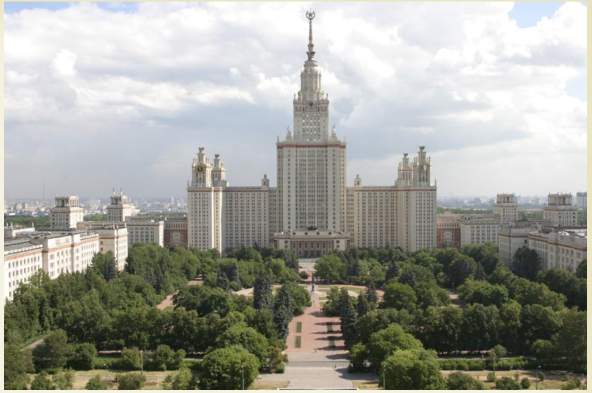
Optimization of the input space for the deep learning data analysis



Волков П.В, Дудко Л.В, Перфилов М.А, Воротников Г.А,
Черноded А. И., A. Shporin, D. Ovchinnikov
Lomonosov Moscow State University

QFTHEP 2019

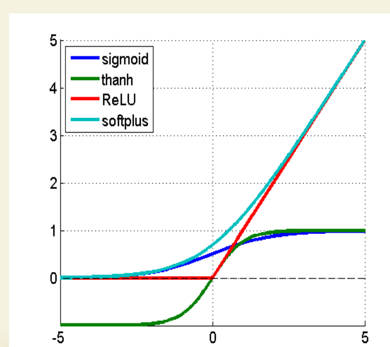
- Previous method to optimize the input space for the Neural Networks in HEP analysis
- Extension and general recipe to optimize the input space for the deep learning analysis in HEP

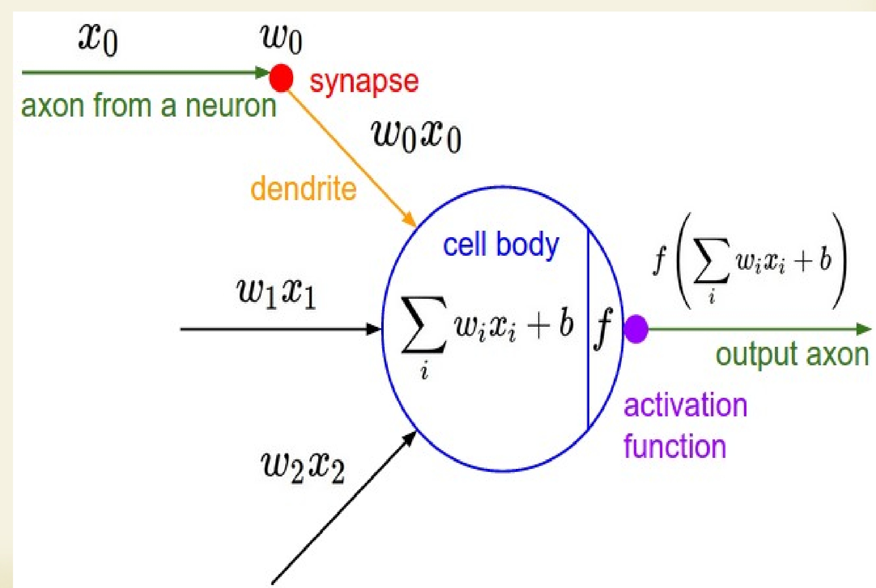
Mathematical basis of NNs

- 13th Hilbert problem (1900)
  whether its solution, x, considered as a function of the three variables can be expressed as the composition of a finite number of two-variable functions

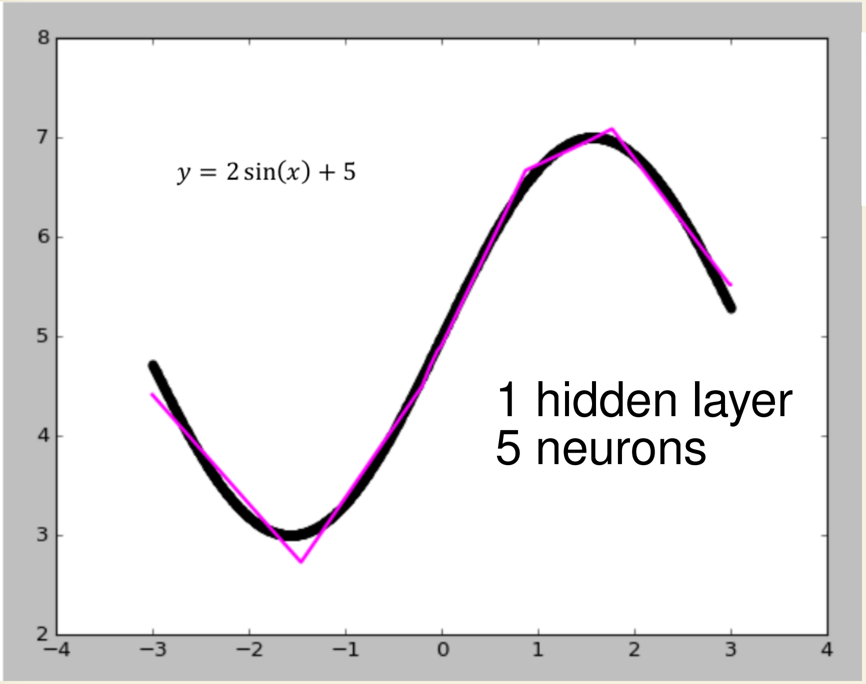
- Kolmogorov–Arnold representation theorem
  (or superposition theorem) (1957)
  if F is a multivariate continuous function, then F can be written as a finite composition of continuous functions of a single variable and the binary operation of addition

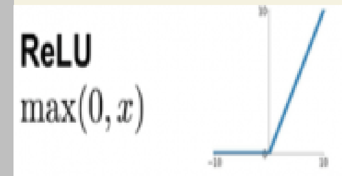
$$f(\mathbf{x}) = f(x_1, \dots, x_n) = \sum_{q=0}^{2n} \Phi_q \left(\sum_{p=1}^{n} \phi_{q,p}(x_p) \right)$$

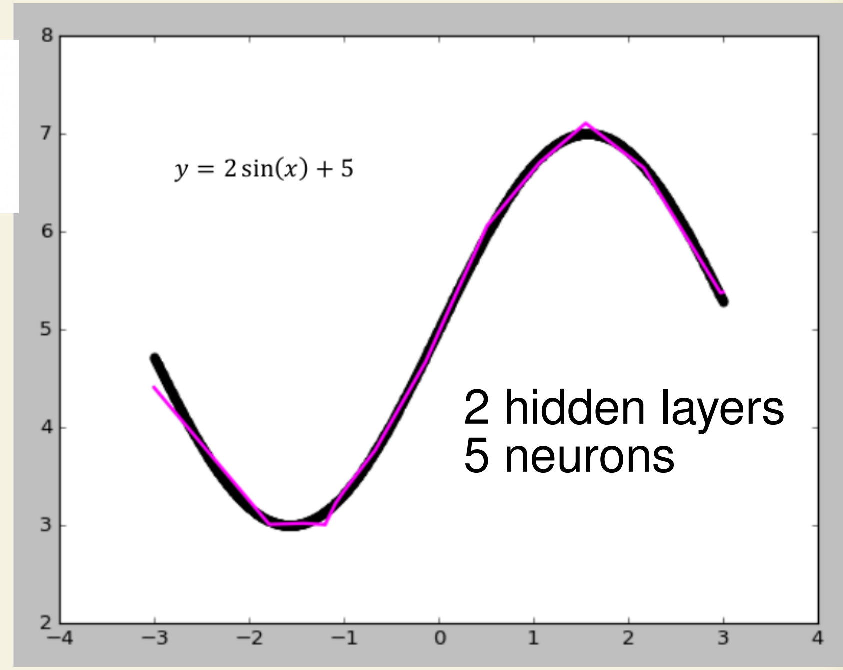


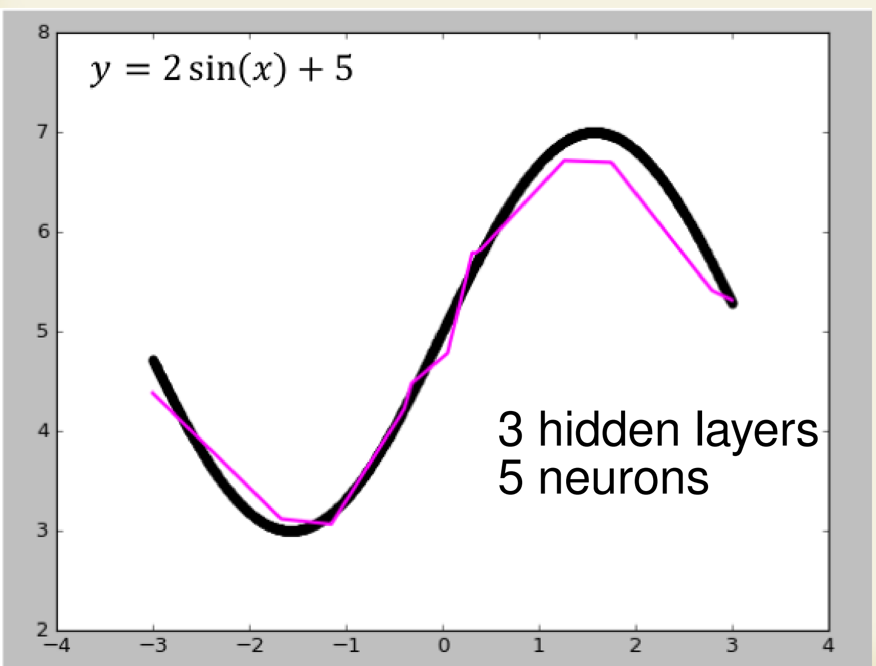


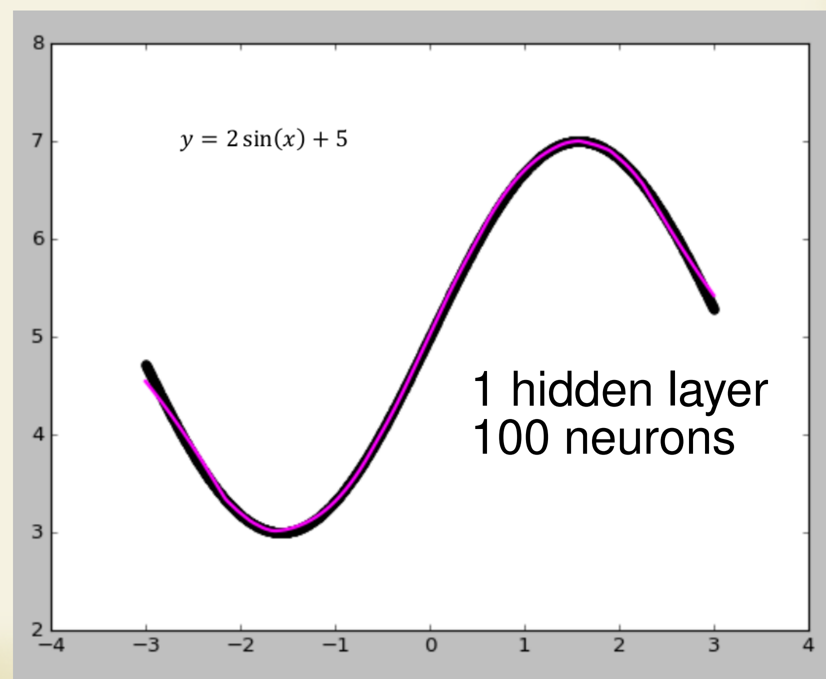
How it looks like in practice











Method of “optimal observables”

- **Provides general recipe how to choose most sensitive high-level variables to separate signal and background**
  - It is based on the analysis of Feynman diagrams (FD) contributing to signal and background processes
  - Distinguish **three classes** of sensitive variables for the signal and each of kinematically different backgrounds: **Singular** variables (denominators of FD), **Angular** variables (numerators of FD) and **Threshold** variables (Energy thresholds of the processes)
  - Set of variables can be extended with other type of information, like detector relative variables (jet width, b-tagging discriminant)

Described in different examples for the top and Higgs searches

  - E.Boos, L.Dudko, T.Ohl Eur.Phys.J. C11 (1999) 473-484
  - E.Boos, L.Dudko Nucl.Instrum.Meth. A502 (2003) 486-488
  - E.Boos, V.Bunichev, L.Dudko, A.Markina, M.Perfilov Phys.Atom.Nucl. 71 (2008) 388-393
- **Applied in different experimental analysis in D0 and CMS**
  - Phys.Lett. B517 (2001) 282-294 and other D0 publications
  - JHEP02(2017)028 , ...

Three Classes of Variables

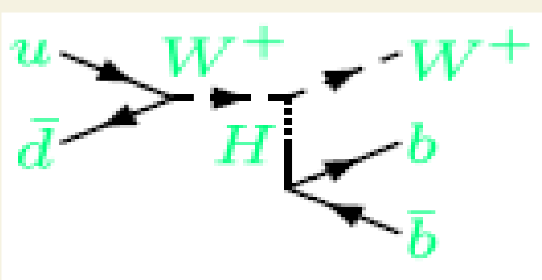
1) "Singular" Sensitive Variables

(denominator of Feynman diagrams)

Most of the rates of signal and background processes come from the integration over the phase space region close to the singularities. If some of the singular variables are different or the positions of the singularities are different the corresponding distributions will differ most strongly

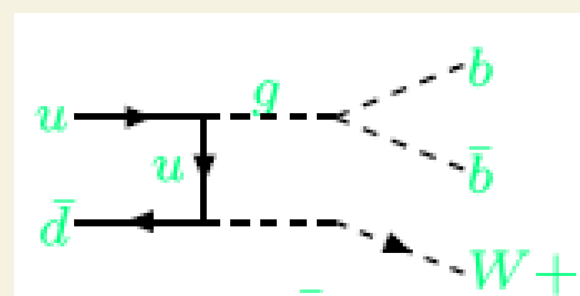
s-channel singularities

$$M^2_{f1,f2} = (p_{f1} + p_{f2})^2$$



t-channel singularities

$$\hat{t}_{i,f} = (p_f - p_i)^2 = -\sqrt{\hat{s}}e^{Y} p_T^f e^{-|y_f|}$$



Three Classes of Variables

2) "Angular" variables, Spin effects

(numerator of Feynman diagrams)

e.g.

$$\frac{1}{\Gamma_T}\frac{d\Gamma}{d(\cos\chi_\ell^W)} = \frac{3}{4}\,\frac{m_t^2\sin^2\chi_\ell^W + 2m_W^2\,\frac{1}{2}(1-\cos\chi_\ell^W)^2}{m_t^2 + 2m_W^2}$$

G. Mahlon, S. Parke Phys.Rev. D55 (1997) 7249-7254

3) "Threshold" variables

e.g. s_hat and Ht variables relate to the fact that various signal and background processes may have very different energy thresholds

Novel approach with deep learning neural networks (DNN)

- Starting from: Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). «A fast learning algorithm for deep belief nets.» Neural computation, 18(7), 1527-1554
- The main advantage of Deep NNs (many layers, neurons) is the possibility to analyze raw, not preprocessed, information.
- One of the first examples in HEP, DNN increases possible significance from 3.1σ up to 5.0σ in comparison with NN with high level variables:

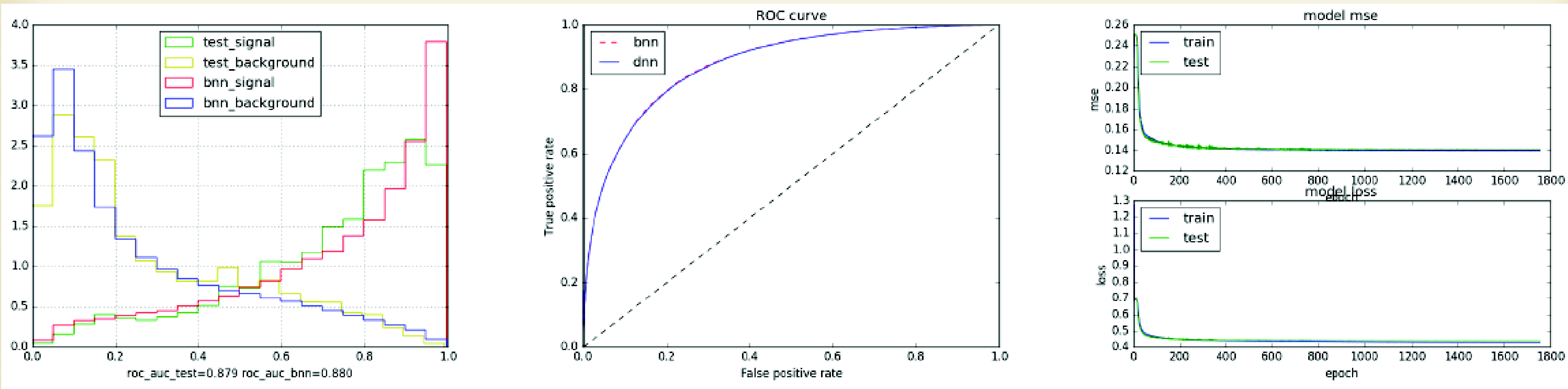
	Discovery significance		
Technique	Low-level	High-level	Complete
NN	2.5σ	3.1σ	3.7σ
DN	4.9σ	3.6σ	5.0σ

$$gg \to H^0 \to W^{\mp}H^{\pm} \to W^{\mp}W^{\pm}h^0$$

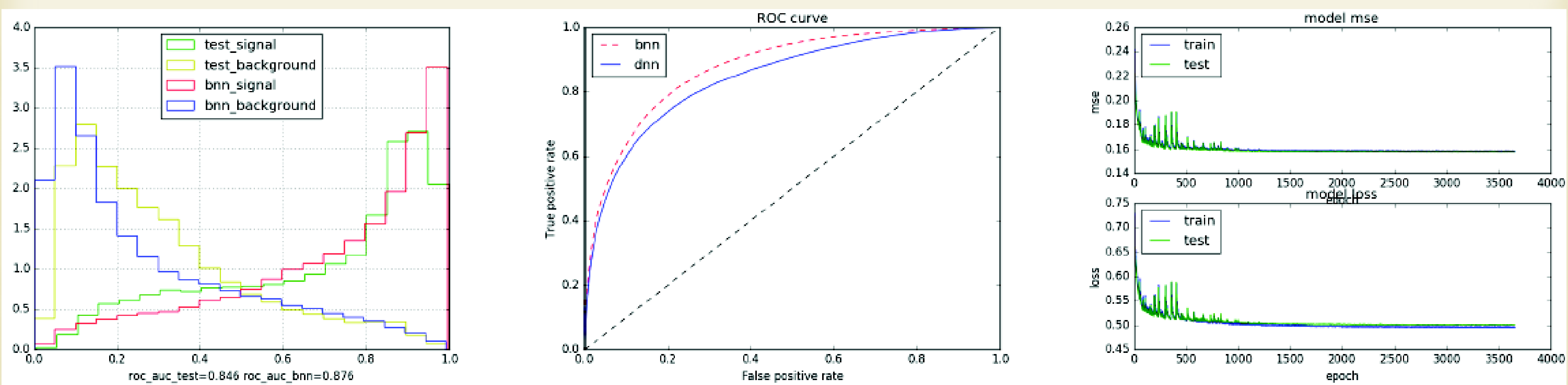
Nature Commun. 5 (2014) 4308

Lets try to add low-level (raw) variables

1. at the beginning, needs to compare BNN and DNN performance on the same set of high level variables. T-channel single top production as a signal and pair ttbar production as a background. The efficiency is almost the same (FBM package for BNN and Tensorflow for DNN):



2. The low level information about hard processes at colliders, is four-momenta of the final particles. The DNN efficiency with four-momenta is significantly worse than with optimized high-level variables:



Need to understand what is the general low-level information for the hard processes on colliders

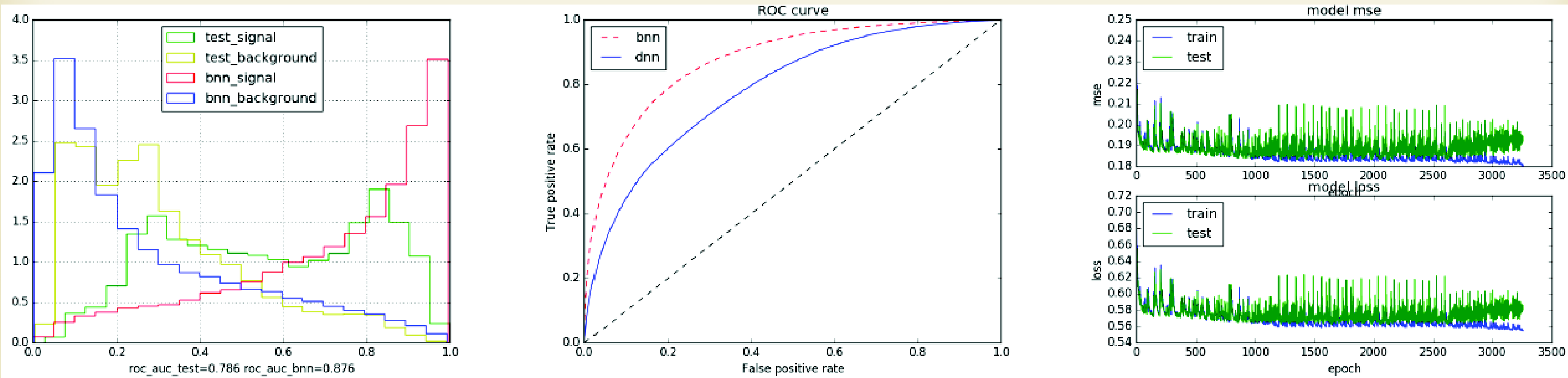
3. From the kinematic properties we know there are 3n-4 independent components, for the processes 2→n particles.
4. The matrix elements can be parametrized in terms of scalar products of four-momenta or in terms of Mandelstam variables s,t,u;
e.g. for the W' production [hep-ph/0610080]:

$$|M|^2 = V_{tb}^2 V_{ud}^2 (g_W)^4 \Bigg[\frac{(p_u p_b)(p_d p_t)}{(\hat{s} - m_W^2)^2 + \gamma_W^2 m_W^2} + \\ + 2 a_{ud}^L a_{tb}^L (p_u p_b)(p_d p_t) \frac{(\hat{s} - m_W^2)(\hat{s} - M_{W'}^2) + \gamma_W^2 \Gamma_{W'}^2}{((\hat{s} - m_W^2)^2 + \gamma_W^2 m_W^2)((\hat{s} - M_{W'}^2)^2 + \Gamma_{W'}^2 M_{W'}^2)} + \\ + \frac{({a_{ud}^L}^2 {a_{tb}^L}^2 + {a_{ud}^R}^2 {a_{tb}^R}^2)(p_u p_b)(p_d p_t) + ({a_{ud}^L}^2 {a_{tb}^R}^2 + {a_{ud}^R}^2 {a_{tb}^L}^2)(p_u p_t)(p_d p_b)}{(\hat{s} - M_{W'}^2)^2 + \Gamma_{W'}^2 M_{W'}^2} \Bigg]$$

$$|M|^2 = V_{tb}^2 V_{ud}^2 (g_W)^4 \Bigg[\frac{\hat{t}(\hat{t} - M_t^2)}{(\hat{s} - m_W^2)^2 + \gamma_W^2 m_W^2} + \\ + 2 a_{ud}^L a_{tb}^L \hat{t}(\hat{t} - M_t^2) \frac{(\hat{s} - m_W^2)(\hat{s} - M_{W'}^2) + \gamma_W^2 \Gamma_{W'}^2}{((\hat{s} - m_W^2)^2 + \gamma_W^2 m_W^2)((\hat{s} - M_{W'}^2)^2 + \Gamma_{W'}^2 M_{W'}^2)} + \\ + \frac{({a_{ud}^L}^2 {a_{tb}^L}^2 + {a_{ud}^R}^2 {a_{tb}^R}^2)\hat{t}(\hat{t} - M_t^2) + ({a_{ud}^L}^2 {a_{tb}^R}^2 + {a_{ud}^R}^2 {a_{tb}^L}^2)\hat{u}(\hat{u} - M_t^2)}{(\hat{s} - M_{W'}^2)^2 + \Gamma_{W'}^2 M_{W'}^2} \Bigg]$$

Lets check the scalar products

5. Use scalar products of four-momenta of all final particles. The efficiency is much worse than for high-level variables. The reason is the absence of four-momenta of the initial quarks.

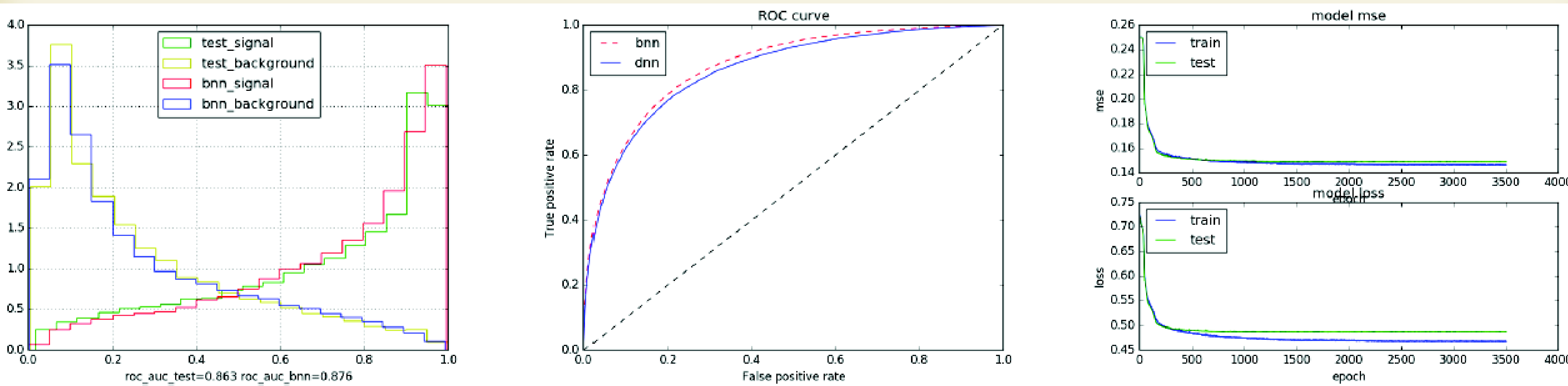


6. In the massless case we can use the following representation in terms of the final particles [Phys.Atom.Nucl. 71 (2008) 388-393]

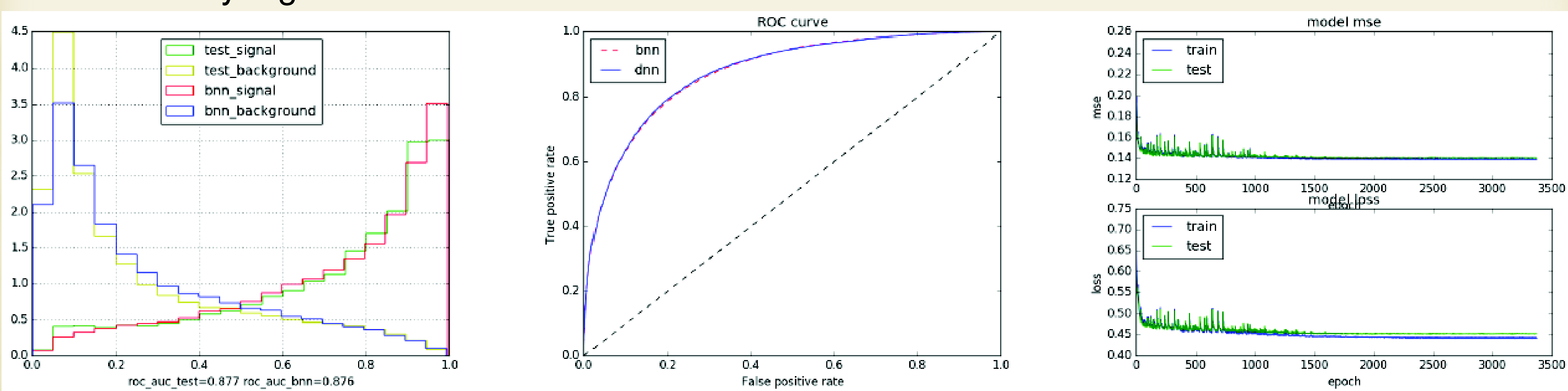
$$\hat{t}_{i,f} = (p_f - p_i)^2 = -\sqrt{\hat{s}}e^{Y} p_T^f e^{-|y_f|}$$

Put all together

7. We can combine the scalar products and four-momenta. The performance is almost the Same as for the very optimized, from physics point of view, set of high level variables.



8. Lets try to add everything (four-momenta, scalar products, high-level variables). The performance is maximal, most of the sensitive information is there and DNN algorithms train well very high dimensional NN



Summary

- We propose the general recipe to optimize the input space for the DNN implementation to analyze hard processes on colliders
- The general approach has almost the same performance as the very optimized set of high-level variables
- The recipe is simple, need to use the following classes:

  - scalar products of four-momenta of all final particles or/and Mandelstam variables

  - four-momenta of the final particles

  - some of the high-level variables to compensate the absence of four-momenta of the initial quarks.
- There are possibility and ideas to optimize the other steps of DNN analysis for the HEP tasks