ELSEVIER

Computer Physics Communications 145 (2002) 351–356

Computer Physics
Communications

www.elsevier.com/locate/cpc

$\vec{\alpha}$PDE: A new multivariate technique for parameter estimation

B. Knuteson [a,*], H.E. Miettinen [b], L. Holmström [c]

[a] *Enrico Fermi Institute, University of Chicago, Chicago, IL 60637, USA*
[b] *Department of Physics and Astronomy, Rice University, Houston, TX 77251, USA*
[c] *Rolf Nevanlinna Institute, University of Helsinki, Finland*

Received 20 August 2001; received in revised form 10 February 2002

Abstract

We present $\vec{\alpha}$PDE, a new multivariate analysis technique for parameter estimation. The method is based on a direct construction of joint probability densities of known variables and the parameters to be estimated. We show how posterior densities and best-value estimates are then obtained for the parameters of interest by a straightforward manipulation of these densities. The method is essentially non-parametric and allows for an intuitive graphical interpretation. We illustrate the method by outlining how it can be used to estimate the mass of the top quark, and we explain how the method is applied to an ensemble of events containing background. © 2002 Published by Elsevier Science B.V.

PACS: 02.50.Sk; 02.50.Ph; 02.50.Rj

Keywords: Parameter estimation; Density estimation; Multivariate

1. Introduction

In an earlier paper [1] we introduced the PDE (Probability Density Estimation) method, an essentially non-parametric and multivariate method designed for identifying small signals among large backgrounds. The method makes use of kernel density estimates for signal and background probability densities, and a simple discriminant function is then used to classify candidate events. The PDE method was applied successfully to the search for the top quark at the Fermilab Tevatron, and it is an integral part of a general search strategy [2] for analyzing data from high-energy physics experiments.

In this paper we present $\vec{\alpha}$PDE, an extension of the PDE method designed for parameter estimation, where $\vec{\alpha}$ represents a vector of parameters to be estimated. In many applications $\vec{\alpha}$ is a single parameter, such as the mass of an unstable particle detected through its decay products. This non-parametric and multivariate method may be particularly applicable to problems such as determining the mass of the top quark in the upcoming collider run (Run II) of the Fermilab Tevatron.

Multivariate methods are now widely recognized as being more powerful than univariate methods, and a non-parametric method has the advantage that one need not make assumptions about the forms of probability distributions. Those who feel uneasy about the "black-box" quality of neural networks should welcome the straightforward manipulation of probability

[*] Corresponding author.
E-mail address: knuteson@fnal.gov (B. Knuteson).

0010-4655/02/$ – see front matter © 2002 Published by Elsevier Science B.V.
PII: S0010-4655(02)00279-5

densities used in this method, and the intuitive graphical interpretation that results. Because probability densities are constructed and manipulated directly, obtaining any additional statistical information—Bayesian credible intervals, for example—is a straightforward exercise.

A typical parameter estimation problem is described in Section 2; our recipe for solving it is provided in Section 3. The salient features of the method and its potential advantages are summarized in Section 4.

2. The problem

The next decade of high energy collider physics will emphasize measurements and searches for new phenomena at the scale of several hundred GeV. The existence of a new particle at this scale can be convincingly demonstrated by observing a peak in an invariant mass distribution, but the signature may be such that more indirect methods of establishing the particle's existence, and subsequently measuring parameters such as its mass and couplings, are required. We introduce $\vec{\alpha}$PDE with an example of this nature: the determination of the top quark mass. Top quarks are pair-produced at the Fermilab Tevatron, each decaying promptly to a W boson and a b quark. Each W boson in turn decays either to a charged lepton and a neutrino, or to two quarks. Quarks hadronize, appearing in the detector as collimated flows of energy (jets). The characteristic experimental signature for a top quark event is therefore a final state containing either an energetic lepton, missing transverse energy, and several energetic jets, or a final state containing two energetic leptons, missing transverse energy, and a pair of jets; decays to six jets are difficult to distinguish from events in which no top quark was produced. The application of selection criteria favoring events with jets originating from b quarks enhances the fraction of top quark events in the sample.

For the sake of simplicity we assume that two variables $\vec{x} = (x, y)$ have been identified for this analysis. This pair might be the transverse energies of the lepton and the leading jet; it might be the invariant mass of the sub-leading jets and the transverse momentum of the W boson; it might be the scalar sum of all jet transverse energies and the output of a neural network built with event-shape variables. No special assumptions about the nature of these variables need be made.

3. The recipe

The goal is to construct a method that performs as well as (or better than) such popular algorithms as neural networks, but to keep the method sufficiently simple that it reads like a recipe. The recipe follows.

3.1. *Specify $p(m)$*

This method has its roots in Bayesian statistics, and as a result it has the advantage (disadvantage) of enabling (requiring) the specification of a function $p(m|\mathcal{I})$ that encodes prior beliefs about the value of the top quark mass m. $\mathcal{I}$ here is used in standard Bayesian notation to represent all assumptions implicit in our specification of this prior probability. The basic assumptions contained in $\mathcal{I}$ will not change, so we drop it from here on, writing simply $p(m)$. A natural choice for $p(m)$, used when there is strong belief that the true mass must lie somewhere between a and b but no reason to prefer any value within that range over any other, is the flat prior: $p(m) = \frac{1}{b-a}$ for $a < m < b$, and 0 elsewhere.

3.2. *Generate Monte Carlo events*

Monte Carlo events are generated with top quark masses m pulled from the distribution $p(m)$ specified above. That is, the probability that an event with a top quark mass between m and $m + \delta m$ is generated is $p(m)\delta m$. For each Monte Carlo event we calculate the two variables $\vec{x} = (x, y)$.

A histogram in (x, y, m) filled with the generated events approximates the joint density $p(x, y, m)$. This function has the property that, given an event in which a top quark is produced and decays to the observed final state, the probability that the top quark mass was between m and $m + \delta m$, the first variable between x and $x + \delta x$, and the second variable between y and $y + \delta y$, is simply $p(x, y, m)\delta x\, \delta y\, \delta m$.

3.3. *Construct a training array T*

Each of the N Monte Carlo events just generated is characterized by three numbers: the value of x, the value of y, and the top quark mass m. The Monte Carlo events are labeled with the index i ($i = 1, \ldots, N$); the three numbers corresponding to the ith event are then x_i, y_i, and m_i. Define the *event vector* $\vec{v}_i$ for the ith Monte Carlo event by

$$\vec{v}_i = (\vec{x}_i, m_i), \tag{1}$$

and define the *training array* T for the entire set of Monte Carlo events by

$$T_{ij} = (\vec{v}_i)_j. \tag{2}$$

Here and below i ranges from 1 to N and indexes the Monte Carlo events; j ranges from 1 to 3 and indexes the components of the event vector $\vec{v}$.

3.4. Calculate the covariance matrix

Having defined the event vector $\vec{v}$, calculate the *mean event vector*

$$\langle\vec{v}\rangle = \frac{1}{N}\sum_{i=1}^{N}\vec{v}_i \tag{3}$$

and construct the *training covariance matrix*

$$\Sigma_{kl} = \frac{1}{N}\sum_{i=1}^{N}\bigl((\vec{v}_i)_k - \langle\vec{v}\rangle_k\bigr)\bigl((\vec{v}_i)_l - \langle\vec{v}\rangle_l\bigr) \tag{4}$$

in the standard way. Σ is a 3 by 3 symmetric matrix, with $\Sigma_{12} = \mathrm{Cov}(x, y)$, $\Sigma_{13} = \mathrm{Cov}(x, m)$, and so on.

3.5. Estimate the joint density $p(\vec{v})$

In Section 3.2 we imagined filling a three-dimensional histogram in $\vec{v}$ with Monte Carlo events, and we recognized that the resulting histogram represents an estimation of a probability density. A well-known technique in multivariate statistics involves estimating a probability density not by filling a histogram, but rather by summing kernels of probability placed around each point. Due to its familiarity and smoothness properties, a favorite kernel choice is the multivariate Gaussian:

$$K(\vec{v}) = \frac{1}{(\sqrt{2\pi}h)^3\sqrt{\det(\Sigma)}}\exp\left(\frac{-\vec{v}^{\mathrm{T}}\Sigma^{-1}\vec{v}}{2h^2}\right). \tag{5}$$

The vector $\vec{v}$ is the same three-component vector defined above, and Σ^{-1} is the inverse of the training array covariance matrix Σ. The parameter h is known in the language of density estimation as a *smoothing parameter*; it controls the width of the kernels placed around each point. Theoretical arguments suggest an optimal choice of $h \approx N^{-1/(d+4)}$ as a function of the number of data points N and the dimensionality d of the variable space.[1]

An estimate of the joint probability density $p(\vec{v})$ is then obtained simply by summing kernels centered about each of the N data points $\vec{v}_i$, so that

$$p(\vec{v}) = \frac{1}{N}\sum_{i=1}^{N}K(\vec{v} - \vec{v}_i). \tag{6}$$

3.6. Compute the posterior density $p(m|\vec{x})$

A physicist attempting to measure the top quark mass is interested in the *posterior density* $p(m|\vec{x})$ for m. In words, $p(m|\vec{x})$ is the probability that the top quark mass is m given that we have observed an event with variable values $\vec{x}$. This posterior density is easily obtained. The probability of obtaining both $\vec{x}$ and m is equal to the probability of obtaining $\vec{x}$ multiplied by the probability of obtaining m given that you have obtained $\vec{x}$:

$$p(\vec{x}, m) = p(\vec{x})p(m|\vec{x}), \tag{7}$$

and the probability of obtaining $\vec{x}$ is given by integrating the probability of obtaining both $\vec{x}$ and m over all values of m:

$$p(\vec{x}) = \int p(\vec{x}, m')\,\mathrm{d}m'. \tag{8}$$

Thus the posterior density $p(m|\vec{x})$ is related to the joint density $p(\vec{x}, m)$ simply by

$$p(m|\vec{x}) = \frac{p(\vec{x}, m)}{\int p(\vec{x}, m')\,\mathrm{d}m'}. \tag{9}$$

3.7. Compute $\hat{m}$ (optional)

In the Bayesian view, the posterior density is the natural result of this recipe. Nonetheless, it is often

[1] This expression for h depends on assumptions about the probability density that we have not made explicit, and is not exact [3,4]. In practice, h may be optimized for any set of Monte Carlo events by constructing and minimizing some appropriate error estimate $\chi(h)$. For $N = 10^5$ and $d = 3$, the optimal choice for h is roughly 0.20.

convenient to reduce the posterior density $p(m|\vec{x})$ to a single number $\hat{m}$ representing the *best estimate* of the parameter in question. Among the natural choices for the best estimate are the mean, median, and mode of the posterior distribution. Adopting the last for the purposes of this discussion, we solve the equation

$$p(\hat{m}|\vec{x}) = \max_m p(m|\vec{x}) \tag{10}$$

numerically for $\hat{m}$. Since the denominator of Eq. (9) is independent of m, maximizing the posterior density $p(m|\vec{x})$ is equivalent to maximizing the joint density $p(\vec{x}, m)$, which we have constructed explicitly. The extent to which the posterior density $p(m|\vec{x})$ peaks around the value $\hat{m}$ depends, of course, on how strongly the variables $\vec{x}$ correlate with the true mass m.

We note that this method can be modified to produce results that obey the frequentist notion of coverage. Assume that a 68% confidence region is desired. Starting with $p(\vec{x}|m)$, draw for each fixed m the contour $\mathcal{C}_m$ in $\vec{x}$-space enclosing 68% of the density and minimal area. Then upon observing $\vec{x}$ in the data, the 68% confidence region for m is the union of all values of m for which $\vec{x}$ lies inside $\mathcal{C}_m$.

Extension to the case of an ensemble of data events is treated in Appendix C.

4. Conclusions

The analysis method described here is quite general, and can be used in the context of any parameter estimation problem. The non-parametric approach used to estimate probability densities is helpful when the distributions under consideration do not lend themselves to an obvious parameterization. $\vec{\alpha}$PDE allows the use of several measured variables, and enables the simultaneous estimation of several parameters. The generalization to arbitrary dimension is provided in Appendix A. Bayesian credible intervals and moments are easily obtained from simple manipulations of the joint probability density.

Appendix A. The general multivariate case

For pedagogical reasons, $\vec{\alpha}$PDE has been introduced through a specific example—determining the mass m of the top quark from two measured quantities x and y—and the expressions in the text are therefore specific to that example. In this appendix we provide the formulae for the general case.

In the general case, let each event be characterized by d_1 known variables $\vec{x}$ and d_2 unknown parameters $\vec{\alpha}$. Let $d = d_1 + d_2$, and let the d-dimensional event vector be $\vec{v} = (\vec{x}, \vec{\alpha})$.

The ith Monte Carlo event is now described by the event vector

$$\vec{v}_i = (\vec{x}_i, \vec{\alpha}_i), \tag{A.1}$$

and the entire Monte Carlo sample is described by the training array

$$T_{ij} = (\vec{v}_i)_j, \tag{A.2}$$

where j now ranges from 1 to d. The mean event vector is

$$\langle\vec{v}\rangle = \frac{1}{N}\sum_{i=1}^{N}\vec{v}_i \tag{A.3}$$

and the training covariance matrix is

$$\Sigma_{kl} = \frac{1}{N}\sum_{i=1}^{N}\big((\vec{v}_i)_k - \langle\vec{v}\rangle_k\big)\big((\vec{v}_i)_l - \langle\vec{v}\rangle_l\big), \tag{A.4}$$

as before, and the general multivariate Gaussian is given by

$$K(\vec{v}) = \frac{1}{(\sqrt{2\pi}h)^d\sqrt{\det(\Sigma)}}\exp\left(\frac{-\vec{v}^{\mathrm{T}}\Sigma^{-1}\vec{v}}{2h^2}\right). \tag{A.5}$$

Finally, in Eqs. (9) and (10), m should be replaced by the vector $\vec{\alpha}$.

In practice, limited computing resources place an upper bound on N, and hence an upper bound on d. The optimal accuracy of the kernel estimate is of order $N^{-s/(2s+d)}$, where s is a positive integer that reflects the assumed smoothness of the unknown density, and a typical assumption of continuous and square integrable derivatives up to second order corresponds to $s = 2$ [3,4].

Appendix B. Alternative to generating a random sample of Monte Carlo events

In this appendix we describe a modification to the procedure described in the text if practical constraints prevent the generation of events pulled from a continuous prior $p(m)$, but allow the generation of events at q discrete values m_j, where $j = 1, \ldots, q$.

Two changes are required in the first five steps of the recipe (Sections 3.1–3.5). First, it is assumed that practical constraints require Monte Carlo events to be generated at the discrete masses m_j, rather than as described in Section 3.2. Second, the function calculated in Eq. (6), which may no longer be interpreted as a joint density, should be re-labeled. For lack of a better alternative, call it $\xi(\vec{v})$.

We now add a step $5\frac{1}{2}$ between Sections 3.5 and 3.6. The function $\xi(\vec{v})$ is clearly not an appropriate density. If events have been generated assuming five different masses m_j, a graph of $\xi(\vec{v})$ might appear as shown in Fig. 1. We see that the density has ridges along the values of m for which events have been generated, with corresponding valleys in the regions between these values.

An appropriately rescaled probability density $p(\vec{x}, m)$ can be generated by multiplying $\xi(\vec{v})$ by a normalizing m-dependent factor $s(m)$:

$$p(\vec{x}, m) = \xi(\vec{x}, m)s(m). \tag{B.1}$$

This normalizing factor will correct for the fact that valleys have been introduced into the density by only generating events at specific masses m_j. The requirement that

$$\int p(\vec{x}, m)\, \mathrm{d}\vec{x} = p(m) \tag{B.2}$$

determines this normalizing factor uniquely. The desired joint probability density $p(\vec{v})$ is then given by

$$p(\vec{v}) = \frac{\xi(\vec{v})p(m)}{\int \mathrm{d}\vec{x}'\xi(\vec{x}', m)}, \tag{B.3}$$

and the final step (Section 3.6) is exactly as before. The rescaled density of Fig. 1 is shown in Fig. 2.

We mention briefly a useful shortcut when calculating integrals such as that appearing in the denominator of Eq. (B.3). Multidimensional integrals are difficult to calculate in general, but this integral can be handled analytically provided one uses Gaussian kernels. Assume as in Appendix A that the vector of known variables $\vec{x}$ is of d_1 dimensions, that the vector of unknown variables $\vec{\alpha}$ is of d_2 dimensions, and that the Monte Carlo has a covariance matrix Σ. Then the relevant formula is

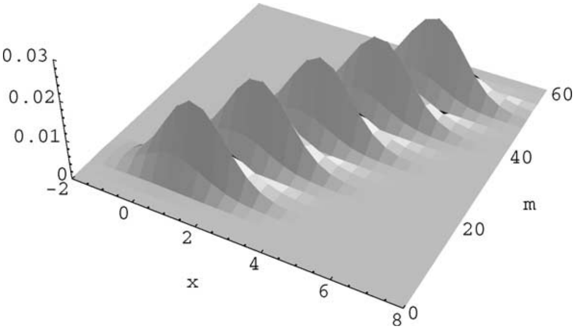

Fig. 1. A sample function $\xi(x, m)$ that might be constructed from Monte Carlo events at masses m = 10, 20, 30, 40, and 50. Notice the ridges in this function, due to the fact that it is constructed from events at specific masses.

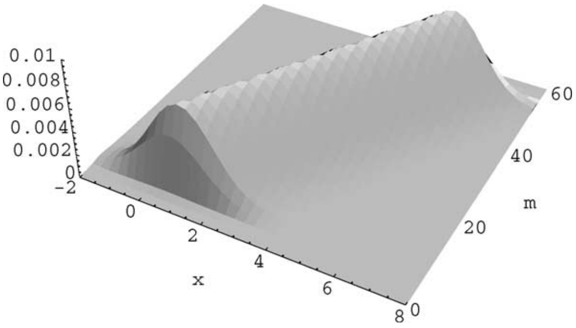

Fig. 2. The density $p(x, m)$ formed by rescaling the function $\xi(x, m)$ shown in Fig. 1. Notice how this rescaling corrects for the fact that only events at specific masses were used in the construction of $\xi(x, m)$.

$$\int K(\vec{x}, \vec{\alpha})\, \mathrm{d}\vec{x} = \frac{1}{(\sqrt{2\pi}h)^{d_2}\sqrt{\det(\Sigma')}} \exp\left(\frac{-\vec{\alpha}^{\mathrm{T}}\Sigma'^{-1}\vec{\alpha}}{2h^2}\right), \tag{B.4}$$

where Σ' is the d_2 by d_2 sub-matrix of Σ formed by retaining elements with row and column numbers larger than d_1.

Appendix C. Background events

In the text we considered the problem of determining the top quark mass m for one candidate event. In a real analysis there will be n such events, and of those some fraction b are expected to be background events—events that do not contain a top quark at all.

This appendix shows how to apply $\vec{\alpha}$PDE to a complete analysis.

Signal and background Monte Carlo events are generated and used to construct the signal density $p_s(\vec{x}, \vec{\alpha})$, as described in Sections 3.1–3.5, and the background density $p_b(\vec{x})$, which is independent of $\vec{\alpha}$. From a careful analysis of background efficiencies we determine the probability $p(b)$ that a fraction b of our events are background events. In previous sections of this article $\vec{x}_i$ referred to Monte Carlo events; in this section we change notation and label the n observed data events by $\vec{x}_1, \dots, \vec{x}_n$.

The goal is to compute the posterior density $p(\vec{\alpha}|\vec{x}_1, \dots, \vec{x}_n)$. Since the observations $\vec{x}_1, \dots, \vec{x}_n$ are assumed to be independent, $p(\vec{x}_1, \dots, \vec{x}_n|\vec{\alpha}, b)$ factors into a product:

$$p(\vec{x}_1, \dots, \vec{x}_n|\vec{\alpha}, b) = \prod_{i=1}^{n} p(\vec{x}_i|\vec{\alpha}, b). \tag{C.1}$$

The probability $p(\vec{x}_i|\vec{\alpha}, b)$ for the ith data event can be written in terms of the signal and background probability densities as

$$p(\vec{x}_i|\vec{\alpha}, b) = (1-b)p_s(\vec{x}_i|\vec{\alpha}) + bp_b(\vec{x}_i), \tag{C.2}$$

where $p(\vec{x}|\vec{\alpha}) = p(\vec{x}, \vec{\alpha})/p(\vec{\alpha})$. Integrating out the *nuisance parameter* b in Eq. (C.1) leaves

$$\begin{aligned} p(\vec{\alpha}|\vec{x}_1, \dots, \vec{x}_n) \\ = \mathcal{N} p(\vec{\alpha}) \int_0^1 \left(\prod_{i=1}^{n} \big[(1-b)p_s(\vec{x}_i|\vec{\alpha}) + bp_b(\vec{x}_i)\big] \right) \\ \times p(b)\, \mathrm{d}b, \end{aligned} \tag{C.3}$$

where $\mathcal{N}$ is a normalization factor ensuring that $\int p(\vec{\alpha}|\vec{x}_1, \dots, \vec{x}_n)\, \mathrm{d}\vec{\alpha} = 1$, and $p(\vec{\alpha}, b) = p(\vec{\alpha})p(b)$ is assumed.

The most likely values of the parameters $\vec{\alpha}$ are then those for which $p(\vec{\alpha}|\vec{x}_1, \dots, \vec{x}_n)$ achieves its maximum, and the uncertainty on these values can be estimated from the width of the peak. Other frequently-used best estimates and their errors are easily computed, if desired, from straightforward manipulation of the posterior density $p(\vec{\alpha}|\vec{x}_1, \dots, \vec{x}_n)$.

References

[1] L. Holmström, S.R. Sain, H.E. Miettinen, Comp. Phys. Comm. 88 (1995) 195.

[2] DØ Collaboration, V. Abazov et al., Phys. Rev. Lett. 87 (2001) 231801.

[3] D. Scott, Multivariate Density Estimation, John Wiley & Sons, 1992.

[4] M. Wand, M. Jones, Kernel Smoothing, Chapman & Hall, 1995.