

Bayesian Techniques for Neural Networks - Review and Case Studies

Jouko Lampinen and Aki Vehtari

Laboratory of Computational Engineering, Helsinki University of Technology

P.O.Box 9400, FIN-02015 HUT, Espoo, Finland

E-mail: {Jouko.Lampinen,Aki.Vehtari}@hut.fi

ABSTRACT

We give a short review on Bayesian techniques for neural networks and demonstrate the advantages of the approach in a number of industrial applications. Bayesian approach provides a principled way to handle the problem of overfitting, by averaging over all model complexities weighted by their posterior probability given the data sample. The approach also facilitates estimation of the confidence intervals of the results, and comparison to other model selection techniques (such as the committee of early stopped networks) often reveals faulty assumptions in the models. In this contribution we review the Bayesian techniques for neural networks and present comparison results from several case studies that include regression, classification, and inverse problems.

1 INTRODUCTION

In non-linear function approximation and classification, neural networks have become popular tools in recent years. With neural networks the main difficulty is in 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. 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.).

Recently, Bayesian methods have become a viable alternative to the older error minimization based ML (Maximum Likelihood) or MAP (Maximum A Posteriori) approaches [11, 13, 2]. The main advantages of Bayesian multilayer perceptron models are:

- Automatic complexity control: the values of the regularization coefficients can be selected using only the training data, without the need to use separate training and validation data.

- Possibility to use prior information and hierarchical models for the hyperparameters.
- Predictive distributions for outputs.

In this contribution we demonstrate the advantages of Bayesian MLPs in three case problems. In sections 2 and 3 we give a review of the Bayesian methods for MLP networks. Then we report results on using Bayesian MLP models in regression problem (section 4), tomographic image reconstruction problem (section 5) and classification problem (section 6), and compare the approach to standard neural network and other statistical methods.

2 BAYESIAN APPROACH

The key principle of Bayesian approach is to construct the posterior probability distributions for all the unknown entities in the models. 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 quantities, as the posterior for a parameter θ given the data D is according to the Bayes' rule, $p(\theta|D) = p(D|\theta)p(\theta)/p(D)$, where $p(D|\theta)$ is the likelihood of the parameters θ and $p(\theta)$ the prior probability of θ .

The use of the explicit prior information distinguishes the Bayesian approach from the maximum likelihood methods. It is worth noticing that every discrete choice in the model, such as the Gaussian noise model, represents infinite amount of prior information [10]. 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. Thus there is large amount of prior information also in the maximum likelihood models (actually it is what separates "good" and "bad" ML models), even though the model parameters are determined solely by the data, to maximize the likelihood $p(D|w)$. In the Bayesian approach there are explicit prior distributions for the model parameters, but as

discussed above, large part of the prior information is still implicit in the form of the choices made in the model.

The marginalization principle leads to complex integrals that cannot be solved in closed form, and thus there are multitude of approaches that differ in the degree of "Bayesianism", that is, how thoroughly this principle is followed.

Closest to the ML approach is the Maximum A Posteriori approach, where the posterior distribution 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. The main drawback of this approach is that it gives no tools for setting the hyperparameters (smoothness coefficients, or model complexity), due to lack of marginalization over these "nuisance parameters". For example, with the Gaussian prior on w , $p(w) \propto \exp(-\alpha w^2)$, the variance term α must be set with some external procedure, such as cross-validation.

A further degree of Bayesian principle is utilized in the evidence framework [11], or type II ML approach, where specific values are estimated for the hyperparameters, so that the marginal probability for the hyperparameters, integrated over the parameters, $p(\alpha|D) = \int p(\alpha, w|D)dw$, is maximized. Gaussian approximation is used for the posterior of the parameters, to facilitate closed form marginalization, and thus the resulting posterior is specified by the mean of the Gaussian approximation.

In a full Bayesian approach no fixed values are estimated for any parameters or hyperparameters. If the model is used for prediction, the marginalization is done over the parameters also, as shown in Eq. 4. The priors are then constructed hierarchically, so that the hyperparameters have hyperpriors, and the parameters of those distributions next level priors and so on. See, e.g., [5] for good introduction to these methods.

Note again, that in such models there are large amounts of fixed prior knowledge, that is based on uncertain assumptions. So, conceptually, in full hierarchical Bayesian model, no guesses are made for any 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 necessitates using some sort of model validation techniques. This also implies that in practice the Bayesian approach may be more sensitive to the prior assumptions than more classical methods. This is discussed in more detail in chapter 3.7.

3 BAYESIAN LEARNING FOR MLP NETWORKS

3.1 MLP and model selection

We concentrate here to one hidden layer MLPs with hyperbolic tangent (tanh) activation function. However, the

Bayesian methods can be used for other types of neural networks, like RBF networks, too. Basic MLP model with k outputs is

$$f_k(\mathbf{x}, \mathbf{w}) = w_{k0} + \sum_{j=1}^m w_{kj} \tanh \left(w_{j0} + \sum_{i=1}^d w_{ji} x_i \right), \quad (1)$$

where \mathbf{x} is a d -dimensional input vector, \mathbf{w} denotes the weights, and indices i and j correspond to input and hidden units, respectively.

MLP is often considered as a generic semiparametric model, in a sense that there is a large number of parameters without any physical meaning, as in non-parametric models, but the actual model may have very low effective complexity, depending on the complexity of the data, resembling in this respect parametric models with modest number of parameters.

Traditionally the complexity of the MLP has been controlled with early stopping or weight decay [2]. In early stopping weights are initialized to small values, so that the sigmoidal hidden units operate on the linear regions and the initial mapping is smooth. Part of the training data is used to train the MLP and the other part is used to monitor the validation error. In the iterative error minimization the training is stopped when the validation error begins to increase, so that the effective complexity may be much less than the number of parameters in the network.

The basic early stopping is rather inefficient, as it is very sensitive to the initial conditions of the weights and only part of the available data is 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. When used with caution early stopping committee is a good baseline method for MLPs.

In weight decay the weights are encouraged to be small by a penalty function, that corresponds to Gaussian prior on the weights, leading to MAP estimate for the model. In practice each layer in the MLP should have different regularization parameter [2], giving the penalty term

$$\alpha_1 \sum_{j,i} w_{ji}^2 + \alpha_2 \sum_{j,k} w_{kj}^2. \quad (2)$$

Problem is how to select good values for α_i . Traditionally this has been done with cross validation (CV). Since CV gives noisy estimate for error, it does not guarantee that good values for α_i can be found. Also it easily becomes computationally prohibitive as computational expenses grow exponentially with number of parameters to be selected.

3.2 Bayesian learning

Consider a regression or classification problem involving the prediction of a noisy vector \mathbf{y} of target variables given the value of a vector \mathbf{x} of input variables.

The process of Bayesian learning is started by defining a model, \mathcal{M} , and prior distribution $p(\theta)$ for the model parameters θ . After observing new data $D = \{(\mathbf{x}^{(1)}, \mathbf{y}^{(1)}), \dots, (\mathbf{x}^{(n)}, \mathbf{y}^{(n)})\}$, prior distribution is updated to the posterior distribution using Bayes' rule

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)} \propto L(\theta|D)p(\theta), \quad (3)$$

where the likelihood function $L(\theta|D)$ gives the probability of the observed data as function of the unknown model parameters.

To predict the new output $\mathbf{y}^{(n+1)}$ for the new input $\mathbf{x}^{(n+1)}$, predictive distribution is obtained by integrating the predictions of the model with respect to the posterior distribution of the model parameters

$$p(\mathbf{y}^{(n+1)}|\mathbf{x}^{(n+1)}, D) = \int p(\mathbf{y}^{(n+1)}|\mathbf{x}^{(n+1)}, \theta)p(\theta|D)d\theta. \quad (4)$$

This is the same as taking the average prediction of all the models weighted by their posterior probability.

3.3 Likelihood models

Statistical model is defined with its likelihood function. If we assume that the n data points $(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})$ are exchangeable we get

$$L(\theta|D) = \prod_{i=1}^n p(\mathbf{y}^{(i)}|\mathbf{x}^{(i)}, \theta). \quad (5)$$

The term $p(\mathbf{y}^{(i)}|\mathbf{x}^{(i)}, \theta)$ in Eq. (5) depends on our problem. In regression problems, it is generally assumed that the distribution of the target data can be described by a deterministic function of inputs, corrupted by additive Gaussian noise of a constant variance. Probability density for a target y_j is then

$$p(y_j|\mathbf{x}, \mathbf{w}, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y_j - f_j(\mathbf{x}, \mathbf{w}))^2}{2\sigma^2}\right), \quad (6)$$

where σ^2 is the noise variance for the target. See [13] for per-case normal noise variance model and [17] for full covariance model assuming correlating residuals. For a two class classification (logistic regression) model, the probability that a binary-valued target, y_j , has the value 1 is

$$p(y_j = 1|\mathbf{x}, \mathbf{w}) = [1 + \exp(-f_j(\mathbf{x}, \mathbf{w}))]^{-1} \quad (7)$$

and for a many class classification (softmax) model, the probability that a class target, y , has value j is

$$p(y = j|\mathbf{x}, \mathbf{w}) = \frac{\exp(f_j(\mathbf{x}, \mathbf{w}))}{\sum_k \exp(f_k(\mathbf{x}, \mathbf{w}))}. \quad (8)$$

In Eqs. (6), (7) and (8) the function $f(\mathbf{x}, \mathbf{w})$ is in this case the MLP network.

Using classical estimation (error minimization) for the MLP the number of free parameters (weights) in the model

need to be adjusted according to the size of the training set, the complexity of the target function and the amount of noise. In Bayesian approach there is no need to restrict the size of the network, but in practice we use modest number of hidden units for computational reasons. In the limit of infinite number of hidden units the MLP converges to the Gaussian process [13], which is, at least for small sample size, a very viable alternative method.

3.4 Priors

Typical priors in Bayesian function approximation are smoothing priors, that state, for example, that functions with small second derivative values are more probable. With MLP these lead to a rather complex treatise [8], [1]. As discussed in section 3.1, complexity of the MLP can be controlled, on coarse level, by controlling the size of the weights \mathbf{w} . This can be achieved by, e.g., Gaussian prior distribution for weights \mathbf{w} given hyperparameter α

$$p(\mathbf{w}|\alpha) = (2\pi)^{-m/2} \alpha^{m/2} \exp(-\alpha \sum_{i=1}^m w_i^2/2). \quad (9)$$

The coarse level of complexity is determined by the hyperparameter α , and since we have no specific knowledge of the right value, we set a vague hyperprior $p(\alpha)$, that merely makes very high and very low values for α improbable. A convenient form for this hyperprior is vague Gamma distribution with mean μ and shape parameter a

$$p(\alpha) \sim \text{Gamma}(\mu, a) \propto \alpha^{a/2-1} \exp(-\alpha a/2\mu). \quad (10)$$

In order to have a prior for the weights which is invariant under the linear transformations of data, separate priors (each having its own hyperparameters α_i) for different weight groups in each layer of a MLP are used [13].

Often very useful prior is called Automatic Relevance Determination (ARD) [12, 13, 14]. In the ARD the input-to-hidden weights connected to the same input have common prior variance, and all the variances have common prior distribution (hyperprior). This allows the posterior values for the priors to adjust so that irrelevant inputs have tighter priors and thus those weights are more efficiently driven towards zero than with a common prior for all the inputs.

For regression models we need also prior for the noise variance σ in Eq. (6), which is often specified in terms of corresponding precision, $\tau = \sigma^{-2}$. As for α , our prior information is usually quite vague, stating that noise variance σ is not zero nor extremely large. This prior can be expressed with vague Gamma-distribution with mean μ and shape parameter a

$$p(\tau) \sim \text{Gamma}(\mu, a) \propto \tau^{a/2-1} \exp(-\tau a/2\mu). \quad (11)$$

3.5 Prediction

Predictive distribution for new data is obtained by marginalizing (integrating) over the posterior distribution of the pa-

rameters and hyperparameters

$$p(\mathbf{y}^{(n+1)}|\mathbf{x}^{(n+1)}, D) = \int p(\mathbf{y}^{(n+1)}|\mathbf{x}^{(n+1)}, \mathbf{w}, \alpha, \tau) p(\mathbf{w}, \alpha, \tau|D) d\mathbf{w} \alpha \tau. \quad (12)$$

We can also evaluate expectations of various functions with respect to the posterior distribution for parameters. For example in regression we may evaluate the expectation for a component of $\mathbf{y}^{(n+1)}$

$$\hat{y}_k^{(n+1)} = \int f_k(\mathbf{x}^{(n+1)}, \mathbf{w}) p(\mathbf{w}, \alpha, \tau|D) d\mathbf{w} \alpha \tau, \quad (13)$$

which corresponds to the best guess with squared error loss.

The posterior distribution for the parameters $p(\mathbf{w}, \alpha, \tau|D)$ is typically very complex, with many modes. Evaluating the integral of Eq. (13) is therefore a difficult task. The integral can be approximated with parametric approximation as in [11] or with numerical approximation as described in next section.

3.6 Markov Chain Monte Carlo method

Neal has introduced an implementation of Bayesian learning for MLPs in which the difficult integration of Eq. (13) is performed using Markov Chain Monte Carlo (MCMC) methods [13]. In [7] there is a good introduction to basic MCMC methods and many applications in statistical data analysis.

The integral of Eq. (13) is the expectation of function $f_k(\mathbf{x}^{(n+1)}, \mathbf{w})$ with respect to the posterior distribution of the parameters. This and other expectations can be approximated by Monte Carlo method, using a sample of values $\mathbf{w}^{(t)}$ drawn from the posterior distribution of parameters

$$\hat{y}_k^{(n+1)} \approx \frac{1}{N} \sum_{t=1}^N f_k(\mathbf{x}^{(n+1)}, \mathbf{w}^{(t)}). \quad (14)$$

Note that samples from the posterior distribution are drawn during the “learning phase” and predictions for new data can be calculated quickly using the same samples and Eq. (14).

In the MCMC, samples are generated using a Markov chain that has the desired posterior distribution as its stationary distribution. Difficult part is to create Markov chain which converges rapidly and in which states visited after convergence are not highly dependent.

Neal has used the hybrid Monte Carlo (HMC) algorithm [4] for parameters and Gibbs sampling [6] for hyperparameters. 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. Use of Gibbs sampling for hyperparameters helps to minimize the amount of tuning that is needed to obtain good performance in HMC.

When the amount of data increases, the evidence from the data causes the probability mass to concentrate to the smaller area and we need less samples from the posterior distribution.

Also less samples are needed to evaluate the mean of the predictive distribution than the tail-quantiles like, 10% and 90% quantiles. So depending on the problem 10–200 samples may be enough for practical purposes. Note that due to autocorrelations in the Markov chain, getting some 100 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 standard workstation.

In our examples (sections 5, 6) we have used Flexible Bayesian Modeling (FBM) software¹, which implements the methods described in [13].

3.7 Some modelling issues

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 cross-validation that is based on fewer assumptions. 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 modelling with simple classical methods (like linear models, early-stopped committees of MLPs, etc.). If the Bayesian model gives inferior results (measured from test set or cross-validated), some of the assumptions are questionable.

The following computer simulation elucidates the sensitivity of the Bayesian approach to the correctness of the noise model, compared to the early-stopped committee (ESC). The target function and data are shown in Fig. 1. The modelling test was repeated 100 times with different realizations of Gaussian or Laplacian (double exponential) noise. The model was 1 – 10 – 1 MLP with Gaussian noise model. The figure shows one sample of noise and 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 as the Bayesian MLP treatment. The summary of the experiment is shown in Table 1. Using classical t-test, the ESC is significantly better than the Bayesian model when the noise model is wrong. The Wilcoxon signed rank test also indicated that ESC is better than Bayesian MLP (comparing medians) for Laplacian noise with P-value 0.04. In this simple problem, the both methods are equal for the correct noise model.

The implication of this phenomenon in practical applications is, that Bayesian approach usually requires more expert

¹<URL: <http://www.cs.toronto.edu/~radford/fbm.software.html>>

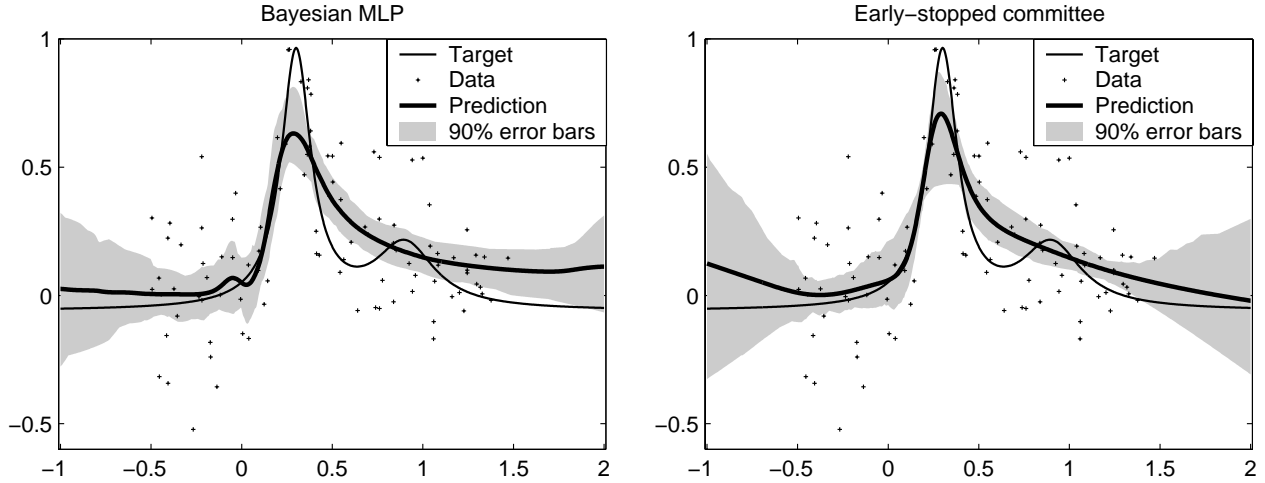


Figure 1: Test function in demonstrating the sensitivity of Bayesian MLP and Early-stopped committee to the wrong noise model. The figure shows one sample of noise realization and the resulting predictions, with Bayesian MLP in left and ESC in right figure. See text for explanation for the error bars.

Table 1: Demonstration of the sensitivity of Bayesian MLP and 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	ESC	Significance of the difference
Gaussian	0.278	0.278	0.43
Laplacian	0.283	0.277	0.006

work than the standard approach, to device the reasonable assumptions for the distributions, but that done, the results are in our experience consistently better than with other approaches.

4 CASE I: REGRESSION TASK IN QUALITY ESTIMATION

In this section we report results of using Bayesian MLPs for regression in concrete quality estimation problem. The goal of the project was to develop a model for predicting the quality properties of concrete. The quality variables included, e.g., compression strengths and densities for 1, 28 and 91 days after casting, bleeding (water extraction) and slump that measure properties of the 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 7 target variables and 19 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. Thus available samples must be used as efficiently as possible, which makes Bayesian techniques a tempting alternative, as they allow fine balance of prior assumptions and evidence from samples. In the study we had 149 samples designed to cover the practical range of the variables, collected by a concrete manufacturing company.

MLP networks containing 6 hidden units were used. Different MLP models tested were:

MLP ESC : Early stopping committee of 20 MLP networks, with different division of data to 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.

Bayes MLP : Bayesian MLP with FBM-software, using t -distribution with 4 degrees of freedom as the noise model, vague priors and MCMC-run specifications similar as used in [13, 14]. 20 networks from the posterior distribution of the network parameters were used.

Bayes MLP +ARD: Similar Bayesian MLP to the previous, but using also the ARD prior.

Error estimates for predicting the slump are collected in Table 2. Results were insensitive to the exact values of the higher level hyperparameter specifications as long as the priors were vague, but the use of the structural ARD prior improved the results significantly.

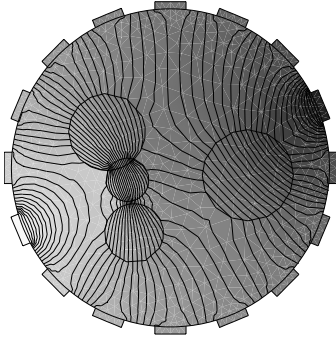


Figure 2: Example of the EIT measurement. The simulated bubble formation is bounded by the circles. The current is injected from the electrode with the lightest color and the opposite electrode is grounded. The gray level and the contour curves show the resulting potential field.

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 [9].

The aim in EIT is to recover the internal structure of an object from surface measurements. Number of electrodes are attached to the surface of the object and current patterns are injected from 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 [15].

Fig. 2 shows a simulated example of the EIT problem. The volume bounded by the circles in the image represent 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. 3 shows the resulting potential signals, from which the image is to be recovered.

In [9] we proposed a novel feedforward 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

Table 2: Ten fold cross-validation error estimates for predicting the slump of concrete.

Method	Root mean square error
MLP ESC	37
Bayes MLP	34
Bayes MLP +ARD	27

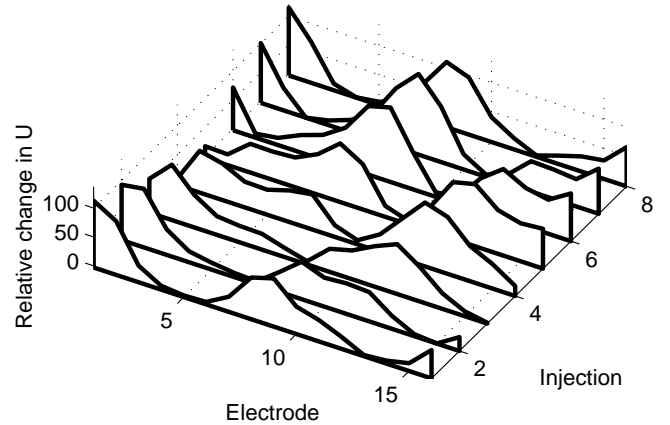


Figure 3: Relative changes in potentials compared to homogeneous background. The eight curves correspond to injections from eight different electrodes.

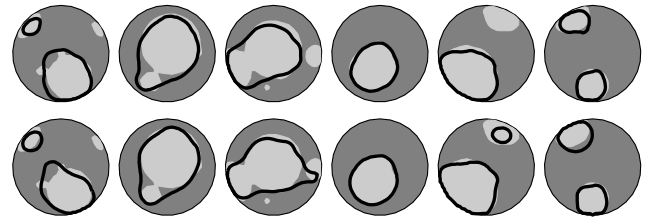


Figure 4: Example of image reconstructions with MLP ESC (upper row) and the Bayesian MLP (lower row)

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 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 *Bayes MLP* (see section 4). Because of the input projection, ARD prior should not make much difference in results (this was verified in preliminary tests), and so model with ARD prior was not used in full tests. We also compared results to *TV-inverse* method, which is a state-of-the-art inverse method based on iterative inversion of the forward model with total variation regularization.

Fig. 4 shows examples of the image reconstruction results. Table 3 shows the quality of the image reconstructions with

Table 3: 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 %	Rel. error in direct VF %
TV-inverse	9.7	22.8	-
MLP ESC	6.7	8.7	3.8
Bayes MLP	5.9	8.1	3.4

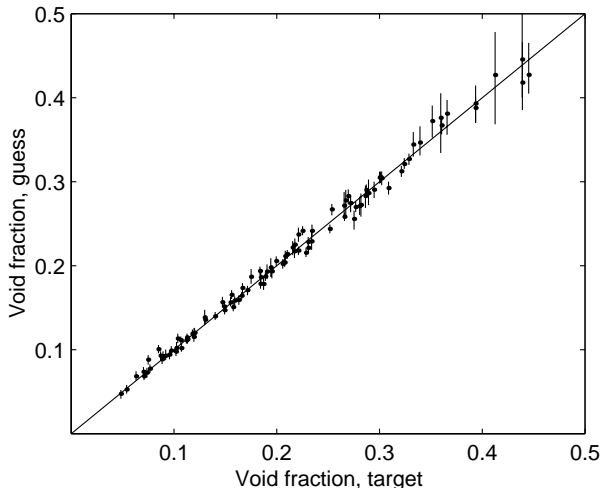


Figure 5: Scatterplot of the void fraction estimate with 10% and 90% quantiles.

models, measured by error in the void fraction and 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 3 shows the relative absolute error in estimating the void fraction directly from the projections of the potential signals.

With Bayesian methods we can easily calculate confidence intervals for outputs. Fig. 5 shows the scatter plot of the void fraction versus the estimate by the Bayesian MLP. The 10% and 90% quantiles are computed directly from the posterior distribution of the model output. See [9] for results for effect of additive Gaussian noise to the performance of the method.

6 CASE III: CLASSIFICATION TASK IN FOREST SCENE ANALYSIS

In this section we report results of using Bayesian MLP for classification of forest scenes, to accurately recognize and locate the trees from any background.

Table 4: CV error estimates for forest scene classification. See text for explanation of the different models.

	Error%
KNN LOOCV	20
CART	30
MLP ESC	13
Bayes MLP	12
Bayes MLP +ARD	11

Forest scene classification task is demanding due to the texture richness of the trees, occlusions of the objects and diverse lighting conditions under operation. This makes it difficult to determine which are optimal image features for the classification. One way to proceed is to extract many different types of potentially suitable features.

In [16] we extracted total of 84 statistical and Gabor-filter based features over different size windows at each spectral channel. Due to the large number of features, many classifier methods would suffer from the curse of dimensionality, but the Bayesian MLP managed well in the high dimensional problem.

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 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. In addition to 20 hidden unit MLP models *MLP ESC* and *Bayesian MLP* (see section 5) the models tested were:

KNN LOOCV : K-nearest-neighbor, where K is chosen by leave-one-out cross-validation.

CART : Classification And Regression Tree [3].

Bayesian MLP +ARD : Same as *Bayesian MLP* plus using Automatic Relevance Determination prior.

CV error estimates are collected in Table 4. Fig. 6 shows example image classified with different methods.

7 SUMMARY

The reviewed case problems in real applications illustrate the advantages of Bayesian MLPs. The approach contains automatic complexity control, as in the Bayesian inference all the results are conditioned on the individual training sample available. Thus the complexity is matched to the support that the training data carries for the models. In addition, the Bayesian approach gives the predictive distributions for the outputs, which can be used to estimate the reliability of the

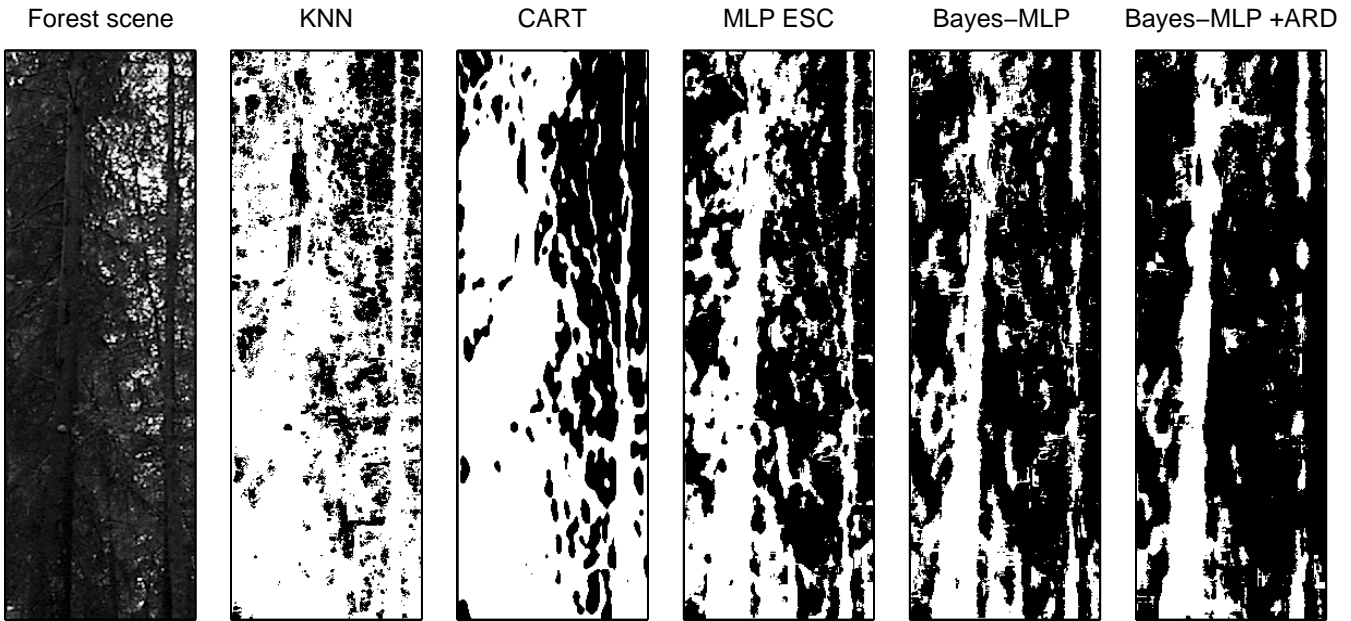


Figure 6: Examples of classified forest scene. See text for explanation of the different models.

predictions. The major problems in the Bayesian MLPs are related to the large amount of computing time needed in the sampling, and in some cases to the need for more accurate assumptions in the likelihood model than in the more traditional methods.

Acknowledgments

This study was partly funded by TEKES Grant 40888/97 (Project *PROMISE, Applications of Probabilistic Modeling and Search*).

References

- [1] Christopher M. Bishop. Curvature-driven smoothing: A learning algorithm for feed-forward networks. *IEEE Transactions on Neural Networks*, 4(5):882–884, Sep 1993.
- [2] Christopher M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.
- [3] Leo Breiman, Jerome Friedman, Richard Olshen, and Charles Stone. *Classification and regression trees*. Chapman & Hall, 1984.
- [4] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth. Hybrid Monte Carlo. *Physics Letters B*, 195:216–222, 1987.
- [5] Andrew Gelman, John B. Carlin, Hal S. Stern, and Donald R. Rubin. *Bayesian Data Analysis*. Chapman & Hall, 1995.
- [6] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.
- [7] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, editors. *Markov Chain Monte Carlo in Practice*. Chapman & Hall, 1996.
- [8] Jouko Lampinen and Arto Selonen. Using background knowledge in multilayer perceptron learning. In M. Frydrych, J. Parkkinen, and A. Visa, editors, *Proc. of The 10th Scandinavian Conference on Image Analysis*, volume 2, pages 545–549, 1997.
- [9] Jouko Lampinen, Aki Vehtari, and Kimmo Leinonen. Using Bayesian neural network to solve the inverse problem in electrical impedance tomography. In B. K. Ersboll and P. Johansen, editors, *Proceedings of 11th Scandinavian Conference on Image Analysis SCIA'99*, pages 87–93, Kangerlussuaq, Greenland, 1999.
- [10] Jörg C. Lemm. 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, December 1996.
- [11] David J. C. MacKay. A practical Bayesian framework for backpropagation networks. *Neural Computation*, 4(3):448–472, 1992.
- [12] David J. C. MacKay. Bayesian non-linear modelling for the prediction competition. In *ASHRAE Transactions, V.100, Pt.2*, pages 1053–1062, Atlanta Georgia, 1994. ASHRAE.
- [13] Radford M. Neal. *Bayesian Learning for Neural Networks*. Springer-Verlag, 1996.
- [14] Radford M. Neal. Assessing relevance determination methods using DELVE. In Christopher M. Bishop, editor, *Neural Networks and Machine Learning*, pages 97–129. Springer-Verlag, 1998.
- [15] M. Vauhkonen, J. P. Kaipio, E. Somersalo, and P. A. Karjalainen. Electrical impedance tomography with basis constraints. *Inverse Problems*, 13(2):523–530, 1997.
- [16] Aki Vehtari, Jukka Heikkonen, Jouko Lampinen, and Jouni Juujärvi. Using Bayesian neural networks to classify forest scenes. In David P. Casasent, editor, *Proceedings of SPIE 3522*, pages 66–73. SPIE, 1998.
- [17] Aki Vehtari and Jouko Lampinen. Bayesian neural networks with correlating residuals. In *Proc. IJCNN'99*, Washington, DC, USA, July 1999.