Bayesian Neural Network and its application in ATLAS

Jiahang ZHONG *(Academia Sinica)*

BNN

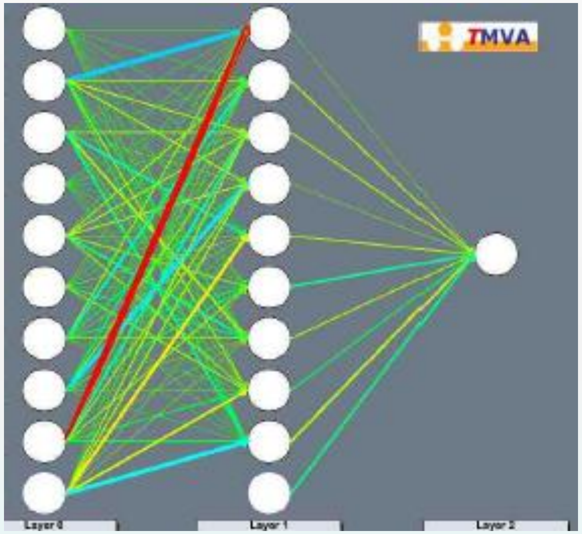
- Neural Network following Bayesian statistical interpretation

- Features:
  - Fitting of probability function
  - Regulator to avoid overtraining
  - Uncertainty estimation

- Based on the MLP algorithm
  - Configurable as options of the algorithm
  - Available in rel. 4.1.0 (ROOT 5.28)

Statistical interpretation of MLP

- Generic multivariate function approximator
  (Think of polynomial in 1D)
- Training:
  - Unbinned fitting with D={$\boldsymbol{x}_i$, t_i}
  - Cost function $(y(\mathbf{x},\boldsymbol{w})-t)^2$= -log(L), L=P(**D**|$\boldsymbol{w}$)
  - Min cost function=Max likelihood
- Prediction:
  - Obtained fitted value in the phase space
  - Classification: cut at the fitted value for discrimination



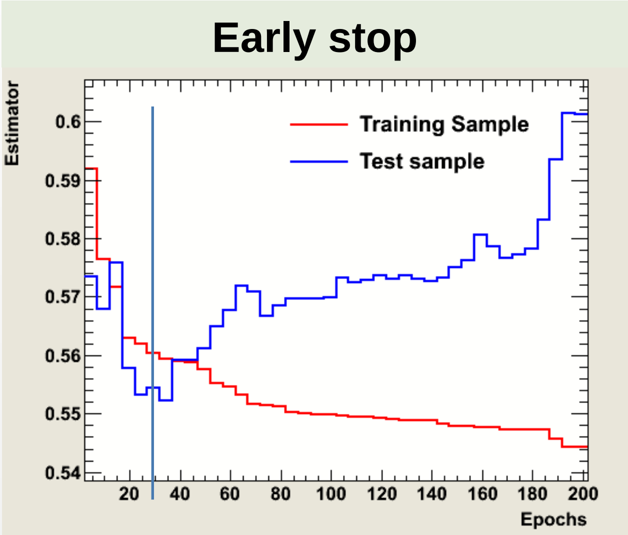
$a_j^{(1)} = \sum_l w_{jl}^{(1)} x_l$	$a_i^{(2)} = \sum_j w_{ij}^{(2)} x_j$
$h_j = f^{(1)}\left(a_j^{(1)}\right)$	$y_i = f^{(2)}\left(a_i^{(2)}\right)$
$f^{(1)}(x) = \tanh(x)$	$f^{(2)}(x) = x$

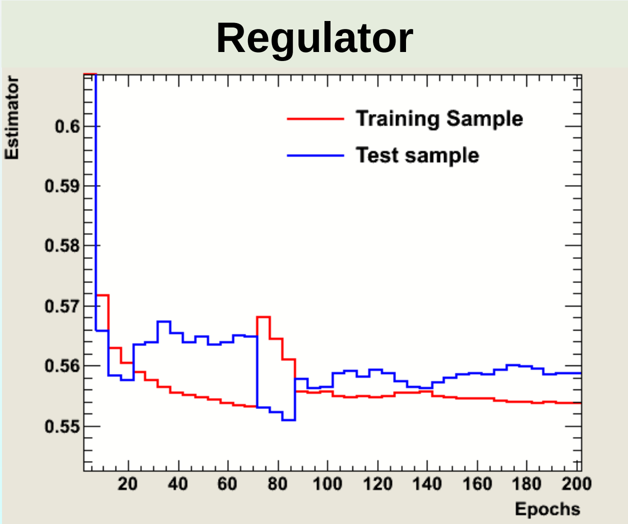
Bayesian implementation (I)

- Fitting probability function
  - Probability $y \in [0,1]$ is more useful than proposition $y \in \{0,1\}$
  - Not for probability density function $\int y d\mathbf{x} = 1$ => PDERS
  - To constrain y between 0 and 1, transform y by
    $f^{(2)}(x) = 1/(1+\exp(-x))$
  - To make y=P(t==1).
    Bernoulli likelihood⇔ Cross entropy cost function
    $$-\log(L) = \sum_i \left(-t_i * \log y(x_i) - (1-t_i) * \log(1-y(x_i))\right)$$
  - MLP option: EstimatorType=CE

Bayesian implementation (II)

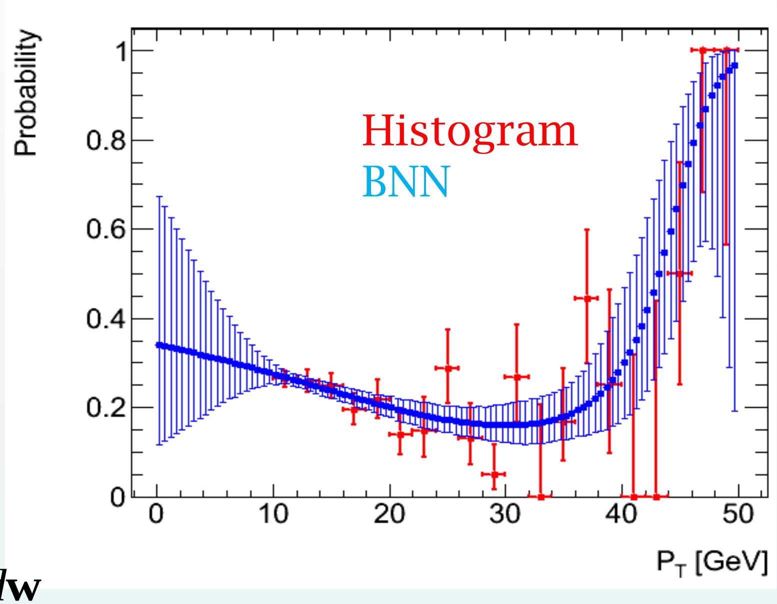
- Regulator to avoid overtraining
  - Overtraining is caused by excessive complexity of NN
  - A prior knowledge prefer "simpler" model ⇔ small w
    - Gaussian prior $-\log(P(\mathbf{w})) = \sum_m (\alpha_m \times w_m^2)$
    - P(*w*|**D**)=P(**D**|*w*)P(*w*). Add to the log likelihood (cost function)
  - Optimize α during training ⇔ Adaptive complexity control
  - MLP option: UseRegulator





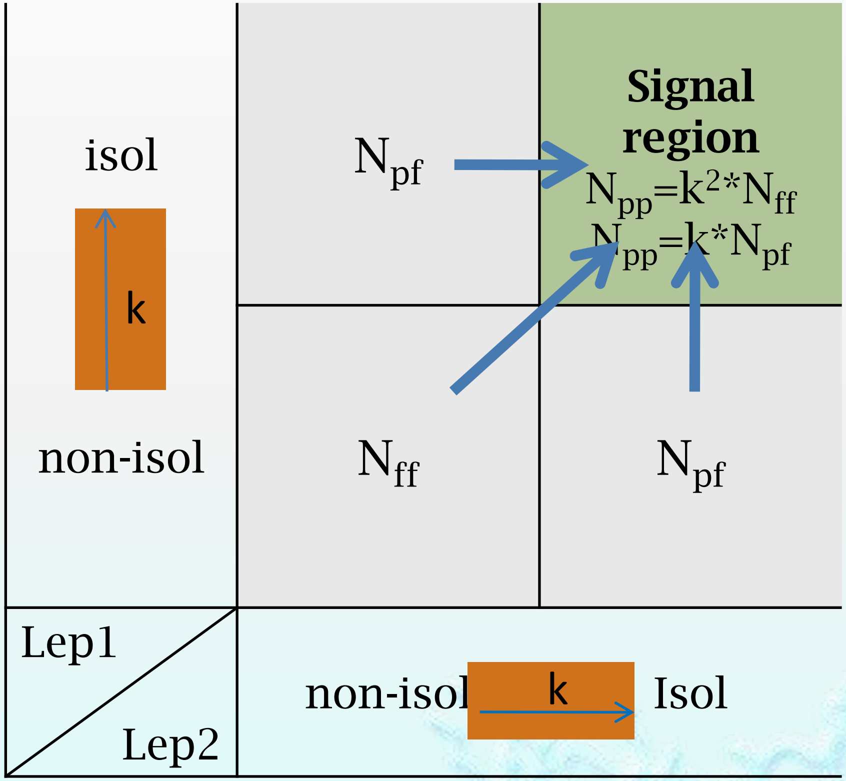
Bayesian implementation (III)

- Uncertainty estimation
  - Training:
    - Most probable value $\boldsymbol{w}_{\mathrm{MP}}$
    - P(***w***|**D**)
      Probability of other w
  - Prediction
    - Probability
      $$P(\mathrm{y}\,|\,\mathbf{x'}) = \int P(\mathrm{y}\,|\,\mathbf{x'},\mathbf{w}) * P(\mathbf{w}\,|\,\mathbf{D})d\mathbf{w}$$
    - Uncertainty of *y*
    - Avoid excessive extrapolation (non-trivial for multivariate analysis)
  - MLP option: CalculateErrors (interface need to be completed)



Application in ATLAS analysis(I)

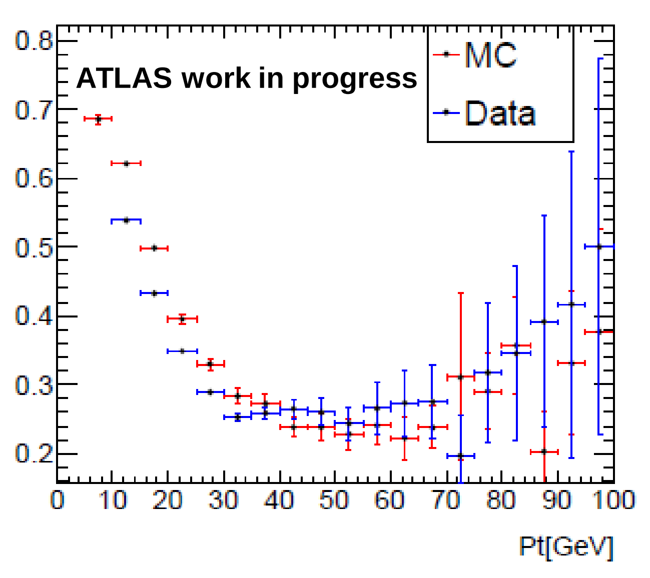
- Isolated di-lepton search (Exotic/SUSY)
  - Double-fake bkg: bb/cc
  - True-fake bkg: W/Z+jet
- Data-driven estimation
  - Extrapolate from non-isolated control samples to signal region
  - Weight each event by the (product of) pass/fail ratio k of the failed muon(s)

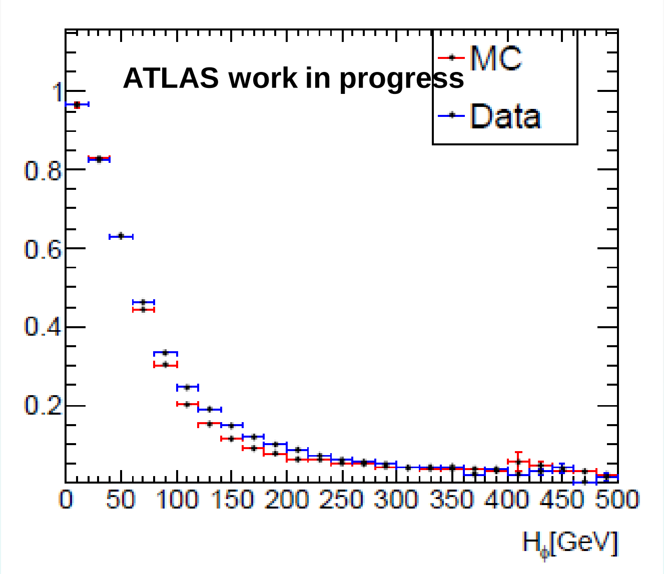


$$k = \frac{P(\text{pass})}{1 - P(\text{pass})}$$

Application in ATLAS analysis(II)

- P(pass) is obtained from a single-muon control sample
- Strong dependence of P(pass) over kinematics ***x***



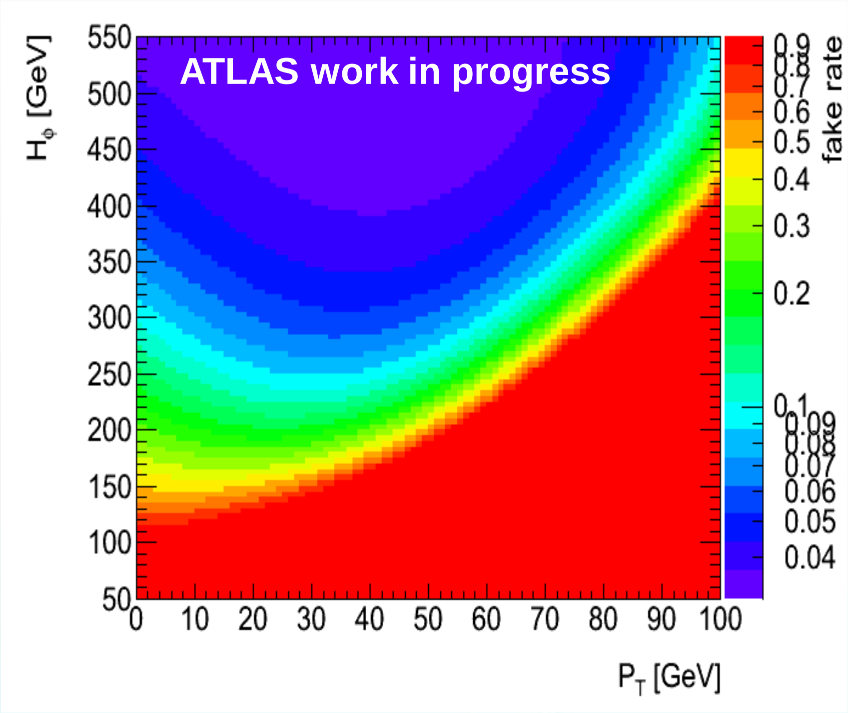


  - Different distributions between control samples
  - Correlation between the two leptons
- Proper parameterization of P(pass|***x***) is crucial

Application in ATLAS analysis(III)

- The BNN is used for unbinned fitting of P(pass|***x***)

- Training sample:
  - Single muons in background control region,
  - **D**={$\boldsymbol{x_i}$, t_i}
    - **x**={p_T , H_ϕ}
    - t=1 (pass isolation cut)
      t=0 (fail isolation cut)
  - Could add more variables for parameterization: Eta, n_vertex

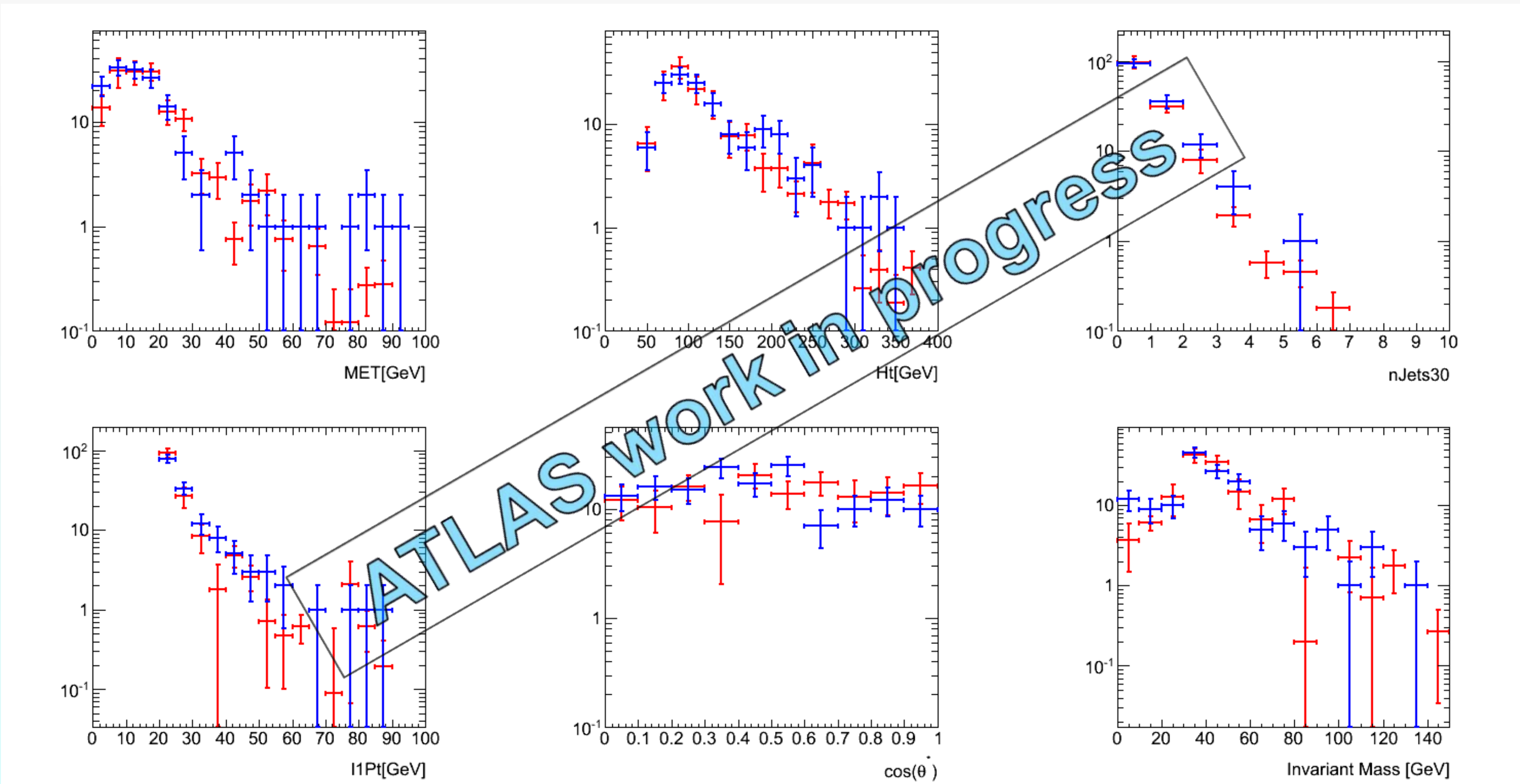


2D fake rate fitted by BNN

Data-driven background estimation(V)

- Number of observed events= 149
- Number of predicted background yield= 141 ± 15

Observation
Prediction



Summary

- The Bayesian implementations
  - As easy as polynomial fitting !!
  - Fitting of probability function
  - Regulator to avoid overtraining
  - Uncertainty estimation
- Probability fitting in data analysis
  - Shows good performance in one ATLAS application
  - Open the door to more delicate study
  - Could be applied to many other analysis: Trigger/Charge/ParticleID