Statistica Sinica **16**(2006), 459-469

SIGNAL PROBABILITY ESTIMATION WITH PENALIZED LIKELIHOOD METHOD ON WEIGHTED DATA

Fan Lu, Gary C. Hill, Grace Wahba and Paolo Desiati

University of Wisconsin-Madison

Abstract: In this work we consider the problem faced by astrophysicists where high energy signal neutrinos must be separated from overwhelming background events. We propose a modification to the usual penalized likelihood approach, to take account of the usage of importance sampling techniques in the generation of the simulated training data. Each simulated multivariate data point has two associated weights, which define its contribution to the signal or background count. We wish to find the most powerful decision boundary at a certain significance level to optimally separate signal from background neutrinos. In this modified penalized likelihood method, the estimation of the logit function involves two major optimization steps and the use of KL (Kullback-Leibler) distance criterion for model tuning. We compare this approach with a non-standard SVM (support vector machine) approach. Results on simulated multivariate normal data and simulated neutrino data are presented. For the neutrino data, since the truth is unknown, we show a way to check whether the proposed method is working properly.

Key words and phrases: Kullback-Leiber distance, logit function, neutrino signal, nonstandard support vector machine, penalized likelihood method, simplex method.

1. Introduction

A neutrino is a particle that has no charge and almost no mass. Neutrinos may be produced in the center of active galaxies or from highly energetic objects like γ-ray bursts or black holes. Physicists are trying to use the giant device called AMANDA (Antarctic Muon and Neutrino Detector Array) buried deep in the Antarctic ice cap to detect certain neutrino signals within comparatively overwhelming background noise (Andrés et al. (2001) and Ahrens et al. (2003)). In computer experiments simulating neutrinos passing through AMANDA, distributions of signal and background are generated by an importance sampling procedure which generates events described by multiple feature variables. Each simulated neutrino can represent both signal and background by assignment of an importance sampling weight. The task is to find the most powerful decision boundary at a certain significance level to distinguish signal neutrino from background neutrino. For a detailed description of the problem and data, see Hill, Lu, Desiati and Wabha (2003). Because of the curse of dimensionality, usual

Monte Carlo methods are not practical. We propose a modified penalized log-likelihood approach to solve the usual multivariate problem with this weighted simulated data. Though this study is motivated by the simulated neutrino data, the proposed method can be applied to other multivariate weighted data.

2. Modified Penalized Likelihood Estimation

2.1. Penalized likelihood method for labeled data

Let x be a possibly multidimensional vector of event observables derived from a reconstructed event. Let $h_s(x)$ be the probability density function for signal vectors and $h_b(x)$ be the probability density function for background vectors, and let π_s and π_b be prior probabilities of a signal and background observation, respectively. Then the posterior probability that x is a signal vector is $p(x) = \pi_s h_s(x)/(\pi_b h_b(x) + \pi_s h_s(x))$. The logit $f(x)$ is defined as $\log[p(x)/(1-p(x))] \equiv \theta + \log[h_s(x)/h_b(x)]$, where $\theta = \log(\pi_s/\pi_b)$. We will estimate the logit $f(x)$ for a particular (implicit) value of θ, but since the end result is to obtain level curves of f, the particular value of θ is not important for the calculations. A modified form of the penalized likelihood estimate (Wahba (1990), Wahba, Wang, Gu, Klein and Klein (1995) and Wahba (2002)) will be used.

Let y_i be a random variable that is 1 (signal) with probability $p(x_i)$ and 0 (background) with probability $1 - p(x_i)$. So the observed data are actually class labels. Then the likelihood of a single observation y_i is $\mathcal{L} = p(x_i)^{y_i}(1-p(x_i))^{1-y_i}$. The negative log likelihood of (independent) data $y_1, \ldots, y_n$ is then, in terms of the logit, given by

$$Q(y, f) = \sum_{i=1}^{n} \left[\log\left(1 + e^{f(x_i)}\right) - y_i f(x_i) \right]. \tag{1}$$

We want to find $f \cong \sum c_k B_k \in H_K$ (a reproducing kernel Hilbert space (RKHS), see Aronszajn (1950), Wahba (1990) and Wahba (2002)) which minimizes the penalized log-likelihood:

$$I_\lambda(c) = Q(y, f) + \lambda \parallel f \parallel^2_{H_K}, \tag{2}$$

where B_k's are basis functions in H_K and $\| \cdot \|_{H_K}$ is the function norm in H_K.

This is essentially the penalized log likelihood estimate of f proposed in O'Sullivan, Yandell and Raynor (1986), and in common use in some fields. Under rather general conditions, which include a proper choice of λ, penalized log likelihood estimates are known to converge to the 'true' f as the sample size becomes large. Cox and Osullivan (1990) RKHS are discussed in Aronszajn (1950) and their use in statistical model building in Wahba (1990) and elsewhere. A wide variety of these spaces is available. An RKHS is characterized by a unique

positive definite function $K(\cdot,\cdot)$, and once K is chosen, the exact minimizer of $I_\lambda(c)$ is known to be in the span of a certain set of basis functions determined from K (Kimeldorf and Wahba (1971)). In Section 3 below we select a particular K, known to be a good general-purpose choice, and use an approximating subset of this set of basis functions. Estimating f rather than p directly gives a strictly convex optimization problem whose gradient and Hessian are simple to compute. This makes the numerical analysis easier and thus is suitable for very large data sets. It is possible to estimate p directly, but this estimate is harder to compute in large data sets and is believed to be not as accurate.

2.2. Penalized likelihood method for weighted data

The form of the negative log likelihood in (1) applies when simulated training data is distributed as $h_b(x)$ and $h_s(x)$ through sampling directly from the generating distributions $\Phi_s(\tilde{E})$ and $\Phi_b(\tilde{E})$. The generating distributions $\Phi_s(\tilde{E})$ and $\Phi_b(\tilde{E})$ are constant multiples of distributions based on track generating parameters, e.g., neutrino energy, position and arrival direction. Then, $\Phi_s(\tilde{E})$ and $\Phi_b(\tilde{E})$ can in principle be processed through a simulation chain which mimics the tracks seen by the AMANDA detector array and the process which extracts variables x from the simulated tracks. However, results are expected to have extremely long tails, the region of interest, so an importance sampling scheme was developed. Observable x vectors were generated according to a convenient sampling distribution $g(\tilde{E})$, and pushed through the detector geometry and x variable extraction. For each x_i so obtained, two weights were assigned, $w_s(x_i) = \Phi_s(\tilde{E}_i)/g(\tilde{E}_i)$ for signal, and $w_b(x_i) = \Phi_b(\tilde{E}_i)/g(\tilde{E}_i)$ for background. This $w_s(x_i) + w_b(x_i)$ plays the role of an estimate of relative frequency of x_i in signal + background while $w_s(x_i)/(w_s(x_i) + w_b(x_i))$ plays the same role for the probability of signal given x_i, similarly for the background. The weights satisfy $\sum_{i=1}^n w_s(x_i) = N_s$ and $\sum_{i=1}^n w_b(x_i) = N_b$, where N_s and N_b are the predicted numbers of events from the weighted simulation.

If we had multiple unbiased observations at some x_i as $y_{ij}, j = 1,\ldots,m(i)$, the likelihood of these observations would be $\mathcal{L} = p(x_i)^{\sum_{j=1}^{m(i)} y_{ij}} (1-p(x_i))^{\sum_{j=1}^{m(i)} (1-y_{ij})}$. If the samplings at x_i are biased, then the exponent sums are weighted by $w_s(x_i)$ and $w_b(x_i)$ respectively, leading to a modified likelihood

$$Q(w,f) = \sum_{i=1}^{n} \sum_{y_i=0}^{1} \{w_{y_i}[\log(1+e^{f(x_i)}) - y_i f(x_i)]\}, \tag{3}$$

where $w_{y_i} = w_s(x_i)$ for $y_i = 1$ and $w_{y_i} = w_b(x_i)$ for $y_i = 0$. Note that every x_i comes with a pair of weights instead of a class label. The incorporation of weighted events is thus simply accounted for by weighting the terms in the

logarithmic likelihood sum. Further, we can substitute $w_s(x_i)$ and $w_b(x_i)$ to obtain an alternative form of the likelihood

$$Q(w, f) = \sum_{i=1}^{n} \{w_t(x_i)[\log(1 + e^{f(x_i)}) - \tilde{p}(x_i)f(x_i)]\}, \tag{4}$$

where $w_t(x_i) = w_s(x_i) + w_b(x_i)$ and $\tilde{p}(x_i) = w_s(x_i)/w_t(x_i)$.

Notice that our extension of the penalized likelihood method to weighted data, by defining $Q(w, f)$ in (2) as in (4), is a natural generalization of the original formulation since (4) reduces to (1) in the case of labeled data, which we consider as a special case of weighted data with $(w_s = 1, w_b = 0)$ representing the '1' class and $(w_s = 0, w_b = 1)$ representing the '0' class.

3. Implementation of The Modified Penalized Likelihood Method

With the modified penalized likelihood formulation, we can move on to look for a 'good' estimate of $f(x)$ whose level curves can be obtained. In our implementation, we use radial basis functions plus constant and linear terms. So,

$$f(x) = \beta_0 + \beta^T x + \sum_{k=1}^{N} c_k K_\sigma(x, x_{i_k}), \tag{5}$$

where $K_\sigma(\cdot, \cdot)$ is the Gaussian kernel with isotropic variance σ^2, N is the total number of basis functions, and the x_{i_k}, $k = 1, \ldots, N$, are chosen as a subset of the x_i, $i = 1, \ldots, n$, as described below. Thus, f will be specified as long as, β_0, β and the c_k's are determined (note that β is a vector). By letting $\lambda \parallel f \parallel^2_{H_K} = \lambda \sum_{k,l=1}^{N} c_k c_l K_\sigma(x_{i_k}, x_{i_l})$, we put a penalty only on the c_k's, leaving constant and linear terms unpenalized.

We used a sequence of data-driven procedures to fit the model in the sense that we let the data choose the 'best' combination of smoothing parameter λ, scale parameter σ and number of basis functions N. For a given weighted multidimensional data set, we can first transform each variable to make them of comparable scale. Though these preprocessing procedures are not always necessary, they often improve the performance of our algorithm. Then, the entire data set is randomly divided into three subsets of some preset sizes: a training set, a tuning set and a testing set. After that, we randomly, but according to weights (large-weighted simulated data points have higher chance to be selected), choose a modest sized set of N x_{i_k}'s which determines basis functions as a subset of the training set. We solve the minimization problem on a coarse 2-D parameter grid of λ (usually on a log scale) and σ^2 using the training set. For each parameter pair (each point on the $\log \lambda, \sigma$ grid), an iterative Newton-Raphson algorithm is

used to solve this convex minimization problem, see Wahba (1990). After the algorithm converges, we calculate the Kullback-Leibler (KL) distance between the tuning set and the fitted model. The KL distance is essentially just the first term of $I_\lambda(c)$ for tuning simulated data with f replaced by the estimate $\hat{f}_{\lambda,\sigma^2}$, that is, $\sum_i \sum_{y_i=0}^{1}\{w_{y_i}[\log(1+e^{\hat{f}_{\lambda,\sigma^2}(x_i)}) - y_i\hat{f}_{\lambda,\sigma^2}(x_i)]\}$, where the first sum is taken over x_i in the tuning set. We then find the best parameter combination based on the calculated KL distance over the coarse grid. Starting from there, a direct-search simplex method (Lagarias, Reeds, Wright and Wright (1998)) is used to search for a locally best parameter combination according to the KL distance criterion, see Ferris, Voelker and Zhang (2004). The whole procedure described above is repeated using $2N$ bases, then $4N$ bases and so on, until the improvement on the KL distance is smaller than some preset threshold. We use the coefficients corresponding to the then-best combination of parameters to construct our final estimate of the logit function. Next, the testing set is used to check the goodness of fit of this final model. Finally, the level curves of $p(x)$ are determined and plotted. These level curves are appropriate for use in conjunction with the approaches in Hill and Rawlins (2003), and Feldman and Cousins (1998). Finally, the real data can be analyzed by applying the thresholds given by the level curves of $p(x)$, see Ahrens et al. (2003).

4. Results on Simulated Multivariate Normal Data

Instead of using the neutrino data, for which we don't know the truth, we first test our algorithm on simulated multivariate data. For plotting convenience we only show a two dimensional example. Our algorithm has been tested extensively on higher dimensional simulated data (in particular 5-D, which is the expected dimension of the neutrino data), with success. We generate a random sample of x's from a 2-D uniform distribution (which plays the role of g) over a square. We then associate with each x_i two distinct 2-D Gaussian density values ($w_s(x_i)$ for signal, $w_b(x_i)$ for background), giving a simulated data set consisting of the x_i's and their associated w_b's and w_s's. The sum of the two 2-D Gaussian distributions is shown in Figure 2(a).

For a particular run with sample size $1,000$ (among which we randomly pick 400 for training and another 400 for tuning), the result for the level curve corresponding to $p = 0.9$ is shown in Figure 1. Since we know the truth here, the data points are colored green if the true p is less than or equal to 0.9 and black otherwise. The red line is the level curve found by our algorithm, which is visually almost identical to the true level curve (see Figure 2(c)). We also implemented a nonstandard support vector machine (SVM), see Lin, Lee and Wahba (2002), (the parameters are tuned in a similar way to our modified penalized likelihood method) here for comparison. To use a nonstandard SVM to find the level surface

corresponding to $e^{f(x)} = p(x)/(1-p(x)) = r$, it is not hard to extend the usual SVM formulation to the following weighted regularization problem:

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{y=-1,1} w_{iy}c_y[(1-yf(x_i))_+] + \lambda \parallel f \parallel^2_{H_k},$$

where

$$w_{iy} = \begin{cases} w_b(x_i) & \text{if} \quad y=-1; \\ w_s(x_i) & \text{if} \quad y=1, \end{cases}$$

$$c_y = \begin{cases} r & \text{if} \quad y=-1; \\ 1 & \text{if} \quad y=1, \end{cases}$$

and $(\tau)_+ = \tau$ if $\tau > 0$ and 0 otherwise. We minimize this nonstandard SVM criterion while tuning the smoothing parameter and scale parameter through iteratively calling the well-known SVM software SVM^{light} (version 4.0), which gives the blue line in Figure 1 as the estimated decision boundary.

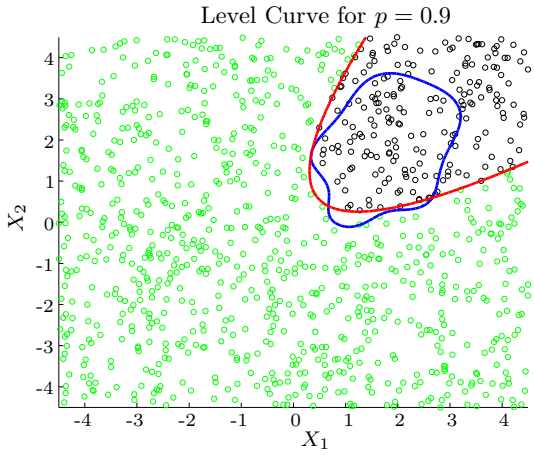

Figure 1. A 2-D EXAMPLE: The red line is the level curve estimated by our penalized likelihood method; the blue line is the level curve estimated by nonstandard SVM; the black points are data points with the true $p \geq 0.9$; the green points are data points with the true $p < 0.9$.

It is worth mentioning that our proposed penalized likelihood method estimates the logit function over the domain of the observed x's, hence it is able to give all level curves of the logit (and thus p) simultaneously, while SVM classifiers are targeted at one level curve at a time, i.e., they are meaningful only for the classification boundary. This point may be understood via Figure 1 of Lee, Lin

and Wahba (2004). SVM classifiers come into their own when the classes are (nearly) separable, but in our application that is not the case. In Figure 2, we show further results from our 2-D example. The color bar beside plot(b) codes the relative importance $w_s + w_b$ (in log scale) for each data point, and the estimated p for the 200 data points in the test set is plotted against the true p. (The 200 points from the test set are particularly dense near 0 and 1.) We can see that the estimated probabilities (obtained from the estimated logit) match the true probability very well except for several points of very small importance that are colored in blue. Furthermore, by comparing the true and estimated level curve plots in (c) and (d), we conclude that we are estimating the true logit function very well over the domain of the data.

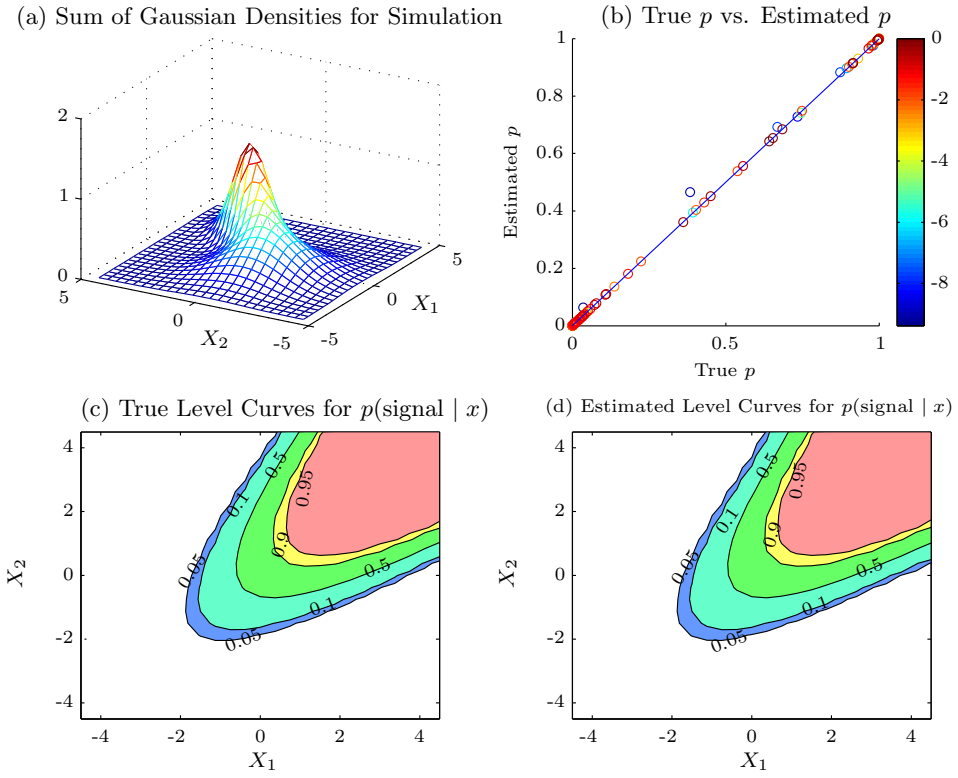

Figure 2. Results from the modified penalized likelihood method for a 2-D EXAMPLE: (a) sum of the two Gaussians used for weights; (b) p-p plot with the color bar coding the sum of the signal and background weights in log scale; (c) true level curves of p=0.05, 0.1, 0.5, 0.9 and 0.95; (d) estimated level curves of p=0.05, 0.1, 0.5, 0.9 and 0.95.

5. Results on Simulated Neutrino Data

5.1. Logit function estimation

The simulated neutrino data consists of five variables together with weights (Ahrens et al. (2004)). The variables are based on parameters derived from a maximum likelihood reconstruction of the particle track in the detector. For detailed information on the detector, reconstruction, variables and analysis procedure, see Ahrens et al. (2004). We had 10,000 simulated neutrino events, and divided them into 40% training, 40% tuning and 20% test sets. The five variables are first rescaled using their own sample weighted standard deviation after a log transformation. Then, we ran our algorithm with the transformed data. We transformed them back when plotting the results. The estimated logit function enables us to estimate the probability of an event being a signal neutrino at any point (within the domain of the data) in the 5-D observational space. We show a level curve plot of a 2-D cross section of that 5-D space in Figure 3, where the other three variables have been fixed at their sample medians.

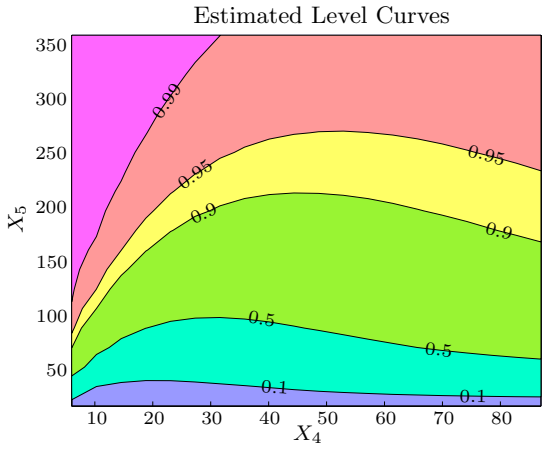

Figure 3. Level curves: $p = 0.1, 0.5, 0.9, 0.95$ and 0.99 on a 2-D cross section in the 5-D observational space of the neutrino data. $N = 500$ basis functions were used.

5.2. Checking the goodness of the estimate

Astrophysicists use very complex computer programs to simulate the observations of neutrinos passing through the AMANDA detector. Even though the parameters related to the simulation are known, the probability of an event being a signal neutrino in the 5-D space is only estimated at the data points where the simulation took place. We can't construct a plot as in Figure 2(b) since we don't

know the truth. We can only check whether the estimates reasonably reflect the simulated information.

We try to evaluate the estimated probability surface using a method in common practice among physicists. We take all the x's whose estimated probability of signal falls into one of ten equally spaced bins from 0 to 1. Then, for each bin, we calculate the ratio of the sum of all the signal weights of the x's to the sum of all the signal + background weights of those x's. Call this the level-binned observed p. We plot the result for the jth bin against the midpoint of the bin, i.e., for the bin [0.4, 0.5], we plot the level-binned observed p against 0.45, and similarly for the other bins. If we estimate the probability reasonably well, based on the simulated, weighted data, we should have these ten points falling close to the 45-degree line, which is what we observe in Figure 4 for our neutrino data. The color coding of the points represents the relative signal + background weights of the x's in each bin, on a log scale.

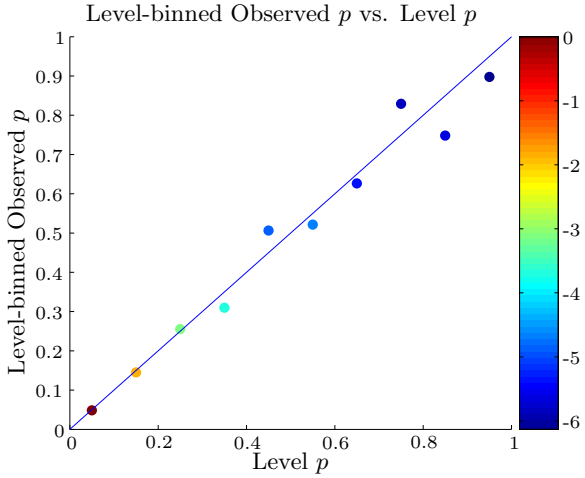

Figure 4. x-axis: Midpoints of 10 probability bins. y-axis: ratio of the sum of signal weights to the sum of signal + background weights for data points in each bin.

6. Conclusions

We have developed a feasible and effective computational method to obtain modified penalized likelihood estimates for signal detection probability in the context of five dimensional data, as might be extracted from tracks observed by the AMANDA neutrino detector, where the data simulator employs importance sampling. We implemented the proposed method on a simulated neutrino data set before describing a way to check the goodness of our estimation. We have also compared the penalized likelihood method to the nonstandard support

vector machine, similarly tuned, on a simulated multi-dimensional problems similar to the AMANDA problem, which can be characterized as having a certain amount of overlap between the signal and background distributions. In examples of that nature, with significant overlap of the signal and background, the penalized likelihood method is competitive to the comparable nonstandard SVM for classification purposes.

Acknowledgement

This research was supported by the National Science Foundation under Grants DMS-0072292 (G. Wahba and F. Lu) and OPP-9980474 (G. C. Hill and P. Desiati).

References

Ahrens, J. and other 112 coauthors (AMANDA collaboration) (2003). Limits on diffuse fluxes of high energy extraterrestrial neutrinos with the AMANDA-B10 detector. *Phys. Rev. Lett.* **90**, 251101.

Ahrens, J. and other 119 coauthors (AMANDA collaboration) (2004). Muon track reconstruction and data selection techniques in AMANDA *Nuclear Inst. and Methods in Physics Research* **A524**, 169. Reconstruction algorithms are described in Section 3, and the event variables are detailed in Section 6.2.

Andrés, E. and other 119 coauthors (2001). Observation of high-energy neutrinos using Cerenkov detectors embedded deep in Antarctic ice. *Nature* **410**, 441.

Aronszajn, N. (1950). Theory of reproducing kernels. *Trans. Amer. Math. Soc.* **68**, 337-404.

Cox, D. and O'Sullivan, F. (1990). Asymptotic analysis of penalized likelihood and related estimators. *Ann. Statist.* **18**, 1676-1695.

Feldman, G. J. and Cousins, R. D. (1998). Unified approach to the classical statistical analysis of small signals. *Phys. Rev. D* **57**, 3873.

Hill, G. C., Lu, F., Desiati, P. and Wahba, G. (2003). Optimizing the limit setting potential of a multivariate analysis using the Bayes posterior ratio. *Proceedings of PHYSTAT*2003 *at SLAC*, November.

Hill, G. C. and Rawlins, K. (2003). Unbiased cut selection for optimal upper limits in neutrino detectors: the model rejection potential technique. *Astropart. Phys.* **19**, 393.

Kimeldorf, G. and Wahba, G. (1971). Some results on Tchebycheffian spline functions. *J. Math. Anal. Appl.* **33**, 82-95.

Lagarias, J. C., Reeds, J. A., Wright, M. H. and Wright, P. E. (1998). Properties of the Nelder-Mead Simplex Method in Low Dimensions. *SIAM J. Optim.* **9**, 112-147.

Lee, Y., Lin, Y. and Wahba, G. (2004). Multicategory Vector Machines, Theory, and Application to the Classification of Microarray Data and Satellite Radiance Data. *J. Amer. Statist. Assoc.* **99**, 67-81.

Lin, Y., Lee, Y. and G. Wahba (2002). Support Vector Machines for Classification in Nonstandard Situations. *Machine Learning* **46**, 191-202.

Ferris, M., Voelker, M. and Zhang, H. H. (2004). Model Building with Likelihood Basis Pursuit. *J. Optim. Methods Software* **19**, 577-594.

O'Sullivan, F., Yandell, B. and Raynor, W. (1986). Automatic smoothing of regression functions in generalized linear models. *J. Amer. Statist. Assoc.* **81**, 96-103.

Wahba, G. (1990). *Spline Models for Observational Data*. CBMS-NSF Regional Conference Series in Applied Mathematics **59**.

Wahba, G. (2002). Soft and Hard Classification by Reproducing Kernel Hilbert Space Methods. *Proc. Natl. Acad. Sci. USA* **99**, 16524-16530.

Wahba, G., Wang, Y., Gu, C., Klein, R. and Klein, B. (1995). Smoothing Spline ANOVA for Exponential Families, with Application to the Wisconsin Epidemiological Study of Diabetic Retinopathy. *Ann. Statist.* **23**, 1865-1895.

Department of Statistics, University of Wisconsin, Madison, U.S.A.

E-mail: flu@stat.wisc.edu

Space Science and Engineering Center, University of Wisconsin, Madison, U.S.A.

E-mail: ghill@icecube.wisc.edu

Department of Statistics, University of Wisconsin, Madison, U.S.A.

E-mail: wahba@stat.wisc.edu

Space Science and Engineering Center, University of Wisconsin, Madison, U.S.A.

E-mail: desiati@icecube.wisc.edu

(Received July 2005; accepted July 2005)