In Advances in Independent Component Analysis,
ed. by Mark Girolami (Springer, 2000). In press.

Ensemble Learning

Harri Lappalainen[1] and James W. Miskin[2]

[1] Helsinki University of Technology, Neural Networks Research Centre, P.O. Box 5400, FIN-02015 HUT, Finland
[2] Cavendish Laboratory, Cambridge CB3 0HE, UK

Summary. This chapter gives a tutorial introduction to Ensemble Learning, a recently developed Bayesian method. For many problems it is intractable to perform inferences using the true posterior density over the unknown variables. Ensemble Learning allows the true posterior to be approximated by a simpler approximate distribution for which the required inferences are tractable.

1 Introduction

When we say we are making a model of a system, we are setting up a tool which can be used to make inferences, predictions and decisions. Each model can be seen as a hypothesis, or explanation, which makes assertions about the quantities which are directly observable and which can only be inferred from their effect on observable quantities.

In the Bayesian framework, knowledge is contained in the conditional probability distributions of the models. We can use Bayes' theorem to evaluate the conditional probability distributions for the unknown parameters, y, given the set of observed quantities, x, using

$$p(y|x) = \frac{p(x|y)\,p(y)}{p(x)} \tag{1}$$

The prior distribution $p(y)$ contains our knowledge of the unknown variables before we make any observations. The posterior distribution $p(y|x)$ contains our knowledge of the system after we have made our observations. The likelihood, $p(x|y)$, is the probability that the observed data will be observed given a specific set of values for the unknown parameters.

There is no clear cut difference between the prior and posterior distributions, since after a set of observations the posterior distribution becomes the prior for another set of observations.

In order to make inferences based on our knowledge of the system, we need to marginalise our posterior distribution with respect to the unknown parameters of the model. For instance in order to obtain the average values of the unknown parameters we would need to perform the expectation

$$\bar{y} = \int y p(y|x)\,dy \tag{2}$$

Alternatively we may be trying to use our model of the system to make decisions about which action to take. In this case we would like to choose the action which maximises some utility function. In this case the expected utility is found by marginalising the utility function over the posterior density of the models. An example would be hypothesis testing where we have a number of explanations for the cause of a set of observed data. By having a different model for each hypothesis, we could choose the model that maximises the expected utility.

In this chapter we shall motivate the idea of considering the posterior distribution to be the result of any experiment instead of just considering a point in the model space. Section 3 will discuss different methods of approximating the posterior when it is intractable to make inferences based on the true posterior. Section 4 will introduce the idea of using Ensemble Learning to approximate the true posterior by a simpler separable distribution. Section 5 will discuss the construction of probabilistic models, both in supervised and unsupervised learning. Section 6 will give examples of using ensemble learning in both a fixed form and free form approximation.

2 Posterior averages in action

Probability theory tells us that the optimal generalisation is the one resulting from a Bayesian approach. Overfitting to the data means that we are making conclusions that the data does not support. Alternatively underfitting means that our conclusions are too diffuse.

Overfitting is an artifact resulting from choosing only one explanation (model) for the observations. Figure 1 shows a hypothetical posterior distribution. If the model is chosen to maximise the posterior probability then the model will be chosen to be in the narrow peak. The problem is that the peak only contains a fraction of the total probability mass. This means that the model will explain the observations very well, but will be very sensitive to the values of the parameters and so may not explain further observations. When making predictions and decisions it is the position of the probability mass that counts and not the position of a maximum.

In order to solve these problems it is necessary to average over all possible hypotheses. That is we should average over all possible models, but weight them by the posterior distribution that we obtain from our observations.

It is important to note that averaging does not mean computing the average of parameter values and then using that as the best guess. For instance in digit recognition the posterior distribution may have a peak in the model for "1"s and in the model for "9"s. Averaging over the posterior does not mean that you compute the average digit as a "5" and then use that as the best guess, it means that you should prepare for the possibility that the digit can be either "1" or "9".

Figure 2 shows a schematic example of fitting a Multi-Layer Perceptron (MLP) to a set of data. The data is fairly smooth and has only a little noise, but there is a region of missing data. The best regularised MLP is able to fit the data quite well in those areas where there are data points and so the noise level will be estimated to be low. Therefore the MLP will give tight error bars even in the region where there is no data to support the fit. This is overfitting because the conclusion are more specific than the data supports. If we were to later obtain some data in the missing region, it is plausible that it would lie significantly outside the MLP fit and

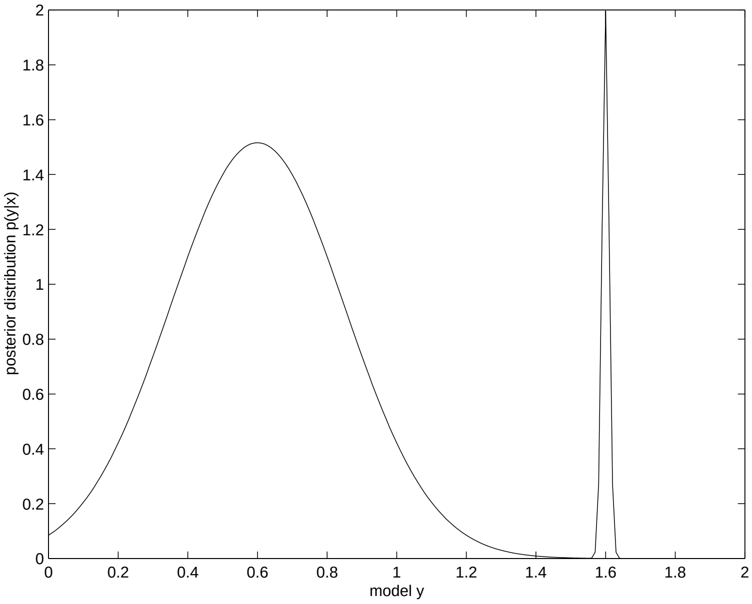


Fig. 1. Schematic illustration of overfitting to the data. If the true posterior over the models, y, is of this form then choosing the model that maximises the posterior probability will mean choosing the model in the sharp peak. But near the maximum the posterior density is sharp and so highly dependent on the model parameters. Therefore the model will explain the observed data very well, but may not generalise to further observations.

so the posterior probability of the chosen model could rapidly drop as more data is obtained.

Instead of chosing a specific MLP it is better to average over several MLPs. In this case we would find that there are multiple explanations for the missing data. Therefore the average fit will give tight error bars where there is data and broad error bars in the regions where there is no data.

If the estimate for the level of noise in the data were estimated to be too high then the error bars for the areas with missing data might be good but the error bars for the areas with data would too broad and we would suffer from underfitting. Alternatively if the noise level is estimated to be too low, the error bars will be too narrow where there is data. For intermediate noise levels we could suffer from overfitting in some regions and underfitting in others.

In order to fix this we could allow the noise level to be part of our model and so the noise level could vary. The problems with overfitting would now be changed to overfitting the noise model. The solution is to average the noise under the posterior distribution which would include the MLP parameters and the noise.

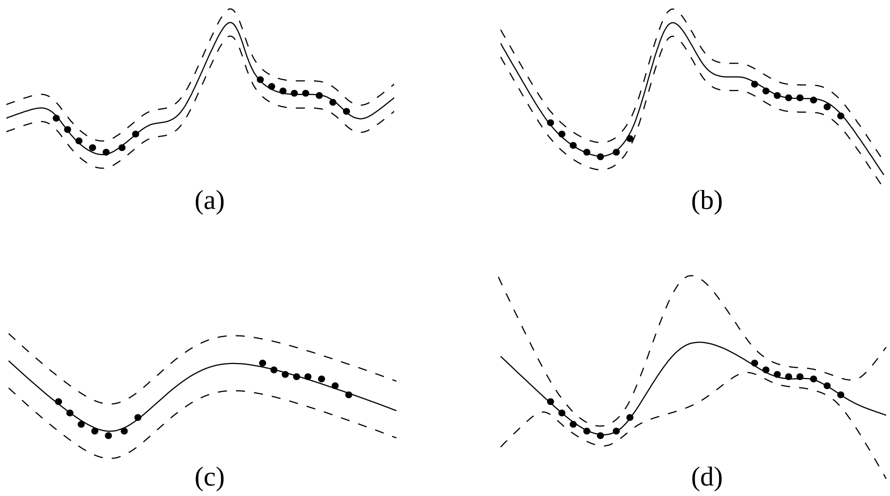


Fig. 2. A schematic illustration of averaging over posterior of models. The data points are denoted by circles and the black lines show the MLP fit. The dashed lines show the error bars the model gives. The models in plots a and b assume fairly small noise. The model in plot c assumes larger noise. The average of the models over the posterior pdf of the models gives larger error bars in the areas where there are no observations.

3 Approximations of posterior pdf

As we have shown earlier, the posterior distribution is a synonym for our available knowledge of a system after we have made a set of observations. In order to use this knowledge we will typically be required to marginalise the posterior or to evaluate expectations of functions under the posterior distribution. In many problems it is intractable to perform the necessary integrals.

If we look again at Bayes equation we have

$$p(y|x) = \frac{p(x|y)\,p(y)}{p(x)} \tag{3}$$

It is easy to compute one point in the joint density $p(x, y)$ (the numerator of the posterior distribution) but in general evaluating the denominator, $p(x)$, is difficult. Similarly marginalising the posterior distribution is difficult.

Therefore it is necessary to approximate the posterior density by a more tractable form for which it is possible to perform any necessary integrals. We cannot take a point estimate (such as the MAP estimate) because this leads to overfitting as shown earlier. This is because the MAP estimate does not guarantee a high probability mass in the peak of the posterior distribution and so the posterior distribution may be sharp around the MAP estimate. We would like to approximate the probability mass of the posterior.

There are in general two types of approximation that retain the probability mass of the true posterior distribution. The first type is the stochastic approximation and the second type is the parametric approximation. In a stochastic approximation (such as Markov-chain Monte Carlo method) the aim is to perform the integrations by drawing samples from the true posterior distribution, [3]. The average of any function is then found by finding the average value of the function given all of the samples from the posterior.

In a parametric approximation (such as Laplace approximation, [6]) the posterior distribution is approximated by an alternative function (such as a Gaussian) such that it is much simpler to perform any necessary approximations.

The problem with the stochastic methods is that when performing a stochastic approximation it is necessary to wait until the sampler has sampled from all of the mass of the posterior distribution. Therefore testing for convergence can be a problem. The problem with the parametric approach is that the integrals that are being performed are not exactly the same as those that would be performed when the true posterior is used. Therefore while the stochastic approximation has to give the correct answer (eventually) the parametric approxmation will give an approximate answer soon (it is the 'quick and dirty' method).

Model selection can be seen as a special form of approximating the posterior distribution. The posterior distribution could contain many peaks, but when there is lots of data, most of the probability mass is typically contained in a few peaks of the posterior distribution. Model selection means using only the most massive peaks and discarding the remaining models.

4 Ensemble learning

Ensemble learning, [1], is a recently introduced method for parametric approximation of the posterior distributions where Kullback-Leibler information, [2], [5], is used to measure the misfit between the actual posterior distribution and its approximation. Let us denote the observed variables by x and the unknown variables by y. The true posterior distribution $p(y|x)$ is approximated with the distribution $q(y|x)$ by minimising the Kullback-Leibler information.

$$\begin{aligned} D(q(y|x)||p(y|x)) &= \int q(y|x) \ln \frac{q(y|x)}{p(y|x)} dy \\ &= \int q(y|x) \ln \frac{p(x)q(y|x)}{p(x,y)} dy \\ &= \int q(y|x) \ln \frac{q(y|x)}{p(x,y)} dy + \ln p(x) \end{aligned} \tag{4}$$

The Kullback-Leibler information is greater than or equal to zero, with equality if and only if the two distributions, $p(y|x)$ and $q(y|x)$, are equivalent. Therefore the Kullback-Leibler information acts as a distance measure between the two distributions.

If we note that the term $p(x)$ is a constant over all the models, we can define a cost function $C_y(x)$ which we are required to minimise to obtain the optimum approximating distribution

$$C_y(x) = D(q(y|x)||p(y|x)) - \ln p(x) = \int q(y|x) \ln \frac{q(y|x)}{p(x,y)} dy \tag{5}$$

We shall adopt the notation that the subindex of C denotes the variables that are marginalised over in the cost function. In general, they are the unknown variables of the model. The notation also makes explicit that the cost function $C_y(x)$ gives an upper bound for $-\ln p(x)$. Here we use the same notation as with probability distributions, that is $C_y(x|z)$ means

$$\begin{aligned} C_y(x|z) &= D(q(y|x,z)||p(y|x,z)) - \ln p(x|z) \\ &= \int q(y|x,z) \ln \frac{q(y|x,z)}{p(x,y|z)} dy \end{aligned} \tag{6}$$

and thus yields the upper bound for $-\ln p(x|z)$.

Ensemble learning is practical if the terms $p(x,y)$ and $q(y|x)$ of the cost function $C_y(x)$ can be factorised into simple terms. If this is the case, the logarithms in the cost function split into sums of many simple terms. By virtue of the definition of the models, the likelihood and priors are both likely to be products of simpler distributions, $p(x,y)$ typically factorises into simple terms. In order to simplify the approximating ensemble, q is also modelled as a product of simple terms.

The Kullback-Leibler information is a global measure, providing that the approximating distribution is a global distribution. Therefore the measure will be sensitive to probability mass in the true posterior distribution rather than the absolute value of the distribution itself.

Training the approximating ensemble can be done by assuming a fixed parametric form for the ensemble (for instance assuming a product of Gaussians). The parameters of the distributions can then be set to minimise the cost function.

An alternative method is to only assume a separable form for the approximating ensemble. The distributions themselves can then be found by performing a functional minimisation of the cost function with respect to each distribution in the ensemble. While this method must always give ensembles with equivalent or lower misfits than those obtained by a assuming a parametric form, the distributions that are obtained are not always tractable and so the fixed form method may be more useful.

4.1 Model selection in ensemble learning

Recall that the cost function $C_y(x|H)$ can be translated into lower bound for $p(x|H)$. Since $p(H|x) = p(x|H)p(H)/p(x)$, it is natural that $C_y(x|H)$ can be used for model selection also by equating

$$p(H|x) \approx \frac{e^{-C_y(x|H)}P(H)}{\sum_{H'} e^{-C_y(x|H')}P(H')} \,. \tag{7}$$

In fact, we can show that the above equation gives the best approximation for $p(H|x)$ in terms of $C_{y,H}(x)$, the Kullback-Leibler divergence between $q(y,H|x)$ and $p(y,H|x)$, which means that the model selection can be done using the same principle of approximating the posterior distribution as learning parameters.

Without losing any generality from $q(y,H|x)$, we can write

$$q(y,H|x) = Q(H|x)q(y|x,H)\,. \tag{8}$$

Now the cost function can be written as

$$\begin{aligned} C_{y,H}(x) &= \sum_H \int q(y,H|x) \ln \frac{q(y,H|x)}{p(x,y,H)} dy \\ &= \sum_H Q(H|x) \int q(y|x,H) \ln \frac{Q(H|x)q(y|x,H)}{P(H)p(x,y|H)} dy \\ &= \sum_H Q(H|x) \left[\ln \frac{Q(H|x)}{P(H)} + C_y(x|H) \right]. \end{aligned} \tag{9}$$

Minimising $C_{y,H}(x)$ with respect to $Q(H|x)$ under the constraint

$$\sum_H Q(H|x) = 1 \tag{10}$$

yields

$$Q(H|x) = \frac{e^{-C_y(x|H)}P(H)}{\sum_{H'} e^{-C_y(x|H')}P(H')}\,. \tag{11}$$

Substituting this into equation 9 yields the minimum value for $C_{y,H}(x)$ which is

$$C_{y,H}(x) = -\ln \sum_H e^{-C_y(x|H)}P(H) \tag{12}$$

If we wish to use only a part of different model structures H, we can try to find those H which would minimise $C_{y,H}(x)$. It is easy to see that this is accomplished by choosing the models corresponding to $C_y(x|H)$. A special case is to use only one H corresponding to the smallest $C_y(x|H)$.

4.2 Connection to coding

The cost function can be derived within MDL framework. The intuitive idea behind MDL is that in order to be able to encode data compactly one has to find a good model for it. Coding is not actually done; only the formula for the code length is needed. The optimal code length of x is $L(x) = -\log P(x)$ bits, and therefore coding has a close connection to the probabilistic framework. Some people prefer to think in terms of coding lengths, some in term of probabilities.

In coding, the sender and the receiver have to agree on the structure of the message in advance. This corresponds to having to state one's priors in the Bayesian framework.

A simple example: let x be the observation to be coded and y be a real valued parameter. We first encode y with accuracy dy. This part of the message takes $L(y) = -\log p(y)dy$ bits. Then we encode x with accuracy dx using the value y. This takes $L(x|y) = -\log p(x|y)dx$ bits. The accuracy of the encoding of the observations can be agreed on in advance, but there is a question of how to decide the accuracy dy. The second part of the code would be shortest if y would be exactly in the maximum likelihood solution. If y is encoded in too high accuracy, however, the first part of the code will be very long. If y is encoded in too low accuracy, the first part will be short but deviations from the optimal y will increase the second part of the code.

The bits-back argument, [1], overcomes the problem by using infinitesimally small dy but picking the y from a distribution $q(y)$ and encoding a secondary message in the choice of y. Since the distribution $q(y)$ is not needed for decoding x, both the sender and the receiver can run the same algorithm for determining $q(y)$ from x. After finding out $q(y)$, the receiver can decode the secondary message which can have $-\log q(y)dy$ bits. As these bits will be got back in the end, the expected message length is

$$\begin{aligned} L(x) &= E_q\{-\log p(x|y)dx - \log p(y)dy + \log q(y)dy\} \\ &= D(q(y)||p(y|x)) - \log p(x)dx = C_y(x) - \log dx\,. \end{aligned} \tag{13}$$

The amount of bits used for individual parameter θ can be measured by looking at $D(q(\theta)||p(\theta))$

It is interesting to notice that although the message has two parts, y and x, if $q(y)$ is chosen to be $p(y|x)$, the expected code length is equal to that of the optimal one-part message where the sender encodes x directly using the marginal probability $p(x) = \int p(x|y)p(y)dy$.

4.3 EM and MAP

Expectation maximisation (EM) algorithm can be seen as a special case of ensemble learning. The set-up in EM is the following: Suppose we have a probability model $p(x, y|\theta)$. We observe x but y remains hidden. We would like to estimate θ with maximum likelihood, i.e., maximise $p(x|\theta)$ w.r.t. θ, but suppose the structure of the model is such that integration over $p(x, y|\theta)$ is difficult, i.e., it is difficult to evaluate $p(x|\theta) = \int p(x, y|\theta)dy$.

What we do is take the cost function $C_y(x|\theta)$ and minimise it alternately with respect to θ and $q(y|x, \theta)$. The ordinary EM algorithm will result when $q(y|x, \theta)$ has a free form in which case $q(y|x, \theta)$ will be updated to be $p(y|x, \hat{\theta})$, where $\hat{\theta}$ is the current estimate of θ. The method is useful if integration over $\ln p(x, y|\theta)$ is easy, which is often the case. This interpretation of EM was given by [4].

EM algorithm can suffer from overfitting because only point estimates for the parameters θ are used. Even worse is to use maximum a posterior (MAP) estimator where one finds the θ and y which maximise $p(y, \theta|x)$. Unlike maximum likelihood estimation, MAP estimation is not invariant under reparametrisations of the model. This is because MAP estimation is sensitive to probability density which changes nonuniformly if the parameter space is changed nonlinearly.[1]

MAP estimation can be interpreted in ensemble learning framework as minimising $C_{y,\theta}(x)$ and using delta-distribution as $q(y, \theta|x)$. This makes the integral $\int q(y, \theta|x) \ln q(y, \theta|x) dy d\theta$ infinite. It can be neglected when estimating θ and y because it is constant with respect to $\hat{y}$ and $\hat{\theta}$, but the infinity of the cost function shows that delta distribution, i.e. a point estimator, is a bad approximation for a posterior density.

5 Construction of probabilistic models

In order to apply the Bayesian approach for modelling, the model needs to be given in probabilistic terms, which means stating the joint distribution of all the variables in the model. In principle, any joint distribution can be regarded as a model, but in practice, the joint distribution will have a simple form.

As an example, we shall see how a generative model turns into a probabilistic model. Suppose we have a model which tells how a sequence $\boldsymbol{y} = y(1), \ldots, y(t)$ transforms into sequence $\boldsymbol{x} = x(1), \ldots, x(t)$.

$$x(t) = f(y(t), \theta) + n(t) \tag{14}$$

This is called a generative model for $\boldsymbol{x}$ because it tells explicitly how the squence $\boldsymbol{x}$ is generated from the sequence $\boldsymbol{y}$ through a mapping f parametrised by θ. As it is usually unrealistic to assume that it would be possible to model all the things affecting $\boldsymbol{x}$ exactly, the models typically include a noise term $n(t)$.

If $y(t)$ and θ are given, then $x(t)$ has the same distribution as $n(t)$ except that it is offset by $f(y(t), \theta)$. This means that if $n(t)$ is Gaussian noise with variance σ^2, equation 14 translates into

$$x(t) \sim N(f(y(t)), \sigma^2) \tag{15}$$

which is equivalent to

$$p(x(t)|y(t), \theta, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{[x(t) - f(y(t), \theta)]^2}{2\sigma^2}} . \tag{16}$$

[1] MAP estimation can be made invariant under reparametrisations by fixing the parametrisation. A natural way to do this is to use a parametrisation which makes the Fisher information matrix of the model constant. Then it is probable that the peaks in the posterior have approximately similar widths and are more or less symmetrical.

The joint density of all the variables can then be written as

$$p(\boldsymbol{x}, \boldsymbol{y}, \theta, \sigma) = p(\boldsymbol{y}, \theta, \sigma) \prod_t p(x(t)|y(t), \theta, \sigma) . \tag{17}$$

Usually also the probability $p(\boldsymbol{y}, \theta, \sigma)$ is stated in a factorisable form making the full joint probability density $p(\boldsymbol{x}, \boldsymbol{y}, \theta, \sigma)$ a product of many simple terms.

In supervised learning, the sequence $\boldsymbol{y}$ is assumed to be fully known, also for any future data, which means that the full joint probability $p(\boldsymbol{x}, \boldsymbol{y}, \theta, \sigma)$ is not needed, only

$$p(\boldsymbol{x}, \theta, \sigma|\boldsymbol{y}) = p(\boldsymbol{x}|\boldsymbol{y}, \theta, \sigma) p(\theta, \sigma|\boldsymbol{y}) . \tag{18}$$

Typically $\boldsymbol{y}$ is assumed to be independent of θ and σ, i.e., $p(\boldsymbol{y}, \theta, \sigma) = p(\boldsymbol{y})p(\theta, \sigma)$. This also means that $p(\theta, \sigma|\boldsymbol{y}) = p(\theta, \sigma)$ and thus only $p(\boldsymbol{x}|\boldsymbol{y}, \theta, \sigma)$, given by the generative model equation 14, and the prior for the parameters $p(\theta, \sigma)$ is needed in supervised learning.

If the probability $p(\boldsymbol{y})$ is not modelled in supervised learning, it is impossible to treat missing elements of the sequence $\boldsymbol{y}$. If the probability $p(\boldsymbol{y})$ is modelled, however, there are no problems. The posterior density is computed for all unknown variables, including the missing elements of $\boldsymbol{y}$. In fact, unsupervised learning can be seen as a special case where the whole sequence $\boldsymbol{y}$ is unknown. In probabilistic framework, the treatment of any missing values is possible as long as the model defines the joint density of all the variables in the model. It is, for instance, easy to treat missing elements of sequence $\boldsymbol{x}$ or mix freely between supervised and unsupervised learning depending on how large part of the sequence $\boldsymbol{y}$ is known.

5.1 Priors and hyperpriors

In the above model, the parameters θ and σ need to be assigned a prior probability and in a more general situation typically also a prior probability for the model structure would also be needed.

In general, prior probabilities for variables should reflect the belief one has about the variables. As people may have difficulties in articulating these beliefs explicitly, some rules of thumb have been developed.

A lot of research has been conducted on uninformative priors. The term means that in some sense the prior gives as little information as possible about the value of the parameter, and as such is a good reference prior although with complex models it is usually impractical to compute the exact form of the uninformative prior.

Roughly speaking, uninformative prior can be defined by saying that in a parametrisation where moving a given distance in parameter space always corresponds the similar change in the probability distribution the model defines, the parameters are uniformly distributed.

A simple example is provided by a Gaussian distribution parametrised by mean μ and variance σ^2. Doubling the variance always results in a qualitatively similar change in the distribution. Similarly, taking a step of size σ in the mean always corresponds to a similar change in the distribution. Reparametrisation by $\mu' = \mu/\sigma$ and $v = \ln \sigma$ will give a parametrisation where equal changes in parameters correspond to equal changes in distribution. A uniform prior on μ' and v would correspond to the prior $p(\mu, \sigma) \propto 1/\sigma^2$ in the original parameter space. If there is additional knowledge that μ sould be independent of σ, then μ and v give the needed parameters and the prior is $p(\mu) \propto 1$ and $p(\sigma) \propto 1/\sigma$.

None of the above uninformative priors can actually be used because they are improper, meaning that they are not normalisable. This can easily be seen by considering a uniform distribution between $\pm\infty$. These priors are nevertheless good references and hint useful parametrisations for models.

Often it is possible to utilise the fact that the model has a set of parameters which have a similar role. It is, for instance, reasonable to assume that all biases in an MLP network have a similar distribution. This knowledge can be utilised by modelling the distribution by a common parametrised distribution. Then the prior needs to be determined to these common parameters, called hyperparameters but as they control the distribution of a set of parameters, there should be less hyperparameters than parameters. The process can be iterated until the structural knowledge has been used. In the end there are usually only a few priors to determine and since there is typically a lot of data, these priors are usually not significant for the learining process.

For some it may be helpful to think about the prior in terms of coding. By using the formula $L(x) = -\ln p(x)$, any probabilites can be translated into encoding. In coding terms, the prior means the aspects of the encoding which the sender and the receiver have agreed upon prior to the transmission of data.

6 Examples

6.1 Fixed form Q

Let us model a set of observations, $\boldsymbol{x} = x(1), \dots, x(t)$, by a Gaussian distribution parametrised by mean m and log-std $v = \ln \sigma$. We shall approximate the posterior distribution by $q(m, v) = q(m)q(v)$, where both $q(m)$ and $q(v)$ are Gaussian. The parametrisation with log-std is chosen because the posterior of v is closer to Gaussian than the posterior of σ or σ^2 would be. (Notice that the parametrisation yielding close to Gaussian posterior distributions is connected to uninformative priors discussed in section 5.1.)

Let the priors for m and v be Gaussian with means μ_m and μ_v and variances σ_m^2 and σ_v^2, respectively. The joint density of the observations and the parameters m and v is

$$p(\boldsymbol{x}, m, v) = \left[\prod_t p(x(t)|m, v) \right] p(m)p(v) \,. \tag{19}$$

As we can see, the posterior is a product of many simple terms.

Let us denote by $\bar{m}$ and $\tilde{m}$ the posterior mean and variance of m.

$$q(m; \bar{m}, \tilde{m}) = \frac{1}{\sqrt{2\pi\tilde{m}}} e^{-\frac{(m-\bar{m})^2}{2\tilde{m}}} \tag{20}$$

The distribution $q(v; \bar{v}, \tilde{v})$ is analogous.

The cost function is now

$$\begin{aligned} C_{m,v}(\boldsymbol{x}) = \int q(m,v) \ln \frac{q(m,v)}{p(\boldsymbol{x},m,v)} dmdv = \\ \int q(m,v) \ln q(m) dmdv + \int q(m,v) \ln q(v) dmdv + \\ -\sum_{t=1}^{N} \int q(m,v) \ln p(x(t)|m,v) dmdv + \\ -\int q(m,v) \ln p(m) dmdv - \int q(m,v) \ln p(v) dmdv \,. \end{aligned} \tag{21}$$

We see that the cost function has many terms, all of which are expectations over $q(m,v)$. Since the approximation $q(m,v)$ is assumed to be factorised into $q(m,v) = q(m)q(v)$, it is fairly easy to compute these expectations. For instance, integrating the term $\int q(m,v) \ln q(m) dmdv$ over v yields $\int q(m) \ln q(m) dm$, since $\ln q(m)$ does not depend on v and. Since $q(m)$ is assumed to be Gaussian with mean $\bar{m}$ and variance $\tilde{m}$, this integral is, in fact, minus the entropy of a Gaussian distribution and we have

$$\int q(m) \ln q(m) dm = -\frac{1}{2}(1 + \ln 2\pi\tilde{m}) \,. \tag{22}$$

A similar term, with $\tilde{m}$ replaced by $\tilde{v}$, comes from $\int q(m,v) \ln q(v) dmdv$.

The terms where expectation is taken over $-\ln p(m)$ and $-\ln p(v)$ are also simple since

$$-\ln p(m) = \frac{1}{2} \ln 2\pi\sigma_m^2 + \frac{(m-\mu_m)^2}{2\sigma_2} \,, \tag{23}$$

which means that we only need to be able to compute the expectation of $(m-\mu_m)^2$ over the Gaussian $q(m)$ having mean $\bar{m}$ and variance $\tilde{m}$. This yields

$$\begin{aligned} E\{(m-\mu_m)^2\} = E\{m^2\} - 2E\{m\mu_m\} + E\{\mu_m^2\} = \\ \bar{m}^2 + \tilde{m} - 2\bar{m}\mu_m + \mu_m^2 = (\bar{m}-\mu_m)^2 + \tilde{m} \end{aligned} \tag{24}$$

since the variance can be defined by $\tilde{m} = E\{m^2\} - E\{m\}^2 = E\{m^2\} - \bar{m}^2$ which shows that $E\{m^2\} = \bar{m}^2 + \tilde{m}$. Integrating the equation 23 and substituting equation 24 thus yields

$$-\int q(m) \ln p(m) dm = \frac{1}{2} \ln 2\pi\sigma_m^2 + \frac{(\bar{m}-\mu_m)^2 + \tilde{m}}{2\sigma_2} \,. \tag{25}$$

A similar term, with m replaced by v, will be obtained from $-\int q(v)\ln p(v)dv$.

The last terms are of the form $-\int q(m,v)\ln p(x(t)|m,v)dmdv$. Again we will find out that the factorisation $q(m,v) = q(m)q(v)$ simplifies the computation of these terms. Recall that $x(t)$ was assumed to be Gaussian with mean m and variance e^{2v}. The term over which the expectation is taken is thus

$$-\ln p(x(t)|m,v) = \frac{1}{2}\ln 2\pi + v + (x(t)-m)^2 e^{-2v} . \tag{26}$$

The expectation over the term $(x(t)-m)^2 e^{-2v}$ is easy to compute since m and v are assumed to be posteriorly independent. This means that it is possible to take the expectation separately from $(x(t)-m)^2$ and e^{-2v}. Using a similar derivation as in equation 24 yields $(x(t)-\bar{m})^2 + \tilde{m}$ for the first term. The expectation over e^{-2v} is also fairly easy to compute:

$$\begin{aligned}\int q(v)e^{-2v}dv &= \frac{1}{\sqrt{2\pi\tilde{v}}}\int e^{-\frac{(v-\bar{v})^2}{2\tilde{v}}} e^{-2v}dv = \frac{1}{\sqrt{2\pi\tilde{v}}}\int e^{-\frac{(v-\bar{v})^2+4v\tilde{v}}{2\tilde{v}}}dv = \\ &\frac{1}{\sqrt{2\pi\tilde{v}}}\int e^{-\frac{v^2-2v\bar{v}+\bar{v}^2+4v\tilde{v}}{2\tilde{v}}}dv = \\ &\frac{1}{\sqrt{2\pi\tilde{v}}}\int e^{-\frac{[v+(2\tilde{v}-\bar{v})]^2+4\bar{v}\tilde{v}-4\tilde{v}^2}{2\tilde{v}}} = \\ &\frac{1}{\sqrt{2\pi\tilde{v}}}\int e^{-\frac{[v+(2\tilde{v}-\bar{v})]^2}{2\tilde{v}}} e^{2\tilde{v}-2\bar{v}}dv = e^{2\tilde{v}-2\bar{v}} .\end{aligned} \tag{27}$$

This shows that taking expectation over equation 26 yields a term

$$-\int q(m,v)\ln p(x(t)|m,v)dmdv = \frac{1}{2}\ln 2\pi + \bar{v} + [(x(t)-\bar{m})^2 + \tilde{m}]e^{2\tilde{v}-2\bar{v}} . \tag{28}$$

Collecting together all the terms, we obtain the following cost function

$$\begin{aligned}C_{m,v}(\boldsymbol{x};\bar{m},\tilde{m},\bar{v},\tilde{v}) = &\sum_{t=1}^{N}\frac{1}{2}[(x(t)-\bar{m})^2+\tilde{m}]e^{2\tilde{v}-2\bar{v}} + N\bar{v} + \\ &\frac{(\bar{m}-\mu_m)^2+\tilde{m}}{2\sigma_m^2} + \frac{(\bar{v}-\mu_v)^2+\tilde{v}}{2\sigma_v^2} + \ln\sigma_m\sigma_v + \\ &\frac{N}{2}\ln 2\pi - \frac{1}{2}\ln\tilde{m}\tilde{v} - 1\end{aligned} \tag{29}$$

Assuming σ_m^2 and σ_v^2 are very large, the minimum of the cost function can be solved by setting the gradient of the cost function C to zero. This yields the following:

$$\bar{m} = \frac{1}{N} \sum_t x(t) \tag{30}$$

$$\tilde{m} = \frac{\sum_t (x(t) - \bar{m})^2}{N(N-1)} \tag{31}$$

$$\bar{v} = \frac{1}{2N} + \frac{1}{2} \ln \frac{1}{N-1} \sum_t (x(t) - \bar{m})^2 \tag{32}$$

$$\tilde{v} = \frac{1}{2N} \tag{33}$$

In case σ_m^2 and σ_v^2 cannot be assumed very large, the equations for $\bar{v}$ and $\tilde{v}$ are not that simple, but the solution can still be obtained by solving the zero-point of the gradient.

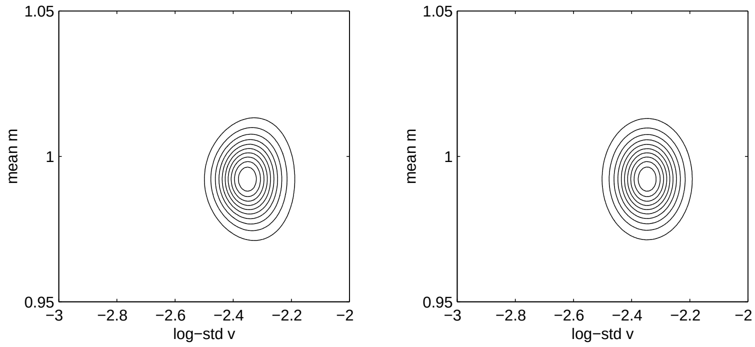


Fig. 3. Comparison of the true and approximate posterior distributions for a test set containing 100 data points drawn from a model with $m = 1$ and $\sigma = 0.1$. The plot on the left shows the true posterior distribution over m and v. The plot on the right shows the approximate posterior distribution consisting of a diagonal Gaussian distribution.

Figure 3 shows a comparison of the true posterior distribution and the approximate posterior. The data set consisted of 100 points drawn from a model with $m = 1$ and $\sigma = 0.1$. The contours in both distributions are centred in the same region, a model that underestimates m.The contours for the two distributions are qualitatively similar although the true distribution is not symmetrical about the mean value of v.

6.2 Free Form Q

Instead of assuming the form for the approximate posterior distribution we could instead derive the optimal separable distribution (the functions that minimise the cost function subject to the constraint that they be normalised).

Instead of learning the log-std $v = \ln\sigma$, we shall learn the inverse noise variance $\beta = \sigma^{-2}$. The prior on β is assumed to be a Gamma distribution of the form

$$p(\beta) = \frac{1}{\Gamma(c_\beta)} b_\beta^{c_\beta} \beta^{(c_\beta - 1)} \exp(-b_\beta \beta) \tag{34}$$

Setting $b_\beta = c_\beta = 10^{-3}$ leads to a broad prior in $\ln\beta$, this is equivalent to assuming that σ_v is large in the log-std parametrisation.

The cost function that must be minimised is now

$$\begin{aligned} C &= D(q(x,m,\beta)||P(x|m,\beta)) - \ln P(m,\beta) \\ &= \int q(m,\beta) \ln \frac{q(m,\beta)}{P(x,m,\beta)} dm d\beta \end{aligned} \tag{35}$$

If we assume a separable posterior, that is $q(m,\beta) = q(m)q(\beta)$ and substitute our priors into the cost function we obtain

$$\begin{aligned} C &= \int q(m)q(\beta) \ln \frac{q(m)q(\beta)}{\left[\prod_t P(x(t)|m,\beta)\right] P(m)P(\beta)} dm d\beta \\ &= \int q(m)q(\beta) \left[\ln q(m) + \ln q(\beta) - \ln P(m) - \ln P(\beta) \right. \\ &\quad \left. - \sum_t \left(\frac{1}{2} \ln \frac{\beta}{2\pi} - \frac{\beta\,(x(t) - m)^2}{2} \right) \right] dm d\beta \end{aligned} \tag{36}$$

Assuming we know $q(\beta)$ we can integrate over β in C (dropping any terms that are independent of m) to obtain

$$C = \int q(m) \left[\ln q(m) - \ln P(m) - \sum_t \left(-\frac{\bar{\beta}\,(x(t) - m)^2}{2} \right) \right] dm \tag{37}$$

where $\bar{\beta}$ is the average value of β under the distribution $q(\beta)$. We can optimise the cost function with respect to $q(m)$ by performing a functional derivative.

$$\begin{aligned} \frac{\partial C}{\partial q(m)} &= 1 + \ln q(m) - \ln P(m) - \sum_t \left(-\frac{\bar{\beta}\,(x(t) - m)^2}{2} \right) + \lambda_m \\ &= 0 \end{aligned} \tag{38}$$

where λ_m is a Lagrange multiplier introduced to ensure that $q(m)$ is normalised. Rearranging we see that

$$\ln q(m) = -1 - \frac{1}{2} \ln 2\pi\sigma_m^2 - \frac{(m - mu_m)^2}{2\sigma_m^2}$$

$$= + \sum_t \left(- \frac{\bar{\beta} \left(x(t) - m \right)^2}{2} \right) + \lambda_m \tag{39}$$

and so the approximate posterior distribution is a Gaussian with variance $\tilde{m} = \left(\sigma_m^{-2} + T\bar{\beta} \right)^{-1}$ and mean $\bar{m} = \tilde{m} \left(\frac{\mu_m}{\sigma_m^2} + \bar{\beta} \sum_t x(t) \right)$.

We can obtain the optimum form for $q(\beta)$ by marginalising the cost function over m and dropping terms independent of β.

$$\begin{aligned} C = \int q(\beta) \Big[& \ln q(\beta) - \ln P(\beta) \\ & - \sum_t \left(\frac{1}{2} \ln \beta - \frac{\beta \left(\left(x(t) - \bar{m} \right)^2 + \tilde{m} \right)}{2} \right) \Big] d\beta \end{aligned} \tag{40}$$

Again we can perform a functional derivative to obtain

$$\begin{aligned} \frac{\partial C}{\partial q(\beta)} &= 1 + \ln q(\beta) - \ln P(\beta) \\ &\quad - \sum_t \left(\frac{1}{2} \ln \beta - \frac{\beta \left(\left(x(t) - \bar{m} \right)^2 + \tilde{m} \right)}{2} \right) + \lambda_\beta \\ &= 0 \end{aligned} \tag{41}$$

and so

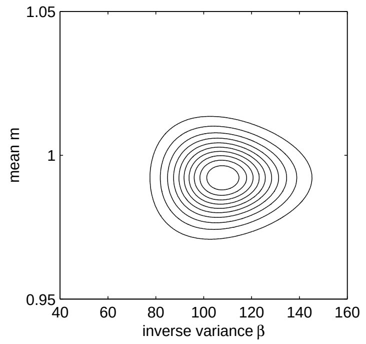
$$\ln q(\beta) = -1 + \sum_t \left(\frac{1}{2} \ln \beta - \frac{\beta \left(\left(x(t) - \bar{m} \right)^2 + \tilde{m} \right)}{2} \right) + \ln P(\beta) + \lambda_\beta \tag{42}$$

So the optimal posterior distribution is a Gamma distribution, with parameters $\hat{b}_\beta = b_\beta + \frac{\sum_t \left(\left(x(t) - \bar{m} \right)^2 + \tilde{m} \right)}{2}$ and $\hat{c}_\beta = c_\beta + \frac{T}{2}$. Therefore the expectation of β under the posterior distribution is $\bar{\beta} = \frac{\hat{c}_\beta}{\hat{b}_\beta}$.

The optimal distributions for m and β depend on each other ($q(m)$ is a function of $\bar{\beta}$ and $q(\beta)$ is a function of $\bar{m}$ and $\tilde{m}$) so the optimal solutions can be found by iteratively updating $q(m)$ and $q(\beta)$.

A general point is that the freeform optimisation of the cost function will typically lead to a set of iterative update equations where each distribution is updated on the basis of the other distributions in the approximation.

We can also see that if the parametrisation of the model is chosen appropriately the optimal separable model has a similar form to the prior model. If the prior distributions are Gaussians the posterior distributions are also Gaussians (likewise for Gamma distributions). If this is the case then we can say that we have chosen conjugate priors.



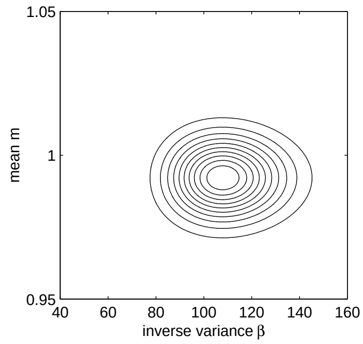


Fig. 4. Comparison of the true and approximate posterior distributions for a test set containing 100 data points drawn from a model with $m = 1$ and $\sigma = 0.1$. The plot on the left shows the true posterior distribution over m and β. The plot on the right shows the approximate posterior distribution derived by obtaining the optimal free form separable distribution.

Figure 4 shows a comparison of the true posterior distribution and the approximate posterior. The data set is the same as for the fixed form example. The contours in both distributions are centred in the same region, a model that underestimates m. The contours for the two distributions are qualitatively similar, the approximate distribution also shows the assymmetric density.

7 Summary

In Ensemble Learning, the search for good models is sensitive to high probability mass and so the problems of over-fitting inherent to maximum likelihood and maximum posterior probability methods are removed.

The approximation of the posterior distribution assumes some degree of factorisation of the true distribution in order to make the approximation more tractable. Additionally the fixed form approximation also assumes some functional form of the factors.

It is often possible to affect the correctness of the approximation by the choice of parameterisation of the model. Also, the learning process tries to make the approximation more correct.

The free form approximation of the separable distribution will often result in Gaussian, Gamma, etc distributions if the parametrization of the model is chosen suitably. Therefore the optimisation process will suggest a parametrization for the problem.

References

1. Geoffrey E. Hinton and Drew van Camp: 'Keeping neural networks simple by minimizing the description length of the weights'. In: *Proceedings of the COLT'93*, (Santa Cruz, California, 1993) pp. 5–13

2. S. Kullback and R. A. Leibler: 'On information and sufficiency'. The Annals of Mathematical Statistics **22**, pp. 79–86 (1951)
3. Radford M. Neal: 'Bayesian Learning for Neural Networks'. Lecture Notes in Statistics No. 118 (Springer-Verlag, 1996)
4. Radford M. Neal and Geoffrey E. Hinton: 'A view of the EM algorithm that justifies incremental, sparse and other variants'. In: *Learning in Graphical Models*. ed. by Michael I. Jordan (1998)
5. Claude E. Shannon: 'A mathematical theory of communication'. Bell System Technical Journal **27**, pp. 379–423 and 623–656 (1948)
6. S. M. Stigler: 'Translation of Laplace's 1774 memoir on "Probability of causes"'. Statistical Science, **1(3)**, pp. 359–378 (1986)