Divergence decision tree classification with
Kolmogorov kernel smoothing in high energy physics

Vaclav Kus and Kristina Jaruskova
Department of Mathematics, FNSPE, Czech Technical University in Prague, Trojanova 13, 12000 Praha 2, Czech Republic
E-mail: vaclav.kus@fjfi.cvut.cz (corresponding author)

Abstract. The binary classification of a given dataset is a task of assigning one of the two possible classes to each observation. This can be achieved by many machine learning techniques, e.g. logistic regression, decision trees, neural networks. The supervised divergence decision tree (SDDT) is our own binary classification algorithm in favour of the Rényi divergence, which incorporates multi-dimensional kernel density estimates (KDEs) as the main part of the splitting process in its tree nodes. However, the KDE needs an efficient smoothing in order to obtain quite satisfactory classification results. In this paper, the D-discrepancy method for selecting the bandwidth was applied. It is based on an evaluation of divergences, or distances, between two estimated distributions. The Kolmogorov metric distance on probability space is used and the performance of such a novel technique is compared to standard smoothing techniques. The final goal is to perform a binary classification and achieve the best possible results with respect to the AUC value (area under ROC curve) on a given high energy physics (HEP) dataset, specifically for d+Au heavy ions decay data. This HEP dataset is described and the main structure of the used SDDT is outlined. Final classification results are presented for KDE under Kolmogorov D-method of smoothing in SDDT algorithm.

1. HEP dataset and SDDT classification
The main task is the binary classification on specific HEP dataset. This dataset contains Monte Carlo simulations of heavy-ion d+Au collisions. The two classes of interest were denoted as S (Signal) and B (Background). The Monte Carlo 7-dimensional data (variables 0-6) was split into train set with 714 577 observations and test set with 89 322 observations, the signal made up for approx. 42 % of all observations, as shown in Table 1. Example of histograms for the first three variables are shown in Figure 1; notice that the observations are concentrated close to zero.

| Set  | Signal | Background | Total   | Signal/Total |
|------|--------|------------|---------|--------------|
| Train| 301 227| 413 350    | 714 577 | 0.4215       |
| Test | 37 673 | 37 673     | 89 322  | 0.4218       |

Table 1. Total numbers of observations in train set and test set.

The SDDT algorithm is a supervised classification method developed in [1, 2]. The main idea here is to assign two separate nonparametric Kernel Density Estimates (KDEs) to the class
Signal and Background, respectively. Based on these kernel estimates of S and B probability density functions (PDFs), the posterior probability of class Signal for an observation \( \vec{t} \) is further computed using the Bayes formula

\[
p(S|\vec{t}) = \frac{P_S p(\vec{t}|S)}{P_S p(\vec{t}|S) + P_B p(\vec{t}|B)},
\]

where \( P_S = \frac{\# S}{\# Total} \) and \( P_B = \frac{\# B}{\# Total} \) are the prior probabilities of classes S and B, estimated from the train dataset. The decision rule for assigning a class to an observation is then \( p(S|\vec{t}) \geq \delta \), where \( \delta \) is a cut-off point determined as the point that maximizes the Youden index of ROC curve [3].

2. Kernel Density Estimate (KDE)

Let \( \vec{X}_1, \vec{X}_2, \ldots, \vec{X}_n \) denote \( d \)-dimensional random variables (observations from train sample). Then the kernel density estimate of the true PDF at a point \( \vec{t} \in \mathbb{R}^d \) is [2]

\[
\hat{f}(\vec{t}) = \frac{1}{nh^d} \sum_{j=1}^{n} K \left( \frac{\vec{t} - \vec{X}_j}{h} \right) = \frac{1}{nh^d \sqrt{\det S}} \sum_{j=1}^{n} k \left( \frac{\vec{t} - \vec{X}_j}{h} \right)^T S^{-1} \left( \frac{\vec{t} - \vec{X}_j}{h} \right),
\]

where the second term refers to the so-called sphering, with \( k(u^T \vec{u}) = K(\vec{u}) \), and sample covariance matrix \( S \). We have to choose the \textit{bandwidth} \( h \in \mathbb{R}^+ \) and \textit{multidimensional kernel} \( K : \mathbb{R}^d \to \mathbb{R}^+ \). Instead of fully multivariate kernel \( K \), for example \( d \)-dimensional Epanechnikov kernel, we use a \textit{product kernel} which gives the PDF estimate

\[
\hat{f}(\vec{t}) = \frac{1}{nh_1 h_2 \cdots h_d} \sum_{j=1}^{n} \prod_{i=1}^{d} K \left( \frac{t_i - X_{ij}}{h_i} \right),
\]

where \( K \) is a one-dimensional kernel function, e.g. standard one-dimensional Epanechnikov kernel \( K(t) = K_{1d}(t) = \frac{3}{4\sqrt{5}} (1 - \frac{1}{5}t^2) \) for all \( |t| < \sqrt{5} \).

Choice of the bandwidth parameter \( h \) is crucial for the shape of the resulting KDE. If \( h \) is too large, it leads to an oversmoothing, choosing \( h \) too small results in overfitting of the KDE. For the univariate case, parameter \( h \) can be chosen according to one of the so-called \textit{rules-of-thumb}, e.g. widely used Silverman’s rule [4] \( h = 0.9 \cdot \min \left\{ s_n, \frac{IQR}{1.34} \right\} n^{-\frac{1}{4}} \), with the sample variance.

![Figure 1. Histograms for the variables 0, 1, and 2.](image-url)
$s_n^2$ and sample interquartile range $IQR$, which works well for densities in the neighbourhood of Gaussian distribution. However this is not our case, thus we select the bandwidth through the D-method (discrepancy method) for each dimension in the case of product kernel. This data dependent method for selecting the bandwidth uses general distance or divergence [5], e.g. the Kolmogorov distance [6], between the empirical cumulative distribution function (ECDF) and the kernel estimate of the true cumulative distribution function. The optimal bandwidth $h$ is determined as a solution of the equation

$$\sqrt{n} \cdot \hat{D}_n(h) = \sqrt{n} \cdot \max \left\{ \hat{D}^+_n(h), \hat{D}^-_n(h) \right\} = \beta,$$

where we denote

$$\hat{D}^+_n = \max_{1 \leq i \leq n} \left( \frac{i}{n} - \hat{F}_n(X(i)) \right), \quad \hat{D}^-_n = \max_{1 \leq i \leq n} \left( \hat{F}_n(X(i)) - \frac{i - 1}{n} \right).$$

Our recommended value for the right side of the equation is $\beta = 0.7$, although we may find $\beta = 0.5$ in the literature as well [7]. Both presented formulas for bandwidth selection apply for the one-dimensional KDE. However, they can be easily extended to multidimensional case by using the product kernel and choosing the parameter $h$ for each dimension separately. Figure 2 depicts multiplicative KDE of the two model distributions with the bandwidth parameters $h$ chosen by the D-method.

![Figure 2](image_url)

**Figure 2.** Kernel estimates of two-dimensional normal and exponential distribution with $h$ determined according to D-method with $\beta = 0.7$.

### 3. Numerical results for HEP d+Au dataset

The KDE with Epanechnikov kernel gives better results when the task is to estimate a smooth 'normal-like' density without sharp edges. Unfortunately, the distributions of the individual variables in the HEP dataset do have sharp edges and look more like the exponential distribution. Thus, an appropriate data transformation was performed to modify the data and to obtain more suitable shape of distributions. Data were transformed by functions stated in Table 2 and then the values of each variable were scaled so that the 0.05-quantile lies at 0 and the 0.95-quantile is located at 1. Histograms of the variables 0, 1, 2 after transformations can be seen in Figure 3.

The classification by the SDDT was finally carried out on both the original dataset and the dataset of transformed variables. The results of the classification are represented by the area
Variables | Transformation  
---|---  
0, 1, 2, 5 | \( y = \log x \)  
4, 6 | \( y = \log (-x - \min (-x) + 0.001) \)  
3 | \( y = \frac{2}{\pi} \cdot \arctan x \)  

Table 2. Specific transformations selected for physical HEP variables.

under ROC curve (AUC). The first/second modifications of the SDDT uses KDE under fully multivariate Epanechnikov kernel and the bandwidths \( h \) computed using the Silverman’s rule with/without sphering. The third and the fourth options, the product kernel with univariate Epanechnikov kernels were implemented, the parameter \( h \) is determined for each dimension separately by the D-method (4) with \( \beta = 0.5 \) and \( \beta = 0.7 \). Moreover, the principle component analysis (PCA) was optionally chosen to be performed as the first step of the splitting sequence in each node.

4. Evaluation and conclusions

It can be observed that the use of PCA with standardization in every node improves the classification in most of the evaluated cases. When performing the classification of the original dataset while not using the PCA, the best results were achieved with the modification using the multivariate Epanechnikov kernel under sphering. The D-method in this case chooses the parameter \( h \) significantly smaller than the Silverman’s rule which leads to an overfitting of the KDEs and the SDDT as a whole.

In all other cases, the more sophisticated choice of the bandwidth \( h \) using D-method improved the classification. In addition to that, the number of nodes and levels of the resulting tree is lower than for the other modifications of the SDDT, which implies faster classification. The highest achieved AUC for the test set was 0.9427 in the case of the SDDT classifier applied to transformed data, where PCA was used in every node and the bandwidth \( h \) was derived from the D-method equation with \( \beta = 0.5 \). Moreover, we observe only a slight and tolerable overfitting. In general, the data transformations which fits the KDE assignments, improve the classification by at least 2% with respect to the AUC.

Acknowledgments

These results were supported by the research grants LTT180001, LM2018068, CAAS EF16_019/0000778, and SGS18/188/OHK4/3T/14.
Table 3. Results of SDDT classification: AUC for train set, AUC for test set, total number of nodes of the trained tree and total number of levels of the tree.

| KDE                  | \(AUC_{\text{train}}\) | \(AUC_{\text{test}}\) | Nodes | Levels |
|----------------------|-------------------------|------------------------|-------|--------|
| **Original data**    |                         |                        |       |        |
| With sphering        | 0.9373                  | 0.9229                 | 118   | 9      |
| Without sphering     | 0.8609                  | 0.8615                 | 422   | 60     |
| D-method (\(\beta = 0.5\)) | 0.9932              | 0.9075                 | 26    | 5      |
| D-method (\(\beta = 0.7\)) | 0.9927              | 0.9149                 | 42    | 7      |
| **Original data, PCA** |                         |                        |       |        |
| With sphering        | 0.9048                  | 0.9036                 | 170   | 10     |
| Without sphering     | 0.8956                  | 0.8948                 | 640   | 60     |
| D-method (\(\beta = 0.5\)) | 0.9697              | 0.9324                 | 76    | 7      |
| D-method (\(\beta = 0.7\)) | 0.9581              | 0.9309                 | 94    | 8      |
| **Transformed data** |                         |                        |       |        |
| With sphering        | 0.9305                  | 0.9291                 | 142   | 9      |
| Without sphering     | 0.8727                  | 0.8723                 | 404   | 60     |
| D-method (\(\beta = 0.5\)) | 0.9719              | 0.9422                 | 70    | 8      |
| D-method (\(\beta = 0.7\)) | 0.9589              | 0.9371                 | 92    | 8      |
| **Transformed data, PCA** |                         |                        |       |        |
| With sphering        | 0.9225                  | 0.9214                 | 248   | 60     |
| Without sphering     | 0.9134                  | 0.9122                 | 654   | 60     |
| D-method (\(\beta = 0.5\)) | 0.9631              | 0.9427                 | 82    | 7      |
| D-method (\(\beta = 0.7\)) | 0.9510              | 0.9377                 | 118   | 7      |

References

[1] Bour P, Kus V, Franc J 2016 Statistical classification techniques in high energy physics (SDDT algorithm)  *J. Phys.: Conf. Series* **738** (012034)
[2] Bour P, Kus V, Franc J 2017 Kernel and divergence techniques in high energy physics separations  *Journal of Physics: Conference Series* **898** (072004)
[3] Unal I 2017 Defining an Optimal Cut-Point Value in ROC Analysis: An Alternative Approach  *Computational and Mathematical Methods in Medicine* [online] [cit. 2018-05-08] DOI: 10.1155/2017/3762651
[4] Silverman BW 1998 *Density estimation for statistics and data analysis* Boca Raton: Chapman & Hall/CRC
[5] Kus V, Morales D and Vajda I 2008 Extensions of the parametric families of divergences used in statistical inference  *Kybernetika* **44**, 95-112
[6] Hrab skov J, Kus V 2017 Notes on consistency of some minimum distance estimators with simulation results  *Metrika* **80** (2), 243-257
[7] Markovich N 2007 *Nonparametric Analysis of Univariate Heavy-Tailed Data* John Wiley & Sons, Chichester UK, 61-97