined for a specific f^0, there exists the possibility of a functional having only a certain probability of being not defined if applied to a function f^0, like $q(f^0) = 1/y$ if $p(y = 0|x, f^0) \neq 0$. Formally one can add the value 'undefined' to the space Y, as it is common in programming.

[8] One could allow a dependence of the definition of q, i.e. the $p(y|q, f, f^0)$, from the state of knowledge f (which is known, as the name indicates), i.e. from the $p(f^0|f)$. However, this is only important when studying the dependence from f. A dependence $p(f^0|q)$ would mean that selecting q already changes the hidden variable f^0. By definition of the model we have $p(y|x, f, f^0) = p(y|x, f^0)$ and $p(f^0) = p(f^0|x)$ for the basic questions.

[9] Compare for example with similar concepts in Ratsaby & Maiorov, 1996 and Smola & Schölkopf, 1997.

[10] This is not a linear basis of a vector space (Compare Jeffrey, 1968).

[11] This is related to learning (see 11., below). The set Q^D depends on what part of information is considered part of the prior A prior f^I can be expressed depending on some data D^0 and another prior $f^{I'}$ according to $p(f^0|D) = p(f^0|f(D, f^I))$ $= p(f^0|f(D, f^I(D^0, f^{I'}))$=$p(f^0|D, D^0)$. Note that here the sloppy notation $p(f^0|D) = p(f^0|D, D^0)$ refers to different priors on the right and left hand side which are not explicitly indicated.

[12] In as far as no linear structure is needed q^D can also be seen as a set, with dummy indices for repeated elements.

[13] Not to be confused with the validation set of empirical test data like it is used in cross–validation (see Section8.4).

[14] See Section 3 and Section 6 for details and justification of the chosen components and their probability distributions.

[15] Minimizing an expected loss was already proposed by (Laplace, 1810ab) and later revived by (Wald, 1938) (See the historical remarks in (Le Cam, Yang, 1990)). A general formalization can be found in (Le Cam, 1986).

We will from now on skip f^I in the notation and write $p(f^0|f(D,f^I)) = p(f^0|D)$ and $p(f^0|f^I) = p(f^0)$. We can write this in a form

$$p(f^0|D) = \frac{\sum_{f'^0} T^D(f^0,f'^0)p(f'^0)}{\sum_{f''^0}\sum_{f'^0} T^D(f''^0,f'^0)p(f'^0)},$$

or shortly, in matrix notation

$$p^D = \frac{Tp}{\mathrm{Tr}\,TP},$$

with $P(f'^0,f''^0) = p(f'^0)$ and Tr denoting the trace. The so defined matrix

$$T^D(f'^0,f^0) = \delta_{f'^0,f^0}p(y^D|q^D,f^0),$$

is diagonal in the f^0–representation. This shows that the pure states f^0 represent the possible fixed points of learning.[16]

Section 2.2 discusses learning in factorial states.

The local basic model including the local probabilities $p(f_x^0)$ represent the *local part of prior information*, the definition of generalized questions the *structural part of prior information*. Thus, a factorial state has only local and structural prior information and in any state of knowledge the nonlocal, nonstructural information should be information resulting from measurement or control added to a factorial starting state. After clarifying the importance of nonlocal information we will discuss their possible measurement. Table 1 summarizes some of the notations.

Notations

$x \in X$	basis questions
$q \in Q$	generalized questions
$y \in Y$	(probabilistic) answers
$\hat{y}, \hat{q}$	(probabilistic) actions
z	internal variables
$D = (D^q, D^y)$	data
$f^0 \in F^0$	pure states
$f \in F$	states of knowledge
$\hat{f} \in \hat{F}$	action states
$p(y\|q,f^0)$	probability (or density) for answer y under question q in state f^0
$p(z\|x,y,q)$	probability (or density) for internal z given data (x,y) in question q
$p(\hat{y}\|q,\hat{f})$	probability (or density) for action $\hat{y}$ under question q in action state $\hat{f}$
$p(\hat{q}\|y,\hat{f})$	same in inverse model for action $\hat{q}$
$p(f^0\|f)$	probability of f^0 under f
$\bar{y}_x(f^0)$	regression function at x in state f^0
$L = \ln p$	logprobability
$l(q,y,z)$	loss function
$\tilde{l}(q,y,\hat{f})$	loss function integrated over probabilistic action or for deterministic action
r	risk functional

Table 1.

[16]Notice, that if T^D has degenerate eigenvalues, also non-pure states may be unchanged. See Theorem in Section 2.2.

Examples of basis questions

The basis questions $p(y|x,f^0)$ define the answer probabilities for the available data Q^D and relevant questions Q^l. Their definition is therefore not independent of the available measurement devices for Q^D and Q^l, and they cannot be choosen arbitrarily. Choosing a set of $p(y|x,f^0)$ is part of the local prior knowledge. Mainly interested in generalization we do here not concentrate on the local part. We assume a formulation of the problem with a local prior $p(f_x^0)$ where the local part is learnable, i.e. where the local state of knowledge $p(f_x^0|f_x(D^x))$ asymptotically converges to one f_x^0 under local questions x. according to some convergence criteria of our choice. This is without loss of generality, because we could always consider questions about the possible densities $p(y|x,f^0)$, i.e. study the density approximation problem. Practically, this does not change the situation, but splits formally the local part into several parts, so the problem appears as a nonlocal one. [17] Also, in a possible loss function for density

Also, x must not necessarily be a single minimal component, but we can combine many x (with e.g. previously learned dependencies) to a larger x vector. Technically, one x just denotes one independently parametrized of subset of F^0, and the question of generalization is the question of generalization betweeb such sets.

Basis questions can be Gaussian

$$p(y|x,f^0) \propto e^{-\frac{(y-\bar{y}_x(f^0))^2}{2\sigma_x^2}},$$

so that states f^0 are parametrized by their regression function $\bar{y}_x(f^0)$. In general, the parametrization of the $p(y|x,f^0)$ (and therefore of the states) can be arbitrary, e.g. also the variance σ_x^2 or higher order moments can be f^0–dependent.

Consider as a more complex example image y producing states (generative models), e.g. with x having the values face and nonface. States f^0 are defined by their generation probabilities for images of faces $p(y|\mathrm{face},f^0)$ and of nonfaces $p(y|\,\mathrm{nonface},f^0)$. Generation of faces in a state f^0 could be defined

$$p(y|\,\mathrm{face},f^0) = \int dv\, p(v|\,\mathrm{face},f^0)p(y|v,\mathrm{face},f^0),$$

with v being an index for the different variants of a face. Possible variants include for example interpersonal differences, varying view points or changing illumination conditions. Using some interpolation scheme, like optical flow and correspondance of some reference points, a continuous v could be constructed out of a a discrete set of examples. Also, human prior knowledge may be that faces have constituents j like two eyes, mouth and nose appearing and being combined in different variants. In the easiest version with independent constituents one could choose

$$p(y|v,\mathrm{face},f^0) \propto e^{-\sum_{i,j}\frac{(d_i^{v,j})^2}{2\sigma^2(i,j,v)}},$$

[17]In a usual density estimation problem there is only one x (with the meaning 'get the next y'), and, besides a local positivity restriction, a natural (nonlocal) normalization condition.

using some distance $(d_i^{v,j}(y_i))^2 = ||y_i - y_{v,j}^i||^2$ with i being the pixel index and $y_i^{v,j} = y_i^{v,j}(f^0)$ being a template for variant v for constituent j. Then the $p(v|\mathrm{face}, f^0)$ parametrize the face states.

2.2 Analysis of generalization

2.2.1 Minimal models and sufficient data

The ability to generalize is essential for any real learning. It is easy to see that generalization requires *nonlocal dependencies* contained in the $p(f^0|f)$. Let us denote by $D_{X\setminus x}$ data which do not depend on x and combine all $f^0_{x'}$ for $x' \neq x$ into $f^0_{X\setminus x}$. Then using $\sum_{f^0_{X\setminus x}} p(f^0_{X\setminus x}|D_{X\setminus x}) = 1 = \sum_{f^0_{X\setminus x}} p(f^0_{X\setminus x})$ we see that in a factorial state where $p(f^0_x|f^0_{X\setminus x}) = p(f^0_x)$

$$p(y|q, f'(D_{X\setminus x}, f)) = \sum_{f^0} p(y|q, f^0)p(f^0|D_{X\setminus x})$$

$$= \sum_{f^0_x} p(y|x, f^0_x) \sum_{f^0_{X\setminus x}} p(f^0_{X\setminus x}|D_{X\setminus x})p(f^0_x|f^0_{X\setminus x}) = p(y|x, f).$$

This means that data not depending on x can never change the answer probabilities to x. We can say that a factorial state allows no inference and represents a 'tabula rasa' situation with respect to generalization. Thus, starting from a factorial state we necessarily need nonlocal information to enable any nonlocal learning. We can always relate $f = f(D, f_{fact})$ to a factorial state f_{fact} by

$$p(f^0|f) = p(f^0|D) = p(D|f^0)p_{fact}(f^0)/p(D).$$

Now we formulate this observation in a bit more general way and show under which conditions the conclusion can be reversed. We begin with a simple example to outline the general idea. We consider an example, where F^0 does not only consist of extremal points of the convex set F. Let a space F^0 with three possible pure states, be defined by $p(y = 1|x, f^0_1) = 1$, $p(y = -1|x, f^0_2) = 1$, $p(y = \pm 1|x, f^0_3) = 0.5$. This may be the probability that a certain gender is marked on an application form for girl schools, boy schools, and coeducated schools. For example a state of knowledge f with $p(y = \pm 1|x, f) = 0.5$ can be expressed as

$$p(y|x, f) = \frac{1-a}{2}(p(y|x, f^0_1) + p(y|x, f^0_2)) + ap(y|x, f^0_3)$$

$$= p(f^0_1)p(y|x, f^0_1) + p(f^0_2)p(y|x, f^0_2) + p(f^0_3)p(y|x, f^0_3)$$

for any $0 \le a = p(f^0_3) = 1 - \sum_{i=1}^2 p(f^0_i) \le 1$. Now assume, that some data not related to x change the probability of $p(f^0_3) = a$. Obviously, this has no influence on $p(y|x, f)$. We will call such a space F^0 nonminimal with respect to x, and the set of data (x, y) not sufficient with respect to F^0. Now consider a space Q_x of local questions q_x which includes also repeated measurements of x. The probability for measuring independent pairs $(y(x), y'(x))$ is the product $p(y, y'|x, f) = p(y|x, f)p(y'|x, (y, x))$ which changes with $p(f^0_3)$. The probability for a measurement $(y = 1, y' = 0)$ would be zero if $p(f^0_3) = 0$, but 1/4 if $p(f^0_3) = 1$. Then, the coefficients $p(f^0_i)$ which define a state f are unique, and if they change, they change the probability distribution of some question q_x. We will say the space F^0 is minimal with respect to the set of data $D_x = (q_x, y_{q_x})$, and data D_x are sufficient for F^0.

We define equivalence classes $f^0_{D_Q} = [f^0]_{D_Q}$ of D_Q–*equivalent* states for the set of data D_Q of the form (q, y) with $q \in Q$ by

$$f^0_{D_Q} = {f^0_{D_Q}}' \Leftrightarrow \forall (q, y) \in D_Q : p(y|q, f^0) = p(y|q, f'^0),$$

forming the set $F^0_{D_Q}$. In the case in which the data contain all $y \in Y_q$ for every question $q \in Q$ we speak of Q–equivalent states and write f^0_Q. We defined $F^0 = F^0_X$. The same constructions can be applied to states of knowledge yielding f_Q and F_Q.

We define data D_Q to be *sufficient* with respect to F^0 or equivalently the set F^0 to be *minimal* with respect to D_Q iff all states of knowledge $f \in F(F^0_{D_Q})$ are uniquely decomposable into the $f^0_{D_Q}$, that is iff there is no solution $p(f^0_{D_Q})$ of the following system of homogeneous linear equations

$$0 = \sum_{f^0_{D_Q}} p(y|q, f^0_{D_Q})p(f^0_{D_Q}).$$

This means that no pure state can be expressed by others

$$p(y|q, f^0_{D_Q}) = \sum_{f'^0_{D_Q} \neq f_{D_Q}} p(y|q, f'^0_{D_Q})p(f'^0_{D_Q}).$$

Then the corresponding system of inhomogeneous linear equations for $p(y|q, f^0_{D_Q})$ is overdetermined and therefore there exists at most one solution for the state $p(f'^0_{D_Q})$. (At least one solution exists by construction.) This means that sufficient data determine the state of knowledge f uniquely. In other words, the convex hull F_{D_Q} of $F^0_{D_Q}$ does not contain equivalent states, i.e. $[F(F^0_{D_Q})]_{D_Q} = F(F^0_{D_Q})$.

To shorten the notation we can write for

$$p(y|q, f) = \sum_{f^0_{D_Q}} p(y|q, {f^0}_{D_Q})p(f^0_{D_Q}),$$

introducing indices $i = (q, y)$, $j = f^0_{D_Q}$

$$p_i = \sum_j A_{ij} p^f_j,$$

i.e.

$$p = Ap^f.$$

The matrix (or integral operator with kernel A_{ij}) $A = A(D, F^0)$ describes the model F^0 with components f^0 on a data vector D with components (q, y) and p^f with components $p(f^0|f)$ is the state of knowledge. Minimality requires the number of independent pure states $f^0_{D_Q}$ to be not larger than the number of data, i.e. question–answer pairs (q, y), which means that the rank of the matrix $A_{((y,q),f^0_{D_Q})} = p(y|q, f^0_{D_Q})$ is equal to the number of $f^0_{D_Q}$. Summarizing, minimality of the model with respect to data, or sufficiency of the data for a model, is defined as a situation where the 'model-data matrix' A has rank equal to the dimension of p^f, i.e the number of $f^0_{D_Q}$.

Minimality/sufficiency can be achieved by deleting some f^0 or by including more y or new q in the minimality condition. This can be done by including new independent questions, using a finer scale or add new dimensions to y. Examples include repeated measurements of the same x, $Q_x^{(n)}$ consisting of n–tuples $(x_1 = x, x_2 = x, \cdots, x_n = x) = \times_{i=1}^n x$, $Q_x = \bigcup_{n=1}^\infty Q_x^{(n)}$, multiple measurements within X, $Q_X^{(n)}$ consisting of n–tuples $\times_{i=1}^n x_i$, with $F_{Q_X^{(n)}} \subset \bigotimes_i \mathcal{L}(F_{x_i})$ or of varying length $Q_X = \bigcup_{n=1}^\infty Q_X^{(n)}$ with $F_{Q_X} \subset \bigoplus_{n=1}^\infty \bigotimes_i^n \mathcal{L}(F_{x_{i,n}})$[18].

For two–components data vectors from $Q_X^{(2)}$ the probability $p(y_1, y_2|x_1, x_2, f) = p_{ij}^{(2)}$ has to fulfill,

$$p_{ij}^{(2)} = \sum_j A_{ik} A_{jk} p_k^f = \sum_j A_{ik} A_{k'j}^T \delta_{k,k'} p_{k'}^f,$$

and therefore

$$p^{(2)} = (A \otimes A) P^{f,(2)},$$

with diagonal $P_{k,k'}^{f,(2)} = \delta_{k,k'} p_k^f$. For general data vectors $Q_X^{(n)}$

$$p^{(n)} = (\bigotimes_n A) P^{f,(n)},$$

with $P^{f,(n)}_{k,k_1,k_2,\cdots,k_{n-1}} = \prod_{i=1}^{n-1} \delta_{k,k_i} p_k^f$. A model minimal for single measurements is minimal for multiple measurements. For a minimal model the f^0 are linear independent, and in case the number of data (equations, conditions) is larger than the number of f^0 there must exist a reduced $n \times n$ system with nonzero determinant, so the solution is unique. Thus, we choose a decomposition $A = \binom{A'}{A''}$ with A' a square $n \times n$ matrix, so its determinant is defined, and for a minimal model there exists an A' with $\det A' = 0$. The relation $\det(A' \otimes A') = (\det A')^{2n}$ for $n \times n$ matrices shows that the determinant for multiple measurements for the reduced system A' is nonzero if it is nonzero for single measurements. If a given solution is already unique for a reduced system, it is also unique in an extended system, where just more conditions are added, consistent with the solution. Thus, models minimal for single measurements $Q_X^{(1)}$ are also minimal for multiple measurements, i.e. vectors $Q_X^{(n)}$.

Nonminimal local spaces F_x^0 are not commonly used. Minimal for example is a local space consisting of Gaussians at different locations. Then a convex linear combination is not Gaussian but a Gaussian mixture state.

2.2.2 Factorial priors

Let us now consider two sets of data, Q^l (e.g. relevant questions) and Q^D (e.g. training questions) with corresponding (test) data D^l and (possible sets of training) data $D = \{D_i, i \in I\}$. The $D_i = (y_i, q_i)$ are allowed to be data vectors and may represent one possible collection of data which can be obtained during training.

In a minimal model the following theorem states that the prior probabilities reflect already the possibility or impossibility of generalization:

Theorem: For a set of (test) data $D^l \subseteq D_{Q^l}$ sufficient for $F_l^0 = F_{Q^l}^0$ (or equivalently for F_l^0 minimal with respect to Q^l) and another set of (potential training) data $D \subseteq D_{Q^D}$ sufficient for $F_D^0 = F_{Q^D}^0$ (or equivalently F_D^0 minimal with respect to D) the following proposition holds

$$\forall (q, y) \in D^l\,, \forall D_i \in D : p(y|q, f) = p(y|q, f'(D_i, f))$$
$$\Leftrightarrow \forall f^0_{l,D} : p(f^0_{l,D}) = p(f_l^0, f_D^0) = p(f_l^0) p(f_D^0),$$

where $f^0_{l,D} = f^0_{l \cup D} = [f^0]_{D_{Q^l} \cup D}$. The backward direction does not require the two sufficiency conditions.

The theorem gives conditions under which conditional independence of all relevant data of all training data, is equivalent to independence between all f_l^0 and f_D^0. The stated conditional independence also means that the actual state of knowledge is an eigenstate for all matrices $T^{D_i}(f'^0, f^0) = \delta_{f'^0, f^0}\; p(y^{D_i}|q^{D_i}, f^0)$, or that all *mutual informations*

$$\ln \frac{p(y|q^l, f, D_i)}{p(y|q^l, f)} = \ln \frac{p(y, q^l, D_i|f)}{p(y, q^l|f) p(D_i|f)},$$

are zero, and therefore also all averages of them.

Proof: We show that factorial priors do not allow generalization, and that sufficiency of D^l and D excludes the stateed no–generalization property for other priors.

For $q \in Q^l$, abbreviating $p(y|q, f'(D_i, f))$ by $p(y|q, D_i)$, we write for the no–generalization condition

$$p(y|q, f) = p(y|q, D_i) = \sum_{f^0} p(y|q, f^0) p(f^0|D_i)$$
$$= \sum_{f_l^0} \sum_{f_D^0} p(y|q, f_l^0) p(f_l^0|D_i) p(f_D^0|f_l^0, D_i)$$
$$= \sum_{f_l^0} p(y|q, f_l^0) p(f_l^0|D_i),$$

because for $q \in Q^l$ the probability $p(y|q, f^0)$ only depends on f_l^0 and $\sum_{f_D^0} p(f_D^0|f_l^0, D_i) = 1$. Another summation over finer classes up to D_X is not necessary because $\sum_{f^0_{l \cup D \cup D_X}} p(f^0_{l \cup D \cup D_X}|f^0_{l \cup D}) = 1$. Setting $p(f_l^0|D_i) = p(f_l^0)$ yields $p(y|q, f) = p(y|q, D_i)$ giving one solution of the nonlearnability condition. For a D^l–sufficient model the state $p(f_l^0|D_i)$ is uniquely determined by the probabilities for the relevant data $p(y|q, f^0)$, $q \in Q^l$ and $y \in y^q$ so $p(f_l^0|D_i) = p(f_l^0)$ is also the only solution. Thus, sufficiency of D^l excludes the possibility that the influence of data only consists in switching between D^l–equivalent states.

Now we show that independence of f_l^0 of the data D_i together with the minimality of F_D^0 with respect to

[18]Which is similar to the construction of the Fock space for a many particle system, except that here the space is restricted to the convex hull (constant L_1 norm) and not to a region with constant L_2 norm.

D only allows a factorized prior probability. We insert a summation over f_D^0 into $p(f_l^0|D_i)$ and write for the condition $p(f_l^0|D_i) = p(f_l^0)$

$$p(f_l^0) = p(f_l^0|D_i) = \sum_{f_D^0} p(f_D^0|D_i)p(f_l^0|f_D^0).$$

One sees that the condition $p(f_l^0|D_i) = p(f_l^0)$ is fulfilled for $p(f_l^0|f_D^0) = p(f_l^0)$. This already solves the backward direction without the need of any minimality or sufficency condition. For a model not minimal on F_D^0 there still might be different states on D leading to the same posterior $p(f_l^0|D_i)$ on F_l^0. We now use minimality of the data D, to exclude the possibility, that there are dependencies which cannot be explored by D. The probability $p(f_D^0|D_i)$ is related to the definition of the training questions, i.e. to the $p(y_i^D|q_i^D, f_D^0)$, by

$$p(f_D^0|D_i) = \frac{p(y_i^D|q_i^D, f_D^0)p(f_D^0)}{\sum_{f_D^0} p(y_i^D|q_i^D, f_D^0)p(f_D^0)}.$$

Inserting this equation into the above equation for $p(f_l^0|D_i)$ shows that according to the assumption of sufficient data D the coefficient vector $a(f_D^0)$ multiplying the matrix $A_{i,f_D^0} = p(y_i^D|q_i^D, f_D^0)$ must be unique:

$$\frac{p(f_l^0)p(f_D^0)}{\sum_{f_D^0} p(y_i^D|q_i^D, f_D^0)p(f_D^0)}, = \frac{p'(f_l^0|f_D^0)p'(f_D^0)}{\sum_{f_D^0} p(y_i^D|q_i^D, f_D^0)p'(f_D^0)},$$

where $p'(f_l^0|f_D^0)p'(f_D^0) = p'(f_l^0, f_D^0)$ denotes another solution of the joint probability. Summation over f_l^0 and f_D^0 gives equality for the f^0–independent denominators on both sides so that $p(f_l^0, f_D^0) = p(f_l^0)p(f_D^0)$ is the only solution. q.e.d.

Without restriction to sufficient data there might exist spurious dependencies between equivalent states of knowledge, which are not observable within the given set of relevant questions.

The formal structure of the Theorem and of its proof becomes clearer if written in a more abstract matrix formulation. With $i = (q, y)$, $j = (q^D, y^D)$, $k = f_l^0$, $l = f_D^0$, $A_{ik} = p(y|q, f_l^0)$, $B_{jl} = p(y^D|q^D, f_D^0)/p(y^D|q^D)$ (where for observed data the denominator is unequal to zero), we can write for the posterior $p(y|q, D)$, in components

$$p_{ij} = \sum_{k,l} M_{ij,kl} p^f_{kl}$$

$$= \sum_{k,l} A_{ik} B_{jl} p^f_{kl},$$

which reads for matrices

$$p = (A \otimes B)p^f.$$

Theorem (matrix formulation):
For a (sub)system of equations with invertible $n \times n$ matrix[19] A and $m \times m$ matrix B, i.e. with $\det A \neq 0 \neq \det B$ (Minimality/Sufficiency), the following holds

$$p_l \otimes 1 = p_l \otimes p_D = (A \otimes B)p^f \Leftrightarrow p^f = p_l^f \otimes p_D^f.$$

Indeed, according to

$$\det(A \otimes B) = (\det A)^n (\det B)^m$$

with A and B also $A \otimes B$ is invertible and using

$$(A \otimes B)^{-1} = (A^{-1} \otimes B^{-1})$$

and

$$(A \otimes B)(p_l \otimes p_D) = (Ap_l \otimes Bp_D)$$

we find

$$p^f = (A^{-1}p_l \otimes B^{-1}p_D) = p_l^f \otimes p_D^f,$$

thus p^f factorizes. Formulated for probabilities conditioned on D, we used $p_D = 1$ for the theorem. Formulating the theorem for joint probabilities $p(y, q, D|f)$, or similarly $p(y, y^D|q, q^D, f)$, symmetrizes the formulation and gives $A_{ik} = p(y, q|f_l^0)$, $B_{jl} = p(y^D, q^D|f_D^0)$, $(p_l \otimes p_D)_{ij} = p(y, q|f_l^0)p(y^D, q^D|f_D^0)$, with still $M = A \otimes B$ according to the definition of the model.

Formulated in a basis X the theorem gives the
Lemma : For $(q_x, y) \in D^x \subseteq D_x$, $D^{X\setminus x} \subseteq D_{X\setminus x}$ and requiring for the forward direction sufficieny of the set D^x for F_x^0 for all x and sufficieny of the set $D^{X\setminus x}$ for $F^0_{X\setminus x}$ for all x the following holds

$$\forall x \in X, \forall (q_x, y) \in D^x, \forall D_i^{X\setminus\{x\}} \in D^{X\setminus\{x\}}:$$

$$p(y|q_x) = p(y|q_x, D_i^{X\setminus\{x\}})$$

$$\Leftrightarrow \forall f^0 \in F^0 \,:\, p(f^0) = \prod_{x\in X} p(f_x^0).$$

Remarks:

1. The theorem states that a factorial state remains factorial after local learning and that there is no generalization possible across a 'factorization border' for *any* learning algorithm. In factorial states any information concerning answers to question x can only come from questions depending explicitly on the point x of interest, but not necessarily depending only on x. Answers to questions depending only on $X \setminus X^D$ cannot be learnt under a factorial prior, and training data not depending on X^l are uninformative with respect to relevant questions if not combined with other information. They can thus only have indirect influence. For analysis of prior information in terms of relevant questions we may choose $X = Q^l$. Then in order to enable learning for every $q^l = x$ for every x we must have data D depending (maybe not only) on that x. For example, a smoothness constraint in statistics depends on all x and can allow generalization. Especially easy to analyse is the situation when $Q^D \subseteq X$. Then all relevant questions are defined with respect to the data. In this case a factorial prior on X refers to the training questions itself and the relevant questions are directly defined in their dependency on the training questions. We discuss the relations in detail in the next subsection.

[19] For simplicity we skip the prime for A and B which we used previously for reduced matrices.

2. The forward direction, factorial state $\Rightarrow$ no generalization, is related to so called 'No Free Lunch'–theorems (Wolpert, 1996a, 1996b), generalized from uniform priors (or uniform meta priors) to factorial states, without explicitly refering to a specific form of loss function or algorithm. Indeed, *uniform priors* of the form $p(f_x^0) = p(f_{x'}^0), \forall x'$ — possibly also with uniform probabilities $p(f_x^0) = p(f_x'^{,0}), \forall f_x'^0, \forall x$ within single x–components so that $p(f^0) = p(f'^{,0}), \forall f'^0$ — are special factorial priors. With respect to nonlearnability of specific sets of questions the theorem is sharper than results from the theory of uniform convergence (Vapnik, 1982) or regularization theory (Tikhonov, 1963), and independent of a specific loss function. However, it does not give quantitative results. Also, a change of $p(y|q,f)$ does not necessarily need to change the decision $\hat{f}$ (see 4.). A quantitative measure of dependency can be obtained by averaging the mutual information over a distribution for data D_i and D^l, or by integrating out D^l and calculate an average change of the risk under a data distribution $p(D_i)$.

3. The backward direction gives sufficient conditions for a no–generalization state to be factorial. The main point is that if the model has only one way to express the same state then it cannot switch between equivalent descriptions and there is no possiblity to create formal dependencies without visible effects.

4. A change of $p(y|q,f)$ does not have to a change the final decision $\hat{f}$. It may be that
   a. the loss function is not sensitive to change in $p(y|x,f)$,
   b. the risk, i.e a certain property of $p(l|f,\hat{f})$ like its expectation, is not sensitive to a change in the loss,
   c. the decison is not sensitive to a change in the risk.

   Of principal interest are the optimality equivalence classes
   $$[f^0]_{r^*} = [f'^0]_{r^*} \Leftrightarrow$$
   $$\hat{f}^* = \text{argmin}_{\hat{f}\in\hat{F}} r(f^0,\hat{f}) = \text{argmin}_{\hat{f}\in\hat{F}} r(f'^0,\hat{f}),$$
   identify f^0 leading to the same decision $\hat{f}^*$. While this requires already calculation of the optimal decision it may be easier to calculate
   $$[f^0]_r = [f'^0]_r \Leftrightarrow \forall \hat{f} \,:\, r(f^0,\hat{f}) = r(f'^0,\hat{f}).$$
   However for both variants, available data may have differing probabilities for different f^0 within such classes, so that $p(f^0|D)$ can vary within a class. Thus, analog to $f_{l,D}$ in the theorem, one has to consider finer equivalence classes
   $$[f^0]_{r^*,D} \quad \text{or} \quad [f^0]_{r,D},$$
   where only data D^f which factorize, like $p([f^0]_{r,D}) = p([f^0]_{r,D^{nf}})p([f^0]_{D^f})$, can be skipped. All those equivalence classes can be finer than $[f]_{r^*}$, so this does not guarantee that data changing $p(f^0)$, i.e. the state of knowledge f, change the decision $\hat{f}^*$.

5. Pure states are special factorial states. In this sense factorial states are the possible starting and ending points of a learning process. For x–variables which are visible and known, f^0 can be seen as collection of the remaining (stationary distributed) random variables. Making part of these random variables f^0 visible, that is moving them into x, breaks the old pure states in new pure components. The learning process would ideally go from the old pure state, now assumed to be incompletely specified and therefore only a factorial state of knowledge, to one of the possible new pure states being factorial and maximaly specified. We do not restrict pure states to be deterministic. Also probabilistic states can be not further decomposable in the situation under investigation.

2.2.3 Factor dimension

We now look for conditions under which learning can occur, i.e. when $p(y|q) \neq p(y|q,D_i)$ so not all mutual informations between available and relevant data are zero. We are mainly interested in the effects of generalization where q itself is not part of D, Indeed, if for continuous q^l the q^l which are directly in the training set have usually measure zero, so learning has to go over generalization.

Data not changing $p(f^0)$ have equal probabilities under all pure states f^0 or, equivalently, equal diagonal elements (and therefore eigenvalues) of the matrix $T(f'^0,f^0)$, projected to the subspace F_x^0, for all f_x^0. A unique maximal $f^{0,*}$, i.e. one with maximum probability for D_i and not excluded from f, would be sufficient to ensure learning for all f_x which are not yet in the state $f_x^{0,*}$. A change in $p(f^0)$ may however correspond to *nonrelevant learning*, changing not the relevant distributions $p(y|q,f)$ but only higher order interactions like $p(y,y'|q,q',f)$.

If the potential data D are sufficient and the model is minimal for the relevant questions, then according to the negated version of the theorem for nonfactorial priors there are training data D_i so that relevant data $(q,y) \in D^l$ change, Consider a locally minimal model with $x = q^l \in X$ local data $D^x = (x^D, y^D)$, $x^D \in X$ and a given prior $p(f^0|f)$. Local data q^D change the probability of y under x if $p(f^0) \neq p(f_x^0|D^x)$ and therefore $p(f^0) \neq p(f_x^0|f_D^0)$. Then within $p(f^0)$ states restricted to relevant and available data cannot factorize. Thus, the state of knowledge must fulfill the generalization condition for x under x^D
$$p(f^0|f) \neq$$
$$p(f_x^0|f_{X\setminus(x,x^D)}^0, f)p(f_D^0|f_{X\setminus(x,x^D)}^0, f)p(f_{X\setminus(x,x^D)}^0, f).$$
We can characterize a given state of knowledge by its *factor dimension* $\dim_F(f)$ with respect to a set X, defined as the maximal number n of $x_i \in \{x_1,\cdots,x_n\} = X^n \subseteq X$ so that still one f_x^0, with $x \in X^n$ the same for all f^0, can be factorized, conditioned on $f_{X\setminus X^n}^0$. Then learning has to occur under f if the number of local data

n is at least the factor dimension of f, i.e. $n \geq \dim_F(f)$, as then no additional x can factorize.

We give three examples:

Consider for a space F^0 parametrized by local means $\bar{y}_x$ for all x, defining a regression function $\bar{y}$ according to $\bar{y}(x) = \bar{y}_x$:

1. A nonlocal prior[20] $p(f^0 = \bar{y}) \propto e^{-c\sum_x(\bar{y}_x - c_x)^2}$. Even though this state explicitly depends on every single x it is factorial in X.
2. A symmetry (e.g. smoothness) prior $p(f^0 = \bar{y}) \propto e^{-c\sum_x(\bar{y}_x - \bar{y}_{sx})^2}$, s indicating some bijective symmetry transformation (e.g. translation) $x' = sx$. It has a degenerated maximum $\bar{y}_x = \bar{y}_{sx}$ which can be made unique by adding local data for every orbit of s. The orbit of x under s consists of all elements $s^i x$ which can be generated out of x by applying s any number of times, $0 \leq i \leq \infty$. The factor dimension for each orbit is equal to one. In case of several orbits the total factor dimension is only $n - m + 1$, with n the total number x, and m the size of the smallest orbit.
3. Data with $p(f^0 = \bar{y}) \propto e^{c(\sum_i^n \bar{y}_{x_i})^2}$ (parity for zero/one variables), which require additional data about $n-1$ different x_i to give a unique maximum. The sum can be seen as a local question in a space F^0 with a linearly transformed basis of f^0. The factor dimension is $n - 1$.

We make the side remark that the factor dimension has similarities to the concept of VC dimension. In both cases generalization is impossible if the number of data is smaller than the corresponding dimension. The latter is usually applied to a family of loss functions indexed by $\hat{f} \in \hat{F}$ in the context of empirical risk minimization. A small $\dim_{VC}$ indicates that the loss function cannot vary too much, so the difference between minimal empirical and minimal expected risk can be bounded. These bounds are independent of the family F^0, as long as certain minimal conditions, e.g. bounded $p(y|q, f^0)$, are fulfilled. In contrast, the factor dimension, as used here, is independent of the loss function and therefore of $\hat{F}$. The factor dimension has a similar interpretation for F^0 instead for $\hat{F}$: Nonconstant, independent data require a family F^0 at least 'large enough' to contain all the different combinations, while a smaller F^0 which excludes certain combinations implies the possibility of generalization. We already mentioned that not only natural choices of initial states but also the asymptotical final, i.e. pure, states of a learning process are factorial states, having maximal factor dimension, Hence, learning can as well decrease as also increase the factor dimension. Usually one would expect a U-shape like dependence of the factor dimension on the learning process, analogous to the mutual information, which provides a quantitative measure of dependency between relevant and available questions. Commonly prior information reduces the factor dimension and, at least deterministic, local data increase it.

Let $\dim_{VC}^{\alpha_{x,y}}(F^0)$ denote the ($\alpha_{x,y}$–dependent) VC dimension of a family of functions $p(y|x, f^0), f^0 \in F^0$, which is the maximal number of different pairs (x, y) so that for all of them exist $f^0, f'^0 \in F^0$ with $p(y|x, f^0) > \alpha_{x,y}$ and $p(y|x, f'^0) \leq \alpha_{x,y}$, $0 \leq \alpha_{x,y} < 1$, i.e. the maximum number of points (x, y) which can be shattered by F^0. Assume now that there exists an $\alpha_{x,y}$ with $p(y|x, D^x_>) > \alpha_{x,y} \geq p(y|x, D^x_<)$ for at least one pair of local data $D^x_> = (x, y_>)$ and $D^x_< = (x, y_<)$, Thus, $\alpha_{x,y}$ separates the posterior probabilities $p(y|x, D_i)$ of at least two possible data D_i, and the actual state cannot be a pure state. Then with $f^0 \in F^0$ the components of f with $p(f^0|f) \neq 0$

$$\dim_F(f) \leq \dim_{VC}^{\alpha_{x,y}}(F^0).$$

Indeed, if we assume $\dim_{VC}^{\alpha_{x,y}}(F^0) = n$ then according to the definition of the VC dimension within a set of $n + 1$ questions x there is at least one for which either $p(y|x, f^0) > \alpha_{x,y}$ or $p(y|x, f^0) \leq \alpha_{x,y}$ is impossible for all $f^0 \in F^0$. As convex combination the probability $p(y|x, f)$ for state f can only lie between extremal points, i.e. pure states f^0. Therefore, the assumption of existence of both $D^x_<$ and $D^x_>$, used to construct $\alpha_{x,y}$, implies that there exist f^0 with $p(y|x, f^0) > \alpha_{x,y}$ as well as f'^0 with $p(y|x, f'^0) \leq \alpha_{x,y}$ with nonzero probability $p(f^0|f) \neq 0 \neq p(f'^0|f)$. Thus, the probability $p(f^0_x, f^0_{x_1}, \cdots, f^0_{x_n}|f)$ cannot factorize, which gives $\dim_F(f) \leq \dim_{VC}^{\alpha_{x,y}}(F^0)$.

If the number of data is smaller than the factor dimension of f learning can still be relevant if Q^l contains multiple measurements (with respect to what has been considered an element in calculating the factor dimension). Multiple measurements of single components q^l means considering their interations also to be relevant. Using a measurement vector with one answer for every $x \in X$, i.e. $q = X$, makes for locally minimal models every learning for local data relevant. For example, knowing $y_1 - y_2 = 0$ does not change $p(y_1)$ or $p(y_2)$ if they have equal prior. However, it drastically changes the probability for a joint measurement of y_1 and y_2. This is an example where learning only occurs for nonrelevant higher order dependencies. If higher order dependencies are missing, so that for a state with factor dimension n already m relevant questions factorize, then only $n - m$ data are necessary to allow generalization to a single component q.

An example of taking into account dependencies between components q^l by multiple measurements, is the special case of the risk of an algorithm which uses past test data (y_i, q^l_i), $i < n$ to improve the selection of the next action for q^l_n. Here the next action and therefore also the loss function does depend on all previous test data (y_i, q^l_i). Then the risk does not consist of a sum of terms depending on disjunct sets of q^l_i, and correlations between components q^l_i up to order $n \geq i$ can be important. Thus, the effective q^l is the vector of components q^l_i, and the expected risk (on–line risk of an algorithm) is an average over this vector q^l. Correlations between

[20]We will discuss the measurement, i.e. the corresponding nonlocal questions in Section 3.2.

answers to different q^l (vectors, not components q^l_i) are not measured by this risk. However, the dependencies between vectors q^l should be smaller than between components q^l_i if $p(f^0|(y,q^l))$ is nearer to a pure state then $p(f^0|y_i,q^l_i)$.

Loosely speaking, we only need to know what we can see, we only can know what we have seen, and it is structural information which allows us to infer indirectly.

2.2.4 Generalization–related sets of questions

We now give the definition of some sets of questions, related to the previous analysis of generalization. We choose $Q = Q^l \cup Q^D$, $X = X^l \cup X^D$ and define the following sets of questions: Training questions q in Q^D but not in Q^l are called *nontest* questions $q^{\neg l} \in Q^D \setminus Q^l = Q^{\neg l}$, test questions corresponding to questions q in Q^l but not in Q^D are *nontraining* questions $q^{\neg D} \in Q^l \setminus Q^D = Q^{\neg D}$. Analogously, we write $x^{\neg D} \in X^l \setminus X^D = X^{\neg D}$ for the *nontraining basis* with $x \in X^l$ but $x \notin X^D$, as well as $x^{\neg l} \in X^D \setminus X^l = X^{\neg l}$ for the *nontest basis* with $x \in X^D$ but $x \notin X^l$. $Q^{l,D} = Q^l \cap Q^D$ denotes the set of *common questions* and $X^{l,D} = X^l \cap X^D$ the *common basis*. Clearly, we have $Q^{l,D} \cap Q^{\neg D} = \emptyset$, $Q^{l,D} \cap Q^{\neg l} = \emptyset$, $Q^l = Q^{l,D} \cup Q^{\neg D}$, $Q^D = Q^{l,D} \cup Q^{\neg l}$ and the corresponding relations for X. Notice that the common (or nontest, nontraining) basis is not necessarily the basis of the common (or non-test, non-training) questions.

Let us further introduce for sets $Q' \subseteq Q$, $X' \subseteq X$ the notation $Q'_{X'} \subseteq Q' \subseteq Q$ for the set of all questions within Q' depending only on X', i.e. with a basis completely within $X' \subseteq X$ so that $X^{Q'_{X'}} \subseteq X' \subseteq X$. Similarly, $Q_{(X')} \subseteq Q$ denotes the set of all questions depending (not necessarily only) on X', i.e. with a basis having nonzero intersection with $X' \subseteq X$ so that $X^{Q_{(X')}} \cap X' \neq \emptyset$. Obviously, $Q'_{X'} \subseteq Q'_{(X')} \subseteq Q'$, especially $Q = Q_{X^Q} = Q_{(X^Q)}$, and $Q'_{X'} \subseteq Q'_X$, $Q'_{(X')} \subseteq Q'_{(X)}$ for $X' \subseteq X$. We can partition a set $Q' \subseteq Q$ into two disjunct subsets with respect to $X' \subseteq X$ according to $Q' = Q'_{X'} \cup Q'_{(X \setminus X')}$. In particular, $Q^l = Q_{X^{\neg D}} \cup Q^l_{(X^D)}$ and $Q^D = Q_{X^{\neg l}} \cup Q^D_{(X^l)}$ with

$$Q^l \supseteq Q^l_{(X^D)} \supseteq Q^l_{X^D} \supseteq Q^{l,D} \subseteq Q^D_{X^l} \subseteq Q^D_{(X^l)} \subseteq Q^D,$$

and accordingly

$$X^l = X^{Q^l} \supseteq X^{Q^l_{(X^D)}} \supseteq X^{l,D} \supseteq X^{Q^l_{X^D}} \supseteq X^{Q^{l,D}}$$

$$X^{Q^{l,D}} \subseteq X^{Q^D_{X^l}} \subseteq X^{l,D} \subseteq X^{Q^D_{(X^l)}} \subseteq X^{Q^D} = X^D.$$

(See Fig.1 and for more details Fig.2.)

We have shown in the previous subsection that in a factorial state the non-training basis $X^{\neg D}$ is not learnable and the nontest basis $X^{\neg l}$ does not influence relevant q^l directly. However, $X^{\neg l}$ might well have indirect influence and act like noise sources within other questions Q^D depending on $X^{l,D}$. Thus, information about states corresponding to $X^{\neg l}$ enables learning about the noise structure within Q^D. Thus, with reference to factorial states, we will call the set of questions $Q^l_{X^{\neg D}} = Q_{X^{\neg D}} \subseteq Q^l$ depending only on $X^{\neg D}$ *unlearnable questions*, $Q^l_{(X^D)} = Q^l \setminus Q^l_{X^{\neg D}}$ (potentially) *learnable questions*, the set $Q^D_{X^{\neg l}} = Q_{X^{\neg l}} \subseteq Q^D$ *indirect questions* depending not on X^l, and $Q^D_{(X^l)} = Q^D \setminus Q^D_{X^{\neg l}}$ *direct questions*. Indirect questions alone cannot contribute to knowledge about Q^l, their influence is indirect by contributing information about the unknown noise sources for questions in $Q^D_{(X^l)}$. Note that $Q^{\neg D} \supseteq Q^l_{X^{\neg D}}$ and $Q^{\neg l} \supseteq Q^D_{X^{\neg l}}$ which means that nontraining questions can (and hopefully do) depend also on X^D (so indirect information about them may be available) and nontest questions also on X^l (so they can contribute directly). The nontraining basis $X^{\neg D}$ could be eliminated by integrating over a corresponding (factorial) prior. Then all relevant questions depend only on $X^{l,D}$, i.e. we have $Q^l_{(X^{l,D})} = Q^l_{X^{l,D}}$. Therefore, we will call $Q^l_{X^{l,D}} = Q^l_{X^D}$ the set of *effective questions*.

In the following we give some simple examples (compare subsection 2.2) that for $Q^l_{(X^D)}$ learning is possible, but not necessarily for individual $q^l \in Q^l_{(X^D)}$ if the dependencies between different q^l are not considered relevant. Consider deterministic x_1 and x_2 with possible values $y_i = \pm 1$ with independent prior probabilities equal $1/2$. Then knowing only $y_1 y_2 = \pm 1$ without knowing something about y_2 says nothing about y_1. Thus, the prior factorizes with respect to the equivalence classes $[f^0]_{x_1 \cup (x_1 x_2)}$ or $[f^0]_{x_2 \cup (x_1 x_2)}$, if we choose $x_1 x_2$ and x_1 (resp. x_2) as basis questions. If we choose all three as basis then $[f^0]_{x_1 \cup x_2 \cup (x_1 x_2)}$ cannot be written in a factorial form. Compare with the question $y_1 + y_2$ for which also no information about y_1 results from $y_1 + y_2 = 0$, however $y_1 + y_2 = 2$ determines y_1 and y_2. As soon as either y_1 or y_2 is known both nonlocal questions give the value of the missing y_i. If we do not choose $Q^l = \{x_1, x_2\}$ but instead e.g. $Q^l = \{|y_1 - y_2|\}$ the prior can be written in a factorial form with respect to $y(x) = |y_1 - y_2|$ and $y'(x') = y_1 + y_2$: $p(f^0_x)p(f^0_{x'})$ as those are in the given situation independent events.

2.3 Why structural information?

Structural information is our knowledge about the definition of questions corresponding to predetermined dependencies between answers, or in other words, our knowledge of what we are measuring. One might wonder whether such predetermined dependencies between answers are necessary and corresponding data can be available and useful.

First, we remark, as discussed in the previous section, that without nonlocal information generalization is not possible. Any nonlocal information has two parts, a structural one represented by the definition and the answer to be determined by empirical measurement or control. Thus, in other words, in any case there must be predefined dependencies between answers to some questions in order to be able to infer to unseen questions. So, as we will discuss in more detail below, requiring a bound on a certain smoothness or symmetry property one must for example know: 1) the definition of smoothness or symmetry in mind corresponding to the structural part of information or the predefined dependency of the smoothness question from the other questions, 2) a bound on the allowed function values obtained by empirical measurement, active enforcing or pure assumption.

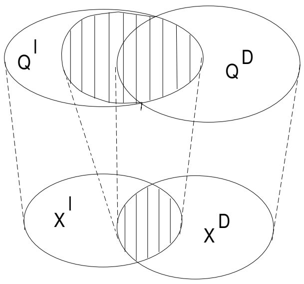

Figure 1: The figure shows the set of questions Q and their corresponding basis X. The shaded area within X represents the common basis $X^{l,D}$, the shaded area within Q the learnable questions $Q^l_{(X^D)}$. Learning related to questions within the shaded area but not within the data Q^D is called generalization. See Fig.2 for more details and notation.

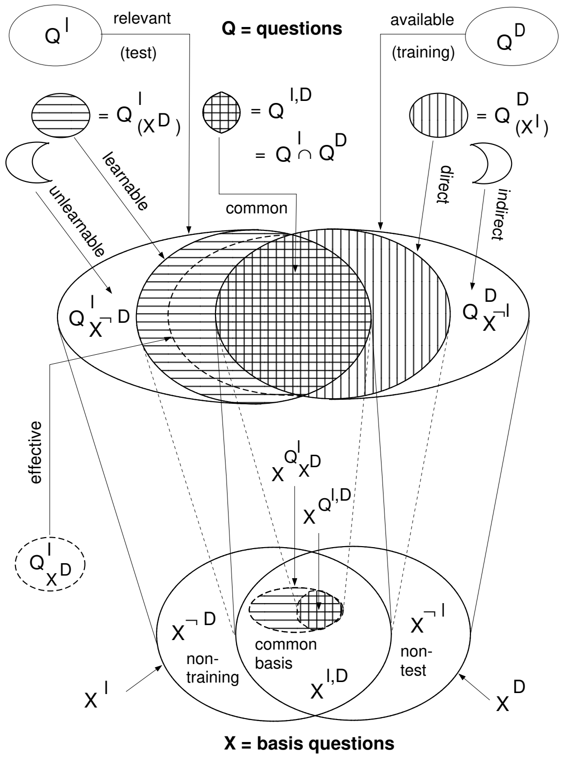

Figure 2: This figure shows the relations in more detail than Fig.1. The basis of relevant questions Q^l is X^l, and for questions Q^D corresponding to available data the basis is X^D. While Q^D is assumed to be always finite Q^l, X^l and X^D can be infinite. A model describing learning can be restricted to the common basis $X^{l,D} = X^l \cap X^D$ which is the double shaded area below. Only for questions depending on that intersection learning can occur. This set $Q^l_{X^D}$ is symbolized by the upper horizontally shaded area. Learning the set $Q^l_{X^D} \setminus Q^{l,D}$ is called generalization. Questions in $Q^l_{X^{\neg D}}$, being in Q^l but not in its shaded area, are unlearnable. Questions $Q^D_{(X^l)}$ can contribute information about Q^l directly. Questions in $Q^D_{X^{\neg l}}$, being in Q^D but not in its vertically shaded area $Q^D_{(X^l)}$, are not directly related to relevant questions and can only contribute indirectly, i.e. in combination with $Q^D_{(X^l)}$.

Secondly, we refer to the observation that it is very common to assume dependencies between answers. In logic these dependencies are called rules. With zero–one function values the logical operations can be written using multiplication, addition and the step function. Fuzzy set theory deals with unsharp rules and Bayesian belief networks are the general probabilistic formulation. Compared with logic based artifical intelligence expert systems which construct many rules with often low nonlocality (humans can better deal with lower order correlations) usual statistical models use only one, but highly nonlocal rule, e.g. some bound on a smoothness functional. However this difference is more of practical than of principal theoretical nature in as far as any number of rules, with maybe individually low nonlocality, can be combined into one ('vector') rule which can have high nonlocality. The definition a macroscopic observable in its dependence on microscopic variables is an example for structural information. One may say the usual macroscopic variables in form of integrals over all x have maximal nonlocality, depending on all basis questions. The definition of an 'energy question' could have the following form in a real, scalar, Euclidian field theory

$$p(y|q_E, f^0) = \delta\left(\int dx\,\left((\frac{d}{dx}f(x))^2 + m(x)f^2(x)\right)\right.$$
$$\left. + \int dx\,(h(x)f(x) + E_0(x)) - y\right),$$

where the first term has the form of a 'smoothness prior'. In terms of this analogy to physics, a statistical approximation problem with mean square error $\sum_i a_{x_i}(f(x_i)-y_i^D)^2$, data y_i^D and a smoothness prior corresponds to minimizing a 'free' massless kinetic term, a x_i–dependent mass $m(x) = a_x \sum_i \delta(x-x_i)$ and external field $h(x) = -2a_x \sum_i \delta(x-x_i)y_i^D$ coupling to $f(x)$ at the data points x_i and a constant $E_0 = a_x \sum_i \delta(x-x_i)(y_i^D)^2$. Higher order (local or nonlocal) interactions correspond to other forms of nonlocal questions. We will discuss in more detail below how answers to questions with infinite nonlocality can be enforced by measuring devices or by controlling the situations under investigation.

Finally, we point out that one always has to know what one measures, at least in a probabilistic sense, and therefore structural information is necessary in any case. When repeating measurement of basis questions x in a fixed state f^0 we assume that we use the same device with a stationary answer distribution. This means, for example, that if we measure in the same state hundred times $y = 5$ followed by a hundred $y = 42$ we might assume that our question x is incompletely specified and another (hidden) variable should have been included which probably has changed its value after hundred measurements. Such not directly observable or hidden variables define the state f^0 and we can say that changing the value of this additional variable represents a change in the unknown state f^0. However, we have to define the (not directly observable) state f^0 to be constant to allow us to infer something about future repeated measurements[21]. All controllable aspects of the

[21] If there is a known dependency on time (external time or

model are attributed to x, so controllable actions like replacing, moving, transforming an object to be measured are part of x. On the other hand variables which are themself stationary distributed just increase the noise and do not necessarily have to be included in x. Thus, stationarity of $p(y|x,f^0)$ is a form of structural information.

In the next sections we will discuss possibilities of the inclusion of a larger number and variety of generalized data in statistical decision making processes (including classification and approximation) which are based on structural information. Those data correspond to prior information in the statistical language, rules in the logical language, interactions in the physical language. From logical (fuzzy set, Bayesian belief network) expert systems one can learn how to deal with a lot of different, heterogeneous rules and from physics the treatment of highly nonlocal (macroscopic) variables different from smoothness.

3 Generalized questions

3.1 How to write them

A generalized questions is defined by giving a set of required basis questions and defining a function $y^q = q(\vec{y}(\vec{x}))$ to be applied to their results. To simplify notation we skip the vector arrow and use the same letter q for the defining function as well as for the functional itself, that is $q(y) = q_q(\vec{y})$. Generalized questions are fully defined by their answer distributions in the pure states $f^0 \in F^0$. As we always assume the model (of nature) to be in a possibly unkown but pure state the $p(y^q|q,f^0)$ represents the real processes and we write all formulas within this section for the f^0. This formulation without reference to some state of knowledge f fits also into the Frequentist interpretation of statistics meaning that the definitions of generalized questions do not require to specify an explicit model of f^0 and are therefore not only meaningful for Bayesians. This holds as far as we make the definition of a question not explicitly f–dependent. Thus, in the following we form products for pure states f^0 and not for states of knowledge f,

$$p(y,y'|q,q',f) = \int df^0 p(f^0|f)p(y,y'|q,q',f^0)$$
$$= \int df^0 p(f^0|f)p(y|q,f^0)p(y'|q',f^0)$$
$$\neq p(y|q,f)p(y'|q',f),$$

including the case $q = x$ and $q' = x'$. The fact that $p(y,y'|q,q',f)$ has not necessarily to factorize enables learning. But notice, that we always assume the different parts of a question to be measured in the same pure state f^0. For f^0 we postulated independence, i.e. factorization

$$p(y,y'|q,q',f^0) = p(y|q,f^0)p(y'|q',f^0),$$

meaning that different questions q, q' use different realizations of answers to x and that learning being in a pure state, i.e. $f = f^0$, is not possible. In this formulation results depending on the same realization are seen as components of one question. On the other hand the following is easily adapted for a notation which allows such dependency between non basis questions.

A simple example is a question asking for the sum of two independent x_i–measurements with answer probability

$$p(y^q|q,f^0) = p(\,\{y_1+y_2=y\}\,|x_1,x_2,f^0)$$
$$= \int dy_1 \int dy_2\, \delta(y_1+y_2-y^q)p(y_1|x_1,f^0)p(y_2|x_2,f^0),$$

where $\delta(y_1+y_2-y^q)$ is the indicator function of the event $y_1+y_2 = y^q$ and as always in this paper the δ–functional has to be understood as Kronecker–δ function δ_{y_1,y_2} in the discrete case.

Deterministic questions

The last example can easily be generalized by replacing y_1+y_2 by any other *defining function* $q(y)$ depending on a data vector y with components y_i [22]

$$p(y^q|q,f^0) = p(\{q(y) = y^q\}|\,x^q,f^0)$$
$$= \int \left(\prod_i dy_i p(y_i|x_i^q,f^0)\right) \delta(q(y_i)-y^q).$$

To simplify notation, we now use the symbol x^q not for a single basis question but for a whole vector of them. Its components x_i^q, correspond to the y_i necessary to evaluate q and can include repeated measurements of the same basis question. Specifically, averages of actual measurements correspond to choosing $q(y) = 1/N\sum_i y_i$. The answer y^q to question q can be a vector with components y_j^q meaning that

$$\delta(q(y)-y^q) = \prod_j \delta(q_j(y)-y_j^q).$$

For noisy x this gives not the product

$$p(y^q|q,f) \neq \prod_j p(y_j^q|q,f),$$

because this would refer to a situation where the y_j^q do not use the same realization but sample their own y_i resulting in additional y–integrals.

While in the following no special attention is paid to the technical difficulties of the case of continuous y we shortly discuss the definition of the δ for this case where

internal time corresponding to the measurement history like the number of repetitions of a certain measurement) then the time variable is part of x. A measurement at a certain time can be repeated if the time variable can be reset (e.g. to zero). If time cannot be reset repeating the same measurement is not possible. One can think of other restrictions, so that a measurement of the same question cannot be repeated. For example, if continous questions x are generated by a random process, the probability of repeating a measurement usually has measure zero. Those cases show the importance of information about relations between answers to different questions and not only between repeated identical questions, i.e. of nonlocal dependencies.

[22] The components y_i of the data vector as well as the answers y^q can be vectors itself.

δ stands not for the Kronecker function but for the δ–functional. This is defined for general $q(y)$ for $\left.\frac{dq(y)}{dy}\right|_{y_0} \neq 0$ according to

$$\int dy p(y|x,f)\delta(q(y)-y^q) = \sum_{y_0} p(y_0|x,f)\left|\left.\frac{dq(y)}{dy}\right|_{y_0}\right|^{-1},$$

where y_0 are the solutions of $q(y) = y^q$, i.e. the zeros of the argument $q(y) - y^q$. The nonzero first derivative guaranties that $q(y)$ is locally invertible so we can write $y_0 = q^{-1}(y^q)$ with q^{-1} defined at least in a neighborhood of y^q. In case $q(y) = y^q$ is fulfilled on a whole intervall the δ–functional has to be replaced by the characteristic function of that event which can be expressed by step functions Θ.

Output noise

A question can also be constructed including additional internal random variables z by using a *probabilistic defining function* $q(y,z)$ defined by f^0–independent output distributions $p(z|y,q) = p(z|y,q,f^0)$. The answer can be seen as created in a two step process

$$p(y^q|q,f^0) = \int dy\, p(y^q|y,q)p(y|x^q,f^0),$$

with

$$p(y^q|y,q) = \int dz\, p(z|y,q)\,\delta(q(y,z)-y^q),$$

and

$$p(y|x_q,f^0) = \prod_i p(y_i|x_i,f^0)$$

The vector of internal random variables z increase the noise in answers while additional (repeated) measurements of the same basis questions reduce the noise. Note that the noise variables of different questions are independent, that is we assume $p(z_q,z_{q'}) = p(z_q)p(z_{q'})$ for $q \neq q'$, and questions always measure their own y–values according to $p(y|q,f^0)$ and do not refer to measurements of other questions. One specific realization of y can be used multiple times only within the same question.

Input noise

The vector x of basis questions contained in q and its dimension can be an f^0–independent random variable. We call this a situation with *input noise*. Then $q(x,y)$ depends on x generated according to $p(x|q)$. The index q indicates that this distribution is part of the definition of the functional q. Thus, we have

$$p(y|q,f^0) = \int dx\, p(x|q) \prod_i^{\dim(x)} p(y_i|x_i,f^0),$$

with $\int dx = \sum_j \int \prod_i^j dx_i$ and $p(x) = p(\dim = j)\; p(x|j)$. We understand all dependencies within the vector x to be included in this notation. Products of $p(y_i|x_i,f^0)$ correspond to a logical AND applied to the results y_i, summations over x_i and z correspond to a logical OR. Input noise can be combined with a probabilistic q–function $q(x,y,z)$

$$p(y^q|q,f^0) = \int dy\, p(y|q,f^0)p(y^q|y,q)$$

$$= \int dx\, p(x|q) \int \left(\prod_i^{\dim(x)} dy_i p(y_i|x_i,f^0)\right) p(y^q|x,y), \tag{1}$$

where $p(y^q|x,y) = \int dz\, p(z|x,y,q)\,\delta(q(x,y,z)-y^q)$ and $\int dy = \sum_j \int \prod_i^j dy_i$. Finally, we remark that input noise has both more active and more passive interpretations. Changing the variable x can be interpreted as changing the device or changing the situation. For example, measuring another location x of an object can be done by moving the measuring device to location x (passive interpretation) or by moving the object (active interpretation) or by moving both (mixed interpretation). The passive interpretation may be the usual one, but as we assume f^0 to be constant over the whole time of interest all active controllable aspects must be part of x.

Causal chains

We can also allow *output dependent input noise* by choosing the x_i depending also from results of previous measurements y_j included in q. For example, active learning algorithms which select the next question according to the data in the past belong to this class of questions. We order the factors according to the causal dependencies of the generating process like

$$p(x,y|q,f^0) = p(x_1|q)p(y_1|x_1,f^0)$$

$$\times\, p(x_2|x_1,y_1,q)p(y_2|x_2,f^0)$$

$$\times\, p(x_3|x_2,y_2,x_1,y_1,q)p(y_3|x_3,f^0)\cdots$$

$$= p(x|y_c,q)\prod_i p(y_i|x_i,f^0).$$

where the subscript c indicates a 'causal' ordering, meaning $p(x|y_c,q) = p(x_1|q)\prod_{i=2} p(x_i|\{x_j,y_j\}_{1\le j\le i-1},q)$, with (the 'last') one of the components of vector y missing. Note that we understand the causal structure for components of x to be implicit in the notation $p(x|y_c,q)$. Chains with variables depending only on those of the previous step $p(x|y_c,q) = p(x_1|q)\prod_{i=2} p(x_i|x_{i-1},y_{i-1},q)$, are sometimes called Markov chains.[23] For finite sequences such a representation can always be achieved by combining x, y of different steps into one vector variable, or in general by including the relevant memory variables. In the extreme case this would lead back to the starting point $p(x|q)p(y|x,q)$. Including internal noise variables z their probability $p(z|x,y)$ may also be written in a causal realization modeling the real causal processes

$$p(x,y,z|q,f^0) = p(x_1|q)p(y_1|x_1,f^0)p(z_1|x_1,y_1,q)$$

$$\times p(x_2|x_1,y_1,z_1,q)p(y_2|x_2,f^0)p(z_2|x_1,y_1,z_1,x_2,y_2,q)\cdots$$

$$= p(x|y_c,z_c,q)\prod_i p(y_i|x_i,f^0)p(z|x,y,q).$$

[23] See for example Golden, 1986. However the term Markov chain is used in many variations. See for example van Kampen, 1992, p.77 and its footnote on p.89 refering to a footnote on p.340 in Feller, 1957.

It is always possible to write a joint probability in this form and we always understand the indices of the variables x, y, z to refer to the same ordering. This gives

$$p(y^q|q,f^0) = \int dx \int dy \int dz \left(\prod_i p(y_i|x_i,f^0)\right)$$
$$\times\, p(x|y_c,z_c,q)p(z|x,y,q)\,\delta(q(x,y,z)-y^q)$$
$$= \int dx \int dy \int dz\, p(y|x,f^0)$$
$$\times\; p(x,|y_c,z_c,q)p(z|x,y,q)p(y^q|x,y,z,q), \qquad (2)$$

showing the separation into q–dependent factors written in index form from and f^0–dependent factors. Special realizations of causal dependencies, including variables z, defining the answer producing process can be modeled by directed acyclic graphs, i.e. graphical models or belief networks (Pearl, 1988; Lauritzen, 1996, Jensen, 1996, Ripley, 1996). A pair of variables $R_q = (x,z)$ can be called realization of q because it represents that part of the definition of a question determining a specific answer y^q and $D_q = (x,y)$ the corresponding data. We denote the set of all questions of the form of eq.(2) including only finite dimensional vectors x, y, z by $\bar{Q}_X^{fin}$.

Combinations, decompositions

Having defined a set Q' of questions q' we obviously can generate new questions q according to eq.(2) by replacing $p(y|x,f^0)$ with $p(y|q',f^0)$ and choosing any input functions $p(q'|y_c,z_c,q)$, internal noise functions $p(z|x,y,q)$, and defining functions $q(x,y,z)$. To avoid circular definitions the q must of course not contain itself. If the new q is expressed in terms of basis questions, i.e. the $p(y|x,f^0)$, it still has the form of eq.(2), meaning it belongs to $\bar{Q}_X^{fin}$. Consequently, for arbitrary $q' \in \bar{Q}_X^{fin}$ a q of the following *general q-form*

$$p(y|q,f^0) = \int dq' \int dy \int dz\, p(q'|y_c,z_c,q)$$
$$\times\, p(y|q',f^0)p(z|x,y,q)\delta(q'(x,y,z)-y) \qquad (3)$$

is also in $\bar{Q}_X^{fin}$. In this sense $\bar{Q}_X^{fin}$ is closed. Also for every $q \in \bar{Q}_X^{fin}$ there are $q' \in \bar{Q}_X^{fin}$ so that q can linearly decomposed in that form. Specifically, the q' can always be chosen as x.

In general a decomposition of q in arbitrary components q' can be written as

$$p(y|q,f^0) = \int dq'\, p(q'|q,f^0)p(y|q',q,f^0),$$

with f^0–dependent $p(q'|q,f^0)$ if q' is sampled depending on y. We can get a y– and therefore f^0–independent decomposition of q into lower noise components q^0 in analogy to x–independent decomposition of states f into pure states f^0. To get this, we separate $R_q = (x,z)$ into a y–independent part q^0 and a y–dependent part. Note that at least one x_i is in q^0 and arbitrary dependencies are allowed within q^0. Then we have an f^0–independent decomposition of q

$$p(y|q,f^0) = \int dq^0\, p(q^0|q)\, p(y|q^0,q,f^0), \qquad (4)$$

in analogy to a x–independent decomposition[24] of f

$$p(y|x,f) = \int df^0\, p(f^0|f)\, p(y|x,f^0),$$

but with, in general, differing 'local states' $p(y^q|q^0,q,f^0)$ for different q. Despite that analogy we do not assume the availability of data which can change the $p(q^0|q)$. This is without loss of generality as all data dependencies of question definitions can be incorporated by enlarging the space F^0 and adding questions with answers depending on $p(q^0|q)$. In fact, in practice an example generating distribution $p(x)$ is often unknown and estimated using the sampling distribution. Formally we can combine x and y into a new $y' = (x,y)$ and define functions $f'^0 = (f_x^0, f_y^0)$ by $p(y'|x',f'^0) = p(y|x,x',f_y^0)p(x|x',f_x^0)$. The new x'–variable can be skipped from the notation having only one value with the minimal meaning of requesting another $y' = (x,y)$ value. A state of knowledge f' contains now also information about $p(x)$ and if $p(f'^0) = p(f_x^0)p(f_y^0)$ we can estimate $p(x)$ independent of $p(y|x)$ by observing pairs (x,y). But note that for continuous X if not parametrized by a finite set of parameters the f_x^0–integration is a functional integration and not necessarily well defined.

If studying f–dependency we may also allow f–dependent definitions of q as the state of knowledge is assumed to be known and therefore this knowledge could be incorporated into the answer producing process. The preceeding equations include this case if q is interpreted as double index $q \to (q,f) = q_f$. Notice that as answers change the state of knowledge f and they also change the definition of q.

General functionals

We allowed the answer probability $p(y^q|q,f^0)$ at y^q of a generalized question to be any functional of the y–distributions $p(y|x,f^0)$ (for all x but fixed f^0). A parametrization of, or in the extreme case the $p(y|x,f^0)$ for all x and y itself, describe a state f^0 completely. Functionals depending on the $p(y|x,f^0)$ include for example

$$p(y^q|q,f^0) = \delta(y^q - \int dy\, yp(y|x,f^0)),$$

giving the local expectation, or in a noisy version

$$p(y^q|q,f^0) \propto e^{-\frac{1}{2}((y^q - \int dy\, yp(y|x,f^0))/\sigma)^2},$$

as well as

$$p(y^q|q,f^0) = \delta(y^q - p(y|x,f^0)),$$

giving a specific probability (density) or general

$$p(y^q|q,f^0) = \int dyp(y|f^0)\delta(y^q - q(z)).$$

We recognize that the questions we have constructed so far in the previous sections contain products of $p(y|x,f^0)$

[24] For simplicity we use from now on the integral notation also for the f^0–variable assuming a well defined (possibly discrete) measure.

(together with a sum implicit in the notation of the last sections) and are therefore functionals with a finite power series

$$p(y|q,f) = \sum_{n=0} \left(\prod_{i=1}^{n} \int dx_i dy_i p(y_i|x_i,f) \right) a_n(x,y).$$

Those generalized questions can be measured using a finite number of answers to basis questions, while questions with an infinite but converging power expansion might be measured approximately. Expectations and probabilities of events can sometimes be approximated using empirical sums but in general not a probability density $p(y|x,f^0)$ for continuous y. We will use the word measurement for the process of getting an answer to a question, but we will show in the next section that for questions with no finite or converging power series this has more the character of enforcing an answer or active control of F^0.

3.2 How to measure them

An answer to a question could be obtained using only measurements of basis questions x when it has a defining function $q(x,y,z)$ depending only on a finite number of outcomes $y_i(x_i)$.

3.2.1 Finite case

We denote the set of questions with finite dimensional y, z by $\bar{Q}_X^{fin}$. With Q denoting the measurable questions we have therefore $X \subseteq Q \Rightarrow \bar{Q}_X^{fin} \subseteq Q$. These questions could (but do not have to) be measured by a finite number of possibly repeated measurements of basis questions $x \in X$ according the following steps:

i. Choose x according to $p(x|q)$,

ii. Get y for the basis questions in x,

iii. Get z according to $p(z|x,y)$, (Repeat i., ii. and iii. in case of $p(x|y_c,z_c)$ dependencies)

iv. Insert result into $q(x,y,z)$.

While the measurements can be performed using only devices measuring basis questions, these questions do not necessarily need the full information contained in the answers to x. That means measuring devices designed specifically for them could be more effective. For example, using interference of waves differences can be measured in physics sometimes much more precisely than absolut values.

We discuss how in the finite case a direct measuring of q might be preferable, a situation which often occurs in practice. Consider output noise which is added in a final step for all measurements during training, like observation and memory errors, or transformation to a lower resolution final output scale. Then measuring devices which are able to directly access the underlying lower noise function and perform the necessary operations before adding the output noise have higher accuracy. As simple example, take a basis set X with output distributions

$$p(y|x,f^0) = N(\mu_x(f^0),\sigma)$$

and

$$\mu_x(f^0) = \int dy\, y\, p(y|x,f^0),$$

where $N(\mu,\sigma)$ stands for a Gaussian centered at μ with variance σ^2. Then, if measurable, the generalized question

$$p(y|q,f^0) = N(\mu_q(f^0),\sigma),\ \mu_q = \mu_{x_1}(f^0) + \mu_{x_2}(f^0),$$

obtaines the sum with greater accuracy than using the sum $y_1 + y_2$ of two basis questions x_1 and x_2 where the independency of the noise gives $2\sigma^2$ for the variance. In this special case the question would be equivalent to four basis questions including one repeated measurement for every x_i to get $(y_1 + y_1' + y_2 + y_2')/2$. Replacing the sum $\mu_{x_1} + \mu_{x_2}$ by an integral $\int \mu_x dx$ (infinite nonlocality) or setting σ in q equal to zero with retaining a finite σ for the x_i (infinite accuracy) the question depends on more than a finite number of answers to basis questions.

3.2.2 Infinite cases

Asymptotic procedures

In some cases a defining function can be found so that a well defined limit with the number of arguments going to infinity gives the desired $p(y^q|q,f^0)$. For example, there are measurements depending on a formally infinite number of basis questions using some inherent, infinite parallelism of natural processes or a process having a well defined limit for the number of involved basis questions going to infinity. Scattering of waves on structures with specific translation and rotation invariances (crystals) create filters for specific Fourier components which depend on a (conceptually) infinite number of coordinate values. Measurement of macroscopic variables in physics, like energy or magnetization depends on many (normally in the order of 10^{23}) microscopic variables. If the answers to those questions converge in the limit system size $n \to \infty$, they can be considered as measurement of an infinite system. In general, a meaningful statement about an infinite property, has ro rely on conditions which can be checked in finite times. In the following we relate such conditions to the preparation of an ensemble and possibilities of control.

Measurement and preparation

We begin with the observation, that questions available during the training situation might be different from those during the preparation of the ensemble f.

Consider objects f^0 which, giving a parameter $\bar{y}$, produce as random output an output distribution with mean $\bar{y}$. We can form a population F^0 of such objects, notate, i.e. measure, the input parameter $\bar{y}$ for each member, and select, with a given (prior) probability $p(f^0 = p(\hat{y})$, one object f^0 with unknown $\bar{y}$. Then, we cannot measure the mean exactly using only training example y_i of the output of f^0. In cases the mean exists, the sampling sum will, according to the theorem of Glivenko–Cantelli, usually asymtotically converge to it, (i.e. the probability δ for the deviation to be larger than some ϵ can be n–dependently bounded). Thus, what have been measurable during the preparation phase by a single measurement (i.e. reading the number $\bar{y}$) is, if at all, only asymptotically measurable during training.

In a general situation we consider a family of parametrized answer distributions F^0 with known parameters like expectation and variance. If we can measure the values of these parameters we can prepare a prior distribution (state f) depending on those parameters. For example, if a certain symmetry property is measurable we can choose an f depending on this property, for example with possible states f^0 possessing (or concentrated around) a certain value of this symmetry. In a specific training situation, however, the symmetry may not be directly measurable, and an additional source of noise be present.

We will denote the set of questions with available answers during preparation by X^P and will call it the *preparation set*. The preparation set X^P is in general not equal to the set of training question X considered actually available. Questions not finite or asymptotically measurable with respect to X can be finite or asymptoticaly measurable with respect to X^P.

As far as generalization requires nonlocal questions not measurable with respect to X, like a symmetry or smoothness constraint for infinite X, there has to be another set X^P to allow the necessary measurements to prepare the prior. Then the prior can be expressed by a question q^P depending on a finite number of questions $x^P \in X^P$ and the corresponding answer y^q

$$p(f^0|y^P, q^P) \propto p(f^0)p(y^P|q^P, f^0).$$

However it could not be written using a question depending on a finite dimensional vector $x \in X$ of training questions. The training data change the prior according to

$$p(f^0|y^D, q^D, y^P, q^P) \propto p(f^0|y^P q^P)p(y^D|q^D, f^0).$$

Preparation questions x^P with deterministic answers, like an exact symmetry, correspond to restrictions of F^0. Probabilistic q^P give probabilistic priors, like a preference for smooth functions or otherwise symmetric functions (approximate symmetries).

The distinction between a (momentarily considered) training set X and a preparation set X^P (on may say implicit training set) is more a practical than a formal aspect, For analysis of the generalization ability questions from X^P have to be treated formally equal to those from X. Thus, effectively one deals with training questions depending on $X \cup X^P$.

Specifically, we will explain below how symmetry and smoothness can be generated by input noise or averaging. To recognize or prepare such a situation a measurement device without (or less) input noise (X^P) must be available.

We summarize, that what appears to be not measurable with a finite amount of data for X may well be measurable within the larger set $X \cup X^P$. This is the case, when the set of actually considered training questions X all share a common noise scource, which is absent for X^P.

Measurement and control

To enable learning we have to assume stationarity of $p(y|q, f^0)$. Thus, all factors changing the answer distributions have to be included in q and f^0. But the model does not specify how stationarity is achieved. In practice, stationarity can result from an active control or just by not disturbing a constant part of nature.

In general, a measurement y of q in state f^0 is the result of an interaction of the 'active' part of posing a question q and the 'passive' reaction of nature in state f^0. Usually, measuring a quantity q emphasizes the passive picture where the value of y reflects a permanent property of nature f^0, however, only seen when measuring q. For preparation questions the complementary active interpretation of measuring as control or selection may also be helpful. Then, the answer is seen as a property of nature enforced by the question q. In this interpretation a question is better called a control action and a measurement device a control device. Thus, y can be seen as reaction of nature to q. In the general case of stochastic control different states f^0 react different to the control action q, described by $p(y|q, f^0)$. This point of view is especially suitable if the variablility of y between different f^0 is small under q. Both interpretations describe the same formalism, the difference being the larger emphasis of either the passive or the active part.

Interpretations may also refer to a more complicated model of interaction between q and f^0. Such more complex models correspond formally to the introducion of hidden variables z,

$$p(y|q, f^0) = \sum_z p(y|q, f^0, z)p(z|q, f^0).$$

That means we think of measurement devices as generalized questions with respect to some underlying set of measurable questions. Then, the x can be called effective questions x^{eff} with respect to another, underlying X. If we do not want just to assume such a structure, we need other measurement devices, we may call them in analogy to the last Subsection $q^P \in \bar{Q}_{X^P}$, which allow to measure or control a given structure. We will see that this can be reasonable to assume and those additional measurement devices will have to be active only a finite number of times. If we can guarantee the structure for all q, this can be equivalent to dependencies between q und therefore interpreted as nonlocal measurement.

For example one can imagine the value y to be the result of an (ideal) measuring process with a following control device, e.g. a filter describing restrictions of the (real) measurement device. For this we separate the index q into two components $q = (q^{ideal}, q^{filter})$ (or alternatively $f^0 = (f^{0,data}, f^{0,filter})$) and have

$$p(y|q, f^0) = \sum_{y^{ideal}} p(y|q^{filter}, y^{ideal})p(y^{ideal}|q^{ideal}, f^0).$$

We will call this a model of *posterior control.*

Analogously, we can think of a scenario where first the functions are filtered and then measured, i.e. we split q into $q = (q^{data}, q^{filter})$ (or alternatively $f^0 = (f^{0,data}, f^{0,filter})$) and have

$$p(y|q, f^0)$$
$$= \sum_{f^0_{filtered}} p(y|q^{data}, f^0_{filtered})p(f^0_{filtered}|q^{filter}, f^0).$$

If we f^0 from the notation, write again f^0 for $f^0_{filtered}$, the filter $q^{filter} = (y^P, q^P)$ creates a prior ensemble.

Here the q independence allows us to have it prepared before the training starts. That is what we will call a model of *prior control.* The situation we discussed in the last Subsection can therefore be realized as prior control. Notice, that also under prior control stationarity of the probability distributions must be controlled, and in this sense there is always a posterior control component present. Adjusting the measurement device and repeat the mesurement if some 'failure' is indicated is an example of such an posterior control, which is also present, if the prior ensemble is prepared in the past.

Now we turn again to the problem of questions depending on an infinite number of basis questions. In the last subsection we used a decomposition $q' = (q, q^P)$ with $q \in \bar{Q}_X^{fin}$, $q^P \in \bar{Q}_{X^P}^{fin}$ and showed that this allows $q^P \notin \bar{Q}_X^{fin}$. Now we examine how also in the example of posterior control measurement of questions depending on an infinite number of basis questions is possible.

The key observation is that the stationarity condition for $p(y|q, f^0)$ only affects an always finite number of measurements. Hence. also (e.g. prior or posterior) control can be restricted to this finite number of measurements. This allows to understand or implement stationarity as part of the measuring process, e.g. as a filter. For example, nothing prevents us to assume that asking q the first time causes stationarity for the following times. Therefore, posing a question q can have the interpretation of an active control leading to some restrictions or dependencies for subsequent measurements, stationary for all f^0. Differentiation between different f^0 is only possible if the controller, but not necessarily the learner, has questions, $q \in \bar{Q}_{X^P}^{fin}$, available to distuinguish between them.

As example, take a family of q measurement devices, all capable only of producing answers smaller than θ, independent of f^0. This can be ensured by one q–independent cut–off device as posterior control. This cut–off device depends itself only on finite dimensional incoming y^{ideal} and has to be active only during the finite number of measurements. Nevertheless, the posterior control model is equivalent to a nonlocal measurement for *all* q, which is possibly an infinite number, with answer $q < \theta$. Here, the control is implemented by f^0–independent properties of the measurement device. The same effect appears if all objects under study underly the same f^0–independent selection. So instead of using restricted measurement devices as posterior control, we could also study restricted objects or situations, i.e. a implementation as prior control. Then the controling filter acts using $q^P \in \bar{Q}_{X^P}$ not on measurement values but on the objects or situations f^0 itself. In analogy to the output bound θ above, the length of object classes under study might be restricted by θ because they all arrive in the same box, must fit into a certain environment or are only produced in that way. Those are prior control devices parametrized by θ. Also, if the number of objects is infinite, the prior control device (e.g. the box into which the object must fit) must only be active during the finite number of measurements.[25]

Thus, because control of stationarity, be it related to training or preparation questions, only has to be active at the always finite number of times of actual measurements, there is no practical impossibility in measurements based upon control depending on an infinite number of basis questions. Nothing is actually done an infinite number of times.[26] It just could be done arbitrarily often because we defined the situation to be so. A filter bounding outcome values only has to be active a finite number of times for every actual measurement even if we defined the situation that it could do so arbitrarily often. This includes also the stationarity conditions for local questions.

We have seen that what exactly choosing question q means in practice and how stationarity is achieved, be it by leaving nature alone or by active control does not enter the formalism. Dependency on an infinite number of questions is no practical impossibility in as far as control over stationarity conditions only has to be active at the always finite number of times of actual measurements.

Another aspect of control will be discussed in more detail in the next Sections: Often the presence of control is easy to recognize, however difficult to formalize. For example, creating a training set for a object recognition task by choosing images as training examples of faces and nonfaces or chairs and nonchairs, depends on one's implicit definition of the concept of a face or chair, respectively. Drawings may be accepted as valid examples of the object class or not, while very unregular, random–like objects are not selected to represent a face or a chair. Consequently, the definition of such object classes is related to linguistic or implicit concepts representing the objects. Such defining concepts, involved for example in a prior or posterior control process, can correspond to a measurement of an infinite number of objects. Thus, it can be expected to be helpful having a method to formalize those concepts and include them as prior information into the statistical inference process. Accordingly, useful

[25]There can be practical differences between such devices, for example between a 'box' device, and a cut–off in the output scale. Indeed, a filter can be implemented as active or as passive filter. A passive filter does not always return an output. For values larger θ for example it may answer 'overflow'. For example, using the box device it may take considerable time to find an object which fits into it. There might even be no objects available which fit in the box, and we could need a formally infinite time to produce such an impossible state. An active filter always returns an output. For example we may assume for a cut–off device that it always returns θ in case the output is larger θ. However, this has nothing to do with the difference between prior and posterior control. Both variants are also possible for prior control. Consider, for example, an 'active box' cutting everything to a fitting size Those aspects of differing complexity of single measurements or control actions are not included in the formalism. Yet, nothing prevents us from modeling the microstructure of single measurements if necessary; we could use the same type of theory, just on another level.

[26]This is related to a constructive view of infinity, not attributing 'existence' to an abstract infinite object itself, but to its constructing procedure. This is, for example, the position of Jaynes, 1996, who 'sails under the banner of Gauss, Kronecker, and Poincaré rather than Cantor, Hilbert, and Bourbaki.'

restrictions might be found from an analysis of the application situations or of the object or situation generating process: a pedestrian detection task might be restricted to pedestrians walking on or near the street in a certain distance and a car (ship, airplane, $\cdots$) detection task may take into account that only certain types of them have been produced up to now.

As men are always more or less involved in the definition of situations or problems of interest this shows the clear need of a *human interface* which enables human knowledge, like linguistic concepts, used to define and control the application situations to be incorporated into the learning process.

Summarizing, we list two variants in which measurement by control depending on a possibly infinite number of questions can appear:

1. prior control: controling the process which generates the prior distribution, e.g. control of the situations or objects of interest,
2. posterior control: controlling the measurement values, e.g. restrictions of the measurement devices.

Stationarity is essential for all questions and the interpretation of measurement as control or active enforcing of stationarity can be applied to all questions, not only to those which cannot be measured using a finite number of basis questions or some asymptotic procedure. One can say that the ability to generalize is based upon the ability to control that the situation of interest and the measuring devices are kept within their restrictions. We have discussed, that the probability concept does in general not allow to exactly verify (even local) nondeterministic conditions using only the training data and no preparation questions. Approaches to test the conditions, i.e. the models f, using training data have to refer to meta models or use the classical testing of null hypotheses, i.e. calculate the probability of the data given the condition f. Applied to model testing this has also been called evidence approach (MacKay, 1992c).

We conclude, that the ability to generalize is based on the ability to measure or, emphasizing more the active connotation and formally equivalent, to control dependencies between basis questions. We have shown that this is also practically possible in case of an infinite X. From this point of view only measurement can generate new information. Actual learning, however, is not the discovery of something new, but the reformulation of actually imposed and measured conditions to give answers to relevant questions. Analogously, assumed learning is the reformulation of assumed conditions.

4 Priors

We discuss two examples of generalized questions often used as priors and give then in the next Section a general method to construct priors.

4.1 Bounds

In practice measurement values have a bounded range and even models using distributions with unbounded range for the variables use normally a bounded range for moments, like the mean. To discuss such bounds we present some variations of questions with q–functions calculating the maxima of somewhat different sets:

1. empirical maximum from a finite sample
$$m_1 = \max_i y_i(x_i, f^0),$$
2. maximum of the local expectation (regression function)
$$m_2 = \max_x \int dy\, y\, p(y|x, f^0) = \max_x \bar{y}_x(f^0),$$
allowing observed values $y_i > m_2$ generated by noise,
3. outcome (answer) being maximal with probability one
$$m_3 = \max_x y^*(x),$$
defining as local maximum $y^*(x)$ at x the minimal y with
$$\forall y' \in Y_x \,:\, p(y' > y|x, f^0) = 0,$$
or
4. maximal potential outcome
$$m_4 = \max_x \max_{y \in Y_x} y$$
being independent of f^0 if the Y_x are defined independent of f^0.

Questions related to m_1 belong to $\bar{Q}_X^{fin}$ and can be empirically measured if X can. In approximation problems where one is interested in modeling the regression function $\bar{y}_x(f^0)$ and not f^0 itself one normally refers to m_2 and allows for example Gaussian noise still to generate arbitray large outcomes even for finite m_2. The answer to m_2 can be bounded only with access to the regression function, which in this case is interpreted as the true underlying function. This 'bounding device' has therefore to be applied before the measurement noise. The bound m_3 is the most interesting one in worst case considerations and is itself bounded by m_4. Fixing (the answer to question) m_4 is done by using cut–off devices. A question q_A^{cut} corresponding to a cut-off device applied to (real) answers to question q can be written

$$p(y^{cut}|q_A^{cut}, f^0) = \int dy\, p(y > A|q, f^0)\delta(A - y^{cut})$$
$$+p(y \le A|q, f^0)\delta(y - y^{cut})$$
$$= \int dy\, p(y|q, f^0)[\delta(\Theta(y - A) - 1)\delta(A - y^{cut})$$
$$+\delta(\Theta(A - y) - 1)\delta(y - y^{cut})].$$

The step function $\Theta(x)$ is defined to give 1 for $x \ge 0$ and 0 for $x < 0$. If only those cut–off questions are available we can restrict to effective states f_{eff}^0 defined by

$$p(y^{cut}|q_A^{cut}, f^0) = p(y|q_A, f_{eff}^0),$$

. We need this device for the next example.

4.2 Approximate invariances: Smoothness and symmetry

Smoothness and symmetries are probably the most important and most often used nonlocal priors. Here we want to show how restrictions of the measurement device (or equivalently restrictions in the situation (object) generating device) can lead to bounds on smoothness and symmetries.

Let us assume a group S represented by operations s_x acting on x. For simplicity we skip the index x if not necessary and write simply s. Then for scalar states f^0 the action of operation s is defined by requiring invariance $p(y|x,f^0) = p(y|sx, s^{-1}f^0)$ that is

$$p(y|x,sf^0) = p(y|s^{-1}x,f^0).$$

Generalization to states with components not invariant under s, like for example vector states, is straight forward

$$p(y|x,sf^0) = p(s_y y|s^{-1}x,f^0),$$

where s_y is the action of s in the y–representation of the group, which may be different from the x–representation $s_x = s$.

Measuring invariance of the regression function under a set of operations S can for example be done by calculating a mean square error (weighted with $w(x,s)$ if necessary) writing $E(y(f^0,x)) = \bar{y}_x$

$$d_S^2(f^0) = \int dx \int ds\, w(x,s)(\bar{y}_x - \bar{y}_{sx})^2,$$

where s denotes also the index of operator $s \in S$ and the integral notation valid for continuous (Lie) groups has to be replaced by a sum for finite groups. (See for example Ferraro, 1992, for Lie groups in pattern recognition). The distance d_S can be used to construct a prior, for example like $p(f^0) \propto e^{d_S^2(f^0)}$, if normalisable in F^0, or $p(f^0) = \Theta(d_{max}^2 - d_S^2(f^0))$. A relative difference is obtained using $w(x,s) = (1/(x-sx))^2$. One could also measure higher order differences like $((\bar{y}_x - \bar{y}_{sx}) - (\bar{y}_{sx} - \bar{y}_{s^2x}))^2$ corresponding in the infinitesimal case to second derivatives. Measuring smoothness is the special case of measuring infinitesimal translational invariance s_0. For example, written for the one–dimensional case with $w(x,s) = \delta(s-s_0)\frac{w(x)}{(x-s_0x)^2}$

$$d_S^2(f^0) = \int dx\, w(x)(\frac{d\bar{y}_x}{dx})^2.$$

The x–integration $p(x)dx$ can be written in the form $\sum_{i \in orbits} p(s'x_0^i)ds'$, with the orbit i of x_0^i defined as Sx_0^i and the sum is restricted to one x_0^i per orbit. If there is only one orbit, all x can be generated as $x = s_x x_0$ out of one x_0 by repeated applications of the infinitesimal s_0 with $s_x = e^{xs^0}$ denoting the corresponding finite transformations.

For discrete symmetries integrals become sums. Examples of discrete symmetries include permutation of components in case x is a vector. Function spaces of functions depending on vector arguments x can be constructed as tensor product of function spaces depending on the single components of x. If every component of x corresponds to a measurement of another object, exact permutation symmetry means indistinguishable objects.[27]

Now, let us assume that there is input noise or averaging associated to operations $s \in S$ of some group S. This is a very natural assumption for smoothness or infinitesimal translational invariance, as no real measurement device has infinite resolution. If only questions with input noise with respect to S are available we can define an effective state f_{eff}^0. Including the identity into the set S it is defined by

$$p(y|x_{eff},f^0) = p(y|x,f_{eff}^0) = \int ds\, p(s|x)p(y|sx,f^0),$$

with the input noise characterized by $p(s|x)$, that is the probability of posing question sx instead of question x. The following is an example of averaging with respect to S by the same weight factor $p(s|x)$

$$p(y|x_{eff},f^0) = p(y|x,f_{eff}^0)$$
$$= \int dy_s \prod_s p(y_s|sx,f^0)\delta(\sum_s p(s|x)y_s - y).$$

.

Now, we are interested in bounds on the approximate symmetry of the effective regression functions

$$\bar{y}_x^{eff} = E(y,(f_{eff}^0,x)) = \int dy\, p(y|x,f_{eff}^0)$$
$$= \int dy \int ds\, y_{eff}\, p(s|x)p(y|sx,f^0)$$
$$= \int ds\, p(s|x)E(y(f^0,sx)) = \int ds\, p(s|x)\bar{y}_{sx},$$

for the input noise version. The average versions give the same expectation but smaller variance. The result is a convex combination of the local averages $\bar{y}_{sx}$. Now we consider a measurement device with finite range (for its real valued components) of output and define according to the last section a new effective state which includes the cut–off with respect to an upper bound A_x and lower bound B_x. Then the effective regression function can only take values between the extremal points $A_x = \min_s E(y(f^0,sx))$ and $B_x = \max_s E(y(f^0,sx))$ and by changing $p(s|x)$ we can obtain any value in between.

Analogously, differences are bounded. If we take for simplicity $p(s|x) = p(s|x') = p(s)$, we have

$$|\bar{y}_x^{eff} - \bar{y}_{sx}^{eff}| = |E(y(f_{eff}^0,x)) - E(y(f_{eff}^0,sx))|$$
$$= |\int ds'(p(s') - p(s's^{-1}))\bar{y}_{s'x}|,$$

for a parametrization s' with $\frac{ds}{ds'} = 1$. Therefore, changing $p(s)$ allows to obtain any bound d_{eff} for the norm of the difference between $d_{eff}^{max} = \max_x \max((A_x - B_x),(B_x - A_x))$ and $d_{eff}^{min} = 0$ (for $p(s) =$ const.) and we can achieve

$$d_S^2(f_{eff}^0) = d_S^2(f_{eff}^0)\Theta(cd_{eff}^2 - d_S^2(f_{eff}^0)),$$

[27] For example, in physics identical particles like bosons are related to an exact permutation symmetry.

with $\int dx \int ds\, w(x,s) = c$. This bounds the smoothness or deviation from perfect symmetry by $0 \le d_S \le c d_{eff}$ for any $0 = d_{eff}^{min} \ge d_{eff} \ge d_{eff}^{max}$.

Thus, input noise in connection with a cut–off device can lead to dependencies between answers of an effective function by bounding their maximal differences, which is a symmetry or smoothness property.[28] For sampling of symmetry priors by virtual examples and references see Section 8.4.

In real measurement devices these combination of cut–off and input noise effects can occur on many different levels and it is an interesting question whether the omnipresent smoothness phenomena in nature could be partly explained in that way.

5 Subjective priors

5.1 General priors: How probabilistic models are obtained

The preparation of an ensemble F^0 with a certain prior distribution requires measurement or control of certain properties of f^0. States f^0 are defined in terms of a parametrization of $p(y|x, f^0)$. Any prior $p(f^0|f^P)$ describing a state of knowledge (i.e. state of preparation) is therefore a deterministic functional of those $p(y|x, f^0)$, and also of their parametrization. Thus the set of $p(y|q, f^0)$ itself are answers to the maximal set X^P_{max}. The x^p_{max} are the deterministic functions giving $p(y|x, f^0)$. This means the number $p(y|x, f^0)$ for given x, y, f^0, not the probabilistic questions x giving answer y. The preparation questions X^P_{max} allow to construct a set of well defined $p(y|x, f^0)$ producing devices. Usually then one of these devices is selected according to some $p(f^0)$, and one has to find out using new data D which one is actually choosen. With respect to X^P_{max} every prior, even if defined directly by a deterministic functional of the $p(y|x, f^0)$ can be reinterpreted (but not in a unique way) as resulting from a uniform prior $p(f^0)$ with additional given data (y^P, q^P). We take the point of view that every nonuniform prior is caused by such data D, sometimes also denoted by D^0 if we want to distiguish them from other training data D.

Given data D the corresponding state $p(f^0|D)$ can be calculated if $p(D|f^0)$ is known. The $p(D|f^0)$ are part of the structural knowledge. They must be determined independent of the actual task under consideration. Thus, their knowledge is always a transfer of knowledge from another task and assumes constancy of this distributions under transfer. Up to now we assumed the $p(D|f^0)$ to be given. Then the preparation of an ensemble $p(f^0)$ can be related to measurement devices with already known (empirically measured) answer probabilities under the various states. Here we discuss the measurement of $p(D|f^0)$. The main problem is, that we have to measure $p(D|f^0)$ in another situation, and therefore have to ensure its constancy (or make its approximate constancy plausible) to allow transfer.

Measurement or control 'devices' include humans e.g. an ensemble is prepared under human control or described by verbal statements. So, images of a training set may be labelled 'chair' or 'nonchair' by some 'expert in chairs'. To allow any meaningful generalization we must at least approximately (maybe implicitly) transform its concept of a chair into a chair approximator, applicable to all possible images, which then can be improved by training examples. Of course, one may rely on some smoothness condition or other implicit model restrictions (corresponding to a special chair approximator for all possible images) and the training examples alone. But obviously, besides the training examples, any verbal description of a chair also adds to the available information and one may have good reasons to believe that there is a better 'universal first guess chair approximator' than just using the implemented smoothness and other implicit model restrictions. Even if many training examples are available, it can help to use verbal descriptions to create also 'virtual' examples (which must not be an infinite set, assuming that the implicitly or explicitly implemented smoothness conditions interpolate in the neighborhood), because the virtual examples can include data which are not available as training data. They may teach the concept 'chair' far more effectively, than images of real chairs. Thus, we want to find a reliable relation of answers distributions of experts to possible states f^0, i.e. approximate the expert answer probability $p(y^E|q^E, f^0)$. This is obviously a very complicated task and as well a subject of psychological research as of statistics.

More general, one may even take the point of view that human experts are always involved: They have to describe 1. the single data probabilities $p(D|f^0)$ in an, already more precise or still less precise, verbal form, and 2. the dependency of the prior on the various data. E.g. single data have to be combined by AND, OR, or

[28] For example, the support vector machine (Vapnik, 1995) applied to classification problems can be seen as such an approach. Here the input space is imbedded in a (often much) higher dimensional feature space. The classification in feature space only requires the calculation of scalar products which are defined through a positive definite kernel, chosen so that it is easy to calculate, In feature space a linear separating hyperplane is constructed maximizing its distance to the nearest data points, which are also called support vectors. Choosing the separating hyperplane with maximal distance to the nearest sample points is equivalent to maximizing the input noise around the sample points without changing the classification. The cut–off consists in the restriction to data within a certain radius. One may interpret the class boundaries resulting from equal class priors and a class membership probability of the form of a mixture of Gaussians centered at the data points, radially symmetric in feature space with respect to the distance induced by the selected kernel, and with equal variance σ. From this point of view one can say that the support vector machine obtaines a solution with a maximal 'smoothness' of the class membership probability with respect to kernel induced distance in feature space, i.e. with maximal σ. Thus, the support vector machine implements a smoothness prior relative to the feature space. One may remark here, that the related VC dimension of the support vector machine can (up to now) not be calculated exactly, because the related function space $\hat{F}$ of optimal hyperplanes is not defined a priori but dependent on the x–values of the training data. (See Shawe-Taylor, Bartlett, Williamson, Anthony, 1996ab and their concept of a luckiness function.)

more complicated operations. These operations depend on the dependency structure of the single data. One must find a procedure to translate the related verbal information into numbers. This may be rather trivial, if the verbal statement is a reference to a (maybe empirically obtained) numbers. However, often this is not quite as easy. Assume we have data (q, T_{y_q}) available from some approximator T of $p(y|q, f^0)$. We might know that it has been trained on some examples. But what would be $p(T|f^0)$ and $p(y_q|T)$? This clearly depends on the learning algorithm, the approximator was using, as well as on f^0. We may not know the details of the algorithm, and determining $p(T|f^0)$ consistently to our model of basis questions may well be a much harder task than solving our actual problem. Thus, we have to perform an approximate 'internal integration over (hyper-)priors' by considering our experience with this and similar approximators in similar situations. The result will be a verbal statement describing a subjective concept related to $p(T|f^0)$, e.g.: 'We may trust the results of this simple approximator, not too much but a little in all situations which are not too similar to the examples A or B'. This has to be changed into a numerical representation, which includes e.g. translation of 'not too much', 'A', 'B', 'similar to A, B', IF ... THEN, OR, AND. Still one can expect this to be in general a better solution than just ignoring unprecise information: Generalization requires implementation of nonlocal dependencies, and trying to match those nonlocal dependencies to unprecise verbal descriptions seems better than using nonlocal dependencies which are implicitly implemented without any further reasoning. Of course, there may be individual situations where the unknown implicit assumptions are better suited to the problem. But if this occurs regularily, the specific method used to include unprecise information has and can be adapted.

In principle, every prior could be seen as resulting from one combined prior question, and therefore be constructed like any generalized question. However, especially for priors, but possibly for every question, the necessary ingredients have to be constructed from more or less unprecise information in verbal form. We therefore shortly comment the standard operations AND and OR for probabilities from this point of view. Consider a prior, prepared by applying preparation question q^P and enforcing or measuring an empirical answer y^P. One obtains

$$p(f^0|f^P) \propto p(f^0)p(y^P|q^P, f^0).$$

According to the construction of generalized questions we can translate our verbal statement that the data (q_i^P, y_i^P) should be combined by an AND for conditional independent events by forming the product

$$p(f^0|f^P) \propto p(f^0) \prod_i p(y_i^P|q_i^P, f^0).$$

But we might not be to sure about their independence, without knowing any explicit dependency structure. Again we may delegate the task to an expert and translate its verbal output either in 1. conditional probabilities determining the dependency between y^p and apply the correct probability theoretical formula, or 2. directly into an adapted combination of the $p(y_i^P|q_i^P, f^0)$ which might well look different from the above formula for independent events.

Similarily, a partial sum (or integration for densities) correspond to a (weighted) OR would tranlate statements, refering to measurement noise for disjunct events:

$$p(f^0|f^P) \propto p(f^0) \sum_i p_i^0 p(y_i^P|q^P, f^0),$$

or

$$p(f^0|f^P) \propto p(f^0) \sum_{i,j} p_i^0 p(q_j^P) p(y_i^P|q_j^P, f^0).$$

The following statement 'The result was either 1 or 7, I am not sure.' would fit quite well into that category, provided we can relate the p_i^0 to subjective beliefs. Again, instead of guessing p_i^0 and the dependency structure and applying the correct probability theoretical formula, one may translate a verbal statement directly into $p(f^0|f^P)$ and the resulting formula could look differently.

In general, instead of constructing the ingredients from verbal statements and then inserting them in probability theoretical formulas, one may also directly construct the result from unprecise knowledge. The results will differ if the subjective concepts do not represent probabilities. This is the case if the dependency structure is easier formulated in a verbal form than in terms of conditional probabilities. On the other hand, one may try to improve verbal statements, by enforcing experts to use probabilistic models. In all cases we start with unprecise verbal information and require a probabilistic model at the end. The transition can be done on various levels.

5.2 Fuzzy priors

5.2.1 Human control and subjective probabilities

Let us therefore consider in more detail the case where preparation and problem definition is under human control and expert knowledge about the problem domain is available, for example in a verbal form instead in terms of a probabilistic model. Those situations rely more on a subjective instead of the empirical interpretation of probability. 'Subjective probability'. may simply mean the answer of an expert, which has been asked to give a probability. But those guesses do not need to be very accurate. In cases empirical probabilities are available one can compare subjective probabilities obtained by different methods with empirical probabilities. Typical tendencies of deviations of subjective estimates from empirical probabilities have been studied (Tversky, 1972; Tversky & Kahneman, 1981; Kahneman & Tversky, 1979, 1982abc; Kahneman, Slovic, Tversky, 1983. See Lemm, 1984, for an example related to information costs in decisions). They include, besides many others, overestimation of small probabilities or of probabilities for easily retrievable, salient events, neglecting the base rate of events, the sample size or correlations, tendency towards a chosen reference point like underestimation for the sum of probabilities and overestimation for joint probabilities. Subjective estimates of probabilities do usually not obey the rules of probability theory, for example, independent obtained estimates for probabilities for events A and NOT A need not necessarily sum up to one. In addition to the difficulties caused by the deviations between subjective estimates and empirical probabilities it is often not obvious how to describe an event A to which a subjective estimate is related. A subjective estimates for probability $p(A)$ refers to an internal representation of A. To relate the estimate to a probability for some external A the internal representation of A (e.g. concept of a mouth) must be related to the external event (e.g. images of mouths). That means A has to be identified with a set of f^0, described by an explicit parametrization.

The process in obtaining subjective probabilities may be outlined as follows: We aim in producing a guess for probability $p(f^0)$. Let us call any deterministic question $C(f^0)$ (i.e. function or functional, respectively) of the parameters of f^0 a *property* of f^0 and $\tilde{C}$, its subjective representation, a *concept*. We want to construct a final property $C(f^0)$ which is related to $p(f^0)$ by some function g, i.e. $p(f^0) = g(C(f^0))$. For example $p(f^0) \propto C(f^0)$, or $p(f^0) \propto e^{C(f^0)}$. The function g could also be used to compensate for known estimation biases. The subjective representation $\tilde{C}$ used to produce subjective probabilities might de difficult to relate directly to a specific function $C(f^0)$ of the parameters of f^0. It can be easier for other, simpler concepts $\tilde{C}_i$. For example, the concept $\tilde{C}$ of having something similar to a nose, a mouth and two eyes with certain possible spatial relations, might be difficult to relate to pixel values directly. But for a simple enough concept $\tilde{C}_i$ a property C_i might be more accurately related.

A property C_i measuring distance can be built out of a property C_i' and a related *template* T_i, using a monotonic function of a 'meaningful' distance of the expectation, e.g.

$$C_i = ||C_i'(f^0) - T_i||^2.$$

C_i' could be an arbitrary question. For $C_i(f^0) = \bar{y}_x(f^0)$ this measures the square distance of $\bar{y}_x(f^0)$ at point x from some reference template $\bar{y}_x^T$, i.e. $||\bar{y}_x(f^0) - T_{\bar{y}_x}||^2$ and represents a usual mean square error term. Templates could also be defined relativ to another question

$$||C_i(f^0) - C_{T(i)}(f^0)||,$$

like in the case of symmetries, but also in cases where not invariance but any arbitrary dependence between C_i and $C_{T(i)}$ is measured.

We will write $C_i = G(\tilde{C}_i)$ for the relation between concepts and properties. Properties C_i related to (linguistic) concepts $\tilde{C}_i$ have been called linguistic variables and have been used in the theory of *fuzzy sets* (see for example the collections Zadeh, 1987, or the more recent one, Zadeh, 1996). Subconcepts are modified and combined by concept functions $\tilde{F}$ to form the final concept C. For example, we may require: 'great similarity to T_1 and T_2 if not already very similar to T_3'. In analogy to concepts, also concept functions $\tilde{F}_i$ must be mapped to functions $F_i = H(\tilde{F}_i)$ acting on C_i. We will call func-

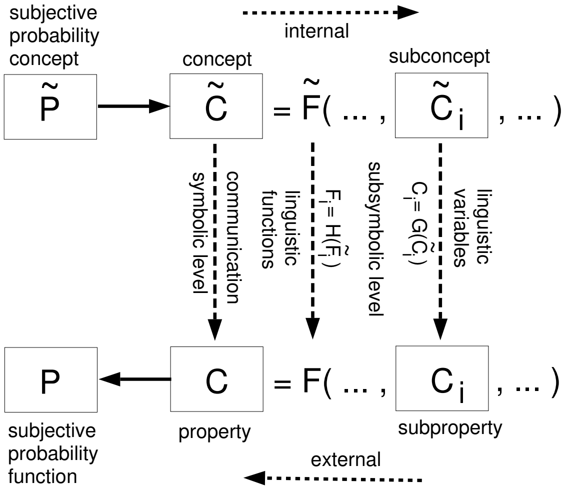

Figure 3: The relation between (internal) concepts and (external) properties. For complex concepts $\tilde{C}$ the relation to property C is often easier to obtain and more invariant under transfer to new situations, if the mapping from concept to properties (subsymbolic mapping for linguistic variables) is done for simpler subconcepts $\tilde{C}_i$. This, however, requires also a (subsymbolic) mapping of concept functions $\tilde{F}$, acting on concepts $\tilde{C}_i$, to property (external, linguistic) functions, acting on properties C. Then, the property C for a communicated symbolic structure $\tilde{C} = \tilde{F}(\{\tilde{C}_i\})$ can be approximated by $C = F(\{C_i\})$.

tions F related to concept functions $\tilde{F}$ linguistic functions. The mappings G for concepts and H for concept functions represent the subsymbolic level. The communication of the (symbolic) structur of $\tilde{C}$ in terms of concept variables and concept functions, i.e. $\tilde{C} = \tilde{F}(\{\tilde{C}_i\})$ is used for the approximation

$$C = F(\{C_i\}) = H(\tilde{F})(\{G(\tilde{C}_i)\}),$$

where $H(\tilde{F})$ stands for the function F with all included subfunctions replaced according to $F_j = H(\tilde{F}_j)$. (See Fig.3). We try to achieve

$$G(\tilde{C}) = G(\tilde{F}(\{\tilde{C}_i\})) \approx F(\{C_i\}) = H(\tilde{F})(\{G(\tilde{C}_i)\}).$$

However, this is difficult to check in general if $G(\{\tilde{C}\})$ cannot be obtained consistently in a direct way. Indeed, this difficulty is the reason for the decomposition into subconcepts. One can instead define C by the right hand side and check variability of dependencies from those properties under tranfer to new situations. Thus, this can be seen as a heuristic method to achieve dependencies with smaller variance under transfer.

We will call priors obtained by this method *fuzzy priors*. In the following we do not intend to give an introduction into fuzzy logic. We mainly want to stress that related techniques can be adapted to construct priors and the more technical point that this can lead to nonlinearities in the equations determining the maximum probability $p(f^0)$ even if single C_i correspond to Gaussian probabilities.

The method can be applied in two variants. The first way of defining $p(f^0|f^P)$ is directly to describe the functional in its dependence from the parameters without explicit reference to data. A second possibility is to define in a first step $p(y^P|q^P, f^0)$ in its complete dependence from all three variables and then choose in a second step y^P to fix the prior. We will mainly study the first possibility and we will study linguistic functions $C_C = F(C_A, C_B)$ depending on two one–dimensional properties. Their combination gives functions depending on more than two variables.

5.2.2 Real valued extensions of logic

Linguistic functions can be constructed by fixing there values at certain, typical points and using some (e.g. smooth, symmetric) interpolation scheme. For example, practically important linguistic functions are such related to logical functions like AND, OR or NOT used in *fuzzy logic* (see e.g. Kandel, 1982, Klir & Yuan, 1995). From such functions we assume that they coincide with the corresponding binary logical functions at the four corners, i.e. where both arguments have a minimal or maximal value. We will mainly concentrate on this functions but also model combinations which do not correspond to binary logical functions at the four corner points.

For example, we may assume high probability for a function f^0 if it is smooth at x_1 AND x_2 AND $\cdots$. In the second variant of the method we begin with the construction of answer distributions $p(y|q, f^0)$ and a prior results by choosing data (y_i^P, q_i^P). There, for example in face recognition, q^P can be defined as question looking for components like eyes in the images y produced by $p(y|\,x = \text{face}, f^0)$ and the y^P can be examples of how eyes can look like. Requiring that f^0 produces (all of) several variants V_i of eyes, i.e. its output is V_1 OR V_2 OR $\cdots$, means ANDing for constructing a prior, so f^0 can produce V_1 AND V_2 AND $\cdots$.

Consider properties with $0 \le C_i \le m$, where we may allow the limit $m \to \infty$. We use here a 'distance interpretation' of the values 0 and m where we interpret the value m as complete absence ('far' or 'False') and the value zero as complete presence ('near' or 'True') of that property (deviation property). Monotonic function from (bounded) distances to some templates are examples. The interpretation of 0 and m can be reversed (similarity property), like in the usual convention in logic or for properties like probabilities where 0 means 'False' or 'impossible' and $m = 1$ 'True' or 'sure'. Then, just the definitions of AND and OR in the following have to be exchanged.

The following are examples of a function equal to a binary logical function when the arguments have values 0 or m:

$$\begin{aligned} C_{(A \text{ AND } B)} &= C_A + C_B - C_A C_B/m, \\ C_{(A \text{ OR } B)} &= C_A C_B/m, \\ C_{(\text{NOT } A)} &= m - C_A. \end{aligned} \tag{5}$$

These operations represent a Boolean Algebra (see for example Whitesitt, 1995) and therefore for variables taking only the values 0 and m for example DeMorgan's law A OR B = NOT ((NOT A) AND (NOT B)) is valid.

The functions are extended to real variables by allowing values $0 \le C \le m$ in the above formula. Of course, this extension to real values cannot be unique. So AND can also be implemented as

$$C_{(A \text{ AND } B)} = c_m(C_A + C_B).$$

where c_m is a cut–off function with $c_m(x) = id$ for $0 \le x \le m$ and $c_m(x) = m$ for $x \ge m$ and therefore always $c_m(C_A) = C_A$. With this, DeMorgan's law gives for OR

$$C_{(A \text{ OR } B)} = m - c_m(2m - C_A - C_B)).$$

Not differentiable near our main point of interest at $(0, 0)$ where the 'good' f^0 should be located but m–independent are the following realizations

$$\begin{aligned} C_{(A \text{ AND } B)} &= \max(C_A, C_B), \\ C_{(A \text{ OR } B)} &= \min(C_A, C_B). \end{aligned}$$

This representation of AND and OR by the maximum resp. minimum operation are the standard fuzzy operations used in fuzzy logic.

A variable C_A can always be written $C_A = C_A^2/m$, $C_A = 2C_A - C_A^2/m$, $C_A = c_m(C_A)$. Any (usually monotonic) interpolating function $\sigma(x)$ with $\sigma(0) = 0$ and $\sigma(m) = m$, allows to replace A by $\sigma(A)$ without changing the limit of binary logic. For example, OR could be defined as $\sigma_{OR}(\sigma_A(C_A)\sigma_B(C_B)/m)$ choosing some (monotonic) functions σ_{OR}, σ_A, σ_B. Rules like the law of deMorgan are valid for variables with values 0 and m but not in general for real values. Fig.4 shows two real valued extensions of AND and OR and Fig.5 some tautologies which can be added to every function without changing the binary limit. Of course, any real valued extension of logical functions is arbitrary except at the four corners. Thus, one might add additional conditions for such functions, like monotonicity or smoothness conditions, or the requirement that certain laws of the Boolean algebra, valid for the binary limit also hold for real values, requiring for example an associative, commutative, and distributive AND and OR. Besides having the correct boundary conditions, usually monotonicity, commutativity, and associativity are required for fuzzy operations. (Such operations are called t-norm or t-conorm for boundary conditions corresponding to AND or OR.)

5.2.3 General combinations

Every logical formula can be expressed by NOT, OR, AND. In particular, one could use the disjunctive or conjunctive normal form. Also, AND or OR could be eliminated using DeMorgan's law. For example, the exclusive OR is defined as XOR = (A AND NOT B) OR (NOT A AND B). For disjunct events XOR is equivalent to OR. Accordingly, real extensions of the exclusive OR are

$$C_{XOR} = \frac{1}{m} c_m(C_A + m - C_B) c_m(m - C_A + C_B).$$

or

$$C_{XOR} = \frac{2C_A C_B}{m} - C_A - C_B + m.$$

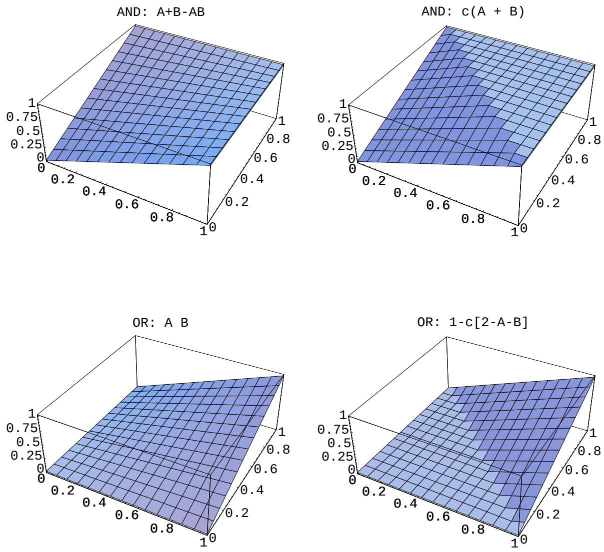

Figure 4: Different realizations of AND and OR, in a 'distance interpretation' where 0 stands for 'True' ('near') and 1 for 'False' ('far'). C_A, C_B are abbreviated A, B and the cut–off function c_1 as c.

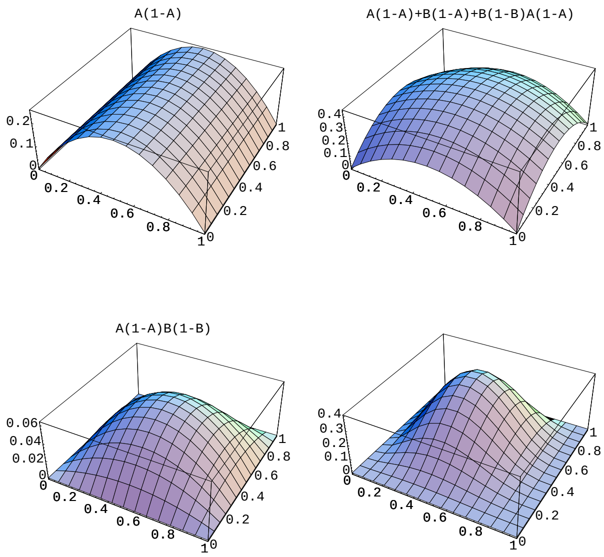

Figure 5: Examples for real valued extensions of tautologies. The last one has the formula $1-(1-A^2(1-A)^3+1-B^2(1-B)-(1-A^2(1-A)^3)(1-B^2(1-B)))^{100}$.

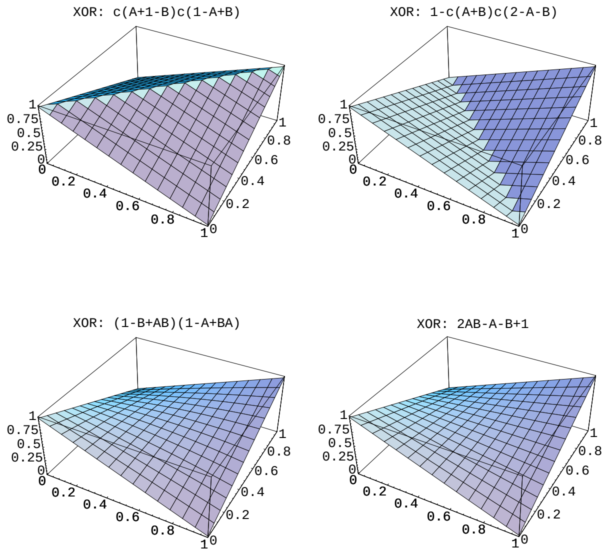

Figure 6: Different realizations of XOR

The latter result can be connected to eqs.(5) if using $C_i^2/m = C_i$. Fig.6 shows some possible variations of XOR.

Another important relation is IF C_1 THEN C_2 = NOT C_1 AND C_2. One way to extend this to real values is

$$C_{IF} = c_m((m - C_1) + C_2).$$

The limit of binary logic requires function values 0 or m at the four corners where the input variables are 0 or m. Any function having more than two different values at those four points cannot correspond to a logical function. The only possible linear functions are for example 1, C_A, C_B and their negations. Therefore combinations like $aC_A + (1-a)C_B$ (LIN = linear) are more general and do not correspond to logical combinations at the four corners. However, setting $a = C_C$ LIN appears to be a combination of three logical properties with the value of C_C fixed (for all f^0).

Thus, such functions can be seen as parts (of combinations) of logical functions with certain values of C_i excluded (e.g. with the value fixed). Specifically, in the limit $m \to \infty$ and C_i finite we obtain for the above rules for $C_A \neq \infty \neq C_B$

$$\begin{aligned} C_{(A \text{ AND } B)} &= C_A + C_B, \\ C_{(A \text{ OR } B)} &= 0, \\ C_{(\text{NOT } A)} &= \infty. \end{aligned} \tag{6}$$

These are only linear functions. To allow the value ∞ for the C one has to extend the definition of OR and NOT according to

$$\begin{aligned} C_{(A \text{ AND } B)} &= \infty \quad \text{for } C_A = \infty \text{ or } C_B = \infty, \\ C_{(A \text{ OR } B)} &= \begin{cases} C_A \text{ for } C_B = \infty \neq C_A, \\ C_B \text{ for } C_A = \infty \neq C_B, \\ \infty \text{ for } C_A = C_B = \infty. \end{cases} \\ C_{(\text{NOT } A)} &= 0 \quad \text{for } C_A = \infty. \end{aligned}$$

and one finds that the functions on the whole intervall $[0, \infty]$ are nonlinear. This representation is interesting in as far as the AND is linear and the nonlinearity of the OR can be implemented by just skipping functions with final $C(f^0) = \infty$ from F^0. We will call this a hard implementation of the OR and other implementations soft. Then for all functions f^0 under consideration, i.e. in F^0, $C(f^0)$ is a linear function of its constituents $C_i(f^0)$. But note that unlike for finite m here $C_{(\text{NOT NOT } A)}$ is not equal C_A for $0 \neq C_A \neq \infty$.

LIN in contrats to AND is strictly monotonically in all C_i even if one $C_{i^*} = m$. This allows also if $C_{i^*} = m$ cannot be changed (momentarily), the use of gradient information to improve the other C_i. To include information related to OR one can combine LIN with a multiplicative implementation of OR:

$$\begin{aligned} C_{(A \text{ LIN } B)} &= aC_A + (1-a)C_B, 0 < a < 1, \\ C_{(A \text{ OR } B)} &= C_A C_B / m. \end{aligned} \tag{7}$$

5.3 Special Properties: probabilities, logprobabilities, distances, and averages

We shortly discuss some special properties: probabilities (or partition sums, if not normalized), logprobabilities (related to free energies), distances (related to scalar products) and averages (expectations, related to energies).

5.3.1 Probabilities

For probabilities (not densities) the logical combination of events are defined

$$\begin{aligned} p(A \text{ AND } B) &= p(A)p(B|A) \\ p(A \text{ OR } B) &= p(A) + p(B) - p(A)p(B|A) \\ p(\text{NOT } A) &= 1 - p(A). \end{aligned}$$

(The integration of densities to yield probabilities can be seen as OR for disjunct events. For densities we can interpret the rules as $p(x_1 AND x_2)dx_1 dx_2 = p(x_1)p(x_2)dx_1 dx_2$, $p(x = (x_1 OR x_2))dx = (p(x = x_1) + p(x = x_2))dx$, $p(NOT x)dx = (\delta(x) - p(x))dx$ for disjunct events x_i.) Notice, that with respect to the two (similarity) properties $P(A)$ and $P(B)$ the probabilistic AND, i.e. $p(A)p(B)(p(B|A)/p(B))$, is not one function but a whole family, parametrized by $p(B|A)$, but all coinciding at the four binary corners. That means the probabilistic AND changes with the dependency of A on B.

To take advantage of independence in the case of OR one can use De Morgan's law and write for a set of independent A_i $p(OR_i A_i) = 1 - \prod_{i=0}(1 - p(A_i))$, where $p(NOT(OR_i A_i))$ factorizes. This form is also known as 'noisy OR' (Pearl, 1988, Jensen, 1996), and one may set one A_i constantly equal to one (or zero) to have a nonzero baseline if all other A_i are zero (or one).

5.3.2 Logprobabilities

Assume $0 \le p \le 1$ is a probability (or for AND and OR a bounded density $0 \le p \le c$) then the logprobability $L = \ln p$ is in the intervall $[0, -\infty]$ (or $[\ln c, -\infty]$, respectively) and the rules for probabilities become

$$\begin{aligned} L(A \text{ AND } B) &= L(A) + L(B|A) \\ L(A \text{ OR } B) &= \ln\left(e^{L(A)} + e^{L(B)} - e^{L(A \text{ AND } B)}\right) \\ L(\text{NOT } A) &= \ln(1 - e^{L(A)}). \end{aligned} \tag{8}$$

We write $L(B|A) = L(B) + \Delta(A, B)$ and expand the OR for logprobabilities around some L_0 (e.g. the minimum of L corresponding to the maximum of P or the average $(L(A) - L(B))/2$), up to quadratic and bilinear terms in $L(A)$, $L(B)$, and Δ

$$L(A \text{ OR } B) \approx c - ((L(A) - L_0)(L(B) - L_0) + \Delta(1 + L(A) + L(B) - 2L_0) + \Delta^2) .$$

Thus, for independent A, B, i.e. $\Delta = 0$, with probabilities $P(A) \approx P(B) \approx e^{L_0}$, OR may be represented by multiplication. For $L_0 = 0$ the rules for negative logprobabilities correspond in linear order for independent A, B to the rules (6) and in second order, up to the factors m, a, to the rules (7).

For disjunct A, B we obtain in first order

$$\begin{aligned} L(A \text{ OR } B) &\approx L_0 + \ln 2 + \frac{1}{2}\left((L(A) - L_0) + (L(B) - L_0)\right) \\ &= \ln 2 + \frac{L(A) + L(B)}{2} \\ &= \frac{1}{2}\left(\ln\frac{e^{L(A)}}{2} + \ln\frac{e^{L(B)}}{2}\right), \end{aligned}$$

independent of L_0, and exact if $L(A) = L(B) (= L_0)$. The last line shows that this is an *annealed approximation*, i.e. the nonlinear function of the average is replaced by the average of the nonlinear functions $g(<x>) = g(\sum_i p_i x_i) \approx \sum_i p_i g(x_i) = <g(x)>$, with $\sum_i p_i = 1$. For convex functions like the logarithm the inequality $g(\sum_i p_i x_i) \le \sum_i p_i g(x_i)$ holds, with equality if all x_i are equal.

5.3.3 Distances

Monotonic functions of negative *distances* $||y^q(f^0) - T_{y^q}||^2$ are often used for logprobabilities. The definition of q can always be changed in such a way that $T_{y_q} = 0$. The most common example are data terms $d^2 = \sum_i \frac{1}{2\sigma_{q_i}^2}(y_i^{q_i} - \bar{y}_{q_i}(f^0))^2$ where the $y_i^{q_i}$ represent the template vector and $\bar{y}_{q_i}$ the deterministic answer $y^{q_i}(f^0)$.

In spaces where the distances fulfill

$$||g - h||^2 = 2||g||^2 + 2||g||^2 - ||g + h||^2$$

there exists a scalar product, written as $<\cdot\,|\,\cdot>$, related to the distance by

$$||h||^2 = <h\,|\,h> .$$

Different positive definite kernels $\mathcal{K}$ define different scalar products

$$<g\,|\,h>_{\mathcal{K}} = \int dx \int dx'\, g(x)\mathcal{K}(x, x')h(x').$$

Minimizing squared distances

$$||y^q(f^0) - T_q||^2 = <y^q(f^0)|y^q(f^0)> - 2<y^q(f^0)|T_q> + c,$$

(written for real $<y^q(f^0)\,|\,T_q> = <y^q(f^0)\,|\,T_q>^* = <T_q\,|\,y^q(f^0)>$ with c an f^0-independent constant and $*$ indicating complex conjugation) is equivalent to maximizing scalar products (*overlaps*), like $\sum_i \frac{1}{2\sigma_{q_i}^2} T_{q,i}\bar{y}_{q_i}$, for normalized y and $\bar{y}$. (But normalization is a maximal nonlocal condition.) For overlaps one has to use rules with zero representing false and one representing true so the definitions of AND and OR are exchanged.

For example, properties which have positive values bounded by m can be obtained from logprobabilities or distances by $C_A = g^{-1}(d^2(A)) = m(1-e^{-d^2(A)}) = m(1-e^{L(A)}) = m(1-p(A))$ with inverse $L = g(C) = -\ln(1-C/m) = \ln(m/(m-C)$. This gives for independent A and B the equations (5) for properties. Thus, using $g(C) = -\ln(1-C/m)$ directly relates properties to probabilities according to $C_i = m(1-p_i)$.

Writing OR in terms of Euclidean square distances by choosing $L_i = -d_i^2/(2\sigma_i^2) + \ln c_i$, with some constants σ_i and c_i, one obtains for disjunct events a *Gaussian mixture model*

$$p(f^0) \propto \sum_i c_i e^{-d_i^2(f^0)/(2\sigma_i^2)}.$$

For nondisjunct events product terms like

$$\prod_i e^{-d_i^2/\sigma_i}$$

must be subtracted (or approximately a cut-off function can be included).

5.3.4 Averages

Unnormalized probabilities

Averages or expectations can be related to unnormalized probabilities $Z(A,\beta)$, (with β a fixed parameter we will discuss below) also called *partition sums*

$$p(A) = \frac{Z(A,\beta)}{Z(\beta)}.$$

Thus,

$$\begin{aligned} Z(A,\beta) &= \sum_{\omega\in A} Z(\omega,\beta) \qquad (9)\\ Z(\beta) = Z(\Omega,\beta) &= \sum_{\omega\in\Omega} Z(\omega,\beta) \end{aligned}$$

for a complete set of disjunct (elementary) events ω, and $p(\Omega) = 1$. Partition sums transform, under logical transformation like probabilities with an additional normalization factor Z analogously to m in the rules (5), if we define $Z(A|B)$ by $p(A|B) = Z(A|B)/Z$ (what we will do for unnormalized probabilities). If we choose $Z(A|B) = Z(A,B)/Z(B) = p(A|B)$ (what one can do for averages, see below) then Z appears only in the NOT. Defining shifted logprobabilities $-\beta F$ by

$$Z(\beta) = e^{-\beta F(\beta)}, \quad Z(A,\beta) = e^{-\beta F(A,\beta)},$$

the F, also called *free energies*,

$$F(\beta) = -\frac{1}{\beta}\ln Z(\beta), \quad F(A,\beta) = -\frac{1}{\beta}\ln Z(A,\beta),$$

transform[29]

$$\begin{aligned} F(A \text{ AND } B) &= F(A) + F(B|A) - F\\ F(A \text{ OR } B) &= -\frac{1}{\beta}\ln\Big(e^{-\beta F(A)} + e^{-\beta F(B)}\\ &\qquad -e^{-\beta F(A \text{ AND } B)}\Big) \qquad (10)\\ F(\text{NOT } A) &= -\frac{1}{\beta}\ln(e^{-\beta F} - e^{-\beta F(A)}). \end{aligned}$$

For the sake of simplicity the β–dependence of F is here not written explicitly.

We choose a family A of disjunct events $\omega \in \Omega$. For the corresponding ω we define the free energies $F(\omega,\beta) = E(\omega)$ to be β–independent. We will call them *energy* of ω

$$p(\omega) = \frac{e^{-\beta E(\omega)}}{Z(\beta)}. \qquad (11)$$

We can think of the random variable E being dependent on other random variables $E_j(\omega)$ so that $\beta F(\omega,\beta) = g(\{E_j(\omega)\})$. Useful will be a linear combination, for which we write[30]

$$\beta F(\omega) = \beta E(\omega) = \sum_j \beta_j E_j(\omega),$$

with E_j independent of β_j and β. For the sake of simplicity we will understand $F(A,\beta)$ to mean $F(A,\beta,\{\beta_j\})$ in that case, and analogously for other variables like Z and p.

Averages and generating functions

Partition sums can be seen as averages under a uniform probability distribution $p(\omega)$. In general, *averages* or *expectations* of a function $h(\omega)$ over a family A of disjunct events $\omega \in A$ are defined as

$$h(A,\beta) =< h(\omega) >_{A,\beta} = \frac{\sum_{\omega\in A} p(\omega,\beta)h(\omega)}{p(A,\beta)},$$

and we can include $h(A|B) = h(A,B)/h(B)$. We can extend the definition of $E(\omega)$ to $E(A,\beta)$ for all events $A \in \Omega$ by defining the energy to transform like (i.e. to be) an average,

Thus, E is a random variable bounded from below, defined by the number (or vector) $E(\omega) > -\infty$ related to every elementary event $\omega \in A$. We may call $p(\omega)$ the distribution generated by βE which can have the form $\sum_j \beta_j E_j$. On the other hand every (vector) βE, with components bounded from below, is a 'generating random variable' of some p, with $Z(A,\beta)$ the normalization constant of $e^{\beta E(\omega)}$ on A.

[29] Clearly, the OR looks quite complicated, even for disjunct events, and indeed, it is the scource of many difficulties. For example, it is related to the calculation of partition sums in statistical physics. The replica approach (Mezard, Parisi, & Virasoro, 1987) deals with the asymptotic limit of an OR over two complete sets of disjunct events (states and quenched interactions), i.e. with an OR for nondisjunct events.

[30] Then in physics usually only one subgroup E_j is called energy, another might be called particle number.

Now we want to calculate the expectations of the generating random variable E_j. For that purpose we use the required β –independency of E and E_j to write

$$E_j(\omega) = \frac{d}{d\beta_j}\sum_j \beta_j E_j(\omega) = -\frac{d}{d\beta_j}\ln Z(\omega,\beta),$$

which includes the case of a one component E.

Analogously, using eqs. (9) and (11) gives

$$-\frac{d}{d\beta_j}Z(\beta) = -E_j(\omega)Z(\beta)$$

so that we find the same relation for general A

$$E_j(A,\beta) = < E_j(\omega) >_{A,\beta} = -\frac{d}{d\beta_j}\ln Z(A,\beta).$$

Linearity of the expectation allows to use this also to calculate averages of random variables which are linear functions of $E(\omega)$. To calculate expectations and higher order moments or cumulants of general random variables $h(\omega)$ which are either nonlinear functions of $E(\omega)$ or which are no functions $h(E(\omega))$ of $E(\omega)$, like e.g. if $E(\omega) = E(h(\omega)^2)$, one can extend the partition function by adding $-\lambda h(\omega)$ ('auxilliary field') to the exponent

$$Z(\omega,\beta,\lambda) = e^{-\beta E(\omega) - \lambda h(\omega)},$$

so that

$$h(A,\beta) = -\frac{d}{d\lambda}\ln Z(A,\beta,\lambda)\big|_{\lambda=0}.$$

In general we find

$$\frac{Z(A,\beta,\lambda)}{Z(A,\beta)} = < e^{-\lambda h(\omega)} >_{A,\beta} = \sum_k^\infty \frac{(-\lambda)^k}{k!} < h^k(\omega) >_{A,\beta}$$

which therefore is the *moment generating function*, i.e. the kth–moments $M_k(h,A,\beta) = < m^k(\omega) >$ can be found by differentiation

$$M_k(h,A,\beta) = \frac{d^k}{d(-\lambda)^k}\frac{Z(A,\beta,\lambda)}{Z(A,\beta)}\big|_{\lambda=0}.$$

Cumulants are defined as derivatives of the logarithm of the generating function, also called *cumulant generating function*,

$$C_k(h,A,\beta) = \frac{d^k}{d(-\lambda)^k}\ln Z(A,\beta,\lambda)\big|_{\lambda=0},$$

where we skipped the λ–independent term $\ln Z(A,\beta)$ which is not relevant after differentiation. It is easy to see that cumulants have the nice property of being additive if the probability $p(\omega) = \prod_i p_i(\omega)$ factorizes into independent subsystems[31] (Beck & Schlögl, 1993). The second cumulant is the well known variance $< h^2 > - < h >^2$.

[31] For examply spatial systems with a weakly enough short range interaction, so they can be seen in the large n–limit as a collection of independent local subsystems, have cumulants which are asymptotically additive in these subsystems. In these cases, besides the p–generating energy, the free energy $F \propto \ln Z$ and not Z is an additive variable. As the variance of a sum of n independent random variables scales with $1/n$ the cumulants have then for large systems low variance, hence they are proportional to the volume (extensive variables) and possible candidates for macroscopic observables of the system.

For the case of a Gaussian $p(\omega,\beta)$, i.e. $E(\omega)$ quadratic in ω, (multidimensional for vector ω) *Wicks theorem* gives higher order moments, often represented by diagrams (see for example Negele, Orland, 1988).

Boltzmann–Gibbs–distributions

Now we want to relate the shifted logprobabilities $-\beta F(\omega)$ to the expectations of the generating random variable E_j. In general, for $A \neq \omega$ the difference between energy and free energy (i.e. between averages and shifted logprobabilities, coinciding on the elementary events ω) reads

$$\begin{aligned} H(A,\beta) &= \beta E(A,\beta) - \beta F(A,\beta) \\ &= -\sum_{\omega\in A}\frac{p(\omega,\beta)}{p(A,\beta)}\ln\frac{p(\omega,\beta)}{p(A,\beta)} \\ &= -< \ln\frac{p(\omega,\beta)}{p(A,\beta)} >_{A,\beta}. \end{aligned}$$

Here $H(A,\beta)$ is the *average information* (in nats, not bits) or *entropy*. It is well known, that the distribution generated by βE maximizes the entropy (or equivalently minimizes the free energy F) under the constraints that $p(\omega)/p(A)$ is normalized and the expectations of all expectations $E_j(A,\beta)$ are fixed (See e.g. Balian, 1991). Indeed, the stationarity equations read, introducing the corresponding Lagrange multipliers β_j, α

$$\frac{d}{dp(\omega,\beta)}\left(H(A,\beta) - \sum_j \beta_j E_j(A,\beta) - \alpha < 1 >\right) = 0,$$

giving the *Boltzmann–Gibbs distribution*:

$$p(\omega,\beta) = \frac{e^{-\sum_j \beta_j E_j(\omega,\beta)}}{Z(\beta)},$$

with $Z(\beta) = e^{\alpha(\beta)+1}$. We recognize the general form (11). Thus we can say a probability distribution is the Boltzmann–Gibbs distribution of its generating variable(s) βE. Families of distributions generated by varying (the vector) $-\beta$ within some parameter space B are called *exponential families* (See for example Barndorff–Nielsen, 1978, Amari, 1985, 1995, or Appendix D in Lauritzen, 1996 and references therein).

The model from Section 2 defines an elementary event $\omega = (q,y,f^0)$ or, if we include the internal variables y^q z^q of questions q, $\omega' = (q, X^q, Y_{X^q}, Z^q, f^0)$, with $q \in Q^l$, X^q the part of the basis used in Q^l, Y_{X^q} the corresponding answers for $x \in X^q$, and Z^q the set of internal noise variables for q. We are for example (in the case of deterministic $\hat{l}$ and of $p(q)$ not dependent on other variables) interested in calculating the expected risk $r = < l(\omega,\hat{f}) >_{p,\Omega}$ under the total posterior probability

$$p(\omega) = p(q)p(y|q,f^0)p(f^0|D) = e^{L(q)+L(y|q,f^0)+L(f^0|D)},$$

in terms of logprobabilities $L(\omega) = L(q) + L(y|q,f^0) + L(f^0|D)$. Thus we can interpret the Bayesian posterior distribution as Boltzmann–Gibbs distribution arising from a maximum entropy procedure for the total logposterior $L(\omega)$ with an average so that $\beta = -1$ and shifted so that $Z(\Omega) = 1$. It seems simpler, to use the traditional Bayesian formulation than the equivalent maximum entropy formulation. However, there are cases where averages are directly measurable, with measurement error nearly zero.[32] Besides deterministic variables, self–averaging variables in large systems belong to this class. Examples, are macroscopic observables in physics, like energy, which lead to the (generalized) canonical ensembles used in statistical physics. An indeed, in these cases the various implementations (i.e. models F^0, like microcanonical, canonical, grandcanonical ensemble) are asymptotically equivalent for large systems.

Approximation and Kullback–Leibler entropy

Consider another probability distribution $p'(\omega)$ generated by $E'(\omega) = -\frac{1}{\beta'}\ln p'(\omega,\beta') + F'(A,\beta') = -\frac{1}{\beta'}(\ln p'(\omega,\beta') - Z(A,\beta'))$, with the same normalization, i.e. $Z(A,\beta) = Z'(A,\beta')$, so the expectation of E' under $p(\omega)$ reads

$$E'(p,A,\beta) = < E' >_{(p,A,\beta)}$$

$$= -\left(\frac{1}{\beta'} < \ln p'(\omega,\beta') >_{(p,A,\beta)} + F(A,\beta)\right) p(A)^{-1},$$

where E' is averaged over $p(\beta)$ and not over $p'(\beta')$ generated by itself. Concavity of the logarithm function allows to apply Jensen's inequality to the difference $\beta' E'(p,A,\beta) - \beta E(A,\beta) = < \ln p > - < \ln p' > = K(p,p')$, i.e. the *Kullback–Leibler entropy* and we obtain

$$\beta' E'(p,A,\beta) \geq \beta E(A,\beta).$$

Thus, the βE generating the averaging distribution has the minimal average under this distribution. $\beta' E'$ represent different loss functions. We define an *approximation problem* to be the problem of minimizing the expectation (expected risk) over A under p for a family of loss functions, defined on the same set ω and having all the same normalization Z over A. Those loss functions can be parametrized in the form $\beta' E' = -c_1 \ln p'(\omega) - c_0$ with p' a normalized probability density. We will call this a family of *approximation losses*. Then, for any parametrized family of $\beta' E'$ the probability related to the solution with minimal expected risk has minimal Kullback–Leibler distance to the actual averaging probability distribution.

Eq. (5.3.4) defines the true expected risk βE as the solution of a minimization problem. This corresponds to a *variational method* for calculating the true expected risk (see e.g. Balian, 1991, Neal & Hinton, 1993, Dayan, Hinton, Neal, & Zemel, 1995):

1. Besides the effects of not included parameters, the difference between the true expected risk βE and an expected risk $\beta^* E^*$, minimal in a parametrized subspace, has only contributions of second order in the parameters used for minimization. 2. The true expected risk is bounded by the approximated risk.[33]

5.4 Including human knowledge and other available preprocessors

A human interface for constructing fuzzy priors consists of two steps:

1. Subsymbolic level:
   i. Defining properties $C_i \in \mathcal{C}$, i.e. functions of a parametrization of f^0, as correlates to internal concepts $\tilde{C}_i$. For example, a property can be specified as a typical variant of an eye or a typical structure of an electrocardiogram, but as well by the output of another available (e.g. approximation) algorithm. The properties C_i are in some contexts also called linguistic variables (Zadeh, 1996).
   ii. Definition of possible combinations of properties, i.e. mappings $\mathcal{C}\times\mathcal{C}\to\mathcal{C}$ or linguistic functions, approximating internal mappings, i.e. the structure of concepts. Specifically, a linguistic 'not', 'and', 'or' can be related to some real valued extension of the logical NOT, AND, OR.
2. Symbolic level: Combining simple properties to create complex ones, i.e. applying linguistic functions (rules) to linguistic variables according to the communicated symbolic structure of the prior. For example the property being a face like object can be build up from properties of having two eyes, a nose and a mouth like object. Probabilistic rules are a special set of linguistic functions which depend on a (communicated) dependence structure of the variables.

Consider we want to construct a property C (deterministic question) of f^0, describing prior knowledge about images of faces, out of subproperties C_i by fuzzy operations. We can choose for example a logprior

$$L \propto -||C - T_C||^2 + c.$$

Also, the subproperties C_i can be deviation properties

$$C_i \propto -||C_i' - T_i||^2 + c_i.$$

For probabilistic questions one might wish to integrate over y_q with $p(y_q|q,f^0)$

$$C_q \propto -\int dy_q\, p(y_q|q,f^0)||y_q - T_{y_q}||^2 + c.$$

The templates T_C, T_i, T_{y_q} can be choosen as output of an available approximation $\hat{y}_q$, e.g. $T_{y_q} = \hat{y}_q$. The approximation $\hat{y}_q$ may be produced from a previously trained artificial neural network, or any other statistical approximator, as well as from an expert. A construction according to fuzzy methods can be as follows: Let $p(y|q,f^0)$ be the probability that the given image q is a

[32] Therefore, for example Jaynes, 1996, sees the two approaches as two different methods, applied in different situations: Maximum entropy to fix averages which are known without much computation, and Bayesian methods dealing with models to calculate the relevant average.

[33] Variational principles for minima (or maxima) have the advantage of giving such a bound, while variational principles related to saddle points (e.g. for complex functions), do not. On the other hand, for minima (maxima) all second order corrections have the same sign, while for saddle points second order contributions with different sign can average away (see for example, Lemm, 1995ab). Especially in highdimensional spaces this may considerably improve the approximation.

face (i.e. y=face). (That means, this is the classification and not the generation probability for faces.) Let T_{y_q} be the answer of some already available face detector. Also, several T_{y_q}, i.e. several answers or different approximation methods, can be included at the same time, according to their known or subjectively beliefed dependency structure. The dependency between approximators may be arbitrary including as special cases approximations which are independent or disjunct (only important for OR not AND).[34] The template T_{y_q} itself can be constructed according to fuzzy methods: Define (f^0–independent) q–templates T_q for q (Not y_q–templates T_{y_q} !). They may correspond to typical constituents like eyes, mouth, nose, and look like $\sum_i ||q_i - T_i^{eye}||^2$ with i denoting the pixel index. Define transformation of templates, including at least translation (represented by a transformation of the index i of the template). Combine the constituents, their possible variations and combinations with fuzzy AND, OR, NOT (which includes more complicated rules like XOR or IF $\cdots$ THEN) to obtain a final property T_{y_q}. This defines a fuzzy face classifier, incorporating human prior knowledge.

We now discuss two principal approaches to incorporate preprocessor information, like the fuzzy templates, into a subsequent algorithm.

5.4.1 Two coupling principles

The loss minimizing algorithm makes the decision $\hat{f}$. Such an optimizer must have an interface for relevant data, i.e. for the q^l and corresponding answers y. This allows to distinguish two principal variants to include the information of a preprocessor (See Fig.7):

1. Feeding the output of the preprocessor in the given interface for relevant data, including the entrance for the y which define the goal of learning. We will call this prior cascade and the preprocessor a data model generator (Fig.7 top).
2. Feeding the preprocessor output to extra input channels of the subsequent optimizer. We will call this case input cascade. Here the preprocessor has not necessarily to produce a data model. It can be implemented in two variants:
   a. asynchronously with the data D^l (Fig.7 middle),
   b. synchronously with the data D^l (Fig.7 bottom).

   q–variables which enter the loss function only indirectly through $\hat{f}(q)$, can then be skipped. This

[34]The problem of combination of different approximations of y_q (or more general of arbitrary available data) to get a better approximation of y_q is just a version of the usual statistical approximation problem. Thus, a great number of possible algorithms is available to deal with the problem. Some methods refer especially to the situation where the data are approximations of the same value (combination of experts/approximators/classifiers). Tree–like methods construct a local 'domain of responsibility' for each expert. (See for example the mixture of experts, (Jacobs, Jordan, Nowlan, & Hinton, 1991; Jordan, Jacobs, 1994).)

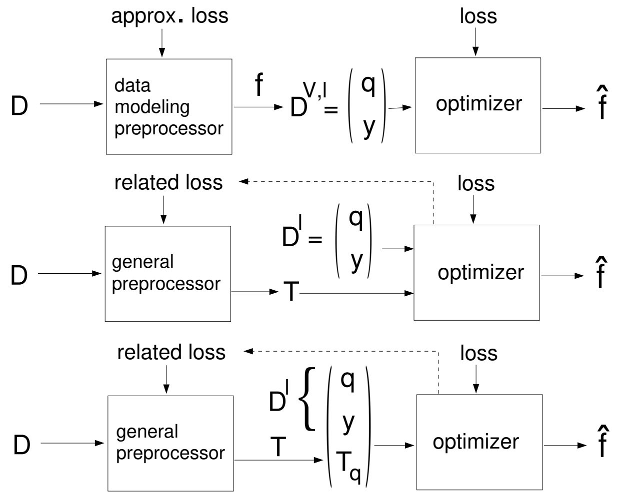

Figure 7: Top: Prior cascade (data modeling preprocessor). The preprocessor uses the data interface of the optimizer. So the interpretation of its information as (virtual) data is implicit. Thus, 1. the preprocessor must represent a data model, 2. the optimizer has not to be adapted. (approx. loss = approximation loss)
Middle: Input cascade (general preprocessor). The preprocessor uses not the whole interface for loss relevant data, i.e. at least not the entrance for y which are the variables for the loss function defining the goal of learning. It can, however, for loss which is not explicitly q dependent (but always indirect over $\hat{f}$), replace part or all of q. Thus, 1. the preprocessor does not need to represent a data model, therefore its loss can also be adapted according to the actual needs of the optimizer, 2. however, the optimizer must in general be adapted if the preprocessor adds information.
Bottom: Special case of an input cascade: The input q is changed, either by replacing q by preprocessor output (only if the loss is q independent, e.g. multilayer neural network) or by adding new dimensions (an extrem case is the support vector machine, with the preprocessing implicit in the kernel). Those additional dimensions can also be the output of a data modeling preprocessor and represent an approximation for y or for the optimal reaction $\hat{f}(q)$. (Which is the same in a pure approximation problem.) For a more detailed explanation see text.

allows to effectively replace q by preprocessor output.

We now discuss the two approaches in more detail.

5.4.2 Prior cascade (data modeling preprocessor)

In a prior cascade, a data modeling preprocessor intends to produce a (fuzzy) model f for the input of the optimizer, i.e. of the relevant data D^l. A Bayesian model is a special model formulated in terms of probabilities. Accepting its validity, the model can be used to create new virtual examples $D^{V,l}$ for relevant questions $q \in Q^l$. For the corresponding relevant data D^l a loss function is defined fixing their 'interpretation, for the optimizer. For virtual examples $D^{V,l}$ it is assumed that the same interpretation, i.e. loss function, applies, so they can use the q and the y entrance of the optimizer. Thus, the flexibility of this approach consists in the fact that the $D^{V,l}$ fit a given input format and a given interpretation, so they do not require adaption of the optimizer. Such a preprocessor may be seen as minimizing (implicitly or explicitly) an approximation loss for the data. This does not necessarily coincide with the loss used by the optimizer, which may, for example, include additional complexity penalties. Indeed, if the optimizer would have to minimize the same loss as the preprocessor, which would include having the same architectural restrictions, it cannot produce new results, and it is necessarily $f = \hat{f}$, as long as no new data are included. If new data are available the preprocessor generates, only part of the optimizer's input, weighted according to its prior probability. Approximation loss is related to any nonapproximation loss which depends on the relevant data.

Consider now a fuzzy preprocessor, like the fuzzy face detector we described in principle. There, new training examples can be generated sampling according to a term $C_q \propto -\int dq p(q) ||y_q(f^0) - T_{y_q}||^2 + c$ in the logprior (for probabilistic f^0 we integrate also over $p(y_q|q, f^0)$). The approximation T_{y_q} for $\hat{y}_q$ itself can be based on approximations implemented by virtual examples. Note, that virtual q can include questions which are not available as training questions, as long as a loss function is defined for them. For a face detector, those q can correspond to images which appear neither as faces nor as nonfaces, i.e. which are no 'natural' images ($= Q^l$). Such 'nonnatural' q can be images of faces with some subconcepts missing or exchanged, for example, one eye transformed to a noneye, e.g. to a square. Such a squared eye face is probably not in the set of 'natural' images Q^l but may be useful to teach the face detector the concept of an eye and its relevance for face detection. Including those examples near the border between faces and (the concept of, not the natural) nonfaces, specifies the generalization intended by the trainer.

The q–integral in the logprior can be treated in several ways:

1. the integration may be performed *analytically*, and the integral (infinite sum) replaced by an equivalent term, which is easier to evaluate,
2. by *virtual examples*, i.e. pairs (q, T_{y_q}), e.g. fuzzy classified (See e.g. Lin, Kung, & Lin, 1997). In case of many virtual examples this estimation may be called a numerical evaluation of the integral.
3. by *application adapted sampling*, i.e. virtual examples are (also) used during the application phase of the algorithm. That means, in a specific application situation q^l one includes newly sampled case–specific virtual examples, before answering to q^l. For example, one can evaluate the corresponding $T_{y_{q^l}}$ (if this can still influence the output and if available), and can also include $T_{y_{q'}}$ for other, e.g. 'similar', q'. We remark that this variant
   a. assumes the availability of the approximation T_{y_q} also during the application phase and not only during training,
   b. requires on–line '*application–adapted relearning*', because the newly, case specific generated examples have to be incorporated in the learning process. An trivial example of application specific on–line 'learning' is interpolating between different available answers, a less trivial example are neural networks which are retrained with application specific 'hints' $T_{y_{q^l}}$ before giving an answer to a specific q^l. Also, a higher level approximator can include local prior terms according to the output of a parallel working fuzzy face detector for a given image, preclassifing faces and nonfaces.
   c. is equivalent to the use of an *infinite* number of virtual examples, because for every q^l (i.e. arbitrarily many) new examples are drawn. That does not mean that a single prediction depends on an infinite number of virtual examples, but the definition of the whole learning machine, and therefore its expected performance.[35]

5.4.3 Input cascade (general preprocessing)

In an input cascade, a general preprocessor produces output T, for which no loss function, i.e. prewired interpretation, has to exist. This kind of information T cannot use the relevant data entrance of the optimizer. In cases, however, where the loss function is not explicitly q–independent (not $y(q)$– or $\hat{y}(q)$–independent!) the q–part of the data can be completely skipped and replaced. This does not include the y variables which define the value of the loss function and therefore the goal for the optimizer. Thus, the output of a preprocessor can replace part of the q or can be added as additional input to the optimizer, which has to be adapted to the new format of the input. This internal adaption defines the interpretation of the additional data.

[35] Again, infinity means nothing else than assuming to be always able to do something if needed, like in this case creating a template for every new input. That is how we defined the algorithm and if there are cases where we cannot create corresponding templates, then we are just not able to apply this algorithm, nevertheless we say the infinite data statement is true for the algorithm by definition. (definition = specifying possibilities of control).

In principle the total (information of the) previous approximator T could be added as new input dimension. However, the increase in complexity by irrelavant data tends to lower the performance in an input cascade (See below). Also a data modelling preprocessor can be used in an input cascade by adding available virtual data pairs to q, leaving y unaltered.

Especially, the information of the preprocessor may be feed in synchronously with every data pair y, q^l. The tranformed and often enlarged input space is also called feature space. Optimizing algorithms can often be adapted relatively easy for higher input dimension of q. Thus, a simple form of an input cascade or preprocessing is including part of the information T in every data pair (q, y). This can be the result $\hat{f}'$ of another optimizer, but also linear independent components of the vector q, or distances to prototypes obtained by unsupervised algorithms. We mentioned that in cases where the loss function is explicitly q–independent the q–part of the data can be completely skipped and replaced by T_q, so that the original data entrance formally fits. This is the standard case of an hierachical optimizer, like for example a neural network.

In practice one can for example add output T_{y_q} corresponding to the same (and maybe a few related) q, i.e. $q^l \rightarrow \{q^l, T_{y_q}\}$. (Also aspects of the learning history might be added, but one usually assume this to be well enough represented by the actual internal state of the optimizer). If T_{y_q} for a given q^l is not available, one may use a q where a similar T_{y_q} is expected, i.e. complete the template generator by adding another prior assumption. For example assuming smoothness with respect to a distance $||\cdot||$, one can use a q with $\text{argmin}_q ||q^l - q||$ ('nearest' q) or interpolate between some of the neighbors. One may encode that a specific value is not available, or its expected correctness. For added as well as for replaced input a function $\hat{f}$ of the augmented and not of the original input is learned. That means, the preprocessor also has to be available during application of $\hat{f}$.

For T to be of any help, the preprocessor, must deal with aspects related to the loss function, i.e. it must have (implicitly or explicitly) a loss function related to the loss function of the optimizer. For example, the preprocessor can explicitly use the same loss function, and represent therefore a previously trained optimizer. As the loss function of the preprocessor does not have to produce a data model, it can be adapted to the state of the optimizer. A typical example of an input cascade with feedback of the loss functions is a multilayer neural network. For all 'on–line' optimizers which do not use the complete available data set D^l at the same time (e.g. backpropagation), the actual parameter values (e.g. weights in a neural network) representing the memory for past data, can be seen as output of a preprocessor. As any architectural restrictions can be related to prior data, any restricted optimizer may be seen as result of some preprocessing.

In a prior cascade the next level algorithm treats the additional information as relevant data, i.e. the Bayesian interpretation of the template data is hardwired. In an input cascade the algorithm is free in its use of the data coming from the preprocessor. More precisely, the preprocessor data have a meaning implicitly implemented in the optimizer's algorithm and its architecture. Consider the extreme case, where the answer, the optimizer is looking for, is already included as an additional input dimension. This is only of any help if the algorithm of the optimizer has with a reasonable probability in its space $\hat{F}$ of possible hypotheses a function which is the projection of the whole input into this dimension. This includes that, even if all other original q–dimensions are deleted, the identity (which might be parametrized very complicately) must be part of the hypothesis space to allow the algorithm to find this 'simple' solution. Most practically used algorithms can probably find the identity or a projection easily, but one can also easily construct a model, which cannot. Formulated more tautological, additional information is helpful if it increases the probability of finding a better solution. (The meaning of 'better' can be specified in many variants.) The solution found by an algorithm must be part of a (learning history dependent) space $\hat{F}^c$ of hypotheses, which have been actually considered as possible solutions by the algorithm. A preprocessor can transform the input so that the transformed optimal solution $\hat{f}^*$ is (averaged over possible learning histories, if the templates are added before seing all q^l, and random variables of the algorithm) with high probability in the set of considered hypotheses $\hat{f}^* \subset \hat{F}^c$. For example, for algorithms for which the projection is easily learnable, adding templates for the output as additional input variable might be a good choice.

If we assume the space of considered hypotheses $\hat{F}^c$ to be bounded, with respect to some resources, then an input transformation effectively interchanges (maybe probabilisticly) some considered with nonconsidered hypotheses. Thus, changing the considered hypotheses by additional information, means changing one factor within a multiple OR (of all $\hat{f}^c$). Realizations of a multiple OR are usually very flat functions within the set $\hat{F}^c$ (e.g. constant in the deterministic case) with (more or less) sharp transition to zero at the boundary. Thus, an input cascade is expected to have a very flat prior being only effective at the cutting edge. As a (highly) multiple OR is difficult to implement, information (D) which seems meaningful, without being easily related to $p(f^0|D)$ can easier be included in an input cascade, (i.e. used for preprocessing) than in a prior cascade (i.e. data modeling). This is usually the case in nonapproximation problems when information is related to the optimal answer and not to the possible state (See also Section 8).

Notice, that in a prior cascade we can change the importance of priors by changing their weight factor. Because this is a parameter of a Bayesian model, there is usually no direct analogous parameter for the input cascade, so the importance of information is encoded implicitly in (the parameters describing) the algorithm. Assuming the Bayesian model to be correct, the prior cascade should give better results. In cases the Bayesian model is not correct, the input cascade has less bias, as it is similar to a prior cascade with smaller weights for the noncorrect priors. However, adding input variables makes the problem more complex by increasing the space of possible solutions. For example the VC dimension may increase, or the (algorithm specific, implicit) prior for the true state of nature might be smaller in an enlarged space. Alternatively, the complexity of the approximator may be reduced, for example by requireing a stronger smoothness in higher dimensions. So for example the support vector machine increases enourmously the input dimensions (feature space), while on the other hand their VC dimension is expected to stay approximately constant. Compensating the addition of new variables by restriction of the search space $\hat{F}$, is only successful if the restrictions reflect correct priors.[36] They should therefore be the result of a prior cascade, which suggests the possibility to include this information also without increasing the input dimension.

Thus, adding input variables, is only expected to help, if they contain enough information to compensate either for the higher dimensionality or for skipped variables in the problem. In contrary, for a prior cascade also including variables with only small effects should improve the performance, as long as those effects are correctly modelled. An input cascade implemented by adding (or replacing) components to the relevant data is an application adapted sampling method which requires on–line availability of the preprocessor producing T_{y_q}, and therefore corresponds for an infinite set Q^l, to a formally infinite amount of data.

These methods of an integration of available approximators, allow to use information contained in other learning systems, and are therefore general methods of knowledge transfer.

6 Decision problems

6.1 Definition

In this section we study questions conditioned on data. Assume we have to choose one question (alternative) $\hat{f}$ out of a set of questions $\hat{F}$[37]. We will now use the symbol l (actual loss) for the answers of $\hat{f}$ and assume that it contains all decision relevant information. Then a decision should only depend on the loss distributions of the various $\hat{f}$. We choose a minimum on the set $\hat{F}$ by defining a (risk) functional $r[p(\cdot|\hat{f}, f)]$ mapping probability densities of answers $p(l|\hat{f}, f(D))$ into a subset of the real numbers, bounded from below. A *decision problem* consists in finding the answer to a question q_r with

$$\hat{f}^* = q_r(f(D)) = \mathrm{argmin}_{\hat{f}\in\hat{F}} r[p(\cdot\,|\hat{f}, f)].$$

If we want to emphasize the data dependency of the decision we speak of a *learning problem*. Approximation and classification problems are special cases of decision problems.

We will now have a closer look to the questions $\hat{f}$. Using $p(y^q|q, f) = \int df^0 p(f^0|f)p(y^q|q, f^0)$ we will from now on write the formulas for states of knowledge f instead for pure states f^0. According to eq.(3) there exist some q so that the probability of suffering loss l in a given state of knowledge f can be written

$$p(l|f, \hat{f}) = \int dq \int dy \int dz\, p(q|y_c, z_c, \hat{f})$$

$$\times p(y|q, f)p(z|q, y, \hat{f})\delta(l(q, y, z, \hat{f}) - l).$$

This implies that noise variables z of different questions are independent and one specific realization of y according to $p(y|q, f^0)$ can only appear multiple times within one question. The situation can be represented by influence diagrams with decision and value nodes (Pearl, 1988) We can define the decision relevant basis set $X = X^l$ by the set Q^l of test data $q^l = q$. Then this formula is in the form usually given in Bayesian decision theory. Starting from an factorial state with respect to X^l structural information relates generalized questions q^D from the training data to the different $q = x$ and all information enabling generalization has to come from nonlocal q^D.

The explicit $\hat{f}$–dependence comes from three factors which are

i. the action z (including noise) producing device $p(z|q, y, \hat{f})$,

ii. the defining (loss) function $l(q, y, z, \hat{f})$,

iii. the test set generator $p(q|y_c, z_c, \hat{f})$.

One usually uses a formulation where the components in 2. and 3. are chosen to be explicitly $\hat{f}$–independent.

1. a) $p(q|y_c, z_c, \hat{f}) = p(q|y_c, z_c)$, or
   b) $p(q|y_c, z_c, \hat{f}) = p(q|y_c)$ (fair), or
   c) $p(q|y_c, z_c, \hat{f}) = p(q)$ (static),
2. $l(q, y, z, \hat{f}) = l(q, y, z)$.

Remark:

1. We always can fulfill conditions 1 a) and 2 by introducing additional $\hat{f}$–dependent variables z (and a zero dimensional x_1) without effectively changing the model, as $\hat{f}$ has always to be parametrized. Therefore we defined $\hat{f}$ in Section 2 by $p(z|q, y, \hat{f})$. To fulfill the stronger condition $p(q|y_c, z_c, \hat{f}) = p(q|y_c)$ also all random processes have to be attributed to the state by including z_c into y_c, and for the strongest condition $p(q|y_c, z_c, \hat{f}) = p(q)$ by including y_c and z_c into q.

[36] In the support vector machine, the restrictions imposed by constructing an optimal hyperplanes in the feature space, can, choosing the appropriate kernels, lead to arbitrary restrictions in the original input space. Thus, the selection of the kernel includes the necessary prior knowledge. In the extreme case were the optimal solution (a one dimensional binary variable in binary classification) is presented as template, a feature space with the dimension of that solution is sufficient to find the optimal hyperplane.

[37] We use the letter $\hat{f}$ instead of q for this set of questions to symmetrize the further notation. This corresponds to an interpretion of the selected question as action state $\hat{f}$ in analogy to the model state f. The selected action state $\hat{f} = \hat{f}(f)$ can be seen as reaction (e.g. approximation) to the state of knowledge f.

2. This covers all cases of function approximation and pattern classification.

We will call a decision problem *fair* if condition 1b is fulfilled so different action states $\hat{f}$ are compared in the same situations q independent of their own previous outcomes, and *static* if condition 1c is fulfilled, and $p(q)$ does not depend on any other outcomes.

If interested in the expectation of l the remaining z can for static problems be integrated out defining an integrated loss function $\tilde{l}$

$$\tilde{l}(q, y, \hat{f}) = \int dz p(z|q, y, \hat{f}) l(q, y, z, \hat{f})$$

being a deterministic function of $\hat{f}$.

6.2 Parallel decision problems

If we assume $p(z|q, y, \hat{f})$ to be y–independent this can be seen as a device producing answer (action) z in situation q when the model (of nature) in state f produces y. Thus, $\hat{f}$ could be interpreted as (action) state of a second, independent model of actions capable in producing answers to the same questions as the original model (of nature). We will now write $\hat{y}$ for those z which are produced independent of y. This independence can also be seen as definition of the q as that part of the visible variables (q, y) which can influence the answer $\hat{y}$. For example, the answer of the model y can be assumed to be available for the device $\hat{f}$ only after the action $\hat{y}$ is produced. Now let us define a decision problem which compares two models in states f and $\hat{f}$ answering to the same questions q under equal conditions. This can be seen as comparing the action state $\hat{f}$ with the state of nature f in situations q according to the criteria $l(q, y, \hat{y})$.

If we want to model a causal dependence of the action $\hat{y}$ from 'previous' y, we choose the general formulation as

PM: a *parallel decision problem with memory*, defined as a model with $p(z|q, y, \hat{f}) = p(z|q, y_c, \hat{f})$. with the causal structure y_c being the same as in $p(q|y_c, \hat{z}_c)$. Then we write $\hat{y}$ for z, i.e. $p(\hat{y}|q, y_c, \hat{f})$. (The notation allows dependencies within the components of $\hat{y}$.) Thus, the action producing device outputs $\hat{y}$ in action state $\hat{f}$ with its components $\hat{y}_i$ independent of the corresponding answer component y_i of the model of nature f. However, it can use the values of previously determined components of y and choose its actions according to its 'success' in the past.

We call a decision problem $\hat{F}$

P : a *parallel decision problem without memory* (or simply a parallel decision problem) if, writing $\hat{y}$ for z, the action model produces $\hat{y}$ in action state $\hat{f}$ independent of the answer y of the model of nature f: $p(\hat{y}|q, y, \hat{f}) = p(\hat{y}|q, \hat{f})$ (Dependencies within components of $\hat{y}$ are allowed).

This leads to a loss probability, written for the more general case:

$$p(l|f, \hat{f}) = \int dq \int dy \int d\hat{y}\, p(q|y_c, \hat{y}_c)$$

$$\times p(y|q, f) p(\hat{y}|q, y_c, \hat{f}) \delta(l(q, y, \hat{y}) - l),$$

Note the difference between not yet seen (expected) test data $\hat{D}^l = (q, y, \hat{y})$, which are not yet determined and correspond to the integration variables, and the known (training) data D, which determines our actual state of knowledge $f = f(D)$ and do not appear explicitly in the above formula. Parallel decision problems with memory show an asymmetry between f and $\hat{f}$ as we allow a dependence of $\hat{y}$ from past values of y while we defined y only to be depending on the question q and pure state f^0 and not from $\hat{y}$. All $\hat{y}$–dependency of real world measurements $p(y|q, f^0)$ by definition has to come from changing the question according to $p(q|y_c, \hat{y}_c)$ or active changes of f^0 by $\hat{y}$ which we do not allow. We can parallel the treatment of $\hat{f}$ and f, by introducing a set of y–independent basis action states $\hat{f}^0$ and an 'algorithm' $p(\hat{f}^0|q, y_c, \hat{y}_c, \hat{D})$. Here we included possible additional data $\hat{D}$ in the formalism, which also determine the available action states $\hat{f}$, and we can write

$$p(\hat{y}|q, y_c, \hat{f}, \hat{D}) = \int d\hat{f}^0 p(\hat{y}|q, \hat{f}^0) p(\hat{f}^0|q, y_c, \hat{y}_c, \hat{D}).$$

For $p(\hat{D}|D) = p(\hat{D})$ we can determine $\hat{f}$ before the training begins, otherwise, the set $\hat{F}$ of available choices has also to be updated with the training data D.

Especially in approximation problems we may wish to include the structure of the q into the action $\hat{y}$–producing devices. For the set of $\hat{f}^0$, which in a parallel decision problem without memory already are the $\hat{f}$, we can then define an action model $p(y^x|x, \hat{f}^0)$ for the basis questions x use for a question

$$p(y|q, f) = \int dx \int dy^x \int dz\, p(x|y_c^x, z_c, q)$$

$$\times p(y^x|x, f) p(z|x, y^x, q) \delta(q(x, y^x, z) - y).$$

an action state

$$p(\hat{y}|q, \hat{f}^0) = \int dx \int dy^x \int dz\, p(x|y_c^x z_c, q)$$

$$\times p(y^x|x, \hat{f}^0) p(z|x, y^x, q) \delta(\hat{q}(x, y^x, z) - \hat{y}).$$

Note that also a $\hat{f}$–independent loss function can penalize complexity or requirement of resources of the $\hat{f}$ simply by including corresponding variables as components into $\hat{y}$.

While the $\hat{f}$ define the possible alternatives of loss distributions, the decision has to be made with respect to some risk functional r applied to this distributions. Choosing, for example, in the case of a real loss function for parallel decision problems the expectation as functional r we have to minimize the 'expectation risk' or expected risk

$$r(f, \hat{f}) = r[p(\cdot\,|f, \hat{f})] = \int dl\, l\, p(l|f, \hat{f})$$

$$= \int dq \int dy \int d\hat{y}\, p(q|y_c, \hat{y}_c) p(y|q, f) p(\hat{y}|q, y_c, \hat{f}) l(q, y, \hat{y}).$$

Density estimations are a special form of parallel decision proplems having only one q with the meaning 'get next data'. Choosing an integrated loss which can for every $\hat{f}$ be interpreted as a logprobability, i.e. $\tilde{l}(y,\hat{f}) = \ln p(y|\hat{f}) + c$ this gives

$$r(f,\hat{f}) = \int df^0 \int dy\, p(f^0|y^D)p(y|f^0)\ln p(y|\hat{f}).$$

One may well include other $\hat{f}$–specific aspects in the loss function, like $\hat{f}$–specific complexity costs, e.g. a term $\int dy\, p(y|\hat{f}) \ln p(y|\hat{f})$ related to the encoding costs of y given$\hat{f}$ (Rissanen, 1989). Sometimes the term 'unsupervised' learning is used for such problems, including problems which are defined by algorithms and do not explicitly refer to a set of parametrized models $p(y|f)$, like e.g. self–organizing maps (see e.g. Kohonen, 1995). This term might be misleading, because all variables y enter the (explicit or implicit) loss function, and are therefore 'supervised' variables. The reason for using nevertheless the word 'unsupervised' is, that density estimation is often applied to variables, which are used in a second step as q in a problem, where the loss function does not depend on q but on y. Indeed, the determination of $p(q)$ can in our formulation always be seen as a preprocessing step, because we explicitly require $p(q)$ to be independent of f^0.

To adapt to new data, parallel decision problems require an inversion to obtain

$$p(f^0|y^D,q^D) = \frac{p(y^D|q^D,f^0)p(f^0)}{p(y^D|q^D)}.$$

While this corresponds to an interchange of the roles of the variables y^D and f^0, we will in the next subsection refer to another inversion: The interchange of y with q.

6.3 Inverse decision problems

Here we will discuss *analysis by synthesis*, where an action device $p(z|y,\hat{f})$ is 'analysed', i.e. its corresponding loss minimized, under a synthesis model $p(y|q,f)$ for y, i.e. the question for $\hat{f}$ instead for its answer z. Often, the variable describing the desired output of the action device is easier to choose as condition q to build possible models $p(y|q,f^0)$ of nature, than its input variable. Consider the following example: Let y^a have the values 'face' and 'nonface' and q^a be corresponding images of faces and nonfaces. The task is to build a face detector, i.e. a device $p(\hat{y}^a|q^a,f)$ which outputs an approximation $\hat{y}^a$ of y^a for a given image q^a. That would mean we need a model $p(y^a|q^a,f)$, i.e describe for a given image possible probability distributions being a face. This might be done using fuzzy priors, but might not be very reliable, as the probability for a given image to be a face is related to the kind of nonface objects included in y^a. Every change in the set of nonfaces, would require an update of the fuzzy prior. Much more natural is it — as probably more related to the subjective process of generating priors by contemplating about typical features of faces — to approximate the inverse probability of images q^a given it is a face y^a. This can be done (as well as for the inverse direction) by a fuzzy decomposition of the image q^a into feature components $q^{a\prime}$ (e.g. eye template) according to the methods we discussed earlier. Thus, we built a generative model for images given faces. Also priors within the class of faces can now be formulated, independent of the nonface objects, in terms of face generative parameters, e.g. specific individuums, illumination conditions, rotation angels, emotional expressions. Also, the distribution for relevant questions $p(q^a)$ for the low dimensional variable face/nonface is more easily adaptable and measurable than for very highdimensional images. For example, one can measure or control the average number of people passing a camera (i.e. $p(q^a)$) much easier than to change or even approximate the probability of the corresponding images $p(y^a)$.

However, according to our convention denoting by q the generalized questions to nature and not the input to the action producing device, we have to exchange the letters in the notation and write $q = y^a$ for the face variable and $y = q^a$ for the image. Then the face detector $p(\hat{q}|y,f)$ gives an approximate classification $\hat{q}$ for the variable q.

We study the situation, where a action device inverts an available generative model, now more formally and define an

IM: *inverse decision problem with memory* as a model with $p(z|q,y,\hat{f}) = p(z|y,q_c\hat{f})$, i.e. where the the action model produces answers $\hat{q}$ depending not on the index of a generalized question q but on their answer y. In this situation we will use for z the notation $\hat{q}$. Dependencies within components of $\hat{q}$ are allowed.

In the same way we define a

I: *inverse decision problem without memory* as a model with $p(\hat{q}|q,y,\hat{f}) = p(\hat{q}|y,\hat{f})$, with $z = \hat{q}$ i.e. where the the action model produces answers $\hat{q}$ dependent on y, however independent of its history. Dependencies within components of $\hat{q}$ are allowed.

Because it also seems quite natural to work with variables representing input and output of the action producing device we will call

AR : a problem to be formulated in *action representation*, if the action producing device is written in a form $p(z|q,y,\hat{f}) = p(\hat{y}^a|q^a,\hat{f}) = p(\hat{y}^a|q^a,y^a,\hat{f})$. This defines (action) output $\hat{y}^a = z$, requires the i–component of the (action) input q_i^a to include at least that part of q_i, y_i on which z_i depends. The remaining variables are called $y^a = \{q,y\} \setminus q^a$. Analogously, we can define an action representation with memory to be of the form $p(z|q,y,\hat{f}) = p(\hat{y}^a|q^a,y_c^a\hat{f}) = p(\hat{y}^a|q^a,y^a,\hat{f})$.

MR : We will call our original formulation with q representing questions and y answers a *measurement representation*.

The following table summarizes the definitions, our convention used in the measurement notation (MR) and the action representation (AR).

Decision problems

$p(z\|q,y,\hat{f})$	parallel	inverse
without memory	$p(z\|q,f)$	$p(z\|y,f)$
MR:	$p(\hat{y}\|q,f)$	$p(\hat{q}\|y,f)$
AR:	$p(\hat{y}^a\|q^a,\hat{f})$	$p(\hat{y}^a\|q^a,\hat{f})$
with memory	$p(z\|q,y_c,\hat{f})$	$p(z\|y,q_c,\hat{f})$
MR:	$p(\hat{y}\|q,y_c,\hat{f})$	$p(\hat{q}\|y,q_c,\hat{f})$
AR:	$p(\hat{y}^a\|q^a,y^a_c,\hat{f})$	$p(\hat{y}^a\|q^a,y^a_c,\hat{f})$

Now we show that every inverse model looks 'pseudo parallel' in action representation without being necessarily equivalent to a parallel decision problem. The reason is that the roles of q and y are not freely exchangeable in a decision model. We defined a model to be completely determined by q, y and f, the f–independent part of variables q, i.e. $p(q|f)=p(q)$, and all f–dependent variables y. That means, the distribution of q is always known, and can only depend on a state of nature over already observed data, i.e. $p(q|y_c,\hat{q}_c,f)=p(q|y_c,\hat{q}_c)$ independent of f. Then the decomposition $p(y|q,f)=\int df^0 p(y|q,f^0)p(f^0|f)$ does not need to include a factor $p(q|f^0)$, and the joint probability $p(q,y|\hat{q}_c f)$ factorizes into a f–dependent and a f–independent factor

$$p(q,y|\hat{q}_c f)=(q|y_c,\hat{q}_c)p(y|q,f). \tag{12}$$

Accordingly, every problem in action representation can be written in at least a 'pseudo parallel' form. For the example of an inverse decision problem we can factorize the probability in the action representation

$$\begin{aligned} p(q,y|\hat{q}_c,f) &= p(y|q_c,\hat{q}_c,f)p(q|y,\hat{q}_c,f) \\ &= p(q^a|y^a_c,\hat{y}^a_c,f)p(y^a|q^a,\hat{y}^a_c,f) \end{aligned} \tag{13}$$

We remind, that the first line has to be read as $p(y_1|f)p(q_1|y_1,f)p(y_2|y_1,q_1,\hat{q}_1,f)p(q_2|y_2,y_1,q_1,\hat{q}_1,f)\cdots$, where $p(q_2|y_2,y_1,q_1,\hat{q}_1,f)$ can not necessarily be simplified to $p(q_2|y_2,f)$ if $p(q|y_c,\hat{q}_c)\neq p(q)$, i.e. for a nonstatic model. The factors of this inverse picture are related to the original model by (See for example Saul, Jaakola, Jordan (1996) for a variational method for inversions in sigmoid belief networks.)

$$p(q|y,\hat{q}_c,f)=\frac{p(q|y_c,\hat{q}_c)p(y|q,f)}{p(y|q_c,\hat{q}_c,f)},$$

and

$$p(y|q_c,\hat{q}_c,f)=\int_c dq\, p(q|y_c,\hat{q}_c)p(y|q,f).$$

where the symbol $\int_c$ denotes a 'causal' or 'conditional' integral defined as $\int_c dq\,p(q)=\prod_i\int dq_i p(q_i|q_{c(i)})$ (not $\int\prod_i dq_i$ p(q) !) with $q_{c(i)}=\{q_j, j<i\}$, so that for every factor the $q_{c(i)}$–dependency remains. Notice, that this is not the inversion necessary to obtain $p(f^0|D)$ from $p(y|q,f^0)$. For a static model, i.e. for $p(q|y_c,\hat{q}_c)=p(q)$, the joint probability factorizes also in the inverse notation, $p(q_i)p(y_i|q_i,f)=p(y_i|f)p(q_i|y_i,f)$, with the individual factors $p(y_i|f)$ still being state dependent. If we try to exchange the role of q and y we find in eq.(13) compared to eq.(12):

1. History dependent questions. To relate y to a question in a measurement representation (or f as a state) f and y must completely determine $p(q|y,\hat{q}_c,f)=p(q|y,f)$, like it is the case in the inverse of a static model. In other cases one can define a set of history (i.e. $\hat{q}_c$, q_c, y_c) dependent questions y_i' (or states) labeled by $\{y_i,y_c(i),q_c(i),\hat{q}_c(i)\}$.

2. State dependent questions. According to our definition of questions y can only be interpreted as question (for nature not for $\hat{f}$!) if state independent, i.e. $p(y|q_c,\hat{q}_c,f)=p(y|q_c,\hat{q}_c)$. In other cases the distribution of relevant 'questions' y in the inverse picture is state dependent, i.e. not exactly known, and they are therefore effectiveley part of the unknown variables, which we call answers. It is always possible to enlarge F^0 to include q and write $p(q|F_q)=p(q|D^q)=\int df^0_q p(f^0_q|D^q)p(q|f^0_q)$ depending on data D^q which have been used to determine the probability distribution. Thus, the space $F^0=F^0_q\otimes F^0_{y|q}$ also contains hypotheses about different possible parametrizations $p(q|f^0)$. However as long as there is a 'factorial border' between f^0_q and $f^0_{y|q}$, i.e. $p(f^0)=p(f^0_q)p(f^0_{y,q})$ factorizes, the problems for q and y given q (not y alone) are independent. As soon as they are dependent, we must perform the decomposition into pure $f^0_{y|q}$ states q–dependent, i.e. we must treat q as answer. For example, when not only $p(y|q,f)$ but also $p(q)$ is sampled when obtaining data, this is relevant for determining $p(q)$, in any case when it is not yet completely known. In this case conceptually both q and y are part of the answers, let us say $(q,y)=y'$, and the set of q' is reduced to one element with the meaning: 'Get next data pair'. However, when the hypothesis spaces factorize and the data do not induce new correlations, both problems can be treated separately, and q can be used as question for the y problem.

We finally notate the probability to suffer loss l in the measurement representation

$$p(l|f,\hat{f})=\int dq\int d\hat{q}\int dy\, p(q|y_c,\hat{q}_c)p(y|q,f)$$
$$\times p(\hat{q}|y,\hat{q}_c,\hat{f})\delta(l(q,\hat{q},y)-l)$$

with expected risk

$$r(f,\hat{f})=\int dq\int d\hat{q}\int dy\, p(q|y_c,\hat{q}_c)p(y|q,f)p(\hat{q}|y,\hat{q}_c\hat{f})l(q,\hat{q},y).$$

and in action representation

$$p(l|f,\hat{f})=\int dq^a\int dy^a\int d\hat{y}^a\, p(q^a|y^a_c,\hat{y}^a_c,f)p(y^a|q^a,\hat{y}^a_c,f)$$
$$\times p(\hat{y}^a|q^a,\hat{y}^a_c,\hat{f})\delta(l(q^a,\hat{y}^a,y^a)-l).$$

with expected risk

$$r(f,\hat{f})=\int dy^a\int d\hat{y}^a\int dq^a\, p(q^a|y^a_c,\hat{y}^a_c,f)p(y^a|q^a,\hat{y}^a,f)$$
$$\times p(\hat{y}^a|q^a,\hat{y}^a_c,\hat{f})l(q^a,y^a,\hat{y}^a).$$

We conclude: An inverse decision problem cannot necessarily be formulated as a parallel decision problem, because its action representation yields in general state dependent q^a.

We will see later that for certain static decision problems a one step maximum posterior approximation is sufficient only for approximation problems, i.e. when $p(y|q,f)$ itself shall be approximated. Then this step has numerically the form of an empirical risk minimization. Obviously, approximating the inverse probability $p(q|y,f)$ is not an approximation problem for $p(y|q,f)$, (except when $p(y|q,f) = p(q|y,f)$) and we have just seen that it cannot be transformed into one. Thus, for inverse problems a maximum posterior approximation should in general be completed by a second step. This two step procedure has for inverse problems the following form: In a first step $p(y|q,f)$ is approximated by $p(\hat{y}|q,\hat{f}_{app})$. In the second step the optimal approximating action device $\hat{f}_{app}$ is identified with a (fixed) state of knowledge $f^* = \hat{f}_{app}$, and accordingly $\hat{y}$ with y. Then $p(\hat{q}|y,\hat{f})$ is chosen to minimize the loss for the approximating device, i.e. for given state f^*. However, for a full Bayesian treatment, and for the justification of an empirical risk minimization the difference between parallel and inverse problems is in no way conceptually important.

6.4 Algorithms

Often the process of calculating the answer to q_r, that is the optimal learning or decision algorithm, is to difficult to be actually performed. Thus, one has to use a simplified *decision or learning algorithm*, i.e. another question $\hat{q}_r = \hat{a}$, to produce an answer $\hat{f} = \hat{a}(f(D))$. We discuss the following for parallel decision problems, but it applies analogously to inverse and general fair decision problems. If we have to decide between several available algorithms this corresponds to another (higher level) decision problem[38], with the $\hat{y}$ producing device with $p(\hat{y}|q,\hat{f})$ replaced by a $\hat{f}$–producing device $\hat{a}$ with $p(\hat{f}|q,y_c,\hat{y}_c,\hat{a})$, and we have to define the $\hat{a}$ dependency of the loss which could be formulated by extending the set of relevant questions $Q^l \to Q^{l,a}$ to include loss relevant aspects. If no algorithm specific aspects have to be included, only an additional $\hat{f}$–integration has to be performed. The same is essentially true if the algorithm specific loss can be represented by a q, y, $\hat{f}$–independent constant. On the other hand, loss related aspects which depend on more than one of the original relevant questions q^l also depend on the correlations between q^l (for which the original loss, and thus also the risk, is insensitive) and can generate a much more complicated set of relevant questions and associated loss l for the new problem.[39] This expression corresponds to the expression $p(\hat{f}^0|q,y_c,\hat{y}_c,\hat{D})$ we discussed for parallel decision problems. Thus, from this point of view the $\hat{f}$ corresponds to possible pure action states $\hat{f}^0$, and the algorithm $\hat{a}$ to the available data $\hat{D}$. Interestingly, one notation only refers to data, and assumes the algorithm to be implicit, while the other refers to the algorithm and assumes the data to be implicit. Thus, we have a parallel decision problem with memory where the $\hat{f}$ are part of the internal noise variables $\hat{y}_{\hat{a}} = (\hat{y},\hat{f})$. The q^l are related to a loss function l which can depend on $\hat{a}$ and $q_r(\hat{a})$ chooses the $\hat{a}$ which minimizes a functional r of the loss distribution

$$p(l|\hat{a}f) = \int d\hat{f} \int dq \int dy \int d\hat{y}\, p(q|y_c,\hat{y}_c,\hat{f}_c)p(y|q,f)$$

$$\times p(\hat{y}|q,\hat{f})p(\hat{f}|q,y_c,\hat{y}_c,\hat{a},f)\delta(l(q,y,\hat{y},\hat{f},\hat{a})-l).$$

A loss $l(q,y,\hat{y},\hat{f})$ can be chosen $\hat{f}$–independent by including additional variables $\hat{y} = \hat{y}(\hat{f})$ if necessary. To include algorithmic specific aspects without using an explicit algorithm dependent loss function we can, for example, include $\hat{a}$–dependent internal variables $\hat{y}_{\hat{a}}$ in the loss function produced with $p(\hat{y}_{\hat{a}}|q,\hat{a})$ measuring for algorithm specific variables like their requirement of resources like calculation time, memory and other aspects.

An algorithm is defined by its $p(\hat{f}|q,y_c,\hat{y}_c,f,\hat{a})$. It produces an answer $\hat{f}$ which should at least be depending on its state of knowledge f, that is its prior probabilities and training data. We always assume its knowledge of the $\hat{f}$, i.e. of the $p(\hat{y}|q,\hat{f})$. The dependence of $p(\hat{f}|q,y_c,\hat{y}_c,f,\hat{a})$ from y_c and $\hat{y}_c$ allows the algorithm to adapt, i.e. learn, during the test phase. In that case the loss evaluates the learning curve of algorithms. One can allow the choice of $\hat{f}$ to be dependent from the test question q being equivalent to enlarging the space of available $\hat{f}$. The variable $\hat{f}$ can be a vector, for example, if a decision is required after presenting part of q. In situations with $p(q|y_c,\hat{y}_c,\hat{f}_c) = p(q|y_c,\hat{y}_c)$ and $l(q,y,\hat{y},\hat{f}) = l(q,y,\hat{y})$ algorithms can be seen as improved f–dependent $\hat{y}$–producing devices with

$$p(\hat{y}|q,y_c,f,\hat{a}) = \int d\hat{f} p(\hat{y}|q,\hat{f})p(\hat{f}|q,y_c,\hat{y}_c,f,\hat{a}).$$

Choosing for real loss r as the expectation functional we are looking for the $\hat{a}$ which minimizes

$$r(f,\hat{a}) = r[p(\cdot\,|f,\hat{a})] = \int dl\, l\, p(l|f,\hat{a})$$

$$= \int d\hat{f} \int dq \int dy \int d\hat{y}\, p(q|y_c,\hat{y}_c,\hat{f}_c)p(y|q,f)p(\hat{y}|q,\hat{f})$$

$$\times\ p(\hat{f}|q,y_c,\hat{y}_c,f,\hat{a})l(q,y,\hat{y},\hat{f},\hat{a}).$$

[38]This problem of finding an optimal algorithm can be a much more complicated problem than finding the optimal decision. So comparison of algorithms can be done in only a few number of (simple enough) cases (See for example, Watkin, Rau, & Biehl, 1993 and references therein for a review). When also using approximations for the high level problem all the same problems appear on the higher level again. But if we assume the decision problems of the different levels to be similar we could at least check for consistency: Does approximation A on level i produce approximation A on level $i-1$.

[39]Compare the distinction between l and L in (Haussler, 1995).

6.5 Minimization

Solving decision problems requires minimization algorithms. In general, also the minimization problem can be seen as a learning problem: Given data of function values and information about the function (e.g. differentiable, symmetric) give the position of the minimum. In so far our discussion of learning and prior information also applies to optimization. Usually, minimization algorithms perform active queries for local data (for example, taking a new data point in direction of the gradient) to improve a given guess for the location of the minimum and proceed iteratively until a certain convergence criterion is fulfilled. An iteration procedure can be written

$$\hat{f}^{i+1} = G(i, D_i, \hat{f}^i) = G_i(\hat{f}^i),$$

where $\hat{f}^i$ denotes the current guess for the location of the minimum at step i and D_i (which for the sake of simplicity will be skipped from now on from the notation) the new and accumulated previous data ($D_i = (r(f, \hat{f}^j), \hat{f}^j)\}_{j \le i}$). Besides data points G depends on prior knowledge about the function (in our case $r(f, \hat{f})$). The past iterations provide data for the minimization problem, and D_i indicates that G changes with the amount of available data, i.e. the number of iterations. Thus, one can say G is trained on the available data. This at least implicitly assumes prior knowledge which allows available data to carry information about other parts of the function.

In general, we can allow any reparametrizations T (of the $\hat{f}$, in our case) as long as T is *locally injective at the locations of global minima* $\hat{f}^*$, i.e. $T^{-1}(T(\hat{f}^*)) = \hat{f}^*$. Reparametrizations may be not injective for points which are no minima, as those can be excluded from further search. But also under *globally bijective reparametrizations* minimization problems can look quite different for the transformed variables. Reparametrisations can be nonlinear, differentiable or nondifferentiable, or linear tranformations for vectors or functions (like $\hat{f}$) i.e. a change of the representing basis if $\hat{F}$ is a vector or Hilbert space, or even a random permutation of the function values. Transformations can create arbitrary neighborhood relations, so the minimization problem can become trivial, like for example with function values ordered monotonically (to find such a permutation, however, the minimum problem has to be solved), or arbitrary hard (nonsmooth, random).

If we define R to be the space of all possible functions $r(f, \hat{f})$ (for fixed f) and give the values of r a linear struture, we can expand any r, at least formally, into basis functions $r = \sum_i a_i b_i(\hat{f})$. Thus any function $r'(\hat{f}) = \sum_i a'_i b_i(\hat{f})$ on $\hat{F}$ taking values in the linear range of $r(f, \hat{f})$, for example $r'(\hat{f}) = r(T(\hat{f}))$, is by construction of R a linear transformation $A(\hat{f}', \hat{f})$ of $r(\hat{f})$, i.e. $r' = Ar$, which is defined by the mapping of the coefficient vectors with $a' = Aa$.[40] The resulting dimensionality may however very soon be intractable huge, e.g. for infinite $\hat{F}$ the resulting space has infinite dimension. To be able to use such a space for calculation, it must be restricted. In Hilbert spaces for example only functions are allowed for which the expansion in a basis can at least be arbitrarily well approximated in some norm by an (arbitrary large, but) finite number of basis functions.

Reparametrizations always change only the arguments not the function (risk) values itself. There is also the possibility to change the function values without changing the location of the minimum. In general, for a function f (in our case the risk, i.e. f does not denote a state here) the positions x^* ($=\hat{f}^*$) of global minimima $f(x^*)$ ($=r(f, \hat{f}^*)$), defined by $y^* = f(x^*) \le y = f(x)$, $\forall x$, do not change under transformations h which obey $y^* \le y \Leftrightarrow h(y^*) \le h(y), \forall y$. We will call such transformations h *strictly monotonically increasing relative to* y^*. (Analogously, we define strictly monotonically decreasing relative to y^* $y^* \le y \Leftrightarrow h(y^*) \ge h(y), \forall y$.) As we do not know $y^* = f(x^*)$ in advance we have to require this relative to all values which are possible candidates for a global minima.

Minimization methods can only return a minimum within some selected finite (sub)set of considered function values. Local methods (see below) for example find local minima. Locations of minima are invariant for all subsets under *strictly monotonically increasing transformations* h defined by $y < y' \Rightarrow h(y) < h(y'), \forall y, y'$, equivalent to strictly monotonically increasing relative to all y, i.e. $y \le y' \Leftrightarrow h(y) \le h(y'), \forall y, y'$. Analogously defined strictly monotonically decreasing transformations change a minimum to a maximum.

There is a large variety of methods and concepts available for minimization, with many possibilities of combination. (For a discussion with respect to neural networks see for example Golden, 1996. In Section 9 optimization methods are needed for maximizing the posterior probability.) The fact that the following principles of minimization algorithms can be applied to transformed (strictly monotonically increasing relative to global minima) and reparametrized (bijective at locations of global minima) problems, makes clear that in general a large number of possibilities can exist to attack a specific problem.

The simplest method, which does not refer to any dependencies of function values, is

1. an *unadapted search* (stochastic, predetermined deterministic, or exhaustive), i.e. with data D_i collected independent of function (risk) values for $\hat{f}$. One iteration step consists in sampling of a new data point, and G compares this point with the current guess $\hat{f}^i$. The sampling distribution does not depend on previous iteration steps.

Deterministic algorithms (deterministic G) are special cases of stochastic algorithms (probabilistic G). Prior information enters G if the sampling distribution at step i

[40] See for example generalized additive models, where interaction terms are added (Hastie & Tibshirani, 1990), the comments in Minski-Papert-1990 (the new edition of the 1969 book) about the general applicability of (linear) perceptrons, the support vector machine, where linear relations in the feature space correspond to nonlinear relations in the input space (Vapnik, 1995), and for a general approch, Smola & Schölkopf, 1997.

depends on D_i. The most common case are local dependencies yielding

2. *local iterative methods.* For smooth functions small values are in the neighborhood of other small values and therefore the search (sampling probability) is concentrated near the current guess of the minima. As differentiable functions have a gradient equal to zero at a minimum this allows to search for those zeros which are possible locations of minima. Sometimes stationary points can be found analytically, but usually nonlinear equations require iterative solutions. (The term 'analytical' is commonly used for solutions where the iteration can be done easily, like determining a certain square root or the numerical evaluation of constants like e or π.) Technically, the iteration is often implemented in the form of a relaxation method (see Section9), which includes common algorithms like those based on the gradient, and its stochastic (like on–line learning), restricted (e.g. line search) variants, and may include higher order derivatives (as in Newton methods) (Pierre, 1986, Bazaraa, Sherali, & Shetty, 1993, Bertsekas, 1995). For these algorithms G depends on the value of the function and some of its derivatives at the current location. In a discretized implementation of derivatives, (and similar in a simplex search method) G depends on a set D_i of function values $r(\hat{f})$ which can be called a local population. Those methods find local minima. A nonlocal aspect can be added simply by combining them with unadapted search methods, usually implemented by comparing different local minima.

There can be nonlocal dependencies between function values, which makes it desirable to have G being dependent not only on a local but also on a nonlocal population of function values, which leads to

3. *parallel or nonlocal iterative methods.* Here G depends on a nonlocal population D_i. like in genetic algorithms, (See Holland, 1975, Goldberg, 1989, Davis 1991, Michalewicz, 1992, Schwefel, 1995, Mitchell, 1996) This allows nonlocal interactions between a possibly large number of function values. The dynamic of the population $D_i \rightarrow D_{i+1}$ corresponds to an iteration for a population vector.

In general, we only have to require that a fixed point of the iteration corresponds to a solution of the minimization problem. This allows transformations of the problem during iteration:

4. *Transformation methods* use transformations of the problem (i.e. of G_i), starting with an easy solvable, e.g. one–minimum problem, and slowly transforming to the problem of interest. They are called homotopy or continuation methods if they approximate a smooth family of transformations (Allgower, 1990, Richter & DeCarlo 1983, or, for an application in scattering theory, Giraud & Nagarajan, 1991, Wierling et al. 1994). Parameter, like the step width in gradient algorithms, mutation rate in genetic algorithms, or the temperatur in simulated annealing (See Ripley, 1987, Davis, 1987, Aarts & Korts, 1989) can be also seen as such deformation parameters.

   An example of transformations which correspond to a strictly monotonic transformation at fixed points are transformations with $h(y) \le h(y^*) \Rightarrow y \le y^*, \forall y$, which we will call *minimality sufficent relativ to y^**, if y^* is updated (at least from time to time) during iteration. (Accordingly $h(y) \ge h(y^*) \Rightarrow y \ge y^*, \forall y$ will be called maximality sufficient relative to y^*.) For example, adding a function $\tilde{r}(\hat{f})$ to the risk $r(f,\hat{f})$ (e.g. a (quasi) distance $D(\hat{f}^i, \hat{f}^{i+1})$) with a minimum at the current guess $(\hat{f}^i)$ ensures, that decreasing both terms $(r+\tilde{r})$ also decreases the original function (r). This is, for example, used in the *EM–like algorithms* (Dempster, Laird, Rubin, 1979, Tanner, 1993, Gelman, Carlin, Stern, Rubin, 1995, for an information geometrical interpretation and the related (most times identical) *em* algorithm see Amari, 1985, 1995).

Nonexhaustive search is always restricted to a subspace. Restricting to a subspace a priori is also called

5. a *variational method.* Here the function is parametrized and only a part of the parameters are used for minimization. In a linear variational method the stationarity condition is, for example, expanded into a linear basis of a Hilbert space of functions, and solved in a linear subspace.

We conclude with the interesting observation, that (nonexhaustive) minimization requires knowledge about nonlocal dependencies also for the risk functional $r(f,\hat{f})$. This suggests the principal possibility of reducing for practically solvable problems the risk functional to independent values, whose number in practice must be finite, corresponding to a finite $\hat{F}$ and finite effective F^0.

7 Bayesian approach

7.1 Inserting model states

The following constituents of a decision problem are assumed to be known: 1.) the action producing device $p(\hat{y}|q,\hat{f})$, 2.) the definition of the test distribution $p(q|y_c,\hat{y}_c)$,[41] 3.) the loss function $l(q,y,\hat{y})$. For evaluating a risk functional r the main problem remains determining the answer probabilities $p(y|q,f^0)$ for the test questions of the realized pure state (of nature). There exists a Bayesian and a frequenstist approach for this problem.

In the *Bayesian* approach model states f^0 are inserted as hidden variables. This is the concept we used in this paper defining a state of knowledge f with probabilities $p(f^0|f)$ and is called in the context of decision problems Bayesian decision theory (Berger, 1985). The answer characteristics $p(y|q,f^0)$ of the possible pure states have

[41] If not under direct control, $p(q)$ can be determined separatly, or the q can be included in the set of y.

to be known.[42] According to the Bayesian paradigm the (training) data dependence $p(f^0|f(D)) = p(f^0|D)$ of the state of knowledge $f = f(D)$ can be written

$$p(f^0|D) = \frac{p(D|f^0)p(f^0)}{p(D)} = \frac{p(y^D|q^D, f^0)p(f^0)}{p(y^D|q^D)},$$

with q^D, y^D the vectors of questions and corresponding answers in the data. A Bayesian expected risk reads

$$r(f, \hat{f}) = \int df^0 \, p(f^0|D) r(f^0, \hat{f})$$

with

$$r(f^0, \hat{f}) = \int dq \int dy \int d\hat{y} \, p(q|y_c, \hat{y}_c)$$
$$\times p(y|q, f^0) p(\hat{y}|q, \hat{f}) l(q, y, \hat{y}),$$

or for an inverse setting

$$r(f^0, \hat{f}) = \int dq \int d\hat{q} \int dy \, p(q|y_c, \hat{q}_c)$$
$$\times p(y|q, f^0) p(\hat{q}|y, \hat{f}) l(q, \hat{q}, y).$$

The posterior probability $p(f^0|D)$ is the only data dependent term. Introduction of model states f^0 makes the treatment of the training data independent of the test set. Both cases, test questions not included in the training data and training data not included in the test data are no conceptual problems.

When $p(y|x, f^0)$ is specified by $p(x|y, f^0)$ and an f^0–specific prior $p(y|f^0)$ according to

$$p(y|q, f^0) = \frac{p(q|y, f^0)p(y|f^0)}{p(q|f^0)}$$

one has to calculate the probability of the data under f^0

$$p(q^D|f^0) = \sum_{y^D} p(q^D|y^D, f^0) p(y^D|f^0),$$

(see for example Saul, Jaakola, Jordan, 1996) to get $p(f^0|D)$ and integrate over different states f^0.

7.2 Maximum posterior approximation

Practical calculations of a Bayesian risk are, if not analytically solvable, in general only possible for a restricted set of f^0 and $\hat{f}$. Numerical methods using Monte Carlo integration techniques[43] have been applied to Bayesian calculations (See Gelfand & Smith,1990, Gelfand, Hills, Racine–Poon, Smith, 1990, Geyer, 1992, Besag & Green, 1993, Smith & Roberts, 1993, Tierney, 1994, or Gelman, Carlin, Stern, & Rubin, 1995) including neural networks (Neal 1993, 1996), making the full integration in some cases feasible.

Those methods perform an plug–in estimate of the expected risk. The difference to the Frequentist method discussed later is that the test data are generated according to the a posteriori distribution. One also has to proof that the Monte Carlo estimate converges without having calculated the exact solution. That means using a finite sample we have to assume or calculate nonlocal knowledge about the risk function. In practice we may check the accuracy by repeating the calculation and estimating the variance of the optimal $\hat{f}$ using for example cross–validation or the bootstrap. Those are methods of classical statistics and a plug–in estimate is technically an empirical risk minimization (see Section 8) with *f–generated virtual examples* sampled according to $p(y|q, f)$.

The problems related to the plug–in principle will be discussed below for the Frequentist approach.

Alternatively, the method of Laplace is used to approximate the risk integral. This is the real version of the saddle point approximation or method of steepest descent for complex functions and also known in physics under the name 'low temperature' expansion (see for example: De Bruijn, 1981; Bleistein, Handelsman, 1986) Taylor expansion of $h(x)$ around its maximal value x^* and performing the resulting integrals gives for a real one–dimensional function h

$$\int_{-\infty}^{\infty} dx \, e^{-\beta h(x)} = \left(\frac{2\pi}{\beta h^{(2)}(x^*)} \right)^{\frac{1}{2}} e^{-\beta h(x^*)}$$
$$\times e^{+\frac{1}{\beta}\left(\frac{5(h^{(3)}(x^*))^2}{24(h^{(2)}(x^*))^3} - \frac{h^{(4)}(x^*}{8(h^{(2)}(x^*))^2} \right) + \mathcal{O}(\frac{1}{\beta^2})}, \qquad (14)$$

written for a function with one minimum $h(x^*)$ (or maximum for $-h$), $h^{(2)} > 0$, $h^{(i)}$ denoting the ith derivative at x^*. (In the multidimensional case $h^{(2)}$ in the square root factor for example has to be replaced by the determinant of the matrix of second derivatives.) For positive quadratic h, i.e. Gaussian $e^{-\beta h}$, all terms $h^{(i)}$ with $i > 2$ vanish. Higher order terms are obtained via Wick's theorem and include, in a graphical notation, all so called 'linked diagrams' (see for example Negele & Orland, 1988). Generalizations of the saddle point formula for multiple extrema exist which care about overlapping parts belonging to sufficiently close extrema (Berry, 1966, Miller, 1970, and Connor & Marcus, 1971). But one usually assumes the extrema to be well separated, and in our case, being not so much interested in the actual value of the Bayesian risk than in finding the best $\hat{f}$, only effects varying between different $\hat{f}$ would be important.

According to eq.(14) one might apply the saddle point approximation to the whole f^0–dependent integrand $p(f^0|D)r(f^0, \hat{f})$ or y– and q–dependent to $p(f^0|D)p(y|q, f^0)$. Both variants are clearly usually too complicated, leading for example to a $\hat{f}$–dependent fac-

[42] Under the assumption of the chosen model the Bayesian approach is (defined as being) optimal, but in practice the method depends of course on the correctness of the model for the situation in mind.

[43] Invented in statistical physics and going back to Metropolis, Rosenbluth, Rosenbluth, Teller, Teller (1953) (Metroplis algorithm) and Alder & Wainwright (1959) (molecular dynamics). For applications and developments in physics see for example Hammersley & Handscomb (1964), Binder (1986, 1987, 1995), Binder & Heermann (1988), or the last chapter in Montvay & Münster (1994), for mathematical background on Markov chains, for example, Seneta, 1981, and for their early use in statistics see Hastings, 1970, Ripley, 1977, Geman & Geman, 1984, Ripley, 1987.

tor depending on the second derivative.[44] But having a large amount of data it is often reasonable to assume the posterior probability $p(f^0|D)$ to be peaked sharply around one maximum. More formally, we may identify β with the number n of training data, and assume that the sample mean $\frac{1}{n}\sum_i \ln p(y_i|q_i,f^0)$ of the variable $z(y,q) = \ln p(y|q,f^0)$ becomes for large n a nearly n–independent function. Then the second order Taylor expansion of the logposterior $\ln p(y|q,f^0)$ (i.e. a Gaussian approximation for the posterior) can be a good approximation. For example, under some regularity conditions, (e.g. the number of parameters included in f^0 is not chosen to increase with the sample size, the limit is not at the edge of the parameter space F^0, or one uses a model specification where different parameter values f^0 correspond to identical probabilities $p(y|q,f^0)$ at the maximum) this will be the case for i.i.d. random variables $z(y,q)$ with finite variance, according to the general asymptotic Gaussian limit theorem (Le Cam, 1953, 1986, Le Cam & Yang, 1990, see also the discussion and references in Chapter 4 and Appendix B of Gelman, Carlin, Stern, Rubin, 1995). Then the posterior will have a variance $(nJ(f^0))^{-1}$ where J is the expectation under the true state of (the matrix of) the second derivatives of the logposterior (Fisher information). For dependent variables this is not necessarily true, but can also be the case. Then, if $r(f^0,\hat f)$ varies only weakly with f^0 compared with $p(f^0|D)$ (Gaussian $p(f^0|D)$ alone is not enough in this case) it does not strongly influence the location of the maximum. Then we can identify β with n, and $h(x)$ with $-\frac{1}{n}\left(\sum_i \ln p(y_i|q_i,f^0) + \ln p(f^0)\right)$. In this case where only part of the exponent is multiplied by β, i.e. the integrand has the form $e^{\beta h(x)+\ln g(x)}$, we can apply the slightly more general formula

$$\int_{-\infty}^{\infty} dx\, g(x) e^{-\beta h(x)} \approx g(x^*) \left(\frac{2\pi}{\beta h^{(2)}(x^*)}\right)^{\frac{1}{2}} e^{-\beta h(x^*)}.$$

with $g(x)$ corresponding to $r(f^0,\hat f)$ and x^* is the location of the maximum of $h(x)$, i.e. independent of $g(x)$.

This is called *maximum posterior approximation* (MaP). Especially for high dimensional spaces F^0 the off–peak contributions can be large requiring a large amount of data to allow a MaP approximation. Note also, that in the context of decision theory we only need to evaluate the Bayesian risk to select a optimal $\hat f$ and we do not have to require a good approximation of the risk itself, and $\hat f$–independent factors are therefore not important, like the ones including the second derivative $h^{(2)}$ or the factor $p(y^D|q^D)$.

Hence, to apply a q–, y–independent MaP approximation with respect to f^0 we write the f^0–dependent but q–, y–independent probabilities in exponential form

$$p(y_i^D|q_i^D,f^0) = e^{L^D(y_i^D|q_i^D,f^0)},$$

and

$$p(f^0|f) = e^{L^0(f^0)},$$

defining the logprobabilities L^D (loglikelihood), L^0 (logprior) and get

$$r(f,\hat f) \propto \int df^0\, e^{\left(\sum_i L^D(y_i^D|q_i^D,f^0)+L^0(f^0)\right)} r(f^0,\hat f).$$

If we include the logprior $\ln p(f^0)$ into $h(x)$, it is also included in the determination of the maximum $h(x^*)$. This allows to discuss situations where the n data are not enough to yield sharp, especially nondegenerate, maxima (and indeed, choosing the relevant questions as basis questions, $X = Q^l$, the prior is always essential for local data if $n < |X|$). In contrast we assumed $r(f^0,\hat f)$ not to be important for the location of the maximum. An extreme example for the opposite would be if the risk for the most probable state $f^{0,*}$ is for all $\hat f$ infinite, meaning that $f^{0,*}$ can be excluded from F^0. Less extreme, a risk can be strongly peaked at an f^0 with low posterior for some $\hat f$. On the other hand, it only matters which $\hat f$ is finally selected as the optimal one. This implies robustness against all errors, made somewhere on the way, which do not change this final decision.

We summarize that the MaP approximation includes probability aspects, but not aspects of relevance related to the loss function.

Maxima may be degenerated or weakly peaked within a subspace of parameters of F^0. Then one may perform a partial saddle point approximation for the subspace where the necessary conditions are fulfilled. For example, the risk may measure aspects of (i.e. depend on parameters describing) f^0 which are not measured by the data, so $p(f^0|D)$ does not dependent on them. Then the locations of the maxima depend on the risk $r(f^0,\hat f)$ and the maxima are necessarily degenerated in direction of those relevant but not measured parameters (e.g. in a model with $X = Q^l$, uniform prior, and only local data). Then the MaP step returns not a single point but a subspace of important possibilities. More general, the maximum can, after incorporating data and prior, still be rather flat in some of the relevant dimensions. Then a MaP approximation should only replace a part of the f^0–integration while another part, i.e an integration over a subspace of parameters of f^0, remains. Performing this integration gives a new effective risk sensitive to the available data.

The MaP approximation consists in finding the most probable state $f^{0,*}$ to approximately calculate the f^0 integral. Then the factor $\sum_i L^D(y_i^D|q_i^D,f^{0,*}) + L^0(f^{0,*})$, being independent of $\hat f$, can be skipped and the optimal action state $\hat f^*$ can be found by minimizing $r(f^{0,*},\hat f)$. Often $\min_{\hat f} r(f^0,\hat f)$ is a constant over all f^0, like in the usual regression case with mean square error, deterministic unrestricted $\hat f$, and Gaussian f^0 with f^0–independent

[44] On the other hand the $\hat f$–dependency is an interesting feature if $r(f^0,\hat f)$ can be included , because then the saddle point approximation can be adapted to $\hat f$. It would lead to coupled maximization and minimization problems. (In contrast we will discuss below a maximization problem which is independent of the subsequent minimization problem.) This dependency of the maximization problem on the the corresponding minimization problem, i.e. its adaption to $\hat f$, suggests for example an iterative procedure where both steps are performed alternately.

variance. Thus, the full approximation procedure consists of two steps (MaP–MiR)

1. Maximization of the posterior (MaP):
$$f^{0,*} = \text{argmax}_{f^0} \left(\sum_i^N L^D(y_i^D|, q_i^D, f^0) + L^0(f^0) \right),$$
2. Minimization of risk (MiR):
(for the state $f^{0,*}$ with maximal posterior)
$$\hat{f}^* = \text{argmin}_{\hat{f}}\, r(f^{0,*}, \hat{f}).$$

Note that in this approximation the MaP step is performed independent of the aspects important for the MiR step. The first step can be interpreted as finding an approximation independent from its application. The second step uses the best found approximation for a specific application situation defined by the loss function. This is the usual implicit setting when looking for approximations without specifying applications for which they will be used. It has the advantage that the independence allows the same approximation to be used for several different applications. Thus, in the every day use of this procedure a statistician performs the first approximation step and potential users of the approximation the second, adapted to their problem. An example for a MaP–MiR related algorithm can be found in (Lemm, Beiu, Taylor, 1995). There the MaP step is implemented as a density approximation. A subsequent constructive algorithm tries to find a solution easy to implement in hardware, not working directly with the data but with the results of the density estimation of the first step. This is an attempt to minimize also aspects of the loss not related to approximation and corresponds to the MiR step.

The loss function depends on the action state $\hat{f}$ which may include aspects like complexity of $\hat{f}$. But note that the loss measures no aspects like complexity of the algorithm used to find the optimal (or a good) $\hat{f}$. So to say, action loss is included but no algorithmic loss. As a two-step procedure can often be expected to be more complex than a one–step procedure, the MaP–MiR procedure seems to be more appropriate for situations where loss related with $\hat{f}$, for example its approximation ability and complexity, are more important than aspects of the loss related to the requirement of resources of the algorithm. But as stated in Section 6 one can also consider algorithms as part of $\hat{f}$ in a higher level problem. Then one can include specific algorithmic aspects of the loss and look for an optimal algorithm for a certain distibution of application (learning) situations. Again, to solve for the best algorithm one has to use a meta–algorithm and the same kind of problem appears on this higher level as here meta–algorithmic loss aspects are not included. Going further, meta–algorithms could be included into $\hat{f}$ using meta–meta–algorithms and so on, but usually the complexity from one level to the next increases so much that such applications are not expected to be feasible in most practical cases.

There are cases where $p(y|q, f^0)$ depends on a huge number of internal ('hidden') integration variables variables z. Then an approximated logposterior must be maximized. Exchanging a nonlinear function with the integration is called annealed approximation (Seung, 1995). The replica approach is a special adaption of a saddle point method for functions representing the OR for (a large number of) disjunct events being of the form $g = \ln \sum_i^n e^{g_i}$ in the limit $n \to \infty$ (Mezard, Parisi, & Virasoro, 1987). This situation can occur, for example, if algorithms are compared with respect to a large ($n \to \infty$) number of application situations.

The second MiR step uses $p(y, q|f^0)$ to find the best alternative $\hat{f}$, i.e. for 'training' of the action model. A numerical evaluation of the integral can be seen as an empirical risk minimization (see Section 8) on a set of *(f^0–generated) virtual examples* sampled according to $p(y, q|f^0)$, like a numerical evaluation of a full Bayesian approch can be based on f–based virtual examples generated according to $p(y, q|f)$.

Note that the distinction between a parallel and inverse decision problem, corresponding to choosing $p(\hat{y}|q, \hat{f})$ or $p(\hat{q}|y, \hat{f})$ to define the action model, only matters for the MiR and not the MaP step. Also, the model needs not necessarily to be reduced to only one remaining state. Thus, MaP–MiR can be generalized with a first step reducing the space F^0 to a smaller $\tilde{F}^0$ and a second step minimizing the risk within that $\tilde{F}^0$. But using more than one state for the minimization step, like taking for example the n most probable states or skipping only very unlikely ones, requires calculation of the relative weights of the states related to second derivatives with respect to the parameters of f^0. Every decision problem defines an optimality mapping by $\hat{f}(f^0) = \text{argmin}_{\hat{f}\in\hat{F}}\, r(f^0, \hat{f})$ for all $f^0 \in F^0$ and analogously for $f \in F$. In principle, one can restrict the search space $\hat{F}$ by eliminating $\hat{f}$ being never optimal. In the case available data have equal probability within $f^0 \in [f^0]_{r^*}$ i.e. they do not distinguish between states f^0 leading to the same decision $\hat{f}$ identifying those makes optimality mapping one–to–one. With respect to such a construction the space F^0, and therefore also the MaP step as maximization over $f^0 \in F^0$, is not independent of the loss function.

7.2.1 Perturbation theory beyond MaP

The MaP approximation can be extended by expanding the exponential to higher orders around the Gaussian reference point. We already mentioned that higher order contributions can be obtained by including all linked diagrams according to Wick's theorem. In general, a perturbation theory can be built upon any reference point based upon the general formula (See the Section about Heim's perturbation theory in Jaynes, 1996)
$$e^{A+\epsilon B} = e^A \left[1 + \sum_{n=1}^{\infty} \epsilon^n S_n \right],$$
with
$$S_n = \int_0^1 dx_1 \int_0^{x_1} dx_2 \cdots \int_0^{x_{n-1}} dx_n \prod_1^n B(x_n),$$
and
$$B(x_n) = e^{-x_n A} B e^{x_n A}.$$

Here A, B stand for matrices, x for a real number and xA means multiplication of each entry of A by x. In matrix notation we have

$$r(f,\hat{f}) =< \tilde{l}(q,y,\hat{f}) >$$

$$= \mathrm{Tr}(\rho(q,y,f^0,f)\tilde{l}(q,y,\hat{f})) = \mathrm{Tr}(e^L(q,y,f^0,f)\tilde{l}(q,y,\hat{f})),$$

with $L(q,y,f^0,f) = \ln(p(f^0|f)p(q)p(y|q,f^0))$, the trace denoting the integrals Tr= $\int df^0 \int dq \int dy$ for diagonal matrices $L(f)_{i,j}$, $\tilde{l}(\hat{f})_{i,j}$ with indices $i = (q,y,f^0)$, $j = (q',y',f'^0)$, and $\rho(q,y,f^0,f) = e^L(q,y,f^0,f)$. Because the trace is invariant under similarity transformations S this formulation allows, if convenient, to work with nondiagonal $L' = SLS^{-1}$, $\tilde{l}' = S\tilde{l}'S^{-1}$.[45]

If we now write $L = A + \epsilon B$, the expected risk can be expressed completely by unperturbed expectations $< \cdots >_A$ with respect to the reference e^A

$$< \tilde{l} > - < \tilde{l} >_A=$$

$$\sum_{n=1}^{\infty} \epsilon^n (< Q_n \tilde{l} >_A - < Q_n >_A < \tilde{l} >_A),$$

with

$$Q_1 = 1; Q_n + S_n - \sum_{k=1}^{n-1} S_k < Q_{n-k} >_A, \quad n > 1.$$

8 The Frequentist approach

8.1 Empirical risk minimization

The Frequentist paradigm is related to the general plug–in or bootstrap principle.[46] It is also called Monte Carlo estimate if applied to calculate expectations (Efron, B. & Tibshirani R.J. 1993). It is based on results of the theory of uniform convergence. A functional of a population distribution is estimated by applying the same functional to an empirical sample drawn according to that distribution.[47] In our case the risk functional r is replaced by an empirical risk $\hat{r}$: Training questions q^D are generated according to $p(q^D|y_c^D, \hat{y}_c^D)$ (or usually $p(q^D)$) and the risk functional r is applied to the empirical distribution of

$$l(q_i^D, y_i^D, \hat{y}_i).$$

In the case the risk functional is chosen as the expectation and $p(q|y_c, z_c) = p(q|y_c)$ we define an integrated loss function

$$\int dz\, p(z|q,y,\hat{f}) l(q,y,z) = \tilde{l}(q,y,\hat{f}),$$

Specifically, in parallel decision problems z is equal to $\hat{y}$ and in inverse problems equal to $\hat{q}$. Note that in $\tilde{l}(q,y,\hat{f})$ $\hat{f}$ just denotes parameters. Yet, for a given loss function we can always introduce some effective deterministic function $\tilde{f}(q)$ (or $\tilde{f}(y)$) (not uniquely defined and possibly vector valued) containing the dependence of $\tilde{l}$ from the parametrization of $\hat{f}$ and (part of the dependence from) q (or y) according to $\tilde{l}(q,y,\hat{f}) = \tilde{l}(q,y,\tilde{f}(q))$ or $\tilde{l}(q,y,\hat{f}) = \tilde{l}(q,y,\tilde{f}(y))$. That means every decision problem with $p(q|y_c,z_c) = p(q|y_c)$ can be seen as equivalent to a decision problem with deterministic function $\tilde{f}$. For a parallel decision problem with a deterministic $\hat{y}$–producing device we choose the functional dependency of $\tilde{l}$ from its first argument q to be equal to that of l from its first argument q, which gives $l = \tilde{l}$ and $\hat{f}(q) = \tilde{f}(q)$. Alternatively, we can simplify the notation by absorbing the first q into the definition of $\tilde{f}(q)$ and write in such cases $\tilde{l}(y, \tilde{f}(q))$. Analogously, in an inverse setting we define $\tilde{f}(y)$ and can choose in the deterministic case $\tilde{f}(y) = \hat{f}(y)$.

The expected risk

$$r(f^0,\hat{f}) = \int dq \int dy\, p(q,y|f^0)\tilde{l}(q,y,\hat{f}),$$

with $p(q,y|f^0) = p(q|y_c)p(y|q,f^0)$ is approximated by

$$\hat{r} = \frac{1}{n}\sum_i^n \tilde{l}(q_i^D, y_i^D, \hat{f}) = \frac{1}{n}\sum_i \tilde{l}_i(\hat{f}),$$

with the data sampled according to $p(q,y|f^0)$.

The plug–in principle assumes the distribution of training data D including the distribution of questions to be identical to that of the test data $\hat{D}^l$ including $p(q^l)$ for the relevent q^l for which we want to calculate the functional r. This holds for example in a setting where both training and test set are generated by the same (stationary) device. Then an explicit knowledge of the generating distribution is not necessary.

Because the empirical risk values are used to make the decision, the chosen $\hat{f}$ depends on the training data and the empirical risk of the chosen $\hat{f}$ does not approximate its expected risk. To estimate the expected risk one has to reevaluate the risk for the chosen $\hat{f}$ with new, independent sample data, the empirical test set, not involved in the decision. (Not to be confused with the (true) test set D^l in the definition of the decision problem.) Bounds for the difference between the expected risk of the true optimal $\hat{f}$ and the chosen $\hat{f}$ are given by the theory of uniform convergence (Vapnik, 1982; Dudley, 1984; Pollard, 1984; Haussler 1995, for an introduction e.g. Kearns & Vazirani 1994). These bounds are based on what we called structural information. Their local part, like bounds for absolut values or the local variance, allows locally the application of probability theoretic inequalities like Hoeffding's or Chebyshev's inequality. Their nonlocal part, formulated for example as finite ϵ-entropy, or finite pseudo or VC dimension of a set $\tilde{l}(q,y,\hat{f})$ for $\hat{f} \in \hat{F}$, allows generalization. Then, when the training data are i.i.d. sampled according to the test distribution, the theory of uniform convergence gives bounds on the

[45]See for example Derka, Bužek, Adam, & Knight (1996) for Bayesian inference with density operators used in quantum theory.

[46]Often the term bootstap refers to the special case of estimating the standard error of some sample estimate which then requires resampling.

[47]As empirical distributions of finite samples are a quite restricted class of distributions, functionals equal for them can differ on general distributions.

probability of deviations of the empirical risk from the expected risk in the true state f^0. For example, (Vapnik, 1995) gives bounds $\delta(f, N)$ in terms of the VC dimension for

$$p^{VC} = \sup_{f^0} p(\sup_{\hat{f}} |\hat{r}(D(N, f^0), \hat{f}) - r(f^0, \hat{f})| > \epsilon | f^0),$$

for bounded risk. The supremum over true states f^0 (with $p(f^0|f) \neq 0$) is implicit in the results using the definition of the VC dimension and can obviously be replaced by $\forall f^0$. Using worst case considerations there is no need in this theory to explicitly calculate posterior probabilities.

There are some recent studies how specific nonlocal information affects the bounds of uniform convergence (Abu–Mostafa, 1990, 1993a, 1993b; Ratsaby, Maiorov, 1996). But for general nonlocal information a reformulation in terms of VC dimension or ϵ–entropy is often difficult or even practically impossible, and one has to use an upper bound for them, giving the results of the theory of uniform convergence another worst case interpretation. Then, especially when only few local data are available, the uniform bounds can be trivial or weak.

8.2 Vocabulary and framework

We formulate the Frequentist setting in a decision theoretic language. Assume the availability of a stationary sampling process S to generate training questions $q \in Q^D$ according to some $p^S(q)$ for which answers are available and the ratio with the distribution of relevant questions $p(q)/p^S(q)$ is known. We call the set Q^S of questions with $p^S(q) \neq 0$ the sampling population (for questions) and specify for the present context Q^D to be the set of sampled or training questions with answers used for the plug–in estimate. Including previously defined sets of questions we have the following listing

1. Q^S the *sampling population* with $q \in Q^S$ the sampling questions,
2. Q^D the set of sampled or *training questions*,
3. Q^0 the set of *prior questions* being the questions with data available but with $q \in Q^0$ not sampled according to S or not used for the plug–in estimate,
4. Q^l the set of *relevant questions*,
5. $Q^c = Q^l \setminus Q^S$ the set of *cost questions*,

Data (q_i, y_i) are obtained by using measurement devices for q_i to find results y_i. With respect to a sampling process S we separate the available data into the two groups:

1. *Sampled data* or *training examples* D^S, we will write more shortly just D skipping the superscript S, which are obtained using the available stationary sampling process S and used to calculate the empirical risk via the plug–in principle. Nonlocal questions can be included in Q^S.
2. *Prior data* D^0, being all other data with questions generated from other processes $S' \neq S$. Such processes can be unknown processes from the past possibly different from S, they can be nonstationary like for active queries, they can use devices measuring other questions or represent active control. All priors $p(f^0|f)$ can be related to a factorial prior $p_{fact}(f^0) = \prod_{x \in X} p(f^0_x)$ by data D^0 with $p(f^0|f(D^0)) \propto p_{fact}(f^0) p(D^0|f^0)$. The data D^0 are not uniquely defined and the corresponding questions need not necessarily to be in the sampling population Q^S. Data which are sampled according to S but not used for the plug–in principle are not sampled data but prior data.

For a Bayesian treatment the distinction is not important, but in a Frequentist approach only sampled data are used for the plug–in principle while prior data only enter in form of restrictions of the space $\hat{F}$. Note that this use of the term prior does not refer to the temporal aspect meaning information collected previously to the data D. We made things simpler by not trying to distinguish nonsampled nonprior data from priors. Thus, identification of the reference factorial prior as well as assumed and not measured data are understood as being part of the prior data D^0. In another context it might well be convenient, but not necessary, to distuingish nonsampled data from prior data with the latter having a more temporal connotation or refering to assumed and not measured data.

According to this distinction of data we can also split the logposterior into a sampled and prior part

$$L(D, D^0, f^0) = L^D(D, f^0) + L^0(D^0, f^0)$$

$$= \sum_i L^D(y_i^D | q_i^D, f^0) + L^0(f^0).$$

Introducing a dummy index one can always achieve $Q^0 \cap Q^S = \emptyset$ even if for a question sampled and prior data are available at the same time. Nonrelevant sampled questions $q \notin Q^l$ can be excluded from Q^S because for them $p(q) = 0$. If a sampling process nevertheless produces them they can instead of simply being eliminated technically be treated like prior data influencing $p(f^0)$. The property of being sampled is for them not important and we choose in the following $Q^S \subseteq Q^l$. Including nonlocal questions like smoothness into Q^S instead of nonlocal prior data could also enable generalization. But this requires nonlocal questions also to be relevant, i.e. in Q^l, what is usually not the case. Also we prefer to treat the $q \in Q^l$ as being independent without priors.

On the other hand we do not assume all relevant questions necessarily to be available for sampling. That allows $Q^l \supset Q^S$ which represents situations where only part of the expected risk can be estimated by sampling. We call the remaining part *costs*. Complexity costs are a typical example. It must be determined by other informations which could come from another sampling process, a Bayesian calculation, or if Q^c is finite and deterministic from a complete set of answers. Fig.8 shows the relations graphically.

We define prior data for question q to be in the set of

a. *Hints* D^0_h if questions from Q^S depend on them, that is if $X^q \cap X^{Q^S} = X^S \neq \emptyset$, (Here X^q denotes the basis of prior question q.)

b. *Cost priors* D^0_c if cost questions from Q^c depend on them, that is if $X^q \cap X^{Q^c} = X^c \neq \emptyset$. Here X^q denotes the basis of prior question q.

A specific prior can be both and therefore the two sets need not to be disjunct.

Splitting the q–integrations of the expected risk into a *sampled risk* r^D with $q \in Q^S$, determined by answers to $q \in Q^D \subset Q^S$ sampled by S, and an additional *(non-sampled) cost term* r^0 for $q \in Q^c$, determined by D^0, gives

$$r(f,\hat{f}) = r^D(f(D,D^0),\hat{f}) + r^0(f(D,D^0),\hat{f}),$$

with $r^D(f,\hat{f})$ being

$$\int_{Q^S} dq \int dy \int df^0 p(f^0|D,D^0)p(q)p(y|q,f^0)\tilde{l}(q,y,\hat{f}),$$

and analogous for $r^0(f,\hat{f})$ with $\int_{Q^c} dq$. We write $\tilde{l}^D$ for the part of the loss function depending on $q \in Q^S$, that means $\tilde{l}^D(q,y,\hat{f}) = \tilde{l}(q,y,\hat{f})$ for $q \in Q^S$ and zero otherwise, i.e. if $q \in Q^c$. Anologously, we write $\tilde{l}^0$ for the part depending on $q \in Q^c$. The parts are only defined up to q–independent terms, because such terms can be shiftet between questions without changing the risk. We can use this freedom for a convenient choice.

Cases in which available data for questions in the loss function are partly sampled as well as not sampled are included in this definition by duplicating questions in Q^S with a dummy index,

$$\tilde{l}(q,y,\hat{f}) \rightarrow \tilde{l}^D(q,y,\hat{f}) + \tilde{l}^0(q',y',\hat{f}),$$

if we define $q' \in Q^c$ for $q \in Q^S$.

The cost term can depend in general on all data including the training examples D. In such cases one may call $r^0(f,\hat{f})$ *posterior (nonsampled) costs* or *data dependent (nonsampled) costs* as they have to be determined after having seen all the data. If $p(f^0|D,D^0)=p(f^0_{Q^S}|D,D^0_h)p(f^0_{Q^c}|D^0_c)$ with $D^0_h \cap D^0_c = \emptyset$ then the $f^0_{Q^S}$–integration vanishes. Understanding $f^0 = f^0_{Q^S}$ to be the restriction on Q^S we can write

$$r(f,\hat{f}) = r^D(f,\hat{f}) + r^0(\hat{f}),$$

and call $r^0(\hat{f}) = \tilde{l}^0(\hat{f})$ *prior costs*. Being q–independent, $\tilde{l}^0$ can also be seen as part of $\tilde{l}^D$.

Examples for possible (prior) costs include, storage requirements for parameters of $\hat{f}$ like the number of weights of a neural network or number of nodes in a decision tree, penalties for on–line evaluation times, criteria related to understandability for human experts, or to an effective and cheap harware implementation in VLSI technology.

8.3 Sampling generalized questions

Reweighting

Here we consider the case of relevant questions $q \in Q^l$ not directly sampled, i.e. $q \notin Q^S$ or equivalently $q \in Q^c$. We show that for such questions which have a basis X^q of q completely within Q^S theoretically, but often not

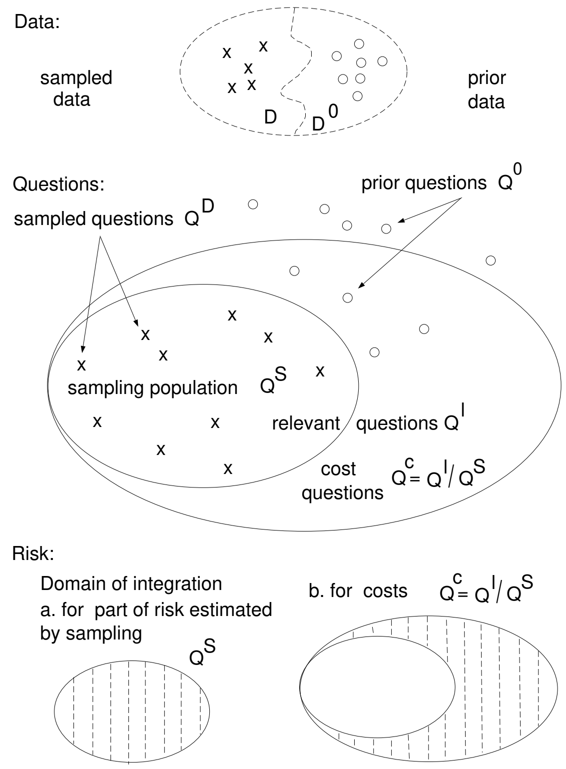

Figure 8: Shown is the distinction for data into the two subgroups of a. sampled or training data being sampled from a stationary process S and b. prior data from any other sources. For questions there are 1. the set of relevant questions Q^l, 2. the sampling population Q^S from which training questions are drawn according to a stationary $p^S(q)$, 3. the training questions ($\times$), and 4. prior or nonsampled questions ($\circ$). The sets of training and prior questions are finite. The part of the loss integration depending on Q^S can be determined by sampling training data. The remaining part of the expected risk is called costs and must be determined for example by a Bayesian risk calculation, another sampling process or using exhaustive queries if $Q^c = Q^l \setminus Q^S$ consists of a finite number of deterministic questions.

practically, the sampling process S can be extended to an S' so that $q \in Q^{S'}$. The basic fact used in reweighting for evaluating an integral by the plug–in principle is that the factorization of the integrand into function and probability is not unique,

$$\int dz\, p(z) g(z) = \int dz\, p'(z) \frac{p(z)}{p'(z)} g(z) = \int dz\, p'(z) g'(z),$$

where $g'(z)$ is equal to $g(z)$ multiplied by the reweighting factor $p(z)/p'(z)$. That means, instead of sampling z according to $p(z)$ and summing up $g(z_i)$ for each sample point z_i we can, assuming $p(z)$ and $p'(z)$ are known, alternatively sample according to $p'(z)$ summing up $g'(z_i) = (p(z_i)/p'(z_i))g(z_i)$. For example, the method of importance sampling (Montvay & Münster, 1994) uses reweighting to reduce the plug–in error by choosing the reweighting factor so that the reweighted function $g'(z)$ is as constant as possible. But note that the factor $p'(z)$ must be a probability and can especially never be negative. In the situation we are discussing here, $p(z)$ corresponds to the relevant distribution of test questions $q \in Q^l$ with $p(q, y|f^0) = p(q)p(y|q, f^0)$ while $p'(z)$ is the distribution available to generate training data.

The simplest case of reweighting is when the set of test and the sampling population of potential training questions are the same , i.e. $Q^S = Q^l$, but have different probability distributions $p^S(q) \neq p(q)$. Then, if the ratio $p(q)/p^S(q)$ is known (not necessarily the test and available training distributions $p(q)$, $p^S(q)$ itself) one can use $p(q)/p^S(q)$ as reweighting factor for the loss function. Thus, the sampling data term of the risk can be related to a sampling process S with $p^S(q)$ by

$$r(f^0, \hat{f}) = \int dy p^S(q) p(y|q, f^0) \tilde{l}^W(q, y, \hat{f}),$$

where the definition of the reweighted loss

$$\tilde{l}^W(q, y, \hat{f}) = \frac{p(q)}{p^S(q)} \tilde{l}(q, y, \hat{f})$$

compensates for deviations between $p(q)$ and $p^S(q)$ and $p^S(q) \neq 0$ according to the definition of Q^S. If not stated otherwise we understand in this paper implicitly $\tilde{l}$ to mean $\tilde{l}^W$ or $p(q)/p^S(q) = 1$.

Only a deterministic function must be included for a $q \in Q^l$ depending deterministically on one $q \in Q^S$. But for a $q \in Q^l$ depending deterministically on two or more $q_i \in Q^S$ the probability being sampled decreases with the product of the probabilities for (independent) q_i.

More general is a probabilistic dependence on answers to questions from Q^S. Consider relevant questions $q \in Q^l$ given in their dependence on the sample questions q^S

$$p(y|q, f^0) = \int dq^S \int dy^S \int dz^S\, p(q^S|y^S_c, z^S_c, q)$$

$$\times p(y^S|q^S, f^0) p(z^S|q^S, y^S, q) \delta(q(q^S, y^S, z^S) - y),$$

with the two f^0–independent and q–dependent probability factors and the defining function $q(q^S, y^S, z^S)$ assumed to be known. The probability $p(y^S|q^S, f^0)$ depends on the state of nature and is unknown. Then eliminating the δ–function by performing the y–integration shows that the empirical risk is a sum of

$$l(q_i, y_i = q(q^S_i, y^S_i, z^S_i), z_i) = l(q_i, q^S_i, y^S_i, z^S_i, z_i),$$

when a sampling procedure for the variables is available according to

$$p(q, z, q^S, y^S, z^S) = p(q|y_c, z_c) p(q^S|y^S_c, z^S_c, q)$$

$$\times p(y^S|q^S, f^0) p(z^S|q^S, y^S, q) p(z|q, y(q^S, y^S, z^S), \hat{f}).$$

Using the corresponding devices the sampling steps within the single components are the following:

$$q \rightarrow q^S \rightarrow y^S \rightarrow z^S \rightarrow z$$

Assuming probability distributions $p^S(q^S|y^S_c, z^S_c, q) =$ to be available to generate training data gives a q–, y^S_c–, z^S_c– dependent reweighting factor for the loss function

$$\frac{p(q^S|y^S_c, z^S_c, q)}{p^S(q^S|y^S_c, z^S_c)}.$$

As already discussed for the deterministic case it is a principal problem for nonlocal questions that the q^S can form a vector. Take as example a question measuring answer differences between points x_1 and $x_2 = x_1 + \Delta$

$$p(y|q, f^0) = \int dx_1 \int dx_2\, p(x_1|q) \delta(x_2 - \Delta - x_1)$$

$$\times p(y_1|x_1, f^0) p(y_2|x_2, f^0) \delta(y_1 - y_2 - y),$$

and assume that the x_i are i.i.d. sampled. Here the probability to have a complete pair of two values $y_1(x_1)$ and $y_2(x_2)$ to insert as $y = (y_1, y_2)$ into the loss function has measure zero in standard continuous cases (even for $\Delta = 0$) if Δ is fixed and not integrated over. If the distribution is not as badly behaved as the δ–function in the example there is still the high dimensionality of x.[48] That means the amount of available data is usually too small to sample such nonlocal questions.

Missing values

One could think in remedy the situation by using guesses for the missing values necessary to answer a generalized question. Indeed, the Monte Carlo method or plug–in principle for expectations can be interpreted as replacing missing values by the data mean. This implicitly assumes the loss function to be constant (or compensating) for missing values. For example, the empirical risk estimate can be seen as minimization of a mean square error of the empirical loss with a prior (for the loss, not for f^0) $\lambda(\tilde{l}(q, y, \hat{f}) - E(\tilde{l}))^2$, with $E(\tilde{l})$ the expectation of $\tilde{l}$, in the $\lambda \rightarrow 0$ limit.

[48] High dimensionality itself is not the problem if there are enough restrictions for the function. The situation is like in one–dimensional problems as a theorem from Kolmogorov shows that every continuous function of several variables can be expressed as superposition of functions of one variable for closed and bounded input domain (Kolmogorov, 1957). But the nonlocal priors used in the theory of uniform convergence, corresponding for example to a specific VC dimension or ϵ–entropy, and related to a restricted $\hat{F}$, might not be considered as weak and harmless in those cases without empirical foundation.

While the plug–in principle replaces missing values y_x by the global sample mean $\sum_x y_x$, one might wish better locally varying approximations of $p(y|x, f^0)$. This can be done by methods of density estimation, including parametric approximations and nonparametric methods like splines, kernel and nearest neighbor methods, (see for example: Silverman, 1986; Härdle, 1990) or neural networks.

Optimal in a Bayesian sense would be sampling according to the posterior probability $p(y|q, f(D))$, but without reference to a specific model of f^0 all methods are somewhat ad–hoc.[49] But all those approximation methods used to replace missing values can be interpreted as an approximation of the posterior $p(f^0|D)$ with respect to some (implicit) model of $p(y|q, f^0)$. Specifically, Girosi, Jones, Poggio (1995) relate common interpolation methods (Radial Basis Functions, splines) to regularization terms, i.e. from a Bayesian point of view to priors within a maximum posterior approximation. Also note, that all these approximation methods which one can use to replace missing values are itself decision algorithms and have therefore, whithin a Frequentist approach, the same problems with general structural information as the Frequentist approach for our original decision problem. Thus, all the aspects we discuss for evaluating the empirical risk for generalized questions appear here again for a specific approximation problem.

8.4 Prior data

Extended loss, indirect priors, and virtual examples

Common is a situation where prior information is available in addition to training data sampled from the test (relevant) questions. This information can always be seen as corresponding to questions not included in the relevant set Q^l. An example are priors like an approximate symmetry and smoothness, being normally not sampled and not included in the set of test questions.[50]

In those cases the loss function can be extended by additional terms to be defined also for data not in the test set. This means, the plug–in principle uses an extended expected risk $\bar{r}$ with respect to an extended set $Q^{\bar{S}}$

$$\int dq^{\bar{S}} \int dy \int d\hat{y}\, p(q^{\bar{S}}|y_c, \hat{y}_c) p(y|q^{\bar{S}}, f^0) p(\hat{y}|q^{\bar{S}}, \hat{f}) \bar{l}(\bar{q}^{\bar{S}}, y, \hat{y}),$$

with an extended loss function $\bar{l}$ to approximate the (true) expected risk under the state of nature f^0. For example a smoothness property can be included as additional term. The optimal weight of the additional questions, i.e. $p(q^{\bar{S}})$, is usually determined by cross–validation or similar methods (see next Subsection). The extended loss should be choosen so that the relevant features according to the risk functional r of $p(\bar{l}|f, \hat{f})$ are similar to $p(l|f, \hat{f})$ but this requires in general a model of f^0 to be determined. In usual approximation problems one chooses an extended loss which enforces $\hat{f}$ to answer similar as f^0 also to the not relevant questions as f hoping that this results in similar answers to relevant questions, too. We will relate this ad–hoc method for approximation problems to the maximum posterior approximation within the Bayesian approach. We now show how extra terms can be interpreted as arising from a Lagrange implementation of indirect priors.

An indirect way to include nonlocal information is trying to transform our knowledge f into knowledge about $\hat{f}$. More precisely, if we call the probability of choosing $\hat{f}$, i.e. $p(\hat{f}|f, \hat{a}, q_r)$ an *indirect prior*, one is interested in excluding alternatives $\hat{f}$ with zero (or small[51]) indirect prior probability. As the probability of selecting $\hat{f}$ depends in general not only on $f(D)$ but also on the decision problem, i.e. the q_r, including the risk functional, loss function, test data distribution, and the algorithm $\hat{a}$, its complete determination is a much more complicated than the decision problem itself requiering the representation of f in a model with certain f^0. Ideally, we are interested in the zero part of the indirect prior resulting from the optimal algorithm defined by q_r.

For example, in approximation problems with a quadratic loss function $\tilde{l}(q, y, \hat{f}) = (y - \hat{y}(\hat{f}))^2$ we know that the optimal solution is the true regression function $E(y(f^0, x))$ if contained in the search space $\hat{F}$. Then we can simply implement deterministic information about the true regression function of f^0 by the corresponding restrictions on $\hat{f}$, that is if we know $p(y|q, f^0) = \delta(q(\{E(y(f^0, x)), x \in X\})) - y)$ we only use $\hat{f}$ with $q(\hat{f}) = \hat{y} = y$.

Sometimes, assuming the existence of a state producing process with stationary distribution corresponding

[49] Such methods can also be used directly for the loss function. This corresponds to a $\hat{f}$–dependent prior on the loss function, usually difficult to relate to priors on f^0. The optimal solution $p(l|f(D), \hat{f})$ of this problem in the Bayesian framework is already nearly equivalent to the solution of the whole decision problem or even more complicated as not the whole loss distribution might be relevant for the risk functional r.

[50] We could define an induction problem as a situation where the generating distribution for test and training data coincide, a transduction problem as a situation where some data do not belong to the test set (Vapnik, 1982). In this formulation, most practical problems are not induction but transduction problems as the nonlocal (e.g. smoothness) information is normally not part of the test set.

[51] Knowledge about the form of the nonzero part of $p(\hat{f}|f, \hat{a}, q_r)$ is, in principle, of no help in searching for the absolute minimum, as the minimization is over all $\hat{f}$ even the unlikely ones, and only the impossible ones can be excluded. Exceptions are cases are where (also subsequent) knowledge of risk values for some $\hat{f}$ can be used to exclude others, i.e. if nonlocal information about the risk functional can be used to exclude certain possibilities. For example in the case of a decision problem with known minimal value, or if one is not looking for the absolute minimum but only for an acceptable minimum, checking $\hat{f}$ with high probability first is of course a good idea. Also in other cases one may ignore $\hat{f}$ with small probability so that the chance of missing the optimum is small. This is a problem on the level of comparing approximations to a decision problem and its analysis requires a corresponding risk and loss to be defined.

to a possibly unknown but reproducable state of knowledge one may also empirically estimate indirect priors by counting the results of learning algorithms.[52]

While some restrictions for $\hat{f}$ like the range of possible output values or specific symmetries are easily implementable, others, e.g. smoothness, are best taken into account by using the method of Lagrange multipliers. (For the exact conditions under which this is possible see for example Bertsekas, 1995.) Formulating the restriction in the form $q_a(\hat{f}) = a$ this means constructing an extended risk $\bar{r}$ by adding the following extra term to the risk

$$\lambda(a)(q_a(\hat{f}) - a).$$

Here λ is the a–dependent Lagrange multiplier, the term $-\lambda a$ can be skipped because $\hat{f}$–independent, and a, q_a and therefore λ can be vectors. Note that for a given problem, including data, a determines λ. For unknown a determination of a by cross–validation (see next Subsection) can be seen similar to imposing a prior on a.

Now we shortly discuss how sometimes additional nonlocal terms in the risk can be approximated by using virtual examples. If $q_a = \int dx\, p(x|q_a) q_a(x, \hat{f})$ is a sum or integral it might sometimes be easier in practice to use only part of the sum if the generalization ability is ensured in another way, for example by restriction of $\hat{F}$. Consider a deterministic $\hat{f}(x)$ with a symmetry or smoothness logprior

$$\lambda(a) q_a = \lambda(a) \int dx\, w_{q_a}(x) (\hat{f}(x) - \hat{f}(sx))^2.$$

Then one can sample x–values according to $w_{q_a}(x)$ if those are positive and can be normalized. This sampling can be done independently of the sampling of the training examples according to $p^S(x)$ (for $x = q^D \in Q^D$). Thus, the term can be approximated by sampling x and calculating $\hat{f}(x)$ and $\hat{f}(sx)$.

For a quadratic loss $\tilde{l}^D(x, y, \hat{f}) = (y - \hat{f}(x))^2$ and some b constant with respect to x with $b(a) p^S(x) = \lambda(a) w_{q_a}(x)$ the sampling for the data and symmetry terms can be combined. As λ depends on a so also does b. With $\tilde{l}^0 = \lambda(a) q_a$ and splitting the data terms in the empirical risk

$$\hat{r} = (1 - b(a)) \hat{r}^D(f^0, \hat{f}) + b(a) \hat{r}^D(f^0, \hat{f}) + \tilde{l}^0.$$

The integrations over x and y do not affect $\tilde{l}^0$ being independent of those variables, and we get for the last two parts

$$b(a) \int dx \int dy\, p^S(x) p(y|x, f^0) \left(\tilde{l}^D + \tilde{l}^0\right)$$

$$= b(a) \int dx \int dy\, p^S(x) p(y|x, f^0)$$

$$\times \left((y - \hat{f}(sx))^2 - 2(y - \hat{f}(x))(\hat{f}(x) - \hat{f}(sx))\right).$$

When the second term in the last line vanishes, the whole integral can therefore be calculated approximately by using *virtual (input) examples*[53] $(sx, y(x))$ Abu–Mostafa, 1990, 1993a, 1993b ('hints'), Pomerleau, 1991 (ALVINN), Sietsma & Dow, 1991 (training with noise in practice), Vetter, Poggio, & Bülthoff, 1992 (virtual views of an object), Girosi & Chan, 1995 (for RBF), and more theoretical Webb, 1994, Leen, 1995, and Bishop (1995ab) who gives the explicit form of regularization terms, for quadratic and for cross–entropy error functions, for infinitesimal translations, by expanding $\hat{f}(sx)$ in a Taylor series around $\hat{f}(x)$. When $\hat{f}(x) \neq \hat{f}(sx)$ then all nonoptimal $\hat{f}(x)$ are not equal to the regression function. For those $\hat{f}(x)$ the second term does not vanish and has to be considered in the actual minimization procedure. But when $(\hat{f}(x) - \hat{f}(sx))$ is zero or the optimal regression function is in $\hat{F}$ the second term vanishes at the minimum.

Stratification: (cross–)validation and structural risk minimization

A trivial toy example may clarify the basic idea: Assume two deterministic questions x_1 and x_2 and a set of possible answers $Y = \{0, 1\}$ corresponding to the four possible functions $\hat{f}_i$ characterized by their answers $(\hat{f}_i(x_1), \hat{f}_i(x_2))$: $\hat{f}_1 : (0,0)$, $\hat{f}_2 : (1,1)$, $\hat{f}_3 : (1,0)$, $\hat{f}_4 : (0,1)$. Let us sample data D until we have answers to both questions, for example $x_1 = 0$ and $x_2 = 1$. Clearly, $\hat{f}_4$ would minimize the mean square error. Now we perform the same minimization in a *hierachical* way. We form the two groups (strata) of smooth functions $S_1 = \{\hat{f}_1, \hat{f}_2\}$ and nonsmooth functions $S_2 = \{\hat{f}_3, \hat{f}_4\}$ and use the first data point, for example $x_1 = 0$ to minimize within the two groups finding $\hat{f}_1$ and $\hat{f}_4$. To decide between the optima of the two groups we sample more data until we get $x_2 = 1$. Again, we choose $\hat{f}_4$. Note that also forming nondisjunct, overlapping groups, e.g. $S_i \subset S_{i+1}$, would lead to the same result as long as they include all four functions. However, a stratified search is not always equivalent, to a full search. The difference is, that the new data are only used to decide between the 'winners' of the strata, and the 'looser' are not reconsidered again. They may however better fit the complete data.

In the case of too large sets $\hat{F}$ the difference between the minima of the expected (training) risk for the empirically chosen and the optimal $\hat{f}$ can become too large (Vapnik, 1982) and solutions can depend too strongly (noncontinuously) from the data (Tikhonov, 1963). Than it is necessary to restrict the minimization to a simpler subtask (regularization).

Subsets or strata are defined by some deterministic question q_a requiring $q_a(\hat{f}) = s$. Then we search for the minimum first within the strata and compare the best solutions of different strata in a second step. We want

[52] Refering to practical experience or literature about the use of specific priors approximates this method.

[53] In this special case the input sx is newly generated, while the same target y for x is used again for sx. We also used the term virtual examples in the Bayesian approach for sampling from the posterior probability $p(y|q, f)$ where for risk minimization both, q (e.g. x) and y, are generated according to the posterior.

to select q_a so that minimization within each stratum s is possible[54]. Selecting a specific stratum is equivalent to implementing an indirect prior, discussed in the last paragraph. Practically, it can be done by direct restriction of function values like using hardwired symmetries, or, more indirectly, by restricting parameter sets. Examples include choosing the number of nodes or the initial values of the weights of a neural network or the learning algorithm used. This can create overlapping strata, but this is no principal problem and only means that some solutions are considered more than once in the search process. (This is indeed the normal case for nonlocal q^a like smoothness, where the value of q^a can be increased without changing any function value at a data point.) All these constraints are easily kept constant during the minimization (learning) process.

Some constraints like smoothness might be difficult to implement directly. Then again, this minimization with constraints can be done by the method of Lagrange multipliers producing an extra term to be added to the empirical risk

$$\lambda(s)(q_a(\hat{f}) - s).$$

The only difference to the preceding paragraph is that the optimal value of λ is not yet determined. The comparision between strata to find λ (and therefore s) has to be done with an independent data set. That means we have to separate the available training data in a training set to be used within the strata and a training set to decide between the strata also called validation set[55]. Repeating the procedure several times with the same total data set but with different splittings into training and validation set is called *cross–validation* (Stone, 1974, 1977, Allen, 1974). A stratification method for strata where the VC dimension can be calculated is the method of structural risk minimization where the validation step is replaced by minimizing the worst case empirical risk. Fig.9 illustrates that the selection of proper strata depends on indirect priors (Wolpert, 1994b, Ripley, 1996) for the example of structural risk minimization.

In general the minimization within some specific strata (e.g. for those strata including functions where a given smoothness functional is not even defined) or also between strata can be very difficult or even impossible. To make those problems solvable an indirect prior for $\hat{f}$ must be available to restrict the number of strata and also to enable minimization between strata. Practically, also the maximal number of different $\lambda(s)$ which can be considered is normally at least restricted by the available computational resources. Therefore, there is more what choosing a good stratification variable q_s can do. It can make the minimization problem between the strata easy (for one or more algorithms under consideration). This is an algorithmic specific aspect not directly related to prior information about f^0. It means that $q_a(\hat{f})$ approximates already relevant f–independent aspects of $r(f, \hat{f})$. And indeed the most figures found in the literature plotting the empirical error against, for instance, some smoothness related s or $\lambda(s)$ show a very simple one–minimum structure. In principle, even for restricted range of values for s such a function could look arbitrarily wild having for example a multiminima or even random–like structure.

If information is available which restricts the range of s this is a conceptual clear case of an indirect prior. However what one usually does is more like the following: begin with a starting value s_0 and explore every of its components in a given direction until the first common minimum is found and stop there. That is, one assumes a certain form of s–dependence of the risk $r(f, \hat{f}^*(s))$ for the $\hat{f}^*$ optimal for s.

8.5 Bayesian interpretation of the Frequentist approach

8.5.1 Approximation and nonapproximation loss

We defined an integrated loss function $\tilde{l}(q, y, \hat{f}) = \int d\hat{y} p(\hat{y}|q, \hat{f}) l(q, y, \hat{y})$ for test questions with $p(q|y_c, z_c) = p(q|y_c)$ or simply $p(q)$ as we will choose for the Frequentist setting. This integrated loss function has the same arguments as a logposterior, except that $\hat{f}$ and f^0 are exchanged.

To interpret the Frequentist approach from a Bayesian point of view we choose the same parameter space for $\hat{F}$ and F^0. This corresponds to a one–to–one mapping, which we call *parameter mapping*, between $\hat{F}$ and F^0 which we can use to identify

$$\hat{F} = F^0.$$

Specifically, $f^0 = \hat{f} \Leftrightarrow \forall q : f^0(q) = \hat{f}(q)$ when f^0 can be parametrized by a deterministic function $f^0(q)$. The *optimality mapping* for the risk functional $\hat{f}(f^0)$ $=\text{argmin}_{\hat{f}} r(f^0, \hat{f})$ defines another mapping between F^0 and $\hat{F}$. We already discussed that a one–to–one optimality mapping is obtained if f^0 which lead to the same optimal $\hat{f}$ can be identified, and by excluding $\hat{f}$ which are for no f^0 optimal. For such a construction the f^0 related to the $\hat{f}$ by the optimality mapping can be seen as effective states of nature for the decision problem involving $\hat{f}$.

We called a loss function approximation loss if for all $\hat{f} = f^0$ (See Subsection 5.3.4)

$$\tilde{l}(q, y, \hat{f}) = -c_1 L^D(y|q, f^0) - c_0,$$

(with some $\hat{f}$–independent constants c_0, c_1 adjusting the value of the minimum of $\tilde{l}$ or the normalization constant of L^D). Equivalently we may say in this case

$$\tilde{l}(q, y, \hat{f}) = \tilde{l}^A(q, y, \hat{f}),$$

[54] Low VC dimension or ϵ–entropy in the theory of uniform convergence or compact for a positive real s and continuous one-to–one extremal equation in regularization theory. See (Verri & Poggio, 1986) for examples for q_a. The existence of a stable, unique solution is the result of practical interest in regularization theory. For finite, noisy cases the convergence theorems are not of so much practical value.

[55] The estimation of the true expected risk after the cross–validation procedure would require a third set of available data, the empirical test set.

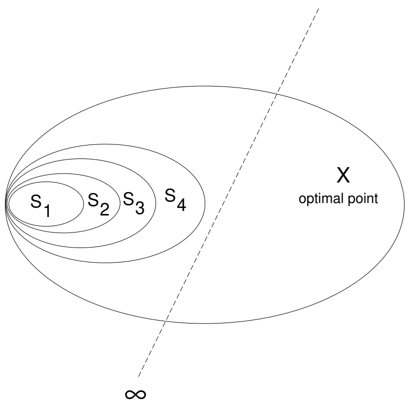

Figure 9: The VC horizon: Consider a set S with a subset of infinite VC dimension. If the sequence of sets $S_1 \subset S_2 \subset S_3 \subset S_4 \subset \cdots S_n$ is chosen unluckily, then S_n approaches for $n \to \infty$ infinite VC dimension before the optimal x is found. On the other hand, choosing for example $S_1 = \{x\}$ to be (or include) x, the optimal solution is already found in the first step. The final uniform VC bound is lower, if x is found earlier in a smaller S_i, and depends therefore also on the choice of the sequence S_i. Deterministic prior information defines the set S however not a chain of sets S_i. As elements with zero prior probability are excluded and elements with nonzero, but possibly extremly small prior probability are not, the size of S can be very sensitive to very small differences in the probability if they change deterministic into probabilistic prior information. (This kind of prior information $p(\hat{f})$ has the form of an indirect prior, i.e. it is about optimal actions and not states of nature. Only in approximation problems, if an interpretation as a Bayesian maximum posterior calculation is possible (See Section 8.5) and we can identify $f^0 = \hat{f}$, this is also a direct prior $p(f^0)$ for state f^0.) If S is large, a prior free construction of the chain S_i by uniform sampling will with high probability not contain good candidates. To yield reasonable results the sequence of S_i has to be chosen depending on the probability that S_i contains the optimal solution, i.e. probabilistic prior information. This increases the chance of testing good candidates, if the prior probability is nonuniform. The uncertainty of probabilistic prior information used to construct the chain S_i is not included in the uniform bounds. The prior information can be the result x of a previous structural risk minimization. Then choosing $S_1 = \{x\}$ for a subsequent minimization would set the bound to its possible minimum, obviously to small. Usually, prior knowledge may not be the result of a previous structural risk minimization, but of other previous experiences, like reported success with certain types of smoothness or other architectural constraints (e.g. tree structures, sigmoidal multilayer neural networks, radial basis functions) for similar tasks. Yet the effect of any optimality related probabilistic preselection on the bounds is in principle the same. Practically, this underestimation of the possible deviation between empirical and expected risk according to probabilistic prior information may in most cases not be very important, because compensated by the worst case assumptions on which the bounds are based.

if we define the approximation part of the loss as

$$\tilde{l}^A(q, y, \hat{f}) = -c_1 L^D(y|q, f^0) - c_0.$$

In general this gives a (c_1, c_0–dependent) decomposition of the loss function into an *approximation* and *nonapproximation part of the loss*

$$\tilde{l}(q, y, \hat{f}) = \tilde{l}^A(q, y, \hat{f}) + \tilde{l}^{NA}(q, y, \hat{f}).$$

Analogously, we define the (loss function dependent) set Q^A of *approximation questions* to consist of all questions q for which for all $\hat{f} = f^0$ and all y we can choose ($\tilde{l}(q, y, \hat{f})$ as function of q is only defined up to q–independent constants, which include for example prior costs $\tilde{l}(\hat{f})$)

$$\tilde{l}(q, y, \hat{f}) = -c_1 L^D(y|q, f^0) - c_0. \tag{15}$$

To achieve a parallel notation for loss and loglikelihoods we will use in this case the convention $l^A(q, y, \hat{f}) = l^D(q, y, \hat{f})$ for $q \in Q^S$, i.e. we choose the sampled loss equal to the approximation part, if possible.

For those questions a large logposterior for f^0 is equivalent to a small loss for $\hat{f}(f^0)$ which characterizes the situation for q in a parallel decision problem as an approximation problem. In an inverse setting this is not necessarily true as for example a loss $l^D = (q - \hat{f}(y))^2$ measuring the reconstruction quality of $\hat{q} = \hat{f}(y)$ would correspond to a logposterior $L^D = (q - f^0(y))^2$ instead of $L^D = (y - f^0(q))^2$.

The decomposition of the loss function into an approximation and nonapproximation part of the loss induces the same decomposition for the risk

$$r(f, \hat{f}) = r^A(f, \hat{f}) + r^{NA}(f, \hat{f}).$$

Especially common is the case with prior costs and $Q^S \subseteq Q^A$ for which we have (using our convention for $\tilde{l}^D$ in such cases)

$$\begin{aligned}
\tilde{l}(q, y, \hat{f}) &= \tilde{l}^A(q, y, \hat{f}) + \tilde{l}^{NA}(\hat{f}) \\
&= \tilde{l}^A(q, y, \hat{f}) + \tilde{l}^0(\hat{f}) \\
&= \tilde{l}^D(q, y, \hat{f}) + \tilde{l}^0(\hat{f}).
\end{aligned}$$

Prior costs have the form of a fixed additional term implementing restrictions within $\hat{F}$ according to the method of Lagrange. The following table summarizes decompositions of loss, risk, and logposterior:

Decompositions

Sampled vs. Nonsampled	
Logposterior	$L(D, D^0, f^0) = L^D(D, f^0) + L^0(D^0, f^0)$
	$= \sum_i L^D(y_i\|q_i, f^0) + L^0(D^0, f^0)$
risk	$r(f, \hat{f}) = r^D(f, \hat{f}) + r^0(f, \hat{f})$
Approximation vs. Nonapproximation	
risk	$r(f, \hat{f}) = r^A(f, \hat{f}) + r^{NA}(f, \hat{f})$
loss	$\tilde{l}(q, y, \hat{f}) = \tilde{l}^A(q, y, \hat{f}) + \tilde{l}^{NA}(q, y, \hat{f})$
Special case: $Q^S \subseteq Q^A$ with prior costs	
loss	$\tilde{l}(q, y, \hat{f}) = \tilde{l}^A(q, y, \hat{f}) + \tilde{l}^0(\hat{f})$
	$= \tilde{l}^D(q, y, \hat{f}) + \tilde{l}^0(\hat{f})$

We now analyze the MaP–MiR procedure. To find the optimal $\hat{f}^*$ in a MaP–MiR approximation the first MaP step is followed by a second MiR step to find $\hat{f}^* = \text{argmin}_{\hat{f}} r(f^0, \hat{f})$ for the most probable state $f^0 = f^{0,*}$. Despite the one–to-one mapping between F^0 and $\hat{F}$ and the same functional dependence of L^D and $-\tilde{l}^D$ for $q \in Q^A$ there is no perfect symmetry between the two problems as the maximization is done with respect only to a finite sum over the set of data D and the minimization with respect to the full y–, and q–integrals conditioned on the pure state $f^{0,*}$ with maximal posterior probability. Only if the mapping between $f^{0,*}$ and $\hat{f}(f^{0,*})$ already corresponds to the optimality mapping

$$\hat{f}(f^{0,*}) = \hat{f}^* = \text{argmin}_{\hat{f}}\, r(f^0, \hat{f})$$

$$= \text{argmin}_{\hat{f}} \int dq \int dy\, p(q) p(y|q, f^0) \tilde{l}(q, y, \hat{f}),$$

with

$$p(y|q, f^{0,*}) = e^{L^D(y|q, f^{0,*})}$$

then finding the state with maximal posterior probability already corresponds to minimizing the loss for it.

We already saw in Subsection 5.3.4 that for approximation problems the relative risk

$$c_1^{-1}(r(f^0, f^0) - r(f^0, \hat{f})) = K(p(y|q, f^0), p(y|q, \hat{f})),$$

is a Kullback–Leibler entropy and using Jensen's inequality that the optimal solution for the minimization step is

$$\tilde{l}^A(q, y, \hat{f}^*) = -c_1 L^D(y|q, f^{0,*}) - c_0.$$

We derive this well known result here again using an explicit calculation. For continuous parameter spaces $\hat{f}$, a necessary optimality condition for minima not on the boundary is a vanishing gradient with respect to the parameter vector $\hat{f}$

$$\frac{d\, r(f^{0,*}, \hat{f})}{d\hat{f}}\Big|_{\hat{f}^*} = 0.$$

We study the stationarity condition for that part of the loss integration depending on approximation questions, i.e. with $q^l \in Q^A$,

$$\frac{d}{d\hat{f}} \int_{Q^A \subseteq Q^l} dq \int dy\, p(q) e^{L^D(y|q, f^{0,*})} \tilde{l}^D(q, y, \hat{f})\Big|_{\hat{f}^*} = 0. \tag{16}$$

Interchanging integration and differentation and using $-c_1 L^D(y|q, f^0) - c_0 = \tilde{l}^D(q, y, \hat{f})$ at $f^{0,*} = \hat{f}^*$ gives

$$\int dq\, p(q) \int dy\, e^{L^D(y|q, f^{0,*})} \frac{d\tilde{l}^D(q, y, \hat{f})}{d\hat{f}}\Big|_{\hat{f}^*}$$

$$= -c_1 \int dq\, p(q) \int dy\, e^{L^D(y|q, f^{0,*})} \frac{dL^D(y|q, f^0)}{df^0}\Big|_{f^{0,*}}$$

$$= -c_1 \frac{d}{df^0} \int dq\, p(q) \int dy\, e^{L^D(y|q, f^0)}\Big|_{f^{0,*}}$$

$$= -c_1 \frac{d}{df^0} \int dq\, p(q) \int dy\, p(y|q, f^0)\Big|_{f^{0,*}}$$

$$= -c_1 \frac{d}{df^0} 1 = 0.$$

That means that for $Q^l = Q^A$ the stationarity conditions of the MiR step are automatically fulfilled if $\hat{F} = F^0$.[56] (The same is true for any f in a full Bayesian approach if $\ln p(y|q, f) = -\tilde{l}(q, y, \hat{f}) + c_0$.) For $Q^l = Q^A$ the MaP–MiR solution is not altered if $p(q)$ is changed (for fixed data). In particular, terms like $r^A(f^0, \hat{f})) = \int dq\, p(q) \int dy\, p(y|q, f^0) l^D(y|q, \hat{f})$, with $l^A(y|q, \hat{f}) = l^D(y|q, \hat{f}) = -c_1 L^D(y|q, f^0) - c_0$ for every $\hat{f} = f^0$, can be added or dropped from the risk (with p(q) readjusted).[57] As long that term is f^0–dependent it cannot be seen as part of the (f^0–independent) loss, i.e. it cannot be written as $\tilde{l}(\hat{f})$. In this situation approximating one point is equivalent to approximating a whole function. This could also be done for the prior part L^0 writing it in its data dependence, giving rise to cost terms $l^0(f^0, \hat{f})$. The nonapproximation part of the loss depending on questions $q \in Q^l \setminus Q^A$ can cause a deviation between optimality mapping and parameter mapping. Examples of typical nonapproximation loss can include

1. time and storage requirements for calculating $\hat{f}$,
2. costs producing a hardware (VLSI) implementation of $\hat{f}$,
3. understandability of the structure of $\hat{f}$.

As discussed earlier nonapproximation loss related to the algorithm and not to $\hat{f}$ has to be included by defining a higher level decision problem.

The typical example for approximation questions combines Gaussian noise with mean square error. Also, a question is an approximation question if a uniform L is combined with a uniform l finite on the same domain. Here logprobabilities L (and analogously costs $\tilde{l}^0$) are called uniform on F^0 (or $\hat{F}$) if they are equal to a constant, i.e. independent of $\hat{f}$ (or f^0), on the domain where they are finite. An example are priors implemented by regularization terms to be determined by cross–validation with restricted intervall for the regularization constant. Uniform priors can be skipped from the formalism by restricting the parameter spaces F^0 to the domain on which the logpriors are finite. The corresponding $\hat{f}(f^0)$ related to f^0 with zero prior probability by the optimality mapping can also be skipped. This is equivalent to the introduction of a uniform cost term. If

[56] For example minimizing a L_1 error (sum of unsquared distances) for Gaussian probabilities the MiR condition is not automatically fulfilled. An example is the support vector machine (Vapnik, 1995) for classification which uses a L_1 type of error in the nonseparable case (which might e.g. result from noise). There the function to minimize consists of two terms, the norm of the weights of the optimal canonical hyperplane $||w||^2$ and the L_1 error, whose relative importance C has to be determined for example by cross–validation or prior knowledge. (Individual C_i for each data point i could account for locally differing variances.)

[57] One could paraphrase this observation by: "One always can require what is already there."

their is no nonapproximation part of the loss the optimality mapping is equal to the parameter mapping and trivially implemented by using the same restrictions for F^0 and $\hat{F}$.

While we saw that risk (not necessarily loss function) terms corresponding to loglikelihoods can be added to the risk integral without changing the problem, (nonapproximation) loss terms on the other hand cannot simply be implemented in the logposterior.[58] For example, uniform prior costs, equivalent to a restriction of $\hat{F}$, do not lead directly to a restriction of F^0. In principle one could regroup the F^0 by forming equivalence classes with respect to the restricted $\hat{F}$, but this may destroy the relation (15).

In general, for the MaP step being sufficient

$$\frac{d\tilde{r}^{NA}(f^{0,*},\hat{f})}{d\hat{f}}\Big|_{\hat{f}=f^{0,*}} = 0$$

must hold. For prior costs $l^0(\hat{f})$ being f^0–independent this reads

$$\frac{d\tilde{l}^0(\hat{f})}{d\hat{f}}\Big|_{\hat{f}=f^{0,*}} = 0$$

for every f^0 if $\tilde{l}^D$ is approximation loss. This allows special cases where $\tilde{l}^0(\hat{f})$ has a minimum $\hat{f}=\hat{f}^*$ at the maximum $f^0 = f^{0,*}$ of the posterior $L^D + L^0$, i.e. $f^{0,*} = \hat{f}^*$, and $\tilde{l}^D(\hat{f})$ coincides with an approximation risk term $r^A(f^{0,*},\hat{f})$. Then, even nonuniform costs do not change the optimal $\hat{f}$. However, this can only happen for specific f^0, as $\tilde{l}^0(\hat{f})$ is f^0–independent and we assume the MaP estimate to be data dependent. Otherwise calculating the MaP approximation would not be interesting. Thus, requiring that, depending on the possible data, every f^0 can be the most probable one, i.e. be a MaP estimate $f^{0,*}$, the derivative of the cost term must be zero at every $\hat{f} = f^{0,*}$

$$\frac{d\tilde{l}^0(\hat{f})}{d\hat{f}} = 0.$$

Then, $\tilde{l}^0$ is uniform. Because nonuniform, data or f^0–independent costs belong to the nonapproximation part of the loss their presence require the full two step MaP–MiR procedure.[59] An example would be using complexity costs to enforce simplicity of a model $\hat{f}$ (e.g. smoothness, sparseness, integer values) independent of the data, maybe even when knowing that this is not true for nature f^0 (which might for example allow real values).

8.5.2 MaP–MiR and ERM: Priors and prior costs

Often, complexity related prior costs, are included in empirical risk minimization, either by explicit penalty terms, or by choosing a specific structure for hypotheses $\hat{f}$. If those complexity aspects are requirements not related to priors ERM cannot be interpreted as MaP-MiR procedure. For example, a tree classifier might be choosen because it can be obtained rather effectively and/or because the resulting rules are relatively easy to interpret. If, there is no prior knowledge about F^0 having tree structure, and at the same time there is a more appropriate parametrization of F^0 available, then this one should be used in the MaP step while a tree classifier could be fitted in the MiR step.

We use the results from the previous paragraph to discuss in more detail the relations between the Bayesian and Frequentist point of view of empirical risk minimization in the presence of priors and prior costs (in the following shortly called costs). Consider the three problems:

1. Maximization of the posterior probability (MaP) given data D
$$\text{argmax}_{f^0}\left(\sum_i^n L^D(y_i^D|q_i^D f^0) + L^0(f^0)\right),$$
2. empirical risk minimization (ERM) given data D
$$\text{argmin}_{\hat{f}}\left(\frac{1}{n}\sum_i^n \tilde{l}^D(q_i,y_i,\hat{f}) + \tilde{l}^0(\hat{f})\right),$$
being a sample estimate of
3. a full Bayesian risk minimization (MiR) for fixed f^0

$$\text{argmin}_{\hat{f}}\left(\int_{Q^S} dq \int dy\, p(q)p(y|q,f^0)\tilde{l}^D(q,y,\hat{f}) + \tilde{l}^0(\hat{f})\right).$$

Within the MaP–MiR procedure the fixed f^0 for the MiR step is the result $f^{0,*}$ of the MaP step while for ERM f^0 is thought to be the true state of nature. We assume the sampling set to consist of approximation questions i.e. $Q^S \subseteq Q^A$ or equivalently $-c_1 L^D(y|q,f^0) - c_0 = \tilde{l}^D(q,y,\hat{f})$ for $q \in Q^S$, choosing in the following $c_1 = 1$ for simplicity. Consider the following cases in which arbitrary constants c_0', c_0'' exist, so that

A: ('uniform costs and uniform priors')
$$\tilde{l}^0(\hat{f}) + c_0' = -L^0(f^0) + c_0'' = 0,$$

B: ('uniform costs but nonuniform priors')
$$\tilde{l}^0(\hat{f}) = c_0', \qquad L^0(f^0) \neq c_0'',$$

C: ('nonuniform costs')
$$\tilde{l}^0(\hat{f}) \neq c_0',$$

D: ('costs $\propto -$ priors')
$$n\,\tilde{l}^0(\hat{f}) + L^0(f^0) = c_0',$$

[58] In contrast to the previous footnote this could be paraphrased by: "Reality is not always like one wants it to be."

[59] This means as soon as one knows model $\hat{f}$ and nature f^0 are not the same one should use two different descriptions for both. Otherwise the knowledge about their difference cannot be used.

In case A we could choose $\tilde{l}^0 = 0$ by including the constant into $\tilde{l}^A = \tilde{l}^D$. It is the case with all priors and related prior costs already implemented as restrictions of F^0 and $\hat{F}$. Here numerical realization of ERM and the MaP step are identical and their interpretation from the Frequentist and Bayesian point of view are fully compatible because a MiR step is not needed. F^0 and $\hat{F}$ can be fully identified and there is no need to use a distinct notation for f^0 and $\hat{f}$.

In case B we also can choose $\tilde{l}^0 = 0$ so no MiR step is necessary, but the MaP step uses logprior terms if available. As those terms are not part of the loss function ERM should in principle ignore them. Namely, the nonuniform parts of priors do not enter the procedure of ERM or the (worst case) bounds for uniform convergence. Using the priors nevertheless as cost terms in ERM leads to complete numerical equivalence with the MaP step and therefore the whole MaP–MiR procedure. In this sense priors for Bayesians are related to costs for Frequentists. Note, however, that while here the numerical calculations coincide, they are interpreted as different models of nature, as costs cannot be identified with priors.

Case C requires the MiR step. Therefore, in this case the two step MaP–MiR and the one step ERM differ. Exceptions are MaP results $f^{0,*}$ for which the nonuniform costs have a minumum. (For uniform costs all $\hat{f} \in \hat{F}$ are minima.) MaP–MiR incorporates priors and takes into account the differences between F^0 and $\hat{F}$. This might be especially important if they differ strongly because priors and costs are related to different aspects. MaP–MiR is expected to improve ERM in situations where the prior had a strong influence compared to the sampled data in the MaP step and/or the cost term is substantial compared to the data term and related to aspects different from those of the prior. The same remarks apply when the sampled loss $\tilde{l}^D(y|q, f^0)$ itself is nonapproximation loss (this means according our convention it cannot be chosen as approximation loss), i.e. $Q^S \notin Q^A$.

Case D seems to show a perfect symmetry between MaP and ERM. Indeed, under these conditions ERM and MaP are numerically identical for the specific n

$$\text{argmax}_{f^0}\left(\sum_i^n L^D(y_i|q_i, f^0) + L^0(f^0)\right).$$

$$= \text{argmin}_{\hat{f}}\left(\frac{1}{n}\sum_i^n \tilde{l}^D(q_i, y_i, \hat{f}) + \tilde{l}^0(\hat{f})\right).$$

But in as far as nonuniform costs are present the MaP step is not sufficient from a Bayesian viewpoint and the MiR step is missing. Indeed, for case D the MiR step takes into account the same function $L^0(f^0)$ again but in form of costs $\tilde{l}^0(\hat{f})$ so the related aspects have a stronger influence in MaP–MiR than in ERM. This means, that, if not already in the minimum, a function $\hat{f}^* \neq f^{0,*}$ can be chosen which a lower $\tilde{l}^0(\hat{f}^*)$ than $\tilde{l}^0(\hat{f} = f^{0,*})$ of the result $f^{0,*}$ of the ERM or MaP step. One may expect this effect usually to be small in practice, as the saddle point approximation assumes a strongly peaked maximum for $L^D + L^0$, usually arising in the limit $n \to \infty$ where case C becomes case B or, for uniform priors, case A.

But note that the theory of uniform convergence does not require all the conditions necessary for a Bayesian interpretation of ERM as long as the training data are sampled according to the (arbitrary) relevant distribution. It applies for general $\tilde{l} \neq -c_1 L^D - c_0$ and also if $\hat{F}$ and F^0 are chosen different, like in many examples of computational learning theory. Costs restricting the search space $\hat{F}$, and changing for example its VC dimension, influence the bounds of the theory of uniform convergence. Its bounds do not depend on the form of the finite parts of logpriors L^0 as they are worst case considerations. Prior costs $\tilde{l}^0$ do not contribute to the difference between empirical and expected risk (if they are not sampled itself), as they are data independent and included in both. The bounds only depend on the infinite part of prior costs $\tilde{l}^0$ restricting the space $\hat{F}$ which can for example reduce the VC dimension. But in most of these cases ERM differs in method and results from a MaP–MiR approach.

We can summarize the results by saying that for $F^0 = \hat{F}$ uniform costs allow an interpretation of the numerical ERM procedure as two step MaP–MiR procedure for approximation loss so that the MiR step is automatically satisfied. On the other hand, training data being sampled according to the test data distribution allow application of the bounds of the theory of uniform convergence to the results of empirical risk minimization.

For example, take a prior $p(f^0)$ depending on a symmetry property like $S(x, f^0) = (f^0(x) - f^0(-x))^2$. If we choose a uniform prior which is zero for f^0 if $S(x, f^0)$ is above a bound B and constant for those f^0 below that bound, we can implement the prior as restriction on F^0 by excluding f^0 with $S(x, f^0) > B$. Through the optimality mapping this corresponds to the same restriction for the $\hat{f} \in \hat{F}$. This is the usual case.

Let us briefly write down the common example of a (one–dimensional) Gaussian probability

$$p(y|q, f^0) = \frac{1}{\sigma_q\sqrt{2\pi}} e^{-\frac{(y - f^0(q))^2}{2\sigma_q^2}},$$

namely a logposterior $L(y, f^0(q)) = \ln \sigma_q\sqrt{2\pi} - (1/2)\sigma_q^{-2}(y - (f^0(q))^2$ corresponds to a quadratic loss function $l(y, \hat{f}) = (1/2)\sigma_q^{-2}(y - \hat{f}(q))^2$ with $\hat{f}(q)$–independent $\sigma(\hat{f}(q)) = \sigma_q$ skipping the constant. Note, that $f^0(q)$ is no random variable but parametrizes the states f^0 and corresponds to the regression function. As the regression function minimizes the mean square error we have $\hat{f}(f^0) = \text{argmin}_{\hat{f}} r(f^0, \hat{f})$ or written explicitly the optimality condition (16) gives

$$\int dy\, p(y|q, f^0) y = \hat{f}(q).$$

because $d\sigma_q/d\hat{f}(q) = 0$ if all $\hat{f}$ have the same $\sigma(\hat{f}(q)) = \sigma_q$. Inserting the form of $p(y|q, f^0)$ and performing the Gaussian integration we find again that the optimality condition is fulfilled. As $\hat{f}(q)$ represents the regression function all deterministic information about the regression function can be incorporated as indirect prior, that is by restricting the search space $\hat{F}$. This holds in particular for a deterministic bound on the smoothness of the regression function. On the other hand for nonuniform costs, like for example higher costs for states with regression far from zero $\tilde{l}^0(f^0) \propto \sum_q f^0(q)^2$, the two step MaP–MiR approximation is not equivalent to an empirical risk minimization even if we choose an prior $L^0 = \tilde{l}^0$. Fig.10 visualizes some of the relations.

We summarize how the classical Frequentist approach of empirical risk minimization with additional (e.g. regularization or penalty) terms can be interpreted as a specific Bayesian model. This 'classical' Bayesian model has the following specifications:

1. Definition of an effective loss function $\tilde{l}$ for z–independent generation of test questions.
2. Identification of the (parameter) space of actions $\hat{f}$ with that of states f^0.
3. The same function, up to a factor and a constant, is chosen for the y–dependent parts of the effective loss $\tilde{l}^D(q, y, \hat{f})$ and loglikelihood $L^D(y|q, f^0)$ depending on the same variables after identifying $\hat{f}$ with f^0. (Sampled data come from approximation questions, $Q^S \subseteq Q^A$.)
4. There are no nonuniform costs $\tilde{l}^0(\tilde{f})$.
5. The decision relevant risk functional is the expectation functional (Bayesian expected risk).

Under these conditions empirical risk minimization corresponds to an exact risk minimization (i.e. no plug–in estimate) for the state with maximal posterior probability, regardless of how the training questions have been sampled.

Remarks on point 3:

i. It characterizes the situation as an approximation problem in a parallel decision setting.

ii. Note that AND and OR are exchanged in the following sense: When y_1 AND y_2 has been observed as training data then we assume y_1 OR y_2 can appear in a test situation and both logposterior and loss consist of a sum. If we only know y_1 OR y_2 could have been the training data this would require adding probabilities and not logprobabilities resulting in a nonadditive structure for L. A nonadditive loss function depending on more than one outcome at a time would have the different interpretation of an interaction of losses for repeated tests, i.e. for cases where y_1 AND y_2 happens.

In physics mean field approximations and classical approximations of field theories are related to saddle point approximations. The relation between the Frequentist approach as a maximum posterior approximation and the full Bayesian approach is for example similar to the relation between classical physics and the path integral formulation of quantum mechanics with a field being the analogon to a pure state f^0.

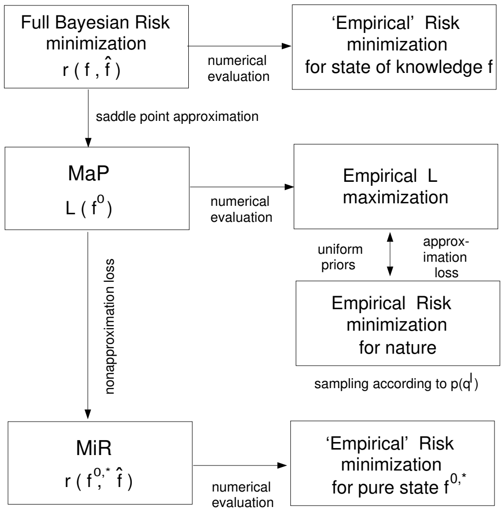

Figure 10: Numerical evaluation of the full Bayesian risk, for example by Monte carlo methods, is technically the same as an empirical risk minimization (i.e. a use of the plug–in principle) for virtual data generated by f. (One might call this also a virtual (empirical) risk, however it is a real empirical risk if state f is prepared as a mixture of f^0 according to $p(f^0|f)$.) A saddle point approximation of the full integral gives the MaP problem, which requires to find the state $f^{0,*}$ with maximal posterior probability. If the prior logprobability L^0 contains integrals (from nonlocal data) this may require a numerical evaluation of L, i.e. use of the plug–in principle. In the 'classical' model with uniform priors already implemented in $F^0 = \hat{F}$ and approximation loss, empirical risk minimization using the sampled data is equivalent to finding f^0 with maximal posterior probability. The bounds of the theory of uniform convergence require sampling according to the relevant distribution $p(q^l)$, which, however, can be arbitrary. For approximation loss, i.e. $\tilde{l} = -c_1 L^D - c_0$, the MiR step is not necessary.

8.5.3 MiR perturbation theory

The MiR step requires minimization of the expectation $<\tilde{l}> = <\tilde{l}^A + \tilde{l}^{NA}>$ under the distribution given by the MaP $f^{0,*}$. For a small enough 'perturbation' $\tilde{l}^{NA}$ we may expand $\tilde{l}(q,y,\hat{f})$ around $\tilde{l}(q,y,\hat{f}=f^{0,*})$. To change the location of a minimum we have to go at least to second order

$$\tilde{l}(q,y,\hat{f}) \approx \tilde{l}(q,y,f^{0,*})$$
$$+(\hat{f}-f^{0,*})\frac{d}{d\hat{f}}\tilde{l}(q,y,\hat{f})\big|_{\hat{f}=f^{0,*}}$$
$$+\frac{1}{2}(\hat{f}-f^{0,*})^2\frac{d^2}{d\hat{f}^2}\tilde{l}(q,y,\hat{f})\big|_{\hat{f}=f^{0,*}}.$$

Normalization is assumed to be ensured by the parametrization of f^0, otherwise for example a Lagrange multiplier can be added. The stationarity condition $\frac{d}{d\hat{f}} <\tilde{l}(q,y,\hat{f})>= 0$ is linear in second order. It gives for the parameter vector $\hat{f}$ the solution, assuming $c_1 = 1$

$$\hat{f}^* - f^{0,*} = -\frac{<\frac{d}{d\hat{f}}\tilde{l}^{NA}\big|_{\hat{f}=f^{0,*}}>}{<\frac{d^2}{d\hat{f}^2}\tilde{l}^A\big|_{\hat{f}=f^{0,*}} + \frac{d^2}{d\hat{f}^2}\tilde{l}^{NA}\big|_{\hat{f}=f^{0,*}}>}$$
$$= -\frac{<\frac{d}{d\hat{f}}\tilde{l}^{NA}\big|_{\hat{f}=f^{0,*}}>}{<(\frac{d}{d\hat{f}}\tilde{l}^A\big|_{\hat{f}=f^{0,*}})^2 + \frac{d^2}{d\hat{f}^2}\tilde{l}^{NA}\big|_{\hat{f}=f^{0,*}}>},$$

where $<\cdots>$ stands for the y and q integrals and we used to get the second line $\int dy\, e^L = 1$, and therefore for $c_1 = 1$ we have $\int dy\, e^L \frac{d}{d\hat{f}}\tilde{l}^A = 0$ and $\int dy\, e^L \left(\frac{d^2}{d\hat{f}^2}\tilde{l}^A - (\frac{d}{d\hat{f}}\tilde{l}^A)^2\right) = 0$ at the location $\hat{f} = f^{0,*}$.

8.6 Occam's Razor

Here is a good opportunity to discuss the celebrated principle of Occam's razor. Occam's razor states: If two theories explain the same phenomena equally well choose the simpler one. From a Bayesian point of view this simply corresponds to including complexity (prior) costs in the decision, but is sometimes also interpreted in the version: Simpler theories have higher prior probabilities.

This can be explained by the fact that empirical risk minimization has no possibility to include $\hat{f}$ dependent complexity in form of (prior) costs $\tilde{l}^0(\hat{f})$ independent of the priors $L(f^0)$. For uniform priors corresponding to uniform costs (case C) Occam's razor is automatically implemented in both versions. For nonuniform costs a second risk minimization step has to be included or if using a one step procedure Occam's razor has to be implemented via the priors. In contrary to the first case where the prior and cost versions of Occam's razor are equivalent, in the second case the Frequentist and Bayesian interpretations of the extra terms do not coincide. Then the prior version of the razor is the Bayesian interpretation of what appears for the Frequentists as the cost version. Relaying on the cost version as the intended one, a Bayesian approach approximated by a two step MaP–MiR approximation differs from a one–step Frequentist approach.

In principle, within a Bayesian framework the concept of costs and priors are independent.[60] A MaP–MiR approximation is possible with arbitrary costs independent of priors, when such costs are included in the second risk minimization step and the MaP step is still justified. For example, a sparsity constraint can come from the implementation of complexity costs related to computational costs for nonzero numbers or from the prior information that the actual data are produced by a small number of prototypes. Both aspects cannot be modeled independently using ERM but could be taken into account using the two step MaP–MiR procedure.

9 Stationarity equations

9.1 General data

In this section we study the stationarity equations for logposteriors. There are a variety of methods to find an extremum of the posterior probability. If those are nonlinear they have usually be calculated by iteration. Gradient based methods, or EM (Expectation–maximization) related algorithms (Dempster, Laird, Rubin, 1979, Tanner, 1993, Gelman, Carlin, Stern, Rubin, 1995) are special iteration schemes. In what we called a classical Bayesian model, with no additional nonuniform costs and $F^0 = \hat{F}$, logposterior maximization (MaP) already includes risk minimization (MiR). But a Bayesian MaP–MiR approximation is not restricted to a classical model. Assuming that the MaP step yields a good approximation of the Bayesian integral, performing a second independent risk minimization step after maximizing the posterior allows

1. states f^0 independently modeled from actions $\hat{f}$, that is $F^0 \neq \hat{F}$,
2. arbitrary costs $\tilde{l}^0$ independent of priors L^0,
3. an inverse setting with states defined by $p(y|q,f^0)$ and inverse actions $p(\hat{q}|y,\hat{f})$.

Here we discuss in more detail the case of Gaussian basis questions and states f^0 parametrized by their regression functions.

Let us now have a sharper look at the MaP approximation in the case of nonlocal questions. We calculate the functional derivatives separately for different classes of questions. We choose one–dimensional Gaussian distributed basis questions X

$$p(y|x,f^0) \propto e^{-\frac{1}{2}\left(\frac{y-\bar{y}_x}{\sigma_x}\right)^2}.$$

[60] There are justifications of a relation between complexity and priors also for the Bayesian framework (MacKay, 1992a). The idea is that complex states appear usually in much more variations than simple ones and if their total probability is fixed their individual probabilities become small. Such a coupling between prior and complexity results from using uniform priors for specifically selected groups of states, and a grouping may be seen as more or less natural for specific situations. The problem of defining uniform priors in a such hierachical situation is similar to the situation for continuous variables where uniform priors do not remain uniform under general transformations.

Thus in this section we assume the f^0 to be parametrized by their regression functions $\bar y_x = \bar y_x(f^0)$. We get

$$\frac{d}{df^0_x} L(y|x',f^0) = \frac{d}{d\bar y_x} L(y|x',f^0) \propto \delta(x-x')\,\sigma_x^{-2}(y-\bar y_x).$$

For general local Gaussian questions including differing variances

$$p(y|q_x,f^0) \propto e^{-\frac{1}{2}\left(\frac{y-\bar y_{q_x}(\bar y_x)}{\sigma_{q_x}}\right)^2},$$

with $\bar y_{q_x}(\bar y_x)$ being a deterministic function of $\bar y_x$, we have

$$\frac{d}{d\bar y_x} L(y|q_{x'},f^0) \propto \delta(x-x')\,\sigma_{q_x}^{-2}(y-\bar y_{q_x})\frac{d}{d\bar y_x}\bar y_{q_x}(\bar y_x),$$

and for nonlocal Gaussian questions like the usual smoothness questions

$$p(y|q,f^0) \propto e^{-\frac{1}{2}\left(\frac{y-\bar y_q[\bar y_x]}{\sigma_q}\right)^2},$$

with $\bar y_q[\bar y_x]$ denoting a deterministic functional depending on the set $\{\bar y_x\}$,

$$\frac{d}{d\bar y_x} L(y|q,f^0) \propto \sigma_q^{-2}(y-\bar y_q)\frac{d}{d\bar y_x}\bar y_q[\bar y_x].$$

For general questions the logposterior has to be expressed as $L = \ln p$ and we find

$$\frac{d}{d\bar y_x} L(y|q,f^0) = \frac{d}{d\bar y_x}\ln p(y|q,f^0) = \frac{\frac{d}{d\bar y_x}p(y|q,f^0)}{p(y|q,f^0)}.$$

Remembering $p(y|x,f^0) = \prod_i p(y_i|x_i,f^0)$, $\int dx' = \sum_d \int \prod_i^d dx_i'^{(d)}$, and $\int dy = \sum_d \int \prod_i^d dy_i^{(d)}$ the derivative of the probability is found as

$$\frac{d}{d\bar y_x} p(y^q|q,f^0) = \frac{d}{d\bar y_x}\int dx' \int dy\, p(y|x',f^0)p(y^q,x'|y,q)$$

$$= \int dx' \int dy \left(\sum_i^d \sigma^{-2}_{x_i'^{(d)}}\delta(x_i'^{(d)}-x)(y_i-\bar y_x)\right)$$

$$\times p(y|x',f^0)p(y^q,x'|y,q)$$

for Gaussian $p(y|x,f^0)$. Not only for questions without input noise but also for $d=1$, including possible input noise, the x'–integration vanishes. In the latter case this gives

$$\int dy\,(y-\bar y_x)p(y|x,f^0)p(y^q|x,y,q).$$

The general stationarity conditions are obtained by setting the functional derivatives of the total logposterior with respect to $\bar y_x$ which parametrize the f^0_x to zero

$$\forall x\;:\; 0 = \frac{d}{d\bar y_x}\sum_i^n L(y_i|q_i,f^0),$$

where n is the number of training data $D = \{(q_i,y_i)\}$. Now, we look to some examples.

9.2 Local data

We study a situation where the logposterior is a sum of quadratic terms each of them with templates T_x only depending on one x. The standard case of training examples consisting only of local Gaussian basis questions has

$$L^S = \sum_i^n L(y_i|x_i,f^0)$$

The corresponding stationarity conditions are for $\sigma_x = 1$

$$0 = \sum_i^n \delta(x-x_i)(y_i^x - \bar y_x) = \sum_i^{n_x}(y_i^x-\bar y_x)$$

with n_x the number of times x is in D_q, y_i^x an answer to question x, and $\sum_i^0 = 0$ x not in the data. For $x\in D_q$ we find for unrestricted F^0 the well–known mean square solution

$$\bar y_x = \frac{1}{n_x}\sum_i^{n_x} y_i^x.$$

The $\bar y_x$ for $x \notin D_q$ are arbitrary.

Including other local Gaussian questions

$$p(y^{q_x}|q^x,f^0) \propto e^{-\frac{1}{2}\frac{(y^{q_x}-\bar y_{q_x})^2}{\sigma^2_{q_x}}},$$

with $\bar y_{q_x} = \bar y_{q_x}(\bar y_x)$ a function of one $\bar y_x$ only, we find

$$0 = \sum_i^{n_x}\sigma_x^{-2}(y_i^x - \bar y_x)$$

$$+\sum_{q_x}\sum_i^{n_{q_x}}\sigma_{q_x}^{-2}(y_i^{q_x}-\bar y_{q_x})\frac{d}{d\bar y_x}\bar y_{q_x},$$

equivalent to

$$\bar y_x = \frac{\sum_i^{n_x}\sigma_x^{-2}y_i^x + \sum_{q_x}\sum_i^{n_{q_x}}\sigma_{q_x}^{-2}y_i^{q_x}\frac{d}{d\bar y_x}\bar y_{q_x}}{n_x\sigma_x^{-2}+\sum_{q_x}n_{q_x}\sigma_{q_x}^{-2}\frac{\bar y_{q_x}}{\bar y_x}\frac{d}{d\bar y_x}\bar y_{q_x}}.$$

For linear $\bar y_{q_x} = a_{q_x}\bar y_x + b_{q_x}$ this is a linear equation with solution

$$\bar y_x = \frac{\sum_{q_x} a_{q_x}\sigma_{q_x}^{-2}\sum_i^{n_{q_x}}(y_i^{q_x}-b_{q_x})}{\sum_{q_x} a_{q_x}^2 n_{q_x}\sigma_{q_x}^{-2}}$$

where we simplified the formula by including the x into the q_x with $a_x = 1$ and $b_x = 0$. This gives the reweighting to be done for varying variances and scaling and the correction for varying bias.

A general local case which yields linear stationarity conditions can be written in form of a sum of quadratic terms

$$\frac{1}{2}\sum_i\int_{T^i} dx\,\Lambda_x^i||\bar y_x - T_x^i||^2 \tag{17}$$

$$= \frac{1}{2}\sum_i <\bar y - T^i\,|\Lambda^i|\,\bar y - T^i>$$

$$= \sum_i\left(\frac{1}{2}<\bar y\,|\Lambda^i|\,\bar y> - <\bar y\,|\Lambda^i|\,T^i>\right) + c.$$

Here c is an f^0–independent constant, Λ^i are nonnegative diagonal matrices withmatrix elements $\Lambda^i_{x,x'} =$

$\delta(x-x')\Lambda^i_x \ge 0$, and we assumed real scalar products. We define the projector $\mathcal{P}^{T^i} = \left(\mathcal{P}^{T^i}\right)^2$ into the T^i–space spanned by the x with nonzero Λ^i_x. Then Λ^i commutes with this projector $\Lambda = \mathcal{P}^{T^i}\Lambda = \Lambda\mathcal{P}^{T^i}$ and without loss of generality the template is understood to be restricted to T^i–space meaning $T^i = \mathcal{P}^{T^i}T^i$.

9.3 Quadratic nonlocal data = linear stationarity equations

Many smoothness and symmetry functionals are examples for Gaussian nonlocal questions. Take as example a functional with L^0 of the form

$$L^0(f^0) - c = -\frac{1}{2}\int dx \int dx'\, \bar{y}_x \mathcal{O}_{x,x'} \bar{y}_{x'} = -\frac{1}{2} < \bar{y}|\mathcal{O}|\bar{y} >, \tag{18}$$

with a real symmetric positive definite $\mathcal{O}$ representing a Gaussian probability and a constant c ensuring correct normalization. $\mathcal{O}_{x,x'} = \mathcal{O}(x,x') =< x\,|\,\mathcal{O}\,|\,x' >$ denotes the matrix elements of a (hermitian) linear operator $\mathcal{O}$ as we use angle brackets $<\cdot\,|\,\cdot>$ to denote a scalar product in a Hilbert space of functions $\bar{y}$. A matrix element

$$< \mathcal{D}\bar{y}|\mathcal{D}\bar{y} >= ||\mathcal{D}\bar{y}||^2 =< \bar{y}|\mathcal{D}^\dagger\mathcal{D}|\bar{y} >,$$

with $\mathcal{D}^\dagger$ denoting the adjoint of $\mathcal{D}$, gives an operator $\mathcal{O} = \mathcal{D}^\dagger\mathcal{D}$ on the domain where the operators are defined. To construct a smoothness functional for square integrable functions, $\mathcal{D}$ can be chosen as a hermitian linear differential operator. A first order example is[61]

$$\mathcal{D}_{x,x'} = \mathcal{D}(x,x') = i\delta'(x-x') = -i\delta(x-x')\frac{d}{dx},$$

giving

$$\mathcal{O}_{x,x'} = \mathcal{O}(x,x') = \delta''(x-x') = \delta(x-x')\frac{d^2}{dx^2}.$$

Instead of giving a template for $\bar{y}_x$ we could give also templates for other functions of $\bar{y}_x$. For example, a template T' for $\mathcal{D}\bar{y}$ can be written

$$L^0(f^0) = -\frac{1}{2} < \mathcal{D}\bar{y} - T|\Lambda_{T'}|\mathcal{D}\bar{y} - T' >$$

$$= -\left(\frac{1}{2} <\bar{y}\,|\mathcal{D}^\dagger\Lambda_{T'}\mathcal{D}\,|\,\bar{y}> - <\bar{y}\,|\,\mathcal{D}^\dagger\Lambda_{T'}T'>\right) + c,$$

for real scalar products and c an f^0–independent constant. Examples include distances in Sobolev spaces. With $\mathcal{O} = D^\dagger\Lambda^T D$ and $T = \Lambda^{T'}\mathcal{D}^\dagger T'$ this reads

$$= -\left(\frac{1}{2} <\bar{y}\,|\mathcal{O}|\,\bar{y}> - <\bar{y}\,|T>\right) + c.$$

For example, a loglikelihood of the form (18) includes a 'null' template for derivatives, i.e. has no term linear in $\bar{y}_x$.

[61] Note that this formal notation does not mean the operators are diagonal in the x–representation. The δ–function only restricts the derivatives to the location $x = x'$, but the derivatives itself depend also on the neighborhood of x. This is most easily seen by replacing the derivatives with a finite difference approximation.

The normalization constant for a d–dimensional y in the presence of linear terms is calculated according to

$$\int\left(\prod_{j=1}^d dy^j\right) e^{-<\bar{y}|\mathcal{O}|\bar{y}>+<\bar{y}|T>}$$

$$= (2\pi)^{\frac{d}{2}} 2^d (\det\mathcal{O})^{-\frac{1}{2}} e^{<T|\mathcal{O}^{-1}|T>}.$$

9.3.1 Continuous local and quadratic nonlocal data

The functional derivative for the sum of a nonlocal quadratic L^0 as in (18) and a local quadratic term is

$$\frac{dL(\bar{y})}{d\bar{y}_x} = -\int dx'\mathcal{O}_{x,x'}\bar{y}_{x'} - \Lambda^T_x(\bar{y}_x - T_x)$$

$$= - < x\,|\,\mathcal{O}\,|\,\bar{y}> - < x\,|\Lambda^T|\,\bar{y}\,> + < x\,|\Lambda^T|\,T> .$$

Mean square error terms are special examples of such terms and therefore encompassed by this formulation. The variable x can be multi–dimensional, assuming the vector is written in a basis where quadratic term is diagonal. (The situation where for one x several different T_x are available will be discussed below.) For invertible $\Lambda^T + \mathcal{O}$ the stationarity condition reads

$$\bar{y} = (\Lambda^T + \mathcal{O})^{-1}\Lambda^T T, \tag{19}$$

which is for a linear operator $\mathcal{O}$ a linear inhomogeneous equation. If not invertible in the full space components of the null space, i.e. solutions of $(\Lambda^T + \mathcal{O})\bar{y} = 0$ can be added to a special solution. In cases $\mathcal{O}^{-1}$ can be calculated, it can be useful to rewrite eq.(19) by separating the parts invariant under projection with $\mathcal{P}^T$ from those which are not

$$\mathcal{O}\bar{y} = \Lambda^T(T - \bar{y}).$$

Then the vector

$$a = \Lambda^T(T - \bar{y}) = \mathcal{P}^T a$$

being invariant under projection $\mathcal{P}^T$ can be calculated purely within the T–space. If $\mathcal{O}$ is invertible, the unknown $\bar{y}$ in the definition of a can be eliminated using $\bar{y} = \mathcal{O}^{-1}a$, giving for a the equation

$$\left(\mathcal{I} + \Lambda^T\mathcal{O}^{-1}\right)a = \Lambda^T T, \tag{20}$$

with $\mathcal{I}$ denoting the identity. Using $\mathcal{P}^T\Lambda^T = \Lambda^T\mathcal{P}^T$, $a = \mathcal{P}^T a$ and $T = \mathcal{P}^T T$ to insert the projector $\mathcal{P}^T$ gives

$$\left(\left(\mathcal{P}^T\Lambda^T\mathcal{P}^T\right)^{-1} + (\mathcal{P}^T\mathcal{O}^{-1}\mathcal{P}^T)\right)a = \mathcal{P}^T T = T.$$

Hence, only matrix elements of $(\mathcal{P}^T\mathcal{O}^{-1}\mathcal{P}^T)$ within the T–space are required to solve for a.

9.3.2 Discrete local and quadratic nonlocal data

Here we discuss more explicitly the special, but most common case of discrete Gaussian data templates with equal variance and quadratic nonlocal data, like e.g. smoothness. Gaussian data terms with $\sigma_x^2 = \lambda$ give

$$\frac{d(L^S + L^0)}{d\bar{y}_x} \propto \sum_i^{n_x}(y_i^x - \bar{y}_x) - \lambda\int dx'\,\mathcal{O}_{x,x'}\bar{y}_{x'},$$

and the stationarity condition reads

$$(\mathcal{N}^D + \lambda\mathcal{O})\bar{y} = D, \tag{21}$$

Vector D has components

$$D_x = \sum_i^n \delta(x - x_i) y_i^x = \sum_i^{n_x} y_i^x,$$

and therefore $D = \mathcal{P}^D D$. The projector $\mathcal{P}^D$ projects into the space spanned by the x included in the data and has matrix elements

$$\mathcal{P}^D_{x,x'} = \mathcal{P}^D(x,x') = \delta(x - x') \sum_i^{\tilde{n}} \delta(x - x_i),$$

$\tilde{n} = n - \sum_x \sum_{i=2}^{n_x} 1$ is the number of different $x = x_i$ in the data. The operator $\mathcal{N}^D = \mathcal{P}^D \mathcal{N}^D \mathcal{P}^D$, has matrix elements

$$\mathcal{N}^D_{x,x'} = \mathcal{N}^D(x,x')$$
$$= \delta(x - x') \sum_i^n \delta(x - x_i) = \delta(x - x') n_x,$$

which give the number of how often x appears in the data. The operator is equal to $\mathcal{P}^D$ and therefore an identity in that subspace if $n_x = 1$ for all x, which is the usual case for i.i.d. sampling of continuous x. In a space where $(\mathcal{N}^D + \lambda\mathcal{O})$ is invertible the linear, inhomogeneous (e.g. integro–differential) equation (21) has the solution

$$\bar{y} = (\mathcal{N}^D + \lambda\mathcal{O})^{-1} D.$$

being a special case of eq.(19) with $D/\lambda = \Lambda^T T$ and $\mathcal{N}^D/\lambda = \Lambda^T$. Components of a null space may be added. The matrix elements $\mathcal{O}^{-1}_{x,x'} = G(x,x')$ (or Green's function) satisfy by definition

$$\mathcal{O} G(x,x') = \delta(x - x').$$

For some $\mathcal{O}$ the Green's function can be calculated analytically. Then the solution of the resulting equation

$$\bar{y} = \frac{1}{\lambda} \mathcal{O}^{-1} \left(D - \mathcal{N}^D \bar{y} \right)$$

or in components

$$\bar{y}_x = \sum_i^n G(x, x_i)(y_i^{x_i} - \bar{y}_{x_i}) = \sum_i^{\tilde{n}} a_i G(x, x_i),$$

is for fixed x in a $\tilde{n}$–dimensional space spanned by known $G(x, x_i)$ with different x_i. In the vector

$$a = \frac{(D - \mathcal{N}^D \bar{y})}{\lambda}$$

with components

$$a_x = \sum_j^{n_x} \frac{y_j^x - \bar{y}_x}{\lambda}$$

only the components $a_{x_i} = a_i$ for x_i belonging to the data are not equal to zero. Inserting $\bar{y} = \mathcal{O}^{-1} a$ into the definition of a gives a $\tilde{n}$–dimensional matrix equation

$$\sum_i^{n_{x_i}} y_i^{x_i} = n_{x_i} \sum_j^d G(x_i, x_j) a_j + \lambda a_i,$$

or in operator notation

$$D = (\mathcal{N}^D (\mathcal{P}^D \mathcal{O}^{-1} \mathcal{P}^D) + \lambda\mathcal{I}) a.$$

The identity $\mathcal{I}$ commutes with $\mathcal{P}^D$ and $\mathcal{P}^D \mathcal{O}^{-1} \mathcal{P}^D$ has the matrix elements $G(x_i, x_j)$. This is the equivalent of eq.(20).

For example, using the (pseudodifferential) operator

$$\mathcal{O} = \sum_{m=0}^{\infty} \mathcal{O}_m = \sum_{m=0}^{\infty} (-1)^m \frac{\sigma^{2m}}{m! 2^m} \nabla^{2m},$$

where ∇^{2m} denotes the m–iterated Laplacian, results in a Gaussian for $G(x,x')$. Diagonal $G(x,x')$ correspond to local questions, radially symmetric (Gaussian) Green's functions are called Radial Basis Functions. This and more examples, including various forms of splines, can be found in (Poggio & Girosi, 1990 and Girosi, Jones, & Poggio, 1995).

Restricting the terms in the summation to a number smaller than $\tilde{n}$, corresponding to an additional prior or cost term, can be combined with an algorithm to determine the optimal selection of x_j (e.g. the centers of Gaussians). Including for example the variance σ_x as parameter for f^0 leads to nonlinear equations.

9.3.3 Discrete and continuous local and quadratic nonlocal data

If choosing Gaussians with equal variance we have to minimize

$$\sum (y_i - \bar{y}_{x_i})^2 + \lambda_T \int dx\, (\bar{y}_x - T_x)^2,$$

or for $h_x = \bar{y}_x - T_x$

$$\sum (y_i - h_{x_i} - T_{x_i})^2 + \lambda_T \int dx\, (h_x)^2,$$

with a shifted local error term and a penalty for deviation from zero. For the unshifted parametrization the stationarity condition reads

$$\bar{y}_x = \frac{\sum_i^{n_x} y_i^x + \lambda_T T_x}{n_x + \lambda_T}$$

which means $\bar{y}_x = T_x$ for nondata points $x \neq x_i, i = 1, \cdots, n$.

The last example becomes more interesting if combined with a nonlocal term, for example a differential operator implementing a smoothness prior. Then one has to minimize

$$\sum (y_i - \bar{y}_{x_i})^2 + \lambda_T \int dx\, (\bar{y}_x - T_x)^2$$
$$+ \lambda_S \int dx \int dx'\, (\bar{y}_x \mathcal{O}_{x,x'} \bar{y}_{x'})^2,$$

with stationarity equation

$$\int dx' \left((n_x + \lambda_T) + \mathcal{O}_{x,x'} \right) \bar{y}_{x'} = \sum_i^{n_x} y_i + \lambda_T T_x.$$

For the example $\mathcal{O}_{x,x'} = \delta(x - x') \frac{d^2}{dx^2}$ this gives the linear inhomogeneous differential equation

$$\left(\frac{d^2}{dx^2} + n_x + \lambda_T \right) \bar{y}_x = \sum_i^{n_x} y_i + \lambda_T T_x.$$

Having two templates, expressing that functions f^0 may be similar to T_1 OR T_2 leads to nonlinear equations discussed in the next paragraph.

9.4 Nonlinear and nonquadratic data = nonlinear stationarity equations

Priors which are constructed by combining quadratic subproperties C_i using a real valued extension of logic do not need to be quadratic in the $\bar{y}_x$, for example, if an OR is implemented in a soft and not in a hard version. Also one could use a parametrization of F^0 so $\bar{y}_x$ is a nonlinear function of the parameters,[62] or use a template for nonlinear questions, like a correlation template for $\bar{y}_x \bar{y}_{x'}$ in terms like

$$||\bar{y}_x \bar{y}_{x'} - T_{x,x'}||^2.$$

Let us consider a case with two templates combined by a soft OR. The two templates could have been obtained by using the human interface discussed above and represent two typical functions T^1 and T^2 to which the actual $\bar{y}_x(f^0)$ is expected to be similar to at least to one of them. For example, two such templates can be prototypical structures for electrocardiograms, or patterns in financial time series. This situation might be approximated according to eqs.(7) by including terms like

$$\lambda_T \left(\int dx\, (\bar{y}_x - T^1_x)^2 \right) \left(\int dx'\, (\bar{y}_{x'} - T^2_{x'})^2 \right),$$

assuming a possible bound m incorporated into λ_T. The term has the functional derivative with respect to $\bar{y}_x(f^0)$

$$\lambda_T \left((\bar{y}_x - T^1_x) \int dx\, (\bar{y}_x - T^2_x)^2 \right.$$
$$\left. + (\bar{y}_x - T^2_x) \int dx\, (\bar{y}_x - T^1_x)^2 \right),$$

yields a nonlinear stationarity equation, and may therefore be called a *nonlinear regularization* term.

More general, we can consider as example a logprobability of the form of a mixture model (For mixture models in general see Everitt & Hand, 1981, Titterington, Smith, & Makov, 1985, Kontkanen, Myllymäki, & Tirri, 1997)

$$L = \ln \left(\sum_i^{N_i} e^{-\frac{1}{2}\left(\sum_{j_i}^{N_{j_i}} <\bar{y} - T^{ij_i}|\mathcal{O}^{ij_i}|\bar{y} - T^{ij_i}>\right) + c^i} \right), \tag{22}$$

where the constants c^i include the normalization factor for the probability. Another example might be

$$L = -g \left(\frac{1}{2} \sum_i \prod_{j_i} <\bar{y} - T^{ij_i}|\mathcal{O}^{ij_i}|\bar{y} - T^{ij_i}> \right). \tag{23}$$

While the first form (22) uses the implementation of OR for loglikehoods (and is general enough to include the product terms of OR for nondisjunct events) the second example (23) might be seen as a fuzzy approximation according to eqs.(7) with some monotonically increasing function g, and m included in the definition of the $\mathcal{O}^{ij_i}$.

[62] Many methods, like for example sigmoidal neural networks use a nonlinear parametrization, but not independent for each $\bar{y}_x(f^0)$.

Because the operator $\mathcal{O}^{ij_i}$ can be diagonal in the x-representation and T^{ij_i} can be a sum of δ–functions this also encompasses the usual mean square data terms. A maximum does not change under strictly monotonically increasing functions $h(L)$, i.e. functions with $L(f^0) > L(f'^0) \Rightarrow h(L(f^0)) > h(L(f'^0))$ and $\frac{dh}{dL} > 0$, while strictly monotonically decreasing functions $h(L)$ only change a maximum into a minimum. Including functions h the stationarity conditions, obtained by setting the functional derivative to zero, read

$$0 = \frac{dh(L)}{dL} \frac{dL(f^0)}{df^0} = \frac{dh(L)}{dL} \frac{dL(\bar{y})}{d\bar{y}},$$

where in our case $\bar{y}$ represents the parameter vector f^0, and we will see that iteration procedures can be related to different h. Using the monotonic transformation $h(L) = e^L = p$ so no denominator arises from the derivative of the ln–function, yields for a logprobability of form (22)

$$\mathcal{O}\bar{y} = t, \tag{24}$$

with

$$\mathcal{O} = \sum_i e^{-\frac{1}{2}\left(\sum_{k_i} <\bar{y} - T^{ik_i}|\mathcal{O}^{ik_i}|\bar{y} - T^{ik_i}>\right) + c^i} \sum_{j_i} \mathcal{O}^{ij_i},$$
$$t = \sum_i e^{-\frac{1}{2}\left(\sum_{k_i} <\bar{y} - T^{ik_i}|\mathcal{O}^{ik_i}|\bar{y} - T^{ik_i}>\right) + c^i} \sum_{j_i} \mathcal{O}^{ij_i} T^{ij_i}. \tag{25}$$

For the example (23) one finds with $h(L) = g^{-1}(L)$

$$\mathcal{O} = \sum_i \sum_{j_i} \prod_{k_i \neq j_i} <\bar{y} - T^{ik_i}|\mathcal{O}^{ik_i}|\bar{y} - T^{ik_i}> \mathcal{O}^{ij_i}$$
$$t = \sum_i \sum_{j_i} \prod_{k_i \neq j_i} <\bar{y} - T^{ik_i}|\mathcal{O}^{ik_i}|\bar{y} - T^{ik_i}> \mathcal{O}^{ij_i} T^{ij_i}. \tag{26}$$

In the case of only one template $T^{ij_i} = T$, i.e. $t = \mathcal{O}T$ this has as trivial solution $\bar{y} = T$. However, in most cases nonlinear equations can only be solved numerically using some iteration procedure.

9.5 Iteration procedures = learning

Here we will discuss iteration procedures to solve for the inhomogeneous integro–diferential equations. Iteration procedures correspond to the actual learning algorithms. We consider here their application to MaP equations, but the methods also apply to MiR equations or a full Bayesian approach.

An iteration procedure is defined by a function G

$$f^{0,i+1} = G^i(f^{0,i}),$$

producing new guesses $f^{0,i+1}$ from a current guess $f^{0,i}$, and with fixed points being solutions of the stationarity conditions. We restrict in the following to solving the stationarity conditions, and assume the maximum (or minimum, saddle point) conditions, i.e. the second derivatives, to be checked separatly. We can construct an iteration procedure by choosing a function H^i we also allow in general to be (also stochastically) i–dependent, and write for $\bar{y}^{i+1} = G^i(\bar{y}^i)$

$$f^{0,i+1} = G^i(f^{0,i}) = f^{0,i} + \tilde{H}^i(\frac{dL(f^{0,i})}{df^{0,i}}).$$

to ensure fixed points are solutions of the stationarity equations we require $\tilde{H}^i(0) = 0$, and not to create additional spurious solutions one must have $\tilde{H}^i(x) \neq 0$ for $x \neq 0$. We can fulfill those conditions by defining a (possibly x–dependent) nonsingular linear mapping (a matrix for vector x) $H^i(x)$ acting on x with $\det(H^i(x)) \neq 0, \forall x \neq 0$, by $\tilde{H}^i(x) = H^i(x)x$. The matrix $H^i(x)$ is usually chosen positive definite (and symmetric) for a maximization problem. $H^i(x)$ may result from a (bijective) transformation of the independent variables $f^0 = T^i(f'^0)$ and/or from a strictly monotonic transformation $h^i(L)$ of the logposterior. For everywhere differentiable transformations the chain rule gives

$$\frac{dh^i(L(T^i(f'^0)))}{df'^0} = \frac{dh^i(L)}{dL}\frac{dL(f^0)}{df^0}\frac{dT(f'^0)}{df'^0}.$$

Strictly monotonic and therefore invertible transformations h^i, T^i do not lead to additional stationary points, i.e. spurious solutions. We write

$$f^{0,i+1} = f^{0,i} + H^i(f^{0,i})\frac{dL(f^{0,i})}{df^{0,i}}. \tag{27}$$

For example, gradient, Newton, or Quasi–Newton algorithms correspond to special H^i, and the i–dependence allows also to formulate methods like coordinate descent, or conjugate gradient in such a form. In general an iteration procedure can be given by an implicit equation $\tilde{G}^i(f^{0,i}, f^{0,i+1}) = 0$. A solution $f^{0,i+1}$, however, has to be given in an explicit form. We can formally extend the parameter vector f^0 to a (parameter vector) population vector by introducing dummy variables m (population indices), split f^0 into several f^0_m. Then H^i can induce interactions between different f^0_m. Iteration procedures of order m can for example be obtained by

$$f^{0,i+1} = f^{0,i} + \sum_{j=i-m+1}^{i} H^{i,j}(\{f^{0,k}\}_{i-m+1\le k\le i})\frac{dL(f^{0,j})}{df^{0,j}}.$$

The matrix H^i can be a stochastic function (e.g. stochastic annealing) deterministic only in some (e.g. zero temperature) limit or individual H^i can be chosen nonsingular only in subspaces like in line search or Gauss–Seidel algorithms, as long as a higher iterated equation can be written with a nonsingular H and the convergence check is made for this iterated equation. Similarly, for transformation methods (homotopy, EM–like algorithms) only at a fixed point of the iteration procedure the logposterior $L = L^i$ corresponds to a strictly monotone transformation of the original problem. For example, L^i can represent a smoothed version of L (e.g. deterministic annealing).

If G^i contains integrals over $\bar{y}_x$, like in the nonlinear terms, already much 'nonlocal information' may be contained in one iteration with G^i. Such integrals are for example introduced by the *EM algorithm* which defines hidden variables u, i.e.

$$p(f^0) = e^{L(f^0)} = \int du\, p(f^0,u) = \int du\, e^{L(f^0,u)},$$

with also nonnegative $p(f^0,u) \ge 0$. Then we define the corresponding conditioned variables which have by construction equal norm if summed over u, independent of f^0

$$p(u|f^0) = e^{L(u|f^0)} = \frac{p(f^0,u)}{\int du\, p(f^0,u)}.$$

This allows to add to L a cross–entropy term in the conditioned variables

$$L(f^0) + \int du\, e^{L(u|f^{0,ref})} L(u|f^0) \tag{28}$$

$$= \int du\, e^{L(u|f^{0,ref})} L(f^0,u) = Q(f^0, f^{0,ref}).$$

This transformation is strictly maximum sufficent relative to $L(f^{0,ref})$ (see Section 6.5) by construction, i.e. because conditioned variables have f^0–independent norm over u, the additional term is positive and maximal if $L(u|f^0) = L(u|f^{0,ref}), \forall u$, (See Section 5.3.4), which is the case for $f^0 = f^{0,ref}$. The reference point $f^{0,ref}$ has to be adapted during iteration. This has to be done at latest if a local maximum for fixed $f^{0,ref}$ is reached. However, maximizing Q does not require necessarily to find a maximum for every fixed $f^{0,ref}$, it is enough to increase Q with every iteration (Generalized EM). Summarizing, EM–like algorithms use a transformation $h^i(L)$ which is strictly monotonic only at a fixed point and during iteration only strictly maximality (or, respectively, minimality) sufficient relative to some previous guess (see Section 6.5).

While nonsingular H^i ensure that the fixed points of the iteration procedure are zeros of the gradient, this does not guarantee convergence. In general iteration procedures can produce all varieties of features known from discrete dynamical systems, including limit cycles or chaotic behaviour (See for example Devaney, 1986, Beck & Schlögl, 1993 and references therein).

We now discuss the example eq.(24) in more detail. Firstly, we write eq.(24) in a form

$$\bar{y} = G(\bar{y}),$$

by choosing some additive decomposition $\mathcal{O} = \mathcal{A} + \mathcal{B}$, with $\mathcal{A}$ positive definite and therefore invertible. Here $\mathcal{A}$ is usually chosen to be a linear operator, but invertability and not linearity is the crucial property. Then we obtain an iteration procedure by defining the left hand side to be $\bar{y}^{i+1}$ and $\bar{y}$ on the right hand side to be $\bar{y}^i$. For our examples this gives a $G(\bar{y})$ of the form

$$G(\bar{y}) = \mathcal{A}^{-1}\left(t - \mathcal{B}\bar{y}\right),$$

$$= \bar{y} - \mathcal{A}^{-1}\left(\mathcal{O}\bar{y} - t\right),$$

or in components

$$G_x(\bar{y}) = \int dx' \mathcal{A}^{-1}_{x,x'}\left(t_{x'} - \int dx'' \mathcal{B}_{x',x''}\bar{y}_{x''}\right).$$

The equations, or their variants described below, are solved by choosing a representation (i.e. a linear basis) to write it in component form, for example, in x–representation:

$$\bar{y}_x^{i+1} = G_x(\bar{y}^i).$$

In other cases one may prefer to work in another basis, for example plane waves (or general coherent states, Blaizot & Ripka, 1986) to transform a differential equation into an algebraic equation.

To achieve convergence one usually has to include a stepsize η, a method which is also called *relaxation*. Then a new guess $\bar{y}^{i+1}$ is generated from a previous guess $\bar{y}^i$ by mixing only part of the new solution to the old one, giving

$$\begin{aligned}\bar{y}^{i+1} = G_\eta(\bar{y}^i) &= (1-\eta)\bar{y}^i + \eta G(\bar{y}^i)\\ &= \bar{y}^i + \eta(G(\bar{y}^i) - \bar{y}^i) \qquad (29)\\ &= \bar{y}^i - \eta\mathcal{A}^{-1}(\mathcal{O}\bar{y} - t)\\ = \bar{y}^i + \tilde{H}_\eta(\bar{y}^i) &= \bar{y}^i + H_\eta(\bar{y}^i)\frac{dL(\bar{y}^i)}{d\bar{y}^i},\end{aligned}$$

where η can be included in the definition of the operator $\mathcal{A}_\eta^{-1} = \eta\mathcal{A}^{-1}$. For linear $\mathcal{A}$ we recognize an iteration procedure of the form (27) with for the logposterior (22)

$$H_\eta(L) = -\eta\mathcal{A}^{-1}e^L,$$

and for the logposterior (23)

$$H_\eta(L) = -\eta\mathcal{A}^{-1}g^{-1}(L),$$

because in both cases we have $\mathcal{O}\bar{y} - t = dh(L(\bar{y}))/d\bar{y} = (dL(\bar{y})/d\bar{y})(dh(L)/dL)$. The stepsize η is usually choosen as a number but can also be a vector (like in diagonally scaled steepest descent) In general $G_\eta(\bar{y}^i)$ $= G_{\eta(i,\bar{y}^i)}(i,\bar{y}^i)$ can be chosen i–dependent by varying $\eta = \eta^i$ or $\mathcal{A} = \mathcal{A}^i$, which can include dependence on past values $y^{0,k}$, $k \le i$. Changing the sign of (the components of a vector) η reverses the direction of the change (for that component), i.e. transforms a search for a maximum into one for a minimum. This can be used to make H^i positive (or negative) definit.

Convergence is not necessarily guaranteed, but depends, besides on choosing a good initial guess, on adjusting the relaxation factor η. Choosing $|\eta|$ small enough the change $|\bar{y}^{i+1} - \bar{y}^i|$ becomes arbitrary small, increasing the 'resolution' of the search and also reducing oscillations. This usually allows methods which search in directions with always positive (or negative) projections on the gradient to reach at least a local maximum (minimum) if the stepsize is small enough. (For convergence results see for example Bertsekas, 1995, Golden, 1996, and references therein.) The stepsize can also be determined by a line search in the direction given by $G^i(\bar{y}^i) - \bar{y}^i$.

The usual *gradient algorithm* or method of steepest descent is a special iteration procedure of this type: If $dL/d\bar{y} = (\mathcal{O}\bar{y} - t)$ and if $(-\mathcal{A})$ is the identity operator, then the term $G(\bar{y}^0) - \bar{y}^0$ is the gradient of L. Iteration schemes can also be related to the gradient of other surfaces but with the same stationary points. For example, for eq.(24) the iteration (29) is for $\mathcal{A} = 1$ a gradient algorithm on the surface $p = e^L$ and for a general linear operator $\mathcal{A}$ a gradient on a surface parametrized by variables ${\mathcal{A}^{-1}}^T\bar{y}$ (with T indicating the transpose).

To apply *EM–like algorithms* we can choose for a log-likelihood of the form (22) the summation index i (which has nothing to do with the iteration i) as hidden variable

$$\int du\, p(f^0,u) = \sum_i p(f^0,u_i),$$

with

$$p(f^0,u_i) = e^{L(f^0,u_i)} = e^{-\frac{1}{2}\left(\sum_{j_i}^{N_{j_i}} <\bar{y}-T^{ij_i}|\mathcal{O}^{ij_i}|\bar{y}-T^{ij_i}>\right)+c^i}$$

so that

$$p(u_i|f^0) = e^{L(u_i|f^0)} = \frac{e^{-\frac{1}{2}\left(\sum_{j_i}^{N_{j_i}} <\bar{y}-T^{ij_i}|\mathcal{O}^{ij_i}|\bar{y}-T^{ij_i}>\right)+c^i}}{\sum_i^{N_i} e^{-\frac{1}{2}\left(\sum_{j_i}^{N_{j_i}} <\bar{y}-T^{ij_i}|\mathcal{O}^{ij_i}|\bar{y}-T^{ij_i}>\right)+c^i}}.$$

The EM transformation (28) does not yet define the maximization procedure used to maximize $Q(f^0, f^{0,ref})$. Those may be maximized by any method, including such based on a random search, gradient–like, or EM–like algorithms. See for example the Helmholtz machine (Dayan, Hinton, Neal, & Zemel, 1995; Hinton, Dayan, Frey, Neal, 1995) for an application of the EM algorithm to hierachically defined $p(f^0)$.

The following table gives the iteration matrices H^i for some common local (gradient–like) iteration (learning) procedures. For the Newton method the inverse second derivative in the multidimensional case has to be read as inverse of the Hessian matrix, and the formula is used where the Hessian is negative definit (or positive definit for minimization) and at other locations it has to be replaced by another method. Quasi–Newton methods try to approximate the Hessian. In the table the abbreviations $\Delta^i_{(0)} = \bar{y}^i - \bar{y}^{i-1}$ and $\Delta^i_{(1)} = \frac{dL}{d\bar{y}}|_{\bar{y}^i} - \frac{dL}{d\bar{y}}|_{\bar{y}^{i-1}}$ are used. The given formula refers to the DFP (Davidon–Fletcher–Powell) method, in the BFGS (Broyden–Fletcher–Goldfarb–Shanno) method, for example, a term $a^i b^i {b^i}^T$ is added with $a^i = {\Delta^i_{(1)}}^T H^{i-1}\Delta^i_{(1)}$ and $b^i = \frac{\Delta^i_{(0)}}{{\Delta^i_{(0)}}^T\Delta^i_{(1)}} - \frac{H^{i-1}\Delta^i_{(1)}}{a^i}$ (T denoting the transpose). Conjugate gradient methods (CG) determine the stepsize by a line search in conjugate directions, obtained by a Gram–Schmidt–procedure. For nonquadratic problems they are usually combined with a heuristic to restart the Gram–Schmidt procedure. (See for example Bertsekas, 1995.) For the EM algorithm H'^i defines the chosen iteration algorithms used for Q with fixed reference state. Especially, EM algorithms are not restricted to local methods. (So for them the word local in the table caption does not necessarily apply.) Note that dL/df^0 in the nonlinear case usually contains integrals over x and therefore even optimization methods which are local with respect to f^0 are nonlocal with respect to x.

Some local learning algorithms

Gradient on L	$H^i = \eta^i$
Gradient on p	$H^i = \eta^i e^L$
Newton on L	$H^i = -1/\frac{d^2L}{(d\bar{y})^2}\|_{y^i}$
Quasi–Newton	$H^i = H^{i-1} + \frac{\Delta^i_{(0)} {\Delta^i_{(0)}}^T}{{\Delta^i_{(0)}}^T \Delta^i_{(1)}}$
(e.g. DFP on L)	$-\frac{H^{i-1}\Delta^i_{(1)}{\Delta^i_{(1)}}^T H^{i-1}}{{\Delta^i_{(1)}}^T H^{i-1}\Delta^i_{(1)}}$
CG on L	$H^i = \eta^i \times$
	$\left(1 - \frac{H^{i-1}}{\eta^{i-1}}\frac{dL}{dy}\|_{\bar{y}^{i-1}} \frac{{\Delta^i_{(1)}}^T}{\left(\left(\frac{dL}{d\bar{y}}\right)^T \frac{dL}{dy}\right)\|_{\bar{y}^{i-1}}}\right)$
Relaxation	$H^i = -\eta^i(\mathcal{A}^i)^{-1}$
EM	$H^i = H'^i \frac{dh^{ref}(L)}{dL}$
	with $h^{ref}(L) =$
	$L + \int du\, e^{L(u\|f^{0,ref})} L(u\|f^0)$

Nonlinear equations do normally have multiple solutions corresponding to several extremal points. If several solutions have to be considered in the last risk minimization step of a MaP–MiR approximation this requires calculation of the relative weight factors of the solutions or widths of the maxima depending on the second derivatives, or a corresponding estimate or assumption (See for example Gelman, Carlin, Stern & Rubin 1995 and especially for neural networks: Buntine & Weigend, 1991; MacKay 1991, 1992b, 1992c; Neal, 1996).

Nonlinear inhomogeneous equations appear for example in scattering theory as approximation to higher dimensional linear inhomogeneous equations (See for example, Austern, 1970, Taylor, 1972, Newton, 1982). There the inhomogeneties (data) are related to the in and out channels representing the boundary conditions or asymptotic states of the wavefunctions. Numerical aspects, applications to scattering theory and related higher order approximations are, for example, discussed in (Giraud & Nagarajan, 1991, Lemm, Giraud, & Weiguny, 1994 and Lemm, 1995ab).

Nonlocal templates creating the inhomogenities can be used in the following way: Instead of using a small space F_1^0 to represent possible states f^0 of nature and corresponding to hard implemented priors, one uses a larger space F_2^0 and implement F_1^0 within F_2^0 as priors with an soft OR. That means taking $f_1^0 \in F_1^0$ as templates for F_2^0 allows to go beyond F_1^0 if the data require. Note that a soft implemented template for $p(f^0|f)$ is not equivalent to using noisy answers for the f^0 as the state of knowledge about f^0, that is $p(f^0|f)$, is updated through data, while the noise levels of pure states are assumed to be stationary. i.e. clamped during learning. Templates can be seen as results from restricted similar tasks and are one method of tranfer of knowledge between tasks.

We have discussed how to use a real extension of logic to construct nonquadratic priors. This relation can be used also in the backward direction. Binary logic itself can be seen as discrete approximation for nonlinear real functions. After solving nonlinear equations approximately by binarization of variables, the solution can than be translated again into a real function, using, for instance, rules like those we discussed. From this point of view, translating logical rules formulated by experts into real valued priors can be seen as 'inverse' of the binary approximation of properties of possible states made by those experts. The existence of logic proves the importance of this approximation.

10 Conclusions

The paper is motivated by the fact that *predetermined dependencies* between answers to different questions are necessary for generalization and, thus, responsible for learning. Standard training examples alone can never lead to any prediction for new data. It seems therefore necessary to concentrate more on informations about the dependencies than it is usually done.

The aim of the paper is to treat those dependencies as explicit as possible, and to discuss possibilities to base information about dependencies upon measurement and control. This is especially important if the objects/situations of interest are complex and/or the amount of available standard training data is small.

For local questions, the predetermined dependencies have the form of stationarity conditions for the answer generating probability distributions. To enable generalization answers to *nonlocal questions* must be available, with the definitions of nonlocal questions representing predetermined dependencies. Commonly implemented nonlocal dependencies correspond to priors about smoothness and other approximate or exact symmetries. However, many more forms can be available in many practical situations. Often they appear as implicit or linguistic concepts defining for example object classes like faces, chairs, pedestrians or cars, and are, because dificullt to formalize, not included in the learning algorithm. Such dependencies can be implemented by an interface using *fuzzy priors*. Nonlocal questions are usually not sampled and not directly included in the loss function.

Often *nonapproximation aspects* are of interest, like the amount of resources (time, memory, money, understandability, complexity) needed by available alternatives $\hat{f}$. Then empirical risk minimization cannot be interpreted as being equivalent to a Bayesian maximum posterior approximation but can be extended to a two–step procedure (MaP–MiR). Common priors often depend formally from an infinite number of function values. Such priors can be implemented by using for the preparation process or for definition and control of the situation under study measuring devices different from those for the training process.

Priors stating that a function is probably similar to a template T^1 OR to a template T^2 can be implemented by some 'soft OR' or mixture model. This leads for the maximum posterior approximation to stationarity equations with *nonlinear* dependence from the local function values, reflecting the nontrivial interactions between different locations. As nonlinear equations have in general multiple solutions. such piors can, for example, be used to model phenomena like ambiguous illusions in perception.

In general, (more or less active) control is needed to relate measurements to situations for which learning is intended. Consequently, the ability to generalize is intimately related to the ability to control required dependencies in the application situation. For example, stationarity of probability distributions has to be controlled, and algorithms must be on–hand to relate available information to relevant data. One might even say: *Generalization is control of relevant conditions.*

References

[1] Aarts, E. & Korts, J. (1989) Simulated Annealing and Boltzmann Machines. New York: Wiley.

[2] Abu–Mostafa, Y. (1990) Learning from Hints in Neural Networks. *Journal of Complexity* **6**, 192–198.

[3] Abu–Mostafa, Y. (1993a) Hints and the VC Dimension. *Neural Computation* **5**, 278–288.

[4] Abu–Mostafa, Y. (1993b) A method for learning from hints. *Advances in Neural Information Processing Systems* **5**, S. Hanson et al (eds), 73–80, San Mateo, CA: Morgan Kauffmann.

[5] Alder, B.J. & Wainwright, T.E. (1959) Studies in molecular dynamics. I. General method. *Journal of Chemical Physics* **31**,459–466.

[6] Allgower, E.L. & Georg, K. (1990) Numerical Continuation Methods, Berlin: Springer–Verlag.

[7] Allen, D.M. (1974) The relationship between variable selection and data augmentation and a method of prediction. *Technometrics* **16**, 125.

[8] Amari, S. (1985) Differential geometrical methods in statistics. Springer Lecture Notes in Statistics, 28, Springer–Verlag.

[9] Amari, S. (1995) Information Geometry of the EM and em Algorithms for Neural Networks. *Neural Networks* **8** (9), 1379–1408.

[10] Austern, N. (1970) Direct Nuclear Reactions. New York: Wiley.

[11] Balasubramanian, V. (1996) Statistical Inference, Occam's Razor and Statistical Mechanics on the Space of Probability Distributions *cond–mat*/9601030

[12] Balian, R. (1991) From Microphysics to Macrophysics. Vol. I. Berlin: Springer.

[13] Barber, D. & Williams, C.K.I. (1997) Gaussian processes for Bayesian classification via hybrid Monte Carlo. To appear in *Advances in Neural Information Processing Sysytems 9.*

[14] Barndorff–Nielsen, O.E. (1978) Information and exponential families in statistical theory. New York: Wiley.

[15] Bazaraa, M.S., Sherali, H.D., & Shetty, C.M. (1993) Nonlinear Programming. (2nd ed.) New York: Wiley.

[16] Beck, C. & Schlögl, F. (1993) Thermodynamics of chaotic systems. Cambridge University Press.

[17] Berger, J.O. (1985) Statistical Decision Theory and Bayesian Analysis. (Second Ed.),New York: Springer.

[18] Bertsekas, D.P. (1995) Nonlinear Programming. Belmont, MA: Athena Scientific.

[19] Berry, M.V. (1966) *Proc. Phys. Soc.* (London) **89**, 479.

[20] Besag, J. & Green, P.J. (1993) Spatial statistics and Bayesian computation. *Journal of the Royal Statistical Society B* **55**, 25–102.

[21] Bialek, W., Callan, C.G., & Strong, S.P. (1996) Field Theories for Learning Probability Distributions. *cond–mat*/9607180.

[22] Binder, K. (ed.) (1986) Monte Carlo methods in statistical physics. (2nd ed.) Berlin: Springer-Verlag.

[23] Binder, K. (ed.) (1987) Applications of the Monte Carlo methods in statistical physics. (2nd ed.) Berlin: Springer-Verlag.

[24] Binder, K. (ed.) (1995) The Monte Carlo method in condensed matter physics. (2nd ed.) Berlin: Springer-Verlag.

[25] Binder, K., Heermann, D.W. (1988) Monte Carlo simulation in statistical physics: an introduction. Berlin: Springer-Verlag.

[26] Bishop, C.M. (1995) Training with noise is equivalent to Tikhonov regularization. *Neural Computation* **7** (1), 108–116.

[27] Bishop, C.M. (1995) Neural Networks for Pattern Recognition. Oxford: Oxford University Press.

[28] Bleistein, N. & Handelsman, R.A. (1986) Asymptotic Expansions of Integrals. New York: Dover.

[29] Buntine, W.L. & Weigend, A.S. (1991) Bayesian back-propagation. *Complex Systems* **5**, 603–643.

[30] Connor, J.N.L. & Marcus, R.A. (1971) *J. Chem. Phys.* **55**, 5636.

[31] Davis, L. (ed.) (1987) Genetic Algorithms and Simulated Annealing. Morgan Kaufmann.

[32] Davis, L. (ed.) (1991) Handbook of Genetic Algorithms. Van Nostrand Reinhold.

[33] Dayan, P., Hinton, G.E., Neal, R.M., & Zemel, R.S. (1995) The Helmholtz Machine. *Neural Computation* **7**, 889–904.

[34] De Bruijn, N.G. (1981) Asymptotic Methods in Analysis. New York: Dover.

[35] Dempster, A.P., Laird, N.M., & Rubin, D.B. (1976) Maximum likelihood from incomplete data via the EM algorithm. *Proceedings of the Royal Statistical Society*, 1-38.

[36] Derka, R., Bužek, V., Adam, G., & Knight, P.L. (1996) From quantum Bayesian inference to quantum tomography. quant-ph/9701029.

[37] Devaney, R.L. (1986) An Introduction to Chaotic Dynamical Systems. Menlo Park: Benjamin/-Cummings.

[38] Dudley, R.M. (1984) A course on empirical processes. *Lecture Notes in Mathematics*, 1097:2-142.

[39] Efron, B. & Tibshirani R.J. (1993) An Introduction to the Bootstrap. New York: Chapman & Hall.

[40] Everitt, B.S. & Hand, D.J. (1981) Finite Mixture Distributions. London: Chapman and Hall.

[41] Feller, W. (1957) An Introduction to Probability Theory and Its Applications. Vol. I (2nd edition) New York: Wiley.

[42] Ferraro, M. (1992) Invariant Pattern Representations and Lie Groups Theory. *Advances in Electronics and Electron Physics*, **84**, 131.

[43] Gardiner, C.W. (1990) Handbook of Stochastic Methods. (2nd edition) Berlin: Springer Verlag.

[44] Gelfand, A.E. & Smith, A.F.M. (1990) Sampling–based approaches to calculating marginal densities. *Journal of the American Statistical Association* **85**, 398–409.

[45] Gelfand, A.E., Hills, S.E., Racine–Poon, A., & Smith, A.F.M. (1990) Illustration of Bayesian inference in normal data models using Gibbs sampling. *Journal of the American Statistical Association* **85**, 972–985.

[46] Gelman, A., Carlin, J.B., Stern, H.S., & Rubin, D.B. (1995) Bayesian Data Analysis. New York: Chapman & Hall.

[47] Geman, S & Geman, D. (1984) Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **6**, 721–741. Reprinted in Shafer & Pearl (eds.) (1990) Readings in Uncertainty Reasoning. San Mateo, CA: Morgan Kaufmann.

[48] Geyer, C. (1992) Practical Markov chain Monte Carlo (with discussion). *Statistical Science* **7**, 473–511.

[49] Giraud, B.G. & Nagarajan, M.A. (1991) *Ann. Phys. (N.Y.)* **212**, 260.

[50] Girosi, F. & Chan, N.T. (1995) Prior Knowledge and the Creation of Virtual Examples for RBF Networks. *IEEE Workshop on Neural Networks for Signal Processing*, September 1995, Cambridge, MA.

[51] Girosi, F., Jones, M., & Poggio, T. (1995) Regularization Theory and Neural Networks Architectures. *Neural Computation* **7** (2), 219–269.

[52] Goldberg, D.E. (1989) Genetic Algorithms in Search, Optimization, and Machine Learning. Addison–Wesley.

[53] Golden, R.M. (1996) Mathematical Methods for Neural Network Analysis and Design. Cambridge, MA: MIT Press.

[54] Härdle, W. (1990) Applied nonparametric regression. Cambridge: Cambridge University Press.

[55] Hammersley, J.M. & Handscomb, D.C. (1964) Monte Carlo Methods. London: Chapman & Hall.

[56] Hastie, T.J. & Tibshirani, R.J. (1990) Generalized Additive Models. London: Chapman & Hall.

[57] Hastings, W.K. (1970) Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**, 97–109.

[58] Haussler, D. (1995) Decision Theoretic Generalizations of the PAC Model for Neural Net and Other Learning Applications. In: Wolpert, D. (Ed.) The Mathematics of Generalization. SFI Proceedings Volume XX, Addison–Wesley.

[59] Hinton, G.E., Dayan, P., Frey, B.J., & Neal R.M., (1995) The "Wake–Sleep" Algorithm for Unsupervised Neural Networks. *Science* **268**, 1158–1161.

[60] Holland, J.H. (1975) Adaption in Natural and Artificial Systems. University of Michigan Press. (2nd ed. MIT Press, 1992.)

[61] Jacobs, R.A., Jordan, M.I., Nowlan, S.J. & Hinton, G.E. (1991) Adaptive mixture of local experts. *Neural Computation* **3**, 79–87.

[62] Jaynes, E.T., (1996) Probability Theory: The Logic of Science. Fragmentary Edition March 1996. (available under ftp://bayes.wustl.edu/Jaynes.book/)

[63] Jordan, M.I. & Jacobs, R.A. (1994) Hierachical mixtures of experts and the EM algorithm. *Neural Computation* **6**, 181–214.

[64] Jeffrey, R. (1968) Probable knowledge. In *The problem of inductive logic*, ed. I. Lakatos. Amsterdam: North Holland.

[65] Jensen, F.V. (1996) An Introduction to Bayesian Networks. New York: Springer–Verlag.

[66] van Kampen, N.G. (1992) Stochastic Processes in Physics and Chemistry. Amsterdam: North–Holland.

[67] Kahneman, D. & Tversky, A. (1979) Prospecttheory: An Analysis of Decision under Risk. *Econometrica*, **47** (263), 269–291.

[68] Kahneman, D. & Tversky, A. (1982a) On the study of statistical intuitions. *Cognition*, **11**, 123–141.

[69] Kahneman, D. & Tversky, A. (1982b) Variants of uncertainty. *Cognition*, **11**, 143–157.

[70] Kahneman, D. & Tversky, A. (1982c) A reply to Evans. *Cognition*, **12**, 325–326.

[71] Kahneman, D., Slovic, P., & Tversky, A. (1983) Judgement under uncertainty: Heuristics and biases. Cambridge: Cambridge University Press.

[72] Kandel, A. (1982) Fuzzy Techniques in Pattern Recognition. New York: John Wiley and Sons.

[73] Kearns, M.J. & Vazirani, U.V. (1994) An Introduction to Computational Learning Theory. Cambridge, MA: MIT Press.

[74] Klir, J.G. & Yuan, B. (1995) Fuzzy Sets and Fuzzy Logic. Upper Saddle River, NJ: Prentice Hall.

[75] Kohonen, T. (1995) Self–Organizing Maps. Berlin: Springer.

[76] Kolmogorov, A.N. (1957) On the representation of continuous functions of several variables by superposition of continuous functions of one variable and addition. *Doklady Akademiia Nauk SSSR* **114** (5), 953–956.

[77] Kontkanen, P., Myllymäki, P., & Tirri, H. (1997) Constructing Bayesian finite mixture models by the EM algorithm. NeuroCOLT Technical Report Series, NC–TR–97–003.

[78] Laplace, P.S. (1810a) Mémoire sur les formules qui sont fonctions de très grands nombres et sur leurs application aux probabilités. *Oeuvres de Laplace* **12**, 301–345.

[79] Laplace, P.S. (1810b) Mémoire sur les intégrales définies et leur application aux probabilités. *Oeuvres de Laplace* **12**, 357–412.

[80] Lauritzen, S.L. (1996) Graphical Models. New York: Oxford University Press.

[81] Le Cam, L. (1953) On some asymptotic properties of maximum likelihood estimates and related Bayes estimates. *University of California Publications in Statistics* **1** (11), 277–330.

[82] Le Cam, L. (1986) Asymptotic Methods in Statistical Decision Theory. New York: Springer.

[83] Le Cam, L. & Yang, G.L. (1990) Asymptotics in Statistics. New York: Springer.

[84] Leen, T.K. (1995) From Data Distributions to Regularization in Invariant Learning. *Neural Computation* **7**, 974–981.

[85] Lemm, J.C. (1984) Entscheidungsstrategien und Informationskosten. Diplomarbeit Universität Münster.

[86] Lemm, J.C., Giraud, B.G., & Weiguny, A. (1994) *Phys. Rev. Lett.* **73** 420.

[87] Lemm, J.C. (1995a) Inhomogeneous Random Phase Approximation for Nuclear and Atomic Reactions. *Annals of Physics* **244** (1), 136–200.

[88] Lemm, J.C. (1995b) Inhomogeneous Random Phase Approximation: A Solvable Model. *Annals of Physics* **244** (1), 201–238.

[89] Lemm, J.C., Beiu, V., & Taylor, J.G. (1995) Density Estimation as a Preprocessing Step for Constructive Algorithms. In Neural Networks: Artificial Intelligence and Industrial Applications. Proceedings of the Third Annual SNN Symposium on Neural networks, Nijmegen, The Netherlands, 14–15 September 1995 B. Kappen & S. Gielen (eds.) London: Springer Verlag.

[90] Lin, S.–H., Kung, S.–Y., & Lin, L–J (1997) Face recognition/detection by probabilistic Decision–Based Neural Network. *IEEE Trans. Neural Networks* **8** (1), 114–132.

[91] MacKay, D.J.C. (1991) Bayesian Methods for Adaptive Models. Ph.D thesis, California Institute of Technology.

[92] MacKay, D.J.C. (1992a) Bayesian interpolation. *Neural Computation* **4** (3), 415–447.

[93] MacKay, D.J.C. (1992b) A practical Bayesian framework for backpropagation networks. *Neural Computation* **4**, 448–472.

[94] MacKay, D.J.C. (1992c) The evidence framework applied to classification networks. *Neural Computation* **4**, 720–736.

[95] Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., & Teller, E., (1953) Equation of state calculations by fast computing machines. *Journal of Chemical Physics* **21**, 1087–1092.

[96] Mezard, M., Parisi, G., & Virasoro, M.A. (1987) Spin Glass Theory and Beyond. World Scientific.

[97] Michalewicz, Z. (1992) Genetic Algorithms + Data Structures = Evolution Programs. Springer–Verlag.

[98] Miller, W.H. (1970) *J. Chem. Phys.* **53**, 3578.

[99] Minski, M.L. & Papert, S.A. (1990) (Expanded Edition, Original edition, 1969) Perceptrons. Cambridge, MA: MIT Press.

[100] Mitchell, M. (1996) An Introduction to Genetic Algorithms. Cambridge, MA: MIT Press.

[101] Montvay, I. & Münster, G., (1994) Quantum Fields on a Lattice, Cambridge University Press.

[102] Neal, R.M., & Hinton, G.E., (1993) A new view of the EM algorithm that justifies incremental and other variants. (submitted to *Biometrika*).

[103] Neal, R.M. (1993) Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG–TR–93–1, Dept. of Comp.. Sc., Univ. of Toronto, Canada.

[104] Neal, R.M. (1996) Bayesian Learning for Neural Networks. New York: Springer.

[105] Neal, R.M. (1997) Monte Carlo Implementation of Gaussian Process Models for Bayesian Regression and Classification. Technical Report No. 9702, Dept. of Statistics, Univ. of Toronto, Canada.

[106] Negele, J., W. & Orland, H. (1988) Quantum Many–Particle Systems. Frontiers In Physics Series (Vol. 68), Addison–Wesley.

[107] Newton, R.G. (1982) Scattering Theory of Waves and Particles. New York: Springer–Verlag.

[108] Pearl, J. (1988) Probabilistic Reasoning in Intelligent Systems. San Mateo, CA: Morgan Kauffmann.

[109] Pierre, D.A. (1986) Optimization Theory with Applications. New York: Dover. (Original edition Wiley, 1969).

[110] Poggio, T. & Girosi, F. (1990) Networks for Approximation and Learning. Proceedings of the IEEE, Vol 78, No. 9.

[111] Pollard, D. (1984) Convergence of Stochastic Processes. New York: Springer Verlag.

[112] Pomerleau, D. (1991) *Neural Computation* **3**, 88.

[113] Ratsaby, J. & Maiorov V. (1996) Learning from Examples and Side Information. NeuroCOLT Technical Report Series, NC–TR–96–050.

[114] Richter, S.L. & DeCarlo, R.A. (1983) Continuation methods: theory and applications. *IEEE Trans. Circuits Syst.* CAS–**30**, 347–352.

[115] Ripley, B.D. (1977) Modelling spatial patterns (with discussion). *Journal of the Royal Statistical Society series B* **39**, 172–212.

[116] Ripley, B.D. (1987) Stochastic Simulation. New York: Wiley.

[117] Ripley, B.D. (1996) Pattern Recognition and Neural Networks. Cambridge University Press.

[118] Rissanen, J. (1989) Stochastic Complexity in Statistical Inquiry. Singapore: World Scientific.

[119] Saul, L.K., Jaakola, T., & Jordan, M.I. (1996) Mean Field Theory for Sigmoid Belief Networks. *Journal of Artificial Intelligence Research* **4**, 61–76.

[120] Schervish, M.J. (1995) Theory of Statistics. New York: Springer–Verlag.

[121] Schwefel, H.–P. (1995) Evolution and Optimum Seeking. Wiley.

[122] Seneta, E. (1981) Non-Negative Matrices and Markov Chains. (2nd ed.) New York : Springer-Verlag,

[123] Seung, H.S. (1995) Annealed Theories of Learning. In Neural Networks: The Statistical Mechanics Perspective, Proceedings of the CTP-PBSRI Joint Workshop on Theoretical Physics, World Scientific.

[124] Shawe–Taylor, J., Bartlett, P.L., Williamson, R.C., & Anthony, M. (1996a) A Framework for Structural Risk Minimization. NeuroColt Technical Report Series, NC-TR-96-032.

[125] Shawe–Taylor, J., Bartlett, P.L., Williamson, R.C., & Anthony, M. (1996b) Structural Risk Minimization over Data–Dependent Hierachies. NeuroColt Technical Report Series, NC-TR-96-053.

[126] Sietsma, J. & Dow, R.J.F. (1991) Creating artificial neural networks that generalize. *Neural Networks* **4** (1), 67–79.

[127] Silverman, B.W. (1986) Density Estimation for Statistics and Data Analysis. London: Chapman& Hall.

[128] Smith, A.F.M. & Roberts, G.O. (1993) Bayesian computation via Gibbs sampler and related Markov chain Monte Carlo methods (with discussion). *Journal of the Royal Statistical Society series B* **55**, 3–23.

[129] Smola, A.J. & Schöelkopf, B. (1997) On a Kernel–based Method for Pattern Recognition, Regression, Approximation and Operator Inversion. Preprint.

[130] Stone, M. (1974) Cross–validation choice and assessment of statistical predictions. *Journal of the Royal Statistical Society B* **36**, 111-147.

[131] Stone, M. (1974) An asymptotic equivalence of choice of model by cross–validation and Akaike's criterion. *Journal of the Royal Statistical Society B* **39**, 44.

[132] Tanner, M.A. (1993) Tools for Statistical Inference. New York: Springer Verlag.

[133] Taylor, J.R. (1972) Scattering Theory. New York: Wiley.

[134] Tierney, L. (1994) Markov chains for exploring posterior distributions (with discussion). *Annals of Statistics* **22**, 1701–1762.

[135] Tikhonov, A.N. (1963) Solution of incorrectly formulated problems and the regularization method. *Soviet Math. Dokl.* **4**, 1035–1038.

[136] Titterington, D.M., Smith, A.F.M., & Makov, U.E. (1985) Statistical Analysis of Finite Mixture Distributions. New York: Wiley.

[137] Tversky, A. (1972) Elimination by Aspects. *Psychological Review*, **79** (4), 281–301.

[138] Tversky, A. & Kahneman, D. (1981) The Framing of Decisions and the Psychology of Choice. *Science*, **211** (4481), 453–458.

[139] Vapnik, V.N. (1982) Estimation of Dependences Based on Empirical Data. New York: Springer–Verlag.

[140] Vapnik, V.N. (1995) The Nature of Statistical Learning Theory. New York: Springer–Verlag.

[141] Verri, A. & Poggio, T. (1986) Regularization Theory and Shape Constraints. M.I.T., Artificial Intellegence Laboratory, Memo No.916.

[142] Vetter, T., Poggio, T., & Bülthoff, H. (1992) 3D Object Recognition: Symmetry and Virtual Views. M.I.T., Artificial Intellegence Laboratory, Memo No.1409.

[143] Wald, A, (1938) Contributions to the theory of statistical estimation and testing hyppotheses. *Ann. Math. Stat.* **10**, 299–326.

[144] Watkin, T.L.H., Rau, A., & Biehl, M. (1993) The statistical mechanics of learning a rule. *Rev. Mod. Phys.*, **65**, 499–556.

[145] Webb, A.R. (1994) Functional approximation by feed–forward networks: a least–squares approach to generalisation. *IEEE Transactions on Neural Networks* **5**, 363–371.

[146] Whitesitt, J.E. (1995) Boolean Algebra and Its Applications. New York: Dover, (Originally puplished: Addison–Wesley, 1961).

[147] Williams, C.K.I. & Rasmussen, C.E. (1996) Gaussian processes for regression. In D.S. Touretzki, M.C. Mozer, & M.E. Hasselmo (eds.) *Advances in Neural Information Processing Systems 8* Cambridge, MA: MIT Press.

[148] Wierling, A., Giraud, B., Mekideche, F., Horiuchi, H., Maruyama, T., Ohnishi, A., Lemm, J.C., & Weiguny A. (1994) *Z. Phys. A* **348**, 153.

[149] Wolpert, D.H. (1994) Bayesian Backpropagation over I–O functions rather than weights. In Cowan, J.D., Tesauro, G., & Alspector, J. (eds.) *Advances in Neural Information Processing Systems 6* San Francisco, CA: Morgan Kaufmann, pp. 200–207.

[150] Wolpert, D.H. (1994b) Discussion of Ripley, B.D., Neural networks and related methods for classification. *Journal of the Royal Statisical Society B* **56**, 450–451.

[151] Wolpert, D.H. (1996a) The Lack of A Priori Distinctions between Learning Algorithms. *Neural Computation* **8** (7), 1341-1390.

[152] Wolpert, D.H. (1996b) The Existence of A Priori Distinctions between Learning Algorithms. *Neural Computation* **8** (7), 1391-1420.

[153] Zadeh, L.A. (1987) Fuzzy Sets and Applications. Selected Papers. R.R. Yager et al. (eds), New York: Wiley.

[154] Zadeh, L.A. (1996) Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems. Selected Papers. G.J. Klir & B. Yuan (eds.) Singapore: World Scientific.

Acknowledgments The author is very grateful to F. Girosi and T. Poggio for many useful discussions.