*BAYESIAN STATISTICS 8, pp. 3–24.*
*J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid,*
*D. Heckerman, A. F. M. Smith and M. West (Eds.)*


# Generative or Discriminative? Getting the Best of Both Worlds

Christopher M. Bishop
*Microsoft Research, UK*
cmbishop@microsoft.com

Julia Lasserre
*Cambridge University. UK*
jal62@cam.ac.uk

Summary

For many applications of machine learning the goal is to predict the value of a vector $\boldsymbol{c}$ given the value of a vector $\boldsymbol{x}$ of input features. In a classification problem $\boldsymbol{c}$ represents a discrete class label, whereas in a regression problem it corresponds to one or more continuous variables. From a probabilistic perspective, the goal is to find the conditional distribution $p(\boldsymbol{c}|\boldsymbol{x})$. The most common approach to this problem is to represent the conditional distribution using a parametric model, and then to determine the parameters using a training set consisting of pairs $\{\boldsymbol{x}_n, \boldsymbol{c}_n\}$ of input vectors along with their corresponding target output vectors. The resulting conditional distribution can be used to make predictions of $\boldsymbol{c}$ for new values of $\boldsymbol{x}$. This is known as a discriminative approach, since the conditional distribution discriminates directly between the different values of $\boldsymbol{c}$.

An alternative approach is to find the joint distribution $p(\boldsymbol{x}, \boldsymbol{c})$, expressed for instance as a parametric model, and then subsequently uses this joint distribution to evaluate the conditional $p(\boldsymbol{c}|\boldsymbol{x})$ in order to make predictions of $\boldsymbol{c}$ for new values of $\boldsymbol{x}$. This is known as a generative approach since by sampling from the joint distribution it is possible to generate synthetic examples of the feature vector $\boldsymbol{x}$. In practice, the generalization performance of generative models is often found to be poorer than than of discriminative models due to differences between the model and the true distribution of the data.

When labelled training data is plentiful, discriminative techniques are widely used since they give excellent generalization performance. However, although collection of data is often easy, the process of labelling it can be expensive. Consequently there is increasing interest in generative methods since these can exploit unlabelled data in addition to labelled data.

Although the generalization performance of generative models can often be improved by 'training them discriminatively', they can then no longer make use of unlabelled data. In an attempt to gain the benefit of both generative and discriminative approaches, heuristic procedure have been proposed which interpolate between these two extremes by taking a convex combination of the generative and discriminative objective functions.

Julia Lasserre is funded by the Microsoft Research European PhD Scholarship programme.

Here we discuss a new perspective which says that there is only one correct way to train a given model, and that a 'discriminatively trained' generative model is fundamentally a new model (Minka, 2006). From this viewpoint, generative and discriminative models correspond to specific choices for the prior over parameters. As well as giving a principled interpretation of 'discriminative training', this approach opens the door to very general ways of interpolating between generative and discriminative extremes through alternative choices of prior. We illustrate this framework using both synthetic data and a practical example in the domain of multi-class object recognition. Our results show that, when the supply of labelled training data is limited, the optimum performance corresponds to a balance between the purely generative and the purely discriminative. We conclude by discussing how to use a Bayesian approach to find automatically the appropriate trade-off between the generative and discriminative extremes.

*Keywords and Phrases:* GENERATIVE, DISCRIMINATIVE, BAYESIAN INFERENCE, SEMI-SUPERVIZED, UNLABELLED DATA, MACHINE LEARNING

## 1. INTRODUCTION

In many applications of machine learning the goal is to take a vector $\boldsymbol{x}$ of input features and to assign it to one of a number of alternative classes labelled by a vector $\boldsymbol{c}$ (for instance, if we have $C$ classes, then $\boldsymbol{c}$ might be a $C$-dimensional binary vector in which all elements are zero except the one corresponding to the class).

In the simplest scenario, we are given a training data set $\boldsymbol{X}$ comprising $N$ input vectors $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ together with a set of corresponding labels $\boldsymbol{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N\}$, in which we assume that the input vectors, and their labels, are drawn independently from the same fixed distribution. Our goal is to predict the class $\widehat{\boldsymbol{c}}$ for a new input vector $\widehat{\boldsymbol{x}}$, and so we require the conditional distribution

$$p(\widehat{\boldsymbol{c}}|\widehat{\boldsymbol{x}}, \boldsymbol{X}, \boldsymbol{C}). \tag{1}$$

To determine this distribution we introduce a parametric model governed by a set of parameters $\boldsymbol{\theta}$. In a *discriminative* approach we define the conditional distribution $p(\boldsymbol{c}|\boldsymbol{x}, \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ are the parameters of the model. The likelihood function is then given by

$$L(\boldsymbol{\theta}) = p(\boldsymbol{C}|\boldsymbol{X}, \boldsymbol{\theta}) = \prod_{n=1}^{N} p(\boldsymbol{c}_n|\boldsymbol{x}_n, \boldsymbol{\theta}). \tag{2}$$

The likelihood function can be combined with a prior $p(\boldsymbol{\theta})$, to give a joint distribution

$$p(\boldsymbol{\theta}, \boldsymbol{C}|\boldsymbol{X}) = p(\boldsymbol{\theta})L(\boldsymbol{\theta}) \tag{3}$$

from which we can obtain the posterior distribution by normalizing

$$p(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{C}) = \frac{p(\boldsymbol{\theta})L(\boldsymbol{\theta})}{p(\boldsymbol{C}|\boldsymbol{X})} \tag{4}$$

where

$$p(\boldsymbol{C}|\boldsymbol{X}) = \int p(\boldsymbol{\theta})L(\boldsymbol{\theta})\, d\boldsymbol{\theta}. \tag{5}$$

Predictions for new inputs are then made by marginalizing the predictive distribution with respect to $\boldsymbol{\theta}$ weighted by the posterior distribution

$$p(\widehat{\boldsymbol{c}}|\widehat{\boldsymbol{x}}, \boldsymbol{X}, \boldsymbol{C}) = \int p(\widehat{\boldsymbol{c}}|\widehat{\boldsymbol{x}}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{C})\, d\boldsymbol{\theta}. \tag{6}$$

In practice this marginalization, as well as the normalization in (5), are rarely tractable and so approximation, schemes such as variational inference, must be used. If training data is plentiful a point estimate for $\boldsymbol{\theta}$ can be made by maximizing the posterior distribution to give $\boldsymbol{\theta}_{\mathrm{MAP}}$, and the predictive distribution then estimated using

$$p(\widehat{\boldsymbol{c}}|\widehat{\boldsymbol{x}}, \boldsymbol{X}, \boldsymbol{C}) \simeq p(\widehat{\boldsymbol{c}}|\widehat{\boldsymbol{x}}, \boldsymbol{\theta}_{\mathrm{MAP}}). \tag{7}$$

Note that maximizing the posterior distribution (4) is equivalent to maximizing the joint distribution (3) since these differ only by a multiplicative constant. In practice, we typically take the logarithm before maximizing as this gives rise to both analytical and numerical simplifications. If we consider a prior distribution $p(\boldsymbol{\theta})$ which is constant over the region in which the likelihood function is large, then maximizing the posterior distribution is equivalent to maximizing the likelihood. More generally, we make predictions by marginalizing over all values of $\boldsymbol{\theta}$ using either Monte Carlo methods or an appropriate deterministic approximation framework (Bishop, 2006). In all cases, however, the key quantity for model training is the likelihood function $L(\boldsymbol{\theta})$.

Discriminative methods give good predictive performance and have been widely used in many applications. In recent years there has been growing interest in a complementary approach based on *generative* models, which define a joint distribution $p(\boldsymbol{x}, \boldsymbol{c}|\boldsymbol{\theta})$ over both input vectors and class labels (Jebara, 2004). One of the motivations is that in complex problems such as object recognition, where there is huge variability in the range of possible input vectors, it may be difficult or impossible to provide enough labelled training examples, and so there is increasing use of *semi-supervised* learning in which the labelled training examples are augmented with a much larger quantity of unlabelled examples. A discriminative model cannot make use of the unlabelled data, as we shall see, and so in this case we need to consider a generative approach.

The complementary properties of generative and discriminative models have led a number of authors to seek methods which combine their strengths. In particular, there has been much interest in 'discriminative training' of generative models (Bouchard and Triggs, 2004; Holub and Perona, 2005; Yakhnenko, Silvescu, and Honavar, 2005) with a view to improving classification accuracy. This approach has been widely used in speech recognition with great success (Kapadia, 1998) where generative hidden Markov models are trained by optimizing the predictive conditional distribution. As we shall see later, this form of training can lead to improved performance by compensating for model mis-specification, that is differences between the true distribution of the process which generates the data, and the distribution specified by the model. However, as we have noted, discriminative training cannot take advantage of unlabelled data. In particular it has been shown (Ng and Jordan, 2002) that logistic regression (the discriminative counterpart of a Naive Bayes generative model) works better than its generative counterpart, but only for a large number of training data points (large depending on the complexity of the problem), which confirms the need for using unlabelled data.

Recently several authors (Bouchard and Triggs, 2004; Holub and Perona, 2005; Raina, Shen, Ng, and McCallum, 2003) have proposed hybrids of the generative

and discriminative approaches in which a model is trained by optimizing a convex combination of the generative and discriminative log likelihood functions. Although the motivation for this procedure was heuristic, it was sometimes found that the best predictive performance was obtained for intermediate regimes in between the discriminative and generative limits.

In this paper we develop a novel viewpoint (Minka, 2005, Bishop, 2006) which says that, for a given model, there is a unique likelihood function and hence there is only one correct way to train it. The 'discriminative training' of a generative model is instead interpreted in terms of standard training of a different model, corresponding to a different choice of distribution. This removes the apparently ad-hoc choice for the training criterion, so that all models are trained according to the principles of statistical inference. Furthermore, by introducing a constraint between the parameters of this model, through the choice of prior, the original generative model can be recovered.

As well as giving a novel interpretation for 'discriminative training' of generative models, this viewpoint opens the door to principled blending of generative and discriminative approaches by introducing priors having a soft constraint amongst the parameters. The strength of this constraint therefore governs the balance between generative and discriminative.

In Section 2 we give a detailed discussion of the new interpretation of discriminative training for generative models, and in Section 3 we illustrate the advantages of blending between generative and discriminative viewpoints using a synthetic example in which the role of unlabelled data and of model mis-specification becomes clear. In Section 4 we show that this approach can be applied to a large scale problem in computer vision concerned with object recognition in images, and finally we draw some conclusions in Section 5.

## 2. A NEW VIEW OF 'DISCRIMINATIVE TRAINING'

A generative model can be defined by specifying the joint distribution $p(\boldsymbol{x}, \boldsymbol{c}|\boldsymbol{\theta})$ of the input vector $\boldsymbol{x}$ and the class label $\boldsymbol{c}$, conditioned on a set of parameters $\boldsymbol{\theta}$. Typically this is done by defining a prior probability for the classes $p(\boldsymbol{c}|\boldsymbol{\pi})$ along with a class-conditional density for each class $p(\boldsymbol{x}|\boldsymbol{c}, \boldsymbol{\lambda})$, so that

$$p(\boldsymbol{x}, \boldsymbol{c}|\boldsymbol{\theta}) = p(\boldsymbol{c}|\boldsymbol{\pi})p(\boldsymbol{x}|\boldsymbol{c}, \boldsymbol{\lambda}) \tag{8}$$

where $\boldsymbol{\theta} = \{\boldsymbol{\pi}, \boldsymbol{\lambda}\}$. Since the data points are assumed to be independent, the joint distribution is given by

$$L_{\mathrm{G}}(\boldsymbol{\theta}) = p(\boldsymbol{X}, \boldsymbol{C}, \boldsymbol{\theta}) = p(\boldsymbol{\theta}) \prod_{n=1}^{N} p(\boldsymbol{x}_n, \boldsymbol{c}_n|\boldsymbol{\theta}). \tag{9}$$

This can be maximized to determine the most probable (MAP) value of $\boldsymbol{\theta}$. Again, since $p(\boldsymbol{X}, \boldsymbol{C}, \boldsymbol{\theta}) = p(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{C})p(\boldsymbol{X}, \boldsymbol{C})$, this is equivalent to maximizing the posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{C})$.

In order to improve the predictive performance of generative models it has been proposed to use 'discriminative training' (Yakhnenko *et al.*, 2005) which involves maximizing

$$L_{\mathrm{D}}(\boldsymbol{\theta}) = p(\boldsymbol{C}, \boldsymbol{\theta}|\boldsymbol{X}) = p(\boldsymbol{\theta}) \prod_{n=1}^{N} p(\boldsymbol{c}_n|\boldsymbol{x}_n, \boldsymbol{\theta}) \tag{10}$$

in which we are conditioning on the input vectors instead of modelling their distribution. Here we have used

$$p(\boldsymbol{c}|\boldsymbol{x},\boldsymbol{\theta}) = \frac{p(\boldsymbol{x},\boldsymbol{c}|\boldsymbol{\theta})}{\sum_{\boldsymbol{c}'} p(\boldsymbol{x},\boldsymbol{c}'|\boldsymbol{\theta})}. \tag{11}$$

Note that (10) is not the joint distribution for the original model defined by (9), and so does not correspond to MAP for this model. The terminology of 'discriminative training' is therefore misleading, since for a given model there is only one correct way to train it. It is not the training method which has changed, but the model itself.

This concept of discriminative training has been taken a stage further (Yakhnenko, Silvescu, and Honavar, 2005) by maximizing a function given by a convex combination of (9) and (10) of the form

$$\alpha \ln L_{\mathrm{D}}(\boldsymbol{\theta}) + (1-\alpha) \ln L_{\mathrm{G}}(\boldsymbol{\theta}) \tag{12}$$

where $0 \leqslant \alpha \leqslant 1$, so as to interpolate between generative ($\alpha = 0$) and discriminative ($\alpha = 1$) approaches. Unfortunately, this criterion was not derived by maximizing the distribution of a well-defined model.

Following (Minka, 2005) we therefore propose an alternative view of discriminative training, which will lead to an elegant framework for blending generative and discriminative approaches. Consider a model which contains an additional independent set of parameters $\widetilde{\boldsymbol{\theta}} = \{\widetilde{\boldsymbol{\pi}}, \widetilde{\boldsymbol{\lambda}}\}$ in addition to the parameters $\boldsymbol{\theta} = \{\boldsymbol{\pi}, \boldsymbol{\lambda}\}$, in which the likelihood function is given by

$$q(\boldsymbol{x},\boldsymbol{c}|\boldsymbol{\theta},\widetilde{\boldsymbol{\theta}}) = p(\boldsymbol{c}|\boldsymbol{x},\boldsymbol{\theta}) p(\boldsymbol{x}|\widetilde{\boldsymbol{\theta}}) \tag{13}$$

where

$$p(\boldsymbol{x}|\widetilde{\boldsymbol{\theta}}) = \sum_{\boldsymbol{c}'} p(\boldsymbol{x},\boldsymbol{c}'|\widetilde{\boldsymbol{\theta}}). \tag{14}$$

Here $p(\boldsymbol{c}|\boldsymbol{x},\boldsymbol{\theta})$ is defined by (11), while $p(\boldsymbol{x},\boldsymbol{c}|\widetilde{\boldsymbol{\theta}})$ has independent parameters $\widetilde{\boldsymbol{\theta}}$.

The model is completed by defining a prior $p(\boldsymbol{\theta},\widetilde{\boldsymbol{\theta}})$ over the model parameters, giving a joint distribution of the form

$$q(\boldsymbol{X},\boldsymbol{C},\boldsymbol{\theta},\widetilde{\boldsymbol{\theta}}) = p(\boldsymbol{\theta},\widetilde{\boldsymbol{\theta}}) \prod_{n=1}^{N} p(\boldsymbol{c}_n|\boldsymbol{x}_n,\boldsymbol{\theta}) p(\boldsymbol{x}_n|\widetilde{\boldsymbol{\theta}}). \tag{15}$$

Now suppose we consider a special case in which the prior factorizes, so that

$$p(\boldsymbol{\theta},\widetilde{\boldsymbol{\theta}}) = p(\boldsymbol{\theta}) p(\widetilde{\boldsymbol{\theta}}). \tag{16}$$

We then determine optimal values for the parameters $\boldsymbol{\theta}$ and $\widetilde{\boldsymbol{\theta}}$ in the usual way by maximizing (15), which now takes the form

$$\left[ p(\boldsymbol{\theta}) \prod_{n=1}^{N} p(\boldsymbol{c}_n|\boldsymbol{x}_n,\boldsymbol{\theta}) \right] \left[ p(\widetilde{\boldsymbol{\theta}}) \prod_{n=1}^{N} p(\boldsymbol{x}_n|\widetilde{\boldsymbol{\theta}}) \right]. \tag{17}$$

We see that the resulting value of $\boldsymbol{\theta}$ will be identical to that found by maximizing (11), since it is the same function which is being maximized. Since it is $\boldsymbol{\theta}$ and not $\widetilde{\boldsymbol{\theta}}$ which determines the predictive distribution $p(\boldsymbol{c}|\boldsymbol{x}, \boldsymbol{\theta})$ we see that this model is equivalent in its predictions to the 'discriminatively trained' generative model. This gives a consistent view of training in which we always maximize the joint distribution, and the distinction between generative and discriminative training lies in the choice of model.

The relationship between the generative model and the discriminative model is illustrated using directed graphs in Fig. 1.

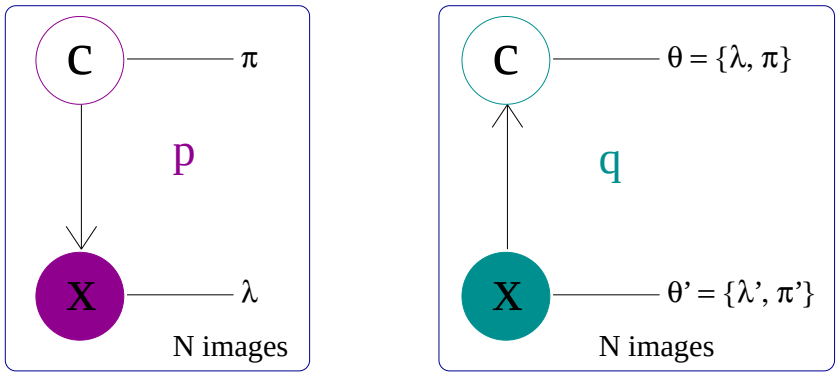


**Figure** 1: *Probabilistic directed graphs, showing on the left, the original generative model, and on the right the corresponding discriminative model.*

Now suppose instead that we consider a prior which enforces equality between the two sets of parameters

$$p(\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) = p(\boldsymbol{\theta})\delta(\boldsymbol{\theta} - \widetilde{\boldsymbol{\theta}}). \tag{18}$$

Then we can set $\widetilde{\boldsymbol{\theta}} = \boldsymbol{\theta}$ in (13) from which we recover the original generative model $p(\boldsymbol{x}, \boldsymbol{c}|\boldsymbol{\theta})$. Thus we have a single class of distributions in which the discriminative model corresponds to independence in the prior, and the generative model corresponds to an equality constraint in the prior.

### 2.1. *Blending Generative and Discriminative*

Clearly we can now blend between the generative and discriminative extremes by considering priors which impose a soft constraint between $\widetilde{\boldsymbol{\theta}}$ and $\boldsymbol{\theta}$. Why should we wish to do this?

First of all, we note that the reason why 'discriminative training' might give better results than direct use of the generative model, is that (15) is more flexible than (9) since it relaxes the implicit constraint $\widetilde{\boldsymbol{\theta}} = \boldsymbol{\theta}$. Of course, if the generative model were a perfect representation of reality (in other words the data really came from the model) then increasing the flexibility of the model would lead to poorer

results. Any improvement from the discriminative approach must therefore be the result of a mis-match between the model and the true distribution of the (process which generates the) data. In other words, the benefit of 'discriminative training' is dependent on model mis-specification.

Conversely, the benefit of the generative approach is that it can make use of unlabelled data to augment the labelled training set. Suppose we have a data set comprising a set of inputs $\boldsymbol{X}_{\mathrm{L}}$ for which we have corresponding labels $\boldsymbol{C}_{\mathrm{L}}$, together with a set of inputs $\boldsymbol{X}_{\mathrm{U}}$ for which we have no labels. For the correctly trained generative model, the function which is maximized is given by

$$p(\boldsymbol{\theta}) \prod_{n \in \mathrm{L}} p(\boldsymbol{x}_n, \boldsymbol{c}_n|\boldsymbol{\theta}) \prod_{m \in \mathrm{U}} p(\boldsymbol{x}_m|\boldsymbol{\theta}) \tag{19}$$

where $p(\boldsymbol{x}|\boldsymbol{\theta})$ is defined by

$$p(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{\boldsymbol{c}'} p(\boldsymbol{x}, \boldsymbol{c}'|\boldsymbol{\theta}). \tag{20}$$

We see that the unlabelled data influences the choice of $\boldsymbol{\theta}$ and hence affects the predictions of the model. By contrast, for the 'discriminatively trained' generative model the function which is now optimized is again the product of the prior and the likelihood function and so takes the form

$$p(\boldsymbol{\theta}) \prod_{n \in \mathrm{L}} p(\boldsymbol{x}_c|\boldsymbol{x}_n, \boldsymbol{\theta}) \tag{21}$$

and we see that the unlabelled data plays no role. Thus, in order to make use of unlabelled data we cannot use a discriminative approach.

Now let us consider how a combination of labelled and unlabelled data can be exploited from the perspective of our new approach defined by (15), for which the joint distribution becomes

$$\begin{aligned} q(\boldsymbol{X}_{\mathrm{L}}, \boldsymbol{C}_{\mathrm{L}}, \boldsymbol{X}_{\mathrm{U}}, \boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) &= p(\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) \\ &\left[\prod_{n \in \mathrm{L}} p(\boldsymbol{c}_n|\boldsymbol{x}_n, \boldsymbol{\theta}) p(\boldsymbol{x}_n|\widetilde{\boldsymbol{\theta}})\right] \left[\prod_{m \in \mathrm{U}} p(\boldsymbol{x}_m|\widetilde{\boldsymbol{\theta}})\right]. \end{aligned} \tag{22}$$

We see that the unlabelled data (as well as the labelled data) influences the parameters $\widetilde{\boldsymbol{\theta}}$ which in turn influence $\boldsymbol{\theta}$ via the soft constraint imposed by the prior.

In general, if the model is not a perfect representation of reality, and if we have unlabelled data available, then we would expect the optimal balance to lie neither at the purely generative extreme nor at the purely discriminative extreme.

As a simple example of a prior which interpolates smoothly between the generative and discriminative limits, consider the class of priors of the form

$$p(\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}) \propto p(\boldsymbol{\theta}) p(\widetilde{\boldsymbol{\theta}}) \frac{1}{\sigma} \exp\left\{-\frac{1}{2\sigma^2} \|\boldsymbol{\theta} - \widetilde{\boldsymbol{\theta}}\|^2\right\}. \tag{23}$$

If desired, we can relate $\sigma$ to an $\alpha$ like parameter by defining a map from $(0, 1)$ to $(0, \infty)$, for example using

$$\sigma(\alpha) = \left(\frac{\alpha}{1 - \alpha}\right)^2. \tag{24}$$

For $\alpha \to 0$ we have $\sigma \to 0$, and we obtain a hard constraint of the form (18) which corresponds to the generative model. Conversely for $\alpha \to 1$ we have $\sigma \to \infty$ and we obtain an independence prior of the form (16) which corresponds to the discriminative model.

## 3. ILLUSTRATION

We now illustrate the new framework for blending between generative and discriminative approaches using an example based on synthetic data. This is chosen to be as simple as possible, and so involves data vectors $\boldsymbol{x}_n$ which live in a two-dimensional Euclidean space for easy visualization, and which belong to one of two classes. Data from each class is generated from a Gaussian distribution as illustrated in Fig. 2.

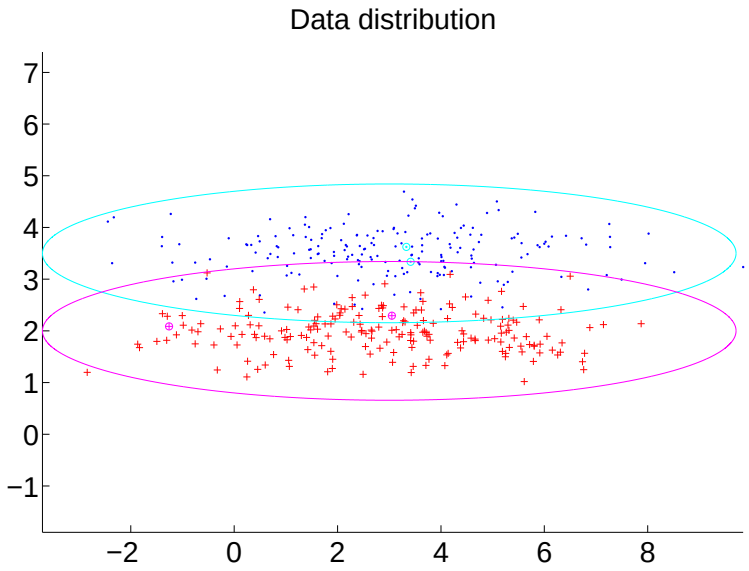


**Figure 2:** *Synthetic training data, shown as crosses and dots, together with contours of probability density for each of the two classes. Two points from each class are labelled (indicated by circles around the data points).*

Here the scales on the axes are equal, and so we see that the class-conditional densities are elongated in the horizontal direction.

We now consider a continuum of models which interpolate between purely generative and purely discriminative. To define this model we consider the generative limit, and represent each class-conditional density using an isotropic Gaussian distribution. Since this does not capture the horizontally elongated nature of the true class distributions, this represents a form of model mis-specification. The parameters of the model are the means and variances of the Gaussians for each class, along with the class prior probabilities.

We consider a prior of the form (23) in which $\sigma(\alpha)$ is defined by (24). Here we choose $p(\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}|\alpha) = p(\boldsymbol{\theta})\ N(\widetilde{\boldsymbol{\theta}}|\boldsymbol{\theta}, \sigma(\alpha))$, where $p(\boldsymbol{\theta})$ is the usual conjugate prior (a Gaussian-gamma prior for the means and variances, and a Dirichlet prior for the class label). The generative model consists of a spherical Gaussian per class, with mean $\boldsymbol{\mu}$ and a diagonal precision matrix $\Delta \boldsymbol{I}$, so that $\boldsymbol{\theta} = \{\boldsymbol{\mu}_k, \Delta_k\}$ and $\widetilde{\boldsymbol{\theta}} = \{\widetilde{\boldsymbol{\mu}}_k, \widetilde{\Delta}_k\}$. Specifically we have chosen

$$p(\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}}|\alpha) \propto \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\theta}', \sigma(\alpha)) \prod_k \left[\mathcal{N}({\boldsymbol{\mu}'}_k|\mathbf{0}, (10{\Delta'}_k)^{-1})\mathcal{G}({\Delta'}_k|0.01, 100)\mathcal{G}(\Delta_k|0.01, 100)\right] \quad (25)$$

where $\mathcal{N}(\cdot|\cdot,\cdot)$ denotes a Gaussian distribution and $\mathcal{G}(\cdot|\cdot,\cdot)$ denotes a gamma distribution.

The training data set comprises 200 points from each class, of which just two from each class are labelled, and the test set comprises 200 points all of which are labelled. Experiments are run 10 times with differing random initializations (including the random selection of which subset of training points to label) and the results used to computer a mean and variance over the test set classification, which are shown by 'error bars' in Fig. 3.

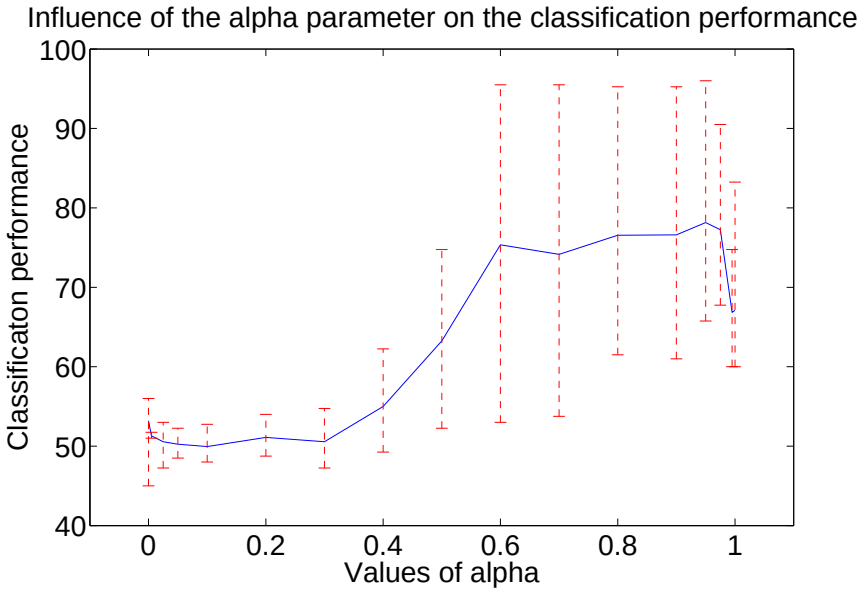


**Figure** 3: *Plot of the percentage of correctly classified points on the test set versus $\alpha$ for the synthetic data problem.*

We see that the best generalization occurs for values of $\alpha$ intermediate between the generative and discriminative extremes.

To gain insight into this behaviour we can plot the contours of density for each class corresponding to different values of $\alpha$, as shown in Fig. 4.

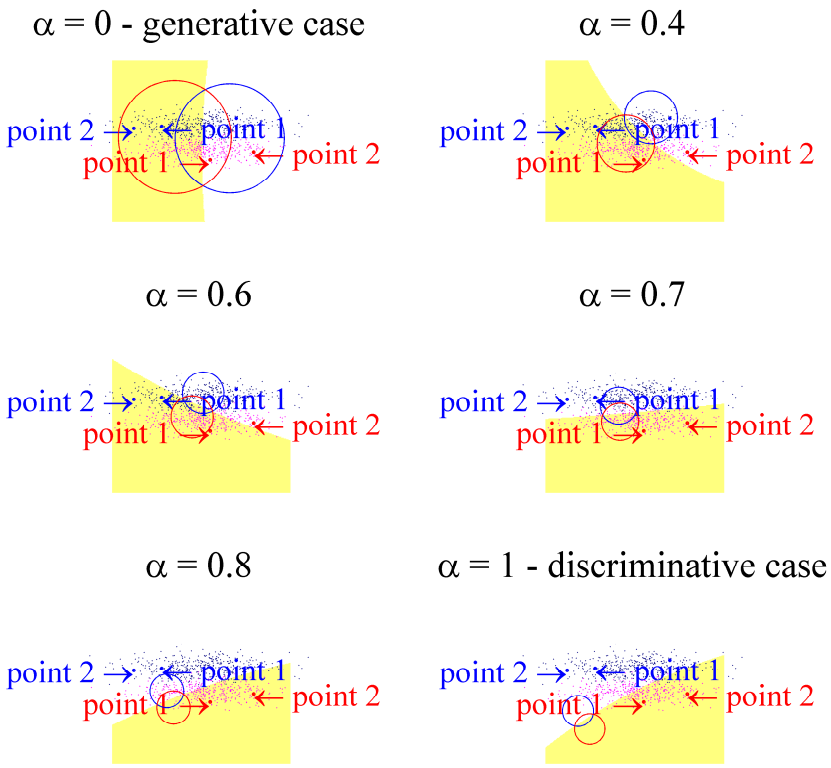


**Figure** 4: *Results of fitting an isotropic Gaussian model to the synthetic data for various values of $\alpha$. The top left shows $\alpha = 0$ (generative case) while the bottom right shows $\alpha = 1$ (discriminative case). The gray area corresponds to points that are assigned to the red class.*

We see that a purely generative model is strongly influenced by modelling the density of the data and so gives a decision boundary which is orthogonal to the correct one. Conversely a purely discriminative model attends only to the labelled data points and so misses useful information about the horizontal elongation of the true class-conditional densities which is present in the unlabelled data.

## 4. OBJECT RECOGNITION

We now apply our approach to a realistic application involving object recognition in static images. This is a widely studied problem which has been tackled using a

range of different discriminative and generative models. The long term goal of such research is to achieve near human levels of recognition accuracy across thousands of object classes in the presence of wide variations in location, scale, orientation and lighting, as well as changes due to intra-class variability and occlusion.

### 4.1. *The Data*

We used eight different classes: airplanes, bikes, cows, faces, horses, leaves, motorbikes, sheep (Bishop, 2006). Together these images exhibit a wide variety of poses, colours, and illumination, as illustrated by the sample images shown in Fig. 5. The goal is to assign images from the test set to one of the eight classes.

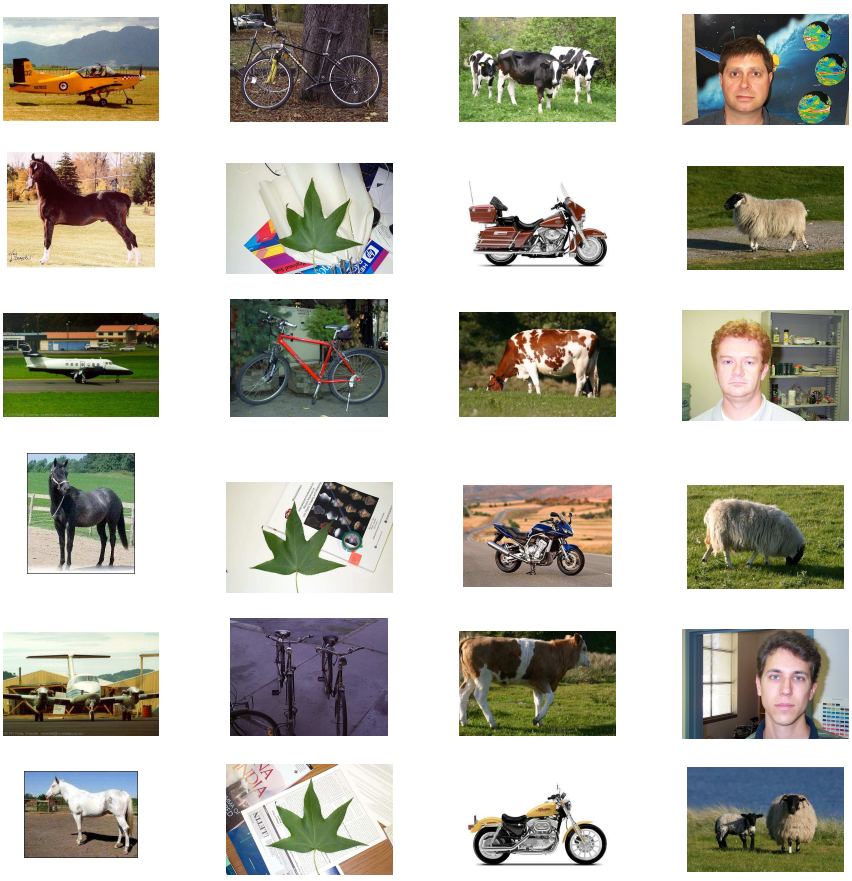

**Figure** 5: *Sample images from the training set.*

### 4.2. *The Features*

Our features are taken from Winn, Criminisi, and Minka (2005), in which the original RGB images are first converted into the CIE $(L, a, b)$ colour space (Kasson and Plouffe, 1992). All images were re-scaled to $300 \times 200$, and raw patches of size $48 \times 48$ were extracted on a regular grid of size $24 \times 24$ (i.e. every 24th pixel). Each patch is then convolved with 17 filters, and the set of corresponding pixels from each of the filtered images represents a 17-dimensional vector. All these feature vectors are clustered using $K$-means with $K = 100$. Since this large value of $K$ is computationally costly in later stages of processing, PCA is used to give a 15-dimensional feature vector. Winn *et al.* (2005) use a more powerful technique to reduce the number of features, but since this is a supervised method based on fully labelled training data, we did not re-implement it here. The cluster centers obtained through $K$-means are called *textons* (Varma and Zisserman, 2005).

The filters are quite standard: the first three filters are obtained by scaling a Gaussian filter, and are applied to each channel of the colour image, which gives $3 \times 3 = 9$ response images. Then a Laplacian filter is applied to the L channel, at four different scales, which gives four more response images. Finally two DoG (difference of Gaussians) filters (one along each direction) are applied to the L channel, at two different scales, giving another four responses.

From these response images, we extract every pixel on a $4 \times 4$ grid, and apply $K$-means to obtain $K$ textons. Now each patch will be represented by a histogram of these textons, i.e. by a $K$-dimensional vector containing the proportion of each texton. Textons were obtained from 25 training images per class (half of the training set). Note that the texton features are found using only unlabelled data. These vectors are then reduced using PCA to a dimensionality of 15.

### 4.3. *The Model*

We consider the generative model introduced by Ulusoy and Bishop (2005), which we now briefly describe. Each image is represented by a feature vector $\boldsymbol{x}_n$, where $n = 1, \ldots, N$, and $N$ is the total number of images. Each vector comprises a set of $J$ patch vectors $\boldsymbol{x} = \{\boldsymbol{x}_{nj}\}$ where $j = 1, \ldots, J$. We assume that each patch belongs to one and only one of the classes, or to a separate 'background' class, so that each patch can be characterized by a binary vector $\boldsymbol{\tau}_{nj}$ coded so that all elements of $\boldsymbol{\tau}_{nj}$ are zero except the element corresponding to the class. We use $\boldsymbol{c}_n$ to denote the image label vector for image $n$ with independent components $c_{nk} \in \{0, 1\}$ in which $k = 1, \ldots C$ labels the class.

The overall joint distribution for the model can be represented as a directed graph, as shown in Fig. 6.

We can therefore characterize the model completely in terms of the conditional probabilities $p(\boldsymbol{c})$, $p(\boldsymbol{\tau}|\boldsymbol{c})$ and $p(\boldsymbol{x}|\boldsymbol{\tau})$. This model is most easily explained generatively, that is, we describe the procedure for generating a set of observed feature vectors from the model.

First we choose the overall class of the image according to some prior probability parameters $\psi_k$ where $k = 1, \ldots, C$, and $0 \leqslant \psi_k \leqslant 1$, with $\sum_k \psi_k = 1$, so that

$$p(\boldsymbol{c}) = \prod_{k=1}^{C} \psi_k^{c_k}. \tag{26}$$

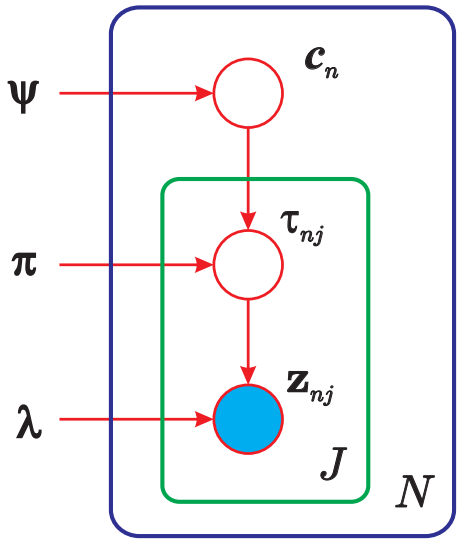


**Figure 6:** *The generative model for object recognition expressed as a directed acyclic graph, for unlabelled images, in which the boxes denote 'plates' (i.e. independent replicated copies). Only the patch feature vectors $\{\boldsymbol{x}_{nj}\}$ are observed, corresponding to the shaded node. The image class labels $\boldsymbol{c}_n$ and patch class labels $\boldsymbol{\tau}_{nj}$ are latent variables.*

Given the overall class for the image, each patch is then drawn from either one of the foreground classes or the background ($k = C + 1$) class. The probability of generating a patch from a particular class is governed by a set of parameters $\pi_k$, one for each class, such that $\pi_k \geqslant 0$, constrained by the subset of classes actually present in the image. Thus

$$p(\boldsymbol{\tau}_j|\boldsymbol{c}) = \left(\sum_{l=1}^{C+1} c_l \pi_l\right)^{-1} \prod_{k=1}^{C+1} (c_k \pi_k)^{\tau_{jk}}. \tag{27}$$

Note that there is an overall undetermined scale to these parameters, which may be removed by fixing one of them, e.g. $\pi_{C+1} = 1$.

For each class, the distribution of the patch feature vector $\boldsymbol{x}$ is governed by a separate mixture of Gaussians which we denote by

$$p(\boldsymbol{x}|\boldsymbol{\tau}_j) = \prod_{k=1}^{C+1} \phi_k(\boldsymbol{x}_j; \boldsymbol{\lambda}_k)^{\tau_{jk}} \tag{28}$$

where $\boldsymbol{\lambda}_k$ denotes the set of parameters (means, covariances and mixing coefficients) associated with this mixture model.

If we assume $N$ independent images, and for image $n$ we have $J$ patches drawn

independently, then the joint distribution of all random variables is

$$\prod_{n=1}^{N}\left[p(\boldsymbol{c}_n)\prod_{j=1}^{J}p(\boldsymbol{x}_{nj}|\boldsymbol{\tau}_{nj})p(\boldsymbol{\tau}_{nj}|\boldsymbol{c}_n)\right]. \tag{29}$$

Here we are assuming that each image has the same number $J$ of patches, though this restriction is easily relaxed if required.

The graph shown in Fig. 6 corresponds to unlabelled images in which only the feature vectors $\{\boldsymbol{x}_{nj}\}$ are observed, with both the image category and the classes of each of the patches being latent variables. It is also possible to consider images which are 'weakly labelled', that is each image is labelled according to the category of object present in the image. This corresponds to the graphical model of Fig. 7 in which the node $\boldsymbol{c}_n$ is shaded.

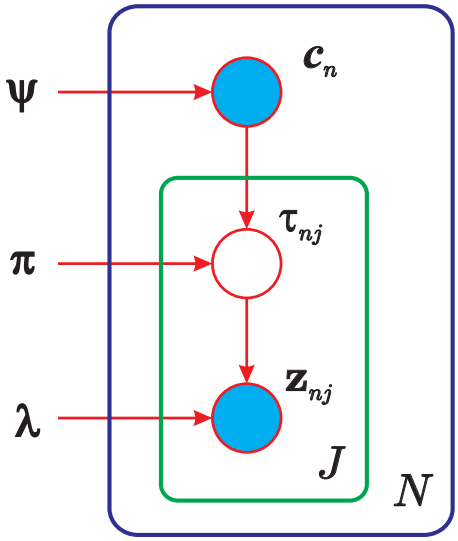


**Figure 7:** *Graphical model corresponding to Fig. 6 for weakly labelled images.*

Of course, for a given size of data set, better performance is expected if all of the images are 'strongly labelled', that is segmented images in which the region occupied by the object or objects is known so that the patch labels $\boldsymbol{\tau}_{nj}$ become observed variables. The graphical model for a set of strongly labelled images is shown in Fig. 8.

Strong labelling requires hand segmentation of images, and so is a time consuming and expensive process as compared with collection of the images themselves. For a given level of effort it will always be possible to collect many unlabelled or weakly labelled images for the same cost as a single strongly labelled image. Since the variability of natural images and objects is so vast we will always be operating in a regime in which the size of our data sets is statistically small (though they will often be computationally large).

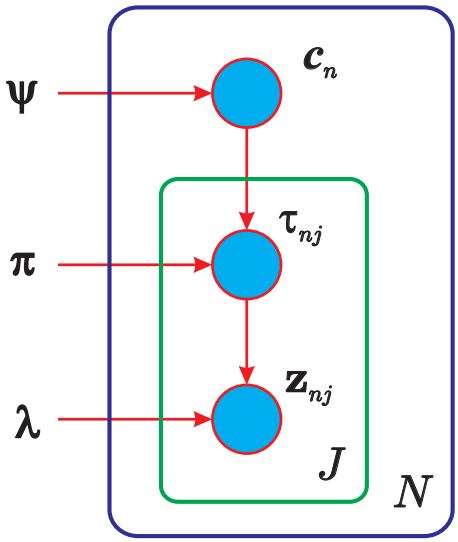


**Figure** 8: *Graphical models corresponding to Fig. 6 for strongly labelled images.*

For this reason there is great interest in augmenting expensive strongly labelled images with lots of cheap weakly labelled or unlabelled images in order to better characterize the different forms of variability. Although the two stage hierarchical model shown in Fig. 6 appears to be more complicated than in the simple example shown in Fig. 1, it does in fact fall within the same framework. In particular, for labelled images the observed data is $\{\boldsymbol{x}_n, \boldsymbol{c}_n, \boldsymbol{\tau}_{nj}\}$, while for 'unlabelled' images only $\{\boldsymbol{x}_n\}$ are observed. The experiments described here could readily be extended to consider arbitrary combinations of strongly labelled, weakly labelled and unlabelled images if desired.

If we let $\boldsymbol{\theta} = \{\psi_k, \pi_k, \boldsymbol{\lambda}_k\}$ denote the full set of parameters in the model, then we can consider a model of the form (22) in which the prior is given by (23) with $\sigma(\alpha)$ defined by (24), and the terms $p(\boldsymbol{\theta})$ and $p(\widetilde{\boldsymbol{\theta}})$ taken to be constant.

We use conjugate gradients to optimize the parameters. Due to lack of space we do not write down all the derivatives of the log likelihood function required by the conjugate gradient algorithm. However, the correctness of the mathematical derivation of these gradients, as well as their numerical implementation, can easily be verified by comparison against numerical differentiation (Bishop, 1995). The conjugate gradients is the most used technique when it comes to blending generative and discriminative models, thanks to its flexibility. Indeed, because of the discriminative component $p(\boldsymbol{c}_n|\boldsymbol{x}_n, \boldsymbol{\theta})$ which contains a normalizing factor, an algorithm such as EM would require much more work, as nothing is directly tractable anymore. However, a comparison of the two methods is currently being investigated.

### 4.4. *Results*

We use 50 training images per class (giving 400 training images in total) of which five images per class (a total of 40) were fully labelled, *i.e.*, both the image and the

individual patches have class labels. All the other images are left totally unlabelled, i.e. not even the category they belong to is given. Note that this kind of training data is (1) very cheap to get and (2) very unusual for a discriminative model. The test set consists of 100 images per class (giving a total of 800 images), the task is to label each image.

Experiments are run five times with differing random initializations and the results used to compute a mean and variance over the test set classification, which are shown by 'error bars' in Fig. 9.

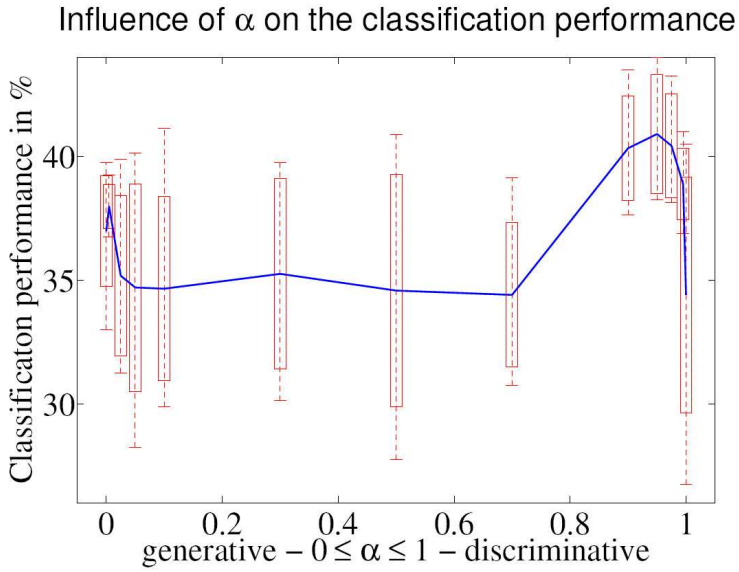


**Figure** 9: *Influence of the term $\alpha$ on the test set classification performance.*

Note that, since there are eight balanced classes, random guessing would give 12.5% correct on average. Again we see that the best performance is obtained with a blend between generative and discriminative extremes.

## 5. CONCLUSIONS

In this paper we have shown that 'discriminative training' for generative models can be re-cast in terms of standard training methods applied to a modified model. This new viewpoint opens the door to a wide range of new models which interpolate smoothly between generative and discriminative approaches and which can benefit from the advantages of both. The main drawback of this framework is that the number of parameters in the model is doubled leading to greater computational cost.

Although we have focussed on classification problems, the framework is equally applicable to regression problems in which $\boldsymbol{c}$ corresponds to a set of continuous variables.

A principled approach to combining generative and discriminative approaches not only gives a more satisfying foundation for the development of new models, but it also brings practical benefits. In particular, the parameter $\alpha$ which governs the trade-off between generative and discriminative is now a hyper-parameter within a well defined probabilistic model which is trained using the (unique) correct likelihood function. In a Bayesian setting the value of this hyper-parameter can therefore be optimized by maximizing the marginal likelihood in which the model parameters have been integrated out, thereby allowing the optimal trade-off between generative and discriminative limits to be determined entirely from the training data without recourse to cross-validation (Bishop, 2006). This extension of the work described here is currently being investigated.

## REFERENCES

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

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Berlin: Springer-Verlag

Bouchard, G. and Triggs, B. (2004). The trade-off between generative and discriminative classifiers. *IASC 16th International Symposium on Computational Statistics, Prague, Czech Republic*, 721–728.

Holub, A. and Perona, P. (2005). A discriminative framework for modelling object classes. *IEEE Conference on Computer Vision and Pattern Recognition*, San Diego (California), USA. IEEE Computer Society.

Jebara, T. (2004). *Machine Learning: Discriminative and Generative*. Dordrecht: Kluwer

Kapadia, S. (1998). *Discriminative Training of Hidden Markov Models*. PhD Thesis, University of Cambridge, UK.

Kasson, J. M. and Plouffe, W. (1992). An analysis of selected computer interchange color spaces. *ACM Transactions on Graphics* **11**, 373–405.

Minka, T. (2005). Discriminative models, not discriminative training. *Tech. Rep.*, Microsoft Research, Cambridge, UK.

Ng, A. Y. and Jordan, M. I. (2002). On discriminative vs. generative: A comparison of logistic regression and naive Bayes. *Advances in Neural Information Processing Systems* **14**, (T. G. Dietterich, S. Becker, and Z. Ghahramani, eds.) Cambridge, MA: The MIT Press, 841–848.

Raina, R., Shen, Y., Ng, A. Y. and McCallum, A. (2003). Classification with hybrid generative/discriminative models. *Advances in Neural Information Processing Systems* **16** Cambridge, MA: The MIT Press, 545–552.

Ulusoy, I. and Bishop, C. M. (2005). Generative versus discriminative models for object recognition. *Proceedings IEEE International Conference on Computer Vision and Pattern Recognition*, CVPR., San Diego.

Varma, M. and Zisserman, A. (2005). A statistical approach to texture classification from single images. *IJCV* **62**, 61–81.

Winn, J., Criminisi, A. and Minka, T. (2005). Ob ject categorization by learned universal visual dictionary. *IEEE International Conference on Computer Vision*, Beijing, China. IEEE Computer Society.

Yakhnenko, O., Silvescu, A. and Honavar, V. (2005). Discriminatively trained Markov model for sequence classification. *5th IEEE International Conference on Data Mining*, Houston (Texas), USA. IEEE Computer Society.

## DISCUSSION

HERBERT K. H. LEE (*University of California, Santa Cruz, USA*)

Let me start by congratulating the authors for this paper. In terms of 'Getting the best of both worlds', this paper can also be seen as crossing between machine learning and statistics, combining useful elements from both. I can only hope that our two communities continue to interact, deepening our connections. The perspectives are often complementary, which is an underlying theme of this discussion.

One of the obstacles to working across disciplines is that a very different vocabulary may be used to describe the same concepts. Statisticians reading this paper may find it helpful to think of supervised learning as classification, and unsupervised learning as clustering. Discriminative learning is thus classification based on a probability model (which it typically is in statistics) while the generative approach is clustering based on a model (such as a mixture model approach).

While machine learning and statistics are really quite similar, there is a key difference in perspective. In machine learning, the main goal is typically to achieve good predictions, and while a probability model may be used, it is not explicitly required. In contrast, most statistical analyses see the probability model as the core of the analysis, with the idea that optimal predictions will arise from accurate selection and fitting of the model. In particular, Bayesian analyses rely crucially on a fully-specified probability model. Thus one of the core points of this paper, that of a unique likelihood function, should seem natural to a Bayesian. Yet it is an important insight in the context of the machine learning literature. Bringing these machine learning algorithms into a coherent probabilistic framework is a big step forward, and one that is not always fully valued. This is an important contribution by these authors and their collaborators.

Uncertainty about the likelihood function can be dealt with by embedding the functions under consideration into a larger family with one or more additional parameters, and this is exactly what has been done here. This follows a strong tradition in statistics, such as Box-Cox transformations and model averaging, but represents a relatively untapped potential in machine learning. In contrast, it is more common in machine learning to use implicit expansions of the model class (or of the fitting algorithm, when the model may not be explicitly stated). Examples include bagging (Breiman, 1996), where individual predictions from over-fit models are averaged over bootstrap samples to reduce over-fitting, and boosting (Freund and Schapire, 1997), where overly-simple models are combined to create an improved ensemble prediction. Such implicit expansion can work well in practice, but it can be difficult to understand or describe the expanded class of models, and hence difficult to leverage related knowledge from the literature.

On a related note, the authors argue that a key benefit of using discriminative training for generative models is that it improves performance when the model is mis-specified, as vividly demonstrated by the example in Section 3. In practice, this is quite useful, as our parametric models are typically only approximations to reality, and the approximations can be quite poor. But this does leave open the possibility of explicit model expansion. A larger parametric family may encompass a model which is close enough to reality. Or taking things even further, one could move to a fully nonparametric approach. Then it becomes less clear what the trade-offs are.

Many highly innovative and creative ideas have arisen in machine learning, and the field of statistics has gained by importing some of these ideas. Statistics, in turn, can offer a substantial literature that can be applied once a machine learning

algorithm can be mapped to a probability model. From the model, one can draw from the literature to better understand when the algorithm will work best, when it might perform poorly, what diagnostics may be applicable, and possibly how to further improve the algorithm. The key is connecting the algorithm to a probability model, either finding the model which implicitly underlies the algorithm, or showing that the algorithm approximates a particular probability model. These sorts of connections benefit both fields.

Thus thinking more about Bayesian probability models, some possible further directions for this current work come to mind. It would seem natural to put a prior on $\alpha$ and to treat it as another parameter. At least from the experiments shown so far, it appears that there may be some information about likely best ranges of $\alpha$, allowing the use of an informative prior, possibly in comparison to a flat prior. In addition to the possibility of marginalizing over $\alpha$, one could also estimate $\alpha$ to obtain a 'best' fit.

Another natural possible direction would be to look at the full posterior, rather than just getting a point estimate, such as the maximum a posteriori class estimate. Knowing about the uncertainty of the classification can often be useful. It may also be useful to move to a more hierarchical model, particularly for the image classes. It would seem that images of horses and cows would be more similar to each other, and images of bicycles and motorbikes would be similar to each other, but that these two sets would be rather different from each other, and further different from faces or leaves. Working within a hierarchical structure should be straightforward in a fully Bayesian paradigm.

In terms of connections between machine learning and statistics, it seems unfortunate that the machine learning literature takes little notice of related work in the statistics literature. In particular, there has been extensive work on model-based clustering, for example, Fraley and Raftery (2002) and even a poster presented at this conference (Frühwirth-Schnatter and Pamminger, 2006). It would be great if the world of machine learning were more cognizant of the statistical literature.

In summary, this paper presents a practical solution to a problem that is definitely in need of attention, and which has received relatively little attention in the statistical literature. The likelihood-based approach is particularly promising. The authors make a very positive contribution in helping to bridge the gap between machine learning and statistics, and I hope to see more such work in the future.

STEPHEN E. FIENBERG (*Carnegie Mellon University, USA*)

The presentation by Bishop and Laserre (BL) is an especially welcome contribution to the Bayesian statistical world. It attempts to formulate a principled approach to combining discriminative and generative approaches to learning in a setting focused on classification. As someone engaged in both activities (although with far more emphasis going into the generative modeling) and whose appointments are currently in both a department of statistics and in the first machine learning department in the world, I admire the clarity of their initial distinction between the two approaches and at their framing of their strengths and weaknesses. I'd like to offer two observations.

First, in the statistical world we actually spend a lot of time worrying about problems other than classification. Thus many of us spend a large amount of our time worrying about generative models in a much broader context. We focus on producing families of parametric models that span the space of reasonable models for the phenomenon of interest and attempt to imbue the parameters $\theta$, often associated with latent quantities which in the Bayesian context are also random variables,

with interpretations that are part of a generative process, often of a dynamic form. Inevitably, our models are oversimplifications and we learn both from the activity of parameter estimation and model choice and from careful examination and interpretation of the posterior distribution of the parameters as well as from the form of various predictive distributions, e.g., see Blei *et al.* (2003a,2003b), Erosheva (2003), and Erosheva *et al.* (2004). When I teach statistical learning to graduate students in the Machine Learning Department, I emphasize that the world of machine learning would be enriched by taking on at least part of this broader perspective.

My second observation is closely related. While I very much appreciated the BL's goal and the boldness of their effort to formulate a different likelihood function to achieve an integration of the two perspectives, I think the effort would be enhanced by consideration of some of the deeper implications of the subjective Bayesian paradigm. As Bishop noted in his oral response to the discussion, machine learning has been moving full force to adopt formal statistical ideas, and so it is now common to see the incorporation of MAP and model averaging methodologies directly into the classification setting. But as some of the other presentations at this conference make clear, model assessment is a much more complex and subtle activity which we can situate within the subjective Bayesian framework, cf., Draper (1999). In particular, I commend a careful reading of the Lauritzen (2006) discussion of the paper by Chakrabarti and Ghosh (2006), in which he emphasized that we can only come to grips with the model choice problem by considering model comparisons and specifications with respect to our own subjective Bayesian prior distribution (thus taking advantage of the attendant coherence in the sense of de Finetti (1937)) until such time as we need to step outside the formal framework and reformulate our model class and likelihoods. Thus BL's new replicate distribution for $\theta'$ possibly should be replaced by a real subjective prior distribution, perhaps of a similar form, and then they could explore the formulation of the generative model without introducing a likelihood that departs from the one that attempts to describe the underlying phenomenon of interest. This, I suspect, would lead to an even clearer formulation that is more fully rooted in both the machine learning and statistical learning worlds.

## REPLY TO THE DISCUSSION

We would like thank the discussants for their helpful remarks and useful insights. We would also like to take the opportunity to make some comments on an important issue raised by both discussants, namely the relationship between the fields of machine learning and statistics.

Historically, these fields evolved largely independently for many years. Much of the motivation for early machine learning algorithms, such as the perceptron in the 1960s, and multi-layered neural networks in the 1980s, came from loose analogies with neurobiology. Later, the close connections to statistics became more widely recognized, and these have strongly shaped the subsequent evolution of the field during the 1990s and beyond (Bishop 1995 and Ripley 1996). Today, there are many similarities in the techniques employed by the two fields, although as the discussants note, both the vocabulary and the underlying motivations can differ.

The discussants express frustration at the lack of appreciation of the statistics literature by some machine learning researchers. Naturally, the converse also holds, and indeed this conference has provided examples of algorithms proposed as novel which are in fact well known and widely cited in the machine learning world. To some extent this lack of appreciation is understandable. The literature in either

field alone is vast, and the issue of vocabulary mentioned above is a further obstacle to cross-fertilization.

It could be even be argued that the relative independence of the two fields has brought some benefits. For example, the use of large, highly parameterized black-box models trained on large data sets, of the kind which characterized much of the applied work in neural networks in the 1990s, did not fit well with the culture of the statistics community at the time, and met with scepticism from some quarters. Yet these efforts have led to numerous large-scale applications of substantial commercial importance.

Nevertheless, it seems clear that greater cross-fertilization between the two communities would be desirable. Conference such as AI Statistics (Bishop 2003) explicitly seek to achieve this, and several text books also span the divide (Hastie 2001 and Bishop 2006).

Increasingly, the focus in the machine learning community is not just on well-defined probabilistic models, but on fully Bayesian treatments in which distributions over unknown variables are maintained and updated. However, almost any model which has sufficient complexity to be of practical interest will not have a closed-form analytical solution, and hence approximation techniques are essential. For many years the only general purpose approximation framework available was that of Monte Carlo sampling, which is computationally demanding and which does not scale to large problems.

A crucial advance, therefore, has been the development of a wide range of powerful deterministic inference techniques. These include variational inference, expectation propagation, loopy belief propagation, and others. Like Markov chain Monte Carlo, these techniques have their origins in physics. However, they have primarily been developed and applied within the machine learning community. They complement naturally the recent advances in probabilistic graphical models, such as the development of factor graphs. Also, they are computationally much more efficient than Monte Carlo methods, and have permitted, for instance, a fully Bayesian treatment of the player ranking problem in on-line computer games through the TrueSkill$^{\text{TM}}$ system, in which millions of players are compared and ranked, with ranking updates and team matching being done in real time (Herbrich 1996). Here the Bayesian treatment, which maintains a distribution over player skill levels, leads to substantially improved ranking accuracy compared to the traditional ELO method , which used a point estimate and which can be viewed as a maximum likelihood technique. This type of application would be inconceivable without the use of deterministic approximation schemes. In our view, these methods represent the single most important advance in practical Bayesian statistics in the last 10 years.

## ADDITIONAL REFERENCES IN THE DISCUSSION

Breiman, L. (1996). Bagging predictors. *Machine Learning* **26**, 123–140.

Blei D. M., Jordan, M. I. and Ng, A. (2003a). Hierarchical Bayesian models for applications in information retrieval. *Bayesian Statistics 7* (J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith and M. West, eds.) Oxford: University Press, 25–45.

Blei D. M., Jordan, M. I. and Ng, A. (2003b). Latent Dirichlet allocation. *J. Machine Learning Research* **3**, 993–1022.

Chakrabarti, A. and Ghosh, J. K. (2006). Some aspects of Bayesian model selection for prediction. *In this volume.*

de Finetti, B. (1937). La prévision: ses lois logiques, ses sources subjectives. *Annales de l'Institut Henri Poincaré* **7**, 1–68.

Draper D. (1999). Model uncertainty yes, discrete model averaging maybe. (discussion of 'Bayesian model averaging: a tutorial,' by Hoeting *et al.* ). *Statist. Science* **14**, 405–409.

Erosheva, E. A. (2003). Bayesian estimation of the grade of membership model. *Bayesian Statistics 7* (J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith and M. West, eds.) Oxford: University Press, 501–510.

Erosheva, E. A., Fienberg, S. E., and Lafferty, J. (2004). Mixed-membership models of scientific publications. *Proc. National Acad. Sci.* **97**, 11885–11892.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *J. Amer. Statist. Assoc.* **97**, 611–631.

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to aoosting. *J. Computer and System Sciences* **55**, 119–139.

Frühwirth-Schnatter, S. and Pamminger C. (2006). Model-based clustering of discrete-valued time deries sata. *Tech. Rep.*, Johannes Kepler Universität Linz, Austria.

Lauritzen, S. (2006). Discussion of Chakrabarti and Ghosh. *In this volume.*