Journal of Physics: Conference Series **219** (2010) 032057 doi:10.1088/1742-6596/219/3/032057

The Toolkit for Multivariate Data Analysis, TMVA 4

P Speckmayer[1], A Höcker[1], J Stelzer[2] and H Voss[3]

[1] CERN, Geneva, Switzerland
[2] DESY, Hamburg, Germany
[3] Max-Planck-Institut fuer Kernphysik, Heidelberg, Germany

E-mail: peter.speckmayer@cern.ch

Abstract. The toolkit for multivariate analysis, TMVA, provides a large set of advanced multivariate analysis techniques for signal/background classification. In addition, TMVA now also contains regression analysis, all embedded in a framework capable of handling the pre-processing of the data and the evaluation of the output, thus allowing a simple and convenient use of multivariate techniques. The analysis techniques implemented in TMVA can be invoked easily and the direct comparison of their performance allows the user to choose the most appropriate for a particular data analysis. This article gives an overview of the TMVA package and presents recently developed features.

1. Introduction

The TMVA project[1, 2] has been started in 2005 and has become since then a mature software package for multivariate data analysis[3, 4, 5]. A recent change in the underlying framework allows for several new applications such as multivariate regression, generic boosting, multi-class classification, cross-validation etc. The first new features, such as regression and generic boosting have been implemented and will be made available to the users in June 2009. In this article an overview of the workflow in TMVA will be given. TMVA relies on the ROOT analysis framework[6] and a recent stable release is included into the ROOT package. The newest TMVA versions are available from `tmva.sourceforge.net`.

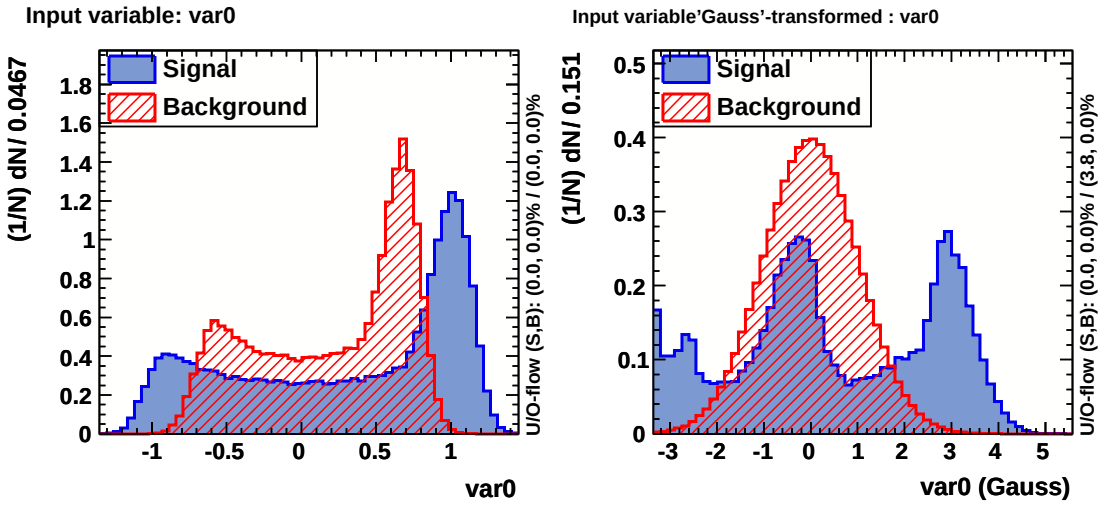
2. Preprocessing

Before performing classification or regression, the data have to be read in into TMVA. TMVA offers several possiblities for doing this task conveniently for the user. Selection cuts and event-weights can be applied if there is the need. For each chosen classification/regression-method a transformation (normalisation, decorrelation, principal components analysis, gaussianisation) of the input and target variables can be defined.

2.1. Events

Data can be read into TMVA from ROOT TTrees or from ASCII files. The user can define as input variables any variable of the tree or mathematical formulas combining one or more of them[1]. For better readability a label can be defined which is used in the TMVA printout and furthermore a title and a unit for the variable which will be used in the evaluation plots.

[1] see TTree::Draw in the ROOT documentation for constraints on formulas



(a) Input variable distributions for signal and background

(b) Variable distribution after application of the gaussianisation transformation

Figure 1: Variable distribution of signal and background before (a) and after (b) the application of the gaussianisation transformation. As reference class, the background has been chosen. The background signal has therefore a gaussian shape.

In the case of a signal/background classification, the class (signal or background) of each tree, which is read in, can be defined. Alternatively the class of the events can be defined by applying cuts.

In the case of regression, at least one *target value* has to be defined apart from the input variables. The target variables are defined in the same way as the input variables.

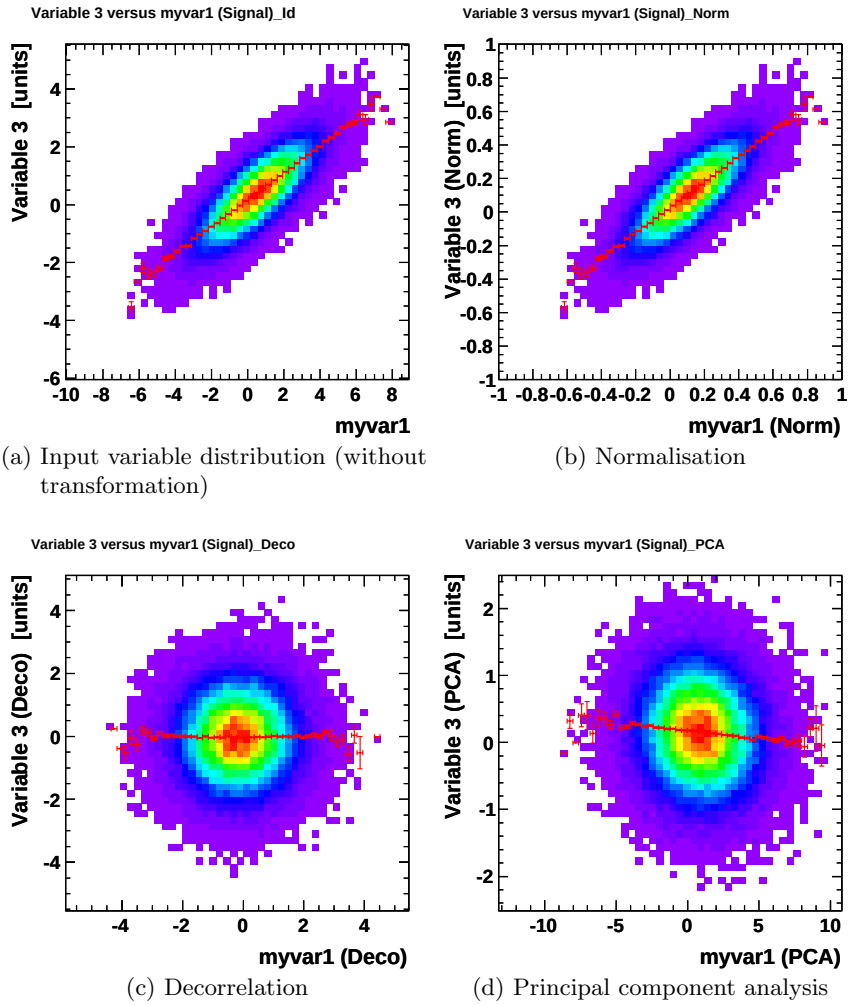
2.2. Selection cuts and event-weights

Selection cuts can be applied on the tree and on the class (signal/background) level. As for the input variables formulas can be used to define the cuts. In the same way event weights can be set. In this this way the actual number of weights can be adjusted to the cross sections the signal and background events (possibly different background compontents) are expected to have in measured data.

2.3. Variable transformations

TMVA provides several mathematical transformations which can be applied to the input variables (and regression targets) alone or in combination with others (e.g. decorrelation and normalisation). For each transformation, the user can define the event class (signal, background or both; for regression typically only one event class is defined) which is taken as the basis for the calculation of the transformation.
The following transformations are available in TMVA:

- Normalisation: The variable values are normalised to the interval $[-1, 1]$ (see figure **??**).
- Principal Component Analysis (PCA) (see figure 2d).
- Decorrelation: Linear correlations are taken into account through computing the square-root of the covariance matrix (see figure 2c).



(a) Input variable distribution (without transformation)

(b) Normalisation

(c) Decorrelation

(d) Principal component analysis

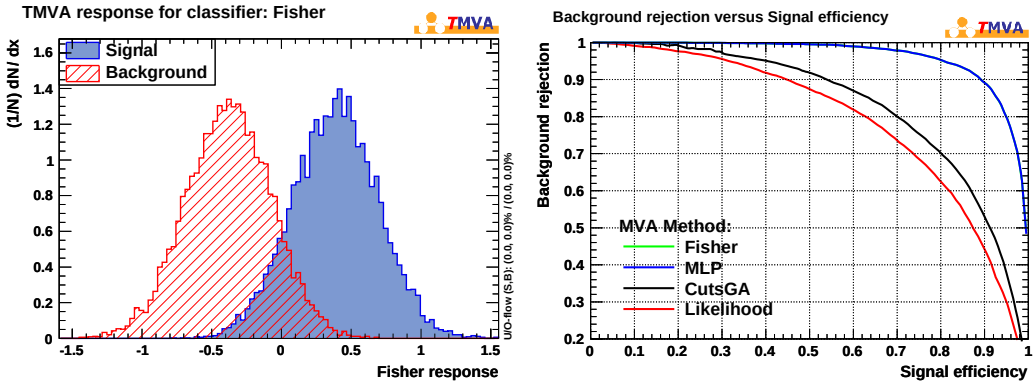
Figure 2: Plots of the input variable distributions of the signal (a). Figure **??** shows the distribution after being normalised, figure (c) after decorrelation transformation and figure (d) after a principal component analysis transformation.

- Gaussianisation: The events of the chosen class will be distributed according to a Gaussian. It is not possible (for mathematical reasons) to transform (non-gaussian) signal and background classes simultanously to gaussian distributions. But a transformation of both (signal+background) to a gaussian distribution is possible (see figure 1b).

In the case of regression, a back-transformation of the targets is automatically performed. The back-transformation is implemented up to now only for the normalisation of the variables.

3. Classification

In the high energy and nuclear physics, the signal which is searched for, for example a signature of a Higgs boson, is typically overlayed by background processes with a similar signature. Commonly used methods of classification into signal and background events reach their limitations when the signal is very small and/or part of the information of if an event is signal or background is hidden in not well known correlations between the observables. The automated multivariate analysis toolkit TMVA provides the ability to exploit the available



(a) classifier output y (b) ROC curve

Figure 3: For each method a classification output y is computed. Classification outputs close to 0 (1) denote the event being background-like (signal-like). Since for the training the true class of the event (signal/background) is known, the classification output is plotted for both classes independently (a). From the classifier outputs y the Reciever Operating Characteristics (ROC) curve of all classifiers are computed (b). They show the background rejection as a function of the signal efficiency. In the case of the example shown in figure (b) the result of the *MLP*-method coincides with the *Fisher* result and thus the two ROC curves cannot be distinguished.

information from the observables efficiently.

For the classification, a mapping from the N-dimensional phase space of the N input variables to one dimension. A further mapping to the signal and background class completes the classification (see equation 1):

$$\mathbb{R}^N \longrightarrow \mathbb{R} \longrightarrow \{C_{\text{signal}}, C_{\text{background}}\} \quad . \tag{1}$$

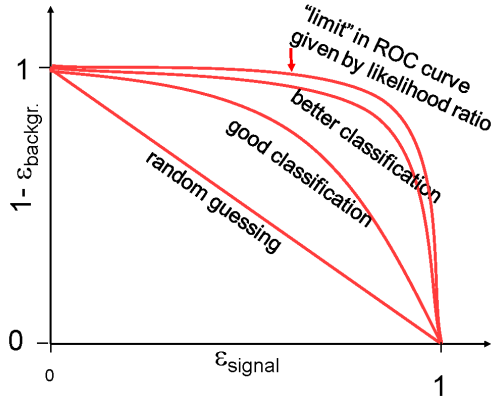
For the "cuts"-classification method, the mapping from $\mathbb{R}^N$ to the signal and background classes is done directly, without the intermediate step to $\mathbb{R}$.

Classification in TMVA derives from the input variables (observables) a classifier output where signal- (background-) like events have values close to 1 (0). The mapping to the signal and background class is done by defining all events with a classifier output $y > y_{\text{cut}}$ as signal and all other events as background. For each cut value y_{cut} the signal efficiency $\epsilon_{\text{sig,eff}}$, purity and background rejection $(1 - \epsilon_{\text{bkg,eff}})$ are calculated.

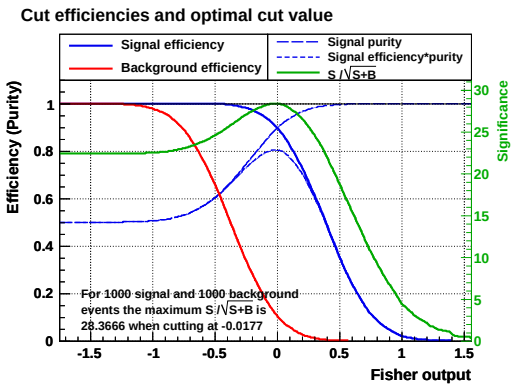
3.1. Evaluation of the Classification

TMVA provides a multitude of evaluation outputs which help the user to decide on the best classifier to choose for a particular classification problem.

For each trained classifier the distribution of the classifier output y for signal and background events can be inspected by the user (see figure 3a). Signal and background efficiencies are computed for a set of cuts on the classifier output. All events with a classfier output larger than the cut value are classified as signal, all events below the cut are classified as background. From the number of events which are classified right or wrongly as signal or background, the efficiencies can be calculated. In figure 4a several exemplary ROC curves with different classification performances are shown. The larger the area below the curve, the better the separation of signal and background which can be achieved.



(a) description of the features of the ROC curve



(b) Optimal cut value for the classifier output y

Figure 4: The ROC curve (a) shows the relationship between signal efficiency ($\epsilon_{\text{eff,signal}}$) and background rejection ($1 - \epsilon_{\text{eff,background}}$). In figure (b), the significance, the purity and the efficiencies for signal and background as well as the signal efficiency multiplied with the purity are shown.

From the sets of signal efficiencies and background rejections ($1 - \epsilon_{\text{eff,background}}$) defined by the cuts on y the Reciever Operating Characteristics (ROC) curve is plotted (see figure 3b). Which point on the ROC curve the user should choose as *working point* (i.e. which *signal efficiency* and which *background rejection*) depends on the type of analysis the user wants to perform. For trigger selection, a high efficiency will be chosen to prevent from signal events being discarded at a too early stage. For a signal search, the best cut is where $S/\sqrt{B}$ has a maximum. When a signal is found, the best cut for measuring the cross section is where $S/\sqrt{S+B}$ has a maximum. Finally for precision measurements one aims for a high purity.

Given the event yields, TMVA calculates significances, purities and efficiencies and proposes an optimal cut value to get the best performance. In figure 4b the output of the tool is shown which computes the those values as a function of the cut on the classifier output.

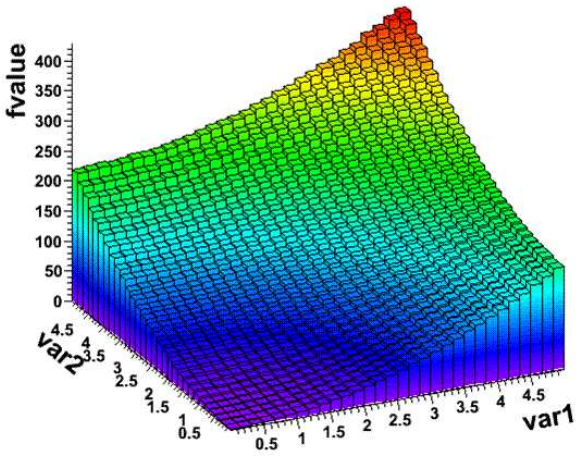
4. Regression

The aim of *regression analysis* is to create a model which maps N observables (input variables) to M response variables (targets) (see equation 2) such that the deviation of the estimated target values (output of the regression analysis model) and the true target values is minimal.

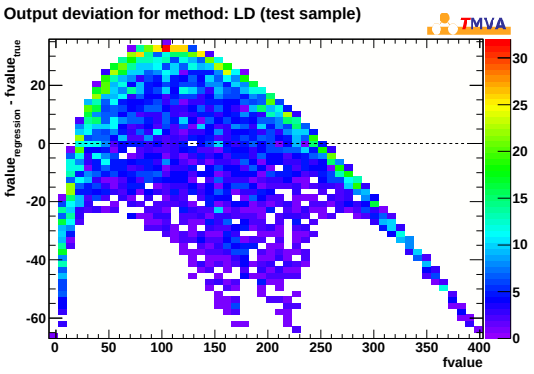
$$\mathbb{R}^N \longrightarrow \mathbb{R}^M \tag{2}$$

Typically M is 1 or at least smaller than N. Most methods which are implemented in TMVA for classification can also be used for regression. In the *linear description* (LD) method a linear model is assumed for the dependence of the target value from the input variables. Using the *formula description analysis* (FDA) method the user defines a model (e.g. a linear model with some quadratic terms). More sophisticated models such as the *neuronal network* (MLP), the *probability density estimator - range search* (PDE-RS)[7], *probability density estimator - foam* (PDE - Foam)[8, 9] or *boosted decision trees*[13, 14] (BDT) implicitly create a problem dependent models. Those methods have the capability to adapt to the problem and deliver a good performance even for complex problems.

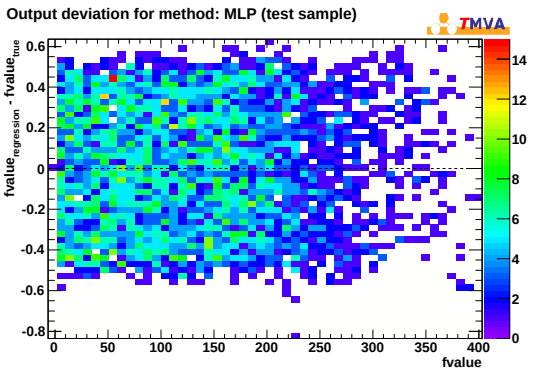
An example where regression analysis can be applied in high energy physics is the estimation of a corrected jet energy from energy and jet shape variables.



(a) Regression example with two input variables and one regression target.



(b) Performance of linear regression



(c) Performance of regression with MLP.

Figure 5: Example of one *target*-value (fvalue) depending on two input variables (var1, var2) (a). Evaluation plot for the LD method and regression. The difference between the estimated and the true target value is plotted as a function of the true target value for *linear description* (b) and *neuronal network* (c). for a perfect result of the regression analysis a straight horizontal line at 0 would be drawn.

4.1. Evaluation of the Regression

For the evaluation of the regression, a plot is produced showing the deviation of the estimated from the true *target*-value as a function of the *target*-value. Further plots are produced which show the deviation as a function of each input variable. In that way, the user can easily detect if there are phase space regions where the regression analysis with one of the methods does not perform well and which of the methods perform well (see figures 5b and 5c).

5. Methods for Classification and Regression implemented into TMVA

A multitude of methods are implemented into TMVA. All of them are capable of signal/background classification and most of them of regression with one regression target. Some

Table 1: Methods implemented in TMVA and their ability to perform classification and regression. If the method can be used for regression analysis, the number of targets is given which can be trained simultanously in one analysis.

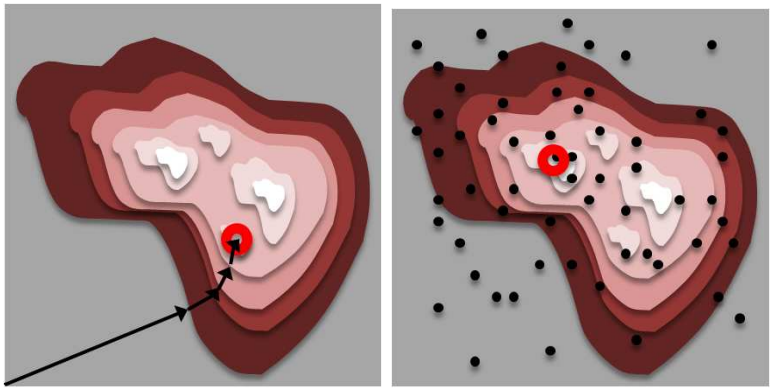
Method name	Classification	Regression (# targets)
Fisher[12]	yes	no
Linear description (LD)	yes	1
Functional description analysis (FDA)	yes	1
Projective likelihood	yes	1
Cuts	yes	no
Probability density estimator—range search (PDE-RS)[7]	yes	N
Probability density estimator—foam (PDE-foam)[9]	yes	N
Neuronal network (MLP)	yes	N
Boosted decision trees (BDT)[13, 14]	yes	1
Support vector machine (SVM)[16, 17, 18]	yes	1
Rule ensembles (RuleFit)[15]	yes	no

		MVA METHOD									
	CRITERIA	Cuts	Likeli-hood	PDE-RS / k-NN	PDE-Foam	H-Matrix	Fisher / LD	MLP	BDT	Rule-Fit	SVM
Perfor-mance	No or linear correlations	⋆	⋆⋆	⋆	⋆	⋆	⋆⋆	⋆⋆	⋆	⋆⋆	⋆
	Nonlinear correlations	○	○	⋆⋆	⋆⋆	○	○	⋆⋆	⋆⋆	⋆⋆	⋆⋆
Speed	Training	○	⋆⋆	⋆⋆	⋆⋆	⋆⋆	⋆⋆	⋆	○	⋆	○
	Response	⋆⋆	⋆⋆	○	⋆	⋆⋆	⋆⋆	⋆⋆	⋆	⋆⋆	⋆
Robust-ness	Overtraining	⋆⋆	⋆	⋆	⋆	⋆⋆	⋆⋆	⋆	○	⋆	⋆⋆
	Weak variables	⋆⋆	⋆	○	○	⋆⋆	⋆⋆	⋆	⋆⋆	⋆	⋆
Curse of dimensionality		○	⋆⋆	○	○	⋆⋆	⋆⋆	⋆	⋆	⋆	
Transparency		⋆⋆	⋆⋆	⋆	⋆	⋆⋆	⋆⋆	○	○	○	○

Table 2: Assessment of MVA method properties. The symbols stand for the attributes "good" (⋆⋆), "fair" (⋆) and "bad" (○). "Curse of dimensionality" refers to the "burden" of required increase in training statistics and processing time when adding more input variables. "Weak variables" refers to variables with no or only a small discrimination power. The FDA method is not listed here since its properties depend on the chosen function.

(e.g. MLP) are capable of multitarget regression as well.

In table 1 all methods and their capabilities are listed. In table 2 an overview of the performance of the different methods is given with respect to linear and non-linear correlations, to speed in training and response, robustness, transparency etc. Note, that the performances indicated with



(a) Minuit (b) Monte Carlo

(c) Genetic algorithm (d) Simulated annealing

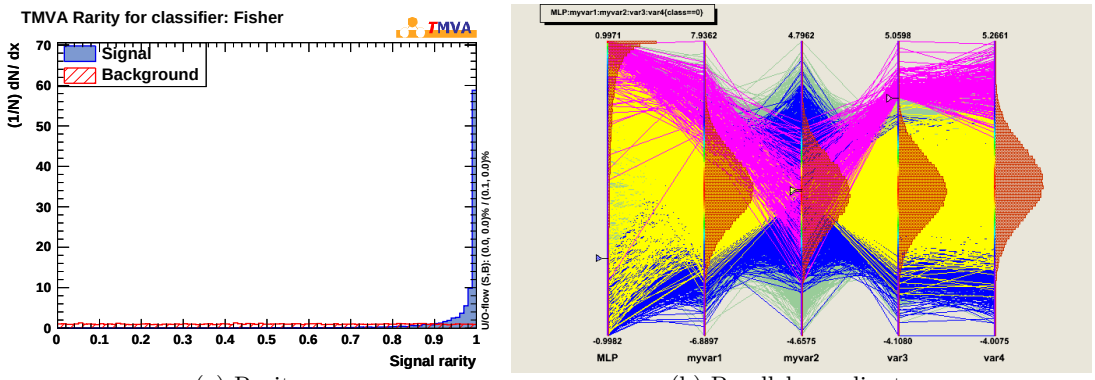
Figure 6: Principle of the minimization algorithms implemented in TMVA. Minuit (a) searches along a path following the steepest ascent. Monte Carlo minimization (b) searches the phase space randomly. The genetic algorithm (c) is modelled after biological evolution. Simulated annealing (d) is modelled after the annealing in metallurgy.

stars and circles only show the tendency ("good" ($\star\star$), "fair" ($\star$) and "bad" ($\circ$)), since the actual performance is influenced as well by the type of the classification/regression problem.

5.1. Minimization in TMVA
Many methods in TMVA require an efficient finding of global minima. In TMVA a flexible system for minimizing with four different minimizers is implemented. In figure 6 the four minimizers Minuit, Monte Carlo, Genetic algorithm and simulated annealing are outlined.

5.1.1. Monte Carlo sampling Monte Carlo sampling (see figure 6b) is a brute force method. The entire parameter space is sampled and the solution which provides the minimum estimator after a user defined number of sampling points is taken as global minimum. Monte Carlo sampling is a good, but very slow global minimum finder. The probability of being tested is equal for each point in the phase space. Hence, usually many points have to be tested before one point is found which is close to the minimum.

5.1.2. Minuit Minuit[10] (see figure 6a) is widely used in high energy physics. It is a gradient-driven search, using a variable metric and can use a quadratic newton-type solution. Minuit is a poor global minimum finder, since it gets quickly stuck in the presence of local minima.



(a) Rarity (b) Parallel coordinates

Figure 7: In figure (a) the ratity of signal and background events is shown. The rarity transformation produces a uniform distribution for the events of the background class. Figure (b) uses the ROOT implemented parallel coordinate plotting function to display the input variables and their dependencies.

5.1.3. Genetic algorithm The genetic algorithm (see figure 6c) is a biology-inspired tool for optimisation. A "genetic" representation of the points in the parameter space is used. Points which produce bad estimator values are discarded, better ones are preserved for *mutation* and *crossover*. That way, the "knowledge" of good regions where the estimator values are closer to the optimum is kept and used to produce new points. The genetic algorithm is capable of approximately finding global minima and it is considerably faster than Monte Carlo sampling.

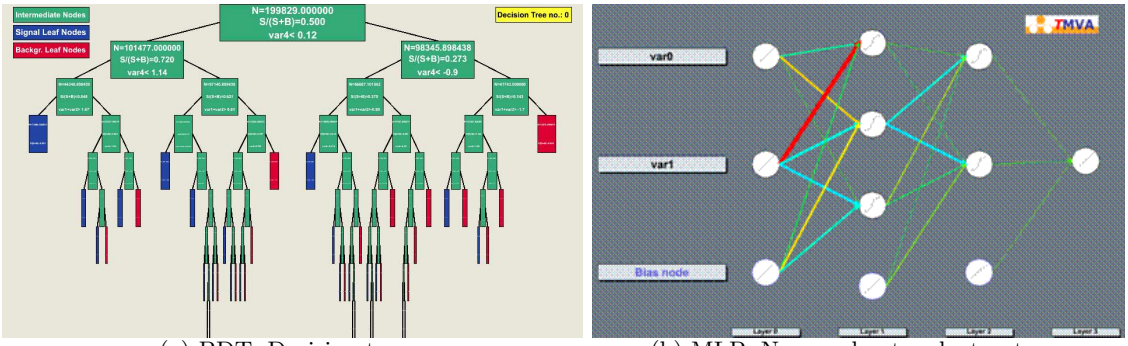
5.1.4. Simulated annealing Simulated annealing[11] is modelled after the "annealing" process done for metals where the metals are heated up and slowly cooled down. The atoms in the metals move towards the state of lowest energy while for sudden cooling atoms tend to freeze in intermediate higher energy states. In simulated annealing (see figure 6d) the system is slowly cooled down to avoid freezing at a local minimum.

5.2. Additional Evaluation Output

A set of additional evaluation outputs are created which visualize different aspects of the classification/regression analysis, such as Correlation matrices for the input variables of all event classes (signal/background or regression), convergence plots for neuronal networks or control plots for BDT. Two examples of evaluation output are the rarity transformation and *parallel coordinates*. The rarity transformation[19]

$$\mathcal{R}(y) = \int_{-\infty}^{y} \hat{y}_B(y')\, dy' \ , \tag{3}$$

with $\hat{y}_B()$ being the probability density function for the background which is calculated for the output classifier of signal and background classes (see figure 7a). With this transformation, the background distribution is uniformly distributed from 0 to 1 and the signal shows peaks where its probability density function differs from the one of the background. The input variables and the output classifier are also shown in *parallel coordinates* (see figure 7b). There, input variables and the classifier or the regression targets are shown in parallel and the user can inspect the mapping of one variable onto the others.



(a) BDT: Decision trees (b) MLP: Neuronal network structure

Figure 8: In figure (a) a BDT decision tree is shown. Each decision tree can be inspected. Figure (b) shows the structure of a trained neuronal network.

Apart from the evaluation output which is produced for all methods, for some methods additional plots are created. Examples for those are the decision trees (see figure 8a) and the network structure of a MLP neuronal network (see figure 8b).

6. Summary

TMVA is a software toolkit for multivariate statistical analyses. It has been originally designed for the use in high energy physics but it is not constrained to that. TMVA is a mature package which is constantly maintained (e.g. bugfixes, implementation of feature requests) and with a very active developer community. TMVA 4 is the first release after an important change of the underlying framework which provides the ability for regression analysis and several planned features which will be implemented in the coming releases (e.g. multiclass classification). Although signal/background classification has been and is still the main task of TMVA, regression analysis will give the users a new powerful tool for their scientific work.

Acknowledgments

The fast growth of TMVA would not have been possible without the contribution and feedback from many developers and users to whom we are indebted. We thank Jan Therhaag, Eckhard von Toerne, Matt Jachowski, Yair Mahalalel, Andrzej Zemla and Marcin Wolter, Rustem Ospanov. We are grateful to Doug Applegate, Kregg Arms, René Brun and the ROOT team, Andrea Bulgarelli, Tancredi Carli, Zhiyi Liu, Elzbieta Richter-Was, Vincent Tisserand, Lucian Ancu and Alexei Volk for helpful conversations and bug reports.

References

[1] Hoecker A, Speckmayer P, Stelzer J, Tegenfeldt F, Voss H, Voss K, Christov A, Henrot-Versillé S, Jachowski M, Krasznahorkay Jr. A, Mahalalel Y, Ospanov R, Prudent X, Wolter M and Zemla A 2007 TMVA: The toolkit for multivariate data analysis, (*Preprint* arXiv:physics/0703039).
[2] `https://tmva.sourceforge.net/`
[3] Friedman J, Hastie T and Tibshirani R 2001 *The Elements of Statistical Learning*, Springer Series in Statistics.
[4] Webb A 2002 *statistical pattern recognition*, 2nd Edition, J. Wiley & Sons Ltd.
[5] The following web pages give information on available statistical tools in HEP and other areas of science: `https://plone4.fnal.gov:4430/P0/phystat/`, `http://astrostatistics.psu.edu/statcodes/`.
[6] Brun R and Rademakers F 1997 ROOT - an object oriented data analysis framework, Nucl. Inst. Meth. in Phys. Res. **A 389**, 81.
[7] Carli T and Koblitz B 2003 Nucl. Instrum. Meth. **A 501**, 576 (*Preprint* hep-ex/0211019).
[8] Jaddach S 2002 foam: a general-purpose cellular monte carlo event generator, CERN-TH/2002-059, (*Preprint* arXiv:physics/0203033)

[9] Carli T, Dannheim D, Voigt A, Speckmayer P PDE-FOAM – a probability-density estimation method based on self-adapting phase-space binning, (*in preparation*)
[10] James F 1994 *MINUIT, Function Minization and ERROR Analysis*, Version 94.1, CERN program Library long writeup D506.
[11] Van Laarhoven P J M and Aart E H L 1987 *Simulated Annealing: Theory and Application*, D. Reidel Publishing, Dordrecht, Holland.
[12] Fisher R A 1936 Annals Eugenics **7**, 179.
[13] Quinlan Y R 1987 simplifying decision trees, Int. J. Man-Machine Studies, **28**, 221.
[14] Breiman L, Friedman J, Olshen R and Stone C 1984 *Classification and Regression Trees*, Wadsworth.
[15] Friedman J and Popescu B E 2004 *Predictive Learning via Rule Ensembles*, Technical Report, Statistics Department, Stanford University.
[16] Cortes C and Vapnik V 1995 support vector networks, Machine Learning, **20**, 273.
[17] Vapnik V 1995 *The Nature of Statistical Learning Theory*, Springer Verlag, New York.
[18] Burges C J C 1998 *A Tutorial on Support Vector Machines for Pattern Recognition*, Data Mining and Knowledge Discovery, 2, 1.
[19] To our information, the *Rarity* has been originally defined by F. Le Diberder in an unpublished Mark II internal note. In a single dimension, as defined in Eq. (3), it is equivalent to the μ-transform developed in: Pivk M, "*Etude de la violation de CP dans la désintégration* $B^0 \to h^+h^-$ $(h = \pi, K)$ *auprès du détecteur BABAR à SLAC*", PhD thesis (in French), `http://tel.archives-ouvertes.fr/documents/archives0/00/00/29/91/index_fr.html` (2003).