PERGAMON

Neural Networks 14 (2001) 257–274

Neural Networks

www.elsevier.com/locate/neunet

Invited article

Bayesian approach for neural networks—review and case studies

Jouko Lampinen*, Aki Vehtari

Laboratory of Computational Engineering, Helsinki University of Technology, PO Box 9400, FIN-02015 HUT, Espoo, Finland

Abstract

We give a short review on the Bayesian approach for neural network learning and demonstrate the advantages of the approach in three real applications. We discuss the Bayesian approach with emphasis on the role of prior knowledge in Bayesian models and in classical error minimization approaches. The generalization capability of a statistical model, classical or Bayesian, is ultimately based on the prior assumptions. The Bayesian approach permits propagation of uncertainty in quantities which are unknown to other assumptions in the model, which may be more generally valid or easier to guess in the problem. The case problem studied in this paper include a regression, a classification, and an inverse problem. In the most thoroughly analyzed regression problem, the best models were those with less restrictive priors. This emphasizes the major advantage of the Bayesian approach, that we are not forced to guess attributes that are unknown, such as the number of degrees of freedom in the model, non-linearity of the model with respect to each input variable, or the exact form for the distribution of the model residuals.

Keywords: Bayesian data analysis; Hirarchical models; Neural networks; Comparison of models

1. Introduction

In Bayesian data analysis all uncertain quantities are modeled as probability distributions, and inference is performed by constructing the posterior conditional probabilities for the unobserved variables of interest, given the observed data sample and prior assumptions. Good references for Bayesian data analysis are Berger (1985), Bernardo and Smith (1994) and Gelman, Carlin, Stern, and Rubin (1995).

For neural networks, the Bayesian approach was pioneered in Buntine and Weigend (1991), Mackay (1992) and Neal (1992), and reviewed in Bishop (1995), MacKay (1995) and Neal (1996). With neural networks, the main difficulty in model building is controlling the complexity of the model. It is well known that the optimal number of degrees of freedom in the model depends on the number of training samples, amount of noise in the samples and the complexity of the underlying function being estimated. With standard neural networks techniques, the means for both determining the correct model complexity and setting up a network with the desired complexity are rather crude and often computationally very expensive.

In the Bayesian approach, these issues can be handled in a natural and consistent way. The unknown degree of complexity is handled by defining vague (non-informative) priors for the hyperparameters that determine the model complexity, and the resulting model is averaged over all model complexities weighted by their posterior probability given the data sample. The model can be allowed to have different complexity in different parts of the model by grouping the parameters that are exchangeable (have identical role in the model) to have a common hyperparameter. If, in addition, it is assumed that the complexities are more probably similar, a hierarchical hyperprior can be defined for the variance of the hyperparameters between groups.

Another problem of standard neural network methods is the lack of tools for analyzing the results (confidence intervals for the results, like 10 and 90% quantiles, etc.). The Bayesian analysis yields posterior predictive distributions for any variables of interest, making the computation of confidence intervals possible.

In this contribution, we discuss the Bayesian approach in statistical modeling (Section 2), with emphasis on the role of prior knowledge in the modeling process. In Section 3 we give a short review of Bayesian MLP models and MCMC techniques for marginalization. Then we present three real world modeling problems, where we assess the performance of the Bayesian MLP models and compare the performance to standard neural networks methods and other statistical models. The application problems are: (i) a regression problem of predicting the quality of concrete in concrete manufacturing process (Section 4); (ii) approximating an

* Corresponding author. Tel.: +358-9-451-4827.
E-mail address: jouko.lampinen@hut.fi (J. Lampinen).

0893-6080/01/$ - see front matter
PII: S0893-6080(00)00098-8

Nomenclature

$\mathscr{H}$	Model, including all implicit assumptions
θ	Parameters and hyperparameters of a model
$\mathbf{w}^1$, $\mathbf{b}^1$, $\mathbf{w}^2$, $\mathbf{b}^2$	Hidden and output layer weights and biases of a one-hidden-layer MLP
D	Data sample, $D = \{(\mathbf{x}^{(1)}, \mathbf{y}^{(1)}), \ldots, (\mathbf{x}^{(n)}, \mathbf{y}^{(n)})\}$
$\mathbf{x}$, $\mathbf{y}$	Model input and output vectors, y in univariate models
e	Model residual, e in univariate models
σ^2	Variance of Gaussian residual model
α	Variance of Gaussian prior for weights
$N(\mu,\sigma^2)$	Normal distribution with mean μ and variance σ^2
Inv-gamma(s, v)	Inverse Gamma distribution with parameters s and v
$t_v(\mu, \sigma^2)$	Student's t-distribution with v degrees of freedom, mean μ and scale σ^2
$U_d(a, b)$	Uniform distribution of integer values between a and b

inverse mapping in a tomographic image reconstruction problem (Section 5); and (iii) a classification problem of recognizing tree trunks in forest scenes (Section 6). Finally, we discuss the conclusions of our experiments in relation to other related studies on Bayesian neural networks.

2. The Bayesian approach

The key principle of Bayesian approach is to construct the posterior probability distributions for all the unknown entities in a model, given the data sample. To use the model, marginal distributions are constructed for all those entities that we are interested in, i.e. the end variables of the study. These can be the parameters in parametric models, or the predictions in (non-parametric) regression or classification tasks.

Use of the posterior probabilities requires explicit definition of the prior probabilities for the parameters. The posterior probability for the parameters θ in a model $\mathscr{H}$ given the data D is, according to the Bayes' rule,

$$p(\theta|D, \mathscr{H}) = \frac{p(D|\theta, \mathscr{H})p(\theta|\mathscr{H})}{p(D|\mathscr{H})}, \quad (1)$$

where $p(D|\theta, \mathscr{H})$ is the likelihood of the parameters θ, $p(\theta|\mathscr{H})$ is the prior probability of θ, and $p(D|\mathscr{H})$is a normalizing constant, called evidence of the model $\mathscr{H}$. The term $\mathscr{H}$ denotes all the hypotheses and assumptions that are made in defining the model, like a choice of MLP network, specific noise model, etc. All the results are conditioned on these assumptions, and to make this clear we prefer to have the term $\mathscr{H}$ explicitly in the equations. In this notation the normalization term $P(D|\mathscr{H})$ is directly understandable as the marginal probability of the data, conditional on $\mathscr{H}$, integrated over everything the chosen assumption $\mathscr{H}$ and prior $p(\theta|\mathscr{H})$ comprise

$$p(D|\mathscr{H}) = \int_\theta p(D|\theta, \mathscr{H})p(\theta|\mathscr{H})\mathrm{d}\theta. \quad (2)$$

When having several models, $p(D|\mathscr{H}_i)$ is the likelihood of the model i, which can be used in comparing the probabilities of the models, hence the term evidence of the model. (A widely used Bayesian model choice method between two models is based on Bayes factors, $p(D|\mathscr{H}_1)/p(D|\mathscr{H}_2)$, see Kass & Raftery, 1995.) The more common notation of Bayes formula, with $\mathscr{H}$ dropped, more easily causes misinterpreting the denominator $P(0)$ as some kind of probability of obtaining data D in the studied problem (or prior probability of data before the modeling).

2.1. Role of prior knowledge in statistical models

Describing the prior information explicitly distinguishes the Bayesian approach from the maximum likelihood (ML) methods. It is important to notice, however, that the role of prior knowledge is equally important in any other approach, including the ML. Basically, all generalization is based on the prior knowledge, as discussed in Lemm (1996, 1999); the training samples provide information only at those points, and the prior knowledge provides the necessary link between the training samples and the not yet measured future samples.

Recently, some important no-free-lunch (NFL) theorems have been proven, that help to understand this issue. Wolpert (1996a,b) shows that if the class of approximating functions is not limited, any learning algorithm (i.e. procedure for choosing the approximating function) can as readily perform worse or better than randomly, measured by off-training set (OTS) error, and averaged over loss functions. This theorem implies that it is not possible to find a learning algorithm that is universally better than random. In other words, if we do not assume anything a priori, the learning algorithm cannot learn anything from the training data that would generalize to the off-training set samples.

In Wolpert (1996b) and Wolpert and Macready (1995), the cross-validation (CV) method for model selection was analyzed in more depth and it was shown that the NFL theorem applies to CV also. The basic result in the papers is, that without priors on functions, choosing the function by CV performs on average as well as a random algorithm, or anti-CV, where the function that has the largest CV error is chosen. In practice, this means that if CV is used to choose from a very large (actually infinite) set of models, there is no guarantee of any generalization at all. This is easy to understand intuitively, as in such a situation the chosen algorithm is the one that happens to minimize the error on the whole training set, and if the set of algorithms is large there is a high chance that a well fitting ('overfitted') solution exists in the set. It should be noted, however, that due to computational limitations, CV can in practice be used to choose between rather few models (typically less than thousands),

so that the choice of the models imposes a very strict prior on the functions. Thus, the NFL theorems do not invalidate the use of CV in practical model selection. The implications are more in principle, emphasizing that the a priori selection of plausible solutions is necessary when using CV for model selection, and in this respect, the CV does not provide an alternative that would not require using prior knowledge in the modeling.

In practice, statistical models, like parametric models or neural networks, probably contain more often too strict priors rather than too little prior knowledge. For example, every discrete choice in the model, such as the Gaussian noise model, represents an infinite amount of prior information (Lemm, 1996). Any finite amount of information would not correspond to probability one for, e.g. the Gaussian noise model and probability zero for all the other alternatives. Also, the functional form of the model may be predetermined (as in polynomial fitting), or the number of degrees of freedom may be fixed (as in neural networks trained with error minimization without regularization). Thus, there is a large amount of prior information also in the ML models, even though the model parameters are determined solely by the data, to maximize the likelihood $p(D|w)$, or to minimize the negative log-likelihood error function. Actually the goodness of this prior knowledge is what separates 'good' and 'bad' ML models.

In the Bayesian approach, a certain part of the prior knowledge is specified more explicitly, in the form of prior distributions for the model parameters, and hyperpriors for the parameters of the prior distributions. In complex models like neural networks, the relation between the actual domain knowledge of the experts and the priors for the model parameters is not simple, and thus it may in practice be difficult to incorporate very sophisticated background information into the models via the priors of the parameters.

However, a considerable advantage of the Bayesian approach is that it gives a principled way to do inference when some of the prior knowledge is lacking or vague, so that one is not forced to guess values for attributes that are unknown. This is done by marginalization, or integrating over the posterior distribution of the unknown variables, as explained in detail in Section 3.

A lot of work has been done to find 'non-informative' priors that could be used to specify complete lack of knowledge of a parameter value. Some approaches are uniform priors, instead of Jeffreys' prior (Jeffreys, 1961), and reference priors (Berger & Bernardo, 1992). See Kass and Wasserman (1996) for a review and Yang and Berger (1997) for a large catalog of different 'non-informative' priors for various statistical models.

Among Bayesians, the use of 'non-informative' priors is often referred to as the 'objective Bayesian approach'; in contrast to informative (subjective) priors that reflect the subjective opinions of the model builder. However, in the light of the NFL theorems, this requires that the hypothesis space is already so constrained that it contains the sufficient amount of prior information that is needed to be able to learn a generalizing model (Lemm, 1999). By using 'non-informative' priors, the fixed, or guessed, choices can be moved to higher levels of hierarchical models. In Goel and Degroot (1981) it was shown that in hierarchical models, the training data contain less information of hyperparameters which are higher in the hierarchy, so that the prior and posterior for the hyperparameters become more equal. Thus, the models are less sensitive to the choices made in higher levels, implying that higher level priors are in general less informative, and thus less subjective.

In this way, the hierarchical prior structure can be used to specify partial lack of knowledge in a controllable way. For example, if it is difficult to choose between a Gaussian and a longer tailed (leptokurtic) noise model, one can include them both in the prediction model using a non-informative uniform prior for the two noise models, and the posterior probabilities of the noise models will be determined 'objectively' from the match of the noise distribution and the realized model residuals. In Section 4 we present an example of using Student's t-distribution with an unknown number of degrees of freedom ν as the noise model (thus comprising near Gaussian and longer tailed distributions), and integrating over the posterior distribution of ν in predictions. Some advice on the design of the hierarchical prior structures and robust noise models can be found in Gelman et al. (1995).

A typical attribute that is difficult to guess in advance in complex statistical models is the correct number of degrees of freedom, as it depends on the number of the training samples, distribution of noise in the samples and the complexity of the underlying phenomenon to be modeled. Also, in general the complexity of the model cannot be defined by only one number, the total number of degrees of freedom, but instead the models have multiple dimension of complexity. In the Bayesian approach, one can use a vague prior for the total complexity (called the effective number of parameters), and use a hierarchical prior structure to allow different complexity in different parts of the model. For example, the parameters may be assigned to different groups, so that in each group the parameters are assumed to have the same hyperparameter, while different groups can have different hyperparameters. Then, a hyperprior is defined to explain the distribution of all the hyperparameters. In Section 3.3 we discuss in more detail an example of this type, called the automatic relevance determination prior.

2.2. Approximations to the marginalization principle

The marginalization principle often leads to complex integrals that cannot be solved in closed form, and thus there is a multitude of approaches that differ in how the integrals are approximated.

Closest to the ML approach is the maximum a

posterior (MAP) approach, where the posterior distribution of the parameters is not considered, but the parameters are sought to maximize the posterior probability $p(w|D) \propto p(D|w)p(w)$, or to minimize the negative log-posterior cost function

$$E = -\log p(D|w) - \log p(w).$$

The weight decay regularization is an example of this technique: for Gaussian prior on the weights w the negative log-prior is $\gamma \sum_i w_i^2$. The main drawback of this approach is that it gives no tools for setting the hyperparameters due to lack of marginalization over these 'nuisance parameters'. In the weight decay example, the variance term $1/\gamma$ must be guessed, or set with some external procedure, such as CV.

A further degree of Bayesian principle is utilized in the empirical Bayesian approach, where specific values are estimated for the hyperparameters. For MLP networks, this approach was introduced by MacKay (1992) in the evidence framework (also called type II ML approach, Berger, 1985), which was the first practical Bayesian method for neural networks. In the evidence framework, the hyperparameters α are set to values that maximize the evidence of the model $p(D|\alpha)$, that is, the marginal probability for the data given the hyperparameters, integrated over the parameters, $p(D|\alpha) = \int p(D|w)p(w|\alpha)\mathrm{d}w$. A Gaussian approximation is used for the posterior of the parameters $p(w|D)$, to facilitate closed form integration, and thus the resulting posterior for w is specified by the mean of the Gaussian approximation (i.e. one network with posterior mean weights).

In a full Bayesian approach, no fixed values are estimated for any parameters or hyperparameters. Approximations are then needed for the integrations over the hyperparameters to obtain the posterior for the parameters and over the parameters to obtain the predictions of the model, as shown in Eq. (3). The correctness of the inference depends on the accuracy of the integration method, hence it depends on the problem which approximation method is appropriate. Methods for approximating the integrations in neural network models include, e.g. Markov chain Monte Carlo techniques for numerical integration, discussed in more detail in Section 3.4, ensemble learning (Barber & Bishop, 1998), which aims to approximate the posterior distribution by minimizing the Kullback–Leibler divergence between the true posterior and a parametric approximating distribution, variational approximations (Jordan, Ghahramani, Jaakkola & Saul, 1998) for approximating the integration by a tractable problem, and mean field approach (Winther, 1998), where the problem is simplified by neglecting certain dependencies between the random variables.

It is worth noticing, that also in full hierarchical Bayesian models there are large amounts of fixed prior knowledge, in the selection of the parametric form for the distributions (priors and noise models), that are based on uncertain assumptions. In such models, no guesses are made for exact values of the parameters or any smoothness coefficients or other hyperparameters, but guesses are made for the exact forms of their distributions. The goodness of the model depends on these guesses, which in practical applications makes it necessary to carefully validate the models, using, e.g. Bayesian posterior analysis (Gelman et al., 1995), or CV (Geisser, 1975; Gelfand, 1996; Vehtari & Lampinen, 2000). This also implies that in practice the Bayesian approach is often more sensitive to the prior assumptions than more classical methods. This is discussed in more detail in Section 3.5.

3. Bayesian learning for MLP networks

In the following, we give a short overview of the Bayesian approach for neural networks. We concentrate on MLP networks and Markov chain Monte Carlo methods for computing the integrations, following the approach introduced in Neal (1992). A detailed treatment can be found in Neal (1996), which also describes the use of the flexible Bayesian modeling (FBM) software package,[1] that was the main tool used in the case problems reviewed in this paper. The result of Bayesian modeling is the conditional probability distribution of unobserved variables of interest, given the observed data. In Bayesian MLP the natural end variables are the predictions of the model for new inputs, while the posterior distribution of the network weights is rarely of much interest.

The posterior predictive distribution of output $\mathbf{y}^{\text{new}}$ for the new input $\mathbf{x}^{\text{new}}$ given the training data $D = \{\mathbf{x}^{(1)}, \mathbf{y}^{(1)}, \ldots, \mathbf{x}^{(n)}, \mathbf{y}^{(n)})\}$, is obtained by integrating the predictions of the model with respect to the posterior distribution of the model,

$$p(\mathbf{y}^{\text{new}}|\mathbf{x}^{\text{new}}, D) = \int p(\mathbf{y}^{\text{new}}|\mathbf{x}^{\text{new}}, \theta)p(\theta|D)\mathrm{d}\theta, \tag{3}$$

where θ denotes all the model parameters and hyperparameters of the prior structures.

The probability model for the measurements, $p(\mathbf{y}|\mathbf{x},\theta)$, contains the chosen approximation functions and noise models. It defines also the likelihood part in the posterior probability term, $p(\theta|D) \propto p(D|\theta)p(\theta)$. The probability model in a regression problem with additive error is

$$y = f(\mathbf{x}; \theta_w) + e, \tag{4}$$

where $f()$ is, e.g. the MLP function

$$f(\mathbf{x}, \theta_w) = \mathbf{b}^2 + \mathbf{w}^2 \tanh(\mathbf{b}^1 + \mathbf{w}^1 \mathbf{x}). \tag{5}$$

The θ_w denotes all the parameters $\mathbf{w}^1$, $\mathbf{b}^1$, $\mathbf{w}^2$, $\mathbf{b}^2$, which are the hidden layer weights and biases, and the output layer weights and biases, respectively. The random variable e is the model residual. Multivariate problems (with several outputs) can be handled by changing the output in Eq. (4) to be a vector (thus having common residual model for all

[1] URL:http://www.cs.toronto.edu/~radford/fbm.software.html

outputs), or by completely separate models, or as a hierarchical model with some common parts (i.e. common hidden layer, separate output weights, and common or hierarchical noise model). In two class classification problems, the probability that a binary-valued target, y, has value 1 can be computed by the logistic transformation as

$$p(y = 1|\mathbf{x}, \theta_w) = [1 + \exp(-f(\mathbf{x}, \theta_w))]^{-1}, \quad (6)$$

and in many class classification problems the probability that a class target, y, has value j can be computed by the softmax transformation (or cross-entropy) as

$$p(y = j|\mathbf{x}, \theta_w) = \frac{\exp(f_j(\mathbf{x}, \theta_w))}{\sum_k \exp(f_k(\mathbf{x}, \theta_w))}. \quad (7)$$

3.1. Residual models

In the following, we use the notation $r \sim F(a)$ as shorthand for $p(r) = F(r|a)$ where a denotes the parameters of the distribution F, and the random variable argument r is not shown explicitly. The commonly used Gaussian noise model is

$$e \sim N(0, \sigma^2), \quad (8)$$

where $N(\mu, \sigma^2)$ denotes a normal distribution with mean μ and variance σ^2. In choosing the hyperprior for σ^2 there may be some knowledge to use a somewhat informative prior. For example, the minimum reasonable value for the noise variance can be estimated from measurement accuracy or from repeated experiments. Whether the hyperprior is informative or non-informative, it is convenient to choose the form of the distribution in accordance with the method used to sample from the posterior distribution. Note that the results are in general not very sensitive to the choices made in the hyperprior level, as discussed in Section 2.1 and confirmed in many studies (see, e.g. Rasmussen, 1996). However, this should be checked in serious analysis, especially if the form of the prior needs to be compromised for reasons of computational convenience. In the framework used in this study (see Section 3.4) the hyperparameters are sampled by Gibbs sampling. Convenient priors are then conjugate distributions, that produce full conditional posteriors of the same form. For the variance of the Gaussian, a conjugate distribution is the inverse Gamma, producing the prior

$$\sigma^2 \sim \text{Inv-gamma}(\sigma_0^2, \nu_\sigma), \quad (9)$$

with parametrization

$$\text{Inv-gamma}(\sigma_0^2, \nu) \propto (\sigma^2)^{-(\nu/2+1)} \exp\left(-\frac{1}{2}\nu\sigma_0^2\sigma^{-2}\right),$$

which is equal to a scaled inverse chi-square distribution (Gelman et al., 1995, Appendix A). The parameter ν is the number of degrees of freedom and σ_0^2 is a scale parameter. In this parametrization, the prior is equivalent to having ν prior measurements with averaged squared deviation σ_0. The fixed values for σ_0 and ν_σ can be chosen so as to produce a vague prior for σ^2, that is reasonably flat over the range of parameter values that could plausibly arise. We have used $\sigma_0 = 0.05$ and $\nu_\sigma = 0.5$, similar to those used in Neal (1996, 1998).

In Vehtari and Lampinen (1999), we analyzed a multivariate regression problem where the residuals of the outputs may be correlated. For a multivariate normal residual model with full covariance matrix, a conjugate hyperprior is the inverse Wishart distribution, allowing Gibbs sampling for the covariance matrix.

In the noise model in Eq. (8), the same noise variance σ^2 is assumed in each sample. In heteroscedastic regression problems each sample $(\mathbf{x}^n, \mathbf{y}^n)$ can have different noise variance $(\sigma^2)^n$ with all the variances governed by a common prior, corresponding to, e.g. a noise model

$$y^n = f(\mathbf{x}^n; \mathbf{w}^1, \mathbf{b}^1, \mathbf{w}^2, \mathbf{b}^2) + e^n \quad (10)$$

$$e^n \sim N(0, (\sigma^2)^n) \quad (11)$$

$$(\sigma^2)^n \sim \text{Inv-gamma}(\sigma_{\text{ave}}, \nu_\sigma) \quad (12)$$

$$\sigma_{\text{ave}} \sim \text{Inv-gamma}(\sigma_0 \nu_{\sigma,\text{ave}}), \quad (13)$$

where the fixed hyperparameters are ν_σ, σ_0 and $\nu_{\sigma,\text{ave}}$. Here, the prior spread of the variances $(\sigma^2)^n$ around the average variance σ_{ave}, determined by ν_σ, is fixed. In this parametrization, the residual model is equal to Student's t-distribution with fixed degrees of freedom. To allow for a higher probability for models with similar noise variances, the hyperparameter ν_σ can also be given a hyperprior, so that models with similar variances can have large ν_σ, corresponding to a tight prior for the spread of variances $(\sigma^2)^n$, and thus giving high probability for each realized variance. This is approximately the same as the t-distribution noise model with unknown degrees of freedom. Thus similar treatment results, whether we assume normal residuals with different variances, or a common longer tailed t-distribution residual model (Geweke, 1993). The latter is preferable, as it leads to simpler noise models, and will be discussed in more detail below.

In heteroscedastic problems, the noise variance can be functionally dependent on some explanatory variables, typically on some subset of the model inputs, so that the model for the noise variance might be

$$(\sigma^2)^n = F(\mathbf{x}^n; \theta_{\text{noise}}) + \epsilon \quad (14)$$

$$\epsilon \sim \text{Inv-gamma}(\sigma_0, \nu_\sigma) \quad (15)$$

with fixed σ_0 and ν_σ. See Bishop and Qazaz (1997) for an example of an input dependent noise model, where a separate MLP model is used to estimate the dependence of the noise variance on the inputs.

Often in practical problems, the Gaussian residual model

is not applicable. There may be error sources that have non-Gaussian density, or the target function may contain peaks, but the training data are not sufficient to estimate them, or the data are heteroscedastic, with different noise variances in each sample. With a Gaussian residual model, samples with exceptionally large residuals must be handled as outliers, using pre- and post-filtering and manual manipulation of data. Another option is to use a longer tailed residual model that allows a small portion of samples to have large errors. An often used model is the Laplace (or double exponential) distribution. When the appropriate form for the residual distribution is not known in advance, the correct Bayesian treatment is to integrate over all a priori plausible forms.

In this study we have used Student's t-distribution, where the tails can be controlled by choosing the number of degrees of freedom v in the distribution. As this number is difficult to guess in advance, we set a hierarchical prior for it, and in the prediction we integrate over the posterior distribution given the data. Thus, the tails are determined by the fit of the model to the data. The integration over the degrees of freedom can be done by Gibbs sampling (see Section 3.4) for discretized values of v, so that the residual model is

$$e \sim t_v(0, \sigma^2) \tag{16}$$

$$v = V[i] \tag{17}$$

$$i \sim U_d(1, K) \tag{18}$$

$$V[1:K] = [2, 2.3, 2.6, 3, 3.5, 4, 4.5, 5:1:10, 12:2:20, 25:5:50] \tag{19}$$

$$\sigma^2 \sim \text{Inv-gamma}(\sigma_0, v_\sigma) \tag{20}$$

where $[a: s: b]$ denotes the set of values from a to b with step s, and $U_d(a, b)$ is a uniform distribution of integer values between a and b. The discretization is chosen so that an equal prior for each value results in roughly (truncated) exponential prior on v (Geweke, 1993; Spiegelhalter et al., 1996). Another simple way to sample for v, without discretization, is by the Metropolis–Hastings algorithm (Hastings, 1970), which in our experiments gave equal results but slightly slower convergence.

3.2. Priors for the model parameters

Typical prior assumptions in regularization theory are related to the smoothness of the approximation. In Tikhonov regularization (Bishop, 1995), which is a widely used regularization method in, e.g. inverse problems, functions with large derivatives of chosen order are penalized. With an MLP model, minimizing the curvature (second derivative, Bishop, 1993) or training the derivatives to given target values (Lampinen & Selonen, 1997) leads to a rather complex treatise as the partial derivatives of the non-linear models depend on all the other inputs and weights.

A convenient commonly used prior distribution is the Gaussian, which in linear models is directly related to model derivatives, but has a more complex interpretation in the non-linear MLP case, as discussed in the next section. The Gaussian priors for the weights are

$$\mathbf{w}^1 \sim N(0, \alpha_{w^1}) \tag{21}$$

$$\mathbf{b}^1 \sim N(0, \alpha_{b^1}) \tag{22}$$

$$\mathbf{w}^2 \sim N(0, \alpha_{w^2}) \tag{23}$$

$$\mathbf{b}^2 \sim N(0, \alpha_{b^2}) \tag{24}$$

where the α's are the variance hyperparameters. The conjugate inverse Gamma hyperprior is

$$\alpha_j \sim \text{Inv-gamma}(\alpha_{0,j}, v_{\alpha,j}) \tag{25}$$

similarly to hyperpriors in the Gaussian noise model. The fixed values for the highest level hyperparameters in the case studies were similar to those used in Neal (1996, 1998). Appropriate hyperpriors depend somewhat on the network topology. As discussed in Neal (1996) the average weights can be assumed to be smaller when there are more feeding units, e.g. the hyperprior for $\mathbf{w}^1$ is scaled according to the number of inputs K. Typical values were

$$v_{\alpha,w^1} = 0.5$$

$$\alpha_{0,w^1} = (0.05/K^{1/v_{\alpha,w^1}})^2.$$

3.3. Automatic relevance determination prior and importance of inputs

In this section, we discuss a simple hierarchical prior for the MLP weights, called automatic relevance determination (ARD) (MacKay, 1994; Neal, 1996, 1998). In ARD, each group of weights connected to the same input $k \in \{1, \ldots, K\}$ has common variance hyperparameters, while the weight groups can have different hyperparameters. Example of the ARD prior, used in this study, is

$$w_{kj} \sim N(0, \alpha_k), \tag{26}$$

$$\alpha_k \sim \text{Inv-gamma}(\alpha_{\text{ave}}, v_\alpha) \tag{27}$$

$$\alpha_{\text{ave}} \sim \text{Inv-gamma}(\alpha_0, v_{\alpha,\text{ave}}), \tag{28}$$

where the average scale of the α_k is determined by the next level hyperparameters, in similar fashion as in the heteroscedastic noise model example above. The ARD can be used also in the evidence framework, where each α_k is estimated by ML, without any priors or penalty for large variability. The fixed values used in the case studies were $v_\alpha = 0.5$, $\alpha_0 = (0.05/K^{1/v_\alpha})^2$ and $v_{\alpha,\text{ave}} = 1$, corresponding to vague hyperpriors that let the α_{ave} and α_k be determined by data.

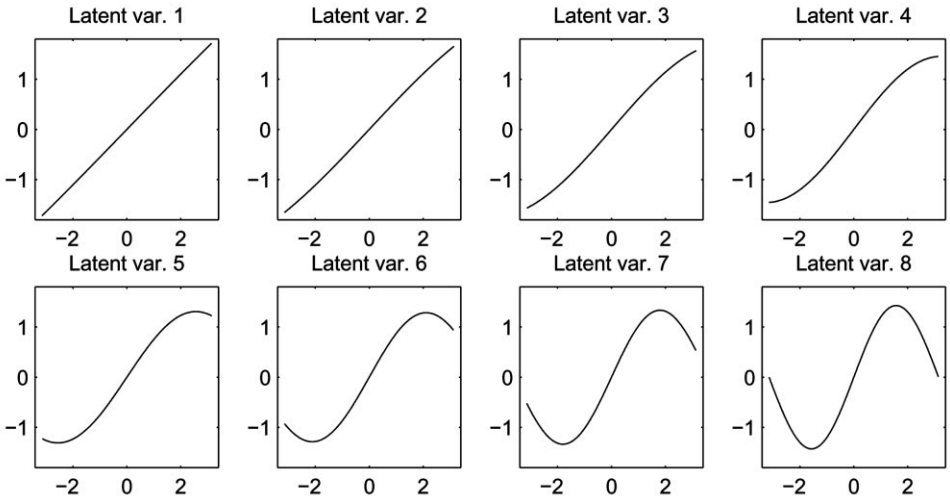

Fig. 1. Example of ARD and importance of inputs. The target function is an additive function of eight inputs. The plots show the univariate transformations of the inputs. The predictive importance of every input is equal, in RMSE terms, as the latent functions are scaled to equal variance over the uniform input distribution $U(-3, 3)$.

Hyperparameter v_α could also be given a hyperprior in similar fashion as in the heteroscedastic noise model example.

The ARD prior was proposed as an automatic method for determining the relevance of the inputs (MacKay, 1994; Neal, 1996), as irrelevant inputs should have smaller weights in the connections to the hidden units than more important weights. With separate hyperparameters, the weights from irrelevant inputs can have tighter priors, which reduces such weights more effectively towards zero than having the common larger variance for all the input weights.

Determining the relevance of inputs has great importance in practical modeling problems, in both choosing the inputs in the models as well as in analyzing the final model. See Sarle (1997) for a general discussion on ways to assess the importance of inputs in non-linear models. The most common notions of importance are predictive importance (the increase in generalization error if the variable is omitted from the model) and causal importance (change of model outputs caused by the change of input variable). Note that causal importance is directly measurable only if the inputs are uncorrelated (so that inputs can be manipulated independently), and that it is not related to the causality relations in the actual system to be modeled.

In ARD, the relevance measure of an input is related to the size of the weights connected to that input. In linear models, these weights define the partial derivatives of the output with respect to the inputs, which is equal to the predictive importance of the input, and in the case of non-correlated inputs, also the causal importance. In non-linear neural networks, the situation is, however, more complex, since small weights in the first layer can be compensated by large weights in other layers, and the non-linearity in the hidden units changes the effect of the input in a way that depends on all the other inputs.

To illustrate the effect of an ARD prior, consider a K-J–1 MLP with linear output layer,

$$y = \sum_{j=1}^{J} v_j S(\sum_{k=1}^{K} w_{kj} x_k).$$

The d-th order partial derivative of the mapping is

$$\frac{\partial^d y}{\partial (x_k)^d} = \sum_j v_j (w_{kj})^d S^{(d)}(\sum_k w_{kj} x_k),$$

where $S^{(d)}$ is the d-th derivative of S. Thus constraining the first layer weights has largest effect on higher order derivatives, in the d-th order polynomial term $(w_{kj})^d$. This may partly explain the success of weight decay regularization, as this type of prior is an effective smoothing prior. On the other hand, to produce a linear mapping with small high order derivatives, the first layer weights would need to be small, so that the sigmoids operate on the linear part, and the second layer weights correspondingly larger. Thus, the first layer weights do not measure the first derivative, or the linear relation, no matter how important it is. The network may also contain direct input-to-output weights to account for any linear relation (Neal, 1996), but the ARD coefficients of these weights are not comparable to the ARD coefficients of the hidden layer weights. Note that adding input-to-output weights makes the model less identifiable and may slow down the convergence of MCMC considerably (Neal, 1998).

In the following simple example, we demonstrate how the non-linearity of the input has the largest effect on the relevance score of the ARD, instead of the predictive or causal importance. The target function is an additive function of eight inputs (see Fig. 1), with equal predictive importance for every input. The network weights (using the evidence approximation, MacKay, 1992) are shown in Fig. 2, from

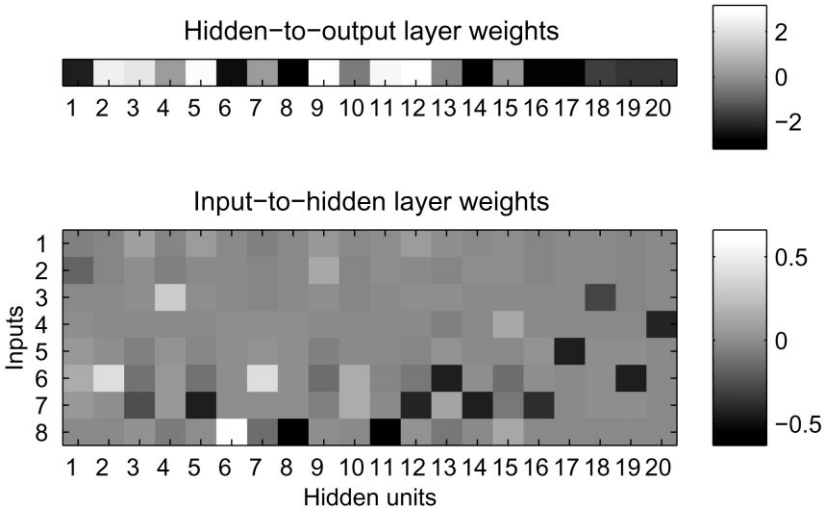

Fig. 2. Network weights for the test function in Fig. 1, estimated using the evidence framework (MacKay, 1992).

where it is easy to see how the weights connected to the inputs with linear transformation are smallest. Fig. 3 shows the predictive importance and the mean absolute values of the first and second order derivatives of the output with respect to each input, and the relevance estimates from the ARD (posterior standard deviation of the Gaussian prior distributions for each weight group). The example illustrates how the inputs with a large but linear effect are given low relevance measures by ARD. For this reason, one should be cautious of using the ARD to choose or remove the inputs in the models, or to rank the variables according to importance in the analysis of the model. Note, however, that ARD is often a very favorable prior, as demonstrated in the case studies in this contribution, since it loosens the more strict assumption that all the input weight groups should have the same variance (or non-linearity). So, unless the variance is actually assumed to be the same, ARD should be used as a less informative but more probably correct prior.

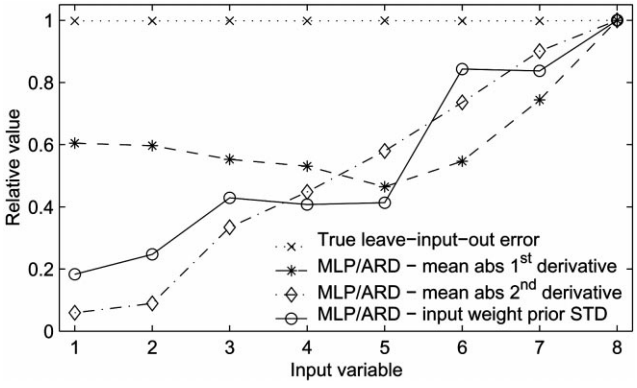

Fig. 3. Different measures of importance of inputs for the test function in Fig. 1. Note how the ARD coefficients are closer to the second derivatives, than to the first derivatives (local causal importance), or to the error due to leaving input out (predictive importance).

3.4. Markov chain Monte Carlo method

Neal has introduced an MCMC implementation of Bayesian learning for MLPs (Neal, 1996). A good introduction to basic MCMC methods and many applications in statistical data analysis can be found in Gilks, Richardson and Spiegelhalter (1996) and a more theoretical treatment in Roberts and Casella (1999).

In MCMC the complex integrals in the marginalization are approximated via drawing samples from the joint probability distribution of all the model parameters and hyperparameters. For example, with squared error loss the best guess for model prediction (with additive zero-mean noise model), corresponds to the expectation of the posterior predictive distribution in Eq. (3)

$$\hat{\mathbf{y}}^{\text{new}} = E[\mathbf{y}^{\text{new}}|\mathbf{x}^{\text{new}}, D] = \int f(\mathbf{x}^{\text{new}}, \theta) p(\theta|D) \mathrm{d}\theta. \tag{29}$$

This is approximated using a sample of values $\theta^{(t)}$ drawn from the posterior distribution of parameters

$$\hat{\mathbf{y}}^{\text{new}} \approx \frac{1}{N} \sum_{t=1}^{N} f(\mathbf{x}^{\text{new}}, \theta^{(t)}). \tag{30}$$

Note that samples from the posterior distribution are drawn during the 'learning phase', which may be computationally very expensive, but predictions for the new data can be calculated quickly using the same stored samples and Eq. (30).

In the MCMC, samples are generated using a Markov chain that has the desired posterior distribution as its stationary distribution. In the framework introduced by Neal the hybrid Monte Carlo (HMC) algorithm (Duane, Kennedy, Pendleton, & Roweth, 1987) is used for sampling the parameters and Gibbs sampling (Geman & Geman, 1984) for hyperparameters. For other possible sampling schemes, see, e.g. Müller and Rios Insua (1998) and de Freitas et al.

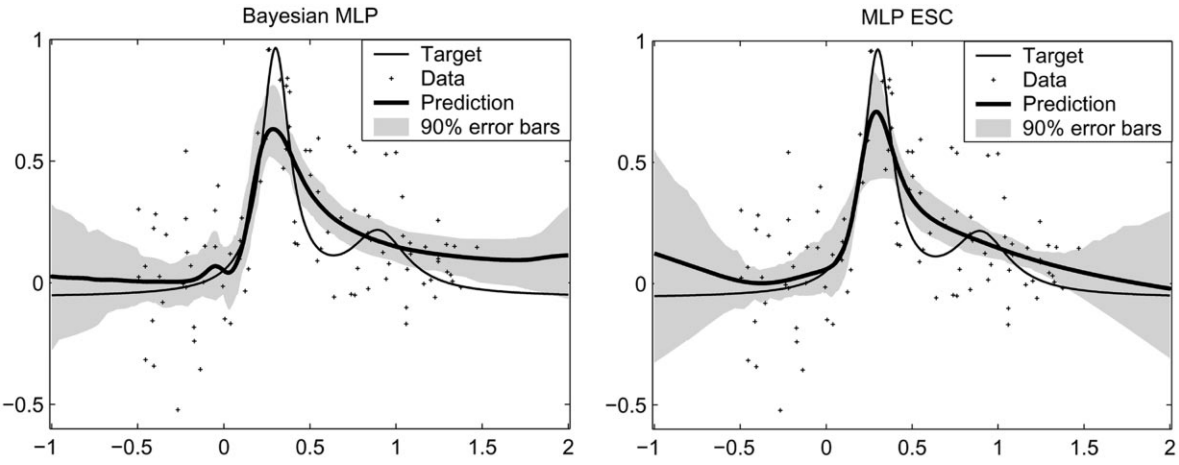

Fig. 4. Test function in demonstrating the sensitivity of Bayesian MLP and early-stopped committee (MLP ESC) to the wrong noise model. The figure shows one sample of noise realization and the resulting predictions, with Bayesian MLP in the left and MLP ESC in the right figure. See text for explanation of the error bars.

(2000). HMC is an elaborate Monte Carlo method, which makes efficient use of gradient information to reduce random walk behavior. The gradient indicates in which direction one should go to find states with high probability. The detailed description of the algorithm is not repeated here, see Neal (1996) or the reference list in the web-page of the FBM software.[2] In the Gibbs sampling, each parameter in turn is sampled from the full conditional distribution of the parameter given all the other parameters and data. As an example, the hyperparameter α_{w^1} in Eq. (21) is sampled from

$$p(\alpha_{w^1}|D, \mathbf{w}^1, \mathbf{b}^1, \mathbf{w}^2, \mathbf{b}^2, \sigma^2, ...) = p(\alpha_{w^1}|\mathbf{w}^1)$$
$$\propto p(\mathbf{w}^1|\alpha_{w^1})p(\alpha_{w^1}) \quad (31)$$

which can be done efficiently if the prior is chosen to be a conjugate distribution. The Gibbs sampling is the main sampling method in the BUGS system[3] by Spiegelhalter, Thomas, Best, and Gilks, (1996) which is a Bayesian modeling tool very convenient in experimenting with the hierarchical Bayesian models.

When the amount of data increases, the evidence from the data causes the probability mass to concentrate in a smaller area and we need less samples from the posterior distribution. Also, fewer samples are needed to evaluate the mean of the predictive distribution than the tail-quantiles, such as the 10 and 90% quantiles. So, depending on the problem, some hundreds of samples may be enough for practical purposes. Note that due to autocorrelations in the Markov chain, getting some 100 near-independent samples from a converged chain may require tens of thousands of samples in the chain, which may require several hours of CPU-time on a standard workstation.

[2] URL:http://www.cs.toronto.edu/~radford/fbm.software.html
[3] URL:http://www.mrc-bsu.cam.ac.uk/bugs

For convergence diagnostics, we used visual inspection of trends and the potential scale reduction method (Gelman, 1996). Alternative convergence diagnostics have been reviewed, e.g. in Brooks and Roberts (1999) and Roberts and Casella (1999). See Vehtari, Särkkä, and Lampinen (2000) for discussion on the choice of the starting values and the number of chains. Choosing the initial values with early-stopping can be used to reduce the burn-in time, when the chain has not yet reached the equilibrium distribution. In general, the authors' experience suggests that the convergence of the MCMC methods for MLP is slower than usually assumed, so that in many of the published studies, the MCMC chains may still have been in the burn-in stage, producing a sort of early-stopping effect to the selection of the model complexity.

3.5. On sensitivity of the Bayesian approach to the prior distribution

As explained above, the Bayesian approach is based on averaging probable models, where the probability is computed from the chosen distributions for the noise models, parameters, etc. Thus the approach may be more sensitive to bad guesses for these distributions than more classical methods, where the model selection is carried out as an external procedure, such as CV that is based on fewer assumptions (mainly the assumption that the training and validation sets are not correlated). In this respect, the Bayesian models can also be overfitted in terms of classical model fitting, to produce too complex models and too small posterior estimates for the noise variance. To check the assumptions of the Bayesian models, we always carry out the modeling with simple classical methods (like linear models, early-stopped committees of MLPs, etc.). If the Bayesian model gives inferior results (measured from test

Table 1
Demonstration of the sensitivity of Bayesian MLP and MLP ESC to wrong noise model. For both models the noise model was Gaussian, and the actual noise Gaussian or Laplacian (double exponential). The statistical significance of the difference is tested by pairwise t-test, and the shown p-value is the probability of observing equal or larger error in the means if the two methods are equal. The errors are RMS errors of the prediction from the true target function

Noise	Bayesian MLP RMSE	MLP ESC RMSE	Significance of the difference (two-tailed)
Gaussian	0.2779	0.2784	0.85
Laplacian	0.2828	0.2766	0.012

set or cross-validated), some of the assumptions are questionable.

The following computer simulation illustrates the sensitivity of the Bayesian approach to the correctness of the noise model, compared to the early-stopped committee (ESC), that is a robust reference method used in all the case studies. The basic early stopping is statistically rather inefficient, as it is very sensitive to the initial conditions of the weights and only part of the available data are used to train the model. These limitations can easily be alleviated by using a committee of early-stopping MLPs, with different partitioning of the data to training and stopping sets for each MLP (Krogh & Vedelsby, 1995). When used with caution, ESC is a good baseline method for MLPs.

The target function and data are shown in Fig. 4. The modeling test was repeated 100 times with different realizations of Gaussian or Laplacian (double exponential) noise. The model was 1–10–1 MLP with a Gaussian noise model. The figure shows one sample of noise and the resulting predictions. The 90% error bars, or confidence intervals, are for the predicted conditional mean of the output given the input. Thus, the measurement noise is not included in the limits. For the ESC, the intervals are simply computed separately for each x-value from 100 networks. Computing the confidence limits for early-stopped committees is not straightforward, but this very simple ad hoc method often gives similar results to the Bayesian MLP treatment. The summary of the experiment is shown in Table 1. Using a paired t-test, the ESC is significantly better than the Bayesian model when the noise model is wrong. In this simple problem, both methods are equal for the correct noise model. The correct Bayesian approach of integrating over the noise models, as explained in Section 2.1 and shown in practice in a case problem in Section 4, would, of course, have no trouble in this example.

The implication of this issue in practical applications is that a Bayesian approach usually requires more expert work than the standard approach, either to devise reasonable assumptions for the distributions, or to include different options in the models and integrate over them, but that done, the results are, in our experience, consistently better than with other approaches.

4. Case I: regression task in concrete quality assumption

In this section, we report results of using Bayesian MLPs for regression in a concrete quality estimation problem. The goal of the project was to develop a model for predicting the quality properties of concrete, as a part of a large quality control program of the industrial partner of the project. The quality variables included, e.g. compression strengths and densities for 1, 28 and 91 days after casting, and bleeding (water extraction), spread, slump and *air*-%, that measure the properties of fresh concrete. These quality measurements depend on the properties of the stone material (natural or crushed, size and shape distributions of the grains, mineralogical composition), additives, and the amount of cement and water. In the study we had 27 explanatory variables.

Collecting the samples for statistical modeling is rather expensive in this application, as each sample requires preparation of the sand mixture, casting the test pieces and waiting for 91 days for the final tests. In the study, we had 215 samples designed to cover the practical range of the variables, collected by the concrete manufacturing company. In small sample problems, the selection of correct model complexity is more important and needs to be done with finer resolution than in problems with large amounts of data. This makes hierarchical Bayesian models a tempting alternative. In the study, we used MLP networks containing 10 hidden units (chosen by coarse experiments, that indicated that this size of network contained a sufficient surplus of degrees of freedom compared to the actually required number). As a reference method, we used an ESC of 10 MLP networks, with different division of data into training and stopping sets for each member. The networks were initialized to near zero weights to guarantee that the mapping is smooth in the beginning.

In the following, we report the results for one variable, *air*-%, which measures the volume percentage of air in the concrete. As the *air*-% is positive and has a very skewed distribution (with mean 2.4% and median 1.7%) we used logarithmic transformation for the variable. This ensures positiveness and allows use of much simpler additive noise models than in the case of a nearly exponentially distributed variable.

The performance of the models was estimated by 10-fold cross-validation. To compare the models we used a paired t-test with the CV. This method exhibits a somewhat elevated probability of type I error (to suggest a difference when no difference exists) and low type II error (to miss a difference when it exists), as analyzed in Dietterich (1998). The reference methods were ESC (MLP ESC), and a Gaussian process (GP) model (Neal, 1999), which is a non-parametric regression method, with priors imposed directly on the correlation function of the resulting approximation. The GP approach is a very viable alternative for MLP models, at least in problems where the training sample size is not very large.

The estimated prediction errors are presented in Table 2.

Table 2
Performance comparison of various MLP models and a Gaussian process (GP) model in predicting the *air-%* variable in concrete manufacturing. The presented RMSE values show the standardized model residuals relative to the standard deviation of the data. See Table 4 for pairwise comparison of the models

	Method	Noise model	ARD	RMSE	std
1	MLP ESC	N		0.30	0.04
2	Bayesian MLP	N		0.26	0.04
3	Bayesian MLP	t_v		0.24	0.03
4	Bayesian MLP	N	yes	0.21	0.02
5	Bayesian MLP	t_v	yes	0.19	0.02
6	Gaussian process	t_v	yes	0.19	0.02

In the column *Noise model* the letter N indicates normal noise model and t_v Student's t-distribution with unknown degrees of freedom v, respectively. The MCMC sampling for the basic models was done with the FBM software, and the sampling of the model with t_v noise distribution was done with Matlab-code derived from the Netlab[4] toolbox.

The posterior values for v, presented in Table 3, correspond to a rather long tailed distribution for the model residuals. With ARD the tails are even longer, as the reduction of some weights very near to zero corresponds to simpler models and larger residuals for some samples.

Table 4 shows p-values for pairwise comparisons of the methods, obtained from paired t-tests. Note that the test quantity in comparing the predictive performance of the models in Tables 2 and 4 is RMS error, even though the long tailed residual models allow large errors, that cost much in the RMS error function. This serves as posterior model checking, since the RMSE is the relevant error measure in the application, and we want to be sure that the long tails of model residuals have not led to undermodeling. In general, making the residual model more flexible shifts the posterior mass towards simpler (a priori more probable) models, since high likelihood can be obtained by matching the residual model and the realized residuals.

Some conclusions from the results are listed in the following.

- The best models, GP model and Bayesian MLP, with ARD and Student's t_v-distribution with unknown degrees of freedom as noise model, had equal performance.
- The best models were those with most flexible (less informative) priors. Within MLP models, the t_v noise model outperformed all the Gaussian noise models on confidence level at least 98% ($p < 0.02$), and the MLP with t_v without ARD at confidence level 93% ($p < 0.07$, often considered as marginally significant).
- The MLP ESC and the basic Bayesian MLP with Gaussian noise model did not differ significantly. Just adding the Bayesian treatment for the basic model does not help in this application, if the possibility for using less informative and hierarchical priors is not utilized.
- Adding ARD made the Bayesian model significantly better ($p < 0.01$) than MLP ESC.
- Using just the longer tail noise model t_v without ARD made the Bayesian model better than the ESC MLP at confidence level 95% ($p < 0.05$).

[4] URL:http://www.ncrg.aston.ac.uk/netlab/

Table 3
Posterior analysis of the number of degrees of freedom v in the t-distribution residual model. The table shows the posterior mean and 10 and 90% quantiles of v

Method	Noise model	ARD	$E[v]$	10%	90%
Bayesian MLP	t_v		3.5	2.3	5.0
Bayesian MLP	t_v	yes	2.8	2.0	3.5

Fig. 5 shows the distribution of the ARD coefficients for model 5. The variable names are not disclosed by request of the industrial partner of the project. The variable indices printed in bold face correspond to the variables chosen in the final model. The selection was done manually, aided by backward elimination technique.

The computational burden of the Bayesian approach in this case study was rather heavy. As an example, sampling of the best model, using ARD and a t-distributed noise model with unknown degrees of freedom, took about 16 h of CPU time on a 660 MHz Compaq alpha workstation. The Matlab-implementation was roughly four times slower than the corresponding native C-implementation in the FBM software, that was used to run the simpler models.

The main HMC parameters were: the length of individual chains was 100, step size 0.5 with Neal's heuristic step size adjustment, persistence parameter 0.9, and window length in windowing 5. The burn-in stage contained 16,000 chains and the actual sampling 80,000 chains, from which 100 samples were stored for the posterior analysis.

For comparison, the number of iterations corresponds to about 1000 trainings of error-minization models with 1000 iterations in each training. To find values for two hyperparameters by CV, with discretization of 10 possible values for

Table 4
Pairwise comparisons of various MLP models in predicting the *air-%* variable. The values in the matrix are p-values, obtained from paired t-tests. The p-values have been rounded up to nearest whole number in percent (so the value 1 indicates a p-value less than 0.01). The p-values are reported in the column of the winning method. Looking row-wise, you see which methods out-performed the method of that row

Reference method		Noise model	ARD	Comparison					
				1	2	3	4	5	6
1	MLP ESC	N			15	5	1	1	1
2	Bayesian MLP	N		–	–	7	4	1	2
3	Bayesian MLP	t_v		–	–	–	18	7	7
4	Bayesian MLP	N	yes	–	–	–	–	2	14
5	Bayesian MLP	t_v	yes	–	–	–	–	–	42
6	Gaussian Process	t_v	yes	–	–	–	–	–	

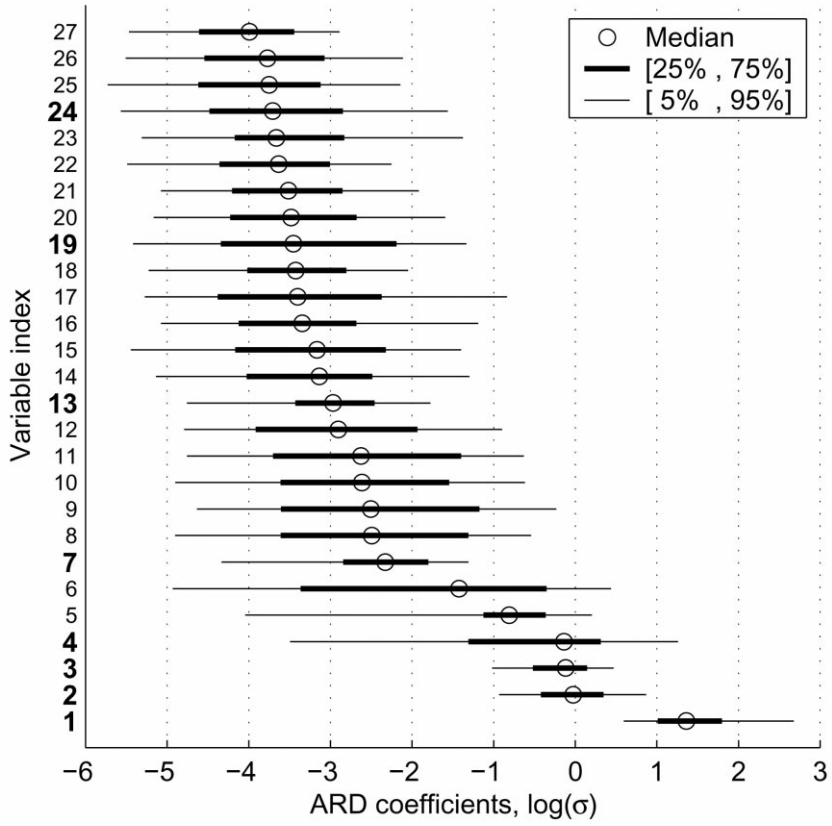

Fig. 5. Posterior distribution of the ARD coefficients (standard deviation of the Gaussian prior of the input weights) for model 5. The variable indices printed in bold face are those chosen in the final model.

each hyperparameter, would require about the same amount of CPU time (note that the 10-fold CV error would require 10,000 iterations). Just in the ARD, there were 27 hyperparameters for the input weights, for which it would have been practically impossible to search for any regularizers by CV. Even the simple forward selection of the inputs in the model, with fixed regularization, would have required $\sum_{n=2}^{27} n = 377$ CV-error evaluations, requiring the same order of magnitude of CPU time as the Bayesian model. The Bayesian approach is undeniably expensive in computational load, but in complex real world problems there are few cheap alternatives.

In the review above, we have presented the results for only one variable in the study. Rather similar results were obtained for the other variables, showing that a worked out Bayesian model with realistic noise models and priors was consistently the best model, measured by CV. The Bayesian MLP and the GP model had very similar performances for all the target variables, so that the choice between them in this application is a matter of convenience. Some hierarchical linear models were also tested in the problem, but the preliminary results were so clearly inferior to those of basic MLP models (MLP ESC, Bayesian MLP + normal noise model) that those models were not analyzed further and are not reported here. Preliminary tests were also performed with the evidence framework, but the re-estimating algorithm of the hyperparameters (MacKay, 1992) did not converge on realistic network sizes, which was expected as it is well known that the Gaussian approximation for the posterior does not hold when the number of samples is small.

5. Case II: inverse problem in electrical impedance tomography

In this section we report results on using Bayesian MLPs for solving the ill-posed inverse problem in electrical impedance tomography (EIT). The full report of the proposed approach is presented in Lampinen, Vehtari and Leinonen (1999).

The aim in EIT is to recover the internal structure of an object from surface measurements. A number of electrodes are attached to the surface of the object and current patterns are injected through the electrodes and the resulting potentials are measured. The inverse problem in EIT, estimating the conductivity distribution from the surface potentials, is known to be severely ill-posed, thus some regularization methods must be used to obtain feasible results (Vauhkonen et al., 1997).

Fig. 6 shows a simulated example of the EIT problem. The volume bounded by the circles in the image represents

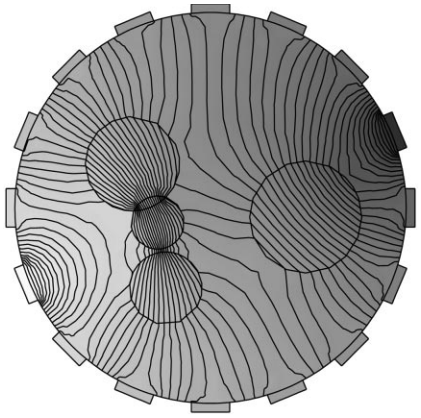

Fig. 6. Example of the EIT measurement. The simulated bubble formation is bounded by the circles. The current is injected from the electrode with the lightest color and the opposite electrode is grounded. The gray level and the contour curves show the resulting potential field.

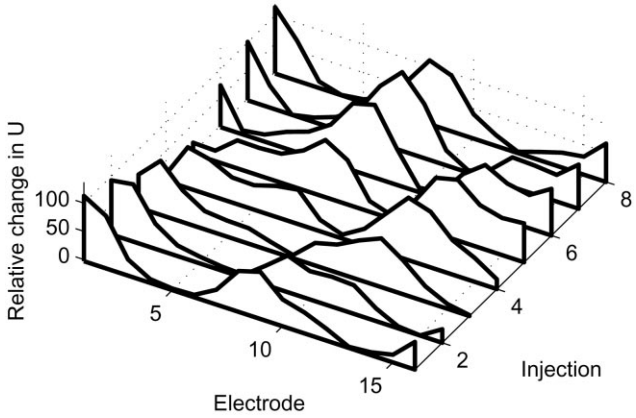

Fig. 7. Relative changes in potentials compared to homogeneous background. The eight curves correspond to injections from eight different electrodes.

gas bubble floating in liquid. The conductance of the gas is much lower than that of the liquid, producing the equipotential curves shown in the figure. Fig. 7 shows the resulting potential signals, from which the image is to be recovered.

In Lampinen et al. (1999) we proposed a novel feed-forward solution for the reconstruction problem. The approach is based on computing the principal component decomposition for the potential signals and the eigenimages of the bubble distribution from the autocorrelation model of the bubbles. The input to the MLP is the projection of the potential signals to the first principal components, and the MLP gives the coefficients for reconstructing the image as a weighted sum of the eigenimages. The projection of the potentials and the images to the eigenspace reduces correlations from the input and the output data of the network and detaches the actual inverse problem from the representation of the potential signals and image data.

The reconstruction was based on 20 principal components of the 128 dimensional potential signal and 30 eigenimages with resolution 41×41 pixels. The training data consisted of 500 simulated bubble formations with one to ten overlapping circular bubbles in each image. To compute the reconstructions, MLPs containing 30 hidden units were used. Models tested were MLP ESC and Bayesian MLP (see Section 4). Because of the input projection, ARD prior should not make much difference to results (this was verified in preliminary tests), and so a model with ARD prior was not used in full tests.

The reference method in the study was iterative inversion of the EIT forward model using total variation regularization (see Vauhkonen, Kaipio, Somersalo & Karjalainen (1997) for further information). In this approach the conductivity distribution is sought to minimize a cost function, that is defined as the squared difference of the measured potentials and the potentials computed from the conductivity distribution by the forward model. The minimization was carried out by Newton's method, requiring about 20 iteration steps. As it was known that the bubbles and the background had constant conductivities, total variation regularization was used. The regularizer penalty function was the total sum of absolute differences between adjacent area elements, forcing the solution to be smoother, but not penalizing abrupt changes (total change in a monotonic curve is equal independent of the steepness, in contrast to, say, squared differences that pull the solution towards low-gradient solutions).

Fig. 8 shows examples of the image reconstruction results. Table 5 shows the quality of the image reconstructions, measured by the error in the void fraction and the percentage of erroneous pixels in the segmentation, over the test set. An important goal in this process tomography application was to estimate the void fraction, which is the proportion of gas and liquid in the image. With the proposed approach, such goal variables can be estimated directly without explicit reconstruction of the image. The last column in Table 5 shows the relative absolute error in estimating the void fraction directly from the projections of the potential signals.

In solving real problems with non-linear learning models, the ability to assess the confidence of the output is necessary. Fig. 9 shows the scatter plot of the void fraction versus the estimate by the Bayesian MLP, together with confidence intervals. The 10 and 90% quantiles are composed directly from the posterior predictive distribution of the model output. When the void fraction is large, the forward model becomes more non-linear (as the current curls around the non-conducting disturbances) and the inverse problem becomes more ill-posed, especially for disturbances far from the electrodes. This ambiguity is clearly visible in the confidence intervals, so that the CIs are wide when the model may make large errors.

Although the model was based on simulated data it has given very promising results in the preliminary experiments with real data.

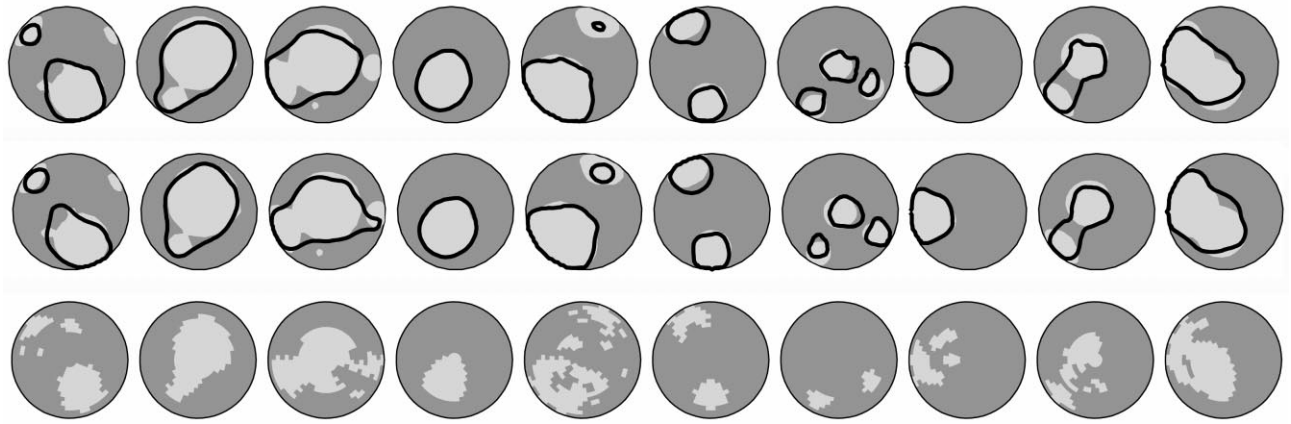

Fig. 8. Example of image reconstruction with MLP ESC (upper row), Bayesian MLP (middle row) and TV inverse (lower row). In the MLP plots the actual bubble is shown by the gray blob and contour of the detected bubble as the black line. For the TV inverse, the estimated bubble is shown as the gray blob, with the same actual bubbles as in the upper images.

6. Case III: classification task in forest scene analysis

In this section, we report results of using the Bayesian MLP for classification of forest scenes. The objective of the project was to assess the accuracy of estimating the volumes of growing trees from digital images. To locate the tree trunks and to initialize the fitting of the trunk contour model, a classification of the image pixels to tree and non-tree classes was necessary. The main problem in the task was the large variance in the classes. The appearance of the tree trunks varies in color and texture due to varying lighting conditions, epiphytes (such as gray or black lichen on white birch), and species dependent variations (such as Scots pine, with bark color ranging from dark brown to orange). In the non-tree class the diversity is much larger, containing e.g. terrain, tree branches and sky. This diversity makes it difficult to choose the optimal features for the classification.

In the work reviewed here (Vehtari et al., 1998), we used a large number of potentially useful features. There was a total of 84 features: 36 Gabor filters (six orientations × six frequencies) that are generic features related to shape and texture, and 46 common statistical features such as texture filters and color histogram features. Due to the large number of features, many classifier methods suffer from the curse of dimensionality. The results of this case demonstrate that the Bayesian MLP is very competitive in this high dimensional problem.

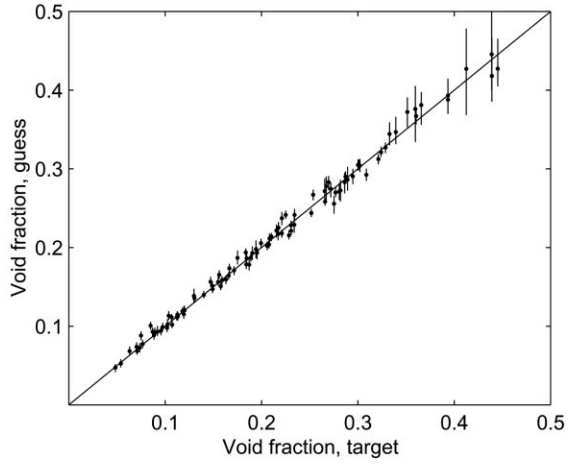

Fig. 9. Scatterplot of the void fraction estimate with 10 and 90% quantiles.

A total of 48 images were collected by using an ordinary digital camera in varying weather conditions. The labeling of the image data was done by hand via identifying many types of tree and background image blocks with different textures and lighting conditions. In this study, only pines were considered.

To estimate classification errors of different methods we used an eight-fold cross-validation error estimate, i.e. 42 of 48 pictures were used for training and the six left out for error evaluation, and this scheme was repeated eight times.

The MLP models contained 20 hidden units, and logistic output layer. The other tested models were:

- KNN LOOCV, K-nearest-neighbor classification, where K is chosen by leave-one-out cross-validation,[5] and
- CART, Classification and Regression Tree (Breiman, Friedman, Olshen & Stone, 1984)

CV error estimates are collected in Table 6. The differences are not very significant, partly due to having only eight-fold CV, but mostly because the different images had very different error rates. This causes extra variance to the classification results during the CV, which reduces the significance of the differences, even though the variance comes from the variations in the task, not variations of the methods. All the MLP models clearly win over the other methods, while the

[5] URL:http://www.cs.utoronto.ca/~delve/methods/knn-class-1/home.html

Table 5
Errors in reconstructing the bubble shape and estimating the void fraction from the reconstructed images. See text for explanation of the models

Method	Classification error	Relative error in VF (%)	Relative error in direct VF (%)
TV-inverse	9.7	22.8	–
MLP ESC	6.7	8.7	3.8
Bayesian MLP	5.9	8.1	3.4

Table 6
CV error estimates for forest scene classification. See text for explanation of the different models

Reference method		Classification error (%)	std	Comparison				
				1	2	3	4	5
1	CART	30	2		1	1	1	1
2	KNN LOOCV	20	2	–		1	1	1
3	MLP ESC	13	1	–	–		37	27
4	Bayesian MLP	12	1	–	–	–		25
5	Bayesian MLP + ARD	11	1	–	–	–	–	

best method, Bayesian MLP with ARD, is just slightly better than the other MLP models ($p < 0.27$).

Fig. 10 shows an example image classified with different methods. Visually, the Bayesian MLP with ARD gives less spurious detections than the other methods. The ARD reduces the effect of features weakly correlating with the classification, and thus larger windows and robust features dominate. On the other hand this causes the classifier to miss some thin trunks and parts of trunks that are not clearly visible.

7. Discussion and conclusions

We have reviewed the Bayesian approach for neural networks, concentrating on the MLP model and MCMC approximation for computing the marginal distribution of the end variables of the study from the joint posterior of all unknown variables given the data. In three real applications, we have assessed the performance of the Bayesian approach and compared it to other methods.

The most important advantage of the Bayesian approach in the case studies was the possibility to handle the situation where some of the prior knowledge is lacking or vague, so that one is not forced to guess values for attributes that are unknown. The best Bayesian models were mostly those with least restrictive hierarchical priors, so that even though the Bayesian approach is based on inherently subjective selection of prior probabilities, the final Bayesian models were much less subjective than the corresponding classical (error minimization) methods. For example, we did not have to (subjectively) guess in advance the number of degrees of freedom in the models, the distribution of the model residuals, or the degree of complexity (non-linearity) of the model with respect to each input variable, which are pre-fixed in ML models. There are ways to handle the selection of total model complexity, to some degree, in the classical approach (such as the ESC of MLP models), but the use of hierarchical models to handle the more difficult issues is characteristic of the Bayesian approach.

An important issue in Bayesian modeling is the sensitivity of the results to the prior assumptions. The goodness of the model is measured by the probability of the model given the data and the prior assumptions, so that incorrect assumptions yield wrong models (with respect to reality) having large probabilities. The Bayesian approach is more sensitive to such bad guesses than the classical approach, if in the latter the choice of the best method is based on partly different assumptions than the choice of the model parameters (e.g. CV of error-minimizing models). In practical applications, the Bayesian approach usually requires more expert work than the standard approach, either to devise reasonable assumptions for the distributions, or to include different options in the models and integrate over them, which is the 'correct' Bayesian approach.

Earlier published studies with conclusions comparable to those of this paper include MacKay (1994), Neal (1996), Thodberg (1996), Rasmussen (1996), Vivarelli and Williams (1997), Husmeier, Penny, and Roberts (1998), Neal (1998) and Penny and Roberts (1999). Cited studies include both regression and classification cases. In classification, the likelihood model (usually) contains no hyperparameters, whereas in regression problems, the noise model is a crucial part of the solution. In the cited studies, the noise model was normal or *t*-distribution with fixed shape. To the authors' knowledge, comparable studies regarding the noise models in Bayesian MLPs has not been published earlier. In other Bayesian models, these issues have been discussed in, e.g. Gelman et al. (1995) and Spiegelhalter et al. (1996).

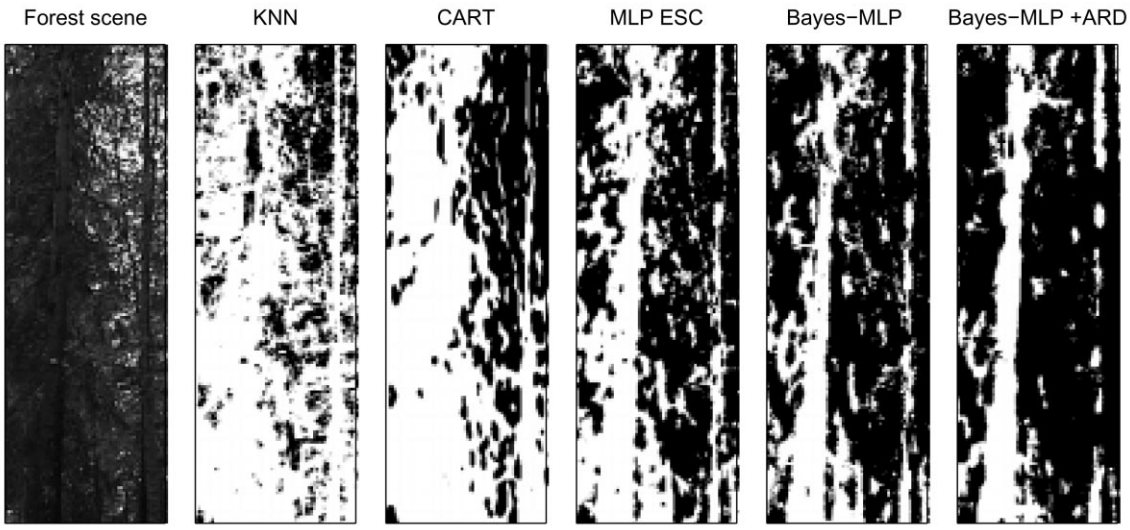

Fig. 10. Examples of a classified forest scene. See text for explanation of the different models.

In MacKay (1994), MLP with ARD and normal noise model was used in the evidence framework in a real regression problem. Instead of using a longer tailed noise model, possible outliers were omitted by hand. Evaluation of the evidence was problematic and results were improved by using a validation set to choose the best models to be used as the committee. Without ARD, errors were significantly increased. This model won the ASHRAE's 1993 prediction competition by a significant margin.

In Thodberg (1996), a slightly modified ARD, where the ranking of the ARD hyperparameters was manually predetermined, performed well in the evidence framework in a real case problem. A normal noise model was used.

In Penny and Roberts (1999), the evidence framework was assessed in artificial and real problems. The performance was comparable to the best alternative methods, which is in accordance with the results in our experiments (the empirical Bayesian approach used should be somewhat inferior to the full Bayesian approach, and in our studies the basic model with Gaussian noise was also similar to or slightly better than the MLP ESC). The other main conclusion was that the ARD is rarely useful, and then only in cases when there are many irrelevant inputs, which is in contradiction to our results. It is quite possible that the hierarchical ARD prior performs considerably better in the full Bayesian approach than in the evidence approach.

In the rest of the cited studies, a similar MCMC method for the full Bayesian approach was used as in this paper.

In Neal (1996), an MLP model with ARD and t_4-distribution noise model performed better than alternative methods in a real regression problem. The performance without ARD or with normal noise model was not compared. In a real classification problem there was no significant benefit of using ARD. Convergence problems were suspected. In artificial problems with irrelevant inputs ARD improved results.

In Rasmussen (1996), MLP and GP models with ARD and a normal noise model performed better than non-Bayesian models (linear model, KNN, MARS, and MLP ESC) in real and artificial regression problems.

In Vivarelli and Williams (1997) the evidence framework was compared with a MCMC method in a real classification problem. The evidence framework performed less well for smaller training sets, but with full data set performance difference was not significant. The conclusion was that as the number of data points gets smaller than the number of weights in the MLP, the Gaussian approximation used in the evidence framework is not good. In this case problem, use of ARD prior did not change classification results significantly.

Several benchmark problems were studied in Neal (1998). All the problems were regression tasks, using normal noise models. The main conclusion was that no proper comparison to the alternative method (MLP ESC) could be made, because (i) apparent lack of convergence of some runs of MCMC, and (ii) a serious flaw in the noise model for one of the benchmark tasks. In cases where the MCMC had probably converged and the noise model was correct, the Bayesian approach performed better than MLP ESC, and the model with ARD prior generally performed slightly better than the corresponding model without ARD.

In Husmeier et al. (1998), several classification tasks were studied. The main conclusions were that: (i) the results were equal to or better than those with alternative (evidence or non-Bayesian) methods; (ii) the results were not sensitive to the exact values of the highest level hyperprior parameters; and (iii) the performance of ARD was controversial, as ARD improved the results in three cases and considerably deteriorated the results in two cases. As a possible reason for this, the authors note that the convergence of MCMC chains with ARD is slower than without, and the results in the paper may be from non-equilibrium states. This is in accordance with our results. In our work, we used MCMC diagnostics to

ensure convergence, and ARD never gave worse results, but, indeed, increased the burn-in time.

To summarize the conclusions of this paper, the Bayesian approach for MLP networks, using MCMC approximation for the marginalization, gave better results than alternative non-Bayesian methods in all the case problems. In general, the best models were those with the least informative priors, underlining the importance of explicitly specifying the lack of knowledge, instead of guessing the values for attributes that are unknown.

It must be emphasized that the results of data analysis depend on the assumptions and approximations made—thus the Bayesian approach does not automatically give better results than a classical approach. Even though the Bayesian models do not need validation data to set the model complexity, validation of the final model is essential, as with any other modeling approach.

Acknowledgements

This study was partly funded by TEKES Grant 40888/97 (Project PROMISE, Applications of Probabilistic Modeling and Search) and Graduate School in Electronics, Telecommunications and Automation (GETA). The authors would like to thank H. Järvenpää for providing her expertise into the case study I, and K. Leinonen and J. Kaipio for aiding in the problem setup and providing the TV inverse method in the case study II.

References

Barber, D., & Bishop, C. M. (1998). Ensemble learning in Bayesian neural networks. In C. M. Bishop, *Neural networks and machine learning*, *volume 168 of NATO ASI Series F: computer and systems sciences*, (pp. 215–237). Springer-Verlag.

Berger, J. O. (1985). *Statistical design theory and Bayesian analysis, Springer series in statistics*. (2nd ed.). Springer.

Berger, J. O., & Bernardo, J. M. (1992). On the development of reference priors. In J. M. Bernardo, J. O. Berger, A. P. Dawid & A. F. M. Smith, *Bayesian statistics 4* (pp. 35–60). Oxford University Press.

Bernardo, J. M., & Smith, A. F. M. (1994). *Bayesian theory*, John Wiley & Sons.

Bishop, C. M. (1993). Curvature-driven smoothing: a learning algorithm for feed-forward networks. *IEEE Transactions on Neural Networks*, *4* (5), 882–884.

Bishop, C. M. (1995). *Neural networks for pattern recognition*, Oxford University Press.

Bishop, C. M., & Qazaz, C. S. (1997). Regression with input-dependent noise: a Bayesian treatment. In M. C. Mozer, M. I. Jordan & T. Petsche, *Advances in neural information processing systems 9* (pp. 347–353). MIT Press.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and regression trees*, Chapman & Hall.

Brooks, S. P., & Roberts, G. O. (1999). Assessing convergence of Markov chain Monte Carlo algorithms. *Statistics and Computing*, *8* (4), 319–335.

Buntine, W. L., & Weigend, A. S. (1991). Bayesian back-propagation. *Complex Systems*, *5* (6), 603–643.

de Freitas, J. F. G., Niranjan, M., Gee, A. H., & Doucet, A. (2000). Sequential Monte Carlo methods to train neural network models. *Neural Computation*, *12* (4), 955–993.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, *10* (7), 1895–1924.

Duane, S., Kennedy, A. D., Pendleton, B. J., & Roweth, D. (1987). Hybrid Monte Carlo. *Physics Letters B*, *195* (2), 216–222.

Geisser, S. (1975). The predictive sample reuse method with applications. *Journal of the American Statistical Association*, *70* (350), 320–328.

Gelfand, A. E. (1996). Model determination using sampling-based methods. In W. R. Gilks, S. Richardson & D. J. Spiegelhalter, *Markov chain Monte Carlo in practice* (pp. 145–162). Chapman & Hall.

Gelman, A. (1996). Inference and monitoring convergence. In W. R. Gilks, S. Richardson & D. J. Spiegelhalter, *Markov chain Monte Carlo in practice* (pp. 131–144). Chapman & Hall.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. R. (1995). *Bayesian data analysis. Texts in statistical science*, Chapman & Hall.

Geman, S., & Geman, D. (1984). *Gibbs distributions and the Bayesian restoration of images*, *IEEE Transactions on Pattern Analysis and Machine Intelligence, 6* (2), 721–741.

Geweke, J. (1993). Bayesian treatment of the independent Student-*t* linear model. *Journal of Applied Econometrics*, *8*, S19–S40.

Gilks, W. R., Richardson, S. & Spiegelhalter, D. J. (1996). Markov chain Monte Carlo in practice Chapman & Hall.

Goel, P. K., & Degroot, M. H. (1981). Information about hyperparameters in hierarchical models. *Journal of the American Statistical Association*, *76* (373), 140–147.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, *57* (1), 97–109.

Husmeier, D., Penny, W. D., & Roberts, S. J. (1998). Empirical evaluation of Bayesian sampling for neural classifiers. In L. Niklasson, M. Boden & T. Ziemke, *ICANN '98: Proceedings of the Eighth International Conference on Artificial Neural Networks*. Springer.

Jeffreys, J. (1961). *Theory of probability*, (3rd ed.). Oxford University Press.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. (1998). An introduction to variational methods for graphical models. In M. I. Jordan, *Learning in graphical models*, *volume 89 of Nato Science Series: D behavioural and social sciences.* Kluwer Academic Publishers.

Kass, R. E., & Raftery, A. E. (1995). Bayes factors. *Journal of the American Statistical Association*, *90* (430), 773–795.

Kass, R. E., & Wasserman, L. (1996). The selection of prior distributions by formal rules. *Journal of the American Statistical Association*, *91* (435), 1343–1370.

Krogh, A., & Vedelsby, J. (1995). Neural network ensembles, cross-validation, and active learning. In G. Tesauro, D. S. Touretzky & T. K. Leen, *Advances in neural information processing systems 7* (pp. 231–238). MIT Press.

Lampinen, J., & Selonen, A. (1997). Using background knowledge in multilayer perceptron learning. In M. Frydrych, J. Parkkinen & A. Visa, *SCIA '97: Proceedings of the Tenth Scandinavian Conference on Image Analysis* (pp. 545–549), vol. 2. Pattern Recognition Society of Finland.

Lampinen, J., Vehtari, A., & Leinonen, K. (1999). Using Bayesian neural network to solve the inverse problem in electrical impedance tomography. In B. K. Ersboll & P. Johansen, *SCIA '99: Proceedings of the 11th Scandinavian Conference on Image Analysis* (pp. 87–93), vol. 1. The Pattern Recognition Society of Denmark.

Lemm, J. C. (1996). Prior information and generalized questions. Technical report AIM 1598, CBCLP 141, Massachusetts Institute of Technology, Artificial Intelligence Laboratory and Center for Biological and Computational Learning, Department of Brain and Cognitive Sciences.

Lemm, J. C. (1999). Bayesian field theory. Technical report MS-TP1-99-1, Universität Münster, Institut für Theoretische Physik.

MacKay, D. J. C. (1992). A practical Bayesian framework for backpropagation networks. *Neural Computation*, *4* (3), 448–472.

MacKay, D. J. C. (1994). Bayesian non-linear modelling for the prediction competition. *ASHRAE Transactions*, *100* (2), 1053–1062.

MacKay, D. J. C. (1995). Probable networks and plausible predictions—a review of practical Bayesian methods for supervised neural networks. *Network: Computation in Neural Systems*, *6* (3), 469–505.

Müller, P., & Rios Insua, D. (1998). Issues in Bayesian analysis of neural network models. *Neural Computation*, *10* (3), 571–592.

Neal, R. M. (1992). Bayesian training of backpropagation networks by the hybrid Monte Carlo method. Technical report CRG-TR-92-1, Department of Computer Science, University of Toronto.

Neal, R. M. (1996). *Bayesian learning for neural networks*, *volume 118 of Lecture notes in statistics*. Springer-Verlag.

Neal, R. M. (1998). Assessing relevance determination methods using DELVE. In C. M. Bishop, *Neural networks and machine learningvolume 168 of NATO ASI Series F: computer and systems sciences* (pp. 97–129). Springer-Verlag.

Neal, R. M. (1999). Regression and classification using Gaussian process priors (with discussion). In J. M. Bernardo, J. O. Berger, A. P. Dawid & A. F. Smith, *Bayesian statistics 6* (pp. 475–501). Oxford University Press.

Penny, W. D., & Roberts, S. J. (1999). Bayesian neural networks for classification: how useful is the evidence framework? *Neural Networks*, *12* (6), 877–892.

Rasmussen, C. E. (1996). *Evaluation of Gaussian processes and other methods for non-linear regression*. PhD thesis, Department of Computer Science, University of Toronto.

Roberts, C. P., & Casella, G (1999). *Monte Carlo statistical methods*, *Springer texts in statistics*. Springer-Verlag.

Sarle, W. S. (1997). How to measure importance of inputs? [online]. Technical report, SAS Institute Inc., Cary, NC, USA. Revised 23 June 2000. Available at: ftp://ftp.sas.com/pub/neural/importance.html

Spiegelhalter, D., Thomas, A., Best, N., & Gilks, W. (1996). *BUGS 0.5 * Examples Volume 1 (version i)*, MRC Biostatistics Unit, Institute of Public Health.

Thodberg, H. H. (1996). A review of Bayesian neural networks with an application to near infrared spectroscopy. *IEEE Transactions on Neural Networks*, *7* (1), 56–72.

Vauhkonen, M., Kaipio, J. P., Somersalo, E., & Karjalainen, P. A. (1997). Electrical impedance tomography with basis constraints. *Inverse Problems*, *13* (2), 523–530.

Vehtari, A. & Lampinen, J. (1999). *Bayesian neural networks with correlating residuals*. In IJCNN '99: Proceedings of the 1999 International Joint Conference on Neural Networks [CD-ROM], number 2061. IEEE.

Vehtari, A. & Lampinen, J. (2000). On Bayesian model assessment and choice using cross-validation predictive densities. Technical report B23, Laboratory of Computational Engineering, Helsinki University of Technology.

Vehtari, A., Heikkonen, J., Lampinen, J. & Juujärvi, J. (1998). *Using Bayesian neural networks to classify forest scenes*. In D. P. Casasent (Ed.), Intelligent robots and computer vision XVII: algorithms, techniques, and active vision, volume 3522 of Proceedings of SPIE (pp. 66-73). SPIE.

Vehtari, A., Särkkä, S. & Lampinen, J. (2000). *On MCMC sampling in Bayesian MLP neural networks*. In S.-I. Amari, C. L. Giles, M. Gori & V. Piuri (Eds.), IJCNN '2000: Proceedings of the 2000 International Joint Conference on Neural Networks, volume 1 (pp. 317-322). IEEE.

Vivarelli, F. & Williams, C. K. I. (1997). *Using Bayesian neural networks to classify segmented images*. In Proceedings of the IEE Fifth International Conference on Artificial Neural Networks, number 40 in Conference Publications (pp. 268-263). The Institution of Electrical Engineers.

Winther, O. (1998). *Bayesian mean field algorithms for neural networks and Gaussian processes*. PhD thesis, University of Copenhagen.

Wolpert, D. H. (1996a). The existence of a priori distinctions between learning algorithms. *Neural Computation*, *8* (7), 1391–1420.

Wolpert, D. H. (1996b). The lack of a priori distinction between learning algorithms. *Neural Computation*, *8* (7), 1341–1390.

Wolpert, D. H. & Macready, W. G. (1995). No free lunch theorems for search. Technical report SFI-TR-95-02-010. The Santa Fe Institute.

Yang, R. & Berger, J. O. (1997). A catalog of noninformative priors. ISDS Discussion paper 97-42, Institute of Statistics and Decision Sciences, Duke University.