1

Confidence Intervals and Prediction Intervals for Feed-Forward Neural Networks[a]

Richard Dybowski
King's College London

Stephen J. Roberts
Imperial College, London

[a] In R. Dybowski and V. Gant (eds.), *Clinical Applications of Artificial Neural Networks*, Cambridge University Press, (In press)

Artificial neural networks have been used as predictive systems for variety of medical domains, but none of the systems encountered by Baxt (1995) and Dybowski & Gant (1995) in their review of the literature provided any measure of confidence in the predictions made by those systems. In a medical setting, measures of confidence are of paramount importance (Holst, Ohlsson, Peterson & Edenbrandt 1998), and we introduce the reader to a number of methods that have been proposed for estimating the uncertainty associated with a value predicted by a feed-forward neural network.

The chapter opens with an introduction to regression and its implementation within the maximum-likelihood framework. This is followed by a general introduction to classical confidence intervals and prediction intervals. We set the scene by first considering confidence and prediction intervals based on univariate samples, and then we progress to regarding these intervals in the context of linear regression and logistic regression. Since a feed-forward neural network is a type of regression model, the concepts of confidence and prediction intervals are applicable to these networks, and we look at several techniques for doing this via maximum-likelihood estimation. An alternative to the maximum-likelihood framework is Bayesian statistics, and we examine the notions of Bayesian confidence and predictions intervals as applied to feed-forward networks. This includes a critique on Bayesian confidence intervals and classification.

1.1 Regression

Regression analysis is a common statistical technique for modelling the relationship between a *response* (or *dependent*) *variable* y and a set $\mathbf{x}$ of *regressors* $x_1, \ldots, x_d$ (also known as *independent* or *explanatory variables*). For example, the relationship could be between whether a patient has a malignant breast tumor (the response variable) and the patient's age and level of serum albumin (the regressors). When an article includes a discussion of artificial neural networks, it is customary to refer to response variables as *targets* and regressors as *inputs.* Furthermore, the ordered set $\{x_1, \ldots, x_d\}$ is sometimes referred to as an *input vector.* We will adopt this practice for the remainder of this chapter.

Regression assumes that target y is related to input vector $\mathbf{x}$ by stochastic and deterministic components. The stochastic component is the random fluctuation of y about its mean $\mu_y(\mathbf{x})$; for example, one possibility is

$$y = \mu_y(\mathbf{x}) + \varepsilon,$$

where *noise* ε, with zero mean, has a Gaussian distribution. The deterministic component is the functional relationship between $\mu_y(\mathbf{x})$ and $\mathbf{x}$.

If the 'true' functional relationship between $\mu_y(\mathbf{x})$ and $\mathbf{x}$ is given by

$$\mu_y(\mathbf{x}) = f(\mathbf{x}; \mathbf{w}_{true}), \tag{1.1}$$

where $\mathbf{w}$ is a set of parameters, regression attempts to estimate this relationship from a finite dataset (a *derivation* or *training set*) by estimating the parameter values from the data. This is done by adjusting the values of $\mathbf{w}$, under the assumption that f is the true function, to give

$$\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}}) = f(\mathbf{x}; \widehat{\mathbf{w}}), \tag{1.2}$$

where a hat denotes an estimated value. The function $f(\mathbf{x}; \widehat{\mathbf{w}})$ will be referred to as a *regression function*[1], and it will be used interchangeably with $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}})$. The best known example of eq.(1.2) is the *simple linear regression function*,

$$\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}}) = \widehat{w}_0 + \sum_{i=1}^{d} \widehat{w}_i x_i, \tag{1.3}$$

where $\widehat{w}_0, \widehat{w}_1, \ldots, \widehat{w}_d$ are the regression coefficients.

1.1.1 The maximum-likelihood framework

Suppose we have a dataset $\{\mathbf{x}^{(1)}, y^{(1)}, \ldots, \mathbf{x}^{(N)}, y^{(N)}\}$, where $y^{(n)}$ is the target value associated with the n-th input vector $\mathbf{x}^{(n)}$, and we wish to fit a regression function $f(\mathbf{x}; \widehat{\mathbf{w}})$ to this data. How do we select $\widehat{\mathbf{w}}$?

Maximum likelihood estimation (MLE) is based on the intuitive idea that the best estimate of $\widehat{\mathbf{w}}$ for $f(\mathbf{x}; \widehat{\mathbf{w}})$ is that set of parameter values $\widehat{\mathbf{w}}_{MLE}$

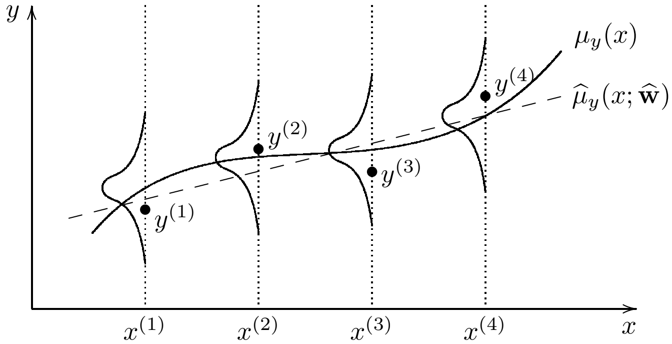


Fig. 1.1. An illustration of a regression function. The 'true' model consists of a probability function $p(y|x)$ for y, with a mean $\mu_y(x)$ (black curve) which is dependent on x. Dataset $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), (x^{(3)}, y^{(3)}), (x^{(4)}, y^{(4)})\}$ can be regarded as having been obtained by first randomly sampling $\{x^{(1)}, x^{(2)}, x^{(3)}, x^{(4)}\}$ from a population and then randomly sampling $y^{(1)}$ from $p(y|x^{(1)})$, $y^{(2)}$ from $p(y|x^{(2)})$, $y^{(3)}$ from $p(y|x^{(3)})$ and $y^{(4)}$ from $p(y|x^{(4)})$. Given the resulting dataset $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), (x^{(3)}, y^{(3)}), (x^{(4)}, y^{(4)})\}$, a regression function $\widehat{\mu}_y(x;\widehat{\mathbf{w}})$ (dashed line) attempts to estimate $\mu_y(x)$ by adjustment of a set of model parameters $\mathbf{w}$.

for which the observed data has the highest probability of arising. More formally,

$$\widehat{\mathbf{w}}_{MLE} = \arg\max_{\widehat{\mathbf{w}}} p(y^{(1)}, \ldots, y^{(N)}|\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}, \widehat{\mathbf{w}}), \tag{1.4}$$

$p(\cdot|\cdot\cdot)$ denoting a probability function[2].

Let the distribution of y about $\mu_y(\mathbf{x})$ be defined by a conditional probability distribution $p(y|\mathbf{x})$. For regression function $f(\mathbf{x};\widehat{\mathbf{w}})$, this distribution is approximated by $\widehat{p}(y|\mathbf{x},\widehat{\mathbf{w}})$ with mean $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$; therefore, if the cases of dataset $\{\mathbf{x}^{(1)}, y^{(1)}, \ldots, \mathbf{x}^{(N)}, y^{(N)}\}$ are sampled independently from the same population, eq.(1.4) can be simplified to

$$\widehat{\mathbf{w}}_{MLE} = \arg\min_{\widehat{\mathbf{w}}} \left[-\sum_{n=1}^{N} \ln \widehat{p}(y^{(n)}|\mathbf{x}^{(n)}, \widehat{\mathbf{w}}) \right]. \tag{1.5}$$

If the distribution of y about $\mu_y(\mathbf{x})$ is assumed to be Gaussian,

$$\widehat{p}(y|\mathbf{x},\mathbf{w}) = \frac{1}{\sqrt{2\pi}\sigma_y} \exp\left\{ \frac{-[\widehat{\mu}_y(\mathbf{x};\mathbf{w}) - y]^2}{\sigma_y^2} \right\}, \tag{1.6}$$

substitution of eq.(1.6) into the negative sum of eq.(1.5) (and ignoring constant terms) gives

$$\widehat{\mathbf{w}}_{MLE} = \arg\min_{\mathbf{w}} Err(\mathbf{w}), \tag{1.7}$$

where

$$Err(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \left[\widehat{\mu}_y(\mathbf{x}^{(n)}; \mathbf{w}) - y^{(n)} \right]^2, \tag{1.8}$$

$Err(\cdot)$ denoting an *error function.*

If a feed-forward neural network (FNN) $f(\mathbf{x}; \widehat{\mathbf{w}})$ is trained on dataset $\{\mathbf{x}^{(1)}, y^{(1)}, \ldots, \mathbf{x}^{(N)}, y^{(N)}\}$ by minimizing $Err(\mathbf{w})$, where $\mathbf{w}$ are the network weights, it can be shown that the resulting network approximates the mean value for y conditioned on $\mathbf{x}$ (Bishop 1995, pp. 201–203),

$$f(\mathbf{x}; \widehat{\mathbf{w}}_{MLE}) \approx \mu_y(\mathbf{x}), \tag{1.9}$$

the approximation becoming equality if N goes to infinity and $f(\mathbf{x}; \widehat{\mathbf{w}})$ has unlimited flexibility. Thus, from eq.(1.2), an FNN trained via $Err(\mathbf{w})$ can be regarded as a regression function.

1.2 Sources of uncertainty

There are two types of prediction that we may want from a regression function for a given input $\mathbf{x}$: one is the mean $\mu_y(\mathbf{x})$; the other is the target value y associated with $\mathbf{x}$.

Even if we are fortunate to have a regression function equal to the true model, so that $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}})$ is equal to $\mu_y(\mathbf{x})$ for all $\mathbf{x}$, y cannot be determined with certainty. This is due to the intrinsic random fluctuation of y about its mean $\mu_y(\mathbf{x})$ (*target noise*). When y is continuously-valued, the best one can do is establish a predictive probability density on y or a region where y is most likely to occur – a prediction interval. We will return to the concept of prediction intervals in the next section, our attention here being focused on $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}})$.

The acquisition of a training set $\{\mathbf{x}^{(1)}, y^{(1)}, \ldots, \mathbf{x}^{(N)}, y^{(N)}\}$ is prone to *sampling variation.* There are two reasons for this. Firstly, there is variability in the random sampling of $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}$ from the associated population. Secondly, for each selected $\mathbf{x}^{(n)}$, there is a random fluctuation in the value of y about the mean $\mu_y(\mathbf{x})$, as defined by $p(y|\mathbf{x})$ (figure 1.1). Consequently, the training set used for an FNN is only one of a large (possibly infinite) number of possibilities. Since each possible training set can give rise to a different set of network weights $\widehat{\mathbf{w}}$, it follows that there is a distribution of $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}})$ values for a given input $\mathbf{x}$.

If we randomly sample (with replacement) an infinite number of datasets $\mathcal{D}$, the resulting $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$ values will be distributed about the mean (or *expected value*) $\mathsf{E}_{\mathcal{D}}[\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})]$ with *sampling variance*

$$\mathsf{E}_{\mathcal{D}}[\{\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}}) - \mathsf{E}_{\mathcal{D}}[\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})]\}^2]$$

but $\mathsf{E}_{\mathcal{D}}[\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})]$ is not necessarily equal to $\mu_y(\mathbf{x})$, the difference

$$\mathsf{E}_{\mathcal{D}}[\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})] - \mu_y(\mathbf{x})$$

being the *bias.* The average proximity of $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$ to $\mu_y(\mathbf{x})$, taken over all $\mathcal{D}$, is related to the bias and sampling variance by the expression

$$\begin{aligned}&\mathsf{E}_{\mathcal{D}}[\{\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}}) - \mu_y(\mathbf{x})\}^2] = \\ &\qquad \underbrace{\{\mathsf{E}_{\mathcal{D}}[\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})] - \mu_y(\mathbf{x})\}^2}_{\{\text{bias}\}^2} + \underbrace{\mathsf{E}_{\mathcal{D}}[\{\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}}) - \mathsf{E}_{\mathcal{D}}[\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})]\}^2]}_{\text{variance}}. \qquad (1.10)\end{aligned}$$

Bias is due to a regression function having insufficient flexibility to model the data adequately. However, on increasing the flexibility in order to decrease bias, sampling variance is increased (this is graphically illustrated by Bishop (1995, p. 336)); thus, optimal

$$\mathsf{E}_{\mathcal{D}}[\{\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}}) - \mu_y(\mathbf{x})\}^2]$$

requires a tradeoff between bias and variance (Gemen, Bienenstock & Doursat 1992). The standard method for achieving this tradeoff with FNNs is to augment the error function with a term that penalizes against overfitting (a regularization term), such as the weight decay procedure (Hinton 1989).

When a regression function is an FNN, there are additional sources of error in $\widehat{\mathbf{w}}$ (Penny & Roberts 1997). One is due to the fact that an error function can have many local minima resulting in a number of possible $\widehat{\mathbf{w}}$. Another potential error in $\widehat{\mathbf{w}}$ arises from suboptimal training, for example, by premature termination of a training algorithm.

In the above discussion, uncertainty in $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$ has been attributed to uncertainty in $\widehat{\mathbf{w}}$, but there are two sources of uncertainty not originating from $\widehat{\mathbf{w}}$, namely, uncertainty in the input values (*input noise*, see section 1.6.4) and uncertainty in the structure of the regression model (*model uncertainty*). As regards the latter, regression model consists of two parts: an assumed structure for the model and a set of parameters $\mathbf{w}$ whose meaning is specific to the choice of model structure; therefore, uncertainty in $\widehat{\mu}_y$ $(\mathbf{x};\widehat{\mathbf{w}})$ should reflect the uncertainty in model structure as well as the uncertainty in $\widehat{\mathbf{w}}$. An approach to this problem has been suggested by Draper (1995), in which a range of structural alternatives are considered, but we are not aware of an application of this method to FNNs.

1.3 Classical confidence intervals and prediction intervals

There is uncertainty in the values of $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}})$ and y due to their respective distributions about the true mean $\mu_y(\mathbf{x})$. Such uncertainties can, in principle, be quantified by confidence and prediction intervals. We will define these terms and consider their application to regression, and thus to FNNs.

Let μ_v be the mean of a population of values v. The mean $\bar{v}$ of a sample $\mathbf{S}$ drawn randomly from the population is a point estimate of μ_v but, given that $\bar{v}$ is unlikely to be exactly equal to μ_v, how reliable a measure of μ_v is $\bar{v}$? A response to this question is to derive a lower limit $\lambda_L(\mathbf{S})$ and an upper limit $\lambda_U(\mathbf{S})$ from $\mathbf{S}$ such that there is a 95% probability that interval $[\lambda_L(\mathbf{S}), \lambda_U(\mathbf{S})]$ will contain μ_v. By this we mean that, if an infinite number of samples $\mathbf{S}_1, \mathbf{S}_2, \ldots$ of equal size are drawn randomly (with replacement) from the population, 95% of the intervals

$$[\lambda_L(\mathbf{S}_1), \lambda_U(\mathbf{S}_1)], [\lambda_L(\mathbf{S}_2), \lambda_U(\mathbf{S}_2)], \cdots$$

associated with these samples will overlap μ_v, which is fixed. Such an interval is referred to as a (*classical*) 95% *confidence interval* for μ_v.[3]

If sample $\mathbf{S}$ consists of univariate values $v^{(1)}, \ldots, v^{(N)}$, one can also consider an interval $[\psi_L(\mathbf{S}), \psi_U(\mathbf{S})]$ such that there is a 95% probability that a new value $v^{(N+1)}$ drawn randomly from the population will occur within the interval. Such an interval is referred to as a 95% *prediction interval* for $v^{(N+1)}$. Whereas a confidence interval is for a population parameter, a prediction interval is for a single value randomly drawn from the population.

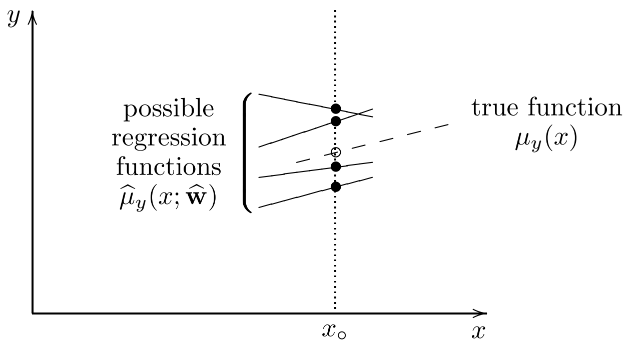
As an example, for sample $v^{(1)}, \ldots, v^{(N)}$, where v is continuously valued, the 95% prediction interval for $v^{(N+1)}$ is given by (Geisser 1993, pp. 6–9)

$$\bar{v} \pm t_{.025[N-1]} \left(s\sqrt{\frac{1}{N} + 1} \right),$$

where $t_{.025[N-1]}$ is the required critical value of Student's t-distribution ($N-1$ degrees of freedom), and s is the standard deviation of the sample. This interval is wider than the 95% confidence interval for μ_v,

$$\bar{v} \pm t_{.025[N-1]} \left(s\sqrt{\frac{1}{N}} \right),$$

because $v^{(N+1)}$ is variable whereas μ_v is constant.



y
possible regression functions $\widehat{\mu}_y(x;\widehat{\mathbf{w}})$
true function $\mu_y(x)$
$x_\circ$
x

Fig. 1.2. True function (*dashed line*) and several regression functions (*solid lines*) in the vicinity of $x_\circ$ (after Wonnacott & Wonnacott). The different regression functions are caused by variation in $\widehat{\mathbf{w}}$ due to sampling variation. Each black dot is a possible value for $\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}})$, the circle representing the correct value $\mu_y(x_\circ)$.

When v is binary valued, μ_v is equivalent to $p(v = 1)$, but the construction of a confidence interval for $p(v = 1)$ is complicated by the fact that $\bar{v}$ is discrete (Dudewicz & Mishra 1988, pp. 561–566). The discrete nature of $\bar{v}$ results in a confidence interval $[\lambda_L(\mathbf{S}), \lambda_U(\mathbf{S})]$ with *at least* a 95% probability of containing $p(v = 1)$. However, for large N, $\bar{v}$ can be assumed to have a normal distribution (Hogg & Craig 1995, pp. 272–273). Given that v is either 0 or 1 when it is binary valued, and nothing in between, there is no prediction interval for $v^{(N+1)}$ as such[4]. For the remainder of this chapter, confidence and prediction intervals will be understood to be of the classical type, unless stated otherwise.

1.3.1 Confidence and prediction intervals for simple linear regression

Confidence and prediction intervals can also be applied to regression, where they are collectively referred to as *error bars* by some authors. Variation in a finite sample $\mathbf{S}$ leads to variation in $\widehat{\mathbf{w}}$ and thus variation in $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$. Consequently, there is a distribution of possible values for $\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}})$ about $\mu_y(x_\circ)$, where $x_\circ$ is a particular value for x. This is illustrated in figure 1.2. In section 1.3, we described the idea of attaching an interval $[\lambda_L(\mathbf{S}), \lambda_U(\mathbf{S})]$ to $\bar{v}$ such that the interval has a 95% probability of overlapping with μ_v. In an analogous manner, we can conceptualize the existence of a 95% confidence interval $[\lambda_L(\mathbf{S}, x_\circ), \lambda_U(\mathbf{S}, x_\circ)]$ for $\mu_y(x_\circ)$ attached to each $\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}})$ by defining it in a manner analogous to the probabilistic interpretation given to

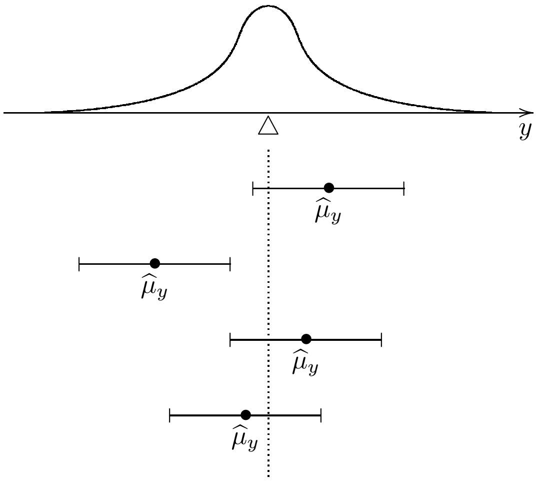


Fig. 1.3. An illustration of classical confidence intervals. Variation in $\widehat{\mathbf{w}}$ due to sampling variation results in a distribution of possible $\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}})$ values (figure 1.2). This distribution is defined by a probability distribution $p(\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}}))$ (the Gaussian curve). Four possible values of $\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}})$ randomly sampled from $p(\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}}))$ are shown (*black dots*). Also shown are the 95% confidence intervals associated with these four values. The triangle indicates the position of $\mu_y(x_\circ)$. 95% of all values sampled from $p(\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}}))$ will have their intervals correctly bracketing $\mu_y(x_\circ)$ if $\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}})$ is not biased. If $\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}})$ is biased then the mean of $p(\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}}))$ will not coincide with $\mu_y(x_\circ)$.

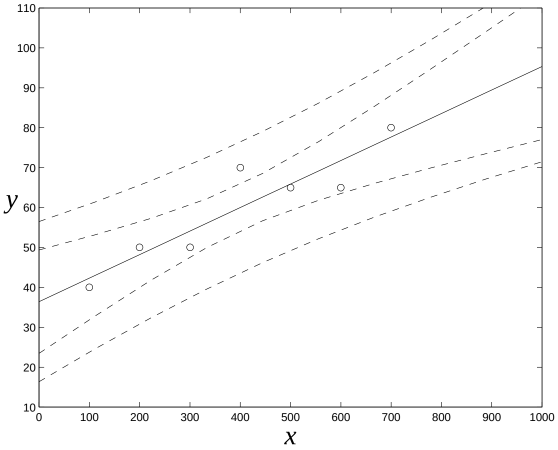


Fig. 1.4. Linear regression function (*solid line*) with a 95% confidence band for $\mu_y(x)$ (*region bounded by the inner dashed lines*) and a 95% prediction band for y (*region bounded by the outer dashed lines*) based on intervals (1.11) and (1.12), respectively. Each circle represents a data point.

confidence interval $[\lambda_L(\mathbf{S}, \lambda_U(\mathbf{S})]$ above, namely that $[\lambda_L(\mathbf{S}, x_\circ), \lambda_U(\mathbf{S}, x_\circ)]$ has a 95% probability of overlapping $\mu_y(x_\circ)$, which is fixed. A conceptual representation of this idea is given in figure 1.3. Furthermore, motivated by the above definition of prediction interval $[\psi_L(\mathbf{S}), \psi_U(\mathbf{S})]$, one can also conceptualize the existence of a 95% prediction interval $[\psi_L(\mathbf{S}, x_\circ), \psi_U(\mathbf{S}, x_\circ)]$ for the unknown value of y associated with $x_\circ$. For example, if we linearly regress y on x using $\{x^{(1)}, y^{(1)}, \ldots, x^{(N)}, y^{(N)}\}$ as the sample $\mathbf{S}$, the 95% confidence interval for $\mu_y(x_\circ)$ is

$$\widehat{\mu}_y(x_\circ; \widehat{\mathbf{w}}) \pm t_{.025[N-2]} \left(s\sqrt{\frac{1}{N} + \frac{(x_\circ - \bar{x})^2}{\sum_{n=1}^{N}(x^{(n)} - \bar{x})^2}} \right), \tag{1.11}$$

and the 95% prediction interval for y at $x_\circ$ is

$$\widehat{\mu}_y(x_\circ; \widehat{\mathbf{w}}) \pm t_{.025[N-2]} \left(s\sqrt{\frac{1}{N} + \frac{(x_\circ - \bar{x})^2}{\sum_{n=1}^{N}(x^{(n)} - \bar{x})^2} + 1} \right), \tag{1.12}$$

where s is the standard deviation for $y^{(1)}, \ldots, y^{(N)}$ and $\bar{x}$ is the mean of $x^{(1)}, \ldots, x^{(N)}$ (figure 1.4). Wonnacott & Wonnacott (1981, pp. 42-47) give a derivation of these intervals in the context of simple linear regression, and Penny & Roberts (1997) have been reviewed prediction intervals associated with other forms of linear regression.

A set of confidence intervals constructed continuously over an input x produces a two-dimensional *confidence band.* In a similar manner, a continuous set of prediction intervals over x produces a *prediction band.*

1.3.2 Confidence intervals for logistic regression

Logistic regression is the most popular technique for modelling a binary target y as a function of input vector $\mathbf{x}$ (Hosmer & Lemeshow 1989, Collett 1991)[5]. This is done by assuming that probability $p(y = 1|\mathbf{x})$ is related to $\mathbf{x}$ by a logistic function,

$$p(y = 1|\mathbf{x}) = \left\{ 1 + \exp\left[-\left(w_0 + \sum_{i=1}^{d} w_i x_i \right) \right] \right\}^{-1}. \tag{1.13}$$

When y is binary, $\mu_y(\mathbf{x})$ is equivalent to $p(y = 1|\mathbf{x})$, therefore, eq.(1.13) can be estimated as a regression function

$$\widehat{p}(y = 1|\mathbf{x}; \widehat{\mathbf{w}}) = \left\{ 1 + \exp\left[-\left(\widehat{w}_0 + \sum_{i=1}^{d} \widehat{w}_i x_i \right) \right] \right\}^{-1}. \tag{1.14}$$

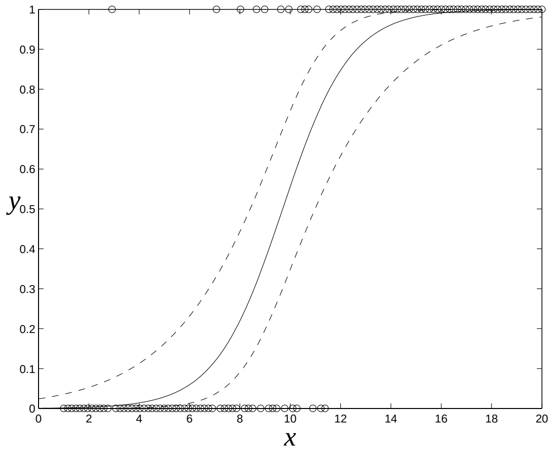


Fig. 1.5. Logistic regression function (*solid line*) with a 95% confidence band for $p(y=1|x)$ (*region bounded by the dashed lines*) according to Hauck's method (i.e., interval (1.16)).

In the context of maximum likelihood, eq.(1.5) still applies but the binary nature of y implies a binomial distribution for y,

$$p(y|\mathbf{x}) = p(y=1|\mathbf{x})^y \, [1 - p(y=1|\mathbf{x})]^{(1-y)} \, .$$

It follows that the error function for eq.(1.7), which is the negative logarithm of the relevant probability density, becomes

$$Err(\mathbf{w}) = -\sum_{n=1}^{N} \{ y^{(n)} \widehat{p}(y=1|\mathbf{x}^{(n)};\mathbf{w}) + [1-y^{(n)}][1-\widehat{p}(y=1|\mathbf{x}^{(n)};\mathbf{w})] \}. \tag{1.15}$$

As with any regression modelling, logistic regression is susceptible to sampling variation, consequently, the regression parameters, and thus the logistic regression function, are subject to variation. A representation of this variation is obtained from figure 1.2 by replacing $\mu_y(x_\circ)$ with $p(y=1|x_\circ)$ and $\widehat{\mu}_y(x_\circ;\widehat{\mathbf{w}})$ with $\widehat{p}(y=1|x_\circ;\widehat{\mathbf{w}})$, respectively. Just as with linear regression, the variation of $\widehat{p}(y=1|x_\circ;\widehat{\mathbf{w}})$ about $p(y=1|x_\circ)$ due to variation in $\widehat{\mathbf{w}}$ leads to the concept of a confidence interval $[\lambda_L(\mathbf{S},x_\circ), \lambda_U(\mathbf{S},x_\circ)]$ for $p(y=1|x_\circ)$. This interval has been derived analytically by Hauck (1983). If sample size N is large ($N > 100$), the 95% confidence interval for $p(y=1|\mathbf{x})$ is approximated by the logistic transform of

$$\mathrm{logit}\widehat{p}(y=1|\mathbf{x};\widehat{\mathbf{w}}) \pm \sqrt{\chi^2_{\alpha[d+1]} \mathbf{x}^{\mathsf{T}} \widehat{\mathbf{\Sigma}} \mathbf{x}/N}, \tag{1.16}$$

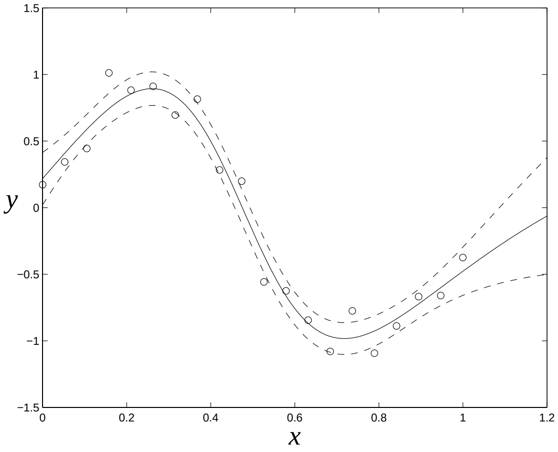


Fig. 1.6. Regression function (*solid line*) obtained from a feed-forward network with a 95% confidence band for $\mu_y(x)$ (*region bounded by dashed lines*) based on the delta method (i.e., interval (1.21)).

where $\mathbf{x}$ is a $d+1$ dimensional vector $(1, x_1, \ldots, x_d)^\mathsf{T}$, $\widehat{\boldsymbol{\Sigma}}$ is the covariance matrix for $\widehat{\mathbf{w}}$, and $\chi^2_{\alpha[d+1]}$ is the χ^2 critical value for the $100(1-\alpha)$ percentage point for $d+1$ degrees of freedom (figure 1.5)^6. See Santner & Duffy (1989, pp. 238–239) for further discussion.

1.4 Confidence intervals for feed-forward neural networks

So far, we have looked at linear and logistic regression, but if we have $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}})$ from an FNN, how can we obtain a confidence interval for $\mu_y(\mathbf{x})$? We start with two approaches : the delta method and the bootstrap method.

1.4.1 The delta method

If a variable v has a Gaussian distribution with variance $\mathsf{Var}(v)$, a 95% confidence interval for the mean of v is given by

$$v \pm z_{.025}\sqrt{\mathsf{Var}(v)},$$

where $z_{.025}$ is the critical point of the standard normal distribution. The delta method provides an estimate of this variance via the Taylor series.

If $\boldsymbol{\mu}_{\widehat{\mathbf{w}}}$ is the mean vector for $\widehat{\mathbf{w}}$, the first-order Taylor expansion of $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}})$ around $\boldsymbol{\mu}_{\widehat{\mathbf{w}}}$ gives the approximation

$$\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}}) \approx \widehat{\mu}_y(\mathbf{x}; \boldsymbol{\mu}_{\widehat{\mathbf{w}}}) + \mathbf{g}(\mathbf{x})(\widehat{\mathbf{w}} - \boldsymbol{\mu}_{\widehat{\mathbf{w}}}), \tag{1.17}$$

where the i-th element of vector $\mathbf{g}(\mathbf{x})$ is the partial derivative $\partial\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})/\partial\widehat{w}_i$ evaluated at $\widehat{\mathbf{w}} = \boldsymbol{\mu}_{\widehat{\mathbf{w}}}$. According to the *delta method* (Efron & Tibshirani 1993, pp. 313–315), it follows from (1.17) that the variance for $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$ over all possible samples is approximated by

$$\widehat{\mathsf{Var}}(\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})) = \mathbf{g}^{\mathsf{T}}(\mathbf{x})\boldsymbol{\Sigma}\mathbf{g}(\mathbf{x}), \tag{1.18}$$

where $\boldsymbol{\Sigma}$ is the covariance matrix for $\widehat{\mathbf{w}}$.

The elements of a Hessian matrix $\mathbf{H}$ are second-order partial derivatives[7]

$$\mathbf{H}_{i,j} = \frac{\partial^2 Err(\mathbf{w})}{\partial\mathbf{w}_i\partial\mathbf{w}_j},$$

evaluated at $\mathbf{w} = \widehat{\mathbf{w}}$, where $Err(\mathbf{w})$ is the relevant error function. Covariance matrix $\boldsymbol{\Sigma}$ is related to the Hessian (Press, Teukolsky, Vetterling & Flannery 1992, pp. 672–673, 685), and if the error function is defined as in eq.(1.8) and noise variance σ_ε^2 is independent of $\mathbf{x}$ then eq.(1.18) can be replaced by [8]

$$\widehat{\mathsf{Var}}(\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})) = \sigma_\varepsilon^2\mathbf{g}^{\mathsf{T}}(\mathbf{x})\mathbf{H}^{-1}\mathbf{g}(\mathbf{x}). \tag{1.19}$$

Tibshirani (1996) estimates σ_ε^2 using

$$\sigma_\varepsilon^2 = \sum_{i=1}^{N}\left[y^{(i)} - \widehat{\mu}_y(\mathbf{x}^{(i)};\widehat{\mathbf{w}})\right]^2/N.$$

From eq.(1.19), and assuming a Gaussian target noise distribution, we have the approximate 95% confidence interval for $\mu_y(\mathbf{x})$ (figure 1.6)

$$\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}}) \pm z_{.025}\sqrt{\sigma_\varepsilon^2\mathbf{g}^{\mathsf{T}}(\mathbf{x})\mathbf{H}^{-1}\mathbf{g}(\mathbf{x})}. \tag{1.20}$$

Regularization is the inclusion of a penalty term in an error function to discourage overfitting of the network to the training data. This improves the ability of the network to generalize from the data. If regularization is implemented by the weight-decay term $(\alpha/2)\Sigma_i w_i^2$, interval (1.20) is replaced by (Tibshirani 1996)[9]

$$\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}}) \pm z_{.025}\sqrt{\mathbf{g}^{\mathsf{T}}(\mathbf{x})(\mathbf{H}/\sigma_\varepsilon^2 - \alpha)^{-1}\mathbf{g}(\mathbf{x})}. \tag{1.21}$$

1.4.2 The bootstrap method

Suppose we have a random sample $\mathbf{S}$ taken from a population with parameter θ, and we obtain an estimate $\widehat{\theta}(\mathbf{S})$ of θ from $\mathbf{S}$. The *bootstrap method* is a remarkable computer-based resampling technique for assigning measures of accuracy to statistical estimates (Efron 1979)[10], and it will provide a confidence interval for any population parameter estimate whatsoever. This involves creating a number[11] of *bootstrap samples* $\mathbf{S}^{(*1)}, \ldots, \mathbf{S}^{(*B)}$ by repeatedly resampling $\mathbf{S}$ in a random manner in order to provide a distribution of $\widehat{\theta}(\mathbf{S})$: $\widehat{\theta}(\mathbf{S}^{(*1)}), \ldots, \widehat{\theta}(\mathbf{S}^{(*B)})$. The bootstrap estimate of the standard error of $\widehat{\theta}(\mathbf{S})$ is given by (Efron & Tibshirani 1993, pp. 45–49)

$$\widehat{SE}_{boot}(\widehat{\theta}(\mathbf{S})) = \sqrt{\frac{1}{B-1}\sum_{b=1}^{B}\left[\widehat{\theta}(\mathbf{S}^{(*b)}) - \widehat{\theta}_{boot}\right]^2},$$

where $\widehat{\theta}_{boot}$ is the bootstrap estimate of $\widehat{\theta}$ given by the mean $\sum_{b=1}^{B} \widehat{\theta}(\mathbf{S}^{(*b)})/B$, and B is typically in the range $25 - 200$.

In the context of regression, two types of bootstrap sample can be considered (Efron & Tibshirani 1993, pp. 113-115):

- *pairs sampling* in which regression is based on the bootstrap sample

  $$\{\mathbf{x}^{(*i,1)}, y^{(*i,1)}, \ldots, \mathbf{x}^{(*i,N)}, y^{(*i,N)}\}$$

  taken from the true sample $\{\mathbf{x}^{(1)}, y^{(1)}, \ldots, \mathbf{x}^{(N)}, y^{(N)}\}$, where $(*i, 1), \ldots, (*i, N)$ is the i-th random sample with replacement of the integers $1, \ldots, N$;
- *residual sampling* in which regression is based on the bootstrap sample

  $$\{\mathbf{x}^{(1)}, \widehat{\mu}_y(\mathbf{x}^{(1)}; \widehat{\mathbf{w}}) + r^{(*i,1)}, \ldots, \mathbf{x}^{(N)}, \widehat{\mu}_y(\mathbf{x}^{(N)}; \widehat{\mathbf{w}}) + r^{(*i,N)}\},$$

  where $r^{(*i,1)}, \ldots, r^{(*i,N)}$ is a random sample of the N residuals associated with $\widehat{\mu}_y(\mathbf{x}^{(1)}; \widehat{\mathbf{w}}), \ldots, \widehat{\mu}_y(\mathbf{x}^{(N)}; \widehat{\mathbf{w}})$, respectively.

Residual sampling has the advantage that it limits inferences to the set of input values $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}$ actually observed (Baxt & White 1995), but, unlike pairs sampling, it uses the strong assumption that residuals are independent of the inputs. Furthermore, the $\mathbf{x}$ values are assumed to be random in pairs sampling but fixed in residual sampling. The algorithms for pairs sampling and residual sampling are as follows.

Algorithm 1. (*Bootstrap pairs sampling*)

```
begin
   let $\{(\mathbf{x}^{(1)}, y^{(1)}), \ldots, (\mathbf{x}^{(N)}, y^{(N)})\}$ be the true sample $\mathbf{S}$;
   for $b = 1$ to $B$ do
      randomly sample (with replacement) $N$ $(\mathbf{x}, y)$-pairs from $\mathbf{S}$;
      let $\{(\mathbf{x}^{(*b,1)}, y^{(*b,1)}), \ldots, (\mathbf{x}^{(*b,N)}, y^{(*b,N)})\}$ be the random sample;
      derive regression function $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}}^{(*b)})$ from training set
         $\{(\mathbf{x}^{(*b,1)}, y^{(*b,1)}), \ldots, (\mathbf{x}^{(*b,N)}, y^{(*b,N)})\}$;
   endfor
```

```
end
```

Algorithm 2. (*Bootstrap residual sampling*)

```
begin
    let $\{(\mathbf{x}^{(1)}, y^{(1)}), \ldots, (\mathbf{x}^{(N)}, y^{(N)})\}$ be the true sample $\mathbf{S}$;
    derive regression function $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}})$ from $\mathbf{S}$;
    let $\mathbf{R}$ be the set of residuals $\{r^{(1)}, \ldots, r^{(N)}\}$, where
        $r^{(n)} = y^{(n)} - \widehat{\mu}_y(\mathbf{x}^{(n)}; \widehat{\mathbf{w}})$;
    for $b = 1$ to $B$ do
        randomly sample (with replacement) $N$ residuals from $\mathbf{R}$;
        let $\{r^{(*b,1)}, \ldots, r^{(*b,N)}\}$ be the random sample;
        derive regression function $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}}^{(*b)})$ from training set
            $\{(\mathbf{x}^{(1)}, \widehat{\mu}_y(\mathbf{x}^{(1)}; \widehat{\mathbf{w}}) + r^{(*b,1)}), \ldots, (\mathbf{x}^{(N)}, \widehat{\mu}_y(\mathbf{x}^{(N)}; \widehat{\mathbf{w}}) + r^{(*b,N)})\}$;
    endfor
end
```

For both the paired-sampling and residual-sampling approaches, the bootstrap estimate of $\widehat{\mu}_y(\mathbf{x})$ is given by the mean provided by the ensemble of regression functions $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}}^{(*1)}), \ldots, \widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}}^{(*B)})$:

$$\widehat{\mu}_{y,boot}(\mathbf{x}) = \frac{1}{B} \sum_{b=1}^{B} \widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}}^{(*b)}). \tag{1.22}$$

Furthermore, the bootstrap estimate of the standard error of $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}})$, which is a function of $\mathbf{x}$, is given by

$$\widehat{SE}_{boot}(\widehat{\mu}_y(\mathbf{x}; \cdot)) = \sqrt{\frac{1}{B-1} \sum_{b=1}^{B} \left[\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}}^{(*b)}) - \widehat{\mu}_{y,boot}(\mathbf{x})\right]^2}, \tag{1.23}$$

with $\widehat{\mu}_{y,boot}(\mathbf{x})$ defined as in eq.(1.22). Assuming a normal distribution for $\widehat{\mu}_y(\mathbf{x}; \widehat{\mathbf{w}})$ over the space of all possible $\widehat{\mathbf{w}}$, we have

$$\widehat{\mu}_{y,boot}(\mathbf{x}) \pm t_{.025[B]} \widehat{SE}_{boot}(\widehat{\mu}_y(\mathbf{x}; \cdot))$$

as the 95% bootstrap confidence interval for $\mu_y(\mathbf{x})$ (Heskes 1997).

As stated earlier, logistic regression provides a regression function that estimates the conditional probability $p(y = 1|\mathbf{x})$. By using a logistic transfer function for the output node, and the cross-entropy error function (1.15), $p(y = 1|\mathbf{x})$ can also be estimated by an FNN trained with binary target values. Furthermore, the bootstrap estimate $\widehat{\mu}_{y,boot}(\mathbf{x})$ provides a mean conditional probability with the advantages of a *bagged* predictor (Breiman 1996). The concept of a confidence interval for $p(y = 1|\mathbf{x})$, as used for logistic

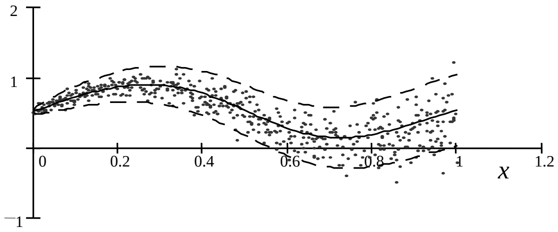


Fig. 1.7. Data with increasing variance. The regression function (*solid line*) was estimated by a feed-forward network. Another feed-forward network was used to estimate the input-dependent variance from which a 95% prediction band (*region bounded by dashed lines*) was obtained by interval (1.26).

regression, can also be applied to an FNN; however, we have not found a published description of a bootstrap confidence interval for $p(y = 1|\mathbf{x})$ via an FNN.

A disadvantage of the bootstrap method is that the computational cost could be high when datasets or networks are large; however, Tibshirani (1996) found that the bootstrap approach provided more accurate confidence intervals than the delta method. A contribution to this success is that bootstrap sampling takes into account the variability of FNNs due to different initial network weights. Another factor in favour of the bootstrap method is the fact that the delta method requires computation of derivatives and Hessian-matrix inversion, the latter being a potential source of failure.

1.5 Prediction intervals for feed-forward neural networks

If y has a Gaussian distribution with mean $\mathsf{E}[y|\mathbf{x}]$ and variance $\mathsf{Var}(y|\mathbf{x})$, a 95% prediction interval for y is given by

$$\mathsf{E}[y|\mathbf{x}] \pm z_{.025}\sqrt{\mathsf{Var}(y|\mathbf{x})}.$$

This is the basis for an approximate prediction interval, as follows.

The variance of y conditioned on $\mathbf{x}$ is defined by

$$\mathsf{Var}(y|\mathbf{x}) = \mathsf{E}[(\mathsf{E}[y|\mathbf{x}] - y)^2|\mathbf{x}].$$

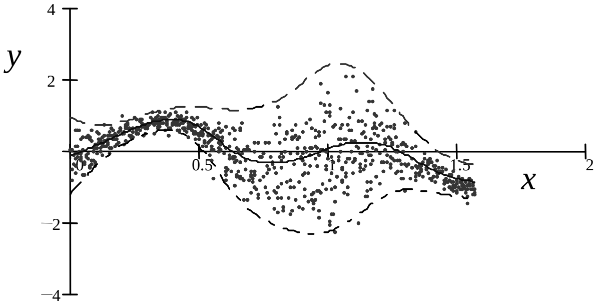


Fig. 1.8. Both the regression function (*solid line*) and its associated 95% prediction band (*region bounded by dashed lines*) were obtained from a Nix-Weigend network.

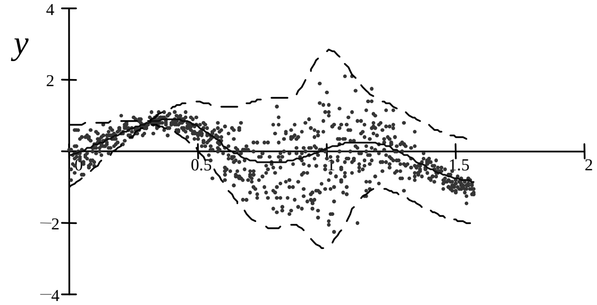


Fig. 1.9. Same data as that in figure 1.8 but, instead of using a Nix-Weigend network, a separate feed-forward network estimated the variance. This resulted in a decrease in the accuracy of the 95% prediction band (*region bounded by dashed lines*).

Recall that an FNN $\widehat{\mu}_y(\mathbf{x}^{(n)}; \widehat{\mathbf{w}})$ trained with respect to error function

$$\frac{1}{2}\sum_{n=1}^{N}\left[\widehat{\mu}_y(\mathbf{x}^{(n)}; \mathbf{w}) - y^{(n)}\right]^2, \tag{1.24}$$

can approximate $\mathsf{E}[y|\mathbf{x}]$. This suggests that, in order to obtain $\mathsf{E}[(\mathsf{E}[y|\mathbf{x}] - y)^2|\mathbf{x}]$ in place of $\mathsf{E}[y|\mathbf{x}]$ by means of an FNN $\widehat{\sigma}_y^2(\mathbf{x}; \widehat{\mathbf{u}})$, we should replace y

in error function (1.24) with $(\mathsf{E}[y|\mathbf{x}]-y)^2$. Therefore, if $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$ is assumed to be equal to $\mathsf{E}[y|\mathbf{x}]$, an FNN $\widehat{\sigma}_y^2(\mathbf{x};\widehat{\mathbf{u}})$ for the estimation of $\mathsf{Var}(y|\mathbf{x})$ can be derived by using

$$\frac{1}{2}\sum_{n=1}^{N}\left[\widehat{\sigma}_y^2(\mathbf{x};\mathbf{u})-[\widehat{\mu}_y(\mathbf{x}^{(n)};\widehat{\mathbf{w}})-y^{(n)}]^2\right]^2 \tag{1.25}$$

as the error function. This leads to the approximate 95% prediction interval

$$\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})\pm z_{.025}\sqrt{\widehat{\sigma}_y^2(\mathbf{x};\widehat{\mathbf{u}})}. \tag{1.26}$$

A 95% prediction band resulting from this interval is shown in figure 1.7.

Rather than using two separate networks, Nix & Weigend (1995) proposed a single network with one output for $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$ and another for $\widehat{\sigma}_y^2(\mathbf{x};\widehat{\mathbf{u}})$, using

$$\frac{1}{2}\sum_{n=1}^{N}\left[\frac{[\widehat{\mu}_y(\mathbf{x}^{(n)};\mathbf{w})-y^{(n)}]^2}{\widehat{\sigma}_y^2(\mathbf{x};\mathbf{u})}+\ln\widehat{\sigma}_y^2(\mathbf{x};\mathbf{u})\right]^2 \tag{1.27}$$

as the error function. This approach can produce improved prediction intervals for y compared with the previous approach as a result of it acting as a form of weighted regression (weighted in favour of low-noise regions) (figure 1.8). The simpler approach based on expression (1.25) tries to fit around high-noise regions, possibly distorting the low-noise regions (figure 1.9), whereas weighted regression is not influenced by regions of high fluctuation.

An underlying assumption in using either (1.25) or (1.27) is that $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$ is equal to $\mathsf{E}[y|\mathbf{x}]$, but, when this assumption is false, there will be uncertainty in $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$, in which case (1.26) will underestimate the 95% prediction interval. A prediction interval that allows for the uncertainty in both the regression function $\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})$ and the noise $y-\mu_y(\mathbf{x})$ is

$$\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}})\pm t_{.025[\nu]}\sqrt{\widehat{\mathsf{Var}}(\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}}))+\widehat{\sigma}_\varepsilon^2}, \tag{1.28}$$

where $\widehat{\sigma}_\varepsilon^2$ is the estimated noise variance, but the degrees of freedom ν required for an FNN is not known at the time of writing. Heskes (1997) proposed the bootstrap method as a way to derive (1.28). Bootstrap estimate (1.23) was used for $\widehat{\mathsf{Var}}(\widehat{\mu}_y(\mathbf{x};\widehat{\mathbf{w}}))$, and an auxiliary FNN, trained on the unused portions of the bootstrap samples, was used to estimate $\widehat{\sigma}_\varepsilon^2$. Although Heskes obtained more realistic prediction intervals than provided by the Nix-Weigend method (1995), we feel that his technique requires further analysis.

The methods used in this section are based on maximum likelihood estimation, but variances estimated by MLE are biased:

$$\mathsf{E}[\widehat{\mathsf{Var}}_{MLE}(y|\mathbf{x})] < \mathsf{Var}(y|\mathbf{x}).$$

This is caused by a tendency of an interpolant to try and fit to the data, thereby underestimating $\mathsf{Var}(y|\mathbf{x})$. Consequently, if interval (1.26) or (1.28) is used as the 95% prediction interval for y, the length of the interval will be underestimated.

1.6 The Bayesian framework

The primary purpose of statistics is to make an inference about a population on the basis of a sample taken from the population. In classical statistics, the inference is based solely on the data consituting the sample, whereas, in *Bayesian statistics*, the inference is based on a combination of prior belief and sample data (Lee 1997). In order to make a Bayesian inference about a random variable θ, prior belief about θ in the form of a *prior (probability) distribution* $p(\theta)$, is combined with a sample $\mathbf{S}$ of values in order to produce a *posterior (probability) distribution* $p(\theta|\mathbf{S})$ for θ.

Confidence and prediction intervals are also defined within the Bayesian framework. Let μ_v be the mean of a population of values v, and let $\mathbf{S}$ be an observed sample of values drawn from the population. If μ_v is regarded as a random variable with posterior distribution $p(\mu_v|\mathbf{S})$, $[\lambda_L(\mathbf{S}_1), \lambda_U(\mathbf{S}_1)]$ is a 95% *Bayesian confidence interval* for μ_v if, according to $p(\mu_v|\mathbf{S})$, there is a 95% probability that μ_v will fall within $[\lambda_L(\mathbf{S}), \lambda_U(\mathbf{S})]$ (Barnett 1982, pp. 198–202). Note the difference between a classical confidence interval and a Bayesian confidence interval: in the classical approach, μ_v is fixed and $[\lambda_L(\mathbf{S}), \lambda_U(\mathbf{S})]$ varies with $\mathbf{S}$; in the Bayesian approach, μ_v is a random variable and $[\lambda_L(\mathbf{S}), \lambda_U(\mathbf{S})]$ is fixed once $\mathbf{S}$ is available (Lee 1997, pp. 49–50).

If sample $\mathbf{S}$ consists of univariate values $v^{(1)}, \ldots, v^{(N)}$, and $p(v^{(N+1)}|\mathbf{S})$ is the posterior distribution for $v^{(N+1)}$, $[\psi_L(\mathbf{S}), \psi_U(\mathbf{S})]$ is a 95% *Bayesian prediction interval* for $v^{(N+1)}$ if, according to $p(v^{(N+1)}|\mathbf{S})$, there is a 95% probability that $[\psi_L(\mathbf{S}), \psi_U(\mathbf{S})]$ will contain $v^{(N+1)}$ (Barnett 1982, pp. 204–205).

1.6.1 Bayesian intervals for regression

Bayesian statistics provides a very different approach to the problem of unknown model parameters such as network weights. Instead of considering just a single value for a model parameter, as done by maximum likelihood

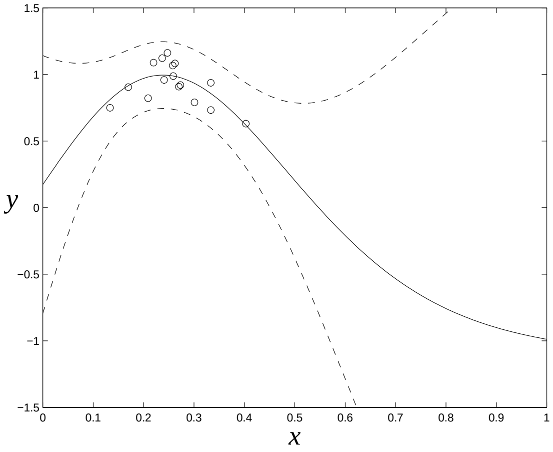


Fig. 1.10. A 95% Bayesian prediction band for y (*region bounded by dashed lines*) based on interval (1.34). The regression function (*solid line*) is from a feed-forward network.

estimation, Bayesian inference expresses the uncertainty of parameters in terms of probability distributions and integrates them out of the distribution of interest. For example, by expressing the uncertainty in weight vector $\mathbf{w}$ as the posterior probability distribution $p(\mathbf{w}|\mathbf{S})$, where $\mathbf{S}$ is the observed sample, we have

$$p(y|\mathbf{x},\mathbf{S}) = \int_{\mathbf{w}} p(y|\mathbf{x},\mathbf{w})p(\mathbf{w}|\mathbf{S})d\mathbf{w} \tag{1.29}$$

$$\propto \int_{\mathbf{w}} p(y|\mathbf{x},\mathbf{w})p(\mathbf{S}|\mathbf{w})p(\mathbf{w})d\mathbf{w}. \tag{1.30}$$

The integral of eq.(1.30) can be solved analytically with approximations (MacKay 1991). If the distribution of the noise and the prior weight distribution $p(\mathbf{w})$ are assumed to be Gaussian, a Gaussian posterior distribution $p(y|\mathbf{x},\mathbf{w}_{MP})$ for y can be derived in which

$$\widehat{\mathsf{E}}[y|\mathbf{x}] = \widehat{\mu}_y(\mathbf{x};\mathbf{w}_{MP}), \tag{1.31}$$

where $\mathbf{w}_{MP}$ is $\mathbf{w}$ at the maximum of the posterior weight distribution $p(\mathbf{w}|\mathbf{S})$, and

$$\widehat{\mathsf{Var}}(y|\mathbf{x}) = \beta^{-1} + \mathbf{g}^{\mathsf{T}}(\mathbf{x})\mathbf{A}^{-1}\mathbf{g}(\mathbf{x}), \tag{1.32}$$

where the elements of matrix $\mathbf{A}$ are the second-order partial derivatives (with respect to $\mathbf{w}$) of the regularized error function

$$\frac{\beta}{2}\sum_{n=1}^{N}\left[\widehat{\mu}_y(\mathbf{x}^{(n)};\mathbf{w}) - y^{(n)}\right]^2 + \frac{\alpha}{2}\sum_i w_i^2 \tag{1.33}$$

evaluated at $\mathbf{w} = \mathbf{w}_{MP}$. The second term in (1.33) (the regularization term) results from the assumption that $p(\mathbf{w})$ in eq.(1.30) is Gaussian. This leads to the approximate 95% Bayesian prediction interval for y (figure 1.10)

$$\widehat{\mu}_y(\mathbf{x};\mathbf{w}_{MP}) \pm z_{.025}\sqrt{\beta^{-1} + \mathbf{g}^{\mathsf{T}}(\mathbf{x})\mathbf{A}^{-1}\mathbf{g}(\mathbf{x})}. \tag{1.34}$$

Note that MLE has been avoided through the use of eq.(1.29).

The Bayesian analysis resulting in expression (1.32) demonstrates that the variance for $p(y|\mathbf{x},\mathbf{S})$ has contributions from the intrinsic noise variance β^{-1} and from the weight uncertainty. Qazaz, Williams & Bishop (1996) discuss how, in the context of generalized linear regression, this is effected by the distribution of data points.

Instead of using a constant value for noise variance β^{-1}, Bishop & Qazaz (1995) allowed it to be dependent on $\mathbf{x}$. From 100 artificially-generated datasets, each consisting of 10 data points, they demonstrated that the Bayesian approach can give an improved estimate of noise variance compared to a more biased estimate obtained from the same data using MLE.

1.6.2 Bayesian intervals for regression-based classification

We consider, as before, a feed-forward system which estimates class-conditional posterior probabilities. For class C_i, say, given datum $\mathbf{x}$, this is denoted as $p(C_i|\mathbf{x}) = p(y_i = 1|\mathbf{x})$. The K outputs $\widehat{p}(y_1 = 1|\mathbf{x};\mathbf{w}),\ldots,\widehat{p}(y_K = 1|\mathbf{x};\mathbf{w})$ of such a classifier, hence, must lie in the interval $[0,1]$ and sum to unity. This may be simply achieved via the *softmax* (or generalized sigmoid) mapping of a set of latent variables, $r_1,\ldots,r_K$, such that

$$\widehat{p}(y_i = 1|\mathbf{x};\mathbf{w}) = \frac{\exp(r_i)}{\sum_{j=1}^{K}\exp(r_j)}. \tag{1.35}$$

For a two-class problem, we need consider only one output, $\widehat{p}(y = 1|\mathbf{x};\mathbf{w})$, and eq.(1.35) reduces to the well-known logistic sigmoid,

$$\widehat{p}(y = 1|\mathbf{x};\mathbf{w}) = g\left(r(\mathbf{x};\mathbf{w})\right) = \{1 + \exp[-r(\mathbf{x};\mathbf{w})]\}^{-1}.$$

For ease of notation we will consider, henceforth, the two-class case, with a single output estimating $p(C_1|\mathbf{x})$ (which may also be denoted $p(y = 1|\mathbf{x})$).

MacKay (1992*b*) suggested approximating the variation of r with $\mathbf{w}$ by a linear (first-order) expansion, and the density over $\mathbf{w}$, $p(\mathbf{w}|\mathbf{S})$ by a unimodal normal distribution. This enables $p(r|\mathbf{x};\mathbf{w},\mathbf{S})$ to be evaluated easily from $p(\mathbf{w}|\mathbf{S})$. If we make a Laplace approximation to the latter (de Bruijn 1970) then $p(r|\mathbf{x};\mathbf{w},\mathbf{S})$ will also be approximated by a Gaussian (normal) distribution with mean (and mode) at

$$r_{MP}(\mathbf{x}) = r(\mathbf{x};\mathbf{w}_{MP}).$$

The variance of $p(r|\mathbf{x};\mathbf{w},\mathbf{S})$ is given as (e.g., Bishop 1995, p. 405)

$$\widehat{\mathsf{Var}}\,(r|\mathbf{x};\mathbf{w},\mathbf{S}) = \mathbf{h}^{\mathsf{T}}(\mathbf{x})\mathbf{B}^{-1}\mathbf{h}(\mathbf{x}), \tag{1.36}$$

where $\mathbf{h}(\mathbf{x})$ is the the partial derivative $\partial r(\mathbf{x};\mathbf{w})/\partial w_i$ evaluated at $\mathbf{w} = \mathbf{w}_{MP}$ and the elements of the Hessian matrix, $\mathbf{B}$, are the second-order partial derivatives of the error function with respect to $\mathbf{w}$, evaluated at $\mathbf{w} = \mathbf{w}_{MP}$,

$$\mathbf{B}_{i,j} = \frac{\partial^2 Err(\mathbf{w})}{\partial w_i \partial w_j}.$$

The error function is normally a cross-entropy measure (eq.(1.15)) with an additive regularization term.

We may consider the location of the mode (most-probable value) of the latent distribution, $r_{MP}(\mathbf{x})$, as propagating through the sigmoidal nonlinearity, $g(.)$, to form a MLE for the posterior,

$$\widehat{p}(y=1|\mathbf{x};\mathbf{w},\mathbf{S}) = g\left(r_{MP}(\mathbf{x})\right).$$

The monotonicity of $g(.)$ means that the upper and lower bounds of a confidence interval on the latent distribution $p(r|\mathbf{x};\mathbf{w},\mathbf{S})$ could be mapped to equivalent points in the posterior space. This is supported by advocates of set-based (or interval-based) probability (e.g., Kyburg & Pittarelli 1996).

From a Bayesian decision-theoretic viewpoint, however, the notion of a confidence interval on posterior probabilities in a classification setting is redundant as uncertainty (confidence) is uniquely taken into account under a Bayesian derivation of the single-valued posteriors. We consider an optimal classifier, which provably operates by assigning an unknown datum $\mathbf{x}$ to class C_{k^*} if and only if

$$p(C_{k^*}|\mathbf{x}) = \max_k\{p(C_k|\mathbf{x})\},$$

in other words, in a two-class setting for which $p(C_1|\mathbf{x}) = p(y=1|\mathbf{x})$, $\mathbf{x}$ is classified to class C_1 if $p(y=1|\mathbf{x}) > 1 - p(y=1|\mathbf{x})$. A strict measure of the loss or uncertainty associated with a decision to C_{k^*} is $1 - p(C_{k^*}|\mathbf{x})$. Our inherent confidence in a decision is given by this quantity. Note that, if equal penalties are accrued for misclassification from all classes (i.e., the so-called *loss matrix* is isotropic) the same *decision* will be made, in a two class case, for $p(C_{k^*}|\mathbf{x}) = 0.51$ or 0.99, but our confidence in the decision is dramatically different. Indeed, it is common practise to include a 'reject' class such that $\mathbf{x}$ is rejected if $p(C_{k^*}|\mathbf{x}) < 1 - d$, where $d \in [1/2, 1]$ is a measure of the cost associated with falsely rejecting of the sample $\mathbf{x}$. How then is uncertainty incorporated in the Bayesian derivation of the posteriors?

Consider the measure $p(y = 1|\mathbf{x}; \mathbf{w}, \mathbf{S})$ (the posterior for class C_1) explicitly dependent upon the input $\mathbf{x}$ and implicitly on the 'training' data set $\mathbf{S}$ and the set of unknown parameters, $\mathbf{w}$, which code the analysis model. The MLE framework considers only the most probable parameter set, $\mathbf{w}_{MP}$, which is used to estimate $p(y = 1|\mathbf{x}; \mathbf{w}, \mathbf{S})$. This results in $p(y = 1|\mathbf{x}; \mathbf{w}_{MP}, \mathbf{S})$.

In contrast the Bayesian paradigm integrates over the unknown parameters,

$$\tilde{p}(y = 1|\mathbf{x}; \mathbf{S}) = \int_{\mathbf{w}} p(y = 1|\mathbf{x}; \mathbf{S}, \mathbf{w}) p(\mathbf{w}|\mathbf{S}) d\mathbf{w}.$$

If we consider our analysis model in which $p(y = 1|\mathbf{x}; \mathbf{w}, \mathbf{S})$ is obtained via a monotone mapping $g(.)$ (the logistic sigmoid) from a continuous-valued latent variable r, i.e. $p(y = 1|\mathbf{x}; \mathbf{w}, \mathbf{S}) = g(r; \mathbf{x}, \mathbf{w}, \mathbf{S})$ then we may re-write the above as

$$\tilde{p}(y = 1|\mathbf{x}; \mathbf{S}) = \int_{r} g(r; \mathbf{x}, \mathbf{w}, \mathbf{S}) p(r|\mathbf{x}; \mathbf{w}, \mathbf{S}) dr,$$

where $p(r|\mathbf{x}; \mathbf{w}, \mathbf{S})$ is the distribution in r induced by the distribution in the weights $\mathbf{w}$ upon which r is dependent. The above integral, however, is typically analytically intractable but may be easily evaluated using numerical techniques. MacKay (1992) popularized some approximations (originally considered by Spiegelhalter & Lauritzen (1990)) which not only avoid this process but also highlight intuitively the way in which uncertainty in $\mathbf{w}$, which propagates as an uncertainty in r (i.e., $p(r|\mathbf{x}; \mathbf{w}, \mathbf{S})$ is wide), changes the posterior probability. This change in the posterior is known as *moderation* and typically results in improved cross-entropy errors (MacKay 1992*a*). The approximation considers a modification to the sigmoid equation of the form

$$\tilde{p}(y = 1|\mathbf{x}, \mathbf{S}) \approx g\left\{\kappa[\sigma_r^2(\mathbf{x})] r_{MP}(\mathbf{x})\right\}, \tag{1.37}$$

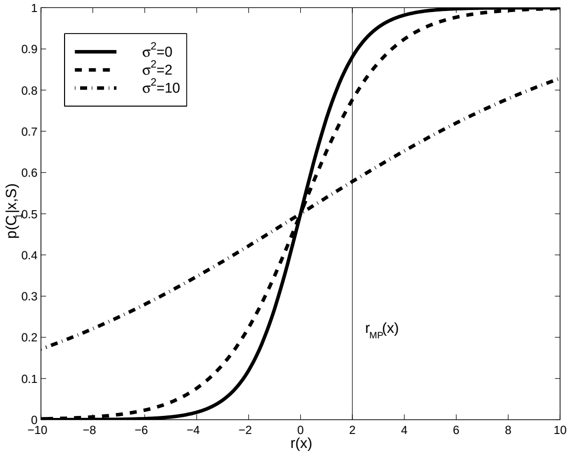


Fig. 1.11. Changes in slope of sigmoid due to latent variable uncertainty.

in which

$$\kappa[\sigma_r^2(\mathbf{x})] = \left(1 + \frac{\pi\sigma_r^2(\mathbf{x})}{8}\right)^{-1/2}$$

and $\sigma_r^2(\mathbf{x})$ is the variance of the latent variable distribution, as defined in eq.(1.36). Figure 1.11 depicts the effect changes in the latent variance (uncertainty) have on the classification probability, $\tilde{p}(y = 1|\mathbf{x}, \mathbf{S}) = p(C_1|\mathbf{x}, \mathbf{S})$. Consider, for example, $r_{MP}(\mathbf{x}) = 2$. Note that the resultant estimated posterior probability goes down towards 1/2 as the uncertainty in r increases. The uncertainty in a decision is the distance from unity of the largest posterior, which is worst when the posterior equals the class prior (1/2 in this two-class problem). In a principled way, therefore, uncertainty (high variance) in the latent distribution is automatically represented as a lower certainty of decision.

The tacit assumption has been made in the above analysis that the density over the weights, $p(\mathbf{w}|\mathbf{S})$, is unimodal. For the vast majority of analysis systems, however, there are many non-equivalent local maxima in the density which would be taken into account if the requisite marginal integral was indeed over all $\mathbf{w}$ space. We may assume, however, that most probability mass is concentrated in the regions of $\mathbf{w}$ space associated with peaks in $p(\mathbf{w}|\mathbf{S})$. Integration over all $\mathbf{w}$ space may hence be approximated by integration over a (finite) number of regions, $\mathcal{R}_i$, each of which contains a peak in $p(\mathbf{w}|\mathbf{S})$.

Hence

$$p(r|\mathbf{x};\mathbf{S}) \approx \sum_i p(\mathcal{R}_i|\mathbf{S}) \int_{\mathbf{w}\in\mathcal{R}_i} p(r|\mathbf{x};\mathbf{w},\mathbf{S},\mathcal{R}_i)p(\mathbf{w}|\mathbf{S},\mathcal{R}_i)d\mathbf{w}$$

which may be written as

$$p(r|\mathbf{x};\mathbf{S}) = \sum_i \gamma_i p(r|\mathbf{x};\mathbf{S},\mathcal{R}_i).$$

This latter equation represents a weighted average (with weightings given by γ_i) of latent densities from a *committee* of classifiers. Each latent distribution in the summation may, for example, be approximated as a Gaussian, as may the resultant committee distribution. The latter has mean

$$r_{MP}^{comm}(\mathbf{x}) = \sum_i \gamma_i r_{MP,i}(\mathbf{x}),$$

where $r_{MP,i}(\mathbf{x})$ are the modes (and means) of $p(r|\mathbf{x};\mathbf{S},\mathcal{R}_i)$, and a variance of

$$\sigma_{comm}^2(\mathbf{x}) = \sum_i \gamma_i \sigma_{r,i}^2(\mathbf{x}) + \sum_i \gamma_i (r_{MP,i}(\mathbf{x}) - r_{MP}^{comm}(\mathbf{x}))^2. \tag{1.38}$$

This variance may thence be used, for example, with eq.(1.37) to provide a moderated posterior probability which takes into account uncertainty due to imprecision in the parameters of each constituent member of the committee (the first term in eq.(1.38)) and also uncertainty due to disagreement between committee members (the second term in eq.(1.38)). It is noted that committees are provably better in performance than the average performance of their members (Bishop 1995, pp. 364–369).

1.6.3 Markov chain Monte Carlo sampling

Determination of the integral in eq.(1.29) can, in principle, be achieved numerically using

$$p(y|\mathbf{x},\mathbf{S}) \approx \frac{1}{L}\sum_{i=1}^{L} p(y|\mathbf{x},\mathbf{w}^{(i)}). \tag{1.39}$$

This avoids the Gaussian approximations adopted in section 1.6.1 and elsewhere.

The set $\{\mathbf{w}^{(1)},\ldots,\mathbf{w}^{(L)}\}$ of weight vectors used for approximation (1.39) is sampled from $p(\mathbf{w}|\mathbf{S})$ by means of *Markov chain Monte Carlo* (MCMC) sampling (Gilks, Richardson & Spiegelhalter 1996). In the two standard versions of MCMC sampling (the Metropolis method and Gibbs sampling), the space of possible $\mathbf{w}$ values (state space) is explored by random walk; however, sampling through a random walk can perform poorly when the state space has a large number of dimensions. In such a situation, Neal (1996) advocates the *hybrid Monte Carlo method* in which state space is replaced by a phase space consisting of $(\mathbf{w}, \mathbf{p})$ pairs in which 'position' vector $\mathbf{w}$ is augmented with a 'momentum' vector $\mathbf{p}$. Unlike Metropolis and Gibbs sampling, this exploits the gradient information contained in a backpropagation-trained network.

An example of the application of MCMC is its use by Goldberg, Williams & Bishop (1998) to model input-dependent variance, which they did using a Gaussian process (Williams 1999).

The assumption that $p(y|\mathbf{x}, \mathbf{w})$ is Gaussian whenever y is continuous-valued will not always be appropriate in the real world as it is possible for $p(y|\mathbf{x}, \mathbf{w})$ to be skewed or multi-modal due to the noise being non-Gaussian. Distribution $p(y|\mathbf{x}, \mathbf{w})$ can take on non-Gaussian forms by setting it equal to a *mixture model* composed of a sum of Gaussian kernel functions (Everitt & Hand 1981). The input-dependent mean and variance of the distribution can be derived from the mixture model by MLE (Bishop 1994) and by MCMC (Dybowski 1997), but there is then the problem of defining an interval when a distribution is asymmetric or multi-modal.

1.6.4 Input noise

As mentioned at the end of section 1.2, one source of uncertainty in the output of an FNN is uncertainty in the input values. Some methods for estimating errors due to input noise have been reviewed by Press et al. (1992, pp. 666-670), and more recent work has been put forward by Tresp, Ahamad & Neuneier (1994) and Townsend & Tarassenko (1997).

Wright (1999) has taken a Bayesian approach to the problem in which the true but unobserved input $\mathbf{x}$ is perturbed by noise to give a noisy, observed input $\mathbf{z}$. If $\mathbf{z}_\circ$ denotes a new observed input, and $y_\circ$ is the associated target value, the predictive distribution $p(y_\circ|\mathbf{z}_\circ, \mathbf{S})$ can be expressed by integrating over the unknown $\mathbf{x}_\circ$:

$$p(y_\circ|\mathbf{z}_\circ, \mathbf{S}) = \int_{\mathbf{x}_\circ} p(y_\circ|\mathbf{x}_\circ, \mathbf{S}) p(\mathbf{x}_\circ|\mathbf{z}_\circ) d\mathbf{x}_\circ. \tag{1.40}$$

If there is a small level of Gaussian noise on the true input, eq.(1.40) leads to the following expression for the variance of $y_\circ$:

$$\widehat{\mathsf{Var}}(y_\circ|\mathbf{z}_\circ) = \beta^{-1} + \sigma_x^2 \mathbf{h}^\mathsf{T}(\mathbf{z}_\circ)\mathbf{h}(\mathbf{z}_\circ) + \mathbf{g}^\mathsf{T}(\mathbf{z}_\circ)\mathbf{A}^{-1}\mathbf{g}(\mathbf{z}_\circ), \tag{1.41}$$

which is similar to eq.(1.32) but with an additional term due to the introduction of noise to $\mathbf{x}$. The extra term consists of the variance σ_x^2 of $\mathbf{x}$ multipled by the squared partial derivative $\partial\widehat{\mu}_y(\mathbf{x};\mathbf{w})/\partial x_i$ evaluated at $\mathbf{x} = \mathbf{z}_\circ$.

If the assumptions leading to eq.(1.41) do not hold then $p(y_\circ|\mathbf{z}_\circ, \mathbf{S})$ is evaluated numerically, with MCMC used to estimate the inner integral in

$$p(y_\circ|\mathbf{z}_\circ, \mathbf{S}) = \int_{\mathbf{x}_\circ} p(\mathbf{x}_\circ|\mathbf{z}_\circ) \left[\int_{\mathbf{x},\mathbf{w}} p(y_\circ|\mathbf{x}_\circ, \mathbf{w}) p(\mathbf{x}, \mathbf{w}|\mathbf{S}) d\mathbf{x} d\mathbf{w} \right] d\mathbf{x}_\circ,$$

but a limitation of this approach is that $p(\mathbf{x}_\circ|\mathbf{z}_\circ)$ is required.

1.7 Conclusion

A neural network correctly trained with binary target values can estimate conditional class probabilities, and although it is possible to define a Bayesian confidence interval for a posterior probability, section 1.6.2 described why, from a Bayesian decision-theoretic viewpoint, such an interval is unnecessary. Furthermore, for the case when target values are real-valued, Bishop & Qazaz (1995) have demonstrated that variances estimated within the Bayesian framework can be less biased than those estimated by MLE; consequently, the Bayesian approach is preferred to MLE.

A problem with the Bayesian approach (whether by hybrid MCMC or via Gaussian approximations) is that implementing it tends to be more troublesome than MLE. These difficulties are restricted to neural networks and are due to the approximations used to obtain the mathematical formalism. When generalized linear models are used, the implementation becomes easy and straightforward because the approximations become exact. The accounting for parameter uncertainty in Bayesian methods works only if the computations are done reasonably exactly, and not by gross approximations. In contrast, MLE is easier to implement in terms of both stability of the algorithm and speed of convergence (as measured by CPU time). Of the MLE-based methods, the bootstrap method has been reported to provide more accurate confidence intervals than the delta method and more accurate prediction intervals than the Nix-Weigend method. Nevertheless, the advantages of the Bayesian framework suggest that efforts should be made toward developing stable techniques in this area so that Bayesian prediction and confidence intervals can be obtained reliably.

Acknowledgements

We thank Cazhaow Qazaz and Brain Ripley for their constructive comments during

the production of this chapter.

Notes

1 We have used the expression *regression function* instead of *regression model* as the former refers to an estimated relationship between $\mu_y(\mathbf{x})$ and $\mathbf{x}$ (Robbins & Munro 1951), whereas the latter refers to a family of possible relationships.

2 Symbol p will be used both for probability density functions and probability mass functions, the correct meaning being understood from the context in which it is used. For those readers unfamiliar with probability theory, we recommend Wonnacott & Wonnacott (1985, pp. 52–150) followed by Ross (1988).

3 Although confidence intervals with equal tails are the most common form, there are other possibilities (Barnett 1982, pp 172–176).

4 The predictive distribution for $v^{(N+1)}$ is given by

$$p(v^{(N+1)} = 1|\bar{v}, N) = (\bar{v}N + 1)/(N + 1).$$

5 Both linear and logistic regression models belong to the class of models called *generalized linear models* (Dobson 1990). These have the general form

$$g(\mu_y(\mathbf{x}; \mathbf{w})) = w_0 + \sum_{i=1}^{d} w_i x_i,$$

where g is the *link function*. In simple linear regression, $g(a) = a$, whereas in logistic regression, $g(a) = \mathrm{logit}(a)$.

6 A clear account of vectors and matrices is provided by Anton (1984).

7 The Hessian matrix and calculation of its inverse $\mathbf{H}^{-1}$ are discussed by Bishop (1995, pp. 150-160).

8 If noise variance σ_ε^2 is *not* independent of $\mathbf{x}$ then $\mathbf{H}/\sigma_\varepsilon^2$ in eq.(1.18) is replaced by a matrix $\mathbf{G}$ defined by (Penny & Roberts 1997)

$$\mathbf{G}_{k,l} = \sum_{i=1}^{N} \frac{1}{\sigma_\varepsilon^2(\mathbf{x}^{(i)})} \left\{ \frac{\partial \widehat{\mu}_y(\mathbf{x}^{(i)}; \widehat{\mathbf{w}})}{\partial \widehat{w}_k} \frac{\partial \widehat{\mu}_y(\mathbf{x}^{(i)}; \widehat{\mathbf{w}})}{\partial \widehat{w}_l} + \left[y^{(i)} - \widehat{\mu}_y(\mathbf{x}^{(i)}; \widehat{\mathbf{w}}) \right] \frac{\partial^2 \widehat{\mu}_y(\mathbf{x}^{(i)}; \widehat{\mathbf{w}})}{\partial \widehat{w}_k \partial \widehat{w}_l} \right\}.$$

9 Maximum likelihood is referred to as *maximum penalized likelihood* if the error function is regularized.

10 The *bootstrap* method should not be confused with the *jacknife* or *cross-validation* (Efron & Gong 1983).

11 The number of bootstrap samples needed for reliable estimates depends on the type of statistics we are after. In the case of estimating the mean of a random variable, a relatively small number of samples are required, whilst for estimating variance, a larger number is needed since the estimate of the variance is more sensitive to noise.

Bibliography

Anton, H. (1984), *Elementary Linear Algebra*, 4th edn, John Wiley, New York.

Barnett, V. (1982), *Comparative Statistical Inference*, 2nd edn, Wiley, Chichester.

Baxt, W.G. (1995), 'Application of artificial neural networks to clinical medicine', *Lancet* **346**, 1135–1138.

Baxt, W.G. & H. White (1995), 'Bootstrapping confidence intervals for clinical input variable effects in a network trained to identify the presence of acute myocardial infarction', *Neural Computation* **7**, 624–638.

Bishop, C.M. (1994), Mixture density networks, Technical report NCRG/4288, Neural Computing Research Group, Aston University.

Bishop, C.M. (1995), *Neural Networks for Pattern Recognition*, Clarendon Press, Oxford.

Bishop, C.M. & C.S. Qazaz (1995), Bayesian inference of noise levels in regression, *in* F.Fogelman-Soulie & P.Gallineri, eds, 'Proceedings of the International Conference on Artificial Neural Networks 1995 (ICANN95)', EC2 & Cie, Paris, pp. 59–64.

Breiman, L. (1996), 'Bagging predictors', *Machine Learning* **24**, 123–140.

Collett, D. (1991), *Modelling Binary Data*, Chapman & Hall, London.

de Bruijn, N.G. (1970), *Asymptotic Methods in Analysis*, North-Holland, Amsterdam.

Dobson, A.J. (1990), *An Introduction to Generalized Linear Models*, Chapman & Hall, London.

Draper, D. (1995), 'Assessment and propagation of model uncertainty (with discussion)', *Journal of the Royal Statistical Society. Series B* **57**(1), 45–97.

Dudewicz, E.J. & S.N. Mishra (1988), *Modern Mathematical Statistics*, John Wiley, Ney York.

Dybowski, R. (1997), Assigning confidence intervals to neural network predictions, Technical report, Division of Infection (St Thomas' Hospital), King's College London.

Dybowski, R. & V. Gant (1995), 'Artificial neural networks in pathology and medical laboratories', *Lancet* **346**, 1203–1207.

Efron, B. (1979), 'Bootstrap methods: another look at the jacknife', *Annals of Statistics* **7**, 1–26.

Efron, B. & G. Gong (1983), 'A leisurely look at the bootstrap, the jacknife, and cross-validation', *The American Statistician* **37**(1), 36–48.

Efron, B. & R.J. Tibshirani (1993), *An Introduction to the Bootstrap*, Chapman & Hall, New York.

Everitt, B.S. & D.J. Hand (1981), *Finite Mixture Distributions*, Chapman & Hall, London.

Geisser, S. (1993), *Predictive Inference: An Introduction*, Chapman & Hall, New York.

Gemen, S., E. Bienenstock & R. Doursat (1992), 'Neural networks and the bias/variance dilemma', *Neural Computation* **4**, 1–58.

Gilks, W.R., S. Richardson & D.J. Spiegelhalter, eds (1996), *Markov Chain Monte Carlo in Practice*, Chapman & Hall, London.

Goldberg, P.W., C.K.I. Williams & C.M. Bishop (1998), Regression with input-dependent noise: A Gaussian process treatment, Technical report NCRG/98/002, Neural Computing Research Group, Aston University.

Hauck, W.W. (1983), 'A note on confidence bands for the logistic response curve', *The American Statistician* **37**(2), 158–160.

Heskes, T. (1997), Practical confidence and prediction intervals, *in* M.Mozer, M.Jordan & T.Petsche, eds, 'Advances in Neural Information Processing Systems 9', MIT Press, Cambridge, MA, pp. 176–182.

Hinton, G.E. (1989), 'Connectionist learning procedures', *Artificial Intelligence* **40**, 185–234.

Hogg, R.V. & A.T. Craig (1995), *Intoduction to Mathematical Statistics*, Prentice Hall, Englewood Cliffs, NJ.

Holst, H., M. Ohlsson, C. Peterson & L. Edenbrandt (1998), 'Intelligent computer reporting 'lack of experience': A confidence measure for decision support systems', *Clinical Physiology* **18**(2), 139–147.

Hosmer, D.W. & S. Lemeshow (1989), *Applied Logistic Regression*, Wiley, New York.

Kyburg, H.E. & M. Pittarelli (1996), 'Set-based Bayesianism', *IEEE Transactions on Systems, Man and Cybernetics* **26**(3), 324–339.

Lee, P.M. (1997), *Bayesian Statistics: An Introduction*, 2nd edn, Arnold, London.

MacKay, D.J.C. (1991), Bayesian Methods for Adaptive Models, Ph.D. thesis, California Institute of Technology, Pasadena, CA.

MacKay, D.J.C. (1992*a*), 'The evidence framework applied to classification networks', *Neural Computation* **4**(5), 720–736.

MacKay, D.J.C. (1992*b*), 'A practical Bayesian framework for back-propagation networks', *Neural Computation* **4**(3), 448–472.

Neal, R.M. (1996), *Bayesian Learning for Neural Networks*, Lecture Notes in Statistics Series No. 118, Springer, New York.

Nix, D.A. & A.S. Weigend (1995), Learning local error bars for nonlinear regression, *in* G.Tesauro, D.Touretzky & T.Leen, eds, 'Advances in Neural Information Processing Systems 7 (NIPS*94)', MIT Press, Cambridge, MA, pp. 489–496.

Penny, W.D. & S.J. Roberts (1997), Neural network predictions with error bars, Research report TR-97-1, Department of Electrical and Electronic Engineering, Imperial College, London.

Press, W.H., S.A. Teukolsky, W.T. Vetterling & B.P. Flannery (1992), *Numerical Recipes in C*, 2nd edn, Cambridge University Press, Cambridge.

Qazaz, C.S., C.K.I. Williams & C.M. Bishop (1996), An upper bound on the Bayesian error bars for generalised linear regression, Technical report NCRG/96/005, Neural Computing Research Group, Aston University.

Robbins, H. & S. Munro (1951), 'A stochastic approximation method', *Annals of Mathematical Statistics* **22**, 400–407.

Ross, S. (1988), *A First Course in Probability*, 3rd edn, Macmillan, New York.

Santner, T.J. & D.E. Duffy (1989), *The Statistical Analysis of Discrete Data*, Springer-Verlag, New York.

Spiegelhalter, D.J. & S.L. Lauritzen (1990), 'Sequential updating of conditional probabilities on directed graphical structures', *Networks* **20**, 579–605.

Tibshirani, R. (1996), 'A comparison of some error estimates for neural network models', *Neural Computation* **8**, 152–163.

Townsend, N.W. & L. Tarassenko (1997), Estimation of error bounds for RBF networks, *in* 'Proceedings of the 5th International Conference on Artificial Neural,Networks', IEE, Stevenage UK, pp. 227–232.

Tresp, V., S. Ahamad & R. Neuneier (1994), Training neural networks with deficient data, *in* J.Cowan, G.Tesauro & J.Alspector, eds, 'Neural Information Processing Systems 6', Morgan Kaufmann, pp. 128–135.

Williams, C.K.I. (1999), Prediction with Gaussian processes: From linear regression to linear prediction and beyond, *in* M.Jordan, ed., 'Learning in Graphical Models', MIT Press, Cambridge MA, pp. 599–621.

Wonnacott, R.J. & T.H. Wonnacott (1985), *Introductory Statistics*, 4th edn, John Wiley, New York.

Wonnacott, T.H. & R.J. Wonnacott (1981), *Regression: A Second Course in*

Statistics, John Wiley, New York.

Wright, W.A. (1999), Neural network regression with input uncertainty, Technical report NCRG/99/008, Neural Computing Research Group, Aston University.