The Multilevel Classification Problem and a Monotonicity Hint

Malik Magdon-Ismail[1], Hung-Ching (Justin) Chen[2], Yaser S. Abu-Mostafa[3]

[1] magdon@cs.rpi.edu.edu
[2] chenh3@rpi.edu
Dept. of Computer Science, RPI, Lally 207, 110 8th Street, Troy, NY, USA 12180
[3] yaser@cs.caltech.edu
Learning Systems Group,136-93,Caltech, Pasadena, CA, USA, 91125

Abstract. We introduce and formalize the multilevel classification problem, in which each category can be subdivided into different levels. We analyze the framework in a Bayesian setting using Normal class conditional densities. Within this framework, a natural monotonicity hint converts the problem into a nonlinear programming task, with non-linear constraints. We present Monte Carlo and gradient based techniques for addressing this task, and show the results of simulations. Incorporation of monotonicity yields a systematic improvement in performance.

1 Introduction

In most disease handbooks, not only are unrelated diseases listed, but versions (or severities) of a given disease are also provided. An example is the heart condition. A suggested categorization is (in order of increasing severity), essential hypertension, hypertension with complications and secondary hypertension, acute myocardial infarction, coronary atherosclerosis (see for example [2]). A grouping of diseases might look something like

$\vdots$
•Diseases of the Circulatory System
•Diseases Affecting the Heart
–Hypertension ...
$\vdots$

While the categories (diseases) may be unrelated, within a category, the various levels are related by some form of a severity criterion. Some additional examples can be found in [2, 4, 5]. The motivation for such a categorization in handbooks is the need to be able to distinguish between these cases. Different categories (diseases) will have essentially different treatments, and different severities within the same category will usually have different levels of treatment. Similar observations also apply in numerous other areas, for example fault diagnosis in machinery (categories correspond to faults, and levels correspond to the seriousness of the fault), weather prediction (categories correspond to weather patterns, and levels to the magnitude of the expression, for example tornado

versus mild winds, rain versus thunderstorm versus hurricane). In all cases, the goal is to predict the category, and the level within the category, given some observed feature vector (in the case of disease prediction, the symptom). Similar issues also exist in multistage image analysis, [6], where larger parts of the image are categorized first and finer detail added later, and data mining, [7], where the mining occurs at increased levels of generality.

Suppose that there are K categories, and l levels within each category. When $l = 1$ we have the usual K-class pattern recognition problem. When $l > 1$, we have a (K, l)-multiclass-multilevel pattern recognition problem. The simplest approach might be to treat this as a $K \times l$ multiclass problem (with independent classes), but to do so would be ignoring valuable information available about the structure of the learning problem - one expects that the nature of mild heart attack symptoms might convey quite a bit of information about the nature of severe heart attack symptoms. For example, if we know the cholesterol level of mild heart attack victims, it is reasonable to guess that the cholesterol level of severe heart attack victims should be higher. Ignoring this additional information could be a severe handicap, especially if the data set is small and noisy. The purpose of this paper is to develop a mathematical framework for exploiting this added structure in the (K, l)-multilevel-multiclass problem.

2 Problem Setup

Assume the classification problem has K categories, $c_1, \ldots, c_K$. Within each category c_i, there are l_{c_i} levels (which can be viewed as severities). As with most fault classification problems, there is usually a special category, c_0, the normal class (for example, a healthy patient). We will represent this (common) normal class by level 0 in each category. Thus, the classification problem has a total of $1 + \sum_{i=1}^{K} l_{c_i}$ classes. The data set consists of N d-dimensional feature vectors $\mathbf{x}_i$, with $N_{c,l}$ features in each category-level combination. We assume that each feature vector was generated using a Normal class conditional density

$$P[\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}] = \mathcal{N}(\mathbf{x} - \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})} \qquad (1)$$

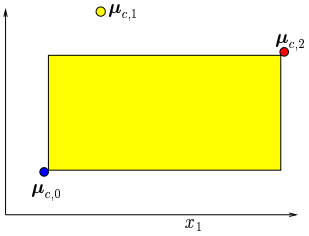
where $(\cdot)'$ represents the transpose operation. In order to keep track of the numerous parameters, we introduce definitions that will be used throughout.

$\mathcal{Q}_{c,l}$: The set of $N_{c,l}$ feature vectors in category c, level l.
$p_{c,l}$: The *a priori* probability of category c, level l. $p_{c,l} \approx N_{c,l}/N$.
$\boldsymbol{\mu}_{c,l}$: The true mean for each class: $E[\mathbf{x}|c, l]$.
$\boldsymbol{\Sigma}_{c,l}$: The true covariance matrix for each class: $E\left[(\mathbf{x} - \boldsymbol{\mu}_{c,l})(\mathbf{x} - \boldsymbol{\mu}_{c,l})'|c, l\right]$.
$\boldsymbol{m}_{c,l}$: The sample mean: $\sum \mathbf{x}/N_{c,l},\ \mathbf{x} \in \mathcal{Q}_{c,l}$.
$\boldsymbol{S}_{c,l}$: The sample covariance matrix: $\sum(\mathbf{x} - \boldsymbol{m}_{c,l})(\mathbf{x} - \boldsymbol{m}_{c,l})'/N_{c,l},\ \mathbf{x} \in \mathcal{Q}_{c,l}$.
$\hat{\boldsymbol{m}}_{c,l}$: The estimated mean.
$\hat{\boldsymbol{S}}_{c,l}$: The estimated covariance matrix: $\sum(\mathbf{x} - \hat{\boldsymbol{m}}_{c,l})(\mathbf{x} - \hat{\boldsymbol{m}}_{c,l})'/N_{c,l},\ \mathbf{x} \in \mathcal{Q}_{c,l}$.
$\mathcal{R}_{c_1,l_1}^{c_2,l_2}$: The risk matrix: cost of classifying (c_2, l_2) when the true class is (c_1, l_1).

We assume that the data is generated independently, according to $p_{c,l}$ and (1), and $\mathcal{R}$ is given. An intuitive example of an asymmetric periodic risk matrix could be as shown to the right, where between category errors are penalized more than within category errors. The periodicity arises because c_1 and c_2 may be any two different categories. The goal is to estimate $p_{c,l}, \boldsymbol{\mu}_{c,l}, \boldsymbol{\Sigma}_{c,l}$, which can be used to implement a Bayes minimal risk classifier [3].

	(c,0)	(c_1,1)	(c_1,2)	(c_1,3)	(c_2,1)	(c_2,2)	(c_2,3)
(c,0)	0	3	7	9	3	7	9
(c_1,1)	3	0	2	4	1	3	5
(c_1,2)	3	2	0	2	3	1	3
(c_1,3)	7	4	2	0	5	3	1
(c_2,1)	3	1	3	5	0	2	4
(c_2,2)	3	3	1	3	2	0	2
(c_2,3)	7	5	3	1	4	2	0

We are now ready to introduce the *monotonicity hint* – it is well known that hints can considerably aid the learning process [1]. The interpretation of the different levels as severities will help motivate the monotonicity hint. Consider a given feature dimension, and a given category c. If the value of that feature increases in going from (say) level 0 (normal) to level 1, then it is reasonable to expect that the value should increase from level 1 to level 2. The situation is illustrated in the figure above where the shaded area represents the allowed region for $\boldsymbol{\mu}_{c,1}$. Monotonicity should hold for every feature dimension, every category c, and for every ordered triple of levels $(i < j < k)$ within that category. Using $\mathbf{a}\cdot^*\mathbf{b}$ to denote component by component multiplication of two vectors, we formalize the monotonicity constraint by



$$(\boldsymbol{\mu}_{c,j} - \boldsymbol{\mu}_{c,i})\cdot^*(\boldsymbol{\mu}_{c,k} - \boldsymbol{\mu}_{c,j}) \geq \mathbf{0}, \qquad \begin{array}{c} \forall\ c \text{ such that } 1 \leq c \leq K \\ \forall\ i,j,k \text{ such that } 0 \leq i < j < k \leq l_c \end{array} \tag{2}$$

It is important that all triples within a category be included in the constraint, and equality with 0 allows for the possibility of irrelevant features.

If the sample means satisfy the monotonicity constraint (2), then there is not much else to do other than estimate the $\boldsymbol{\Sigma}$'s. On the other hand, due to the randomness in the data, the sample means may not satisfy (2), in which case, one ought to be able to improve the risk of the classifier by updating the sample means, taking into account (2). This is the focus of this paper.

2.1 Incorporating the Monotonicity Hint

The likelihood of the sample means, given estimates for the class means and covariance matrices will be Normal with the same mean and a covariance matrix decreased by a factor of $N_{c,l}$, i.e., $P[\mathbf{m}_{c,l}|\hat{\mathbf{m}}_{c,l}, \hat{\mathbf{S}}_{c,l}, N_{c,l}] = \mathcal{N}(\mathbf{m}_{c,l} - \hat{\mathbf{m}}_{c,l}, \hat{\mathbf{S}}_{c,l}/N_{c,l})$. The joint distribution of the sample means is independent given the estimated means, hence

$$P[\{\mathbf{m}_{c,l}\}|\{\hat{\mathbf{m}}_{c,l}, \hat{\mathbf{S}}_{c,l}, N_{c,l}\}] = \prod_{c,l} \mathcal{N}(\mathbf{m}_{c,l} - \hat{\mathbf{m}}_{c,l}, \hat{\mathbf{S}}_{c,l}/N_{c,l}) \tag{3}$$

To convert this likelihood into a posterior, we use the monotonicity constraint to guide the choice of a prior. The only implication of (2) is that the support of

the prior be for assignments to the means that satisfy (2). There are many ways to assign such a prior, and we pick the simplest, namely a prior that assigns a uniform probability density to means that satisfy (2) and a 0 probability density otherwise. We thus conclude that the posterior density for the means is given by

$$P[\{\hat{\mathbf{m}}_{c,l}\}|\{\mathbf{m}_{c,l}, \hat{\mathbf{S}}_{c,l}, N_{c,l}\}] \propto P[\{\hat{\mathbf{m}}_{c,l}\}] \prod_{c,l} \mathcal{N}(\hat{\mathbf{m}}_{c,l} - \mathbf{m}_{c,l}, \hat{\mathbf{S}}_{c,l}/N_{c,l}) \quad (4)$$

where the prior $P[\{\hat{\mathbf{m}}_{c,l}\}]$ is some constant when the set of means satisfies (2) and zero otherwise. This posterior could be used in a Bayesian formalism to obtain expectations. However, we are presently after a specific estimate of the $\boldsymbol{\mu}$'s, namely the maximum *a posteriori* probability (MAP) estimate. Taking the logarithm and discarding constant terms, we get the following optimization problem.

Minimize with respect to $\{\hat{\mathbf{m}}_{c,l}\}$

$$\frac{1}{2} \sum_{l,c} N_{c,l} (\hat{\mathbf{m}}_{c,l} - \mathbf{m}_{c,l})' \hat{\mathbf{S}}_{c,l}^{-1} (\hat{\mathbf{m}}_{c,l} - \mathbf{m}_{c,l}) + \frac{1}{2} \log |\hat{\mathbf{S}}_{c,l}| \quad (5)$$

subject to the non-linear inequality constraints given in (2).

Notice that $\hat{\mathbf{S}}_{c,l}$ depends on $\hat{\mathbf{m}}_{c,l}$. Without the constraints, the solution is given by the sample means, $\hat{\mathbf{m}}_{c,l} = \mathbf{m}_{c,l}$. The objective function encourages the estimates to be close to the sample means, favoring low variance directions of $\hat{\mathbf{S}}$ and classes in which more data are available. However, aside from such intuitive observations, the analytical solution of this problem is elusive. We thus resort to numerical techniques. It is tempting to treat each category independently, but the means interact with each other due to the monotonicity constraint (through the normal class). Two approaches immediately suggest themselves. The first is essentially a global search for the solution, which is feasible in low dimensional problems with few classes. The second is a gradient based technique that is considerably trickier to implement, but more efficient.

Monte Carlo approach: First generate an estimate for the normal mean from $\mathcal{N}(\hat{\mathbf{m}}_{c,0} - \mathbf{m}_{c,0}, \mathbf{S}_{c,0}/N_{c,0})$. Given the normal mean, we generate means in each category independently, according to $\mathcal{N}(\hat{\mathbf{m}}_{c,l} - \mathbf{m}_{c,l}, \mathbf{S}_{c,l}/N_{c,l})$, and accept if they satisfy the monotonicity constraint (2). After generating the monotonic means for every category, we compute the objective (5) and repeat, keeping the set of monotonic means that attains the minimum for (5). It can be shown that with probability approaching 1, for any $\epsilon > 0$, the resulting set of monotonic means will have an objective value at most ϵ greater than the optimal value, as long as the number of Monte Carlo events is allowed to be arbitrarily large.

Gradient based approach: At the expense of introducing a regularization parameter Ω, we convert the optimization problem to one of minimizing an unconstrained objective function $E = E_{prob} + \Omega E_{mon}$. E_{prob} is given in (5), and E_{mon} is designed to have a minimal value when (2) is satisfied. We will use

$$E_{mon} = -\sum_{c=0}^{N} \sum_{0 \le i < j < k \le l_c} \sum_{\alpha=1}^{d} \left[(\hat{m}_{c,j}(\alpha) - \hat{m}_{c,i}(\alpha))(\hat{m}_{c,k}(\alpha) - \hat{m}_{c,j}(\alpha)) \right]^{-} \quad (6)$$

where $[x]^- = x$ if $x < 0$ and 0 otherwise, and $\hat{m}(\alpha)$ is the α^{th} component of $\hat{\mathbf{m}}$. The value of Ω determines the tradeoff between E_{prob} and E_{mon}, and it is only in the limit of large Ω that this solution can approach a solution to the original problem, (5). Among the challenges are the facts that E_{mon} is quite non-linear and non-differentiable. This means that gradient based approaches need to be careful at the non-differentiable points. Nevertheless, we can compute the gradient when it is defined. A complication arises due to the fact the $\hat{\mathbf{S}}$ depends on $\hat{\mathbf{m}}$. Ignoring this dependence, by approximating $\hat{\mathbf{S}} \approx \mathbf{S}$, we get

$$\frac{\partial E_{prob}}{\partial \hat{\mathbf{m}}_{c,l}} = N_{c,l}\mathbf{S}_{c,l}^{-1}(\hat{\mathbf{m}}_{c,l} - \mathbf{m}_{c,l}) \tag{7}$$

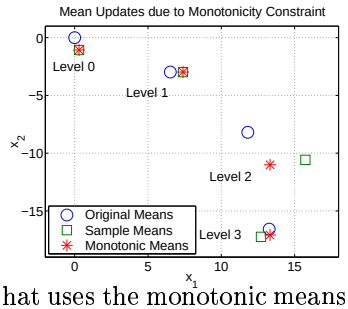
To compute the gradient of the monotonicity error with respect to $\hat{m}_{c,l}(\alpha)$, there are three types of terms that we need to consider – in the summation over triples, l could be the lowest, middle or highest level. The result is

$$\frac{\partial E_{mon}}{\partial \hat{m}_{c,l}(\alpha)} = \sum_{\substack{l<j<k \\ or \\ k<j<l}} \hat{m}_{c,k}(\alpha) - \hat{m}_{c,j}(\alpha) + \sum_{j<l<k} 2\hat{m}_{c,l(\alpha)} - \hat{m}_{c,j}(\alpha) - \hat{m}_{c,k}(\alpha) \tag{8}$$

where contributions to the gradient occur only when a term contributes to E_{mon}. This gradient can now be used to descend on the objective function.

3 Experimental Simulations and Further Study

Since we focus on the update of the means, we show results for a 2-dimensional learning problem in which the covariance matrices of each class were equal, and the means satisfied the monotonicity constraint (2). Since each class had the same covariance matrix, we estimated the covariance matrix using the data from every class. This is the covariance matrix that was used for the classifier using the monotonic means as well as the sample means. We also chose a risk matrix that is 0 along the diagonal and 1 everywhere else, hence the risk is the probability of error. The general problem presents no additional difficulties.



We consider both a 1 category and a 3 category problem, with 4 levels per category (including normal). An example of how the means get updated due to the monotonicity constraint is shown to the right (for 1 category). The total number of data points generated varied from 30 to 300, and for each data set size, we ran 5000 simulations. A simulation entailed generating the data, and computing the risk of the Bayes minimal risk classifier that uses the sample means, compared to the one that uses the monotonic means (obtained using the Monte Carlo and gradient based approaches). These risks were computed using a test set of size 10000. Results of the simulations are shown in Figure 1. The monotonicity hint clearly gives a systematic improvement.

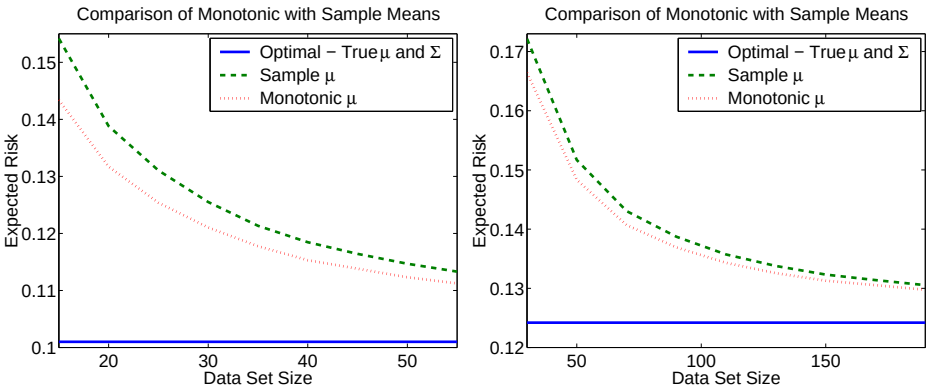


Fig. 1. Risk for the (a) single category and (b) three category learning problems.

Work in progress includes further study of the optimization problem (2), the extension to more complicated learning models such as neural networks, and the application to real world problems.

4 Acknowledgments

Many have contributed to the progress of this work. In particular, we single out Honeywell Corporation for alerting us to the problem and providing initial motivation, James Psota and Amir Atiya for useful discussion.

References

1. Y. Abu-Mostafa. Hints. *Neural Computation*, 4(7):639–671, 1995.
2. Agency for Healthcare Research and Quality. Clinical classifications software. fact sheet. *http://www.ahrq.gov/data/hcup/ccsfact.htm*, Internet Citation.
3. C. Bishop. *Neural Networks for Pattern Recognition*. Clarendon Pr., Oxford, 1995.
4. D. Rixen, J. H. Siegel, and H. P. Friedman. "Sepsis/SIRS" physiologic classification, severity stratification, relation to cytokine elaboration and outcome prediction in posttrauma critical illness. *Journal of Trauma*, 41:581–598, 1996.
5. J. H. Siegel, D. Rixen, and H. P. Friedman. Physiological classification and stratification of illness severity of posttrauma 'sepsis' patients as a basis for randomization of clinical trials. *Journal of Endotoxin Research*, 2(17):177–188, 1995.
6. J. Smith and S. Chang. Multi-stage classification of images from features and related text. *In 4th Europe EDLOS Workshop*, Aug 1997.
7. M. Taylor. Discovering multi-level classification rules in platelet transfusion databases. *Ph.D. Dissertation Proposal, University of Maryland*, December 1996.