Neural Networks in HEP

František Hakl

hakl@cs.cas.cz

Department of Machine Learning

ml.cs.cas.cz

Institute of Computer Science
Prague, Czech Republic

Edmonton
Nov 2018

Department of Machine Learning

- research in the area of mathematical foundations of computational models and their learning
- development of theory-based data-dependent architectures and algorithms, analysis of their efficiency and robustness
- application to medical, chemical, physical and environmental data
- gradual and post-gradual education

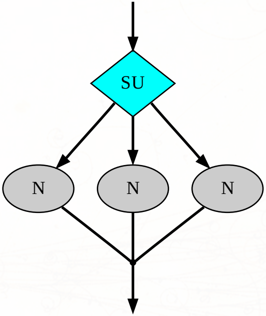
ml.cs.cas.cz

... major research activities in more detail ...

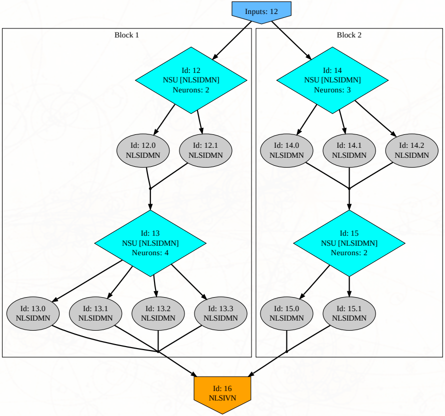
- Capabilities and limitations of deep and shallow networks – *estimates of model complexity of feedforward networks. When and why are deep networks better than shallow ones?*
- Kernel methods – *theoretical analysis of properties of convolutional kernel and radial networks in shallow and deep architectures*
- Automatic design of deep architectures – *investigation of vulnerability of machine learning models to adversarial images*
- Reliability of supervised learning – *probabilistic evaluation of classification reliability*
- Robustness to outliers – *sensitivity of standard machine learning methods (perceptron neural networks, SVM) to data contamination (outliers, severe noise) in regression and classification*
- Fast and efficient classifiers – *design of efficient classifier for data generated with high-frequency (GHz)*

Application of neural networks with schwitching units (NNSU) in HEP data separation.

NNSU architecture



- switching units (SU): Jance'y algorithm
- predefined number of clusters
- clusters of data are propagated into neural units (N)
- neural units: linear, quadratic regression, probit, logit



- switching and neural units form linear blocks
- NNSU is acyclic graph of linear blocks

Jance'y/Forgy clustering

1. for randomly chosen sequence $1 \leq j_1 < j_2 < \cdots < j_d \leq p$ set

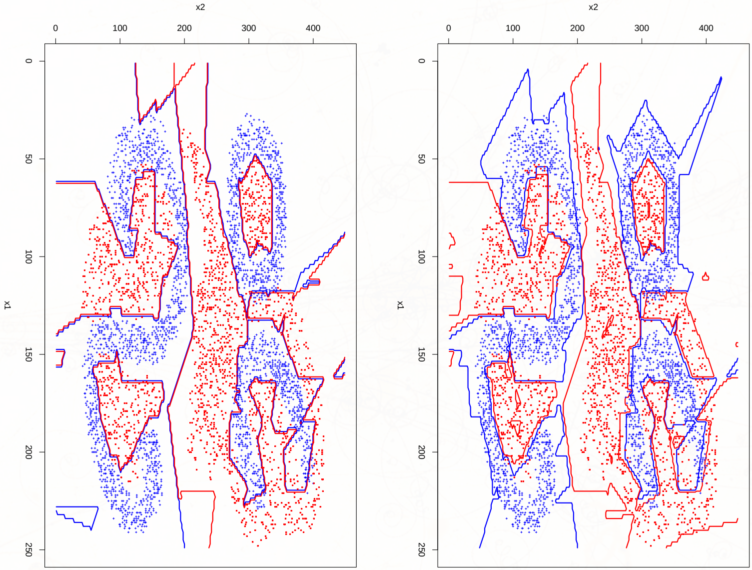
$$\mathbf{c}_q^{new} = \mathbf{c}_q^{old} = \mathbf{z}_{j_q} \quad \text{and} \quad S_q^{new} = S_q^{old} = \{\mathbf{z}_{j_q}\}, q = 1, \cdots, d$$

2. let $r_1, \cdots, r_p$ is random permutation of the $1, \cdots, p$,
3. FOR ALL $k = r_1, \cdots, r_p$
   DO

   let q be such index that $\mathbf{z}_k \in S_q^{old}$,
   $i = \min \left\{ v \,\middle|\, \|\mathbf{c}_v^{old} - \mathbf{z}_k\| = \min_{q=1,\cdots,h} \left\{ \|\mathbf{c}_h^{old} - \mathbf{z}_k\| \right\} \right\}$,
   $c_q^{old} = \mathbf{c}_q^{old} - \frac{\mathbf{z}_k - \mathbf{c}_q^{old}}{|S_q^{old}|}$, $c_i^{old} = \mathbf{c}_i^{old} + \frac{\mathbf{z}_k - \mathbf{c}_i^{old}}{|S_i^{old}|}$
   $S_q^{old} = S_q^{old} \setminus \{\mathbf{z}_k\}$, $S_i^{old} = S_i^{old} \cup \{\mathbf{z}_k\}$,

   END
4. IF $(\exists q)(S_q^{new} \neq S_q^{old})$
   THEN for all such q let $\mathbf{c}_q^{new} = \mathbf{c}_q^{old}$, $S_q^{new} = S_q^{old}$ and GOTO 2
5. STOP

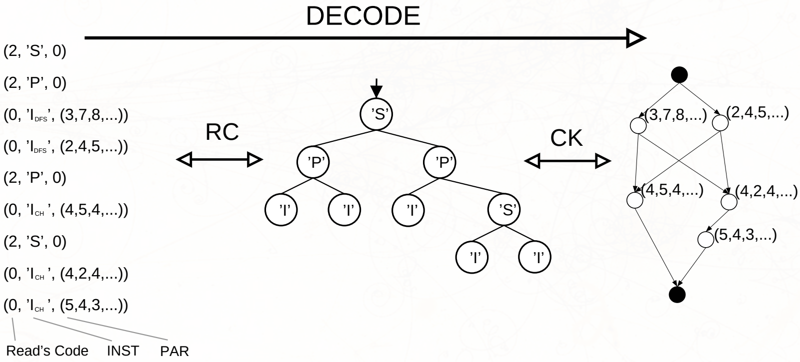
Separation border - two 2D examples



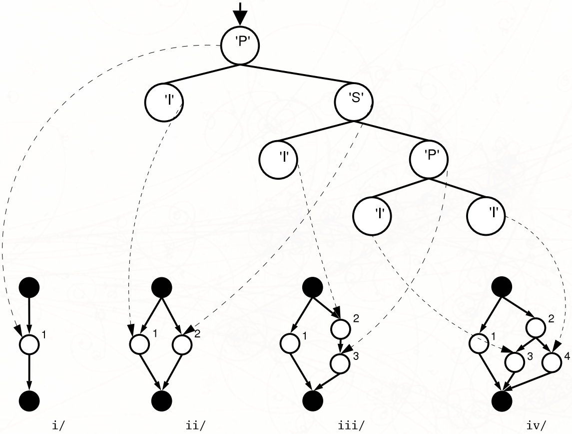
DAG representation for genetic optimization

The two main requirements on representation

1. a representation must correspond in an acceptable way to a directed acyclic graph
2. on the set of all representations the evolutionary operators (mutation, crossover) can be defined, so the set of representations is closed against such operators

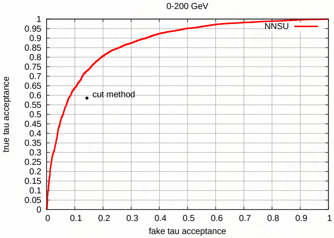


Construction of DAG – example

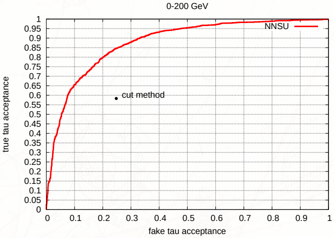


DAG construction according to an instruction tree.
Note: not all acyclic graphs can be constructed in this way

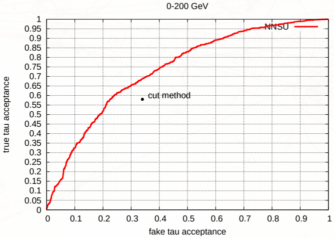
Tau hadron separation - NNSU versus cut methods



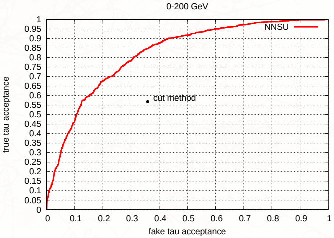
d1p0



d3p0



w1p0



w3p0

Decay tree of p–p with Higgs/Gluon production (LHC CERN)

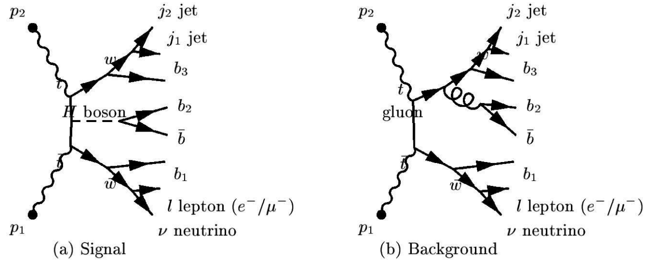
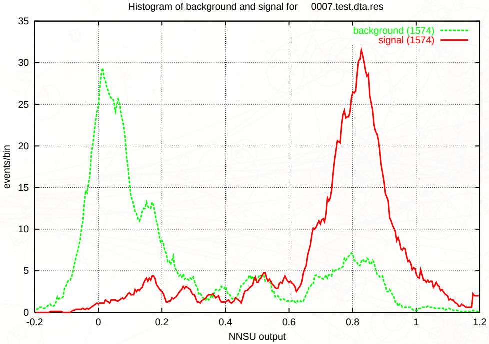


Fig. 3. Feynman diagram of decay trees.

$$M_{b_2,\bar{b}} = \sqrt{\left(E_i + E_j\right)^2 - \left((p_x)_i + (p_x)_j\right)^2 - \left((p_y)_i + (p_y)_j\right)^2 - \left((p_z)_i + (p_z)_j\right)^2}$$

NNSU output



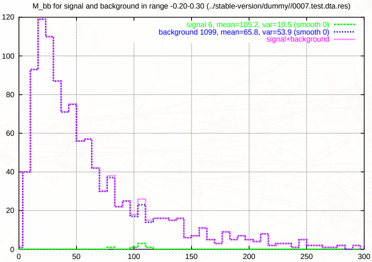
signal should be mapped to 1, background should be mapped to 0 (in an ideal case)

best signal window: $(0.5, 1.2)$

best background window: $(-0.2, 0.4)$

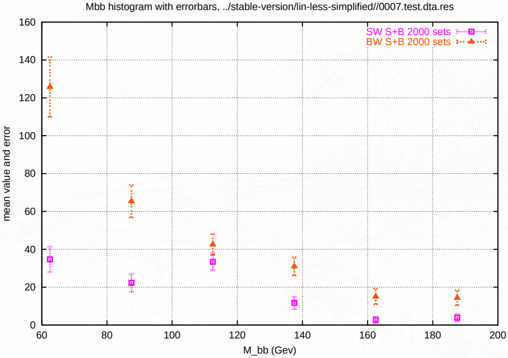
M_{bb} distribution

best background window



best signal window

M_{bb} distribution - robustness



2000 sets of 40 signals and 120 backgrounds with $M_{bb} \in (90, 150) GeV$ are randomly selected.

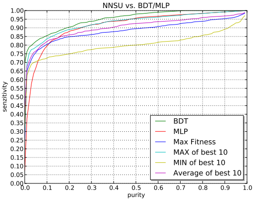
For each set i numbers S_i and B_i of all signals and backgrounds accepted by 20GeV bins in signal (background) window are computed.

Mean values of S_i and B_i for signal (background) window are ploted with coresponding σ.

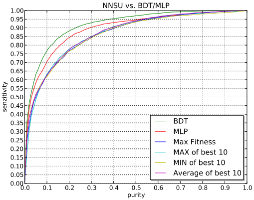
DØ Tevatron (FNAL) data

$\downarrow$ *NNSU* $\uparrow$ *BDT&MLP TMVA ROOT*

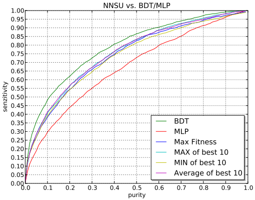
7-CC-OneTag-FourJet-tb-QCD



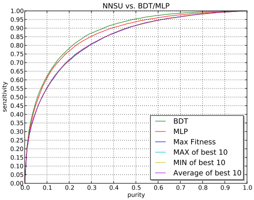
17-CC-TwoTag-ThreeJet-tb-wlp



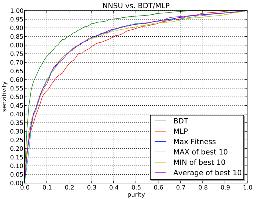
17-CC-OneTag-FourJet-TbTqb-all



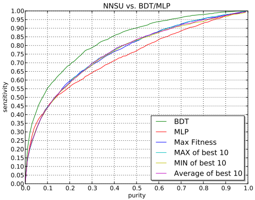
17-MU-OneTag-TwoJet-tqb-all



17-CC-TwoTag-FourJet-tb-wlp



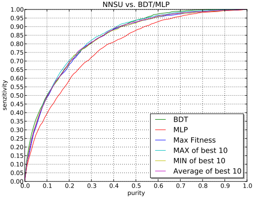
17-MU-TwoTag-ThreeJet-TbTqb-all



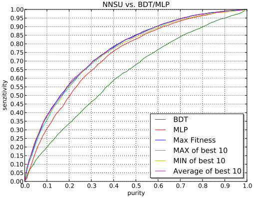
DØ Tevatron (FNAL) data

↑ *NNSU* ↓ *BDT&MLP TMVA ROOT*

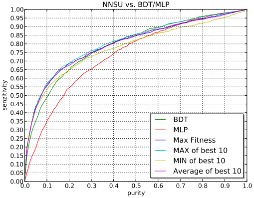
17-CC-OneTag-FourJet-tqb-tt-dilep



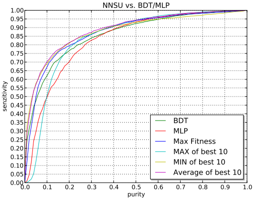
20-MU-TwoTag-TwoJet-TbTqb-wcc



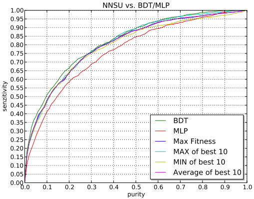
17-MU-TwoTag-FourJet-tb-tt-lepjet



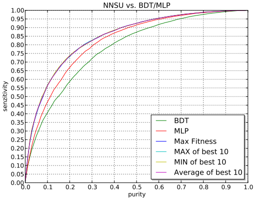
20-MU-TwoTag-TwoJet-TbTqb-wlp



17-MU-TwoTag-FourJet-tqb-tt-dilep

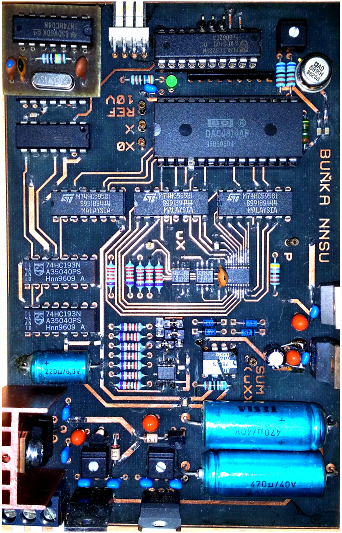


20-MU-TwoTag-TwoJet-tb-wcc



NNSU results are uncorelated with TMVA ones

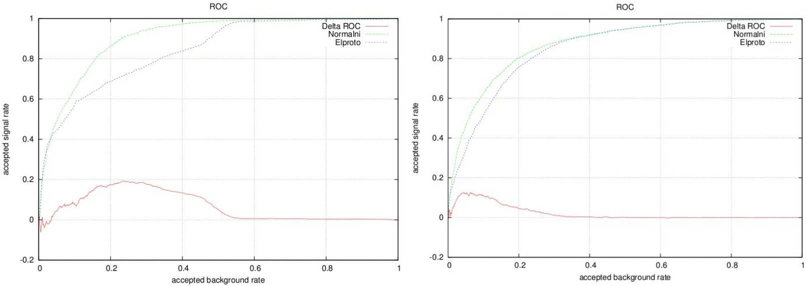
Hardware implementation - a study



- evaluating of response to input is based on "$+$", "$*$" and "$\leq$" operations only
- potential to speed up event processing targeted to low level triggering and other time edge applications
- we use cheap implementation of one NNSU data channel in order to measure HW disturbances
- tested speed was approximately 5 kilosamples per second
- processing of overall NNSU hardware implementation has been simulated

Comparative study

HW disturbance error for Cherenkov Gamma-Ray Telescope (left) and Hadronic-tau separation (right) - separation is still acceptable



Estimated speed: $20 - 25 \times 10^6$ **events per second** - if

- best electronic components (but still commercial) will be used
- parallel HW implementation

Summary

- NNSU - general separation tool
- GA optimization of separation performance
- fitness functions of GA allow meet specific user defined requirements
- tested on simulated LHC (CERN) and D$\emptyset$ (FNAL, both simulated and measured) physical data sets
- improve cut based methods and comparable with standart TVMA ROOT methods
- potentially very fast HW implementation – Czech patent No. 306533