Bayesian Methods for Adaptive Models

Thesis by

David J.C. MacKay

In Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy

California Institute of Technology
Pasadena, California

(Submitted December 10, 1991)

Acknowledgements

During the last three years, my work has benefited greatly from discussions with Ron Benson, John Bridle, Peter Cheeseman, Sidney Fels, Steve Gull, Andreas Herz, John Hopfield, Doug Kerns, Allen Knutsen, David Koerner, Mike Lewicki, Tom Loredo, Steve Luttrell, Ronny Meir, Ken Miller, Marcus Mitchell, Radford Neal, Steve Nowlan, David Robinson, Ken Rose, Sibusiso Sibisi, John Skilling, Haim Sompolinsky and Nick Weir. The comments of referees on chapter 4 were also helpful.

I especially thank my advisor, John Hopfield, for his support, criticism and advice.

I am very grateful to Brooke Anderson, Dawei Dong, Brian Fox and Tom Tromey for their expert management of the Hopfield and CNS computers.

I would like to thank Dr. R. Goodman and Dr. P. Smyth for funding my trip to Maxent 90, where I learnt the final tools needed for this research.

This work was supported by a Caltech Fellowship and a Studentship from SERC, UK.

I hope that the ideas in this dissertation will only be used towards peaceful ends.

Current Address

Cavendish Laboratory,
Madingley Road,
Cambridge,
CB3 0HE, United Kingdom.

`mackay@mrao.cam.ac.uk`

Second Edition

This is the second edition of my thesis. Some typographical errors have been corrected, and a small number of clarifications of the text have been made.

The postscript files for this document may be obtained by anonymous ftp from `mraos.ra.phy.cam.ac.uk (131.111.48.8)` in `pub/mackay`.

Typeset with LaTeX.

Bayesian Methods for Adaptive Models

Thesis by David John Cameron MacKay

In Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy

California Institute of Technology
Pasadena, California

1992
(Submitted December 10, 1991)

Advisor: Prof. J.J. Hopfield

Abstract

The Bayesian framework for model comparison and regularisation is demonstrated by studying interpolation and classification problems modelled with both linear and non–linear models. This framework quantitatively embodies 'Occam's razor'. Over–complex and under–regularised models are automatically inferred to be less probable, even though their flexibility allows them to fit the data better.

When applied to 'neural networks', the Bayesian framework makes possible (1) objective comparison of solutions using alternative network architectures; (2) objective stopping rules for network pruning or growing procedures; (3) objective choice of type of weight decay terms (or regularisers); (4) on–line techniques for optimising weight decay (or regularisation constant) magnitude; (5) a measure of the effective number of well–determined parameters in a model; (6) quantified estimates of the error bars on network parameters and on network output. In the case of classification models, it is shown that the careful incorporation of error bar information into a classifier's predictions yields improved performance.

Comparisons of the inferences of the Bayesian framework with more traditional cross–validation methods help detect poor underlying assumptions in learning models.

The relationship of the Bayesian learning framework to 'active learning' is examined. Objective functions are discussed which measure the expected informativeness of candidate data measurements, in the context of both interpolation and classification problems.

The concepts and methods described in this thesis are quite general and will be applicable to other data modelling problems whether they involve regression, classification or density estimation.

Contents

List of Figures

Chapter 1

Summary

1.1 The need for Occam's razor

There are countless problems in science, statistics and technology which require that, given a limited data set, preferences be assigned to alternative models of differing complexities. For example, two alternative hypotheses accounting for planetary motion are the geocentric 'epicyclic' model, and the simpler Copernican model of the solar system. In the less theologically contentious but similar problem of fitting a curve to data, alternative models assign different functional forms to the curve, for example 'a linear function with two free parameters', 'a quadratic with three', or 'a cubic function with four parameters'. It would be nice if we could just rank the models by how well they 'fit' the data, but it is a familiar difficulty that a more complex model typically fits the data better: when we fit a curve to data, a quadratic curve with three parameters can always fit the data better than a linear model with two parameters, and a polynomial with a hundred terms fits the data even better; preferring the 'best fit' model leads us to choose implausibly detailed and over–parameterised models, which interpolate and generalise poorly. 'Occam's razor' is the principle that states that unnecessarily complex models should not be preferred to simpler ones. How can we quantify this intuitive principle so as to make it an objective part of our modelling method?

Bayesian probability theory provides a framework for inductive inference which has been called 'common sense reduced to calculation'; it is a poorly known fact that Bayesian methods actually embody Occam's razor automatically and quantitatively [26, 38]. Bayesian model comparison is the central theme of this thesis. In particular, the power of the Bayesian Occam's razor is demonstrated on 'neural networks'. Neural networks are novel modelling tools capable of 'learning from examples'. These currently popular models are notorious for their lack of an objective grounding; the main goal of this thesis is to provide an objective and practical framework for the use of neural network techniques by applying the methods of Bayesian model comparison. In the process several enhancements to current neural network methods arise.

1.2 What is Bayesian modelling?

Bayesian methods for inductive inference were first developed in detail early this century by the Cambridge geophysicist, Sir Harold Jeffreys [38]. At that time, Jeffreys' ideas were opposed by Fisher and others, and since then a debate has persisted between the 'orthodox' view of statistics and the minority Bayesian camp. I will not dwell here on the details of

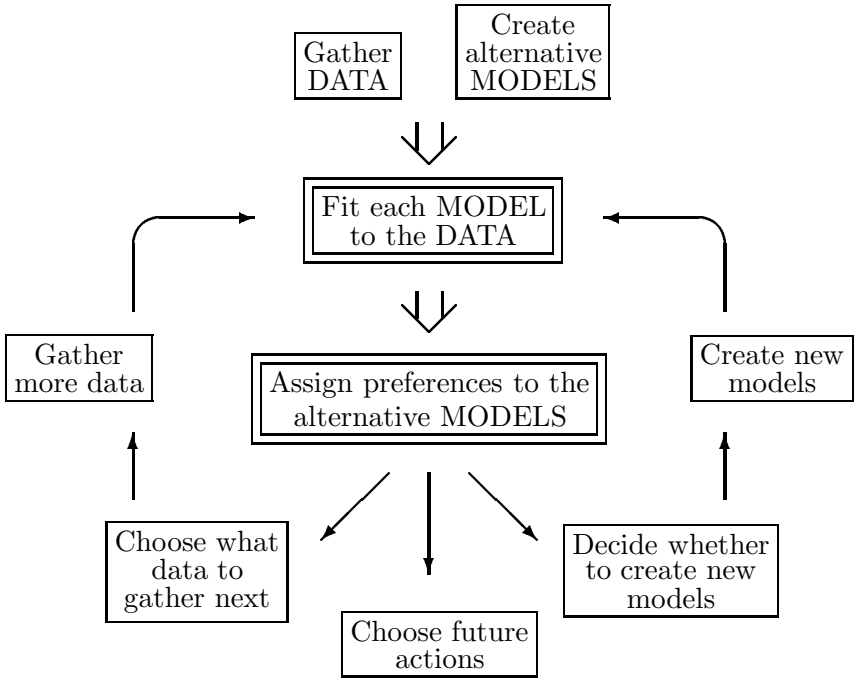

Figure 1.1: **Abstraction of the data modelling process.**
The two central boxes are the *inference* steps, where Bayesian methods can be applied.

the philosophical argument, which goes deep down to the meaning of a probability [17, 36]; rather, this thesis will demonstrate that it is possible using Bayesian methods to solve problems in neural networks which have otherwise been found laborious or impossible. Since the 1960's, the Bayesian minority has been steadily growing, especially in the fields of economics [89] and pattern processing [20]. At this time, the state of the art for the problem of speech recognition is a Bayesian technique (Hidden Markov Models), and the best image reconstruction algorithms are also based on Bayesian probability theory (Maximum Entropy), but Bayesian methods are still viewed with mistrust by the orthodox statistics community; the framework for model comparison is especially poorly known, even to most people who call themselves Bayesians. This thesis therefore takes some time to thoroughly review the flavour of Bayesianism that I am using. To some, the word Bayesian denotes a decision strategy that minimises the expectation of a cost [24]; to others, a Bayesian is someone who tries to incorporate prior knowledge into their inference and decision process [8]. In fact, according to Good, there are 46656 varieties of Bayesian![1] This thesis presents a flavour of Bayesianism in which decisions are not involved. Inference and decision are cleanly separated. The terms 'Bayes risk' and 'Bayes optimal' are not in the vocabulary of this thesis. The genealogy of this flavour is Laplace–Jeffreys–Cox–Jaynes–Gull [80, 38, 17, 36, 26]. A further difference between this approach and other work known as Bayesian is that the emphasis is on inverse rather than forward probability. Forward probability uses probabilities and priors, but it does not make use of Bayes' rule. Forward probability is used for example to evaluate the typical performance of a modelling procedure averaged over different data sets from a defined ensemble [82, 32]. Here the philosophy is, using inverse probability, to evaluate the relative plausibilities of several alternative *models* in the light of the *single* data set that we actually observe.

[1] Good was unaware of the Bayesian Occam's razor.

Where inference fits into the data modelling process

Figure 1.1 illustrates an abstraction of the data modelling process; this summary applies for example to the tasks of fitting a curve to data, reconstructing a blurred image, and making an automatic pattern recognition system; the figure is also descriptive of the general scientific method.

We start by gathering data and creating models to account for those data. There are then two levels of *inference*, which are marked by the double–framed boxes. At the first level, 'fitting each model to the data', the task is to infer what the free parameters of each model might be given the data. The second level of inference is the task of model comparison. Here, we wish to rank how plausible the alternative models are in the light of the data.

Having fitted the models and compared them, we can then decide to gather more data or to invent new models for the data, and we can repeat the inference process. We can also use the knowledge we have gained from the data to make decisions about our future actions in the world.

Bayesian methods can be used to solve the two inductive inference problems, which are the two central boxes in the figure; the other tasks in the modelling process are not directly addressed by Bayes' rule, which applies to inductive inference problems only. The first level of inference, fitting each model to the data, is usually a straightforward task, and differences between Bayesian and non–Bayesian solutions are often not pronounced at this level. This thesis will especially emphasise the second level of inference, the task of model comparison. This inference problem is not straightforward because a quantitative Occam's razor is needed to penalise over–complex models. The other boxes in this diagram will also be visited during the thesis.

Bayes' rule

The fundamental concept of Bayesian analysis is that the plausibilities of alternative hypotheses are represented by probabilities, and inference is performed by evaluating those probabilities. Suppose that we have a collection of models, $\mathcal{H}_1, \mathcal{H}_2, \ldots \mathcal{H}_L$ competing to account for the data we gather. Our initial beliefs about the relative plausibility of these models are quantified by a list of probabilities, $P(\mathcal{H}_1), P(\mathcal{H}_2), \ldots P(\mathcal{H}_L)$, which sum to 1. Each model $\mathcal{H}_i$ makes predictions about how likely different data sets 'D' are, if that model is true. These predictions are described by a probability distribution $P(D|\mathcal{H}_i)$ ('the probability of D *given* $\mathcal{H}_i$'). When we observe the actual data D, **Bayes' rule** describes how we should update our beliefs in the models in the light of the data. The plausibility of model $\mathcal{H}_i$, given that we have observed D, written $P(\mathcal{H}_i|D)$, is obtained by multiplying together two quantities: first, $P(\mathcal{H}_i)$, *i.e.*, how plausible we thought $\mathcal{H}_i$ was *before* the data arrived; and second, $P(D|\mathcal{H}_i)$, *i.e.*, how much the model $\mathcal{H}_i$ *predicted* the data. In symbols, Bayes' rule is written:

$$P(\mathcal{H}_i|D) = \frac{P(\mathcal{H}_i)P(D|\mathcal{H}_i)}{P(D)}.$$

The denominator $P(D)$ is a normalising constant which makes our final beliefs $P(\mathcal{H}_i|D)$ add up to 1. A Bayesian addresses any inference problem by using this equation. The hard line Bayesian position is that the Cox axioms [17] prove that consistent inference can *only* be Bayesian, and no other inference methods should be used, on pain of inconsistency [75]. However, I will develop the more moderate position that the Bayesian method is an important tool which should be used alongside other pragmatic modelling tools. I will

demonstrate that the simultaneous application of Bayesian and non–Bayesian methods leads to insights that could not be obtained by using either tool alone.

1.3 What are neural networks and why do they need Occam's razor?

Research in neural–like networks is motivated by the observation that the brain has a 'connectionist' computational architecture: the brain is composed of many simple devices (neurons) which are massively interconnected with each other; the computational abilities of the brain are an 'emergent phenomenon' arising from the cooperative interactions of these simple components. Workers in the field of neural networks create novel connectionist devices so as to try and understand 'how the brain does it', and to try to create new and useful tools for such tasks as speech recognition, character recognition, and robotics.

The most popular neural network algorithm is 'backpropagation', which is capable of 'learning from examples' [66]. In this case, a neural network can be viewed as a black box which produces an output when we give it an input. How the output depends on the input is controlled by some tens or thousands of knobs on the black box, which we, the teacher, are able to twiddle. The object of the 'learning' process is to adjust these knobs so as to get the black box to give a desired output in response to each input. What is inside the black box is not essential to this discussion: usually it contains a network of simple 'neurons' feeding from the inputs to the outputs, and the 'knobs' are the strengths of the 'synapses' between the 'neurons'.

Imagine that what we feed to the inputs of the black box is a simple encoding of a piece of English text; and imagine that we want the outputs of the black box to be the pronunciation of that piece of text, in a simple code we have defined. When we present an untrained black box with a piece of English text, its outputs are very likely to be complete garbage, compared with the coded pronunciation that we wanted it to produce. What we would like to do is adjust the knobs on the black box a little, so that the next time we give the same piece of text as input, the output of the box will be *a little closer* to what it should have been. The backpropagation learning algorithm is a prescription for how to tweak all the knobs on the black box to achieve precisely this goal. (Backpropagation performs gradient descent on the error function.) Now the perhaps surprising outcome of this procedure is that after repeated 'training' on a dictionary of 50,000 English words, a black box consisting of 200 'neurons' can learn not only to pronounce correctly a large fraction of the words it was trained on, but also to perform equally well on other words which were not in the training set. Thus the device is able to extract the underlying structure in the examples it was trained on and 'generalise' from them.

The backpropagation algorithm has been applied to many other tasks (the text pronunciation example above is one of the earliest successes), and a performance equalling the ability of human experts is often obtained. Recently, especially impressive results have been obtained for adaptive optics [4].

However, the performance of these algorithms depends on a considerable number of design choices, most of which are currently made by rules of thumb and trial and error. For example, in designing the neural network for text pronunciation, one has to decide how many 'neurons' there should be in the architecture of the black box, how they should be connected to each other, and what constraints should be imposed on the parameters of the network. The problem of Occam's razor rears its head repeatedly when we try to make these design choices because a more complex and unconstrained neural network will nearly

always learn the examples in the training set better than a simpler one; however the simpler neural network may actually be a better model of the problem, and generalise better to new examples.

The fact that we cannot use the performance on the training set to choose between different solutions would not matter if we had plenty of data and limitless computational resources: we could generate solutions using thousands of different models with different complexities and rank them by evaluating the test error on some reserved test data. But since we have limited resources we would like to be able to use *all* our data both to fit all the models and also to rank them. We would furthermore like to find a technique for automatically optimising the choice of model design, without having to perform massive computational searches through 'design space'.

The Bayesian framework presented in this thesis satisfies these desiderata.

Overview

The thesis consists of four papers. The first paper (Chapter 2) reviews in detail the Bayesian framework for model comparison and regularisation due to Gull and Skilling, by studying the problem of interpolating a noisy data set with traditional linear models. This chapter demonstrates that Bayesian methods do indeed embody Occam's razor in a consistent, intuitive and quantitative way.

In the second paper (Chapter 3) this framework is applied to 'neural networks', and it is demonstrated that (at least for the toy problem studied) Bayesian probability theory chooses between alternative solutions found using networks with different architectures in a way that succesfully embodies Occam's razor. Another enhancement to neural network training methods concerns regularisation. Neural networks sometimes perform poorly because the parameters ('weights') in the network blow up to implausibly large values in order to fit the details of the training set. To prevent this it is popular to use a procedure called 'weight decay' during the training. However, no objective procedure previously existed for setting the weight decay *rate* (apart from the computationally expensive option of testing multiple decay rates in parallel experiments). The Bayesian framework for neural network learning yields a simple prescription for optimising the weight decay rate, which is interpreted as a regularisation constant. This prescription can be easily approximated and implemented 'on line', and it may be one of the most useful practical tools to emerge from this research. In Chapter 3 we also see how the combination of Bayesian and non–Bayesian model assessment techniques can draw attention to defects in our hypothesis space, helping us traverse the loop to the right of figure 1.1, in which we invent new models.

In the third paper (Chapter 4), information–based utility functions are discussed for the purpose of data selection, the left–hand loop in figure 1.1. The evaluation of data utility is a problem relevant to a scientist whose data measurements are expensive, and to an autonomous robot which has to decide where to explore next so as to satisfy a pre–programmed curiosity about its environment; we also need to evaluate data utility in situations where data is so abundant that we have to decide which data to throw away. The information–based criteria derived in this chapter have promising properties, but I do not believe that they are the final solution to the data selection problem, because artefacts may result when these criteria are applied to poor models.

The fourth paper (Chapter 5) applies the methods developed in the first three papers to neural networks solving classification problems, rather than regression problems. One of the simplest but most important results in this chapter is a demonstration that careful

incorporation of error bar information into the outputs of a classifier can give improved predictions. As in Chapter 3, the Bayesian Occam's razor does its job surprisingly well.

Chapter 6 is a short note extending the framework of Chapters 2 and 3 to allow modelling of an input–dependent noise level. A maximum likelihood solution to this problem would have singularities where the interpolant fits the data exactly; the Bayesian solution naturally avoids these problems.

In the final chapter I reflect on the strengths and weaknesses of the Bayesian approach to adaptive modelling, and the open questions and frontiers facing this framework.

Relevance to Biology

This work is not intended to shed any direct light on the functioning of biological neural networks. But it is clear that biological neural networks have solved the Occam's razor problem — we are expert adaptive modelling systems. I believe that if we are ever to understand the brain, a prerequisite will be that we should understand the problems that it has solved. We need to understand how to model, and how to infer. Of course, I do not expect that the brain embodies any of the equations in this thesis; I am sure that Nature has found far more elegant solutions to these problems. But I hope that the Bayesian normative theory of learning will serve as a guide in trying to elucidate how learning is performed by natural systems.

Chapter 2

Bayesian Interpolation

Abstract

Although Bayesian analysis has been in use since Laplace, the Bayesian method of *model–comparison* has only recently been developed in depth. In this chapter, the Bayesian approach to regularisation and model–comparison is demonstrated by studying the inference problem of interpolating noisy data. The concepts and methods described are quite general and can be applied to many other data modelling problems.

Regularising constants are set by examining their posterior probability distribution. Alternative regularisers (priors) and alternative basis sets are objectively compared by evaluating the *evidence* for them. 'Occam's razor' is automatically embodied by this process.

The way in which Bayes infers the values of regularising constants and noise levels has an elegant interpretation in terms of the effective number of parameters determined by the data set. This framework is due to Gull and Skilling.

2.1 Data modelling and Occam's razor

In science, a central task is to develop and compare models to account for the data that are gathered. In particular this is true in the problems of learning, pattern classification, interpolation and clustering. Two levels of **inference** are involved in the task of data modelling (figure 2.1). At the first level of inference, we assume that one of the models that we invented is true, and we fit that model to the data. Typically a model includes some free parameters; fitting the model to the data involves inferring what values those parameters should probably take, given the data. The results of this inference are often summarised by the most probable parameter values and error bars on those parameters. This is repeated for each model. The second level of inference is the task of model comparison. Here, we wish to compare the models in the light of the data, and assign some sort of preference or ranking to the alternatives.[1]

[0] Chapter 2 of Ph.D. thesis 'Bayesian Methods for Adaptive Models' by David MacKay, California Institute of Technology, submitted December 10 1991.

[1] Note that both levels of *inference* are distinct from *decision theory*. The goal of inference is, given a defined hypothesis space and a particular data set, to assign probabilities to hypotheses. Decision theory typically chooses between alternative actions on the basis of these probabilities so as to minimise the expectation of a 'loss function'. This chapter concerns inference alone and no loss functions or utilities are involved.

Another misconception concerns the relationship between model comparison and model choice. In emphasising the Bayesian method of model comparison I do not mean to imply that the correct action is to choose the most probable model. The 'right way' to make Bayesian predictions is to integrate over our model space.

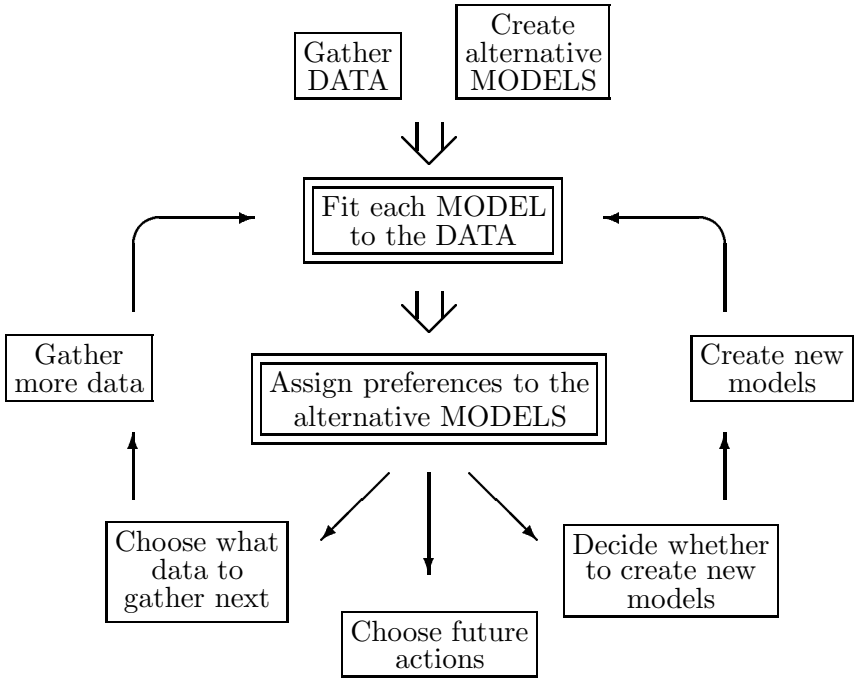

Figure 2.1: **Where Bayesian inference fits into the data modelling process.** This figure illustrates an abstraction of the part of the scientific process in which data is collected and modelled. In particular, this figure applies to pattern classification, learning, interpolation, etc.. The two double–framed boxes denote the two steps which involve *inference*. It is only in those two steps that Bayes' rule can be used. Bayes does not tell you how to invent models, for example.
The first box, 'fitting each model to the data', is the task of inferring what the model parameters might be given the model and the data. Bayes may be used to find the most probable parameter values, and error bars on those parameters. The result of applying Bayes to this problem is often little different from the answers given by orthodox statistics.
The second inference task, model comparison in the light of the data, is where Bayes is in a class of its own. This second inference problem requires a quantitative Occam's razor to penalise over–complex models. Bayes can assign objective preferences to the alternative models in a way that automatically embodies Occam's razor.

For example, consider the task of interpolating a noisy data set. The data set could be interpolated using a splines model, using radial basis functions, using polynomials, or using feedforward neural networks. At the first level of inference, we take each model individually and find the best fit interpolant for that model. At the second level of inference we want to rank the alternative models and state for our particular data set that, for example, 'splines are probably the best interpolation model', or 'if the interpolant is modelled as a polynomial, it should probably be a cubic'.

Bayesian methods are able consistently and quantitatively to solve both these inference tasks. There is a popular myth that states that Bayesian methods only differ from orthodox (also known as 'frequentist' or 'sampling theory') statistical methods by the inclusion of subjective priors which are arbitrary and difficult to assign, and usually don't make much difference to the conclusions. It is true that at the first level of inference, a Bayesian's results will often differ little from the outcome of an orthodox attack. What is not widely

We may however sometimes make model choices for reasons of computational economy, or because only a few models are needed to give a sufficiently accurate approximation to the ideal Bayesian solution.

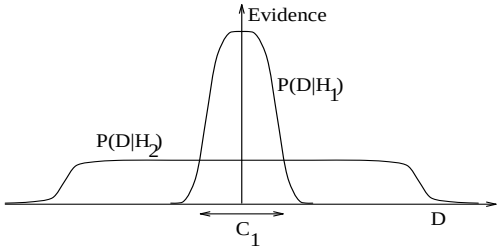

Figure 2.2: **Why Bayes embodies Occam's razor**
This figure gives the basic intuition for why complex models are penalised. The horizontal axis represents the space of possible data sets D. Bayes' rule rewards models in proportion to how much they *predicted* the data that occurred. These predictions are quantified by a normalised probability distribution on D. In this paper, this probability of the data given model $\mathcal{H}_i$, $P(D|\mathcal{H}_i)$, is called the evidence for $\mathcal{H}_i$.
A simple model $\mathcal{H}_1$ makes only a limited range of predictions, shown by $P(D|\mathcal{H}_1)$; a more powerful model $\mathcal{H}_2$, that has, for example, more free parameters than $\mathcal{H}_1$, is able to predict a greater variety of data sets. This means however that $\mathcal{H}_2$ does not predict the data sets in region $\mathcal{C}_1$ as strongly as $\mathcal{H}_1$. Assume that equal prior probabilities have been assigned to the two models. Then if the data set falls in region $\mathcal{C}_1$, the *less powerful* model $\mathcal{H}_1$ will be the *more probable* model.

appreciated is how Bayes performs the second level of inference. It is here that Bayesian methods are totally different from orthodox sampling theory methods. Indeed, when regression and density estimation are discussed in most statistics texts (for example [24]), the task of model comparison is virtually ignored; no general orthodox method exists for solving this problem.

Model comparison is a difficult task because it is not possible simply to choose the model that fits the data best: more complex models can always fit the data better, so the maximum likelihood model choice would lead us inevitably to implausible over–parameterised models which generalise poorly. 'Occam's razor' is the principle that states that unnecessarily complex models should not be preferred to simpler ones. Bayesian methods automatically and quantitatively embody Occam's razor [26, 38], without the introduction of ad hoc penalty terms. Complex models are automatically self–penalising under Bayes' rule. Figure 2.2 gives the basic intuition for why this should be expected; the rest of this chapter will explore this property in depth.

Bayesian methods, simultaneously conceived by Bayes [6] and Laplace [80], were first laid out in depth by the Cambridge geophysicist Sir Harold Jeffreys [38]. The logical basis for the Bayesian use of probabilities as measures of plausibility was subsequently established by Cox [17], who proved that consistent inference in a closed hypothesis space can be mapped onto probabilities. For a general review of Bayesian philosophy the reader is encouraged to read the excellent papers by Jaynes and Loredo [36, 47], and the recently reprinted text of Box and Tiao [13]. Since Jeffreys, the emphasis of most Bayesian probability theory has been 'to formally utilize prior information' [8], *i.e.*, to perform inference in a way that makes explicit the prior knowledge and ignorance that we have, which orthodox methods omit. However, Jeffreys' work also laid the foundation for Bayesian model comparison, which does not involve an emphasis on prior information, but rather emphasises getting maximal information from the data. Jeffreys applied this theory to simple model comparison problems in geophysics, for example testing whether a single additional parameter is justified by the data. Since the 1960s, Jeffreys' model comparison methods have been applied and

extended in the economics literature [89] and by a small number of statisticians [10, 11, 12]. Only recently has this aspect of Bayesian analysis been further developed and applied to more complex problems in other fields.

This chapter will review Bayesian model comparison, 'regularisation', and noise estimation, by studying the problem of interpolating noisy data. The Bayesian framework I will describe for these tasks is due to Gull and Skilling [26, 27, 29, 70, 74], who have used Bayesian methods to achieve the state of the art in image reconstruction. The same approach to regularisation has also been developed in part by Szeliski [81]. Bayesian model comparison is also discussed by Smith and Spiegelhalter [77] and by Bretthorst [14], who has used Bayesian methods to push back the limits of NMR signal detection. The same Bayesian theory underlies the unsupervised classification system, AutoClass [31]. The fact that Bayesian model comparison embodies Occam's razor has been rediscovered by Kashyap in the context of modelling time series [40]; his paper includes a thorough discussion of how Bayesian model comparison is different from orthodox 'Hypothesis testing'. One of the earliest applications of these sophisticated Bayesian methods of model comparison to real data is by Patrick and Wallace [60]; in this fascinating paper, competing models accounting for megalithic stone circle geometry are compared within the description length framework, which is equivalent to Bayes. It is pleasing to note the current appearance of an increasing number of publications using Bayesian model comparison [37, 53].

As the quantities of data collected throughout science and engineering continue to increase, and the computational power and techniques available to model that data also multiply, I believe Bayesian methods will prove an ever more important tool for refining our modelling abilities. I hope that this review will help to introduce these techniques to the 'neural' modelling community. Chapter 3 will demonstrate how these techniques can be fruitfully applied to backpropagation neural networks. Chapter 4 will show how this framework relates to the task of selecting where next to gather data so as to gain maximal information about our models.

2.2 The evidence and the Occam factor

Let us write down Bayes' rule for the two levels of inference described above, so as to see explicitly how Bayesian model comparison works. Each model $\mathcal{H}_i$ ($\mathcal{H}$ stands for 'hypothesis') is assumed to have a vector of parameters $\mathbf{w}$. A model is defined by its functional form and two probability distributions: a 'prior' distribution $P(\mathbf{w}|\mathcal{H}_i)$ which states what values the model's parameters might plausibly take; and the predictions $P(D|\mathbf{w}, \mathcal{H}_i)$ that the model makes about the data D when its parameters have a particular value $\mathbf{w}$. Note that models with the same parameterisation but different priors over the parameters are therefore defined to be different models.

1. **Model fitting.** At the first level of inference, we assume that one model $\mathcal{H}_i$ is true, and we infer what the model's parameters $\mathbf{w}$ might be given the data D. Using Bayes' rule, the **posterior probability** of the parameters $\mathbf{w}$ is:

$$P(\mathbf{w}|D, \mathcal{H}_i) = \frac{P(D|\mathbf{w}, \mathcal{H}_i)P(\mathbf{w}|\mathcal{H}_i)}{P(D|\mathcal{H}_i)}. \tag{2.1}$$

In words:

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Evidence}}.$$

The normalising constant $P(D|\mathcal{H}_i)$ is commonly ignored, since it is irrelevant to the first level of inference, *i.e.*, the choice of $\mathbf{w}$; but it will be important in the second level of inference, and we name it the **evidence** for $\mathcal{H}_i$. It is common to use gradient–based methods to find the maximum of the posterior, which defines the most probable value for the parameters, $\mathbf{w}_{\rm MP}$; it is then common to summarise the posterior distribution by the value of $\mathbf{w}_{\rm MP}$, and error bars on these best fit parameters. The error bars are obtained from the curvature of the posterior; writing the Hessian $\mathbf{A} = -\nabla\nabla \log P(\mathbf{w}|D, \mathcal{H}_i)$ and Taylor–expanding the log posterior with $\Delta\mathbf{w} = \mathbf{w} - \mathbf{w}_{\rm MP}$,

$$P(\mathbf{w}|D, \mathcal{H}_i) \simeq P(\mathbf{w}_{\rm MP}|D, \mathcal{H}_i) \exp\left(-\tfrac{1}{2}\Delta\mathbf{w}^{\rm T}\mathbf{A}\Delta\mathbf{w}\right) \tag{2.2}$$

we see that the posterior can be locally approximated as a Gaussian with covariance matrix (error bars) $\mathbf{A}^{-1}$.[2]

2. **Model comparison.** At the second level of inference, we wish to infer which model is most plausible given the data. The posterior probability of each model is:

$$P(\mathcal{H}_i|D) \propto P(D|\mathcal{H}_i)P(\mathcal{H}_i). \tag{2.3}$$

Notice that the data–dependent term $P(D|\mathcal{H}_i)$ is the evidence for $\mathcal{H}_i$, which appeared as the normalising constant in (2.1). The second term, $P(\mathcal{H}_i)$, is a 'subjective' prior over our hypothesis space which expresses how plausible we thought the alternative models were before the data arrived. We will see later that this subjective part of the inference will typically be overwhelmed by the objective term, the evidence. Assuming that we have no reason to assign strongly differing priors $P(\mathcal{H}_i)$ to the alternative models, **models $\mathcal{H}_i$ are ranked by evaluating the evidence.** Equation (2.3) has not been normalised because in the data modelling process we may develop new models after the data have arrived (figure 2.1), when an inadequacy of the first models is detected, for example. So we do not start with a completely defined hypothesis space. Inference is open–ended: we continually seek more probable models to account for the data we gather. New models are compared with previous models by evaluating the evidence for them.

The key concept of this chapter is this: to assign a preference to alternative models $\mathcal{H}_i$, a Bayesian evaluates the evidence $P(D|\mathcal{H}_i)$. This concept is very general: the evidence can be evaluated for parametric and 'non–parametric' models alike; whether our data modelling task is a regression problem, a classification problem, or a density estimation problem, the evidence is the Bayesian's transportable quantity for comparing alternative models. In all these cases the evidence naturally embodies Occam's razor; we will examine how this works shortly.

Of course, the evidence is not the whole story if we have good reason to assign unequal priors to the alternative models $\mathcal{H}$. (To only use the evidence for model comparison is equivalent to using maximum likelihood for parameter estimation.) The classic example is

[2] Whether this approximation is a good one or not will depend on the problem we are solving. For the interpolation models discussed in this chapter, there is only a single maximum in the posterior distribution, and the Gaussian approximation is exact. For more general statistical models we still expect the posterior to be dominated by locally Gaussian peaks on account of the central limit theorem [84]. Multiple maxima which arise in more complex models complicate the analysis, but Bayesian methods can still successfully be applied [31, 50, 55].

the 'Sure Thing' hypothesis, © E.T Jaynes, which is the hypothesis that the data set will be D, the precise data set that actually occurred; the evidence for the Sure Thing hypothesis is huge. But Sure Thing belongs to an immense class of similar hypotheses which should all be assigned correspondingly tiny prior probabilities; so the posterior probability for Sure Thing is negligible alongside any sensible model. Models like Sure Thing are rarely seriously proposed in real life, but if such models are developed then clearly we need to think about precisely what priors are appropriate. Patrick and Wallace, studying the geometry of ancient stone circles (about which some people have proposed extremely elaborate theories!), discuss a practical method of assigning relative prior probabilities to alternative models by evaluating the lengths of the computer programs that decode data previously encoded under each model [60]. This procedure introduces a second sort of Occam's razor into the inference, namely a *prior* bias against complex models. However, we will not include such prior biases here; we will address only the data's preference for the alternative models, *i.e.*, the evidence, and the Occam's razor that it embodies. In the limit of large quantities of data this objective Occam's razor will always be the more important of the two.

A modern Bayesian approach to priors

It should be pointed out that the emphasis of this modern[3] Bayesian approach is not on the inclusion of priors into inference. There is not one significant 'subjective prior' in this entire chapter. (For problems where significant subjective priors do arise see [28, 73].) The emphasis is on the idea that consistent degrees of preference for alternative hypotheses are represented by probabilities, and relative preferences for models are assigned by evaluating those probabilities. Historically, Bayesian analysis has been accompanied by methods to work out the 'right' prior $P(\mathbf{w}|\mathcal{H})$ for a problem, for example, the principles of insufficient reason and maximum entropy. The modern Bayesian however does not take a fundamentalist attitude to assigning the 'right' priors — many different priors can be tried, allowing the data to inform us which is most appropriate. Each particular prior corresponds to a different hypothesis about the way the world is. We can compare these alternative hypotheses in the light of the data by evaluating the evidence. This is the way in which alternative regularisers are compared, for example. If we try one model and obtain awful predictions, we have *learnt* something. 'A failure of Bayesian prediction is an opportunity to learn' [36], and we are able to come back to the same data set with new models, using new priors for example.

Evaluating the evidence

Let us now explicitly study the evidence to gain insight into how the Bayesian Occam's razor works. The evidence is the normalising constant for equation (2.1):

$$P(D\,|\mathcal{H}_i) = \int P(D|\mathbf{w},\mathcal{H}_i)P(\mathbf{w}|\mathcal{H}_i)\,d\mathbf{w}. \tag{2.4}$$

For many problems, including interpolation, it is common for the posterior $P(\mathbf{w}|D,\mathcal{H}_i) \propto P(D|\mathbf{w},\mathcal{H}_i)P(\mathbf{w}|\mathcal{H}_i)$ to have a strong peak at the most probable parameters $\mathbf{w}_{\rm MP}$ (figure 2.3). Then the evidence can be approximated by the height of the peak of the integrand $P(D|\mathbf{w},\mathcal{H}_i)P(\mathbf{w}|\mathcal{H}_i)$ times its width, $\Delta\mathbf{w}$:

[3] Under this use of the word, Box and Tiao [10, 11, 12] must be counted as 'modern' Bayesians.

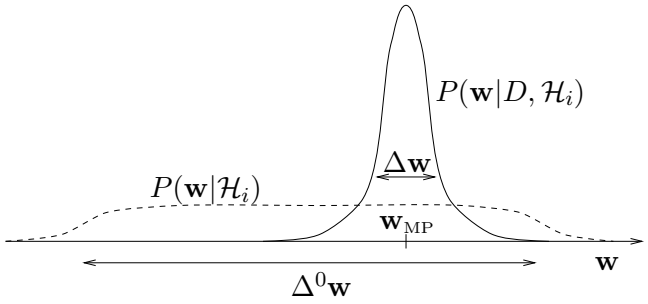

Figure 2.3: **The Occam factor**
This figure shows the quantities that determine the Occam factor for a hypothesis $\mathcal{H}_i$ having a single parameter $\mathbf{w}$. The prior distribution (dotted line) for the parameter has width $\Delta^0\mathbf{w}$. The posterior distribution (solid line) has a single peak at $\mathbf{w}_{\rm MP}$ with characteristic width $\Delta\mathbf{w}$. The Occam factor is $\frac{\Delta\mathbf{w}}{\Delta^0\mathbf{w}}$.

$$\begin{aligned} P(D\,|\mathcal{H}_i) \;&\simeq\; \underbrace{P(D\,|\mathbf{w}_{\rm MP}, \mathcal{H}_i)}\;\underbrace{P(\mathbf{w}_{\rm MP}|\mathcal{H}_i)\,\Delta\mathbf{w}}. \\ \text{Evidence} \;&\simeq\; \text{Best fit likelihood} \quad \text{Occam factor} \end{aligned} \tag{2.5}$$

Thus the evidence is found by taking the best fit likelihood that the model can achieve and multiplying it by an 'Occam factor' [26], which is a term with magnitude less than one that penalises $\mathcal{H}_i$ for having the parameter $\mathbf{w}$.

Interpretation of the Occam factor

The quantity $\Delta\mathbf{w}$ is the posterior uncertainty in $\mathbf{w}$. Imagine for simplicity that the prior $P(\mathbf{w}|\mathcal{H}_i)$ is uniform on some large interval $\Delta^0\mathbf{w}$, representing the range of values of $\mathbf{w}$ that $\mathcal{H}_i$ thought possible before the data arrived (figure 2.3). Then $P(\mathbf{w}_{\rm MP}|\mathcal{H}_i) = \frac{1}{\Delta^0\mathbf{w}}$, and

$$\text{Occam factor} = \frac{\Delta\mathbf{w}}{\Delta^0\mathbf{w}},$$

i.e., **the ratio of the posterior accessible volume of $\mathcal{H}_i$'s parameter space to the prior accessible volume,** or the factor by which $\mathcal{H}_i$'s hypothesis space collapses when the data arrive [26, 38]. The model $\mathcal{H}_i$ can be viewed as being composed of a certain number of equivalent submodels, of which only one survives when the data arrive. The Occam factor is the inverse of that number. The log of the Occam factor can be interpreted as the amount of information we gain about the model when the data arrive.

Typically, a complex model with many parameters, each of which is free to vary over a large range $\Delta^0\mathbf{w}$, will be penalised with a larger Occam factor than a simpler model. The Occam factor also provides a penalty for models which have to be finely tuned to fit the data; the Occam factor promotes models for which the required precision of the parameters $\Delta\mathbf{w}$ is coarse. The Occam factor is thus a measure of complexity of the model, but unlike the V–C dimension or algorithmic complexity, it relates to the complexity of the predictions that the model makes in data space; therefore it depends on the number of data points and other properties of the data set. Which model achieves the greatest evidence is determined by a trade–off between minimising this natural complexity measure and minimising the data misfit.

Occam factor for several parameters

If $\mathbf{w}$ is k-dimensional, and if the posterior is well approximated by a Gaussian, the Occam factor is obtained from the determinant of the Gaussian's covariance matrix:

$$\begin{aligned} P(D\,|\mathcal{H}_i) \;&\simeq\; \underbrace{P(D\,|\mathbf{w}_{\rm MP}, H_i)}\;\underbrace{P(\mathbf{w}_{\rm MP}|\mathcal{H}_i)\,(2\pi)^{k/2}\det{}^{-\frac{1}{2}}\mathbf{A}}, \\ \text{Evidence} \;&\simeq\; \text{Best fit likelihood} \qquad\quad \text{Occam factor} \end{aligned} \tag{2.6}$$

where $\mathbf{A} = -\nabla\nabla \log P(\mathbf{w}|D, \mathcal{H}_i)$, the Hessian which we already evaluated when we calculated the error bars on $\mathbf{w}_{\rm MP}$. As the amount of data collected, N, increases, this Gaussian approximation is expected to become increasingly accurate on account of the central limit theorem [84]. For the linear interpolation models discussed in this chapter, this Gaussian expression is exact for any N.

Comments

- Bayesian model selection is a simple extension of maximum likelihood model selection: **the evidence is obtained by multiplying the best fit likelihood by the Occam factor.**

  To evaluate the Occam factor all we need is the Hessian $\mathbf{A}$, if the Gaussian approximation is good. Thus the Bayesian method of model comparison by evaluating the evidence is computationally no more demanding than the task of finding for each model the best fit parameters and their error bars.

- It is common for there to be degeneracies in models with many parameters, *i.e.*, several equivalent parameters could be relabelled without affecting the likelihood. In these cases, the right hand side of equation (2.6) should be multiplied by the degeneracy of $\mathbf{w}_{\rm MP}$ to give the correct estimate of the evidence.

- 'Minimum description length' (MDL) methods are closely related to this Bayesian framework [65, 85, 86]. The log evidence $\log_2 P(D|\mathcal{H}_i)$ is the number of bits in the ideal shortest message that encodes the data D using model $\mathcal{H}_i$. Akaike's criterion, originally derived as a predictor of generalisation error [3], can be viewed, like Schwartz's 'B.I.C.', as an approximation to MDL and Bayes [68, 89]. Any implementation of MDL necessitates approximations in evaluating the length of the ideal shortest message. Although some of the earliest work on complex model comparison involved the MDL framework [60], MDL has no apparent advantages, and in my work I approximate the evidence directly.

- It should be emphasised that the Occam factor has nothing to do with how computationally complex it is to *use* a model. The evidence is a measure of *plausibility* of a model. How much CPU time it takes to use each model is certainly an interesting issue which might bias our decisions towards simpler models, but Bayes' rule does not address that issue. Choosing between models on the basis of how many function calls they need is an exercise in *decision theory*, which is not addressed in this chapter. Once the probabilities described above have been inferred, optimal actions can be chosen using standard decision theory with a suitable utility function.

2.3 The noisy interpolation problem

Bayesian interpolation through *noise–free* data has been studied by Skilling and Sibisi [70]. In this chapter I study the problem of interpolating through data where the dependent variables are assumed to be noisy (a task also known as 'regression', 'curve–fitting', 'signal estimation', or, in the neural networks community, 'learning'). I am not examining the case where the independent variables are also noisy. This different and more difficult problem has been studied for the case of straight line–fitting by Gull [28].

Let us assume that the data set to be interpolated is a set of pairs $D = \{x_m, t_m\}$, where $m = 1 \ldots N$ is a label running over the pairs. For simplicity I will treat x and t as scalars, but the method generalises to the multidimensional case. To define a linear interpolation model, a set of k fixed basis functions[4] $\mathcal{A} = \{\phi_h(x)\}$ is chosen, and the interpolated function is assumed to have the form:

$$y(x) = \sum_{h=1}^{k} w_h \phi_h(x), \tag{2.7}$$

where the parameters w_h are to be inferred from the data. The data set is modelled as deviating from this mapping under some additive noise process $\mathcal{N}$:

$$t_m = y(x_m) + \nu_m. \tag{2.8}$$

If ν is modelled as zero–mean Gaussian noise with standard deviation σ_ν, then the probability of the data[5] given the parameters $\mathbf{w}$ is:

$$P(D\,|\mathbf{w}, \beta, \mathcal{A}, \mathcal{N}) = \frac{\exp(-\beta E_D(D|\mathbf{w}, \mathcal{A}))}{Z_D(\beta)}, \tag{2.9}$$

where $\beta = 1/\sigma_\nu^2$, $E_D = \sum_m \frac{1}{2}(y(x_m) - t_m)^2$, and $Z_D = (2\pi/\beta)^{N/2}$. $P(D\,|\mathbf{w}, \beta, \mathcal{A}, \mathcal{N})$ is called the likelihood. It is well known that finding the maximum likelihood parameters $\mathbf{w}_{\rm ML}$ may be an 'ill–posed' problem. That is, the $\mathbf{w}$ that minimises E_D is underdetermined and/or depends sensitively on the details of the noise in the data; the maximum likelihood interpolant in such cases oscillates wildly so as to fit the noise. Thus it is clear that to complete an interpolation model we need a prior $\mathcal{R}$ that expresses the sort of smoothness we expect the interpolant $y(x)$ to have. A model may have a prior of the form

$$P(y|\mathcal{R}, \alpha) = \frac{\exp(-\alpha E_y(y|\mathcal{R}))}{Z_y(\alpha)}, \tag{2.10}$$

where E_y might be for example the functional $E_y = \int y''(x)^2 dx$ (which is the regulariser for cubic spline interpolation[6]). The parameter α is a measure of how smooth $f(x)$ is expected to be. Such a prior can also be written as a prior on the parameters $\mathbf{w}$:

$$P(\mathbf{w}|\alpha, \mathcal{A}, \mathcal{R}) = \frac{\exp(-\alpha E_W(\mathbf{w}|\mathcal{A}, \mathcal{R}))}{Z_W(\alpha)}, \tag{2.11}$$

[4] The case of *adaptive* basis functions, also known as feedforward neural networks, is examined in Chapter 3.

[5] Strictly, this probability should be written $P(\{t_m\}|\{x_m\}, \mathbf{w}, \beta, \mathcal{A}, \mathcal{N})$, since these interpolation models do not predict the distribution of input variables $\{x_m\}$; this liberty of notation will be taken throughout this thesis.

[6] Strictly, this particular prior may be improper because a $y(x)$ of the form $w_1 x + w_0$ is not constrained by this prior.

where $Z_W = \int d^k\mathbf{w} \exp(-\alpha E_W)$. E_W (or E_y) is commonly referred to as a regularising function.

The interpolation model $\mathcal{H}$ is now complete, consisting of a choice of basis functions $\mathcal{A}$, a noise model $\mathcal{N}$ with parameter β, and a prior (regulariser) $\mathcal{R}$, with regularising constant α. Particular settings of the hyperparameters α and β will be viewed as sub-models of $\mathcal{H}$.

The first level of inference

If α and β are known, then the posterior probability of the parameters $\mathbf{w}$ is:[7]

$$P(\mathbf{w}|D, \alpha, \beta, \mathcal{A}, \mathcal{R}, \mathcal{N}) = \frac{P(D|\mathbf{w}, \beta, \mathcal{A}, \mathcal{N})P(\mathbf{w}|\alpha, \mathcal{A}, \mathcal{R})}{P(D|\alpha, \beta, \mathcal{A}, \mathcal{R}, \mathcal{N})}. \tag{2.12}$$

Writing[8]

$$M(\mathbf{w}) = \alpha E_W + \beta E_D, \tag{2.13}$$

the posterior is

$$P(\mathbf{w}|D, \alpha, \beta, \mathcal{A}, \mathcal{R}, \mathcal{N}) = \frac{\exp(-M(\mathbf{w}))}{Z_M(\alpha, \beta)} \tag{2.14}$$

where $Z_M(\alpha, \beta) = \int d^k\mathbf{w} \exp(-M)$. We see that minimising the combined objective function M corresponds to finding the *most probable interpolant*, $\mathbf{w}_{\rm MP}$. Error bars on the best fit interpolant[9] can be obtained from the Hessian of M, $\mathbf{A} = \nabla\nabla M$, evaluated at $\mathbf{w}_{\rm MP}$.

This is the well known Bayesian view of regularisation [63, 83], also known as 'maximum penalised likelihood' or 'ridge regression'.

Bayesian methods provide far more than just an interpretation for regularisation. What we have described so far is just the first of three levels of inference. (The second level described in sections 1 and 2, 'model comparison', splits into a second and a third level for this problem, because each interpolation model is made up of a continuum of sub–models with different values of α and β.) At the second level, Bayes allows us to objectively assign values to α and β, which are commonly unknown *a priori*. At the third, Bayes enables us to quantitatively rank alternative basis sets $\mathcal{A}$, alternative regularisers (priors) $\mathcal{R}$, and, in principle, alternative noise models $\mathcal{N}$.[10] Furthermore, we can quantitatively compare interpolation under any model $\mathcal{H} = \{\mathcal{A}, \mathcal{N}, \mathcal{R}\}$ with other interpolation and learning models such as neural networks, if a similar Bayesian approach is applied to them. Neither the second nor the third level of inference can be successfully executed without Occam's razor.

The Bayesian theory of the second and third levels of inference has only recently been worked out [27]; this chapter's goal is to review that framework. Section 2.4 will describe the Bayesian method of inferring α and β; section 2.5 will describe Bayesian model comparison for the interpolation problem. Both these inference problems are solved by evaluation of the appropriate evidence.

2.4 Selection of parameters α and β

[7] The regulariser $\alpha, \mathcal{R}$ has been omitted from the conditioning variables in the likelihood because the data distribution does not depend on the prior once $\mathbf{w}$ is known. Similarly the prior does not depend on β,$\mathcal{N}$.

[8] The name M stands for 'misfit'; it will be demonstrated later that M is the natural measure of misfit, rather than $\chi^2_D = 2\beta E_D$.

[9] These error bars represent the uncertainty of the interpolant, and should not be confused with the typical scatter of noisy data points relative to the interpolant.

[10] Bayesian inference of a slightly non–Gaussian distribution is performed in Box and Tiao [10, 12].

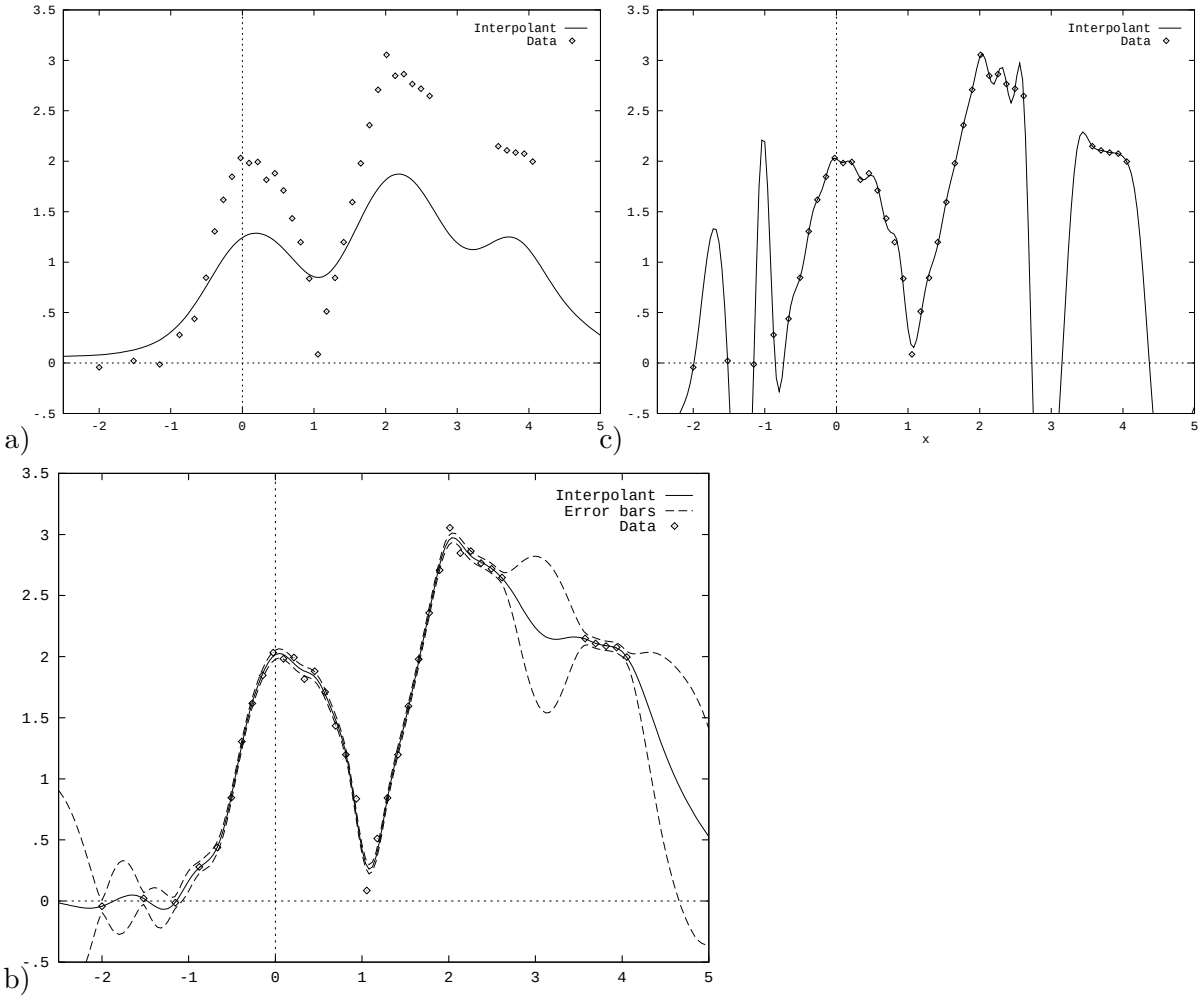

Figure 2.4: **How the best interpolant depends on** α

These figures introduce a data set, 'X', which is interpolated with a variety of models in this chapter. Notice that the density of data points is not uniform on the x–axis. In the three figures the data set is interpolated using a radial basis function model with a basis of 60 equally spaced Cauchy functions, all with radius 0.2975. The regulariser is $E_W = \frac{1}{2}\sum w^2$, where w are the coefficients of the basis functions. Each figure shows the most probable interpolant for a different value of α: a) 6000; b) 2.5; c) 10^{-7}. Note at the extreme values how the data are oversmoothed and overfitted respectively. Assuming a flat prior, $\alpha = 2.5$ is the *most probable* value of α. In b), the most probable interpolant is displayed with its 1σ error bars, which represent how uncertain we are about the interpolant at each point, under the assumption that the interpolation model and the value of α are correct. Notice how the error bars increase in magnitude where the data are sparse. The error bars do not get bigger near the datapoint close to (1,0), because the radial basis function model does not expect sharp discontinuities; the error bars are obtained *assuming the model is correct,* so that point is interpreted as an improbable outlier.

Typically, α is not known *a priori*, and often β is also unknown. As α is varied, the properties of the best fit (most probable) interpolant vary. Assume that we are using a prior that encourages smoothness, and imagine that we interpolate at a very large value of α; then this will constrain the interpolant to be very smooth and flat, and it will not fit the data at all well (figure 2.4a). As α is decreased, the interpolant starts to fit the data better (figure 2.4b). If α is made even smaller, the interpolant oscillates wildly so as to overfit the noise in the data (figure 2.4c). The choice of the 'best' value of α is our first 'Occam's razor' problem: large values of α correspond to simple models which make constrained and precise predictions, saying 'the interpolant is expected to not have extreme curvature anywhere'; a tiny value of α corresponds to the more powerful and flexible model that says 'the interpolant could be anything at all, our prior belief in smoothness is very weak'. The task is to find a value of α which is small enough that the data are fitted but not so small that they are overfitted. For more severely ill–posed problems such as deconvolution, the precise value of the regularising parameter is increasingly important. Orthodox statistics has ways of assigning values to such parameters, based for example on misfit criteria, the use of test data, and cross–validation. Gull has demonstrated why the popular use of misfit criteria is incorrect and how Bayes sets these parameters [27]. The use of test data may be an unreliable technique unless large quantities of data are available. Cross–validation, the orthodox 'method of choice' [22], will be discussed more in section 2.6 and chapter 3. I will explain the Bayesian method of inferring α and β after first reviewing some statistics of misfit.

Misfit, χ^2, and the effect of parameter measurements

For N independent Gaussian variables with mean μ and standard deviation σ, the statistic $\chi^2 = \sum(x-\mu)^2/\sigma^2$ is a measure of misfit. If μ is known *a priori*, χ^2 has expectation $N \pm \sqrt{N}$. However, if μ is fitted from the data by setting $\mu = \bar{x}$, we 'use up a degree of freedom', and χ^2 has expectation $N-1$. In the second case μ is a 'well–measured parameter'. When a parameter is determined by the data in this way it is unavoidable that the parameter fits some of the noise in the data as well. That is why the expectation of χ^2 is reduced by one. This is the basis of the distinction between the σ_N and σ_{N-1} buttons on your calculator. It is common for this distinction to be ignored, but in cases such as interpolation where the number of free parameters is similar to the number of data points, it is essential to find and make the analogous distinction. It will be demonstrated that the Bayesian choices of both α and β are most simply expressed in terms of the effective number of well–measured parameters, γ, to be derived below.

Misfit criteria are 'principles' which set parameters like α and β by requiring that χ^2 should have a particular value. The discrepancy principle requires $\chi^2 = N$. Another principle requires $\chi^2 = N - k$, where k is the number of free parameters. We will find that an intuitive misfit criterion arises for the most probable value of β; on the other hand, the Bayesian choice of α will be unrelated to the value of the misfit.

Bayesian choice of α and β

To infer from the data what value α and β should have, Bayesians evaluate the posterior probability distribution:

$$P(\alpha, \beta | D, \mathcal{H}) = \frac{P(D|\alpha, \beta, \mathcal{H}) P(\alpha, \beta | \mathcal{H})}{P(D|\mathcal{H})}. \tag{2.15}$$

The data dependent term $P(D|\alpha, \beta, \mathcal{H})$ has already appeared earlier as the normalising constant in equation (2.12), and it is called the evidence for α and β. Similarly the normalising constant of (2.15) is called the evidence for $\mathcal{H}$, and it will turn up later when we compare alternative models $\mathcal{H} = \{\mathcal{A}, \mathcal{N}, \mathcal{R}\}$ in the light of the data.

If $P(\alpha, \beta|\mathcal{H})$ is a flat prior[11] (which corresponds to the statement that we don't know what value α and β should have), the evidence is the function that we use to assign a preference to alternative values of α and β. It is given in terms of the normalising constants defined earlier by

$$P(D|\alpha, \beta, \mathcal{H}) = \frac{Z_M(\alpha, \beta)}{Z_W(\alpha) Z_D(\beta)}. \tag{2.16}$$

Occam's razor is implicit in this formula: if α is small, the large freedom in the prior range of possible values of $\mathbf{w}$ is automatically penalised by the consequent large value of Z_W; models that fit the data well achieve a large value of Z_M. The optimum value of α achieves a compromise between fitting the data well and being a simple model.

Now to assign a preference to (α, β), our computational task is to evaluate the three integrals Z_M, Z_W and Z_D. We will come back to this task in a moment.

But that sounds like determining your prior after the data have arrived!

When I first heard the preceding explanation of Bayesian regularisation I was discontent because it seemed that the prior is being chosen from an ensemble of possible priors *after* the data have arrived. To be precise, as described above, the most probable value of α is selected; then the prior corresponding to that value of α alone is used to infer what the interpolant might be. This is not how Bayes would have us infer the interpolant. It is the combined ensemble of priors that define our prior, and we should integrate over this ensemble when we do inference.[12] Let us work out what happens if we follow this proper approach. The preceding method of using only the most probable prior will emerge as a good approximation.

The true posterior $P(\mathbf{w}|D, \mathcal{H})$ is obtained by integrating over α and β:

$$P(\mathbf{w}|D, \mathcal{H}) = \int P(\mathbf{w}|D, \alpha, \beta, \mathcal{H}) P(\alpha, \beta|D, \mathcal{H}) \, d\alpha \, d\beta. \tag{2.17}$$

In words, the posterior probability over $\mathbf{w}$ can be written as a linear combination of the posteriors for all values of α, β. Each posterior density is weighted by the probability of α, β given the data, which appeared in (2.15). This means that if $P(\alpha, \beta|D, \mathcal{H})$ has a dominant peak at $\hat{\alpha}, \hat{\beta}$, then the true posterior $P(\mathbf{w}|D, \mathcal{H})$ will be dominated by the density $P(\mathbf{w}|D, \hat{\alpha}, \hat{\beta}, \mathcal{H})$. As long as the properties of the posterior $P(\mathbf{w}|D, \alpha, \beta, \mathcal{H})$ do not change rapidly with α, β near $\hat{\alpha}, \hat{\beta}$ and the peak in $P(\alpha, \beta|D, \mathcal{H})$ is strong, we are justified in using the approximation:

$$P(\mathbf{w}|D, \mathcal{H}) \simeq P(\mathbf{w}|D, \hat{\alpha}, \hat{\beta}, \mathcal{H}). \tag{2.18}$$

This approximation is valid if under the same conditions as in footnote 13. It is a matter of ongoing research to develop computational methods for cases where this approximation is invalid (Sibisi and Skilling, personal communication, Neal, personal communication). In some cases, including the linear models of this chapter, the integral (2.17) can be performed

[11] Since α and β are scale parameters, this prior should be understood as a flat prior over $\log \alpha$ and $\log \beta$.

[12] It is remarkable that Laplace almost got this right in 1774 [80]; when inferring the mean of a Laplacian distribution, he both inferred the posterior probability of a nuisance parameter like β in (2.15), and then attempted to integrate out the nuisance parameter as in equation (2.17).

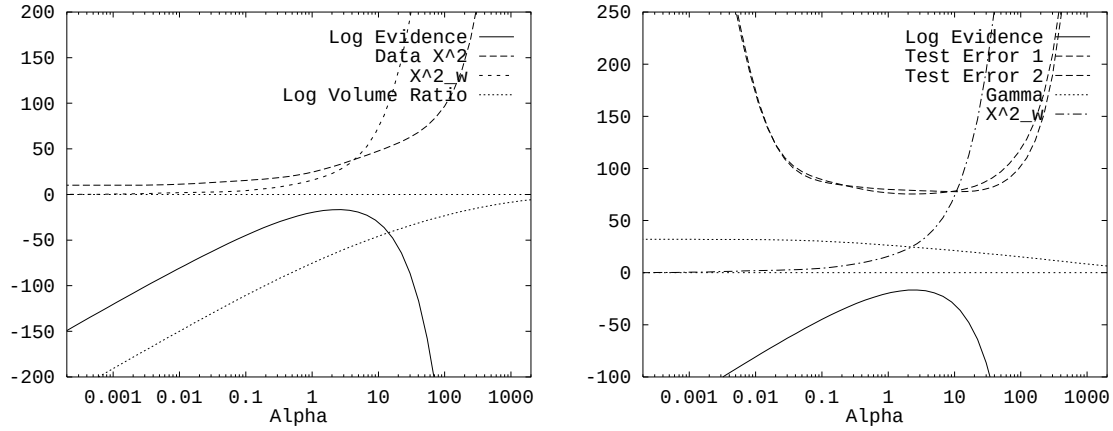

Figure 2.5: **Choosing** α

a) **The evidence as a function of** α**:** Using the same radial basis function model as in figure 2.4, this graph shows the log evidence as a function of α, and shows the functions which make up the log evidence, namely the data misfit $\chi^2_D = 2\beta E_D$, the weight penalty term $\chi^2_W = 2\alpha E_W$, and the log of the volume ratio $(2\pi)^{k/2}\det{}^{-\frac{1}{2}}\mathbf{A}/Z_W(\alpha)$.

b) **Criteria for optimising** α**:** This graph shows the log evidence as a function of α, and the functions whose intersection locates the evidence maximum: the number of good parameter measurements γ, and χ^2_W. Also shown is the test error (rescaled) on two test sets; finding the test error minimum is an alternative criterion for setting α. Both test sets were more than twice as large in size as the interpolated data set. Note how the point at which $\chi^2_W = \gamma$ is clear and unambiguous, which cannot be said for the minima of the test energies. The evidence gives α a 1-σ confidence interval of $[1.3, 5.0]$. The test error minima are more widely distributed because of finite sample noise.

analytically. I have chosen to use the approximations regardless, because 1) the approximations give a clearer intuition for how Bayesian methods solve regularisation problems; 2) the approximations are applicable to cases where there is no analytic solution; and 3) the approximations relate most closely to alternative regularisation methods, which seek to find 'optimal' values of α, β.

Why not find the joint optimum in $\mathbf{w}, \alpha, \beta$?

It is not satisfactory to simply maximise the likelihood or the posterior probability simultaneously over $\mathbf{w}$, α and β; the posterior and likelihood both have skew peaks such that the maximum likelihood value for the parameters is not in the same place as most of the posterior probability [27]. To get a feeling for this here is a more familiar problem: examine the posterior probability for the parameters of a Gaussian (μ, σ) given N samples: the maximum likelihood value for σ is σ_N, but the most probable value for σ (found by integrating over μ) is σ_{N-1}. It should be emphasised that this distinction has nothing to do with the prior over the parameters α and β, which is flat here. It is the process of marginalisation that corrects the bias which afflicts both maximum likelihood and maximum a posteriori.

Evaluating the evidence

Let us return to our train of thought at equation (2.16). To evaluate the evidence for α, β, we want to find the integrals Z_M, Z_W and Z_D. Typically the most difficult integral to evaluate is Z_M.

$$Z_M(\alpha, \beta) = \int d^k\mathbf{w} \, \exp(-M(\mathbf{w}, \alpha, \beta)).$$

If the regulariser $\mathcal{R}$ is a quadratic functional (and the favourites are), then E_D and E_W are quadratic functions of $\mathbf{w}$, and we can evaluate Z_M exactly. Letting $\nabla\nabla E_W = \mathbf{C}$ and $\nabla\nabla E_D = \mathbf{B}$ then using $\mathbf{A} = \alpha\mathbf{C} + \beta\mathbf{B}$, we have:

$$M = M(\mathbf{w}_{\rm MP}) + \frac{1}{2}(\mathbf{w} - \mathbf{w}_{\rm MP})^{\rm T}\mathbf{A}(\mathbf{w} - \mathbf{w}_{\rm MP}),$$

where $\mathbf{w}_{\rm MP} = \beta\mathbf{A}^{-1}\mathbf{B}\mathbf{w}_{\rm ML}$. This means that Z_M is the Gaussian integral:

$$Z_M = e^{-M_{\rm MP}}(2\pi)^{k/2}\det{}^{-\frac{1}{2}}\mathbf{A}. \tag{2.19}$$

In many cases where the regulariser is not quadratic (for example, entropy–based), this Gaussian approximation is still servicable [27]. Thus we can write the log evidence for α and β as:

$$\log P(D|\alpha, \beta, \mathcal{H}) = -\alpha E_W^{\rm MP} - \beta E_D^{\rm MP} - \frac{1}{2}\log\det\mathbf{A} - \log Z_W(\alpha) - \log Z_D(\beta) + \frac{k}{2}\log 2\pi. \tag{2.20}$$

The term $\beta E_D^{\rm MP}$ represents the misfit of the interpolant to the data. The three terms $-\alpha E_W^{\rm MP} - \frac{1}{2}\log\det\mathbf{A} - \log Z_W(\alpha)$ constitute the log of the 'Occam factor' penalising small values of α: the ratio $(2\pi)^{k/2}\det{}^{-\frac{1}{2}}\mathbf{A}/Z_W(\alpha)$ is the ratio of the posterior accessible volume in parameter space to the prior accessible volume, and the term $\alpha E_W^{\rm MP}$ measures how far $\mathbf{w}_{\rm MP}$ is from its null value. Figure 2.5a illustrates the behaviour of these various terms as a function of α for the same radial basis function model as illustrated in figure 2.4.

Now we could just proceed to evaluate the evidence numerically as a function of α and β, but a more deep and fruitful understanding of this problem is possible.

Properties of the evidence maximum

The maximum over α, β of $P(D|\alpha, \beta, \mathcal{H}) = Z_M(\alpha, \beta)/(Z_W(\alpha)Z_D(\beta))$ has some remarkable properties which give deeper insight into this Bayesian approach. The results of this section are useful both numerically and intuitively.

Following Gull [27], we transform to the basis in which the Hessian of E_W is the identity, $\nabla\nabla E_W = \mathbf{I}$. This transformation is simple in the case of quadratic E_W: rotate into the eigenvector basis of $\mathbf{C}$ and stretch the axes so that the quadratic form E_W becomes homogeneous. This is the natural basis for the prior. I will continue to refer to the parameter vector in this basis as $\mathbf{w}$, so from here on $E_W = \frac{1}{2}\sum w_i^2$. Using $\nabla\nabla M = \mathbf{A}$ and $\nabla\nabla E_D = \mathbf{B}$ as above, we differentiate the log evidence with respect to α and β so as to find the condition that is satisfied at the maximum. The log evidence, from (2.20), is:

$$\log P(D|\alpha, \beta, \mathcal{H}) = -\alpha E_W^{\rm MP} - \beta E_D^{\rm MP} - \frac{1}{2}\log\det\mathbf{A} + \frac{k}{2}\log\alpha + \frac{N}{2}\log\beta - \frac{N}{2}\log 2\pi. \tag{2.21}$$

First, differentiating with respect to α, we need to evaluate $\frac{d}{d\alpha}\log\det\mathbf{A}$. Using $\mathbf{A} = \alpha\mathbf{I} + \beta\mathbf{B}$,

$$\begin{aligned}\frac{d}{d\alpha}\log\det\mathbf{A} &= \mathrm{Trace}\left(\mathbf{A}^{-1}\frac{d\mathbf{A}}{d\alpha}\right) \\ &= \mathrm{Trace}\left(\mathbf{A}^{-1}\mathbf{I}\right) = \mathrm{Trace}\mathbf{A}^{-1}.\end{aligned}$$

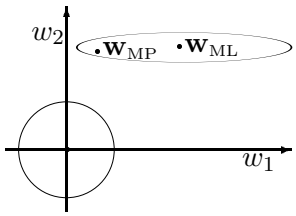

Figure 2.6: **Good and bad parameter measurements**
Let w_1 and w_2 be the components in parameter space in two directions parallel to eigenvectors of the data matrix $\mathbf{B}$. The circle represents the characteristic prior distribution for $\mathbf{w}$. The ellipse represents a characteristic contour of the likelihood, centred on the maximum likelihood solution $\mathbf{w}_{\rm ML}$. $\mathbf{w}_{\rm MP}$ represents the most probable parameter vector. w_1 is a direction in which λ_1 is small compared to α, *i.e.*, the data have no strong preference about the value of w_1; w_1 is a poorly measured parameter, and the term $\frac{\lambda_1}{\lambda_1+\alpha}$ is close to zero. w_2 is a direction in which λ_1 is large; w_2 is well determined by the data, and the term $\frac{\lambda_2}{\lambda_2+\alpha}$ is close to one.

This result is exact if E_W and E_D are quadratic. Otherwise this result is an approximation, omitting terms in $\partial\mathbf{B}/\partial\alpha$. Now, differentiating (2.21) and setting the derivative to zero, we obtain the following condition for the most probable value of α:

$$2\alpha E_W^{\rm MP} = k - \alpha {\rm Trace}\mathbf{A}^{-1}. \tag{2.22}$$

The quantity on the left is the dimensionless measure of the amount of structure introduced into the parameters by the data, *i.e.*, how much the fitted parameters differ from their null value. It can be interpreted as the χ^2 of the parameters, since it is equal to $\chi^2_W = \sum w_i^2/\sigma_W^2$, with $\alpha = 1/\sigma_W^2$.

The quantity on the right of (2.22) is called the number of good parameter measurements, γ, and has value between 0 and k. It can be written in terms of the eigenvalues of $\beta\mathbf{B}$, λ_a, where the subscript a runs over the k eigenvectors. The eigenvalues of $\mathbf{A}$ are $\lambda_a + \alpha$, so we have:

$$\gamma = k - \alpha\,{\rm Trace}\mathbf{A}^{-1} = k - \sum_{a=1}^{k} \frac{\alpha}{\lambda_a + \alpha} = \sum_{a=1}^{k} \frac{\lambda_a}{\lambda_a + \alpha}. \tag{2.23}$$

Each eigenvalue λ_a measures how strongly one parameter is determined by the data. The constant α measures how strongly the parameters are determined by the prior. The ath term $\gamma_a = \lambda_a/(\lambda_a + \alpha)$ is a number between 0 and 1 which measures the strength of the data relative to the prior in direction a (figure 2.6): the components of $\mathbf{w}_{\rm MP}$ are given by $\mathbf{w}_{{\rm MP}a} = \gamma_a \mathbf{w}_{{\rm ML}a}$.

A direction in parameter space for which λ_a is small compared to α does not contribute to the number of good parameter measurements. γ is thus a measure of the effective number of parameters which are well determined by the data. As $\alpha/\beta \to 0$, γ increases from 0 to k. The condition (2.22) for the most probable value of α can therefore be interpreted as an estimation of the variance σ_W^2 of the Gaussian distribution from which the weights are drawn, based on γ effective samples from that distribution: $\sigma_W^2 = \sum w_i^2/\gamma$.

This concept is not only important for locating the optimum value of α: it is only the γ good parameter measurements which are expected to contribute to the reduction of the data misfit that occurs when a model is fitted to noisy data. In the process of fitting $\mathbf{w}$

to the data, it is unavoidable that some fitting of the model to noise will occur, because some components of the noise are indistinguishable from real data. Typically, one unit (χ^2) of noise will be fitted for every well–determined parameter. Poorly determined parameters are determined by the regulariser only, so they do not reduce χ^2_D in this way. We will now examine how this concept enters into the Bayesian choice of β.

Recall that the expectation of the χ^2 misfit between the true interpolant and the data is N. However we do not know the true interpolant, and the only misfit measure to which we have access is the χ^2 between the *inferred* interpolant and the data, $\chi^2_D = 2\beta E_D$. The 'discrepancy principle' of orthodox statistics states that the model parameters should be adjusted so as to make $\chi^2_D = N$. Work on un–regularised least–squares regression suggests that we should estimate the noise level so as to set $\chi^2_D = N - k$, where k is the number of free parameters. Let us find out the opinion of Bayes' rule on this matter.

We differentiate the log evidence (2.21) with respect to β and obtain, setting the derivative to zero:

$$2\beta E_D = N - \gamma. \tag{2.24}$$

Thus the most probable noise estimate, $\hat{\beta}$, does not satisfy $\chi^2_D = N$ or $\chi^2_D = N - k$; rather, $\chi^2_D = N - \gamma$. This Bayesian estimate of noise level naturally takes into account the fact that the parameters which have been determined by the data inevitably suppress some of the noise in the data, while the poorly measured parameters do not. The quantity $N - \gamma$ may be called the effective number of degrees of freedom. Note that the value of χ^2_D only enters into the determination of β: misfit criteria have no role in the Bayesian choice of α [27].

In summary, at the optimum value of α and β, $\chi^2_W = \gamma$, $\chi^2_D = N - \gamma$. Notice that this implies that the total misfit $M = \alpha E_W + \beta E_D$ satisfies the simple equation $2M = N$.

The interpolant resulting from the Bayesian choice of α is illustrated by figure 2.4b. Figure 2.5b illustrates the functions involved with the Bayesian choice of α, and compares them with the 'test error' approach. Demonstration of the Bayesian choice of β is omitted, since it is straightforward; β is fixed to its true value for the demonstrations in this chapter. Inference of an input–dependent noise level $\beta(x)$ will be demonstrated in a future publication.

These results generalise to the case where there are two or more separate regularisers with independent regularising constants $\{\alpha_c\}$ [27]. In this case, each regulariser has a number of good parameter measurements γ_c associated with it. Multiple regularisers will be used for neural networks in chapter 3.

Finding the evidence maximum with a head–on approach would involve evaluating $\det \mathbf{A}$ while searching over α, β; the above results (2.22,2.24) enable us to speed up this search (for example by the use of re–estimation formulae like $\alpha := \gamma / 2E_W$) and replace the evaluation of $\det \mathbf{A}$ by the evaluation of $\mathrm{Trace}\mathbf{A}^{-1}$. For large–dimensional problems where this task is demanding, Skilling has developed methods for estimating $\mathrm{Trace}\mathbf{A}^{-1}$ statistically in k^2 time [72].

2.5 Model comparison

To rank alternative basis sets $\mathcal{A}$, noise models $\mathcal{N}$ and regularisers (priors) $\mathcal{R}$ in the light of the data, we examine the posterior probabilities for alternative models $\mathcal{H} = \{\mathcal{A}, \mathcal{N}, \mathcal{R}\}$:

$$P(\mathcal{H}|D) \propto P(D|\mathcal{H})P(\mathcal{H}). \tag{2.25}$$

The data–dependent term, the evidence for $\mathcal{H}$, appeared earlier as the normalising constant in (2.15), and it is evaluated by integrating the evidence for (α, β):

$$P(D|\mathcal{H}) = \int P(D|\alpha, \beta, \mathcal{H}) P(\alpha, \beta|\mathcal{H}) \, d\alpha \, d\beta. \tag{2.26}$$

Assuming that we have no reason to assign strongly differing priors $P(\mathcal{H})$, alternative models $\mathcal{H}$ are ranked just by examining the evidence. The evidence can also be compared with the evidence found by an equivalent Bayesian analysis of other learning and interpolation models so as to allow the data to assign a preference to the alternative models. Notice as pointed out earlier that this modern Bayesian framework includes no emphasis on defining the 'right' prior $\mathcal{R}$ with which we ought to interpolate. Rather, we invent as many priors (regularisers) as we want, and allow the data to tell us which prior is *most probable*. Having said this, experience recommends that the 'maximum entropy principle' and other respected guides should be consulted when inventing these priors (see [26], for example).

Evaluating the evidence for $\mathcal{H}$

As α and β vary, a single evidence maximum is obtained, at $\hat{\alpha}$,$\hat{\beta}$ (at least for quadratic E_D and E_W). The evidence maximum is often well approximated[13] by a separable Gaussian, and differentiating (2.21) twice we obtain Gaussian error bars for $\log \alpha$ and $\log \beta$:

$$(\Delta \log \alpha)^2 \simeq 2/\gamma$$
$$(\Delta \log \beta)^2 \simeq 2/(N - \gamma).$$

Putting these error bars into (2.26), we obtain the evidence.[14]

$$P(D|\mathcal{H}) \simeq P(D|\hat{\alpha}, \hat{\beta}, \mathcal{H}) P(\hat{\alpha}, \hat{\beta}|\mathcal{H}) \, 2\pi \, \Delta\log \alpha \, \Delta\log \beta \tag{2.27}$$

How is the prior $P(\hat{\alpha}, \hat{\beta}|\mathcal{H})$ assigned? This is the first time in this chapter that we have met one of the infamous 'subjective priors' which are supposed to plague Bayesian methods. Here are some answers to this question. (a) Any coherent method of assigning a preference to alternatives must implicitly assign such priors [46]. Bayesians adopt the healthy attitude of not sweeping them under the carpet. (b) With some thought, reasonable values can usually be assigned to subjective priors, and the degree of reasonable subjectivity in these assignments can be quantified, and the sensitivity of our inferences to these priors can be quantified [10, 12]. For example, a reasonable prior on an unknown standard deviation states that σ is unknown over a range of (3 ± 2) orders of magnitude. This prior contributes a subjectivity of about ±1 to the value of the log evidence. This degree of subjectivity is often negligible compared to the log evidence differences. (c) In the noisy interpolation example, all models considered include the free parameters α and β. So in this chapter I do not need to assign a value to $P(\hat{\alpha}, \hat{\beta}|\mathcal{H})$; I assume that it is a flat prior (flat over $\log \alpha$ and $\log \beta$, since α and β are scale parameters) which cancels out when we compare alternative interpolation models.

2.6 Demonstration

These demonstrations will use two one–dimensional data sets, in imitation of [70]. The first data set, 'X', has discontinuities in derivative (figure 2.4), and the second is a smoother

[13] This approximation is valid when $\gamma \gg 1$, and, in the spectrum of eigenvalues of $\beta\mathbf{B}$, the number of eigenvalues within e–fold of $\hat{\alpha}$ is $\ll \gamma$.

[14] There are analytic methods for performing such integrals over β [14].

data set, 'Y' (figure 2.8). In all the demonstrations, β was not left as a free parameter, but was fixed to its known true value.

Error bars on one model's interpolant

The Bayesian method of setting α, assuming a single model is correct, has already been demonstrated, and quantified error bars have been placed on the most probable interpolant (figure 2.4). The method of evaluating the error bars is to use the posterior covariance matrix of the parameters w_h, $\mathbf{A}^{-1}$, to get the variance on $y(x)$, which for any x is a linear function of the parameters, $y(x) = \sum_h \phi_h(x) w_h$. The error bars at a single point x are given by $\mathrm{var}\, y(x) = \phi^{\mathrm{T}} \mathbf{A}^{-1} \phi$. These error bars are directly related to the expected generalisation error at x, assuming that the model is true, evaluated in [43, 82]. The error bars are also related to the expected information gain per data point (chapter 4). Actually we have access to the full covariance information for the entire interpolant, not just the pointwise error bars. It is possible to visualise the *joint* error bars on the interpolant by making typical samples from the posterior distribution, performing a random walk around the posterior 'bubble' in parameter space [70, 74]. Figure 2.8 shows data set Y interpolated by three typical interpolants found by random sampling from the posterior distribution. These error bar properties are found under the assumption that the model is correct; so it is possible for the true interpolant to lie significantly outside the error bars of a poor model.

Model comparison

In this section Bayesian model comparison will be demonstrated first with models differing only in the number of free parameters (for example polynomials of different degrees), then with comparisons between models as disparate as splines, radial basis functions and feedforward neural networks. The characters of some of these models are illustrated in figure 2.9, which shows a typical sample from each. For each individual model, the value of α is optimised, and the evidence is evaluated by integrating over α using the Gaussian approximation. All logarithms are to base e.

Legendre polynomials: Occam's razor for the number of basis functions

Figure 2.7a shows the evidence for Legendre polynomials of different degrees for data set X. The basis functions were chosen to be orthonormal on an interval enclosing the data, and a regulariser of the form $E_W = \sum \frac{1}{2} w_h^2$ was used.

Notice that an evidence maximum is obtained: beyond a certain number of terms, the evidence starts to decrease. This is the Bayesian Occam's razor at work. The additional terms make the model more powerful, able to make more predictions. This flexibility is automatically penalised. Notice the characteristic shape of the 'Occam hill'. On the left, the hill is steep as the over–simple models fail to fit the data; the penalty for misfitting the data scales as N, the number of data measurements. The other side of the hill is much less steep; the log Occam factors here only scale as $k \log N$, where k is the number of parameters. We note in table 2.1 the value of the maximum evidence achieved by these models, and move on to alternative models.

The choice of orthonormal Legendre polynomials described above was motivated by a maximum entropy argument [26]. Models using other polynomial basis sets have also been tried. For less well motivated basis sets such as Hermite polynomials, it was found that the Occam factors were far bigger and the evidence was substantially smaller. If the size of the

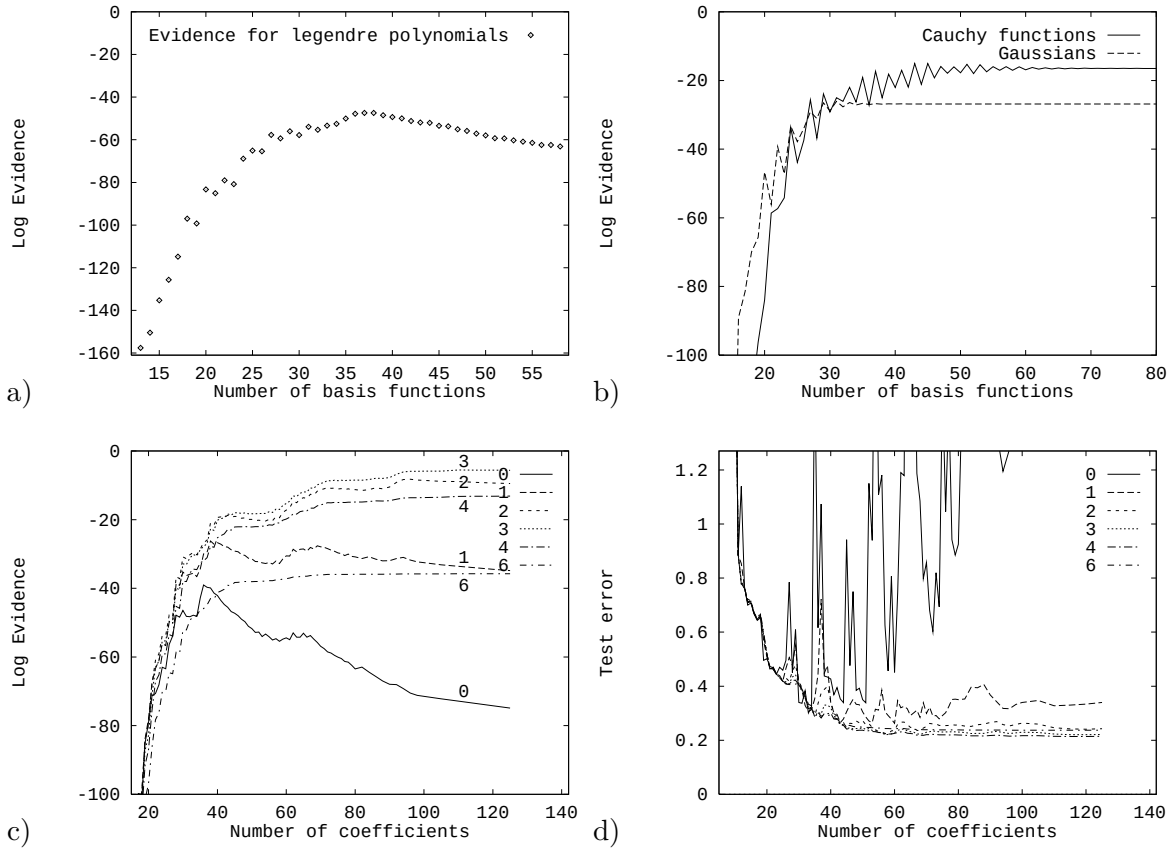

Figure 2.7: **The evidence for data set X** (see also table 1)
a) Log Evidence for Legendre polynomials. Notice the evidence maximum. The gentle slope to the right is due to the 'Occam factors' which penalise the increasing complexity of the model. **b) Log Evidence for radial basis function models.** Notice that there is no Occam penalty for the additional coefficients in these models, because increased density of radial basis functions does not make the model more powerful. The oscillations in the evidence are due to the details of the pixellation of the basis functions relative to the data points. **c) Log Evidence for splines.** The evidence is shown for the alternative splines regularisers $p = 0 \ldots 6$ (see text). In the representation used, each spline model is obtained in the limit of an infinite number of coefficients. For example, $p = 4$ yields the cubic splines model. **d) Test error for splines.** The number of data points in the test set was 90, *c.f.* number of data points in training set = 37. The y axis shows E_D; the value of E_D for the true interpolant has expectation 0.225 ± 0.02.

Table 2.1: **Evidence for models interpolating data sets X and Y**
All logs are natural. The evidence $P(D|\mathcal{H})$ is a density over D space, so the absolute value of the log evidence is arbitrary within an additive constant. Only differences in values of log evidences are relevant, relating directly to probability ratios.

Model	Data Set X		Data Set Y	
	Best parameter values	Log evidence	Best parameter values	Log evidence
Legendre polynomials	$k = 38$	-47	$k = 11$	23.8
Gaussian radial basis functions	$k > 40$, $r = .25$	-28.8 ± 1.0	$k > 50$, $r = .77$	27.1 ± 1.0
Cauchy radial basis functions	$k > 50$, $r = .27$	-18.9 ± 1.0	$k > 50$, $r = 1.1$	25.7 ± 1.0
Splines, $p = 2$	$k > 80$	-9.5	$k > 50$	8.2
Splines, $p = 3$	$k > 80$	-5.6	$k > 50$	19.8
Splines, $p = 4$	$k > 80$	-13.2	$k > 50$	22.1
Splines, $p = 5$	$k > 80$	-24.9	$k > 50$	21.8
Splines, $p = 6$	$k > 80$	-35.8	$k > 50$	20.4
Hermite functions	$k = 18$	-66	$k = 3$	42.2
Neural networks	8 neurons, $k = 25$	-12.6	6 neurons, $k = 19$	25.7

Occam factor increases rapidly with over–parameterisation, it is generally a sign that the space of alternative models is poorly matched to the problem.

Fixed radial basis functions

For a radial basis function or 'kernel' model, the basis functions are $\phi_h(x) = g((x-x_h)/r)/r$; here the x_h are equally spaced over the range of interest. I examine two choices of g: a Gaussian and a Cauchy function, $1/1+x^2$. We can quantitatively compare these alternative models of spatial correlation for any data set by evaluating the evidence. The regulariser is $E_W = \sum \frac{1}{2} w_h^2$. Note that this model includes one new free parameter, r; in these demonstrations this parameter has been set to its most probable value (*i.e.*, the value which maximises the evidence). To penalise this free parameter an Occam factor is included, $\sqrt{2\pi} P(\log r) \Delta \log r$, where $\Delta \log r =$ posterior uncertainty in $\log r$, and $P(\log r)$ is the prior on $\log r$, which is subjective to a small degree (I used $P(\log r) = 1/(4 \pm 2)$). This radial basis function model is the same as the 'intrinsic correlation' model of Charter, Gull, Skilling and Sibisi [16, 27, 70].

Figure 2.7b shows the evidence as a function of the number of basis functions, k. Note that for these models there is *not* an increasing Occam penalty for large numbers of parameters. The reason for this is that these extra parameters do not make the model any more powerful (for fixed α and r). The increased density of basis functions does not enable the model to make any significant new predictions because the kernel g band–limits the possible interpolants.

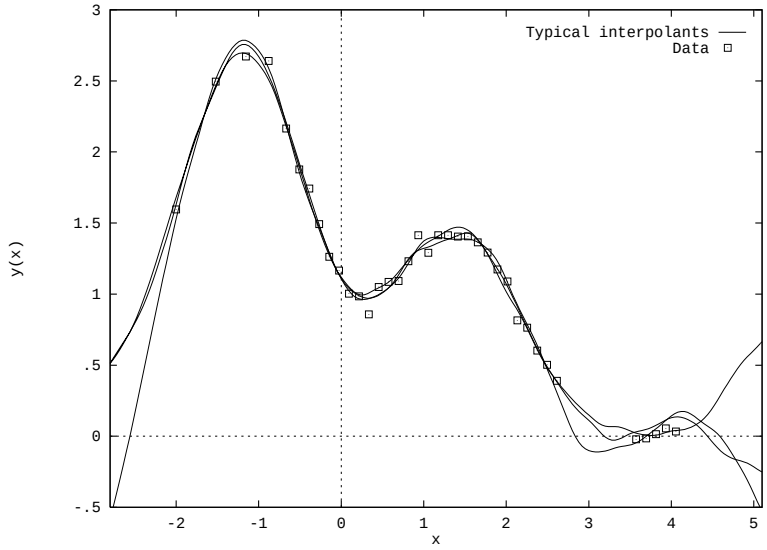

Figure 2.8: **Data set 'Y', interpolated with splines,** $p = 5$**.**
The data set is shown with three typical interpolants drawn from the posterior probability distribution. Contrast this with figure 2.4b, in which the most probable interpolant is shown with its pointwise error bars.

Splines: Occam's razor for the choice of regulariser

The splines models were implemented as follows: let the basis functions be a Fourier set $\cos hx, \sin hx$, $h = 0, 1, 2, \ldots$. Use the regulariser $E_W = \sum \frac{1}{2} h^p w_{h(\cos)}^2 + \sum \frac{1}{2} h^p w_{h(\sin)}^2$. If $p = 4$ then in the limit $k \to \infty$ we have the cubic splines regulariser $E_y^{(4)} = \int y''(x)^2 dx$; if $p = 2$ we have the regulariser $E_y^{(2)} = \int y'(x)^2 dx$, etc. Notice that the 'non–parametric' splines model can easily be put in an explicit parameterised representation. However, none of these splines models include 'knots'.

Figure 2.7c shows the evidence for data set X as a function of the number of terms, for $p = 0, 1, 2, 3, 4, 6$. Notice that in terms of Occam's razor, both cases discussed above occur: for $p = 0, 1$, as k increases, the model becomes more powerful and there is an Occam penalty. For $p = 3, 4, 6$, increasing k gives rise to no penalty. The case $p = 2$ seems to be on the fence between the two.

As p increases, the regulariser becomes more opposed to strong curvature. Once we reach $p = 6$, the model becomes less probable because the data demand sharp discontinuities. The evidence can choose the order of our splines regulariser for us. For this data set, it turns out that $p = 3$ is the most probable value of p, by a few multiples of e.

In passing, the radial basis function models described above can be transformed into the Fourier representation of the splines models. If the radial basis function kernel is $g(x)$ then the regulariser in the splines representation is $E_W = \sum \frac{1}{2} (w_{h(\cos)}^2 + w_{h(\sin)}^2) G_h^{-2}$, where G_h is the discrete Fourier transform of g.

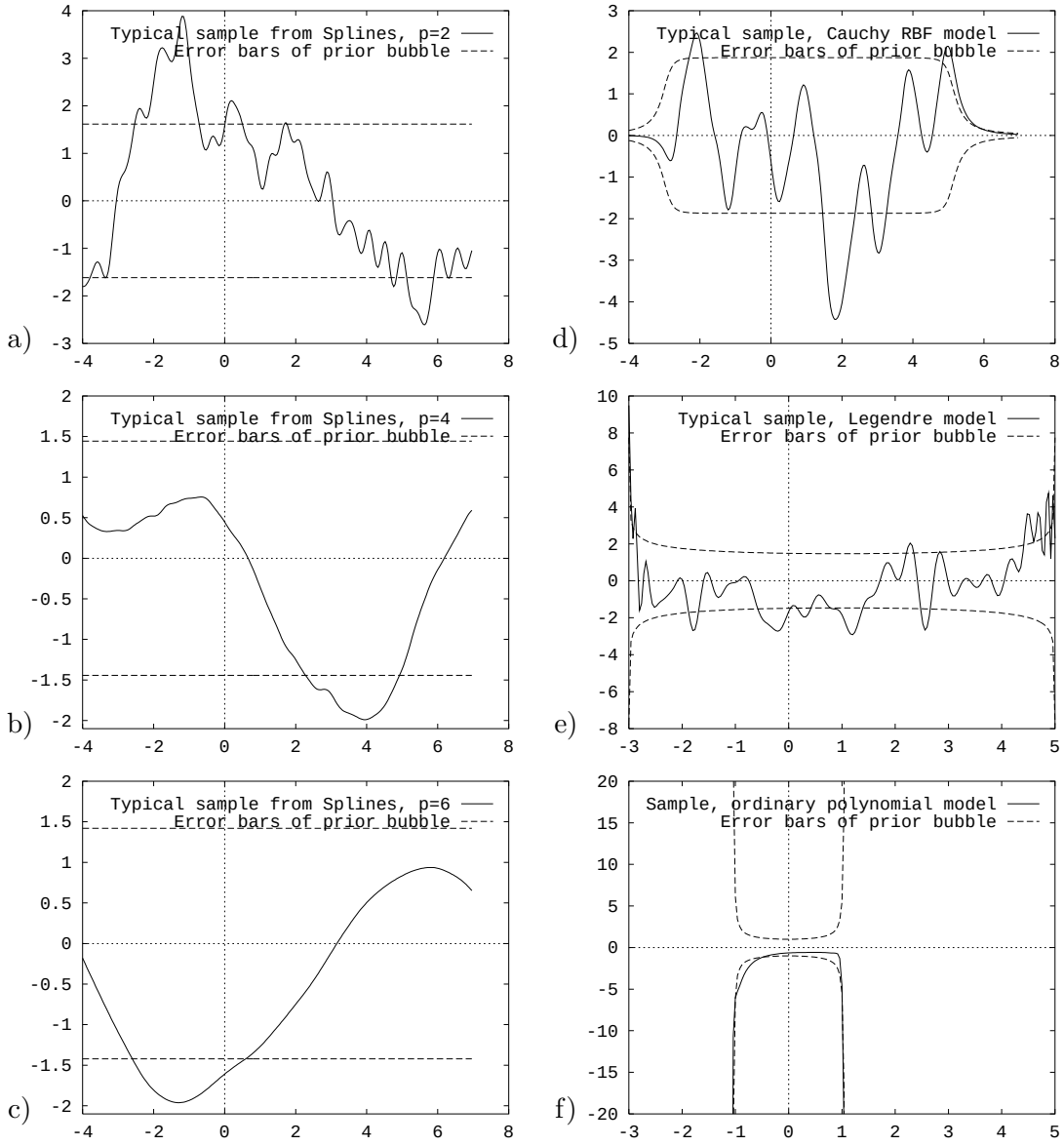

Figure 2.9: **Typical samples from the prior distributions of six models** This figure illustrates the character of some of the models used in this chapter. Each model was represented with 60 basis functions, and a typical sample from the prior distribution is shown. The regularisation constant was in each case set to make the typical magnitude of the interpolants similar. a) Splines, $p = 2$. b) Splines, $p = 4$ (cubic splines). c) Splines, $p = 6$. The splines were represented with a Fourier set with period 12.0. Notice how the spikiness of the typical sample decreases as the order of the spline increases. d) Cauchy radial basis functions. The basis functions were equally spaced from -3.0 to 5.0, and had scale $r = 0.2975$. e) Legendre polynomials. The polynomials were stretched so that the interval [-3.0,5.0] corresponds to the natural interval. Notice that the characteristic amplitude diverges at the boundaries, and the characteristic frequency of the typical sample also increases towards the boundaries. f) Ordinary polynomials. This figure illustrates what bad results can be obtained if a prior is carelessly assigned. A uniform prior over the coefficients of $y = \sum w_h x^h$ yields a highly non–uniform typical sample.

Results for a smoother data set

Figure 2.8 shows data set Y, which comes from a much smoother interpolant than data set X. Table 2.1 summarises the evidence for the alternative models. We can confirm that the evidence behaves in a reasonable manner by noting the following differences between data sets X and Y:

In the splines family, the most probable value of p has shifted upwards to the stiffer splines with $p=4-5$, as we would intuitively expect.

Legendre polynomials: an observant reader may have noticed that when data set X was modelled with Legendre polynomials, the most probable number of coefficients $k=38$ was suspiciously similar to the number of data points $N=37$. For data set Y, however, the most probable number of coefficients is 11, which confirms that the evidence does not always prefer the polynomial with $k=N$. Data set X behaved in this way because it is very poorly modelled by polynomials.

The Hermite function model, which was a poor model for data set X, is now the most probable, by a long way (over a million times more probable). The reason for this is that actually the data *were* generated from a Hermite function!

Why Bayes can't systematically reject the truth

Let us ask a sampling theory question: if one of the models we offer to Bayes is actually true, *i.e.*, it is the model from which the data were generated, then is it possible for Bayes to systematically (over the ensemble of possible data sets) prefer a false model? Clearly under a worst case analysis, a Bayesian's posterior may favour a false model. Furthermore, Skilling demonstrated that with some data sets a free form (maximum entropy) model can have greater evidence than the truth [73]; but is it possible for this to happen in the *typical* case, as Skilling seems to claim? I will show that the answer is no, that the effect that Skilling demonstrated cannot be systematic. To be precise, the expectation over possible data sets of the log evidence for the true model is greater than the expectation of the log evidence for any other fixed model [59].[15]

Proof. Suppose that the truth is actually $\mathcal{H}_1$. A single data set arrives and we compare the evidences for $\mathcal{H}_1$ and $\mathcal{H}_2$, a different fixed model. Both models may have free parameters, but this will be irrelevant to the argument. Intuitively we expect that the evidence for $\mathcal{H}_1$, $P(D|\mathcal{H}_1)$, should usually be greatest. Let us examine the difference in log evidence between $\mathcal{H}_1$ and $\mathcal{H}_2$. The expectation of this difference, given that $\mathcal{H}_1$ is true, is

$$\left\langle \log \frac{P(D|\mathcal{H}_1)}{P(D|\mathcal{H}_2)} \right\rangle = \int d^N D \, P(D|\mathcal{H}_1) \log \frac{P(D|\mathcal{H}_1)}{P(D|\mathcal{H}_2)}.$$

(Note that this integral implicitly integrates over all $\mathcal{H}_1$'s parameters according to their prior distribution under $\mathcal{H}_1$.) Now it is well known that for normalised p and q, $\int p \log \frac{p}{q}$ is minimised by setting $q = p$ (Gibbs' theorem). Therefore a distinct model $\mathcal{H}_2$ is never expected to *systematically* defeat the true model, for just the same reason that it is not wise to bet differently from the true odds. •

This result has two important implications. First, it gives us confidence in the ability

[15] Skilling's result presumably occurred because the particular parameter values of the true model that generated the data were not typical of the prior used when evaluating the evidence for that model. In such a case, the log evidence difference can show a transient bias against the true model, for small quantities of data; such biases are usually corrected by greater quantities of data.

of Bayesian methods on the average to identify the true model. Secondly, it provides a stringent test of numerical implementations of Bayesian model comparison. Imagine that we have written a program that evaluates the evidence for models $\mathcal{H}_1$ and $\mathcal{H}_2$. Then we can generate mock data from sources simulating $\mathcal{H}_1$ and $\mathcal{H}_2$ and evaluate the evidences. If there is any systematic bias, averaged over several mock data sets, for the estimated evidence to favour the false model, then we can be sure that our numerical implementation is not evaluating the evidence correctly.

This issue is illustrated using data set Y. The 'truth' is that this data set was actually generated from a quadratic Hermite function, $1.1(1-x+2x^2)e^{-x^2/2}$. By the above argument the evidence ought probably to favour the model 'the interpolant is a 3–coefficient Hermite function' over our other models. Table 2.1 shows the evidence for the true Hermite function model, and for other models. As already stated, the truth is indeed considerably more probable than the alternatives.

Having demonstrated that Bayes cannot systematically fail when one of the models is true, we now examine the way in which this framework can fail, if none of the models offered to Bayes is any good.

Comparison with 'generalisation error'

It is a popular and intuitive criterion for choosing between alternative interpolants (found using different models) to compare their errors on a test set that was not used to derive the interpolants. 'Cross–validation' is a more refined and more computationally expensive version of this same idea. How does this method relate to the evaluation of the evidence described in this chapter?

Figure 2.7c displayed the evidence for the family of spline interpolants. Figure 2.7d shows the corresponding test error, measured on a test set with size over twice as big (90) as the 'training' data set (37) used to determine the interpolant. A similar comparison was made in figure 2.5b. Note that the overall trends shown by the evidence are matched by trends in the test error (if you flip one graph upside down). Also, for this particular problem, the ranks of the alternative spline models under the evidence are similar to their ranks under the test error. And in figure 2.5b, the evidence maximum over α was surrounded by the test error minima. Thus, this suggests that the evidence might be a reliable predictor of generalisation ability. However, this is not necessarily the case. There are five reasons why the evidence and the test error might not be correlated.

First, the test error is a noisy quantity. It is necessary to devote large quantities of data to the test set to obtain a reasonable signal to noise ratio. In figure 2.5b more than twice as much data is in each test set but the difference in $\log \alpha$ between the two test error minima exceeds the size of the Bayesian confidence interval for $\log \alpha$.

Second, the model with greatest evidence is not expected to be the best model all the time — Bayesian inferences are uncertain. The whole point of Bayes is that it quantifies precisely those uncertainties: the relative values of the evidence for alternative models express the plausibility of the models, given the data and the underlying assumptions.

Third, there is more to the evidence than there is to the generalisation error. For example, imagine that for two models, the most probable interpolants happen to be identical. In this case, the two solutions will have the same generalisation error, but the evidence will not in general be the same: typically, the model that was *a priori* more complex will suffer a larger Occam factor and will have a smaller evidence.

Fourth, the test error is a measure of performance only of the single most probable interpolant: the evidence is a measure of plausibility of the entire posterior ensemble around

the best fit interpolant. Probably a stronger correlation between the evidence and the test statistic would be obtained if the test statistic used were the average of the test error over the posterior ensemble of solutions. This ensemble test error is not so easy to compute.

The fifth and most interesting reason why the evidence might not be correlated with the generalisation error is that there might be a flaw in the underlying assumptions such that the models being compared might all be poor models. If a poor regulariser is used, for example, one that is ill–matched to the statistics of the world, then the Bayesian choice of α will often not be the best in terms of generalisation error; Bayesian methods are more sensitive to poor model assumptions than, say, cross–validation [18, 27, 32]. Such a failure occurs in chapter 3. What is our attitude to such a failure of Bayesian prediction? The failure of the evidence does not mean that we should discard Bayes' rule and use the generalisation error as our criterion for choosing α. A failure is an opportunity to learn; a healthy scientist actively searches for such failures, because they yield insights into the defects of the current model. The detection of such a failure (by evaluating the generalisation error for example) motivates the search for new models which do not fail in this way; for example alternative regularisers can be tried until a model is found that makes the data more probable.

If one only uses the generalisation error as a criterion for model comparison, one is denied this mechanism for learning. The development of maximum entropy image deconvolution was held up for years because no–one used the Bayesian choice of α; once the Bayesian choice of α was used [27], the results obtained were most dissatisfactory, making clear what a poor regulariser was being used; this motivated an immediate search for alternative priors; the new, more probable priors discovered by this search are now at the heart of the state of the art in image deconvolution [88].

The similarity between regularisation and 'early stopping'

While an over–parameterised model is fitted to a data set using gradient descent on the data error, it is sometimes noted that the model's generalisation error passes through a minimum, rather than decreasing monotonically. This is known as 'over–learning' in the neural networks community, and some researchers advocate the use of 'early stopping', that is, stopping gradient descent before the data error minimum is reached, so as to try to obtain solutions with smaller generalisation error.

This author believes that 'over–learning' should be viewed as a symptom of a model ill–matched to the data set, and that the appropriate response is not to patch up a bad model, but rather to search for models which are better matched to our data. In particular, the use of models incorporating simple regularisers is expected to give results qualitatively similar to the results of early stopping. This can be seen by examining figure 2.6. The regulariser moves the minimum of the objective function from $\mathbf{w}_{\rm ML}$ to $\mathbf{w}_{\rm MP}$; as the strength of the regulariser α is increased, $\mathbf{w}_{\rm MP}$ follows a knee–shaped trajectory from $\mathbf{w}_{\rm ML}$ to the origin; a typical solution $\mathbf{w}_{\rm MP}$ is shown in figure 2.6. If on the other hand gradient descent on the likelihood (data error) is used, and if the typical initial condition is close to the origin, then gradient descent will follow a similar knee–shaped trajectory. Thus, qualitatively similar solutions are expected from increasingly early stopping and from increasingly strong regularisation with complete minimisation. Regularisation is to be preferred as a more robust, repeatable and comprehensible procedure.

Admitting neural networks into the canon of Bayesian interpolation models

Chapter 3 will discuss how to apply this Bayesian framework to feedforward neural networks. Preliminary results using these methods are included in table 2.1. Assuming that the approximations used were valid, it is interesting that the evidence for neural nets is actually good for both the spiky and the smooth data sets. Furthermore, neural nets, in spite of their arbitrariness, yield a relatively compact model, with fewer parameters needed than to specify the splines and radial basis function solutions.

2.7 Conclusions

The recently developed methods of Bayesian model comparison and regularisation have been presented. Models can be ranked by evaluating the evidence, a solely data–dependent measure which intuitively and consistently combines a model's ability to fit the data with its complexity. The precise posterior probabilities of the models also depend on the subjective priors that we assign to them, but these terms are typically overwhelmed by the evidence.

Regularising constants are set by maximising the evidence. For many regularisation problems, the theory of the number of well–measured parameters makes it possible to perform this optimisation on–line.

In the interpolation examples discussed, the evidence was used to set the number of basis functions k in a polynomial model; to set the characteristic size r in a radial basis function model; to choose the order p of the regulariser for a spline model; and to rank all these different models in the light of the data.

Further work is needed to formalise the relationship of this framework to the pragmatic model comparison technique of cross–validation. Using the two techniques in parallel, it is possible to detect flaws in the underlying assumptions implicit in the data models being used. Such failures direct our search for superior models, providing a powerful tool for human learning.

There are thousands of data modelling tasks waiting for the evidence to be evaluated. It will be exciting to see how much we can learn when this is done.

Chapter 3

A Practical Bayesian Framework for Backpropagation Networks

Abstract

A quantitative and practical Bayesian framework is described for learning of mappings in feedforward networks. The framework makes possible: (1) objective comparisons between solutions using alternative network architectures; (2) objective stopping rules for network pruning or growing procedures; (3) objective choice of magnitude and type of weight decay terms or additive regularisers (for penalising large weights, etc.); (4) a measure of the effective number of well–determined parameters in a model; (5) quantified estimates of the error bars on network parameters and on network output; (6) objective comparisons with alternative learning and interpolation models such as splines and radial basis functions. The Bayesian 'evidence' automatically embodies 'Occam's razor', penalising over–flexible and over–complex models. The Bayesian approach helps detect poor underlying assumptions in learning models. For learning models well matched to a problem, a good correlation between generalisation ability and the Bayesian evidence is obtained.

3.1 The gaps in backprop

There are many knobs on the black box of 'backprop' (learning by back–propagation of errors [66]). Generally these knobs are set by rules of thumb, trial and error, and the use of reserved test data to assess generalisation ability (or more sophisticated cross–validation). The knobs fall into two classes: (1) parameters which change the effective learning model, for example, number of hidden units, and weight decay terms; and (2) parameters concerned with function optimisation technique, for example, 'momentum' terms. This chapter is concerned with making objective the choice of the parameters in the first class, and with ranking alternative solutions to a learning problem in a way which makes full use of all the available data. Bayesian techniques will be described which are both theoretically well–founded and practically implementable.

Let us review the basic framework for learning in networks, then discuss the points at which objective techniques are needed. The training set for the mapping to be learned is a set of input–target pairs $D = \{\mathbf{x}^m, \mathbf{t}^m\}$, where m is a label running over the pairs. A

[0]Chapter 3 of Ph.D. thesis 'Bayesian Methods for Adaptive Models' by David MacKay, California Institute of Technology, submitted December 10 1991.

neural network architecture $\mathcal{A}$ is invented, consisting of a specification of the number of layers, the number of units in each layer, the type of activation function performed by each unit, and the available connections between the units. If a set of values $\mathbf{w}$ is assigned to the connections in the network, the network defines a mapping $\mathbf{y}(\mathbf{x};\mathbf{w},\mathcal{A})$ from the input activities $\mathbf{x}$ to the output activities $\mathbf{y}$.[1] The distance of this mapping to the training set is measured by some error function; for example the error for the entire data set is commonly taken to be

$$E_D(D\,|\mathbf{w},\mathcal{A}) = \sum_m \frac{1}{2}\Big(\mathbf{y}(\mathbf{x}^m;\mathbf{w},\mathcal{A}) - \mathbf{t}^m\Big)^2. \tag{3.1}$$

The task of 'learning' is to find a set of connections $\mathbf{w}$ which gives a mapping which fits the training set well, *i.e.*, has small error E_D; it is also hoped that the learned connections will 'generalise' well to new examples. Plain backpropagation learns by performing gradient descent on E_D in $\mathbf{w}$–space. Modifications include the addition of a 'momentum' term, and the inclusion of noise in the descent process. More efficient optimisation techniques may also be used, such as conjugate gradients or variable metric methods. This chapter will not discuss computational modifications concerned only with speeding the optimisation. It will address however those modifications to the plain backprop algorithm which implicitly or explicitly modify the objective function, with decay terms or regularisers.

It is moderately common for extra regularising terms $E_W(\mathbf{w})$ to be added to E_D; for example, terms which penalise large weights may be introduced, in the hope of achieving a smoother or simpler mapping [33, 39, 57, 67, 87]. Some of the 'hints' in [2] also fall into the category of additive weight–dependent energies. A sample weight energy term is:

$$E_W(\mathbf{w}|\mathcal{A},\mathcal{R}) = \sum_i \frac{1}{2} w_i^2. \tag{3.2}$$

The weight energy may be implicit, for example, 'weight decay' (subtraction of a multiple of $\mathbf{w}$ in the weight change rule) corresponds to the energy in (3.2). Gradient–based optimisation is then used to minimise the combined function:

$$M = \alpha E_W(\mathbf{w}|\mathcal{A},\mathcal{R}) + \beta E_D(D\,|\mathbf{w},\mathcal{A}), \tag{3.3}$$

where α and β are 'black box' parameters.

The constant α should not be confused with the 'momentum' parameter sometimes introduced into backprop; in the present context α is a decay rate or regularising constant. Also note that α should not be viewed as causing 'forgetting'; E_D is defined as the error on the entire data set, so gradient descent on M treats all data points equally irrespective of the order in which they were acquired.

What is lacking

The above procedures include a host of free parameters such as the choice of neural network architecture, and of the regularising constant α. There are not yet established ways of objectively setting these parameters, though there are many rules of thumb (see [39, 87] for examples).

One popular way of comparing networks trained with different parameter values is to assess their performance by measuring the error on an unseen test set or by similar cross–validation techniques. The data are divided into two sets, a training set which is used to

[1] The framework developed in this chapter will apply not only to networks composed of 'neurons', but to any regression model for which we can compute the derivatives of the outputs with respect to the parameters, $\partial\mathbf{y}(\mathbf{x};\mathbf{w},\mathcal{A})/\partial\mathbf{w}$.

optimise the parameters $\mathbf{w}$ of the network, and a test set, which is used to optimise control parameters such as α and the architecture $\mathcal{A}$. However, the utility of these techniques in determining values for the parameters α and β or for comparing alternative network solutions, etc., is limited because a large test set may be needed to reduce the signal to noise ratio in the test error, and cross–validation is computationally demanding. Furthermore, if there are several parameters like α and β, it is out of the question to optimise such parameters by repeating the learning with all possible values of these parameters and using a test set. Such parameters must be optimised on line.

It is therefore interesting to study objective criteria for setting free parameters and comparing alternative solutions, which depend only on the data set used for the training. Such criteria will prove especially important in applications where the total amount of data is limited, so that one doesn't want to sacrifice good data for use as a test set. Rather, we wish to find a way to use *all* our data in the process of optimising the parameters $\mathbf{w}$ *and* in the process of optimising control parameters like α and $\mathcal{A}$.

This chapter will describe practical Bayesian methods for filling the following holes in the neural network framework just described:

1. **Objective criteria for comparing alternative neural network solutions, in particular with different architectures $\mathcal{A}$.**

   Given a single architecture $\mathcal{A}$, there may be more than one minimum of the objective function M. If there is a large disparity in M between the minima then it is plausible to choose the solution with smallest M. But where the difference is not so great it is desirable to be able to assign an objective preference to the alternatives.

   It is also desirable to be able to assign preferences to neural network solutions using different numbers of hidden units, and different activation functions. Here there is an 'Occam's razor' problem: the more free parameters a model has, the smaller the data error E_D it can achieve. So we cannot simply choose the architecture with smallest data error. That would lead us to an over–complex network which generalises poorly. The use of weight decay does not fully alleviate this problem; networks with too many hidden units still generalise worse, even if weight decay is used (see section 3.4).

2. **Objective criteria for setting the decay rate α.** As in the choice of $\mathcal{A}$ above, there is an 'Occam's razor' problem: a small value of α in equation (3.3) allows the weights to become large and overfit the noise in the data. This leads to a small value of the data error E_D (and a small value of M), so we cannot base our choice of α only on E_D or M. The Bayesian solution presented here can be implemented on–line, *i.e.*, it is not necessary to do multiple learning runs with different values of α in order to find the best.

3. **Objective choice of regularising function E_W.**

4. **Objective criteria for choosing between a neural network solution and a solution using a different learning or interpolation model**, for example, splines or radial basis functions.

The probability connection

Tishby *et al.* [82] introduced a probabilistic view of learning which is an important step towards solving the problems listed above. The idea is to force a probabilistic interpretation onto the neural network technique so as to be able to make objective statements. This

interpretation does not involve the addition of any new arbitrary functions or parameters, but it involves assigning a meaning to the functions and parameters that are already used.

My work is based on the same probabilistic framework, and extends it using concepts and techniques adapted from Gull and Skilling's Bayesian image reconstruction methods [27]. This chapter also adopts a shift in emphasis from Tishby *et al.*'s paper. Their work concentrated on predicting the average generalisation ability of one network trained on a task drawn from a known prior ensemble of tasks. This is called *forward probability*. In this thesis the emphasis will be on quantifying the relative plausibilities of many alternative solutions to an interpolation or classification task; that task is defined by a single data set produced by the real world, and we do not know the prior ensemble from which the task comes. This is called *inverse probability*. This thesis avoids using the language of statistical physics, partly so as to avoid concepts that would sound strange in that language; for example 'the probability distribution of the temperature' is unfamiliar in physics, but 'the probability distribution of the noise variance' is its innocent counterpart in literal terms.

Let us now review the probabilistic interpretation of network learning.

- **Likelihood.** A network with specified architecture $\mathcal{A}$ and connections $\mathbf{w}$ is viewed as making predictions about the target outputs as a function of input $\mathbf{x}$ in accordance with the probability distribution:

$$P(\mathbf{t}^m|\mathbf{x}^m, \mathbf{w}, \beta, \mathcal{A}, \mathcal{N}) = \frac{\exp(-\beta E(\mathbf{t}^m|\mathbf{x}^m, \mathbf{w}, \mathcal{A}))}{Z_m(\beta)}, \tag{3.4}$$

  where $Z_m(\beta) = \int d\mathbf{t} \, \exp(-\beta E)$. E is the error for a single datum, and β is a measure of the presumed noise included in $\mathbf{t}$. If E is the quadratic error function then this corresponds to the assumption that $\mathbf{t}$ includes additive Gaussian noise with variance $\sigma_\nu^2 = 1/\beta$. The symbol $\mathcal{N}$ denotes the implicit noise model.

- **Prior.** A prior probability is assigned to alternative network connection strengths $\mathbf{w}$, written in the form:

$$P(\mathbf{w}|\alpha, \mathcal{A}, \mathcal{R}) = \frac{\exp(-\alpha E_W(\mathbf{w}|\mathcal{A}, \mathcal{R}))}{Z_W(\alpha)}, \tag{3.5}$$

  where $Z_W = \int d^k\mathbf{w} \, \exp(-\alpha E_W)$. Here α is a measure of the characteristic expected connection magnitude. If E_W is quadratic as specified in equation (3.2) then weights are expected to come from a Gaussian with zero mean and variance $\sigma_W^2 = 1/\alpha$. Alternative 'regularisers' $\mathcal{R}$ (each using a different energy function E_W) implicitly correspond to alternative hypotheses about the statistics of the environment.

- The **posterior probability** of the network connections $\mathbf{w}$ is then:

$$P(\mathbf{w}|D, \alpha, \beta, \mathcal{A}, \mathcal{N}, \mathcal{R}) = \frac{\exp(-\alpha E_W - \beta E_D)}{Z_M(\alpha, \beta)}, \tag{3.6}$$

  where $Z_M(\alpha, \beta) = \int d^k\mathbf{w} \, \exp(-\alpha E_W - \beta E_D)$. Notice that the exponent in this expression is the same as (minus) the objective function M defined in (3.3).

So under this framework, minimisation of $M = \alpha E_W + \beta E_D$ is identical to finding the (locally) most probable parameters $\mathbf{w}_{\rm MP}$; minimisation of E_D alone is identical to finding the maximum likelihood parameters $\mathbf{w}_{\rm ML}$. Thus an interpretation has been given to back-propagation's energy functions E_D and E_W, and to the parameters α and β. It should

be emphasised that 'the probability of the connections $\mathbf{w}$' is a measure of *plausibility* that the model's parameters should have a specified value $\mathbf{w}$; this has nothing to do with the probability that a particular algorithm might converge to $\mathbf{w}$.

This framework offers some partial enhancements for backprop methods: The work of Levin *et al.* [43] makes it possible to predict the average generalisation ability of neural networks trained on one of a defined class of problems. However, it is not clear whether this will lead to a practical technique for choosing between alternative network architectures for real data sets.

Le Cun *et al.* have demonstrated how to estimate the 'saliency' of a weight, which is the change in M when the weight is deleted [41]. They have used this measure successfully to simplify large neural networks. However, no stopping rule for weight deletion was offered other than measuring performance on a test set.

Also Denker and Le Cun demonstrated how the Hessian of M can be used to assign error bars to the parameters of a network and to its outputs [19]. However, these error bars can only be quantified once β is quantified, and how to do this without prior knowledge or extra data has not been demonstrated. In fact β can be estimated from the training data alone.

3.2 Review of Bayesian regularisation and model comparison

In chapter 2 it was demonstrated how the control parameters α and β are assigned by Bayes, and how alternative interpolation models $\mathcal{H} = \{\mathcal{A}, \mathcal{N}, \mathcal{R}\}$ can be compared. It was noted there that it is not satisfactory to optimise α and β by finding the joint maximum likelihood value of $\mathbf{w}, \alpha, \beta$; the likelihood has a skew peak whose maximum is not located at the most probable values of the control parameters. Chapter 2 also reviewed how the Bayesian choice of α and β is neatly expressed in terms of a measure of the number of well–determined parameters in a model, γ. However that chapter assumed that $M(\mathbf{w})$ only has one significant minimum which was well approximated as quadratic. (All the interpolation models discussed in chapter 2 can be interpreted as two–layer networks with a fixed non–linear first layer and adaptive linear second layer.) In this section I briefly review the Bayesian framework, retaining that assumption. The following section will then discuss how the framework can be modified to handle neural networks, where the landscape of $M(\mathbf{w})$ is certainly not quadratic.

Determination of α and β

By Bayes' rule, the posterior probability for these parameters is:

$$P(\alpha, \beta \,|D, \mathcal{H}) = \frac{P(D|\alpha, \beta, \mathcal{H}) P(\alpha, \beta|\mathcal{H})}{P(D|\mathcal{H})}. \tag{3.7}$$

Now if we assign a uniform prior to (α, β), the quantity of interest for assigning preferences to (α, β) is the first term on the right hand side, the **evidence** for α, β, which can be written as[2]

$$P(D|\alpha, \beta, \mathcal{H}) = \frac{Z_M(\alpha, \beta)}{Z_W(\alpha) Z_D(\beta)}, \tag{3.8}$$

[2]The same notation, and the same abuses thereof, will be used as in chapter 2.

where Z_M and Z_W were defined earlier and $Z_D = \int d^N D\, e^{-\beta E_D}$.

Let us use the simple quadratic energy functions defined in equations (3.1,3.2). This makes the analysis easier, but more complex cases can still in principle be handled by the same approach. Let the number of degrees of freedom in the data set, *i.e.*, the number of output units times the number of data pairs, be N, and let the number of free parameters, *i.e.*, the dimension of $\mathbf{w}$, be k. Then we can immediately evaluate the Gaussian integrals Z_D and Z_W: $Z_D = (2\pi/\beta)^{N/2}$, and $Z_W = (2\pi/\alpha)^{k/2}$. Now we want to find $Z_M(\alpha, \beta) = \int d^k\mathbf{w}\, \exp(-M(\mathbf{w}, \alpha, \beta))$. Supposing for now that M has a single minimum as a function of $\mathbf{w}$, at $\mathbf{w}_{\rm MP}$, and assuming we can locally approximate M as quadratic there, the integral Z_M is approximated by:

$$Z_M \simeq e^{-M(\mathbf{w}_{\rm MP})}(2\pi)^{k/2}\det{}^{-\frac{1}{2}}\mathbf{A}, \tag{3.9}$$

where $\mathbf{A} = \nabla\nabla M$ is the Hessian of M evaluated at $\mathbf{w}_{\rm MP}$.

The maximum of $P(D|\alpha, \beta, \mathcal{H})$ has the following useful properties:

$$\chi^2_W \equiv 2\alpha E_W = \gamma \tag{3.10}$$

$$\chi^2_D \equiv 2\beta E_D = N - \gamma, \tag{3.11}$$

where γ is the effective number of parameters determined by the data,

$$\gamma = \sum_{a=1}^{k} \frac{\lambda_a}{\lambda_a + \alpha}, \tag{3.12}$$

where λ_a are the eigenvalues of the quadratic form βE_D in the natural basis of E_W.

Comparison of different models

To rank alternative architectures, noise models, and penalty functions E_W in the light of the data, we simply evaluate the evidence for $\mathcal{H} = \{\mathcal{A}, \mathcal{N}, \mathcal{R}\}$, $P(D\,|\mathcal{H})$, which appeared as the normalising constant in (3.7). Integrating the evidence for (α, β), we have:

$$P(D|\mathcal{H}) = \int P(D|\alpha, \beta, \mathcal{H})P(\alpha, \beta|\mathcal{H})\, d\alpha\, d\beta. \tag{3.13}$$

The evidence is the Bayesian's transportable quantity for comparing models in the light of the data.

3.3 Adapting the framework

For neural networks, $M(\mathbf{w})$ is not quadratic. Indeed it is well known that M typically has many local minima. And if the network has a symmetry under permutation of its parameters, then we know that $M(\mathbf{w})$ must share that symmetry, so that every single minimum belongs to a family of symmetric minima of M. For example if there are H hidden units in a single layer then each non–degenerate minimum is in a family of size $g = H!\, 2^H$. Now it may be the case that the significant minima of M are locally quadratic, so we might be able to evaluate Z_M by evaluating (3.9) at each significant minimum and adding up the Z_Ms; but the number of those minima is unknown, and this approach to evaluating Z_M would seem dubious.

Luckily however, we do not actually want to evaluate Z_M. We would need to evaluate Z_M in order to assign a posterior probability over α, β for an entire model, and to evaluate

the evidence for alternative entire models. This is not quite what we wish to do: when we use a neural network to perform a mapping, we typically only implement one neural network at a time, and this network will have its parameters set to a *particular solution* of the learning problem. Therefore the alternatives we wish to rank are the different solutions of the learning problem, *i.e.*, the different minima of M. We would only want the evidence as a function of the number of hidden units if we were somehow able to simultaneously implement the entire posterior ensemble of networks for one number of hidden units. Similarly, we do not want the posterior over α, β for the entire posterior ensemble; rather, it is reasonable to allow each solution (each minimum of M) to choose its own optimal value for these parameters. The same method of chopping up a complex model space is used in the unsupervised classification system, AutoClass [31].

Having adopted this slight shift in objective, it turns out that to set α and β and to compare alternative solutions to a learning problem, the integral we now need to evaluate is a local version of Z_M. Assume that the posterior probability consists of well separated islands in parameter space each centred on a minimum of M. We wish to evaluate how much posterior probability mass is in each of these islands. Consider a minimum located at $\mathbf{w}^*$, and define a solution $S_{\mathbf{w}^*}$ as the ensemble of networks in the neighbourhood of $\mathbf{w}^*$, and all symmetric permutations of that ensemble. Let us evaluate the posterior probability for alternative solutions $S_{\mathbf{w}^*}$, and the parameters α and β:

$$P(S_{\mathbf{w}^*}, \alpha, \beta, \mathcal{H}|D) \propto g \frac{Z_M^*(\mathbf{w}^*, \alpha, \beta)}{Z_W(\alpha) Z_D(\beta)} P(\alpha, \beta|\mathcal{H}) P(\mathcal{H}), \tag{3.14}$$

where g is the permutation factor, and $Z_M^*(\mathbf{w}^*, \alpha, \beta) = \int_{S_{\mathbf{w}^*}} d^k\mathbf{w} \, \exp(-M(\mathbf{w}, \alpha, \beta))$, where the integral is performed only over the neighbourhood of the minimum at $\mathbf{w}^*$. I will refer to the quantity $g \frac{Z_M^*(\mathbf{w}^*, \alpha, \beta)}{Z_W(\alpha) Z_D(\beta)}$ as the evidence for $\alpha, \beta, S_{\mathbf{w}^*}$. The parameters α and β will be chosen to maximise this evidence. Then the quantity we want to evaluate to compare alternative solutions is the evidence[3] for $S_{\mathbf{w}^*}$,

$$P(D, S_{\mathbf{w}^*}|\mathcal{H}) = \int g \frac{Z_M^*(\mathbf{w}^*, \alpha, \beta)}{Z_W(\alpha) Z_D(\beta)} P(\alpha, \beta|\mathcal{H}) \, d\alpha \, d\beta. \tag{3.15}$$

This thesis uses the Gaussian approximation for Z_M^*:

$$Z_M^* \simeq e^{-M(\mathbf{w}^*)} (2\pi)^{k/2} \det{}^{-\frac{1}{2}} \mathbf{A}, \tag{3.16}$$

where $\mathbf{A} = \nabla\nabla M$ is the Hessian of M evaluated at $\mathbf{w}^*$. For general α and β this approximation is probably unacceptable; however we only need it to be accurate for the small range of α and β close to their most probable value. The regime in which this approximation will definitely break down is when the number of constraints, N, is small relative to the number of free parameters, k. For large N/k the central limit theorem encourages us to use the Gaussian approximation [84]. It is a matter for further research to establish how large N/k must be for this approximation to be reliable.

What obstacles remain to prevent us from evaluating the local Z_M^*? We need to evaluate or approximate the inverse Hessian of M, and we need to evaluate or approximate its determinant and/or trace [49].

[3]Bayesian model comparison is performed by evaluating and comparing the evidence for alternative models. Gull and Skilling defined the evidence for a model $\mathcal{H}$ to be $P(D|\mathcal{H})$. The existence of multiple minima in neural network parameter space complicates model comparison. The quantity in (3.15) is not $P(D|S_{\mathbf{w}}, \mathcal{H})$ (it includes the prior for $S_{\mathbf{w}}|\mathcal{H}$), but I have called it the evidence because it is the quantity we should evaluate to compare alternative solutions with each other and with other models.

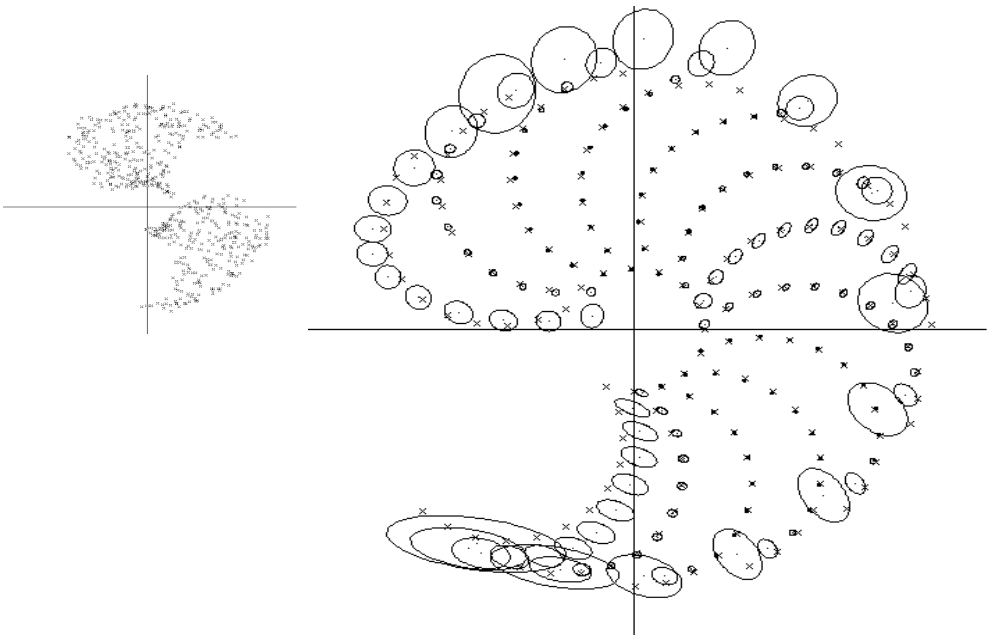

Figure 3.1: **Typical neural network output.** (Inset – training set)
This is the output space (y_a, y_b) of the network. The target outputs are displayed as small x's, and the output of the network with 1σ error bars is shown as a a dot surrounded by an ellipse. The network was trained on samples in two regions in the lower and upper half planes (inset). The outputs illustrated here are for inputs extending a short distance outside the two training regions, and bridging the gap between them. Notice that the error bars get much larger around the perimeter. They also increase slightly in the gap between the training regions. These pleasing properties are not obtained if the diagonal Hessian approximation of [19] is used. The above solution was created by a three layer network with 19 hidden units.

Denker *et al.* have already discussed how to approximate the Hessian of E_D for the purpose of evaluating weight saliency and for assigning error bars to weights and network outputs [19, 41]. The Hessian can be evaluated in the same way that backpropagation evaluates ∇E_D (see [9] for a complete algorithm and the appendix of this chapter for a useful approximation). Alternatively $\mathbf{A}$ can be evaluated by numerical methods, for example second differences. A third option: if variable metric methods are used to minimise M instead of gradient descent, then the inverse Hessian is automatically generated during the search for the minimum. It is important, for the success of this Bayesian method, that the off–diagonal terms of the Hessian should be evaluated. Denker *et al.*'s method can do this without any additional complexity. The diagonal approximation is no good because of the strong posterior correlations in the parameters.

3.4 Demonstration

This demonstration examines the evidence for various neural net solutions to a small interpolation problem, the mapping for a two joint robot arm,

$$(\theta_1, \theta_2) \rightarrow (y_a, y_b) = (r_1 \cos\theta_1 + r_2 \cos(\theta_1 + \theta_2), r_1 \sin\theta_1 + r_2 \sin(\theta_1 + \theta_2)).$$

For the training set I used $r_1 = 2.0$ and $r_2 = 1.3$, random samples from a restricted range of (θ_1, θ_2) were made, and Gaussian noise of magnitude 0.05 was added to the outputs. The

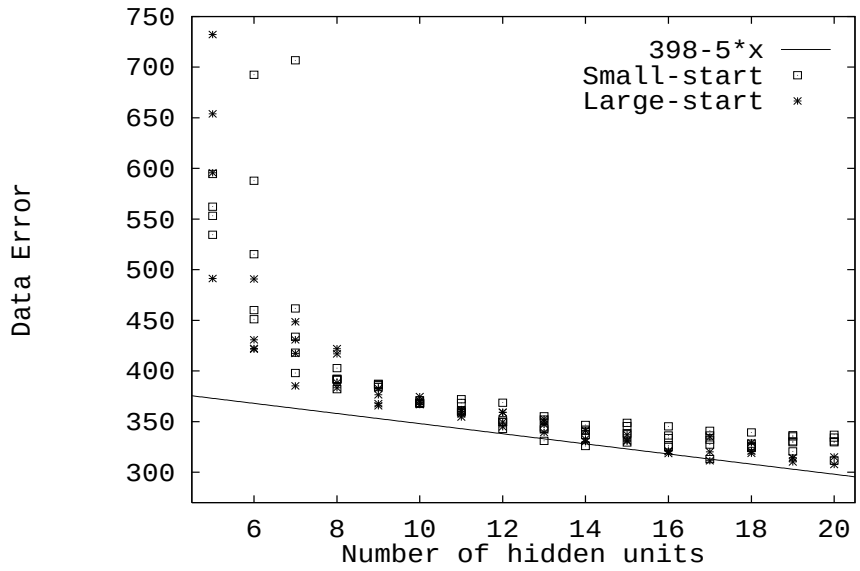

Figure 3.2: **Data error versus number of hidden units.**
Each point represents one converged neural network, trained on a 200 i/o pair training set. Each neural net was initialised with different random weights and with a different initial value of $\sigma_W^2 = 1/\alpha$. The two point–styles correspond to small and large initial values for σ_W. The error is shown in dimensionless χ^2 units such that the expectation of error relative to the truth is 400 ± 20. The solid line is $400 - k$, where k is the number of free parameters.

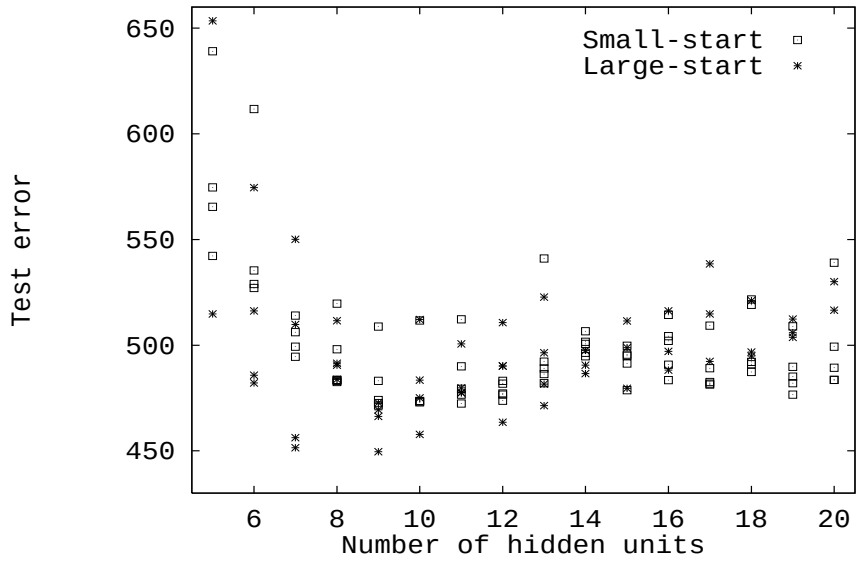

Figure 3.3: **Test error versus number of hidden units**
The training set and test set both had 200 data points. The test error for solutions found using the first regulariser is shown in dimensionless χ^2 units such that the expectation of error relative to the truth is 400 ± 20.

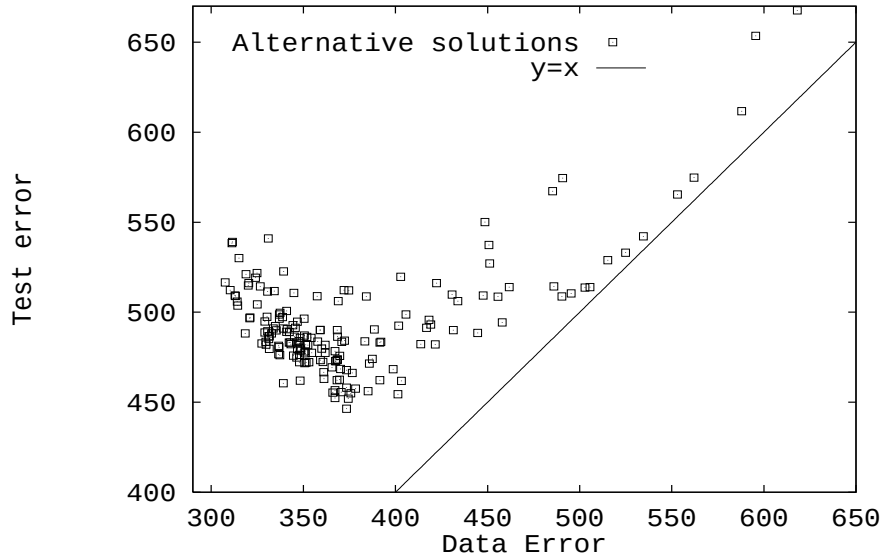

Figure 3.4: **Test error vs. data error.**
Each point represents the performance of a single trained neural network on the training set and on the test set. The horizontal axis displays the data error, that is, the network's performance on the training data. A small value of data error corresponds to a network that has learnt the training data well. The vertical axis displays the test error, that is, how well each network generalises to new examples. The smaller the test error, the better the generalisation ability of the network.

This graph illustrates the 'Occam problem' — the best generalisation is not achieved by the models which fit the training data best.

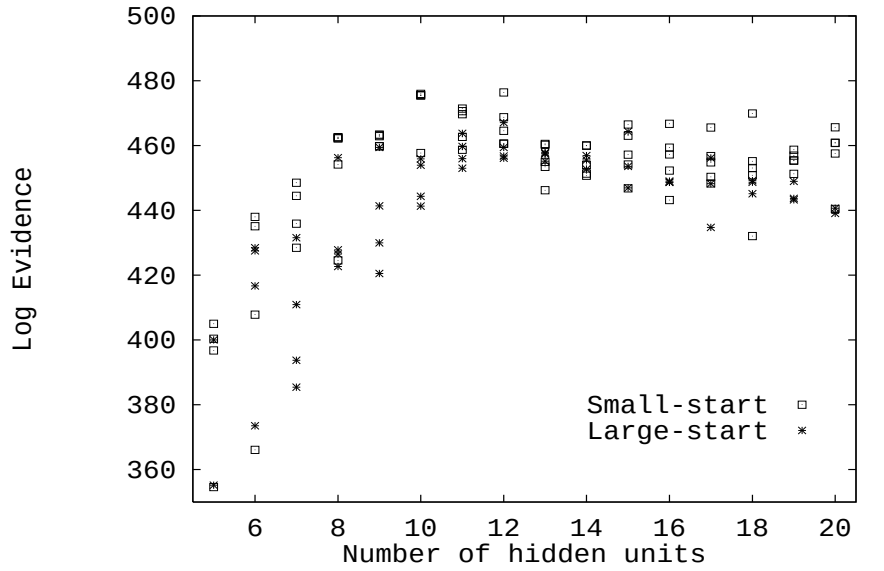

Figure 3.5: **Log evidence for solutions using the first regulariser.**
For each solution, the evidence was evaluated. Notice that an evidence maximum is achieved by neural network solutions using 10, 11 and 12 hidden units. For more than ~ 19 hidden units, the quadratic approximations used to evaluate the evidence are believed to break down. The number of data points N is 400 (*i.e.*, 200 i/o pairs); *c.f.* number of parameters in a net with 20 hidden units = 102.

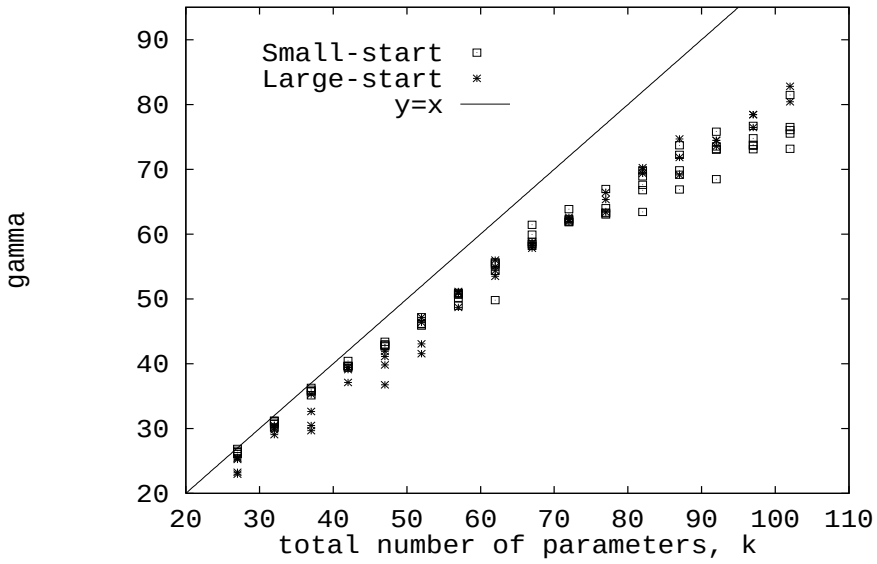

Figure 3.6: **The number of well–determined parameters.**
This figure displays γ as a function of k, for the same network solutions as in figure 3.5.

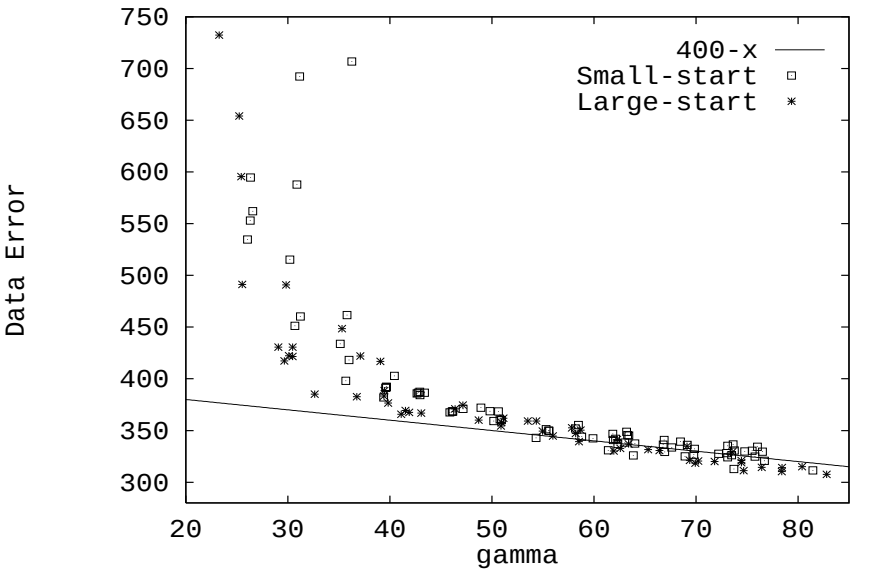

Figure 3.7: **Data misfit versus γ.**
This figure shows χ^2_D against γ, and a line of gradient -1. Towards the right, the data's misfit χ^2_D is reduced by 1 for every well–measured parameter. When the model has too few parameters however (towards the left), the misfit gets worse at a greater rate.

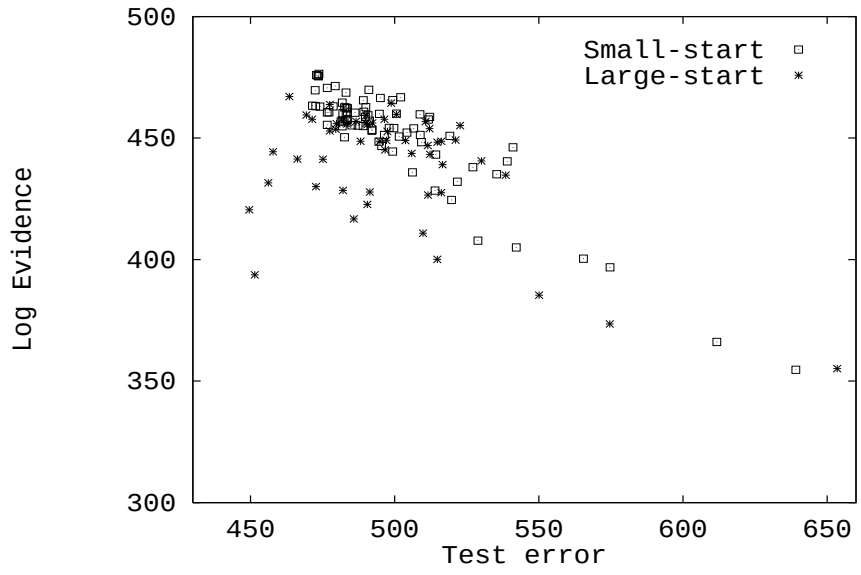

Figure 3.8: **Log evidence versus test error for the first regulariser**
The desired correlation between the evidence and the test error has negative slope. A significant number of points on the lower left violate this desired trend, so we have a failure of Bayesian prediction. The points which violate the trend are networks in which there is a significant difference in typical weight magnitude between the two layers. They are all networks whose learning was initialised with a large value of σ_w. The first regulariser is ill–matched to such networks, and the low evidence is a reflection of this poor prior hypothesis.

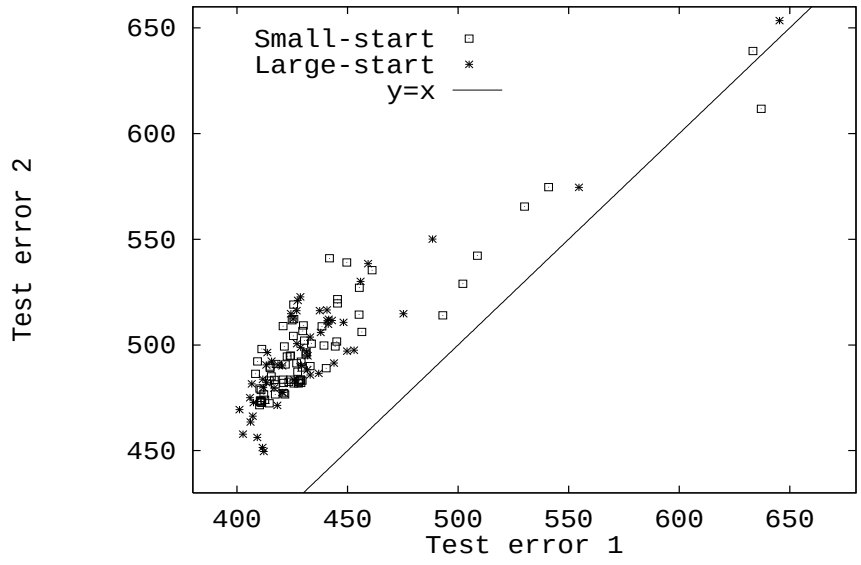

Figure 3.9: **Comparison of two test errors.**
This figure illustrates how noisy a performance measure the test error is. Each point compares the error of a trained network on two different test sets. Both test sets consist of 200 data points from the same distribution as the training set.

neural nets used had one hidden layer of sigmoid units and linear output units. During optimisation, the regulariser (3.2) was used initially, and an alternative regulariser was introduced later; β was fixed to its true value (to enable demonstration of the properties of the quantity γ), and α was allowed to adapt to its locally most probable value.

Figure 3.1 illustrates the performance of a typical neural network trained in this way. Each output is accompanied by error bars evaluated using Denker *et al.*'s method, *including off–diagonal Hessian terms.* If β had not been known in advance, it could have been inferred from the data using equation (3.11). For the solution displayed, the model's estimate of β in fact differed negligibly from the true value, so the displayed error bars are the same as if β had been inferred from the data.

Figure 3.2 shows the data misfit versus the number of hidden units. Notice that, as expected, the data error tends to decrease monotonically with increasing number of parameters. Figure 3.3 shows the error of these same solutions on an unseen test set, which does not show the same trend as the data error. This Occam problem is illustrated by figure 3.4, which compares the test error with the data error. The data misfit cannot serve as a criterion for choosing between solutions.

Figure 3.5 shows the evidence for about 100 different solutions using different numbers of hidden units. Notice how the evidence maximum has the characteristic shape of an 'Occam hill' — steep on the side with too few parameters, and shallow on the side with too many parameters. The quadratic approximations break down when the number of parameters becomes too big compared with the number of data points.

The next figures introduce the quantity γ, discussed in chapter 2, the number of well–measured parameters. In cases where the evaluation of the evidence proves difficult, it may be that γ will serve as a useful tool. For example, sampling theory predicts that the addition of redundant parameters to a model should reduce χ^2_D by one unit per well–measured parameter; a stopping criterion could detect the point at which, as parameters are deleted, χ^2_D started to increase faster than with gradient 1 with decreasing γ (figure 3.7).[4] This use of γ requires prior knowledge of the noise level β; that is why β was fixed to its known value for these demonstrations.

Now the question is how good a predictor of network quality the evidence is. The fact that the evidence has a maximum at a reasonable number of hidden units is promising. A comparison with figure 3.3 shows that the performance of the solutions on an unseen test set has similar overall structure to the evidence. However, figure 3.8 shows the evidence against the performance on a test set, and it can be seen that a significant number of solutions with poor evidence actually perform well on the test set. Something is wrong! Let us discuss the relationship between the evidence and generalisation ability. We will then return to the failure in figure 3.8 and see that it is rectified by the development of new, more probable regularisers.

Relation to 'generalisation error'

What is the relationship between the evidence and the generalisation error (or its close relative, cross–validation)? A correlation between the two is certainly expected. But the evidence is not necessarily a good predictor of generalisation error (see discussion in chapter 2). First, as illustrated in figure 3.9, the error on a test set is a noisy quantity, and a lot of data has to be devoted to the test set to get an acceptable signal to noise ratio. Furthermore, imagine that two models have generated solutions to an interpolation problem, and that

[4] This suggestion is closely related to Moody's 'generalised prediction error', GPE $= \frac{1}{N}(\chi^2_D + 2\gamma)$ [54].

their two most probable interpolants are completely identical. In this case, the generalisation error for the two solutions must be the same, but the evidence will not in general be the same: typically, the model that was *a priori* more complex will suffer a larger Occam factor and will have smaller evidence. Also, the evidence is a measure of plausibility of the whole ensemble of networks about the optimum, not just the optimal network. Thus there is more to the evidence than there is to the generalisation error.

What if the Bayesian method fails?

I do not want to dismiss the utility of the generalisation error: it can be important for detecting failures of the model being used. For example, if we obtain a poor correlation between the evidence and the generalisation error, such that Bayes fails to assign a strong preference to solutions which actually perform well on test data, then we are able to detect and attempt to correct such failures.

A failure indicates one of two things, and in either case we are able to learn and improve: either numerical inaccuracies in the evaluation of the probabilities caused the failure; or else the alternative models which were offered to Bayes were a poor selection, ill–matched to the real world (for example, using inappropriate regularisers). When such a failure is detected, it prompts us to examine our models and try to discover the implicit assumptions in the model which the data didn't agree with; alternative models can be tried until one is found that makes the data more probable.

We have just met exactly such a failure. Let us now establish what assumption in our model caused this failure and *learn* from it. Note that this mechanism for human learning is not available to those who just use the test error as their performance criterion. Going by the test error alone, there would have been no indication that there was a serious mismatch between the model and the data.

Back to the demonstration: comparing different regularisers

The demonstrations thus far used the regulariser (3.2). This is equivalent to a prior that expects all the weights to have the same characteristic size. This is actually an inconsistent prior: the input and output variables and hidden unit activities could all be arbitrarily rescaled; if the same mapping is to be performed (a simple consistency requirement), such transformations of the variables would imply *independent* rescaling of the weights to the hidden layer and to the output layer. Thus, the scales of the two layers of weights are unrelated, and it is inconsistent to force the characteristic decay rates of these different classes of weights to be the same. This inconsistency is the major cause of the failure illustrated in figure 3.8. *All the networks deviating substantially from the desired trend have weights to the output layer far larger than the weights to the input layer;* this poor match to the model implicit in the regulariser causes the evidence for those solutions to be small.

This failure enables us to progress with insight to new regularisers. The alternative that I now present is a prior which is not inconsistent in the way explained above, so there are theoretical reasons to expect it to be 'better'. However, we will allow the data to choose, by evaluating the evidence for solutions using the new prior; we will find that the new prior is indeed *more probable*.

The second prior has three independent regularising constants, corresponding to the characteristic magnitudes of the weights in three different classes c, namely hidden unit weights, hidden unit biases, and output weights and biases (see figure 3.10). The term αE_W is replaced by $\sum_c \alpha_c E_W^c$, where $E_W^c = \sum_{i \in c} w_i^2/2$. Hinton and Nowlan [57] have used

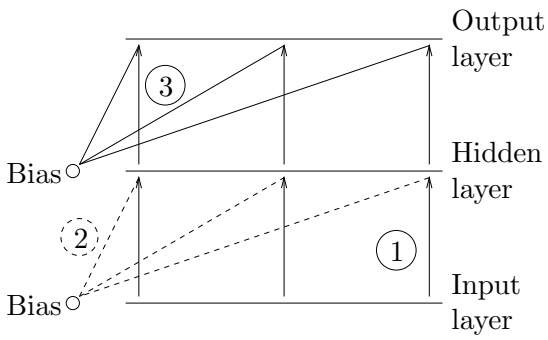

Figure 3.10: **The three classes of weights under the second prior**
1: Hidden unit weights. 2: Hidden unit biases. 3: Output unit weights and biases. The weights in one class c share the same decay constant α_c.

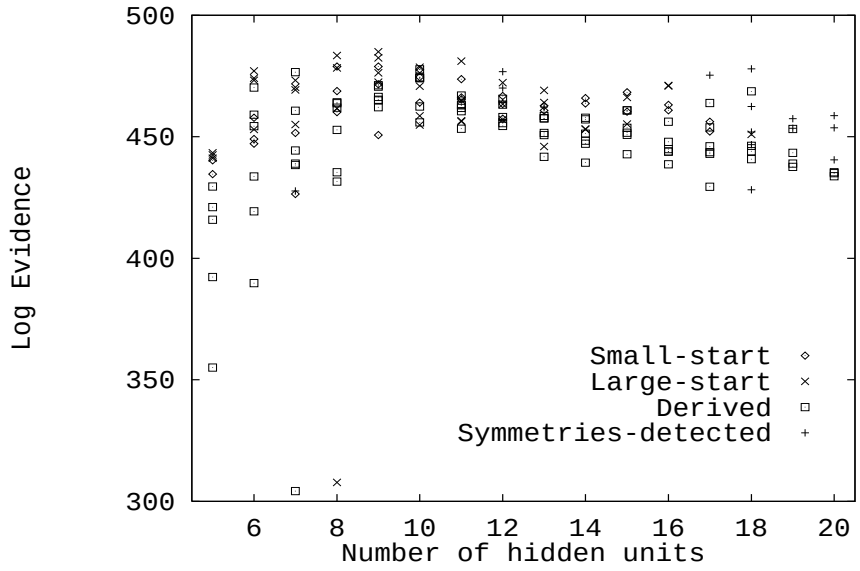

Figure 3.11: **Log evidence versus number of hidden units for the second prior**
The different point styles correspond to networks with learning initialised with small and large values of σ_W; networks previously trained using the first regulariser and subsequently trained on the second regulariser; and networks in which a weight symmetry was detected (in such cases the evidence evaluation is possibly less reliable).

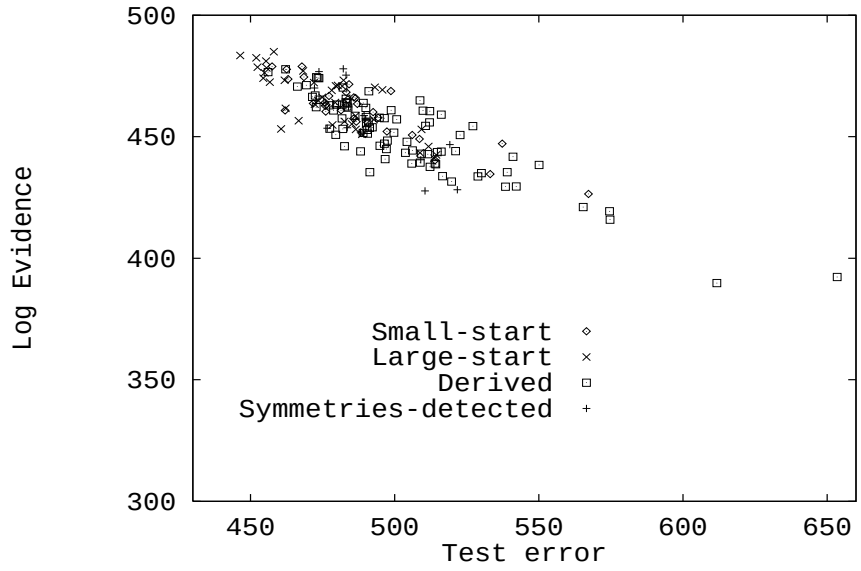

Figure 3.12: **Log evidence for the second prior versus test error.**
The correlation between the evidence and the test error for the second prior is very good. Note that the largest value of evidence has increased relative to figure 3.8, and the smallest test error has also decreased.

a similar prior modelling weights as coming from a Gaussian mixture, and using Bayesian re–estimation techniques to update the mixture parameters; they found such a model was good at discovering elegant solutions to problems with translation invariances. This model also achieves better performance on the task of sunspot time series prediction than any published model [58].

Using the second prior, each regularising constant is independently adapted to its most probable value by evaluating the number of well–measured parameters γ_c associated with each regularising function, and finding the optimum where $2\alpha_c E_W^c = \gamma_c$. The increased complexity of this prior model is penalised by an Occam factor for each new parameter α_c (see chapter 2). Let me preempt questions along the lines of 'why didn't you use four weight classes, or non–zero means?' — any other way of assigning weight decays is just another model, and you can try as many as you like; by evaluating the evidence you can then find out what preference the data have for the alternative decay schemes.

New solutions have been found using this second prior, and the evidence evaluated. The evidence for these new solutions with the new prior is shown in figure 3.11. Notice that the evidence has increased compared to the evidence for the first prior. For some solutions the new prior is more probable by a factor of 10^{30}.

Now the crunch: does this more probable model make good predictions? The evidence for the second prior is shown against the test error in figure 3.12. The correlation between the two is greatly improved. Notice furthermore that not only is the second prior more probable, the best test error achieved by solutions found using the second prior is slightly better than any achieved using the first prior, and the number of good solutions has increased substantially. Thus, the Bayesian evidence is a good predictor of generalisation ability, and the Bayesian choice of regularisers has enabled the best solutions to be found.

3.5 Discussion

The Bayesian method that has been presented is well–founded theoretically, and it works practically, though it remains to be seen how this approach will scale to larger problems. For a particular data set, the evaluation of the **evidence** has led us objectively from an inconsistent regulariser to a more probable one. The evidence is maximised by networks which generalise best, showing that Occam's razor has been successfully embodied with no ad hoc terms. Furthermore the solutions with greatest evidence perform better on a test set than any other solutions found. I believe there is currently no other technique that could reliably find and identify better solutions using only the training set. Essential to this success was the simultaneous Bayesian optimisation of the three regularising constants (decay terms) α_c. Optimisation of these parameters by any orthodox search technique such as cross–validation would be laborious; if there were many more than three regularising constants, as could easily be the case in larger problems, it is hard to imagine any such search being possible.[5]

This brings up the question of how these Bayesian calculations scale with problem size. In terms of the number of parameters k, calculation of the determinant and inverse of the Hessian scales as k^3. Note that this is a computation that needs to be carried out only a small number of times compared with the immense number of derivative calculations involved in a typical learning session. However, for large problems it may be too demanding to evaluate the determinant of the Hessian. If this is the case, numerical methods are available to approximate the determinant or trace of a matrix in k^2 time [72].

Application to classification problems

This chapter has thus far discussed the evaluation of the evidence for backprop networks trained on interpolation problems. Neural networks can also be trained to perform classification tasks. A future publication [52] will demonstrate that the Bayesian framework for model comparison can be applied to these problems too.

Relation to V–C dimension

Some papers advocate the use of V–C dimension [1] as a criterion for penalising over–complex models [2, 42]. V–C dimension is most often applied to classification problems; the evidence, on the other hand, can be evaluated equally easily for both interpolation and classification problems. V–C dimension is a worst case measure, so it yields different results from Bayesian analysis [32]. For example, V–C dimension is indifferent to the use of regularisers like (3.2), and to the value of α, because the use of such regularisers does not rule out absolutely any particular network parameters. Thus V–C dimension assigns the same complexity to a model whether or not it is regularised.[6] So it cannot be used to set regularising constants α or to compare alternative regularisers. In contrast, the preceding

[5] Radford Neal (personal communication) has pointed out that it is possible to evaluate the gradient of a validation error with respect to parameters such as $\{\alpha_c\}$, using $\partial E_{\rm val}/\partial\alpha_c = \partial E_{\rm val}/\partial \mathbf{w}_{\rm MP} \cdot \partial \mathbf{w}_{\rm MP}/\partial\alpha_c$. The first quantity could be evaluated by backprop, and the second term could be found within the quadratic approximation which gives $\partial \mathbf{w}_{\rm MP}/\partial\alpha_c = \mathbf{A}^{-1}\mathbf{I}_c\mathbf{w}_{\rm MP}$, where $\mathbf{I}_c$ is the identity matrix for the weights regularised by α_c, and zero elsewhere. Alternatively, Radford Neal has suggested that the gradients $\partial E_{\rm val}/\partial\alpha_c$ could be more efficiently calculated using 'recurrent backpropagation' [61], viewing $\mathbf{w}$ as the vector of activities of a recurrent network, and $\mathbf{w}_{\rm MP}$ as the fixed point whose error $E_{\rm val}$ we wish to minimise.

[6] However, E. Levin and I.Guyon *et al.*[30] have developed a measure of 'effective V–C dimension' of a regularised model. This measure is identical to γ, equation (3.12), and their predicted generalisation error based on Vapnik's structural risk theory has exactly the same scaling behaviour as the evidence!

demonstrations show that careful objective choice of regulariser and α is essential for the best solutions to be obtained.

Worst case analysis has a complementary role alongside Bayesian methods. Neither can substitute for the other.

Future tasks

Further work is needed to formalise the relationship of this framework to the pragmatic model comparison technique of cross–validation. Moody's work on 'generalised prediction error' (GPE) is an interesting contribution in this direction [54]. His sampling theory approach predicts that the generalisation error, in χ^2 units, will be $\frac{1}{N}(\chi^2_D + 2\gamma)$. However, I have evaluated the GPE for the interpolation models in this chapter's demonstration, and found the correlation between GPE and the actual test error was poor. More work is needed to understand this.

The Gaussian approximation used to evaluate the evidence breaks down when the number of data points is small compared to the number of parameters. For the model problems I have studied so far, the Gaussian approximation seemed to break down significantly for $N/k < 3 \pm 1$. It is a matter for further research to characterise this failure and investigate techniques for improving the evaluation of the integral Z^*_M, for example the use of random walks on M in the neighbourhood of a solution.

It is expected that evaluation of the evidence should provide an objective rule for deciding whether a network pruning or growing procedure should be stopped, but a careful study of this idea has yet to be performed.

It will be interesting to see the results of evaluating the evidence for networks applied to larger real–world problems.

Appendix: Numerical methods

Quick and dirty version

The three numerical tasks are automatic optimisation of α_c and β, calculation of error bars, and evaluation of the evidence. I will describe a cheap approximation for solving the first of these tasks without evaluating the Hessian. If we neglect the distinction between well–determined and poorly–determined parameters, we obtain the following update rules for α and β:

$$\begin{aligned} \alpha_c &:= k_c/2E^c_W \\ \beta &:= N/2E_D. \end{aligned}$$

If you want an easy–to–program taste of what a Bayesian framework can offer, try using this procedure to update your decay terms.

Hessian evaluation

The Hessian of M, $\mathbf{A}$, is needed to evaluate γ (which relates to Trace $\mathbf{A}^{-1}$), to evaluate the evidence (which relates to det $\mathbf{A}$), and to assign error bars to network outputs (using $\mathbf{A}^{-1}$).

I used two methods for evaluating $\mathbf{A}$: a) an approximate analytic method and b) second differences. The approximate analytic method was, following Denker *et al.*, to use backprop

to obtain the second derivatives, neglecting terms in f'', where f is the activation function of a neuron. The Hessian is built up as a sum of outer products of gradient vectors:

$$\nabla\nabla E_D \simeq \sum_{i,m} \mathbf{g}_i^m \mathbf{g}_i^{m\mathrm{T}}, \tag{3.17}$$

where $\mathbf{g}_i^m = \frac{dy_i(\mathbf{x}^m)}{d\mathbf{w}}$. Unlike Denker *et al.*, I did not ignore the off–diagonal terms; the diagonal approximation is not good enough! For the evaluation of γ the two methods gave similar results, and either approach seemed satisfactory. However, for the evaluation of the evidence, the approximate analytic method failed to give satisfactory results. The 'Occam factors' are very weak, scaling only as $\log N$, and the above approximation apparently introduces systematic errors greater than these. The reason that the evidence evaluation is more sensitive to errors than the γ evaluation is because γ is related to the sum of eigenvalues, whereas the evidence is related to the product; errors in small eigenvalues jeopardise the product more than the sum. I expect an exact analytic evaluation of the second derivatives [9] would resolve this. To save programming effort I instead used second differences, which is computationally more demanding ($\sim kN$ backprops) than the analytic approach ($\sim N$ backprops). There were still problems with errors in small eigenvalues, but it was possible to correct these errors, by detecting eigenvalues which were smaller than theoretically permitted.

Demonstrations

The demonstrations were performed as follows:
Initial weights: random weights drawn from a Gaussian with $\sigma_W = 0.3$.
Optimisation algorithm for $M(\mathbf{w})$: variable metric methods, using code from [64], used several times in sequence with values of the fractional tolerance decreasing from 10^{-4} to 10^{-8}. Every other loop, the regularising constants α_c were allowed to adapt in accordance with the re–estimation formula:

$$\alpha_c := \gamma_c / 2E_W^c. \tag{3.18}$$

Precaution

When evaluating the evidence, care must be taken to verify that the permutation term g is appropriately set. It may be the case (probably mainly in toy problems) that the regulariser makes two or more hidden units in a network adopt identical connection values; alternatively some hidden units might switch off, with all weights set to zero; in these cases the permutation term should be smaller. Also in these cases, it is likely that the quadratic approximation will perform badly (quartic rather than quadratic minima are likely), so it is preferable to automate the deletion of such redundant units.

Chapter 4

Information-based Objective Functions for Active Data Selection

Abstract

Learning can be made more efficient if we can actively select particularly salient data points. Within a Bayesian learning framework, objective functions are discussed which measure the *expected informativeness* of candidate measurements. Three alternative specifications of what we want to gain information about lead to three different criteria for data selection. All these criteria depend on the assumption that the hypothesis space is correct, which may prove to be their main weakness.

4.1 Introduction

Theories for data modelling often assume that the data is provided by a source that we do not control. However, there are two scenarios in which we are able to actively select training data. In the first, data measurements are relatively expensive or slow, and we want to know where to look next so as to learn as much as possible. According to Jaynes [36], Bayesian reasoning was first applied to this problem two centuries ago by Laplace, who in consequence made more important discoveries in celestial mechanics than anyone else. In the second scenario, there is an immense amount of data and we wish to select a subset of data points that are most useful for our purposes. Both these scenarios will benefit if we have ways of objectively estimating the utility of candidate data points.

The problem of 'active learning' or 'sequential design' has been extensively studied in economic theory and statistics [21, 23]. Experimental design within a Bayesian framework using the Shannon information as an objective function has been studied by Lindley [44] and by Luttrell [48]. A distinctive feature of this approach is that it renders the optimisation of the experimental design independent of the 'tests' that are to be applied to the data and the loss functions associated with any decisions. This chapter uses similar information–based objective functions and discusses the problem of optimal data selection within the Bayesian framework for interpolation described in chapters 2 and 3. Most of the results in this chapter have direct analogs in Fedorov [23], though the quantities involved have different

[0]Chapter 4 of Ph.D. thesis 'Bayesian Methods for Adaptive Models' by David MacKay, California Institute of Technology, submitted December 10 1991.

interpretations: for example, Fedorov's dispersion of an estimator becomes the Bayesian's posterior variance of the parameter. This work was directly stimulated by a presentation given by John Skilling at Maxent 91 [76].

Recent work in the neural networks literature on active data selection, also known as 'query learning', has concentrated on slightly different problems: The work of Baum [5] and Hwang *et al.* [35] relates to perfectly separable classification problems only; in both these papers a sensible query–based learning algorithm is proposed, and empirical results of the algorithm are reported; Baum also gives a convergence proof. But since the algorithms are both human–designed, it is not clear what objective function their querying strategy optimises, nor how the algorithms could be improved. In contrast, this chapter (which discusses noisy interpolation problems) *derives* criteria from *defined* objective functions; each objective function leads to a different data selection criterion. Chapter 5 will discuss the application of the same ideas to classification problems.

Plutowski and White [62] study a different problem from the above, in the context of noise–free interpolation: they assume that a large amount of data has already been gathered, and work on principles for selecting a subset of that data for efficient training; the entire data set (inputs *and* targets) is consulted at each iteration to decide which example to add to the training subset, an option that is not permitted here.

Statement of the problem

Imagine that we are gathering data in the form of a set of input–output pairs $D_N = \{\mathbf{x}^{(m)}, \mathbf{t}^{(m)}\}$, where $m = 1 \ldots N$. This data is modelled with an interpolant $\mathbf{y}(\mathbf{x}; \mathbf{w}, \mathcal{A})$. An interpolation model $\mathcal{H}$ specifies the 'architecture' $\mathcal{A}$, which defines the functional dependence of the interpolant on the parameters w_i, $i = 1 \ldots k$. The model also specifies a regulariser $\mathcal{R}$, or prior on $\mathbf{w}$, and a cost function, or noise model $\mathcal{N}$ describing the expected relationship between $\mathbf{y}$ and $\mathbf{t}$. We may have more than one interpolation model, each of which may be linear or non–linear in $\mathbf{w}$. Chapters 2 and 3 described the Bayesian framework for fitting and comparing such models, assuming a fixed data set. This chapter discusses how the same framework for interpolation relates to the task of selecting *what data to gather next.*

Our criterion for how informative a new datum is will depend on what we are interested in. Several alternatives spring to mind:

1. If we have decided to use one particular interpolation model, we might wish to select new data points to be maximally informative about the values that that model's parameters $\mathbf{w}$ should take.

2. Alternatively, we might not be interested in getting a globally well–determined interpolant; we might only want to be able to predict the value of the interpolant accurately in a limited region, perhaps at a point in input space which we are not able to sample directly.

3. Lastly, we might be unsure which of two or more models is the best interpolation model, and we might want to select data so as to give us maximal information to discriminate between the models.

This chapter will study each of these tasks for the case where we wish to evaluate the utility as a function of $\mathbf{x}^{N+1}$, the input location at which a single measurement of a scalar t^{N+1} will be made. The more complex task of selecting *multiple* new data points will not be addressed here, but the methods used can be generalised to solve this task, as is discussed in [23, 48].

The similar problem of choosing the $\mathbf{x}^{N+1}$ at which a *vector* of outputs $\mathbf{t}^{N+1}$ is measured will not be addressed either.

The first and third definitions of information gain have both been studied in the abstract by Lindley [44]. All three cases have been studied by Fedorov [23], mainly in non–Bayesian terms. In this chapter, solutions will be obtained for the interpolation problem by using a Gaussian approximation and in some cases assuming that the new datum is a relatively weak piece of information. In common with most other work on active learning, the utility is evaluated assuming that the probability distributions defined by the interpolation model are correct. For some models, this assumption may be the Achilles' heel of this approach, as discussed in section 4.6.

Can our choice bias our inferences?

One might speculate that the way we choose to gather data might be able to bias our inferences systematically away from the truth. If this were the case we might need to make our inferences in a way which undoes such biases by taking into account how we gathered the data. In orthodox statistics many estimators and statistical tests do depend on the sampling strategy.

However, the *likelihood principle* states that our inferences should depend on the likelihood of the actual data received, not on other data that we might have gathered but didn't. Bayesian inference is consistent with this principle; there is no need to undo biases introduced by the data collecting strategy, because it is *not possible* for such biases to be introduced — as long as we perform inference using all the data gathered [8, 47]. When the models are concerned with estimating the distribution of output variables $\mathbf{t}$ given input variables $\mathbf{x}$, we are allowed to look at the $\mathbf{x}$ value of a datum, and decide whether or not to include the datum in the data set. This will not bias our inferences about the distribution $P(\mathbf{t}|\mathbf{x})$.

4.2 Choice of information measure

Before we can start, we need to select a measure of the information gained about an unknown variable when we receive the new datum $\mathbf{t}^{N+1}$. Having chosen such a measure we will then select the $\mathbf{x}^{N+1}$ for which the *expected* information gain is maximal. Two measures of information have been suggested, both based on Shannon's entropy, whose properties as a sensible information measure are well known. Let us explore this choice for the first task, where we want to gain maximal information about the parameters of the interpolant, $\mathbf{w}$.

Let the probability distributions of the parameters before and after we receive the datum $\mathbf{t}^{N+1}$ be $P^N(\mathbf{w})$ and $P^{N+1}(\mathbf{w})$. Then the *change in entropy* of the distribution is $\Delta S = S_N - S_{N+1}$, where:

$$S_N = \int d^k\mathbf{w}\, P^N(\mathbf{w}) \log \frac{m(\mathbf{w})}{P^N(\mathbf{w})}, \tag{4.1}$$

where m is the measure on $\mathbf{w}$ that makes the argument of the log dimensionless.[1] The greater ΔS is, the more information we have gained about $\mathbf{w}$. In the case of the quadratic

[1] This measure m will be unimportant in what follows but is included to avoid committing dimensional crimes. Note that the sign of ΔS has been defined so that our information gain corresponds to positive ΔS.

models discussed in chapter 2, if we set the measure $m(\mathbf{w})$ equal to the prior $P^0(\mathbf{w})$, the quantity S_N is closely related to the log of the 'Occam factor'.[2]

An alternative information measure is the *cross entropy* between $P^N(\mathbf{w})$ and $P^{N+1}(\mathbf{w})$:

$$G = \int d^k\mathbf{w}\, P^{N+1}(\mathbf{w}) \log \frac{P^N(\mathbf{w})}{P^{N+1}(\mathbf{w})}. \tag{4.2}$$

Let us define $G' = -G$ so as to obtain a positive quantity; then G' is a measure of how much information we gain when we are informed that the true distribution of $\mathbf{w}$ is $P^{N+1}(\mathbf{w})$, rather than $P^N(\mathbf{w})$.

These two information measures are not equal. Intuitively they differ in that if the measure $m(\mathbf{w})$ is flat, ΔS only quantifies how much the probability 'bubble' of $P(\mathbf{w})$ shrinks when the new datum arrives; G' also incorporates a measure of how much the bubble *moves* because of the new datum. Thus according to G', even if the probability distribution does not shrink and become more certain, we have *learnt* something if the distribution moves from one region to another in $\mathbf{w}$–space.

The question of which information measure is appropriate is potentially complicated by the fact that G' is not a consistent additive measure of information: if we receive datum A then datum B, in general, $G'_{AB} \neq G'_A + G'_B$. This intriguing complication will not however hinder our task: we can only base our decisions on the *expectations* of ΔS and G'; we will now see that in expectation ΔS and G' are equal, so for our purposes there is no distinction between them. This result holds independent of the details of the models we study and independent of any Gaussian approximation for $P(\mathbf{w})$.

Proof that $E(\Delta S) = E(G')$

To evaluate the expectation of these quantities, we have to assume a probability distribution from which the datum $\mathbf{t}^{N+1}$ (hence abbreviated as $\mathbf{t}$) comes. We will define this probability distribution by assuming that our current model, complete with its error bars, is correct. This means that the probability distribution of $\mathbf{t}$ is $P(\mathbf{t}|D_N, \mathcal{H})$, where $\mathcal{H}$ is the total specification of our model. The conditioning variables on the right will be omitted in the following proof.

We can now compare the expectations of ΔS and G'.

$$\begin{aligned} G' &= -\int d^k\mathbf{w}\, P(\mathbf{w}|\mathbf{t}) \log \frac{P(\mathbf{w})}{P(\mathbf{w}|\mathbf{t})} \\ &= -\int d^k\mathbf{w}\, P(\mathbf{w}|\mathbf{t}) \log \frac{m(\mathbf{w})}{P(\mathbf{w}|\mathbf{t})} + \int d^k\mathbf{w}\, P(\mathbf{w}|\mathbf{t}) \log \frac{m(\mathbf{w})}{P(\mathbf{w})}, \end{aligned} \tag{4.3}$$

where m is free to be any measure on $\mathbf{w}$; let us make it the same measure m as in (4.1). Then the first term in (4.3) is $-S_{N+1}$. So

$$\begin{aligned} E(G') &= -E(S_{N+1}) + \int d\mathbf{t}\, P(\mathbf{t}) \int d^k\mathbf{w}\, P(\mathbf{w}|\mathbf{t}) \log \frac{m(\mathbf{w})}{P(\mathbf{w})} \\ &= -E(S_{N+1}) + \int d^k\mathbf{w}\, P(\mathbf{w}) \log \frac{m(\mathbf{w})}{P(\mathbf{w})} \\ &= E(-S_{N+1} + S_N) = E(\Delta S). \end{aligned}$$

•

[2] If the Occam factor is $O.F. = (2\pi)^{k/2} \det{}^{-\frac{1}{2}}\mathbf{A} \exp(-\alpha E_W^{\rm MP})/Z_W(\alpha)$, then $S_N = \log O.F. + \gamma/2$, using notation from chapter 2.

Thus the two candidate information measures are equivalent for our purposes. This proof also implicitly demonstrates that $E(\Delta S)$ is independent of the measure $m(\mathbf{w})$. Other properties of $E(\Delta S)$ are proved in [44]. The rest of this chapter will use ΔS as the information measure, with $m(\mathbf{w})$ set to a constant.

4.3 Maximising total information gain

Let us now solve the first task: how to choose $\mathbf{x}^{N+1}$ so that the expected information gain about $\mathbf{w}$ is maximised. Intuitively we expect that we will learn most about the interpolant by gathering data at the $\mathbf{x}$ location where our error bars on the interpolant are currently greatest. Within the quadratic approximation, we will now confirm that intuition.

Notation

The likelihood of the data is defined in terms of a noise level $\sigma_\nu^2 = \beta^{-1}$ by $P(\{\mathbf{t}\}|\mathbf{w}, \beta, \mathcal{N}) = \exp(-\beta E_D(\mathbf{w}))/Z_D$, where $E_D(\mathbf{w}) = \sum_m \frac{1}{2}(\mathbf{t}^m - \mathbf{y}(\mathbf{x}^{(m)}; \mathbf{w}))^2$, and Z_D is the appropriate normalising constant. The likelihood could also be defined with an $\mathbf{x}$–dependent noise level $\beta^{-1}(\mathbf{x})$, or correlated noise in multiple outputs (in which case β^{-1} would be the covariance matrix of the noise). From here on $\mathbf{y}$ will be treated as a scalar y for simplicity. When the likelihood for the first N data is combined with a prior $P(\mathbf{w}|\alpha, \mathcal{R}) = \exp(-\alpha E_W(\mathbf{w}))/Z_W$, in which the regularising constant (or weight decay rate) α corresponds to the prior expected smoothness of the interpolant, we obtain our current probability distribution for $\mathbf{w}$, $P^N(\mathbf{w}) = \exp(-M(\mathbf{w}))/Z_M$, where $M(\mathbf{w}) = \alpha E_W + \beta E_D$. The objective function $M(\mathbf{w})$ can be quadratically approximated near to the most probable parameter vector, $\mathbf{w}_{\mathrm{MP}}$, by

$$M(\mathbf{w}) \simeq M^*(\mathbf{w}) = M(\mathbf{w}_{\mathrm{MP}}) + \frac{1}{2}\Delta\mathbf{w}^{\mathrm{T}}\mathbf{A}\Delta\mathbf{w}, \tag{4.4}$$

where $\Delta\mathbf{w} = \mathbf{w} - \mathbf{w}_{\mathrm{MP}}$ and the Hessian $\mathbf{A} = \nabla\nabla M$ is evaluated at the minimum $\mathbf{w}_{\mathrm{MP}}$. We will use this quadratic approximation from here on. If M has other minima, those can be treated as distinct models as in chapter 3.

First we will need to know what the entropy of a Gaussian distribution is. It is easy to confirm that if $P(\mathbf{w}) \propto e^{-M^*(\mathbf{w})}$, then for a flat measure $m(\mathbf{w}) = m$,

$$S = \frac{k}{2}(1 + \log 2\pi) + \frac{1}{2}\log\left(m^2 \det \mathbf{A}^{-1}\right). \tag{4.5}$$

Thus our aim in minimising S is to make the size of the joint error bars on the parameters, $\det \mathbf{A}^{-1}$, as small as possible.

Expanding $\mathbf{y}$ around $\mathbf{w}_{\mathrm{MP}}$, let

$$\mathbf{y}(\mathbf{x}) \simeq \mathbf{y}(\mathbf{x}; \mathbf{w}_{\mathrm{MP}}) + \mathbf{g}(\mathbf{x}) \cdot \Delta\mathbf{w}, \tag{4.6}$$

where $g_j = \frac{\partial y}{\partial w_j}$ is the ($\mathbf{x}$–dependent) sensitivity of the output variable to parameter w_j, evaluated at $\mathbf{w}_{\mathrm{MP}}$.

Now imagine that we choose a particular input $\mathbf{x}$ and collect a new datum. If the datum $\mathbf{t}$ falls in the region such that our quadratic approximation applies, the new Hessian $\mathbf{A}_{N+1}$ is:

$$\mathbf{A}_{N+1} \simeq \mathbf{A} + \beta\mathbf{g}\mathbf{g}^{\mathrm{T}}, \tag{4.7}$$

where we have used the approximation $\nabla\nabla\frac{1}{2}(\mathbf{t} - \mathbf{y}(\mathbf{x}; \mathbf{w}))^2 \simeq \mathbf{g}\mathbf{g}^{\mathrm{T}}$. This expression neglects terms in $\frac{\partial^2 y}{\partial w_j \partial w_k}$; those terms are exactly zero for the linear models discussed in chapter

2, but they are not necessarily negligible for non–linear models such as neural networks. Notice that this new Hessian is independent of the value that the datum $\mathbf{t}$ actually takes, so we can specify what the information gain ΔS will be for any datum, because we can evaluate $\mathbf{A}_{N+1}$ just by calculating $\mathbf{g}$.

Let us now see what property of a datum causes it to be maximally informative. The new entropy S_{N+1} is equal to $-\frac{1}{2}\log(m^2 \det \mathbf{A}_{N+1})$, neglecting additive constants. This determinant can be analytically evaluated [23], using the identities

$$[\mathbf{A}+\beta\mathbf{g}\mathbf{g}^{\mathrm{T}}]^{-1} = \mathbf{A}^{-1} - \frac{\beta\mathbf{A}^{-1}\mathbf{g}\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}}{1+\beta\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}} \quad \text{and} \quad \det[\mathbf{A}+\beta\mathbf{g}\mathbf{g}^{\mathrm{T}}] = (\det\mathbf{A})(1+\beta\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}), \quad (4.8)$$

from which we obtain:

$$\text{Total information gain} = \frac{1}{2}\Delta\log\left(m^2\det\mathbf{A}\right) = \frac{1}{2}\log(1+\beta\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}). \quad (4.9)$$

In the product $\beta\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}$, the first term β tells us that, not surprisingly, we learn more information if we make a low noise (high β) measurement. The second term $\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}$ is precisely the variance of the interpolant at the point where the datum is collected.

Thus we have our first result: to obtain maximal information about the interpolant, take the next datum at the point where the error bars on the interpolant are currently largest (assuming the noise σ_ν^2 on all measurements is the same). This rule is the same as that resulting from the 'D–optimal' and 'minimax' design criteria [23].

For many interpolation models, the error bars are largest beyond the most extreme points where data have been gathered. This first criterion would in those cases lead us to repeatedly gather data at the edges of the input space, which might be considered non–ideal behaviour; but we do not necessarily need to introduce an ad hoc procedure to avoid this. The reason we do not want repeated sampling at the edges is that we do not want to *know* what happens there. Accordingly, we can derive criteria from alternative objective functions which only value information acquired about the interpolant in a defined region of interest.

4.4 Maximising information about the interpolant in a region of interest

Thus we come to the second task. First assume we wish to gain maximal information about the value of the interpolant at a particular point $\mathbf{x}^{(u)}$. Under the quadratic approximation, our uncertainty about the interpolant $\mathbf{y}$ has a Gaussian distribution, and the size of the error bars is given in terms of the Hessian of the parameters by

$$\sigma_u^2 = \mathbf{g}_{(u)}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}_{(u)},$$

where $\mathbf{g}_{(u)}$ is $\partial y/\partial\mathbf{w}$ evaluated at $\mathbf{x}^{(u)}$. As above, the entropy of this Gaussian distribution is $\frac{1}{2}\log\sigma_u^2 + \text{const}$. After a measurement t is made at $\mathbf{x}$ where the sensitivity is $\mathbf{g}$, these error bars are scaled down by a factor of $1-\rho^2$, where ρ is the correlation between the variables t and $y^{(u)}$, given by $\rho^2 = (\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}_{(u)})^2/(\sigma_u^2(\sigma_\nu^2+\sigma_{\mathbf{x}}^2))$, where $\sigma_{\mathbf{x}}^2 = \mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}$. Thus the information gain about $y^{(u)}$ is:

$$\text{Marginal information gain} = \frac{1}{2}\Delta\log\sigma_u^2 = -\frac{1}{2}\log\left(1-\frac{(\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}_{(u)})^2}{\sigma_u^2(\sigma_\nu^2+\sigma_{\mathbf{x}}^2)}\right). \quad (4.10)$$

The term $\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}_{(u)}$ is maximised when the sensitivities $\mathbf{g}$ and $\mathbf{g}_{(u)}$ are maximally correlated, as measured by their inner product in the metric defined by $\mathbf{A}^{-1}$. The second task is thus solved for the case of extrapolation to a single point. This objective function is demonstrated and criticised in section 4.6.

Generalisation to multiple points

Now imagine that the objective function is defined to be the information gained about the interpolant at a set of points $\{\mathbf{x}^{(u)}\}$. These points should be thought of as representatives of the region of interest, for example, points in a test set. This case also includes the generalisation to more than one output variable y; however the full generalisation, to optimisation of an experiment in which many measurements are made, will not be made here (see Fedorov [23] and Luttrell [48]). The preceding objective function, the information about $y^{(u)}$, can be generalised in several ways, some of which lead to dissatisfactory results.

First objective function for multiple points

An obvious objective function is the *joint entropy* of the output variables that we are interested in. Let the set of output variables for which we want to minimise the uncertainty be $\{y^{(u)}\}$, where $u=1\ldots V$ runs either over a sequence of different input locations $\mathbf{x}^{(u)}$, or over a set of different scalar outputs, or both. Let the sensitivities of these outputs to the parameters be $\mathbf{g}_{(u)}$. Then the covariance matrix of the values $\{y^{(u)}\}$ is

$$\mathbf{Y} = \mathbf{G}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G}, \tag{4.11}$$

where the matrix $\mathbf{G} = \left[\mathbf{g}_{(1)}\mathbf{g}_{(2)}\ldots\mathbf{g}_{(V)}\right]$. Disregarding the possibility that $\mathbf{Y}$ might not have full rank, which would necessitate a more complex treatment giving similar results, the joint entropy of our output variables $S(P(\{y^{(u)}\}))$ is related to $\log\det\mathbf{Y}^{-1}$. We can find the information gain for a measurement with sensitivity vector $\mathbf{g}$, under which $\mathbf{A}\to\mathbf{A}+\beta\mathbf{g}\mathbf{g}^{\mathrm{T}}$, using the identities (4.8).

$$\text{Joint information gain} = \frac{1}{2}\Delta\log\det\mathbf{Y}^{-1} = -\frac{1}{2}\log\left[1-\frac{(\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G})\mathbf{Y}^{-1}(\mathbf{G}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g})}{\sigma_\nu^2+\sigma_{\mathbf{x}}^2}\right]. \tag{4.12}$$

The row vector $\mathbf{v}=\mathbf{g}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{G}$ measures the correlations between the sensitivities $\mathbf{g}$ and $\mathbf{g}_{(u)}$. The quadratic form $\mathbf{v}\mathbf{Y}^{-1}\mathbf{v}^{\mathrm{T}}$ measures how effectively these correlations work together to reduce the joint uncertainty in $\{y^{(u)}\}$. The denominator $\sigma_\nu^2+\sigma_{\mathbf{x}}^2$ moderates this term in favour of measurements with small uncertainty.

Criticism

I will now argue that actually the joint entropy $S(P(\{y^{(u)}\}))$ of the interpolant's values is *not* an appropriate objective function. A simple example will illustrate this.

Imagine that $V=k$, *i.e.*, the number of points defining our region of interest is the same as the dimensionality of the parameter space $\mathbf{w}$. The resulting matrix $\mathbf{G}=\left[\mathbf{g}_{(1)}\mathbf{g}_{(2)}\ldots\mathbf{g}_{(V)}\right]$ may be almost singular if the points $\mathbf{x}^{(u)}$ are close together, but typically it will still have full rank. Then the parameter vector $\mathbf{w}$ and the values of the interpolant $\{y^{(u)}\}$ are in one to one (locally) linear correspondence with each other. This means that the change in entropy of $P(\{y^{(u)}\})$ is *identical* to the change in entropy of $P(\mathbf{w})$ [44]. This can be confirmed by substitution of $\mathbf{Y}^{-1}=\mathbf{G}^{-1}\mathbf{A}\mathbf{G}^{-1^{\mathrm{T}}}$ into (4.12), which yields (4.9). So if the

datum is chosen in accordance with equation (4.12), so as to maximise the expected joint information gain about $\{y^{(u)}\}$, exactly the same choice will result as is obtained maximising the first criterion, the expected total information gain about $\mathbf{w}$ (section 4.3)! Clearly, this choice is independent of our choice of $\{y^{(u)}\}$, so it will have nothing to do with our region of interest.

This criticism of the joint entropy is not restricted to the case $V = k$. The reason that this objective function does not achieve what we want is that the joint entropy is decreased by measurements which introduce *correlations* among predictions about $\{y^{(u)}\}$ as well as by measurements which reduce the individual uncertainties of predictions. However, we don't want the variables $\{y^{(u)}\}$ to be strongly correlated in some *arbitrary* way; rather we want each $y^{(u)}$ to have small variance, so that if we are subsequently asked to predict the value of y at any one of the u's, we will be able to make confident predictions.

Second objective function for multiple points

This motivates an alternative objective function: to maximise the average over u of the information gained about $y^{(u)}$ alone. Let us define the mean marginal entropy,

$$S^{\mathrm{M}} = \sum_u P_u \, S(P(y^{(u)})) = \frac{1}{2} \sum_u P_u \, \log \sigma_u^2 + \text{const},$$

where P_u is the probability that we will be asked to predict $y^{(u)}$, and $\sigma_u^2 = \mathbf{g}_{(u)}^{\mathrm{T}} \mathbf{A}^{-1} \mathbf{g}_{(u)}$. For a measurement with sensitivity vector $\mathbf{g}$, we obtain from (4.10):

$$\text{Mean marginal information gain} = -\frac{1}{2} \sum_u P_u \, \log \left(1 - \frac{(\mathbf{g}^{\mathrm{T}} \mathbf{A}^{-1} \mathbf{g}_{(u)})^2}{\sigma_u^2 (\sigma_\nu^2 + \sigma_{\mathbf{x}}^2)} \right). \tag{4.13}$$

The mean marginal information gain is demonstrated and criticised in section 4.6.

Two simple variations on this objective function can be derived. If instead of minimising the mean marginal entropy of our predictions $y^{(u)}$, we minimise the mean marginal entropy of the predicted noisy variables $t^{(u)}$, which are modelled as deviating from $y^{(u)}$ under additive noise of variance σ_ν^2, we obtain (4.13) with σ_u^2 replaced by $\sigma_u^2 + \sigma_\nu^2$. This alternative may lead to significantly different choices from (4.13) when any of the marginal variances σ_u^2 fall below the intrinsic variance σ_ν^2 of the predicted variable.

If instead we take an approach based on loss functions, and require that the datum we choose minimises the expectation of the mean squared error of our predictions $\{y^{(u)}\}$, which is $E^{\mathrm{M}} = \sum_u P_u \sigma_u^2$, then we obtain as our objective function, to leading order, $\Delta E^{\mathrm{M}} \simeq \sum_u P_u (\mathbf{g}^{\mathrm{T}} \mathbf{A}^{-1} \mathbf{g}_{(u)})^2 / (\sigma_\nu^2 + \sigma_{\mathbf{x}}^2)$; this increases the bias in favour of reducing the variance of the variables $y^{(u)}$ with largest σ_u^2. This is the same as the 'Q–optimal' design [23].

Comment on the case of linear models

It is interesting to note that for a linear model (one for which $\mathbf{y}(\mathbf{x}; \mathbf{w}) = \sum w_h \phi_h(\mathbf{x})$) with quadratic penalty functions, the solutions to the first and second tasks depend only on the $\mathbf{x}$ locations where data were previously gathered, not on the actual data gathered $\{\mathbf{t}\}$; this is because $\mathbf{g}(\mathbf{x}) = \phi(\mathbf{x})$ independent of $\mathbf{w}$, so $\mathbf{A} = \alpha \nabla\nabla E_W + \beta \sum_m \mathbf{g}\mathbf{g}^{\mathrm{T}}$ is independent of $\{\mathbf{t}\}$. A complete data–gathering plan can be drawn up before we start. It is only for a non–linear model that our decisions about what data to gather next are affected by our previous observations!

4.5 Maximising the discrimination between two models

Under the quadratic approximation, two models will make slightly different Gaussian predictions about the value of any datum. If we measure a datum t at input value $\mathbf{x}$, then

$$P(t|\mathcal{H}_i) = \text{Normal}(\mu_i, \sigma_i^2),$$

where the parameters μ_i, σ_i^2 are obtained for each interpolation model $\mathcal{H}_i$ from its own best fit parameters $\mathbf{w}_{\text{MP}}(i)$, its own Hessian $\mathbf{A}$, and its own sensitivity vector $\mathbf{g}_i$:

$$\begin{aligned} \mu_i &= \mathbf{y}(\mathbf{x}; \mathbf{w}_{\text{MP}}(i)) \\ \sigma_i^2 &= \mathbf{g}_i^{\text{T}} \mathbf{A}_i^{-1} \mathbf{g}_i + 1/\beta. \end{aligned}$$

Intuitively, we expect that the most informative measurement will be at a value of $\mathbf{x}$ such that μ_1 and μ_2 are as separated as possible from each other on a scale defined by σ_1, σ_2. Further thought will also confirm that we expect to gain more information if σ_1^2 and σ_2^2 differ from each other significantly; at such points, the 'Occam factor' penalising the more powerful model becomes more significant.

Let us define the information gain to be $\Delta S = S_N - S_{N+1}$, where $S = -\sum_i P(\mathcal{H}_i) \log P(\mathcal{H}_i)$. Exact calculations of ΔS are not analytically possible, so I will assume that we are in the regime of small information gain, *i.e.*, we expect measurement of t to give us a rather weak likelihood ratio $P(t|\mathcal{H}_1)/P(t|\mathcal{H}_2)$. This is the regime where $|\mu_1 - \mu_2| \ll \sigma_1, \sigma_2$.

Using this assumption we can take the expectation over t, and a page of algebra leads to the result:

$$E(\Delta S) \simeq \frac{P(\mathcal{H}_1)P(\mathcal{H}_2)}{2} \left[\left(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2} \right) (\mu_1 - \mu_2)^2 + \left(\frac{\sigma_1^2 - \sigma_2^2}{\sigma_1 \sigma_2} \right)^2 \right]. \tag{4.14}$$

These two terms correspond precisely to the two expectations stated above. The first term favours measurements where μ_1 and μ_2 are well separated; the second term favours places where σ_1^2 and σ_2^2 differ. Thus the third task has been solved.

Fedorov [23] makes a similar derivation but he uses a poor approximation which loses the second term.

4.6 Demonstration and Discussion

A data set consisting of 21 points from a one–dimensional interpolation problem was interpolated with an eight hidden unit neural network. The data were generated from a smooth function by adding noise with standard deviation $\sigma_\nu = 0.05$. The neural network was adapted to the data using weight decay terms α_c which were controlled using the methods of chapter 3 and noise level β fixed to $1/\sigma_\nu^2$. The data and the resulting interpolant, with error bars, are shown in figure 4.1a.

The expected total information gain, *i.e.*, the change in entropy of the parameters, is shown as a function of x in figure 4.1b. This is just a monotonic function of the size of the error bars. The same figure also shows the expected marginal information gain about three points of interest, $\{x^{(u)}\} = \{-1.25, 0.0, 1.75\}$. Notice that the marginal information gain is in each case peaked near the point of interest, as we would expect. Note also that the height of this peak is greatest for $x^{(u)} = -1.25$, where the interpolant oscillates rapidly, and lower for $x^{(u)} = 1.75$, where the interpolant is smoother. At each $x = x^{(u)}$, the marginal information gain about $x^{(u)}$ and the total information gain are equal.

Figure 4.1c shows the mean marginal information gain, where the points of interest, $\{x^{(u)}\}$, were defined to be a set of equally spaced points on the interval $[-2.1, 4.1]$ (the same interval in which the training data lie). The mean marginal information gain gradually decreases to zero away from the region of interest, as hoped. In the region to the left where the characteristic period of the interpolant is similar to the data spacing, the expected utility oscillates as x passes through the existing data points, which also seems reasonable. The only surprising feature is that the estimated utility in that region is lower on the data points than the estimated utility in the smooth region towards the right.

The Achilles' heel of these methods

This approach has a potential weakness: there may be models for which, even though we have defined the region of interest by the points $\{x^{(u)}\}$, the expected marginal information gain for a measurement at x still blows up as $x \to \pm\infty$, like the error bars. This can occur because the information gain estimates the utility of a data point *assuming* that the model is correct; if we know that the model is actually an approximation tool that is incorrect, then it is possible that undesirable behaviour will result.

A simple example that illustrates this problem is obtained if we consider modelling data with a straight line $y = w_1 x$, where w_1 is the unknown parameter. Imagine that we want to select data so as to obtain a model that predicts accurately at $x^{(u)}$. Then if we assume that the model is right, clearly we gain most information if we sample at the largest possible $|x|$, since such points give the largest signal to noise ratio for determining w_1. If however we assume that the model is actually not correct, but only an approximation tool, then common sense tells us we should sample closer to $x^{(u)}$.

Thus if we are using models that we know are incorrect, the marginal information gain is really the right answer to the wrong question. It is a task for further research to formulate a new question whose answer is appropriate for any approximation model. Meanwhile, the mean marginal information gain seems a promising objective function to test further.

Computational complexity

The computation of the suggested objective functions is moderately cheap once the inverse Hessian $\mathbf{A}^{-1}$ has been obtained for the models concerned. This is a $O(Nk^2)+O(k^3)$ process, where N is the number of data points and k is the number of parameters; this process may already have been performed in order to evaluate error bars for the models, to evaluate the 'evidence', to evaluate parameter 'saliencies', and to enable efficient learning. This cost can be compared with the cost of locating a minimum of the objective function M, which in the worst case scales as $O(Nk^3)$ (taking the result for a quadratic function). Evaluation of the mean marginal information gain at C candidate points $\mathbf{x}$ then requires $O(Ck^2)+O(CVk)$ time, where V is the number of points of interest $\mathbf{x}^{(u)}$ ($O(k^2)$ to evaluate $\mathbf{A}^{-1}\mathbf{g}$ for each $\mathbf{x}$, and $O(Vk)$ to evaluate the dot product of this vector with each $\mathbf{g}_{(u)}$). So if $C = O(k)$ and $V = O(k)$, evaluation of the mean marginal information gain will be less computationally expensive than the inverse Hessian evaluation.

For contexts in which this is too expensive, work in progress is exploring the possibility of reducing these calculations to $O(k^2)$ or smaller time by statistical methods.

The question of how to efficiently search for the most informative $\mathbf{x}$ is not addressed here; gradient–based methods could be constructed, but figure 4.1c shows that the information gain may be locally non–convex, on a scale defined by the inter–datum spacing.

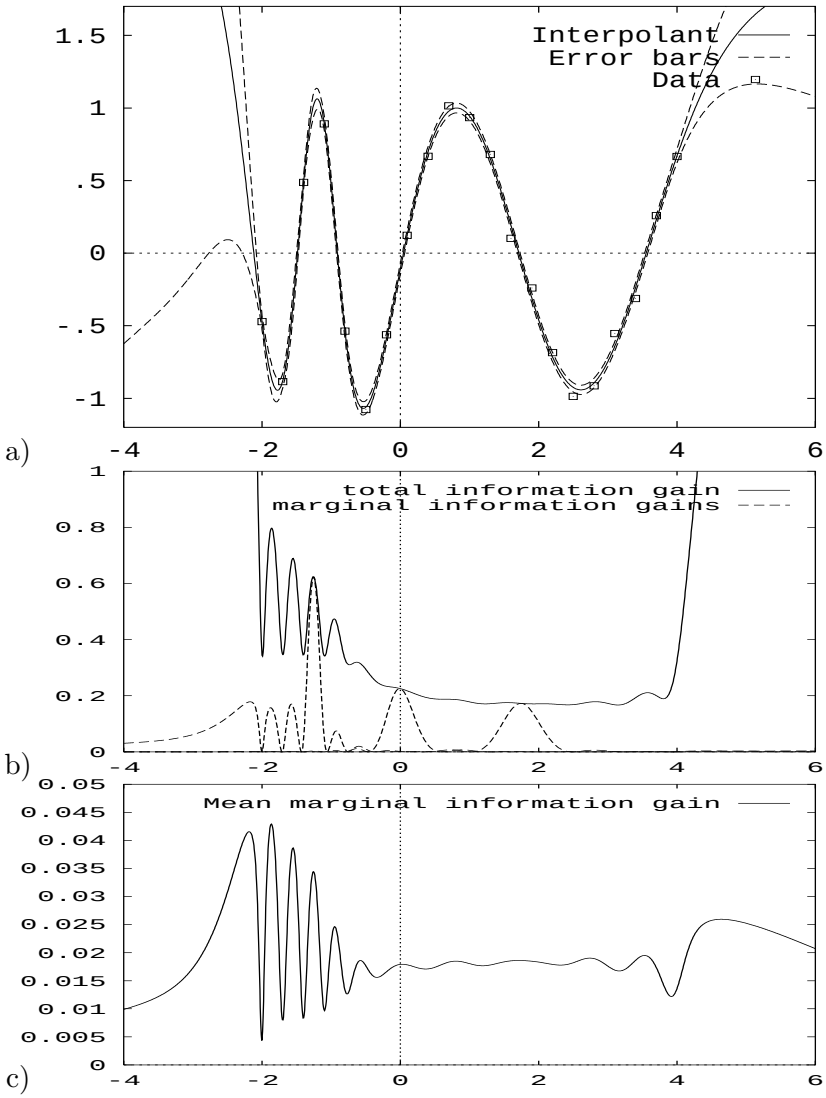

Figure 4.1: **Demonstration of total and marginal information gain**
a) The data set, the interpolant, and error bars. b) The expected total information gain and three marginal information gains. c) The mean marginal information gain, with the region of interest defined by 300 equally spaced points on the interval $[-2.1, 4.1]$. The information gains are shown on a scale of nats (1 nat $= \log_2 e$ bits).

4.7 Conclusion

For three specifications of the information to be maximised, a solution has been obtained. The solutions apply to linear and non–linear interpolation models, but depend on the validity of a local Gaussian approximation. Each solution has a direct analog in the non–Bayesian literature [23], and generalisations to multiple measurements and multiple output variables can be found there, and also in [48].

In each case a function of $\mathbf{x}$ has been derived that predicts the information gain for a measurement at that $\mathbf{x}$. This function can be used to search for an optimal value of $\mathbf{x}$ (which in large–dimensional input spaces may not be a trivial task). This function could also serve as a way of reducing the size of a large data set by omitting the data points that are expected to be least informative. And this function could form the basis of a stopping rule, *i.e.*, a rule for deciding whether to gather more data, given a desired exchange rate of information gain per measurement [44].

A possible weakness of these information–based approaches is that they estimate the utility of a measurement assuming that the model is correct. This might lead to undesirable results. The search for ideal measures of data utility is still open.

Chapter 5

The Evidence Framework applied to Classification Networks

Abstract

Three Bayesian ideas are presented for supervised adaptive classifiers. First, it is argued that the output of a classifier should be obtained by marginalising over the posterior distribution of its parameters; a simple approximation to this integral is proposed and demonstrated. This involves a 'moderation' of the most probable classifier's outputs, and yields improved performance. Second, it is demonstrated that the Bayesian framework for model comparison described for regression models in chapters 2 and 3 can also be applied to classification problems. This framework successfully chooses the magnitude of weight decay terms, and ranks solutions found using different numbers of hidden units. Third, an information–based data selection criterion is derived and demonstrated within this framework.

5.1 Introduction

A quantitative Bayesian framework has been described for learning of mappings in feed-forward networks in chapters 2 and 3. It was demonstrated that this 'evidence' framework could successfully choose the magnitude and type of weight decay terms, and could choose between solutions using different numbers of hidden units. The framework also gives quantified error bars expressing the uncertainty in the network's outputs and its parameters. In chapter 4 information–based objective functions for active learning were discussed within the same framework.

These three chapters concentrated on interpolation (regression) problems. Neural networks can also be trained to perform classification tasks.[1] This chapter will show that the Bayesian framework for model comparison can be applied to these problems too.

Assume that a set of candidate classification models are fitted to a data set, using standard methods. Three aspects of the use of classifiers can then be distinguished:

1. The individual classification models are used to make predictions about new targets.

[0] Chapter 5 of Ph.D. thesis 'Bayesian Methods for Adaptive Models' by David MacKay, California Institute of Technology, submitted December 10 1991.

[1] In regression the target variables are real numbers, assumed to include additive errors; in classification the target variables are discrete class labels.

2. The alternative models are ranked in the light of the data.

3. The expected utility of alternative new data points is estimated for the purpose of 'query learning' or 'active data selection'.

This chapter will present Bayesian ideas for these three tasks. Other aspects of classifiers use such as prediction of generalisation ability are not addressed.

First let us review the framework for supervised adaptive classification.

Derivation of the objective function $G = \sum t \ln p$

The same notation and conventions will be used as in chapters 2 and 3. Let the data set be $D = \{\mathbf{x}^{(m)}, t_m\}$, $m = 1 \ldots N$. In a classification problem, each target t_m is a binary (0/1) variable (more than two classes can also be handled [15]), and the activity of the output of a classifier is viewed as an estimate of the probability that $t = 1$. It is assumed that the classification problem is noisy, that is, repeated sampling at the same $\mathbf{x}$ would produce different values of t with certain probabilities; those probabilities, as a function of $\mathbf{x}$, are the quantities that a discriminative classifier is intended to model. It is well known that the natural objective function in this case is an information–based distance measure, rather than the sum of squared errors [15, 33, 34, 78].

A classification model $\mathcal{H}$ consists of a specification of its architecture $\mathcal{A}$ and the regulariser $\mathcal{R}$ for its parameters $\mathbf{w}$. When a classification model's parameters are set to a particular value, the model produces an output $y(\mathbf{x}; \mathbf{w}, \mathcal{A})$ between 0 and 1, which is viewed as the probability $P(t = 1|\mathbf{x}, \mathbf{w}, \mathcal{A})$. The likelihood, *i.e.*, the probability of the data[2] as a function of $\mathbf{w}$, is then:

$$\begin{aligned} P(D\,|\mathbf{w}, \mathcal{A}) &= \prod_m y^{t_m}(1-y)^{1-t_m} \\ &= \exp G(D\,|\mathbf{w}, \mathcal{A}), \end{aligned}$$

where

$$G(D\,|\mathbf{w}, \mathcal{A}) = \sum_m t_m \log y + (1 - t_m)\log(1 - y). \tag{5.1}$$

This is the probabilistic motivation for the cross–entropy objective function $\sum p \log \frac{q}{p}$. Now if we assign a prior to alternative parameter vectors $\mathbf{w}$,

$$P(\mathbf{w}|\{\alpha_c\}, \mathcal{A}, \mathcal{R}) = \frac{\exp(-\sum_c \alpha_c E_W^{(c)})}{Z_W}, \tag{5.2}$$

where $E_W^{(c)}$ is a cost function for a subset (c) of the parameters, and α_c is the associated regularisation constant (see chapter 3), we obtain a posterior:

$$P(\mathbf{w}|D, \{\alpha_c\}, \mathcal{A}, \mathcal{R}) = \frac{\exp(-\sum_c \alpha_c E_W^{(c)} + G)}{Z_M}, \tag{5.3}$$

where Z_W and Z_M are the appropriate normalising constants. Thus, the identical framework is obtained to that in chapter 3, with $-G$ replacing the term βE_D. Note that in contrast to the framework in chapter 3 there is now no free parameter β and no $Z_D(\beta)$. If however a teacher were to supply probability estimates t instead of binary targets, then a constant

[2] Strictly this is the probability of $\{t_m\}$ given $\{\mathbf{x}^{(m)}\}, \mathbf{w}, \mathcal{A}$; the density over $\{\mathbf{x}\}$ is not modelled by the 'discriminative' classifiers discussed in this chapter.

equivalent to β would appear, expressing the precision of the teacher's estimates. This constant would correspond to the effective number of observations on which the teacher's opinion is based.

The calculation of the gradient and Hessian of G is as easy as for a quadratic E_D, if the output unit's activation function is the traditional logistic $f(a) = 1/(1 + \bar{e}^{-a})$, or the generalised 'softmax' in the case of more than two classes [15]. The appropriateness of a logisitic output function for a classifier is well known: it is the function that converts a log probability ratio a into a probability $f(a)$.

Gradient: If $y(\mathbf{x}^{(m)}) = f(a(\mathbf{x}^{(m)}))$ as defined above, the gradient of G with respewct to the parameters $\mathbf{w}$ is

$$\nabla G = \sum_m (t_m - y)\mathbf{g}_{(m)}, \tag{5.4}$$

where $\mathbf{g}_{(m)} = \partial a / \partial \mathbf{w}|_{\mathbf{x}=\mathbf{x}^{(m)}}$.

Hessian: The Hessian can be analytically evaluated [9], but a useful approximation neglecting terms in $\partial^2 a / \partial^2 \mathbf{w}$ is:

$$\nabla\nabla G \simeq -\sum_m f' \mathbf{g}_{(m)} \mathbf{g}_{(m)}^{\mathrm{T}}. \tag{5.5}$$

This approximation is expected to be adequate for the evaluation of error bars, for use in data selection and for the evaluation of the number of well–determined parameters γ. A more accurate evaluation of the Hessian is probably needed for estimation of the evidence. In this chapter's demonstrations, the Hessian is evaluated using second differences.

Validity of approximations

On account of the central limit theorem, we expect the posterior distribution to converge to a set of locally Gaussian peaks with increasing quantities of data. However, the quadratic approximation to G is expected to converge more slowly than the quadratic approximation to E_D, the error function for regression models, because (a) G is not a quadratic function even for a linear model (a model for which $a = \sum w_h \phi_h(\mathbf{x})$): each term in G has the large scale form of a ramp function; and (b) only inputs which fall in the 'bend' of the ramp contribute curvature to G. If we have the opportunity for active data selection we could improve the convergence of this quadratic approximation by selecting inputs that are expected to contribute maximal curvature. A related data selection criterion is derived in section 5.4.

5.2 Every classifier should have two sets of outputs

Consider a classifier with output $y(\mathbf{x}; \mathbf{w}) = f(a(\mathbf{x}; \mathbf{w}))$. Assume that we receive data D and infer the posterior probability of the parameters $\mathbf{w}$ (*i.e.*, we perform 'learning'). Now if we are asked to make predictions with this classifier, it is common for the most probable or best fit parameter vector $\mathbf{w}_{\mathrm{MP}}$ to be used as the sole representative of the posterior distribution. This strategy seems unwise, however, since there may be regions in input space where the posterior ensemble is very uncertain about what the class is; in such regions the output of the network should be $y \simeq 0.5$ (assuming equiprobable classes *a priori*), whereas typically the network with parameters $\mathbf{w}_{\mathrm{MP}}$ will give a more extreme, unrepresentative and overconfident

output. *The error bars on the parameters should be taken into account when predictions are made.*

In regression problems, it is also important to calculate error bars on outputs, but the problem is more acute in the case of classification because, on account of the non–linear output, the mean output over the posterior distribution is not equal to the most probable network's output. To obtain an output representative of the posterior ensemble of networks around $\mathbf{w}_{\rm MP}$, we need to *moderate* the output of the most probable network in relation to the error bars on $\mathbf{w}_{\rm MP}$.

Of course this idea of averaging over the hidden parameters is not new: marginalisation goes back to Laplace. More recently, and in a context closer to the present one, the same message can be found for example in [79]. But it seems that most practitioners of adaptive classification do not currently use marginalisation.

I suggest that any classifier should have two sets of outputs. The first set would give the usual class probabilities corresponding to $\mathbf{w}_{\rm MP}$, $y(\mathbf{x};\mathbf{w}_{\rm MP})$; these outputs would be used for learning, *i.e.*, for calculating the error signals for optimisation of $\mathbf{w}_{\rm MP}$. The second set would be the moderated outputs $y(\mathbf{x};P(\mathbf{w}|D)) = \int d^k\mathbf{w}\, y(\mathbf{x};\mathbf{w})P(\mathbf{w}|D)$; these outputs would be used for all other applications, *e.g.*, prediction, evaluation of test error, and for evaluating the utility of candidate data points (section 5.4). Let us now discuss how to calculate the moderated outputs. It will then be demonstrated that these outputs do indeed make better predictions.

Calculating the moderated outputs

If we assume a locally Gaussian posterior probability distribution[3] over $\mathbf{w} = \mathbf{w}_{\rm MP} + \Delta\mathbf{w}$, $P(\mathbf{w}|D) \simeq P(\mathbf{w}_{\rm MP})\exp(-\frac{1}{2}\Delta\mathbf{w}^{\rm T}\mathbf{A}\Delta\mathbf{w})$, and if we assume that the activation $a(\mathbf{x};\mathbf{w})$ is a locally linear function of $\mathbf{w}$ with $\partial a/\partial\mathbf{w} = \mathbf{g}$, then for any $\mathbf{x}$, the activation a is approximately Gaussian distributed:

$$P(a(\mathbf{x})|D) = \text{Normal}(a^{\rm MP}, s^2) = \frac{1}{\sqrt{2\pi s^2}}\exp\left(-\frac{(a-a^{\rm MP})^2}{2s^2}\right), \tag{5.6}$$

where $a^{\rm MP} = a(\mathbf{x};\mathbf{w}_{\rm MP})$ and $s^2 = \mathbf{g}^{\rm T}\mathbf{A}^{-1}\mathbf{g}$. This means that the moderated output is:

$$P(t=1|\mathbf{x},D) = \psi(a^{\rm MP}, s^2) \equiv \int da\, f(a)\,\text{Normal}(a^{\rm MP}, s^2). \tag{5.7}$$

This is to be contrasted with the most probable network's output, $y(\mathbf{x};\mathbf{w}_{\rm MP}) = f(a^{\rm MP})$. The integral of a sigmoid times a Gaussian cannot be solved analytically; here I suggest a simple numerical approximation to it:

$$\psi(a^{\rm MP}, s^2) \simeq \phi(a^{\rm MP}, s^2) \equiv f(\kappa(s)a^{\rm MP}) \tag{5.8}$$

with $\kappa = 1/\sqrt{1+\pi s^2/8}$. This approximation is not globally accurate over $(a^{\rm MP}, s^2)$, (for large $s^2 > a$ the function should tend to an error function, not a logisitic) but it breaks down gracefully. The value of κ was chosen so that the approximation has the correct gain at $a^{\rm MP} = 0$, as $s^2 \to \infty$. A representative of this approximation is given in figure 5.1 which compares ϕ and ϕ' with numerical evaluations of ψ and ψ'. A similar approximation in terms of the error function is suggested in [79].

[3]Conditioning variables such as $\mathcal{A}, \mathcal{R}, \{\alpha_c\}$ will be omitted in this section, since the emphasis is not on model comparison.

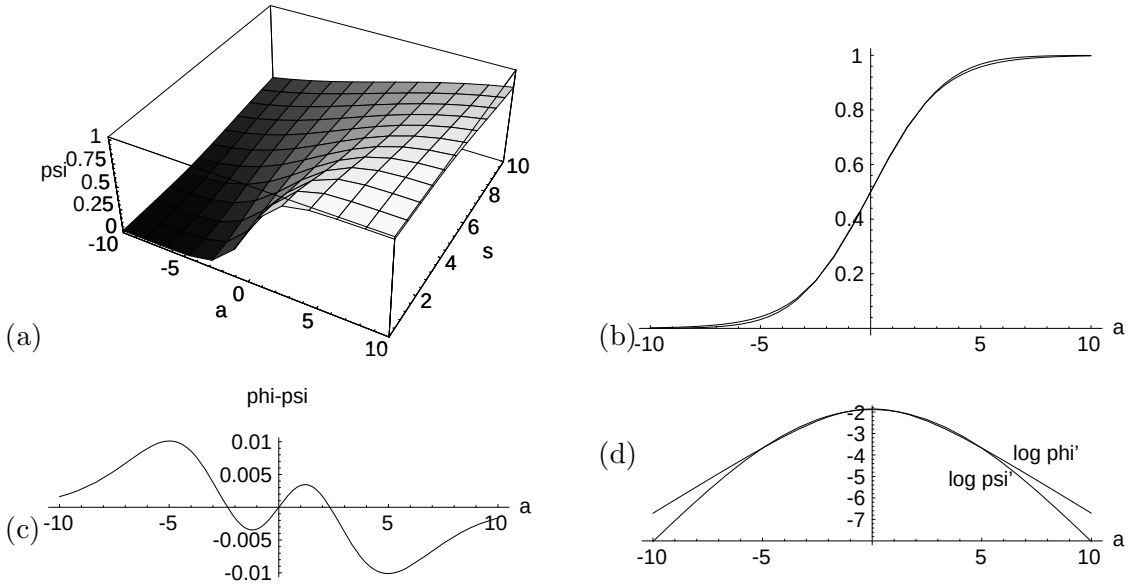

Figure 5.1: **Approximation to the moderated probability**
(a) The function $\psi(a, s^2)$, evaluated numerically. In (b) the functions $\psi(a, s^2)$ and $\phi(a, s^2)$ defined in the text are shown as a function of a for $s^2 = 4$. In (c), the difference $\phi - \psi$ is shown for the same parameter values. In (d), the breakdown of the approximation is emphasised by showing $\log \phi'$ and $\log \psi'$ (derivatives with respect to a). The errors become significant when $a \gg s$.

If the output is immediately used to make a (0/1) decision, then the use of moderated outputs will make no difference to the performance of the classifier (unless the costs associated with error are assymetrical), since both functions pass through 0.5 at $a^{\mathrm{MP}} = 0$. But moderated outputs will make a difference if a more sophisticated penalty function is involved. In the following demonstration the performance of a classifier's outputs is measured by the value of G achieved on a test set.

A model classification problem with two input variables and two possible classes is shown in figure 5.2a. Figure 5.2b illustrates the output of a typical trained network, using its *most probable* parameter values. Figure 5.2c shows the *moderated* outputs of the same network. Notice how the moderated output is similar to the most probable output in regions where the data are dense. In contrast, where the data are sparse, the moderated output becomes significantly less certain than the most probable output; this can be seen by the widening of the contours. Figure 5.2d shows the correct posterior probability for this problem given the knowledge of the true class densities.

Several hundred neural networks having two inputs, one hidden layer of sigmoid units and one sigmoid output unit were trained on this problem. During optimisation, the second weight decay scheme of chapter 3 was used, using independent decay rates for each of three weight classes: hidden weights, hidden unit biases, and output weights and biases. This corresponds to the prior that models the weights in each class as coming from a Gaussian; the scale of the Gaussians for different classes are independent and are specified by regularising constants α_c. Each regularising constant is optimised on line by intermittently updating it to its *most probable* value as estimated within the 'evidence' framework.

The prediction abilities of a hundred networks using their 'most probable' outputs and using the moderated outputs suggested above are compared in figure 5.3. It can be seen

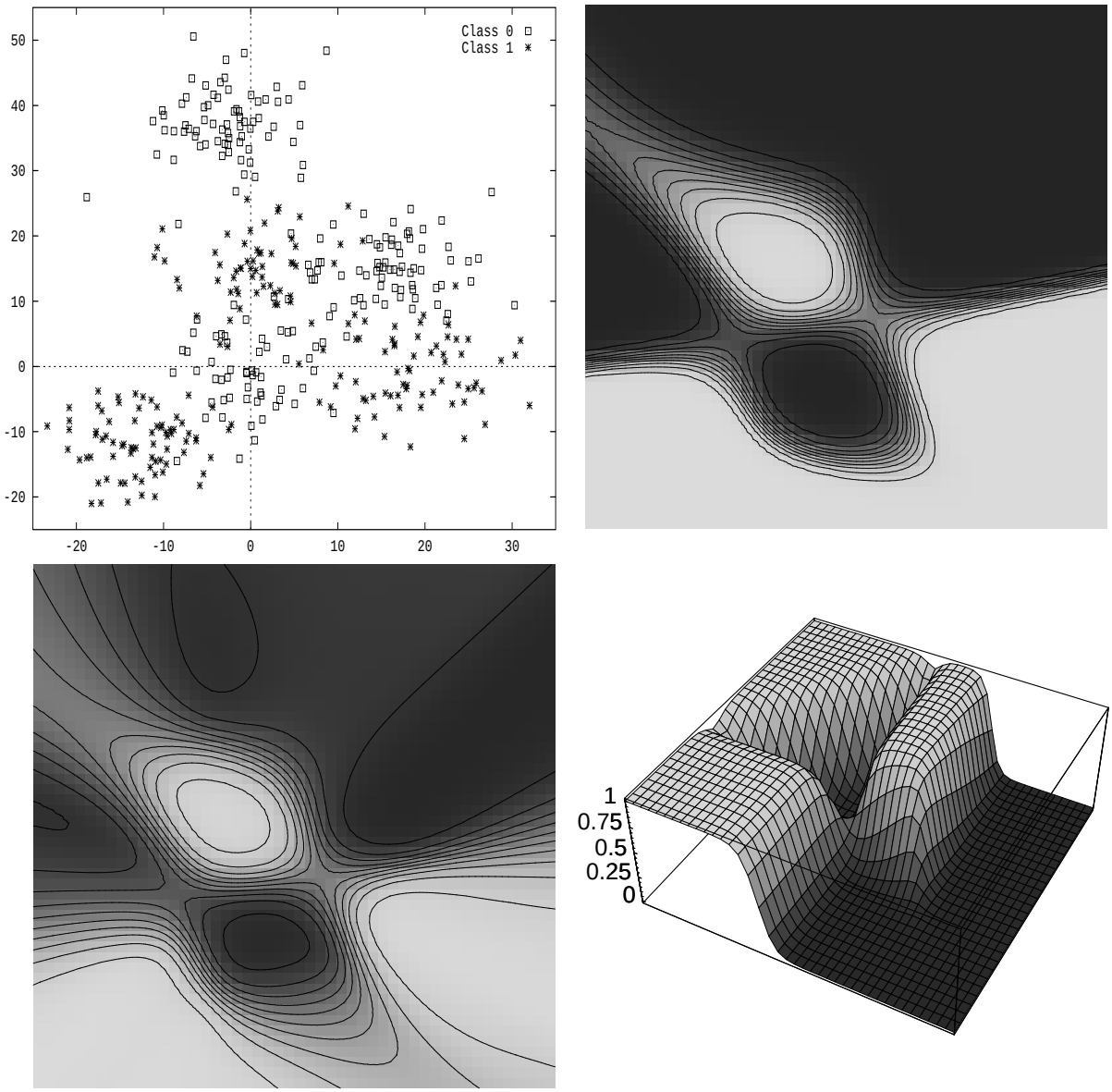

Figure 5.2: **Comparison of most probable outputs and moderated outputs for a toy problem**

a) The data set. The data were generated from six circular Gaussian distributions, three Gaussians for each class. The training sets for the demonstrations use between 100 and 1000 data points drawn from this distribution. b) (Upper right) 'Most probable' output of an eight hidden unit network trained on 100 data points. The contours are equally spaced between 0.0 and 1.0. c) (Lower left) 'Moderated' output of the network. Notice that the output becomes less certain compared with the most probable output as the input moves away from regions of high training data density. d) The true posterior probability, given the class densities that generated the data. The viewpoint is from the upper right corner of (a). In (b,c,d) a common grey scale is used, linear from 0 (dark grey) to 1 (light grey).

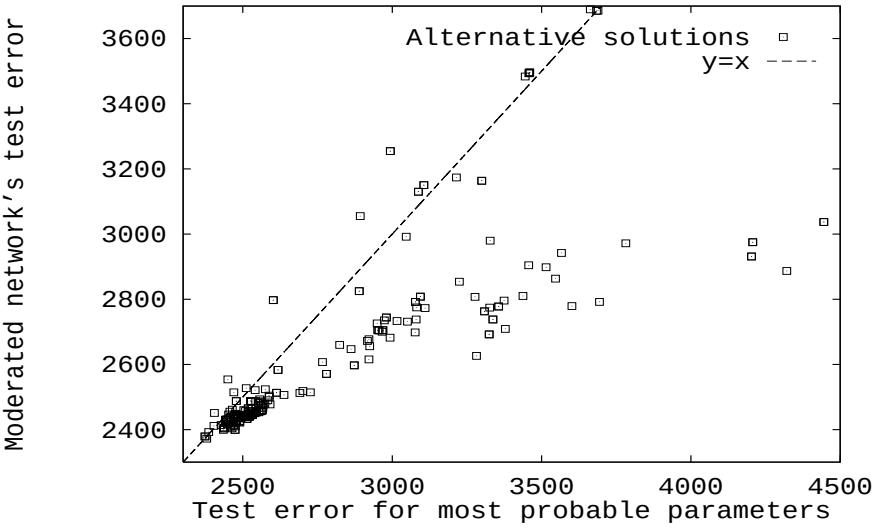

Figure 5.3: **Moderation is a good thing!**
The training set for all the networks contained 300 data points. For each network, the test error of the 'most probable' outputs and the 'moderated' outputs were evaluated on a test set of 5000 data points. The test error is the value of G. Note that for most solutions, the moderated outputs make better predictions.

that the predictions given by the moderated outputs are in nearly all cases superior. The improvement is most substantial for underdetermined networks with relatively poor performance. In a small fraction of the solutions however, especially among the best solutions, the moderated outputs are found to have slightly but significantly inferior performance.

5.3 Evaluating the evidence

Having established how to use a particular model $\mathcal{H} = \{\mathcal{A}, \mathcal{R}\}$ with given regularising constants $\{\alpha_c\}$ to make predictions, we now turn to the question of model comparison. As discussed in chapter 2, three levels of inference can be distinguished: parameter estimation, regularisation constant determination, and model comparison.[4] The second two levels of inference both require 'Occam's razor'; that is, the solution that best fits the data is not the most plausible model, and we need a way to balance goodness of fit against complexity. Bayesian inference embodies such an Occam's razor automatically.

At the first level, a model $\mathcal{H}$, with given regularising constants $\{\alpha_c\}$ is fitted to the data D. This involves inferring what value the parameters $\mathbf{w}$ should probably have. Bayes' rule for this level of inference has the form:

$$P(\mathbf{w}|D, \{\alpha_c\}, \mathcal{H}) = \frac{P(D|\mathbf{w}, \{\alpha_c\}, \mathcal{H})P(\mathbf{w}|\{\alpha_c\}, \mathcal{H})}{P(D|\{\alpha_c\}, \mathcal{H})}. \tag{5.9}$$

Throughout this thesis this posterior is approximated locally by a Gaussian:

$$P(\mathbf{w}|D, \{\alpha_c\}, \mathcal{H}) = \frac{\exp(-M(\mathbf{w}))}{Z_M} \simeq \frac{\exp(-M_{\rm MP} - \frac{1}{2}\Delta\mathbf{w}^{\rm T}\mathbf{A}\Delta\mathbf{w})}{Z_M^*}, \tag{5.10}$$

[4]The use of a specified model to predict the class of a datum can be viewed as the zeroeth level of inference.

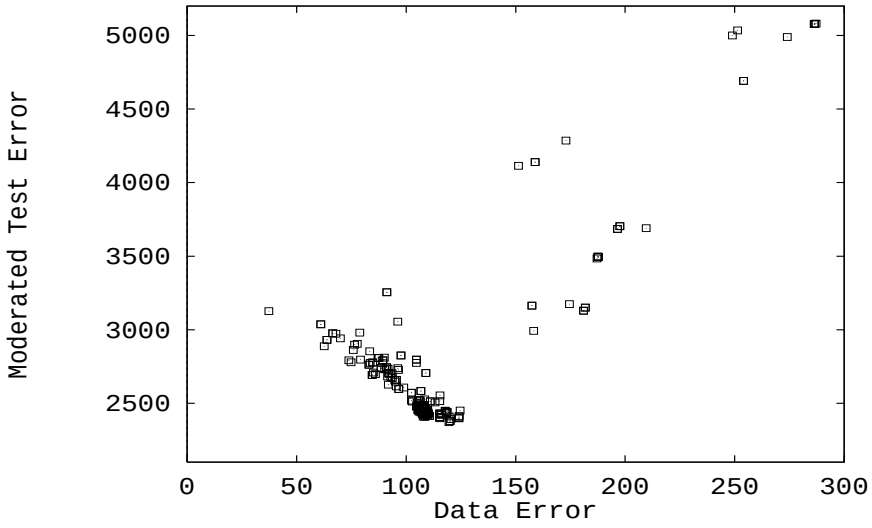

Figure 5.4: **Test error versus data error**
This figure illustrates that the task of ranking solutions to the classification problem requires Occam's razor; the solutions with smallest data error do not generalise best.

where $\Delta\mathbf{w} = \mathbf{w} - \mathbf{w}_{\rm MP}$, $M(\mathbf{w}) = \sum_c \alpha_c E_W^{(c)} - G$, and $\mathbf{A} = \nabla\nabla M$.

At the second level of inference, the regularising constants are optimised:

$$P(\{\alpha_c\}|D,\mathcal{H}) = \frac{P(D|\{\alpha_c\},\mathcal{H})P(\{\alpha_c\}|\mathcal{H})}{P(D|\mathcal{H})}. \tag{5.11}$$

The data–dependent term $P(D|\{\alpha_c\},\mathcal{H})$ is the 'evidence', the normalising constant from equation (5.9). The evaluation of this quantity and the optimisation of the parameters $\{\alpha_c\}$ is accomplished using a framework due to Gull and Skilling, discussed in detail in chapters 2 and 3.

Finally, at the third level of inference, the alternative models are compared:

$$P(\mathcal{H}|D) \propto P(D|\mathcal{H})P(\mathcal{H}). \tag{5.12}$$

Again, the data's opinion about the alternatives is given by the evidence from the previous level, in this case $P(D|\mathcal{H})$.

Omitting the details of the second level of inference, since they are identical to the methods in chapter 3, this demonstration presents the final inferences, the evidence for alternative solutions. The evidence is evaluated within the Gaussian approximation from the properties of the 'most probable' fit $\mathbf{w}_{\rm MP}$, and the error bars $\mathbf{A}^{-1}$, as described in chapter 2.

Figure 5.4 shows the test error (calculated using the moderated outputs) of the solutions against the data error, and the 'Occam's razor' problem can be seen: the solutions with smallest data error do not generalise best. Figure 5.5 shows the log evidence for the solutions against the test error, and it can be seen that a moderately good correlation is obtained. The correlation is not perfect. It is speculated that the discrepancy is mainly due to inaccurate evaluation of the evidence under the quadratic approximation, but further study is needed here. Finally, figure 5.6 explores the dependence of the correlation between evidence and generalisation on the amount of data. It can be seen that the correlation improves as the number of data points in the test set increases.

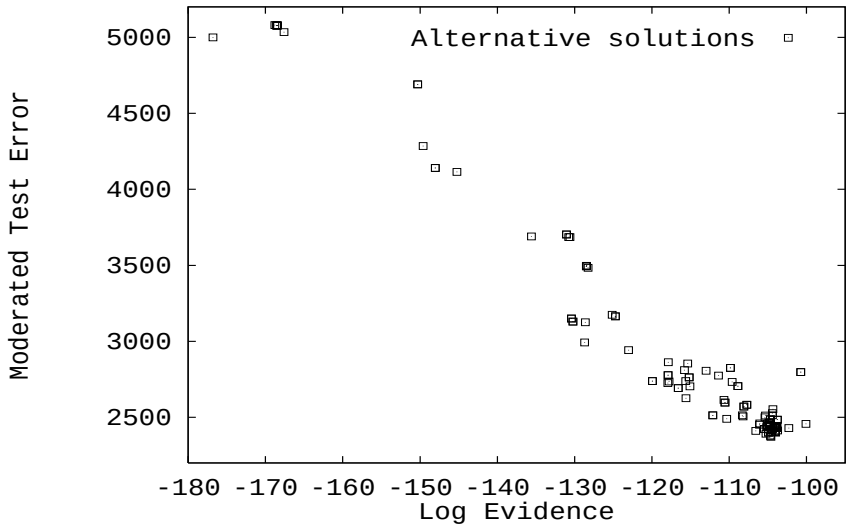

Figure 5.5: **Test error versus evidence**
Each solution was found using the same training set of $N = 300$ data points. All solutions in which a symmetry was detected among the hidden units were omitted from this graph because the evidence evaluation for such solutions is unreliable.

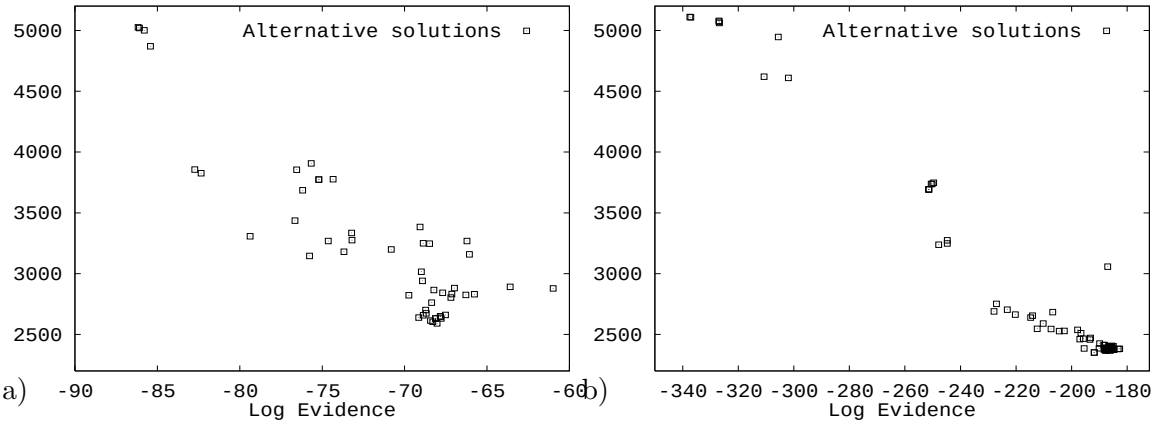

Figure 5.6: **Correlation between test error and evidence as the amount of data varies.**
a) $N = 150$ data points. b) $N = 600$ data points. c.f. Figure 5.5, for which $N = 300$. For comparison, the number of parameters in a typical (10 hidden unit) network is 41. Note that only about 25% of the data points fall in informative decision regions; so the effective number of data points is smaller in each case; bear in mind also that each data point only consists of one bit. All solutions in which a symmetry was detected among the hidden units were omitted because the evidence evaluation for such solutions is unreliable.

5.4 Active learning

Assume now that we have the opportunity to select the input $\mathbf{x}$ where a future datum will be gathered ('query learning'). Several papers have suggested strategies for this active learning problem, for example Hwang *et al.* [35] propose that samples should be made on and near the current decision boundaries. This strategy and that of Baum [5] are both human–designed strategies and it is not clear what objective function if any they optimise, nor is it clear how the strategies could be improved. In this chapter, as in chapter 4, the philosophy will be to *derive* a criterion from a *defined* sensible objective function that measures how useful a datum is expected to be. This criterion may then be used as a guide for query learning, or for the alternative scenario of pruning uninformative data points from a large data set.

Desiderata

Let us criticise Hwang *et al.*'s strategy to try to establish a reasonable objective function. The strategy of sampling on decision boundaries is motivated by the argument that we are unlikely to gain information by sampling in a region where we are already confident of the correct classification. But similarly, if we have already sampled a great deal on one particular boundary then we don't gain useful information by repeatedly sampling there either, because the location of the boundary has been established! Repeated sampling at such locations generates data with large entropy that are 'informative' in the same way that white noise is informative. There must be more to the utility of a sample than its distance from a decision boundary. We would prefer to sample near boundaries whose location has not been well determined, because this would probably enable us to make more precise predictions there. Thus we are interested in measurements which convey *mutual* information about the unknowns that we are interested in.

A second criticism is that a strategy that only samples near existing boundaries is not likely to make new discoveries; a strategy that also samples near *potential* boundaries is expected to be more informative. A final criticism is that to be efficient, a strategy should take into account how influential a datum will be: some data may convey information about the discriminant over a larger region than others. So we want an objective function that measures the global expected informativeness of a datum.

Objective function

This chapter will study the 'mean marginal information'. This objective function was suggested in chapter 4, and a discussion of why it is probably more desirable than the joint information is given there. To define this objective function, we first have to define a region of interest. (The objective of maximal information gain about the model's parameters without a region of interest would lead us to sample at unsampled extremes of the input space.) Here this region of interest will be defined by a set of representative points $\mathbf{x}^{(u)}$, $u = 1 \ldots V$, with a normalised distribution P_u on them. P_u can be interpreted as the probability that we will be asked to make a prediction at $\mathbf{x}^{(u)}$. (The theory could be worked out for the case of a continuous region defined by a density $\rho(\mathbf{x})$, but the discrete case is preferred since it relates directly to practical implementation.) The marginal entropy of a distribution over $\mathbf{w}$, $P(\mathbf{w})$, at one point $\mathbf{x}^{(u)}$ is defined to be

$$S_M^{(u)} = y_u \log y_u + (1 - y_u) \log(1 - y_u), \tag{5.13}$$

where $y_u = y(\mathbf{x}^{(u)}; P(\mathbf{w}))$ is the average output of the classifier over the ensemble $P(\mathbf{w})$. Under the Gaussian approximation for $P(\mathbf{w})$, y_u is given by the moderated output (5.7), and may be approximated by $\phi(a_u^{\rm MP}, s_u^2)$ (5.8).

The mean marginal entropy is

$$\bar{S}_M(P(\mathbf{w})) = \sum_u P_u S_M^{(u)}. \tag{5.14}$$

The sampling strategy studied here is to maximise the expected change in mean marginal entropy. (Note that our information gain is *minus* the change in entropy.)

Estimating marginal entropy changes

Let a measurement be made at $\mathbf{x}$. The result of this measurement is either $t = 1$ or $t=0$. Assuming that our current model, complete with Gaussian error bars, is correct, the probability of $t=1$ is $\psi(a^{\rm MP}(\mathbf{x}), s^2(\mathbf{x})) \simeq \phi(a^{\rm MP}, s^2)$. We wish to estimate the average change in marginal entropy of t_u at $\mathbf{x}^{(u)}$ when this measurement is made.

This problem can be solved by calculating the joint probability distribution $P(t, t_u)$ of t and t_u, then finding the mutual information between the two variables. The four values of $P(t, t_u)$ have the form:

$$P(t=1, t_u=1) = \iint da\, da_u\, f(a) f(a_u) \frac{1}{Z} \exp\left(-\frac{1}{2}\Delta\mathbf{a}^{\rm T}\Sigma^{-1}\Delta\mathbf{a}\right), \text{ etc.,} \tag{5.15}$$

where $\Delta\mathbf{a}^{\rm T} = (\Delta a, \Delta a_u)$ and the activations $a = a^{\rm MP} + \Delta a$ and $a_u = a_u^{\rm MP} + \Delta a_u$ are assumed to have a Gaussian distribution with covariance matrix

$$\Sigma = \begin{pmatrix} \mathbf{g}^{\rm T}\mathbf{A}^{-1}\mathbf{g} & \mathbf{g}^{\rm T}\mathbf{A}^{-1}\mathbf{g}_{(u)} \\ \mathbf{g}^{\rm T}\mathbf{A}^{-1}\mathbf{g}_{(u)} & \mathbf{g}_{(u)}^{\rm T}\mathbf{A}^{-1}\mathbf{g}_{(u)} \end{pmatrix} \equiv \begin{pmatrix} s^2 & \rho s s_u \\ \rho s s_u & s_u^2 \end{pmatrix}. \tag{5.16}$$

The normalising constant is $Z = 2\pi s s_u (1-\rho^2)^{\frac{1}{2}}$. The expected change in entropy of t_u is:

$$E(\Delta S_M^{(u)} | t) = S(P(t, t_u)) - S(P(t)) - S(P(t_u)). \tag{5.17}$$

Notice that this mutual information is symmetric in t and t_u. We can approximate $E(\Delta S_M^{(u)}|t)$ by Taylor–expanding $P(t, t_u)$ about independence ($\rho = 0$). The first order perturbation to $P(t, t_u)$ introduced by ρ can be written in terms of a single variable c:

$$\begin{array}{ll} P(t=1,t_u=1) = P(t=1)P(t_u=1) + c & P(t=1,t_u=0) = P(t=1)P(t_u=0) - c \\ P(t=0,t_u=1) = P(t=0)P(t_u=1) - c & P(t=0,t_u=0) = P(t=0)P(t_u=0) + c. \end{array} \tag{5.18}$$

Taylor–expanding (5.17), we find

$$E(\Delta S_M^{(u)}|t) \simeq -\frac{1}{P(t=1)P(t_u=1)P(t=0)P(t_u=0)}\, c^2/2. \tag{5.19}$$

Finally, we Taylor–expand (5.15) so as to obtain the dependence of c on the correlation between the activations. The derivative of $P(t=1, t_u=1)$ with respect to ρ at $\rho = 0$ is

$$\begin{aligned} \frac{\partial}{\partial\rho} P(t=1, t_u=1) &= \iint da\, da_u\, f(a) f(a_u) \frac{\Delta a\, \Delta a_u}{s s_u} \frac{1}{Z} \exp\left(-\frac{1}{2}\Delta\mathbf{a}^{\rm T}\Sigma^{-1}\Delta\mathbf{a}\right) \\ &= s\psi'(a^{\rm MP}, s^2)\, s_u \psi'(a_u^{\rm MP}, s_u^2), \end{aligned}$$

where ψ is the moderated probability defined in (5.8) and ψ' denotes $\partial\psi/\partial a$. This yields

$$c \simeq \rho \frac{\partial}{\partial \rho} P(t{=}1, t_u{=}1) = \mathbf{g}^{\mathrm{T}} \mathbf{A}^{-1} \mathbf{g}_{(u)} \; \psi'(a^{\mathrm{MP}}, s^2) \psi'(a_u^{\mathrm{MP}}, s_u^2). \tag{5.20}$$

Substituting this into (5.19), we find

$$E(\Delta S_M^{(u)} | t) \simeq -\frac{(\mathbf{g}^{\mathrm{T}} \mathbf{A}^{-1} \mathbf{g}_{(u)})^2 \; \psi'(a^{\mathrm{MP}}, s^2)^2 \; \psi'(a_u^{\mathrm{MP}}, s_u^2)^2}{2 \; P(t{=}1) P(t_u{=}1) P(t{=}0) P(t_u{=}0)}. \tag{5.21}$$

Assuming that the approximation $\psi \simeq \phi \equiv f(\kappa(s) a^{\mathrm{MP}})$ is good, we can numerically approximate $\partial\psi(a^{\mathrm{MP}}, s^2)/\partial a$ by $\kappa(s) f'(\kappa(s) a^{\mathrm{MP}})$.[5] Using $f' = f(1-f)$ we obtain

$$E(\Delta S_M^{(u)} | t) \simeq -\kappa(s)^2 \kappa(s_u)^2 \; f'(\kappa(s) a^{\mathrm{MP}}) f'(\kappa(s_u) a_u^{\mathrm{MP}}) \; (\mathbf{g}^{\mathrm{T}} \mathbf{A}^{-1} \mathbf{g}_{(u)})^2 / 2. \tag{5.22}$$

The two f' terms in this expression correspond to the two intuitions that sampling near decision boundaries is informative, and that we are able to gain more information about points of interest if they are near boundaries. The term $(\mathbf{g}^{\mathrm{T}} \mathbf{A}^{-1} \mathbf{g}_{(u)})^2$ modifies this tendency in accordance with the desiderata.

The expected mean marginal information gain is computed by adding up the $\Delta S_M^{(u)}$s over the representative points $\mathbf{x}^{(u)}$. The resulting function is plotted on a grey scale in figure 5.7, for the network solving the toy problem described in figure 5.2. For this demonstration the points of interest $\mathbf{x}^{(u)}$ were defined by drawing 100 input points at random from the test set. A striking correlation can be seen between the regions in which the moderated output is uncertain and regions of high expected information gain. In addition the expected information gain tends to increase in regions where the training data were sparse.

Now to the negative aspect of these results. The regions of greatest expected information gain lie *outside* the region of interest to the right and left; these regions extend in long straight ridges hundreds of units away from the data. This estimation of utility, which reveals the 'hyperplanes' underlying the model, seems unreasonable. The utility of points so far from the region of interest, if they occurred, could not really be so high. There are two plausible explanations of this. It may be that the Taylor approximations used to evaluate the mean marginal information are at fault, in particular (5.20). Or as discussed in chapter 4, the problem might arise because the mean marginal information estimates the utility of a point assuming that the model is true; if we assume that the classification surface really can be described in terms of hyperplanes in the input space, then it may be that the greatest torque on those planes can be obtained by sampling away from the core of the data. Comparison of the approximation (5.22) with numerical evaluations of $\Delta S_M^{(u)}$ indicate that the approximation is never more than a factor of two wrong. Thus the latter explanation is favoured, and we must tentatively conclude that the mean marginal information gain is likely to be most useful only for models well matched to the real world.

5.5 Discussion

Moderated outputs: The idea of moderating the outputs of a classifier in accordance with the uncertainty of its parameters should have wide applicability, for example to hidden Markov models for speech recognition. Moderation should be especially important

[5] This approximation becomes inaccurate where $a^{\mathrm{MP}} \gg s \gg 1$ (see figure 5.1c). Because of this it might be wise to use numerical integration then implement $\Delta S_M^{(u)}$ in look–up tables.

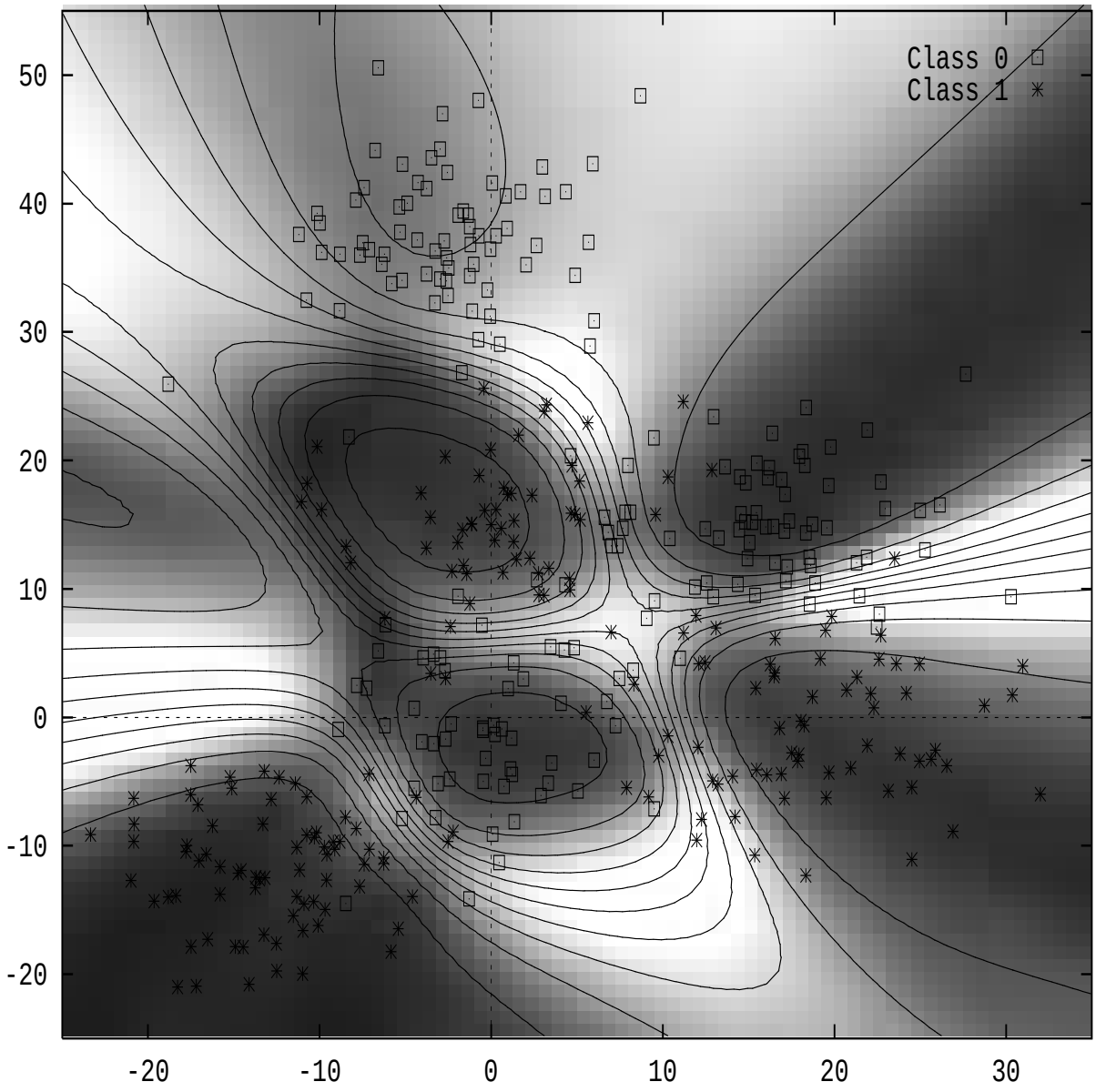

Figure 5.7: **Demonstration of expected mean marginal information gain**
The mean marginal information gain was computed for the network demonstrated in figures 5.2b,c. The region of interest was defined by 100 data points from the test set. The grey level represents the utility of a single observation as a function of where it is made. The darkest regions are expected to yield little information, and white corresponds to large expected information gain. The contours that are superposed represent the moderated output of the network, as shown in figure 5.2c. The mean marginal information gain is quantified: the grey scale is linear from 0 to 0.0025 nats.

where a classifier is expected to extrapolate to points outside the training region. There is presumably a relationship of this concept to the work of Seung *et al.* [69] on generalisation 'at non–zero temperature'.

If the suggested approximation to the moderated output and its derivative is found dissatisfactory, a simple brute force solution would be to set up a look–up table of values of $\psi(a, s^2)$ and $\psi'(a, s^2)$.

It is likely that an implementation of marginalisation that will scale up well to large problems will involve Monte Carlo methods [56].

Evidence: The evidence has been found to be well correlated with generalisation ability. This depends on having a sufficiently large amount of data. There remain open questions, including what the theoretical relationship between the evidence and generalisation ability is, and how large the data set must be for the two to be well correlated; how well these calculations will scale up to larger problems; and when the quadratic approximation for the evidence breaks down.

Mean marginal information gain: This objective function was derived with active learning in mind. It could also be used for selection of a subset of a large quantity of data, as a filter to weed out fractions of the data which are unlikely to be informative. Unlike Plutowski and White's approach [62] this filter only depends on the *input* variables in the candidate data. A strategy that selectively omits data on the basis of their *output* values would violate the likelihood principle and risk leading to inconsistent inferences.

A comparison of the mean marginal information gain in figure 5.7 with the contours of the most probable networks output in figure 5.2b indicates that this proposed data selection criterion offers some improvements over the simple strategy of just sampling on and near decision boundaries: the mean marginal information gain shows a plausible preference for samples in regions where the decision boundary is uncertain. On the other hand, this criterion may give artefacts when applied to models that are poorly matched to the real world. How useful the mean marginal information gain will be for real applications remains an open question.

Chapter 6

Inferring an Input-dependent Noise Level

Abstract

Assume that when interpolating a data set, we wish to model an input–dependent noise level. This short chapter shows how to calculate an unbiased gradient.

Given a data set $D = \{\mathbf{x}_m, t_m\}$ modelled with an interpolant–plus–noise model $t_m = y(\mathbf{x}_m; \mathbf{w}) + \nu_m$, chapter 2 described the Bayesian framework for regularisation and model comparison assuming a single global noise level $\beta^{-1} = \sigma_\nu^2$. It is also possible to invent models in which β is $\mathbf{x}$–dependent. For example we might use two coupled neural networks, the first of which predicts $y(\mathbf{x})$, and the second of which predicts $\log \beta(\mathbf{x})$. These networks would be coupled in that the gradient of the objective function for each network would be calculated by consulting the output of the other. Intuitively, if the first network's errors in the neighbourhood of $\mathbf{x}$ are large, then we encourage the second network to give a large value of $\beta^{-1}(\mathbf{x}) = \sigma_\nu^2(\mathbf{x})$ there; and similarly the error signals that teach the first network need to be scaled up and down by the second network — where there is a small value of $\sigma_\nu^2(\mathbf{x})$, errors are penalised more strongly.

What should the gradients for optimisation of such a model be? A simple approach would be to maximise the likelihood of the data. For the traditional quadratic model, the log likelihood of the data is $-\sum_m \beta(\mathbf{x}_m) E_{Dm} + \sum_m \frac{1}{2} \log \beta(\mathbf{x}_m)$, where $E_{Dm} = \frac{1}{2}(t_m - y(\mathbf{x}_m; \mathbf{w}))^2$. However, maximisation of this function would lead to *biased* estimates of $\beta(\mathbf{x})$. As was discussed earlier, the maximum likelihood noise estimate is not the most probable value of the noise. This distinction is not caused by the use of priors; rather it is a result of *marginalisation*. The worst symptom of maximising the likelihood would be that in regions in which data is sparse, such that the best fit interpolant passes very close to the data, the maximum likelihood estimate of β blows up: the estimated noise level goes to zero.

Separating the two levels of inference

Let us consider the case of a single noise level σ^2 for a moment, and imagine that we are estimating a single parameter μ corresponding to the mean of a Gaussian cluster. We already examined this problem in chapter 2. The likelihood, as a function of μ and σ^2, has a skew peak. The maximum is located at $(\bar{x}, \sigma_N^2)$, where $\sigma_N^2 = \sum (x - \bar{x})^2 / N$, but this peak

[0]Chapter 6 of Ph.D. thesis 'Bayesian Methods for Adaptive Models' by David MacKay, California Institute of Technology, submitted December 10 1991.

is not in the same place as the centre of mass of the likelihood. When we marginalise over μ, with a flat prior, we find that the most probable value of σ^2 is $\sigma^2_{N-1} = \sum(x-\bar{x})^2/(N-1)$. The subtraction of one from the denominator represents the fact that one parameter μ has been determined by the data, which typically consumes one unit (χ^2) of noise. It is well known that σ^2_N and σ^2_{N-1} are respectively 'biased' and 'unbiased' estimators of variance.

The generalisation of this distinction has already been given. When we fit a regularised interpolation model, there are distinct levels of inference. At the first level, we assume a particular noise level β and regularisation constant α, and find the most probable parameters $\mathbf{w}$ with error bars. Then at the second level of inference, we compare alternative values of α, and alternative values of β. When this separation is made, we find that the most probable noise level is given by $\sigma^2 = 2E_D/(N-\gamma)$, where γ is the number of well–determined parameters. This quantity will be significantly less than the total number of parameters k if the regulariser (prior) is playing a significant role in determining the interpolant. The separation of the two levels of inference and the use of marginalisation thus leads to an unbiased estimator for β and an automatic Occam's razor for the choice of α.

Let us now see how this should work in the case of an $\mathbf{x}$–dependent noise level.

At the first level of inference, the gradients with respect to model parameters will be calculated in the obvious way: $\partial M/\partial \mathbf{w} = \alpha \partial E_W/\partial \mathbf{w} + \sum_m \beta(\mathbf{x}_m)\partial E_{Dm}/\partial \mathbf{w}$.

The second level of inference centres on the log evidence for α and β, which can be written (neglecting additive constants):

$$\log P(D|\alpha, \beta(\mathbf{x}), \mathcal{H}) = -\alpha E_W^{\mathrm{MP}} - \sum_m \beta(\mathbf{x}_m) E_{Dm}^{\mathrm{MP}} - \frac{1}{2}\log\det \mathbf{A} - \log Z_W(\alpha) + \sum_m \frac{1}{2}\log\beta(\mathbf{x}_m). \tag{6.1}$$

The most probable values of α and β, if we have a flat prior, are obtained by maximising the log evidence. Of course if we infer an $\mathbf{x}$–dependent noise level, we typically *will* be imposing a prior on $\beta(\mathbf{x})$ by the choice of model; in this case we will still need the gradient of the evidence, which at this level serves as the likelihood driving the learning. When we differentiate the evidence with respect to $\log\beta(\mathbf{x}_m)$, we obtain:

$$\frac{\partial \log P(D|\alpha, \beta, \mathcal{A}, \mathcal{R})}{\partial \log \beta(\mathbf{x}_m)} = -\beta(\mathbf{x}_m) E_{Dm}^{\mathrm{MP}} - \frac{1}{2}\beta(\mathbf{x}_m)\,\mathrm{Trace}\left[\mathbf{A}^{-1}\mathbf{B}_m\right] + \frac{1}{2}, \tag{6.2}$$

where $\beta(\mathbf{x}_m)\mathbf{B}_m$ is the contribution to $\mathbf{A}$ made by the mth datum, which in the case of a linear model is $\beta(\mathbf{x}_m)\mathbf{B}_m = \beta(\mathbf{x}_m)\mathbf{g}_m\mathbf{g}_m^{\mathrm{T}}$, with $\mathbf{g}_m = \partial y(\mathbf{x}_m)/\partial \mathbf{w}$. The quantity $\mathrm{Trace}\left[\mathbf{A}^{-1}\mathbf{B}_m\right]$ is precisely the magnitude of the error bars on the interpolant at $\mathbf{x}_m$, $\mathbf{g}_m^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}_m$. Thus the term $\beta(\mathbf{x}_m)\mathrm{Trace}\left[\mathbf{A}^{-1}\mathbf{B}_m\right] = \mathbf{g}_m^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}_m/\sigma^2(\mathbf{x}_m)$ is the ratio of the error bars on the interpolant to the presumed variance of the measurement at $\mathbf{x}_m$. The quantity $\gamma_m \equiv 1 - \mathbf{g}_m^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{g}_m/\sigma^2(\mathbf{x}_m)$ is a measure of *how good a noise measurement* the datum at $\mathbf{x}_m$ contributed.

The gradient can then be written:

$$\frac{\partial \log P(D|\alpha, \beta, \mathcal{A}, \mathcal{R})}{\partial \log \beta(\mathbf{x}_m)} = -\beta(\mathbf{x}_m) E_{Dm}^{\mathrm{MP}} + \gamma_m/2. \tag{6.3}$$

The terms γ_m are easy to evaluate from the error bars at $\mathbf{x}_m$. It can be noted that in the case of an isolated data point, the contribution to the gradient is well–behaved, because E_{Dm} and γ both go to zero. So there is no singular behaviour involving β blowing up, as occurs if the likelihood is maximised.

If $\gamma_m \simeq 0$, the interpolant is determined locally only by the datum at $\mathbf{x}_m$, and the measurement gives no information about the noise. If $\gamma_m \simeq 1$, the interpolant is locally

determined more by the other measurements and by the regulariser than by t_m, and the error E_{Dm} does convey information about the noise level.

The similarity of γ_m to the concept of the number of well–determined parameters, γ, will be obvious, and in fact there is a mathematical relationship too. The number of *bad* noise measurements is identical to the number of well–determined parameters, *i.e.*, $\sum_m (1 - \gamma_m) = \gamma$.

An implementation of this framework will depend on two further issues. First, a prior should of course be placed on the parameters of the network that produces $\log \beta(\mathbf{x})$; this prior might contain unknown regularisation constants which can be controlled using the methods of chapter 3. Secondly, the management of these different levels of optimisation will not be trivial. A suggested procedure is to start optimising the noise model only once the interpolant is fitting the data quite well; then the three levels (fitting the interpolant, inferring the noise level, and setting regularisation constants of the noise model) could be optimised cyclically.

Chapter 7

Postscript

It is common that, following several years' devotion to a religion, a student's views about that religion will have matured. This postscript is intended to communicate the reservations and criticisms I now have about the Bayesian methods described, and the open questions and problems that remain.

7.1 The closed hypothesis space

Bayesian hard–liners often thump the 'Cox axioms' drum, proclaiming that consistent inference can only be Bayesian; but it is rarely made clear what Cox's axioms are. In fact Cox's result assumes (among other things) that we are performing inductive inference in a defined closed hypothesis space.

This is a two edged sword. The good aspect is that Bayesian inductive inference cannot proceed until all properties of our hypothesis space have been articulated; thus we are forced to make explicit all our assumptions. Furthermore, once the hypothesis space is defined, Bayesian inference is a mechanical and well–defined process. Bayesianism does not need to consult sampling theory criteria such as 'efficiency', 'unbiasedness', 'consistency', 'sufficiency', 'uniform convergence', or 'minimum variance' — desiderata which can often be mutually conflicting! We simply write down the conditional assumptions that we are making (for example the data that have occurred), and the propositions whose plausibilities we wish to infer, and evaluate $P(\text{Proposition}|\text{Assumptions})$ by using the sum and product rules of probability within the defined hypothesis space. The axioms on which probabilistic inference are based guarantee that inferences made in this way will be coherent.

On the other hand, several deficiencies arise from the constraint of a closed hypothesis space. The central problem is that our Bayesian inferences are obtained assuming that the hypothesis space is right, but we have no Bayesian way of assessing whether our hypothesis space *is* right, apart from coming up with alternative hypothesis spaces with which comparisons can be made. Box and Tiao [13] share the view that 'model criticism', an essential part of the modelling process, is not addressed by Bayesian inference.

For example, the error bars discussed throughout this thesis are evaluated assuming that the model is true. I do not think that any non–Bayesian procedures improve on this (orthodox confidence intervals are identical to Bayesian error bars in the Gaussian case), but it is important to be aware that it is possible for the true interpolant to lie well outside the error bars assigned by a model, if that model is defective in some way. An example of this can be seen in figure 2.4b, where the error bars fail to include a point that in fact was not an outlier. The Bayesian resolution of this is to examine other models; when the radial

basis function model in figure 2.4 is replaced by a more probable neural network model (Table 2.1), the interpolant goes much closer to this data point (which is probably why the neural network model is more probable).

A second defect of the closed hypothesis space assumption is discussed in Chapter 4. The expected information gain provided by a datum was defined, like the error bars, by assuming that the model is correct. In extreme cases this may lead to ludicrous results — distant data points may be evaluated as mutually informative because of a misleading interdependence in the model. In Chapter 5 it is shown that the mean marginal expected information gain does seem to be marred by artifacts arising from the assumption that the model is correct.

Thirdly, Bayes' rule provides no mechanism for 'alternative–free tests' of a hypothesis space. This reservation about Bayesian methods has also been expressed by Lindley [45]. Hard line Bayesians would retort that there is no such thing as an alternative–free test, and certainly most classical alternative–free tests do have an implicit alternative. For example, a classical test of a parameter being zero has as an implicit alternative the hypothesis that the parameter has a value in an interval with a derivable prior width [47]. But I do believe that we perform alternative–free tests. Often, we become dissatisfied with a theory because it seems to be making unusually poor predictions. This prompts us to start searching for superior theories. Without any alternative being more than very vaguely specified, we are able to *infer* that something is wrong (as, for example, in chapter 3). Once we find a superior alternative, we can *then* come back and use Bayes' rule to reject the original theory; but the initial decision to search for a new theory was alternative–free and could not be made with Bayes' rule alone. Nor can Bayes' rule alone direct you towards new models. The invention of hypothesis spaces remains a human's domain.

Having recognised that Bayes' rule cannot perform the alternative–free inferences that are part of the modelling process (the right–hand loop of figure 1.1), I would end on a more optimistic note. I believe that Bayesian methods, *together* with traditional methods such as cross–validation, yield a powerful tool for alternative–free hypothesis testing and new model formation. This was illustrated in Chapter 3, where the poor correlation between the evidence and the test error highlighted an inconsistency in the model space; if one only used the test error (cross–validation) for model comparison, this opportunity for learning would be lost; likewise, if only the Bayesian evidence were evaluated, we would be none the wiser. I think that Bayesians would do well to include similar alternative–free 'warning bells' in their algorithms.

7.2 For approximation, are probabilities relevant?

The Bayesian approach to model comparison evaluates how *probable* alternative models are given the data. In fields such as image reconstruction and NMR, this may be precisely the right thing to do. In contrast, in adaptive modelling, the real problem is often to estimate *how well each model is expected to generalise.* We know perfectly well that the truth is *not* that the data were generated by some neural network whose parameters we now wish to infer! We know that all the models are false, so the Bayesian assessment of the relative probabilities of alternative parameterised models seems almost irrelevant to what we are interested in, which is how well each of the models approximates. Really the Bayesian solution to this task would be to use a model that we really believe in to *infer* what the truth might be, then use *decision theory* to select from the false models the one that is expected to minimse the appropriate cost function.

The startling fact is that in spite of this, the evidence for the false models does seem to be well correlated with generalisation ability, when the model space is well–matched to the problem (figures 3.12 and 5.5). There are theories which attempt to directly predict generalisation ability, leading to Akaike's FPE criterion, and Moody's GPE [3, 54]. But for the toy problems I have studied, neither of these criteria has a better correlation with the generalisation error than that achieved by the evidence. Theories based on the V–C dimension lead to structural risk minimisation criteria [30], which seem better correlated with generalisation error. In fact, it is interesting that the form of Guyon *et al.*'s predicted generalisation error has scaling behaviour identical to that of the evidence!

More work is called for on the relationship between the evidence, cross–validation and generalisation ability to understand these results.

7.3 Having to make too much explicit

Statistical problems that are precisely enough stated for the sampling theory school can be too vague for a Bayesian [45]. When the Bayesian adds additional constraints to such a problem to make it solveable, he comes under fire for making assumptions that may be, in detail, not justified. 'Order statistics' provide an example of such a problem.

Imagine that we wish to infer the median of a density given N samples from it, without a precise specification of what type of density we are dealing with (in particular, we do not know the distribution is Gaussian). A Bayesian analysis would have to assume an explicit parameterised form for the density (for example a free form density with a maxent prior), solve for the posterior distribution of the parameters, then marginalise over that distribution to get a posterior for the median of the density. I think that in principle this is the right thing to do (and, to their credit, some Bayesians have shown that it can be done [16]), but it is an approach that involves introducing and then eliminating a large amount of irrelevant baggage (the explicit parameterised form for the density). The details of this parameterisation will probably be hard to justify, and anyway the answer that we obtain is unlikely to depend sensitively on them. It seems unfortunate to have to introduce so much explicit and arbitrary detail in order to answer a simple question. Having said this, I should make it clear that I am not advocating the orthodox sampling theory approach to order statistics; like most sampling theory procedures, order statistics are incoherent. It will be interesting if Bayesian methods can be developed which avoid having to explicitly handle detailed parameterisations that are then marginalised away again. Perhaps Monte Carlo methods like Radford Neal's [55] are a step in this direction.

7.4 An alternative interpretation of weight decay

Geoff Hinton (personal communication) has suggested an alternative view of mixture weight decay. The decay mechanism is still viewed as implementing prior knowledge, but not the literal prior that says the w_i are modelled as coming from a mixture of Gaussians. Rather, our real prior assumption is that a fraction of the weights ought to be exactly zero. Thus the true width of the component at the origin ought to be zero; it is only set to a non–zero value as a computational artifice. This view, that weight decay is intended to switch off weights, is apparently shared by other workers [39, 87].

Under this interpretation, there is no reason to suppose that the Bayesian choice of the width of this component should be appropriate.[1] (The width of the other broad compo-

[1]All the same, Nowlan and Hinton have applied the Bayesian procedure to networks predicting sunspot

nent(s) of the mixture distribution should still be inferred using Bayesian methods.) It will be interesting to see if this interpretation can be formalised, leading to an alternative well–founded procedure for setting the parameters of a zero mixture. This would also necessitate changes in the evaluation of the evidence.

7.5 Future tasks, open problems

More expensive evidence calculations, cheaper Hessian calculations

The Bayesian calculations throughout this thesis all depend on the inverse Hessian $\mathbf{A}^{-1}$, under the Gaussian approximation. There are two directions for further work. In the more expensive direction, we can ask how to make the Bayesian calculations more accurate by improving on the Gaussian approximation, by the use of Monte Carlo methods, for example. In addition, methods for integrating over regularisation constants need to be developed, rather than fixing those constants to their most probable values. These are questions that Skilling and Sibisi (personal communication) are working on.

In the cheaper direction, we can ask how to make the Hessian calculations more approximate and more efficient so as to reduce the cost of these calculations. I hope to investigate statistical methods for reducing the $O(Nk^2) + O(k^3)$ calculation of properties of $\mathbf{A}^{-1}$ to $O(k^2)$ or less time.

Noisy input variables

The tasks of interpolation and classification given noisy input variables has yet to be integrated into the evidence framework.

Missing data

Imagine that we are asked to interpolate a data set $D = \{\mathbf{x}, \mathbf{t}\}$, in which some of the elements $\mathbf{x}$ are incomplete, lacking a specification of some of their components. The interpolation framework described in this thesis cannot handle this case because the density over $\mathbf{x}$ is not modelled. It is an open problem to find a simple, well-defined way to integrate data with missing components into these interpolation models.

This problem has much in common with the task of combining unlabelled data with discriminative training. In discriminative training we adapt a classifier that models $P(\mathbf{t}|\mathbf{x}, \mathbf{w}, \mathcal{H})$ to a data set $D_1 = \{\mathbf{x}, \mathbf{t}\}$; now if additional unlabelled data $D_2 = \{\mathbf{x}\}$ is available, it is likely to provide useful information, if we assume some sort of density over $\mathbf{x}$. The reason that we are reluctant to specify such a density over $\mathbf{x}$ in speech recognition, however, is because that is where we have come from — discriminative training is used instead of full probabilistic modelling because it obtains better performance from poor models. Ideally we would like to be able to develop superior word models but what if we are stuck with a particular model space, either because of computational constraints or because of lack of creativity? Combining discriminative training with unlabelled data seems to me to be one of the current frontiers of Bayesian methods.

---

time series, and obtained better performance than any published model [58].

Other applications

The concepts of Bayesian data modelling described in this thesis have great generality and should be relevant to any experimental scientist. Example applications include the following:

Speech recognition: automated control of hidden Markov model structure
In speech recognition, selection between alternative models for a single word could be made using the evidence, and the concept of moderation (*i.e.*, incorporation of error bar information) is expected to be useful when fitting a model to utterances.

Point source image reconstruction
When estimating point sources in an astronomical image, Occam's razor is needed to avoid fitting too many stars to the image.

Neurophysiology: multi–neuron recording
The task is to infer the activities of multiple neurons in a piece of brain tissue from the signals in an array of recording electrodes. This will require development of non–parametric Bayesian methods.

Density estimation
The evidence could be evaluated for the problem of choosing the number of Gaussians in a mixture model, and the problem of choosing between Gaussian models and more 'robust' clustering models. The latter problem would also be relevant to regression problems where non–Gaussian noise models are thought appropriate; a definitive Bayesian attack on the problem of inferring a slightly non–Gaussian distribution has been made by Box and Tiao [10, 12].

Further applications in neural networks
It remains to be investigated whether these methods scale up to real, larger problems. Also this framework has yet to be applied to more sophisticated regularisers such as the mixture decay models of Hinton and Nowlan [57].

The power and unifying perspective of Bayesian methods are becoming more widely appreciated. This thesis has demonstrated their utility for adaptive models such as neural networks. There are thousands more data modelling tasks waiting for the 'evidence' to be evaluated. It will be exciting to see how much we can learn when this is done.

Bibliography

[1] Y.S. Abu-Mostafa (1990). The Vapnik–Chervonenkis dimension: information versus complexity in learning, *Neural Computation* **1** 3, 312–317.

[2] Y.S. Abu-Mostafa (1990). Learning from hints in neural networks, *J. Complexity* **6**, 192–198.

[3] H. Akaike (1970). Statistical predictor identification, *Ann. Inst. Statist. Math.* **22**, 203–217.

[4] J.R.P. Angel, P. Wizinowich, M. Lloyd-Hart, and D. Sandler (1990). Adaptive optics for array telescopes using neural–network techniques, *Nature* **348**, 221–224.

[5] E.B. Baum (1991). Neural net algorithms that learn in polynomial time from examples and queries, *IEEE Trans. on neural networks* **2** 1, 5–19.

[6] T. Bayes (1763). An essay towards solving a problem in the doctrine of chances, *Philos. Trans. R. Soc. London* **53**, 370–418, reprinted in *Biometrika* (1958) **45**, 293–315.

[7] S. Becker and Y. Le Cun (1988). Improving the convergence of back–propagation learning with second order methods, in *Proc. of the connectionist models Summer school,* Ed. D.S. Touretzky *et al.*, 29, Morgan Kaufmann.

[8] J. Berger (1985). *Statistical decision theory and Bayesian analysis,* Springer.

[9] C.M. Bishop (1992). Exact calculation of the Hessian matrix for the multilayer perceptron, *Neural Computation* **4** 4, 494–501.

[10] G.E.P. Box and G.C. Tiao (1962). A further look at robustness via Bayes' theorem, *Biometrika* **49**, 419–432.

[11] G.E.P. Box and G.C. Tiao (1964). A Bayesian approach to the importance of assumptions applied to the comparison of variances, *Biometrika* **51**, 153–167.

[12] G.E.P. Box and G.C. Tiao (1968). A Bayesian approach to some outlier problems, *Biometrika* **55**, 119–129.

[13] G.E.P. Box and G.C. Tiao (1973). *Bayesian inference in statistical analysis,* Addison–Wesley.

[14] G.L. Bretthorst (1990). Bayesian Analysis. I. Parameter Estimation Using Quadrature NMR Models. II. Signal Detection and Model Selection. III. Applications to NMR, *J. Magnetic Resonance* **88** 3, 533–595.

[15] J.S. Bridle (1989). Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition, in *Neuro–computing: algorithms, architectures and applications,* F. Fougelman–Soulie and J. Hérault, editors, Springer–Verlag.

[16] M.K. Charter (1991). Quantifying drug absorption, in [25], 245–252.

[17] R.T. Cox (1946). Probability, frequency, and reasonable expectation, *Am. J. Physics* **14**, 1–13.

[18] A.R. Davies and R.S. Anderssen (1986). Optimization in the regularization of ill–posed problems, *J. Austral. Mat. Soc. Ser. B* **28**, 114–133.

[19] J.S. Denker and Y. Le Cun (1991). Transforming neural-net output levels to probability distributions, in *Advances in neural information processing systems 3,* ed. R.P. Lippmann *et al.*, 853–859, Morgan Kaufmann.

[20] R. Duda and P. Hart (1973). *Pattern Classification and Scene Analysis,* Wiley.

[21] M.A. El–Gamal (1991). The role of priors in active Bayesian learning in the sequential statistical decision framework, in [25], 33–38.

[22] R.L. Eubank (1988). *Spline smoothing and non–parametric regression,* Marcel Dekker.

[23] V.V. Fedorov (1972). *Theory of optimal experiments,* Academic press.

[24] K. Fukunaga (1972). *Introduction to statistical pattern recognition,* Academic press.

[25] W.T. Grandy, Jr. and L.H. Schick, eds. (1991). *Maximum Entropy and Bayesian Methods, Laramie, Wyoming, 1990,* Kluwer.

[26] S.F. Gull (1988). Bayesian inductive inference and maximum entropy, in *Maximum Entropy and Bayesian Methods in science and engineering, vol. 1: Foundations*, G.J. Erickson and C.R. Smith, eds., Kluwer.

[27] S.F. Gull (1989). Developments in Maximum entropy data analysis, in [71], 53–71.

[28] S.F. Gull (1989). Bayesian data analysis: straight–line fitting, in [71], 511–518.

[29] S.F. Gull and J. Skilling (1991). *Quantified Maximum Entropy.* `MemSys5` *User's manual,* M.E.D.C., 33 North End, Royston, SG8 6NR, England.

[30] I. Guyon, V.N. Vapnik, B.E. Boser, L.Y. Bottou and S.A. Solla (1992). Structural risk minimization for character recognition, in *Advances in neural information processing systems 4,* ed. J.E. Moody, S.J. Hanson and R.P. Lippmann, Morgan Kaufmann.

[31] R. Hanson, J. Stutz and P. Cheeseman (1991). Bayesian classification theory, *NASA Ames* TR FIA–90-12-7-01.

[32] D. Haussler, M. Kearns and R. Schapire (1991). Bounds on the sample complexity of Bayesian learning using information theory and the V–C dimension, preprint.

[33] G.E. Hinton and T.J. Sejnowski (1986). Learning and relearning in Boltzmann machines, in *Parallel Distributed Processing,* Rumelhart *et al.*, MIT Press.

[34] J.J. Hopfield (1987). Learning algorithms and probability distributions in feed–forward and feed–back networks, *Proc. Natl. Acad. Sci. USA* **84**, 8429–33.

[35] J-N. Hwang, J.J. Choi, S. Oh, and R.J. Marks II (1991). Query–based learning applied to partially trained multilayer perceptrons, *IEEE Trans. on neural networks* **2** 1, 131–136.

[36] E.T. Jaynes (1986). Bayesian methods: general background, in *Maximum Entropy and Bayesian Methods in applied statistics,* ed. J.H. Justice, C.U.P..

[37] W.H. Jefferys and J.O. Berger (1992). Ockham's razor and Bayesian analysis, *American Scientist* **80**, 64–72.

[38] H. Jeffreys (1939). *Theory of Probability,* Oxford Univ. Press.

[39] C. Ji, R.R. Snapp and D. Psaltis (1990). Generalizing smoothness constraints from discrete samples, *Neural Computation* **2** 2, 188-197.

[40] R.L. Kashyap (1977). A Bayesian comparison of different classes of dynamic models using empirical data, *IEEE Transactions on Automatic Control* **AC-22** 5, 715–727.

[41] Y. Le Cun, J.S. Denker and S.S. Solla (1990). Optimal Brain Damage, in *Advances in neural information processing systems 2,* ed. David S. Touretzky, 598–605, Morgan Kaufmann.

[42] W.T. Lee and M.F. Tenorio (1991). On Optimal Adaptive Classifier Design Criterion — How many hidden units are necessary for an optimal neural network classifier?, *Purdue University* TR-EE-91-5.

[43] E. Levin, N. Tishby and S. Solla (1989). A statistical approach to learning and generalization in layered neural networks, *COLT '89: 2nd workshop on computational learning theory,* 245–260.

[44] D.V. Lindley (1956). On a measure of the information provided by an experiment, *Ann. Math. Statist.* **27**, 986–1005.

[45] D.V. Lindley (1970). Bayesian analysis in regression problems, in *Bayesian statistics,* D.L. Meyer and R.O. Collier, eds., Peacock publishers.

[46] D.V. Lindley (1972). *Bayesian statistics, a review,* Society for Industrial and Applied Mathematics, Philadelphia.

[47] T.J. Loredo (1989). From Laplace to supernova SN 1987A: Bayesian inference in astrophysics, in *Maximum Entropy and Bayesian Methods,* ed. P. Fougere, Kluwer.

[48] S.P.Luttrell (1985). The use of transinformation in the design of data sampling schemes for inverse problems, *Inverse Problems* **1**, 199–218.

[49] D.J.C. MacKay (1991). Bayesian interpolation, *Neural Computation* **4** 3 415–447; Chapter 2 of this dissertation.

[50] D.J.C. MacKay (1991). A practical Bayesian framework for backprop networks, *Neural Computation* **4** 3 448–472; Chapter 3 of this dissertation.

[51] D.J.C. MacKay (1991). Information–based objective functions for active data selection, *Neural Computation* **4** 4 589–603; Chapter 4 of this dissertation.

[52] D.J.C. MacKay (1991). The evidence framework applied to classification networks, *Neural Computation* **4** 5 698–714; Chapter 5 of this dissertation.

[53] K.E. Mark and M.I. Miller (1992). Bayesian model selection and minimum description length estimation of auditory–nerve discharge rates, *J. Acoust. Soc. Am.* **91** 2, 989–1002.

[54] J.E. Moody (1991). Note on generalization, regularization and architecture selection in nonlinear learning systems, *First IEEE–SP Workshop on neural networks for signal processing.*IEEE Computer society press

[55] R.M. Neal (1991). Bayesian mixture modelling by Monte Carlo simulation, *Technical Report CRG–TR–91–2* Dept. of Computer Science, University of Toronto.

[56] R.M. Neal (1992). Bayesian training of backpropagation networks by the hybrid Monte Carlo method, *Technical Report CRG–TR–92–1* Dept. of Computer Science, University of Toronto.

[57] S.J. Nowlan (1991). Soft competitive adaptation: neural network learning algorithms based on fitting statistical mixtures, *Carnegie Mellon University* Doctoral thesis CS–91–126.

[58] S.J. Nowlan and G.E. Hinton (1991). Soft weight sharing, preprint.

[59] D.B. Osteyee and I.J. Good (1974). *Information, weight of evidence, the singularity between probability measures and signal detection,* Springer.

[60] J.D. Patrick and C.S. Wallace (1982). Stone circle geometries: an information theory approach, in *Archaeoastronomy in the Old World,* D.C. Heggie, editor, Cambridge Univ. Press.

[61] F.J. Pineda (1989). Recurrent back–propagation and the dynamical approach to adaptive neural computation, *Neural Computation* **1**, 161–172.

[62] M. Plutowski and H. White (1991). Active selection of training examples for network learning in noiseless environments, *Dept. Computer Science, UCSD* TR 90-011.

[63] T. Poggio, V. Torre and C. Koch (1985). Computational vision and regularization theory, *Nature* **317** 6035, 314–319.

[64] W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling (1988). *Numerical Recipes in C,* Cambridge.

[65] J. Rissanen (1978). Modeling by shortest data description, *Automatica* **14**, 465–471.

[66] D.E. Rumelhart, G.E. Hinton and R.J. Williams (1986). Learning representations by back–propagating errors, *Nature* **323**, 533–536.

[67] D.E. Rumelhart (1987). Cited in [39].

[68] G. Schwarz (1978). Estimating the dimension of a model, *Ann. Stat.* **6** 2, 461–464.

[69] H.S. Seung, H. Sompolinsky and N. Tishby (1991). Statistical mechanics of learning from examples, preprint.

[70] S. Sibisi (1991). Bayesian interpolation, in [25], 349–355.

[71] J. Skilling, editor (1989). *Maximum Entropy and Bayesian Methods, Cambridge 1988,* Kluwer.

[72] J. Skilling (1989). The eigenvalues of mega–dimensional matrices, in [71], 455–466.

[73] J. Skilling (1991). On parameter estimation and quantified MaxEnt, in [25], 267–273.

[74] J. Skilling, D.R.T. Robinson, and S.F. Gull (1991). Probabilistic displays, in [25], 365–368.

[75] J. Skilling (1991). Fundamentals of MaxEnt in data analysis, in *Maximum Entropy in action*, B. Buck and V.A. MacAulay, eds., Oxford, 19–40..

[76] J. Skilling (1992). Bayesian solution of ordinary differential equations, in *Maximum Entropy and Bayesian Methods, Seattle 1991,* G.J. Erickson and C.R. Smith, eds., Kluwer.

[77] A.F.M. Smith and D.J. Spiegelhalter (1980). Bayes factors and choice criteria for linear models, *Journal of the Royal Statistical Society B* **42** 2, 213–220.

[78] S.A. Solla, E. Levin and M. Fleisher (1988). Accelerated learning in layered neural networks, *Complex systems* **2**, 625–640.

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

[80] S.M. Stigler (1986). Laplace's 1774 memoir on inverse probability, *Stat. Sci.* **1** 3, 359–378.

[81] R. Szeliski (1989). *Bayesian modeling of uncertainty in low level vision,* Kluwer.

[82] N. Tishby, E. Levin and S.A. Solla (1989). Consistent inference of probabilities in layered networks: predictions and generalization, *Proc. IJCNN, Washington.*

[83] D. Titterington (1985). Common structure of smoothing techniques in statistics, *Int. Statist. Rev.* **53**, 141–170.

[84] A.M. Walker (1967). On the asymptotic behaviour of posterior distributions, *J. R. Stat. Soc. B* **31**, 80–88.

[85] C. S. Wallace and D. M. Boulton (1968). An information measure for classification, *Comput. J.* **11** 2, 185–194.

[86] C. S. Wallace and P. R. Freeman (1987). Estimation and Inference by Compact Coding, *J. R. Statist. Soc. B* **49** 3, 240-265.

[87] A.S. Weigend, D.E. Rumelhart and B.A. Huberman (1991). Generalization by weight–elimination with applications to forecasting, in *Advances in neural information processing systems 3,* ed. R.P. Lippmann *et al.*, 875–882, Morgan Kaufmann.

[88] N. Weir (1991). Applications of maximum entropy techniques to HST data, in Proceedings of the ESO/ST–ECF Data Analysis Workshop, April 1991.

[89] A. Zellner (1984). *Basic issues in econometrics,* Chicago.