VERSITA

TATRA MOUNTAINS
Mathematical Publications

DOI: 10.2478/v10127-012-0013-0
Tatra Mt. Math. Publ. **51** (2012), 131–140

NOVEL APPROACHES OF DATA-MINING IN EXPERIMENTAL PHYSICS

Gennady A. Ososkov

ABSTRACT. Data mining for processing experimental data in high energy and nuclear physics led to many multiparametric problems, two of them are considered: (i) hypothesis testing and classification approaches based on artificial neural networks and boosted decision trees (ii) clustering of large amounts of data by so-called growing neural gas. Some examples from the practice of the Joint Institute for Nuclear Research are given to show how to prepare data to deal with those approaches.

1. Introduction

Contemporary experiment in high energy and nuclear physics are characterized by very high rate of produced data and their extremely sophisticated structure, while the wanted physical effects are hidden in huge background exceeding useful information by many orders of magnitude. Such the process of analyzing large amounts of data about experimental results held on a computer in order to get essential information about them is not immediately available, was called data mining (DM) [1]. It is accomplished by extracting dependencies and patterns from large data sets by combining methods from statistics and artificial intelligence with database management. Nevertheless, data processing for really great data sets produced in experimental physics appeared to be not included in the known DM systems although all characteristics of data and approaches for their handling look appropriate to DM applications. In this paper we study how DM approaches should be extended and modified if data will be taken from high energy physics.

2010 Mathematics Subject Classification: Primary 62J05; Secondary 62F03, 62F10, 62F30.
Keywords: data mining, high energy physics, computer experiment, artificial neural network, boosted decision trees, growing neural gas.

2. Data mining peculiarities for experimental high energy physics (HEP)

The main HEP data specifics are determined by ways of their acquisition in the course of various experiments. Since any HEP experiment is usually intended to observe a sequence of events, i.e., physical collisions, registered data consist of distinct portions named also *events*, which structures are quite different depending on the experimental setup. To explain HEP data and arisen DM problems let us consider a couple of examples chosen due to the author involvement into corresponding experiments.

Example 1. The Compressed Baryonic Matter (CBM) experiment at the future FAIR accelerator at Darmstadt, Germany is being designed for a comprehensive measurement of hadron and lepton production in heavy ion collisions from 8–45 AGeV beam energy. CBM main characteristics are: 10^7 events per sec, ∼1000 tracks per event, ∼100 numbers per track. Totally one has to handle *terabytes of data per every second*.

One of CBM important detectors is the RICH–Cherenkov radiation detector [2]. Whenever electron or other alternative particle, as pion, passed the RICH gas radiator it produced several photons registered by the RICH photosensitive plane forming a ring, of which radius allows to identify this particle (see Fig. 1a). Our problems are (i) to recognize all of these rings and evaluate their parameters despite of their overlapping, noise and optical shape distortions, (ii) elaborate a reliable criterion which uses these parameters for particle identification.

The next example we consider is the CBM Transition radiation detector (TRD) [3], which measurements allow for each particle to calculate its *energy loss* (EL) during its passage through several TRD stations in order to distinguish electrons $\boldsymbol{e-}$ from pions $\boldsymbol{\pi\pm}$. Unlike $\boldsymbol{\pi\pm}$, electrons generate additionally *the transition radiation* (TR) in TRD. Our problem is to use the distributions of EL+TR for $e-$ and $\pi\pm$ in order to elaborate a criterion for testing hypothesis about a particle attributing to one of these alternatives keeping the probability α of the 1st kind of error on the fixed level $\alpha = 0.1$ and the probability β of the 2nd kind of error on the level less than $\beta < 0.004$. As it is shown in Fig. 1b, distributions of EL for pions and EL+TR for electrons both belong to long-tailed Landau type and, therefore, any test based on direct cut on the sum of energy losses fails to satisfy these requirements.

Example 2. **The OPERA experiment** is intended to search for neutrino oscillations (OPERA is running [4]). The target part of the detector (see Fig. 2a) named the Target Tracker (TT) consists of two great modules with 31 target brick walls each (photoemulsion layers, interlaced with led layers, are organized in bricks). Currently only 53 walls are active. Each wall consists of 3328 bricks and

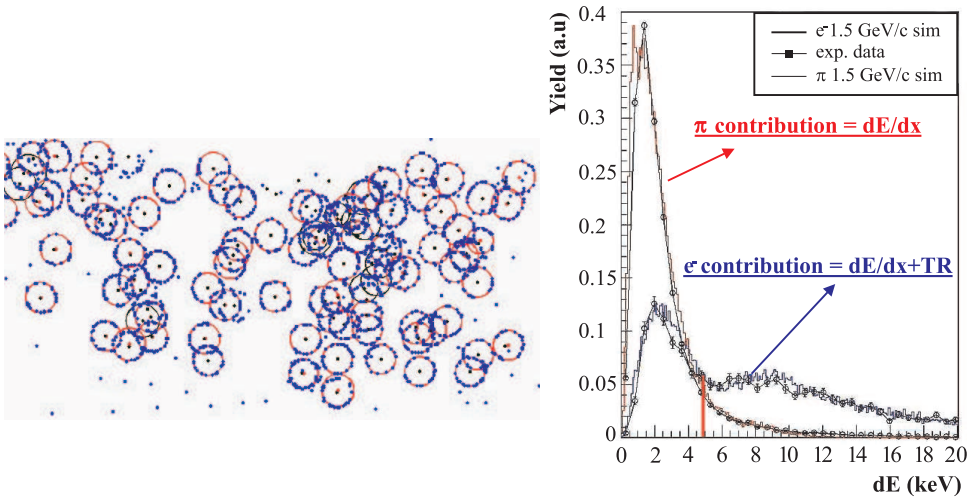


FIGURE 1. a) View of Cherenkov radiation rings. b) Distributions of EL for pions and EL+TR for electrons.

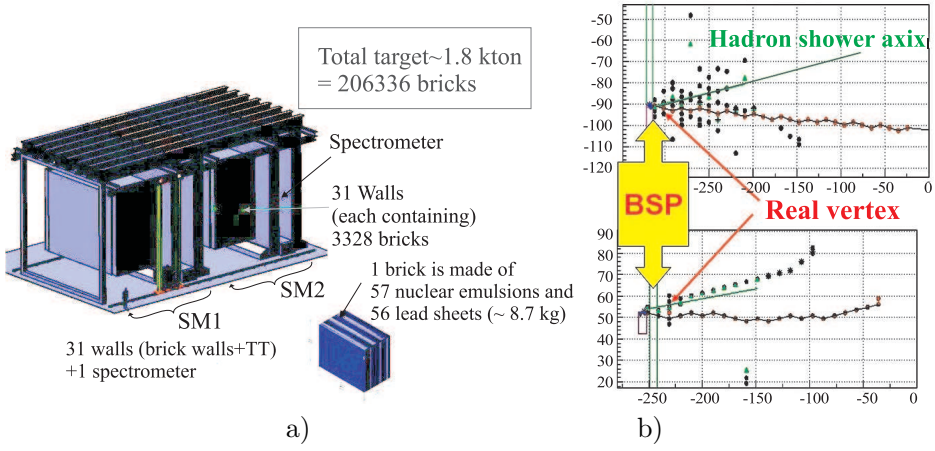


FIGURE 2. a) Schematic view of target tracker detector. b) Two types of OPERA events with BSP.

is accompanied by two planes (X–Y) of 2×31 scintillator strips. Signals from each strip are read out to trace all particles passages through TT that is necessary for a location of the event vertex position, i.e., identifying a target brick which is to be extracted for the further investigation on a special scanning device. Since the extracted brick could not be returned back due to technical reasons, it must be

identified extremely accurately. Thus the crucial issue in OPERA DM is finding of that particular brick where the neutrino interaction takes place. Tracks formed by scintillator hits should originate from a single point-vertex. However the main obstacle in vertex finding is back-scattered particles (BSP) occurring in 50 % of events (Fig. 2b). Our goal was to elaborate a flexible program system for brick finding from data registered by TT detector using known effective methods of data filtering and track recognition.

Both experiments look very different from experimental point of view. However from data handling site, one sees, although terabytes of data per second for the CBM require a sophisticate triggering procedure and parallel data processing, the rest of data-mining process for both types of experiments has quite similar stages and methods.

The very important stage of DM process in experimental HEP is *pre-processing*, which consists of the following steps:

(i) *data acquisition*: before data mining algorithms can be used, a target data set must be assembled and converted from the rough format of detector counters into natural unit format;

(ii) *data selection*: then data must be cleaned to remove noisy, and inconsistent observations, what results in a significant reduction of target data;

(iii) *data transformation*: to transform data into forms appropriate for mining, they must be corrected from detector distortions and misalignment by special calibration and alignment transformation procedures.

Next HEP DM stages and methods can be summarized as:

(1) *Pattern recognition*: tracking, vertex finding, revealing Cherenkov rings, fake objects removing that employ the methods, as Hough transform, Kalman filter, artificial neural networks, cellular automata, wavelet analysis and so on;

(2) *Physical parameters estimation* with applying robust M-estimations;

(3) *Hypothesis testing* by likelihood ratio test, neural network and boosted decision trees (BDT) approach.

It should be pointed out the importance of *Monte-Carlo simulations* in HEP that are used on all stages. It is based on the very advanced physical theory of the studied particle interactions for HEP experiments. Special programming packages were developed to simulate physical processes taking into account all details of experimental setups. It allows to accomplish in advance the experimental design of hardware setup and data mining algorithms and optimize them from financial, material and time point of view, then develop needed software framework and test it. Simulations allow also optimize structure and needed

equipment of planned detectors minimizing costs, timing with a proposed efficiency and accuracy. Besides by simulations it is possible to calculate in advance all needed distributions or thresholds for goodness-of-fit tests.

3. Neural network applications in HEP

In further expounding, to demonstrate the effectiveness of methods employed for data processing in HEP, we focus on the artificial neural network (ANN) approaches that used almost on all stages of HEP DM. For the sake of brevity we remind briefly only the basic concepts of the ANNs referring to our surveys [5] and widely known literature [6]. Artificial neurons are simple logical units specified by:

(i) activation level;

(ii) the measure of interaction with other neurons, which is referred to as synaptic weight;

(iii) output level, which is related to the activation level by a certain, usually sigmoidal, function;

(iv) topology of connections between neurons.

The weights of these connections are different and can be defined in dependence of the problem under consideration. The entire system consists of a vast number of identical neurons and the result of the operation of an ANN is almost not sensitive to the characteristics of a specific neuron. The key characteristics of ANN are the type of connections between neurons and network evolution dynamics determined by the activation function for neurons and the rule of varying weights upon this evolution.

3.1. Three examples of ANN application in physics

ANNs that are extensively used in physics are determined by connections of two types: *feed-forward networks without feedback*, e.g., multi-layer perceptrons (MLPs) or *recurrent networks*, where neurons are all connected with each other as in the Hopfield neural network.

We do not consider here examples of recurrent NN applications in HEP since many of them were reported already in details (see, for example, [5, Chapter 3]).

The first type, in particular, MLPs with the back propagation of error algorithm for NN training are quite popular in experimental physics mainly due to the possibility to generate training samples of any arbitrary needed length by Monte Carlo on the basis of some new physical models.

We use three above mentioned examples of identification problems for the CBM and OPERA experiments to demonstrate how effective are MLP applications in solving these problems.

3.1.1. NN for the CBM RICH detector

We omit details of important data processing stages for the Cherenkov radiation ring recognition, compensating their optical distortions lead to *elliptic shapes of rings*, evaluating of ellipse parameters for each found ring and matching it with that of particle tracks which are interesting to physicists, referring to [3]. However, the application of ANN on the next stage devoted to the particle identification (PID) with the fixed level of the ring recognition efficiency should be elucidate more in details. The experimentalist requirement was to elaborate a PID procedure intended to test an input set of a ring features and make a decision whether electron or pion possesses those features.

Instead of considering such the common statistical test characteristic as the PID test power, i.e., $1 - \beta$, where β is the probability of the 2nd kind error, physicists prefer to use so-called "pion suppression value", i.e., $1/\beta$, as more expressive. More advanced methods of analyzing classification results known in data mining, as ROC (receiver operating characteristic) curves are not yet in use in experimental high energy physics.

The study has been made to select the most informative ring features needed to identify electrons. Ten of them have been chosen to be input to ANNs, such as number of points in the found ring; its distance to the nearest track; track momentum; $\boldsymbol{\chi}^2$ of ellipse fitting; ellipse half-axes and angle of the ellipse inclination and so on. Two samples with 3000 electrons ($\boldsymbol{e}$) and 3000 pions $\boldsymbol{\pi}$ were simulated to train ANN with 10 input neurons, 20 hidden and one output neuron. The latter should be set to 1, if $\boldsymbol{e}$ occurs on input, and -1 in the case of $\boldsymbol{\pi}$. When the recognition efficiency was fixed on 90 % the great testing sample with half million events was simulated which was used in the particle identification procedure. The probabilities of the 1st kind error 0.018 and the 2nd kind error 0.0004, correspondingly, were obtained. That gave pion suppression as 2500, which was quite satisfactory for experimentalists.

3.1.2. NN for $e-/\pi\pm$ separation by transition radiation [3]

To avoid mentioned above obstacles with the long tails of energy loss (ΔE) distributions we use for testing the ANN with preliminary prepared input as the likelihood ratios calculated for ΔE of each TRD station. We use Monte-Carlo to simulate a representative sample of TRD signals for given experimental conditions and then obtain energy losses from all $\boldsymbol{n}$ TRD stations for both $\boldsymbol{e}-$ and $\boldsymbol{\pi}\pm$, sort them and estimate probability density functions for ordered ΔEs.

Then we repeated simulation in order to train neural network with $\boldsymbol{n}$ inputs and one output neuron, which should be equal to $+1$ in case of electron and and -1 in case of pion. As inputs, the likelihood ratios for each ΔE were calculated. The result of testing the trained neural network on the new great testing sample gave the probability of the 2nd kind of error $\beta = 0.002$. It satisfied the required experimental demands, but physicists wanted to study whether any better method to suppress pions in experimental data exists.

3.1.3. NN for brick finding in the OPERA experiment [4]

To facilitate the neutrino vertex location from data points formed by scintillator signals (see Fig. 2), data preprocessing has been fulfilled by use a special cellular automaton to filter scintillator data from isolated points having no nearest neighbours. Then several methods were applied to facilitate the brick finding procedure: Hough transform and Kalman filter used to reconstruct muon tracks; robust M-estimates for determining hadron shower axes were developed with 2D robust weights depending not only on the distance of a point to the shower axis, but also amplitudes of scintillator signals at this point. After a comprehensive analysis of already obtained and simulated data, 3 classes of events were separated according to their topology in the TT detector and 15 parameters to input them to 3 neural networks of MLP type were determined. MLPs were then trained for each class on 20000 simulated events to make a decision about the wall with the event vertex. The NN efficiency of wall finding was on the level of 88–98 %. Combination of results from muon tracking, shower direction and NN wall finding was used in the brick finding procedure, which performance was found so good that the corresponding JINR program was chosen as the default one for the OPERA experiment.

3.2. Boosted decision trees in Particle IDentification (PID)

We apply our PID study some of classifiers from TMVA (Toolkit for MultiVariate data Analysis with ROOT) software package [11]. In particular, we were interested in comparing the efficiency of neural networks and the method known as boosted decision trees (BDT). The boosting algorithm is considered in [9] as one of the most powerful learning techniques introduced in the past decade. The motivation for the boosting algorithm is to design a procedure that combines many "weak" classifiers to achieve a final powerful classifier. BDT algorithm increases the weights of misclassified events (background which is classified as signal, or vice versa), such that they have a higher chance of being correctly classified in subsequent trees. Trees with more misclassified events are also weighted, having a lower weight than trees with fewer misclassified events. Then many trees ($\sim$1000) are build in the course of BDT training and weighted sum of event scores from all trees are calculated. The renormalized sum of all the scores, possibly

weighted, is the final score of the event. High scores mean the event is most likely signal and low scores that it is most likely background.

After comparative testing several PID methods from the TMVA package for the CBM TRD and OPERA data we found that for the dichotomy problems like in the CBM TRD case the efficiency of the BDT method exceeded ANN's efficiency for 10–15 % [4]. However in more sophisticated cases of classifying data into 3 and more classes, as in the case of OPERA experimental information, the TMVA BDT realization should be applied in a sequence loosing its privileges with regard to ANN.

Therefore, as a result of our study of new approaches in classifying of very large amount of data, what is the typical data-mining problem, we propose a method based on preliminary data reduction by using an advanced algorithm of the growing neural gas.

3.3. Growing neural gas for clustering large amounts of data

In the often case, when the feature space used for classifying the input data has many dimensions one needs to use clustering algorithms of rather high complexity, but if at the same time the number of measurements to be processed is extremely large (exceeding 10^6 and more), those complex algorithms are becoming unsuitable. Although for such cases there is well-known k-means clustering algorithm [10], it has such disadvantages, as NP complexity, a necessity to know number of clusters in advance and a tendency to treat clusters, as compact clouds distributed in a normal fashion in the feature space. Some example of inapplicability k-means clustering is shown below in Fig. 3.

To avoid those disadvantages of k-mean algorithm we choose a new one--*growing neural gas* (GNG) [11], which produces a preliminary partition of a large input into so-called Voronoi regions, then turning to conventional clustering algorithms to process a significantly smaller number of such regions. The Voronoi partition divides the vector space so that for each subset Q_i of the partition one can choose such reference vector C_i that all objects $x \in Q_i$ are nearer to it than to any other reference vector $C_j (j \neq i)$. The optimality function of clustering is understood as the total quantization error, which is the sum of intragroup variances.

The process of generation of Voronoi partition with GNG involves finding a set of reference vectors $\{C_j\}$ and is called *learning*. Each reference vector is set in correspondence with an entity called *neuron*. When the neurons are trained, a symmetric binary relation $B = \{(i; j)\}$is defined which determines the dynamic inter-neuron connections. The neurons connected through this relation are called topological neighbors. Since the learning of the GNG is considered as a process, the variable $\boldsymbol{E}_j$ of local error of neuron j introduced to determine the

quantization error which arises when input vectors are substituted by a corresponding reference vector during learning. To terminate the learning it should be introduced a stopping rule and parameter λ intended to express the GNG growth speed. The training of the GNG includes two subprocesses: *adaptation* and *network growth*.

Adaptation involves selection of an input vector $\boldsymbol{x}$ according to its distribution function and short-distance movement of the nearest neuron $\boldsymbol{w}$ (which is called winning neuron) and its topological neighbors towards $\boldsymbol{x}$. If during the training for a pair of neurons (i, j) at least one input vector can be found between their Voronoi regions, then this pair is included in relation B. Otherwise, it is excluded from the relation B.

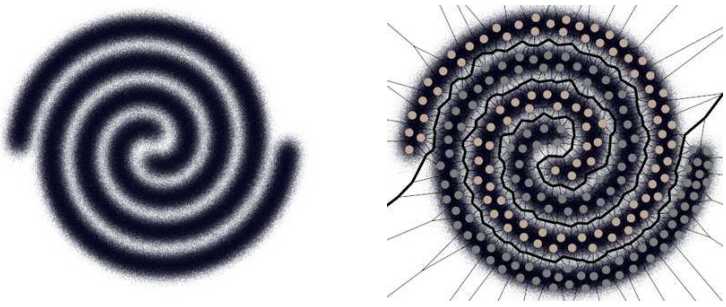
The growth of the net is that once in λ steps of adaptation a new neuron is added to the net. The neuron should be placed at such point that it reduces the total quantization error.

The detailed description of the GNG training algorithm one can find in [11].

The algorithm has complexity $O(n^2)$, where $\boldsymbol{n}$ is the number of neurons and independent of the amount of input data, what is a considerable advantage over the k-means algorithm. Besides it allows us to use the algorithm for large amounts of input data.

In the second step we apply one of well-known hierarchical algorithms of clustering [10] to clusterise obtained reference vectors which number is smaller than input data volume in several orders of magnitude.

Example of GNG clustering. Two adjacent clusters (Fig. 3) forming (from 10^6 points) two embedded spirals are considered. This is well-known benchmark of the most difficult object for clustering, which is nontraceable, in particular, by k-means approach.



a) b)

Figure 3. a) input data; b) clustering of 200 GNG neurons by the single linkage method.

The single linkage method [10], which is the most suitable for this kind of input data distribution was applied to cluster 200 GNG neurons obtained on the first step. The result of the clustering (Fig. 3b) shows that our algorithm succeeded in partition. More interesting examples of GNG application to simulated and real data one can find in [11].

REFERENCES

[1] HAN, J.—KAMBER, M.: *Data Mining: Concepts and Techniques*. Morgan Kaufmann, San Francisco, 2000.
[2] LEBEDEV, S.—OSOSKOV, G.: *Fast algorithms for ring recognition and electron identification in the CBM RICH detector*, Phys. Particles Nuclei Lett. **6** (2009), 161–176.
[3] LEBEDEV, S.—HOEHNE, C.—OSOSKOV, G.: *Status of the electron identification algorithms for the RICH and TRD detectors in the CBM experiment,* CBM Progress Report 2010, GSI, Darmstadt, 2010, p. 73.
[4] DMITRIEVSKY, S.: *On behalf of the OPERA collaboration, status of the OPERA neutrino oscillation experiment,* Acta Phys. Polon. B **41** (2010), 1539–1546.
[5] OSOSKOV, G. A.—POLANSKI, A.—PUZYNIN, I. V.: *Current methods of processing experimental data in high energy physics,* Phys. Particles Nuclei **33** (2002), 347–382.
[6] HAYKIN, S.: *Neural Networks: A Comprehensive Foundation* (2nd ed.). IEEE, New York, NY, 1999.
[7] http://cbmroot.gsi.de.
[8] *TMVA Users Guide* http://tmva.sf.net
[9] FREUND, Y.—SCHAPIRE, R. E.: *A short introduction to boosting,* J. Japanese Soc. Artificial Intelligence **14** (1999), 771–780.
[10] DURAN, B. S.—ODELL, P. L.: *Cluster Analysis: A Survey*, in: Lecture Notes in Econom. and Math. Systems. Econometrics, Vol. 100, Springer-Verlag, New York, 1974.
[11] MITSYN, S. V.—OSOSKOV, G. A.: *The growing neural gas and clustering of large amounts of data*, Optical Memory & Neural Networks **20** (2011), 260–270.

Received October 31, 2011

Joint Institute for Nuclear Research
Joliot-Courie str. 6
Dubna
Moscow region, 141980
RUSSIA
E-mail: ososkov@jinr.ru