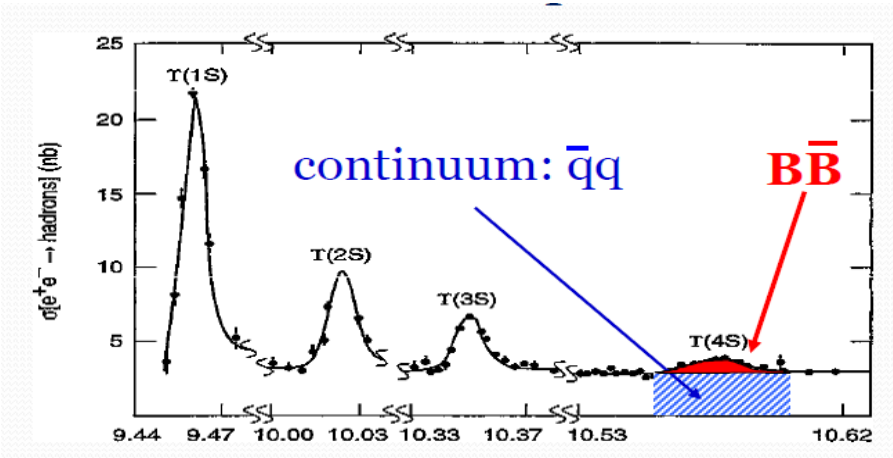
NEURAL NETWORK

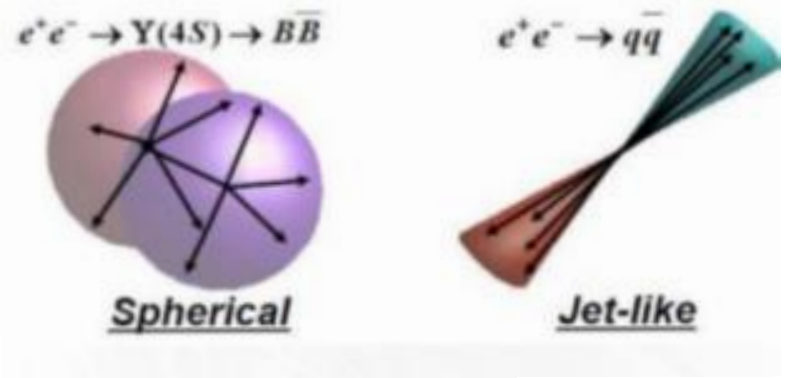
Subhashree Mohanty
Utkal University

Post CKM School
December 3-7, 2016

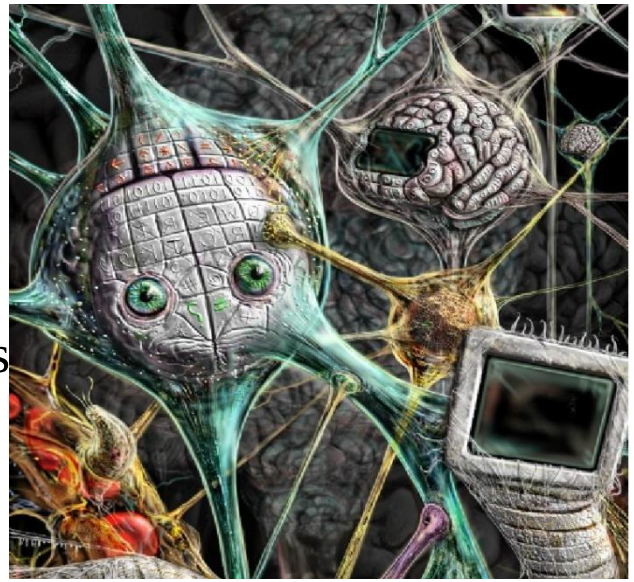
Introduction

- Every analysis needs to maximize S/B ratio or distinguish shapes of signal and background distributions.
- The optimal way is to prepare a classifier to separate signal from background.
- There are so many common analysis techniques used to separate signal from background : Cut-based, Likelihoods, Decision trees, **Neural Networks**, Fisher etc.
- There is large contribution from $e^+e^- \rightarrow q\bar{q}$ continuum process under Y(nS) resonances.
- Continuum suppression is important for analysing B decays especially for analysing decays which have small signal yields.

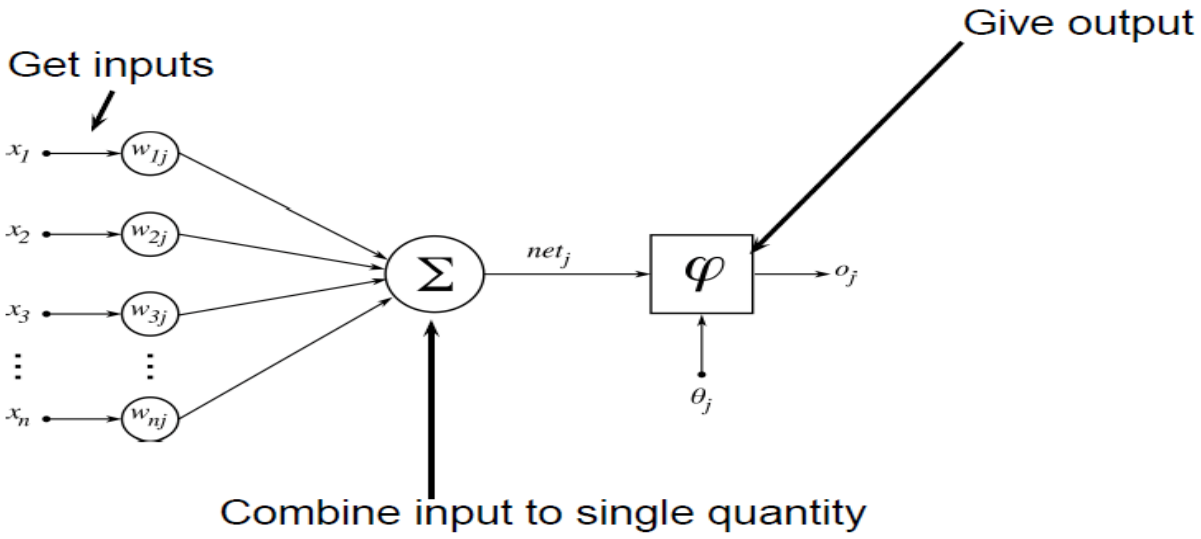


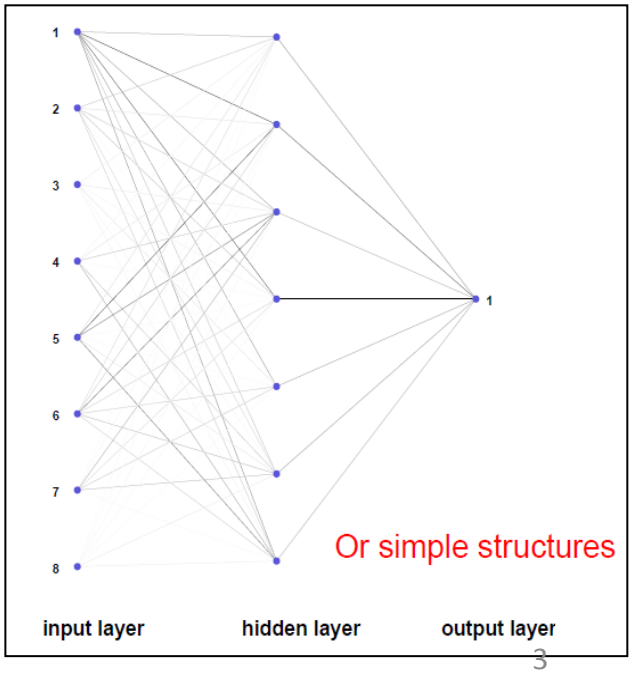


Introduction



- Neural Network technique is analogy with brain
- Get input from other neurons
- Knowledge stored in the weights of the connections
- Merge inputs ($S = \Sigma w_i x_i$)
- Send output

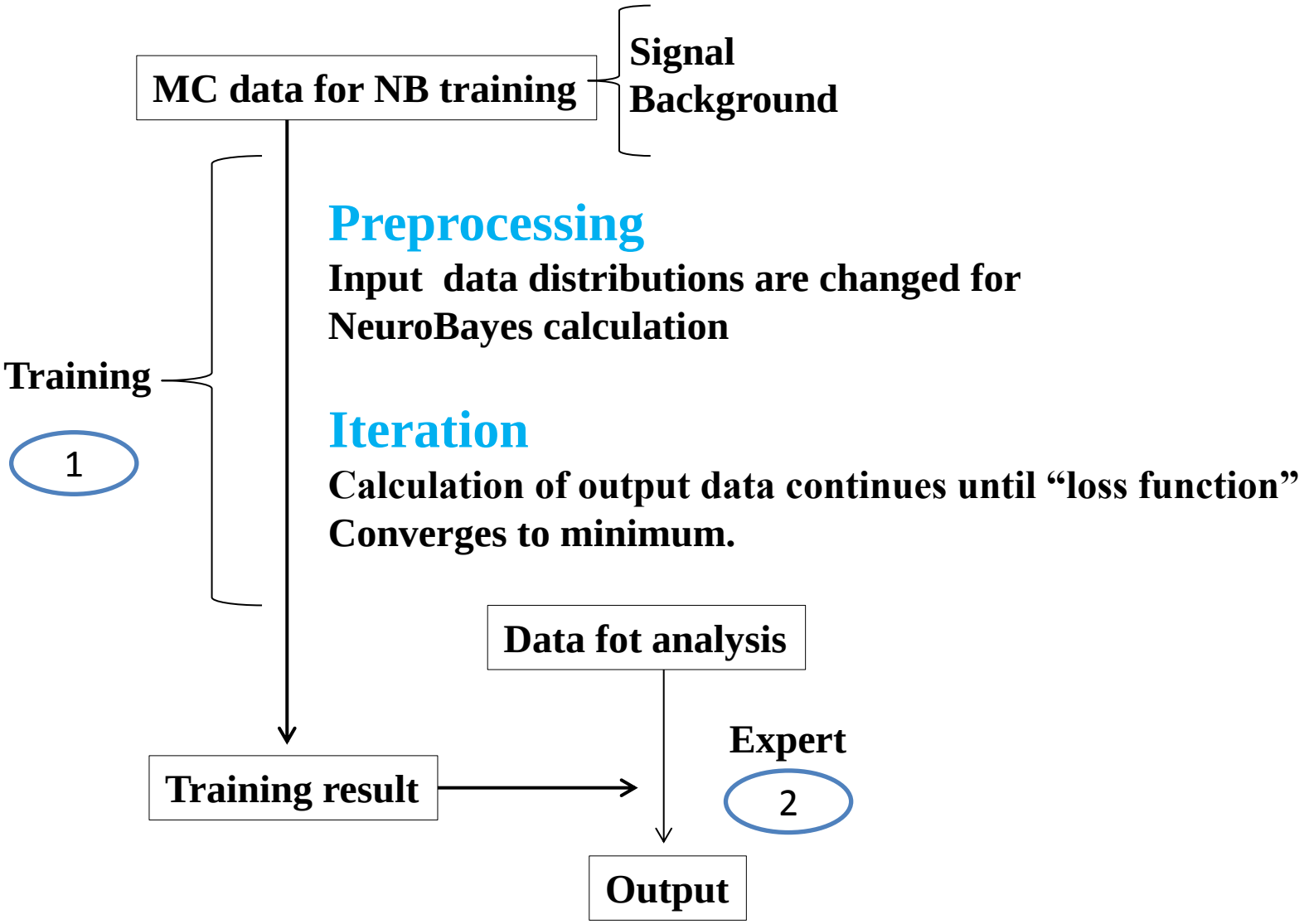




Motivation

- Easily can deal with complicated correlations among inputs
- Risk of overtraining extremely low due to Bayesian regularisation
- Extremely robust due to sophisticated and automatic pre-processing
- Allow as well estimation of probability densities
- Can evolve to complex non-linear model
- Easy to handle
- Minimal risk to get ‘stuck’ in bad local minimum
- Low storage of CPU and RAM resources

Overview of the Process



Continuum Fighting Variables

1. **cosb : cosine of the angle between the B momentum and the Z axis**

2. **cosbt : cosine of the angle between the B thrust and the Z axis**

3. **costhr : cosine of the angle between the B and the non-B thrust axis**

4. **LR(RooKSFW) : Likelihood ratio for RooKSFW**

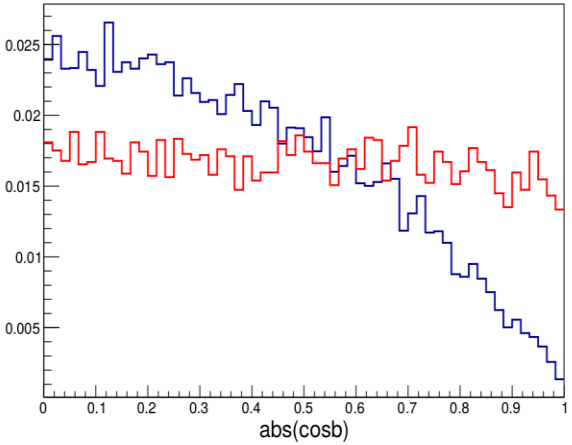
5. **R2 : Ratio of the 2nd and the 0th Fox-Wolfram moments**

6. **Δz : vertex separation along the z axis between the B candidate and the remaining tracks.**

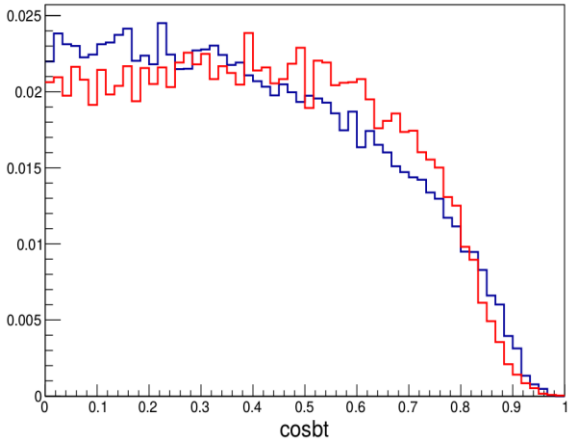
7. **qr : flavor tagging Information**

Continuum Fighting Variables

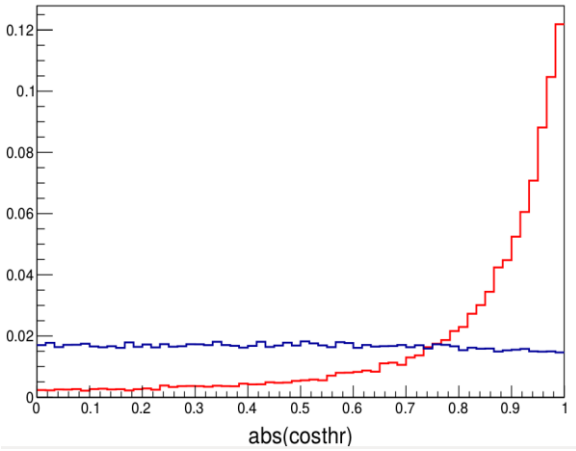
Cosine of angle between the B momentum and z axis



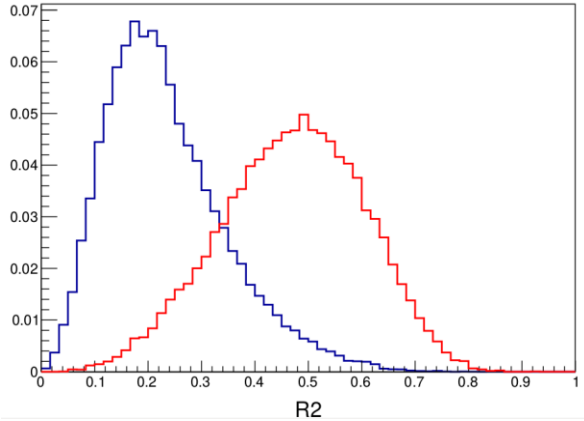
Cosine of angle between the B thrust and z axis



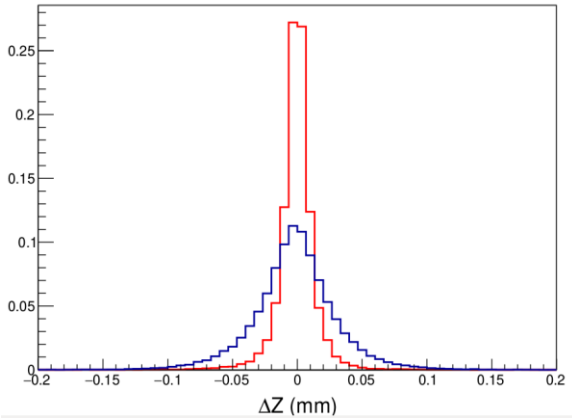
Cosine of angle between the B and non B thrust axis



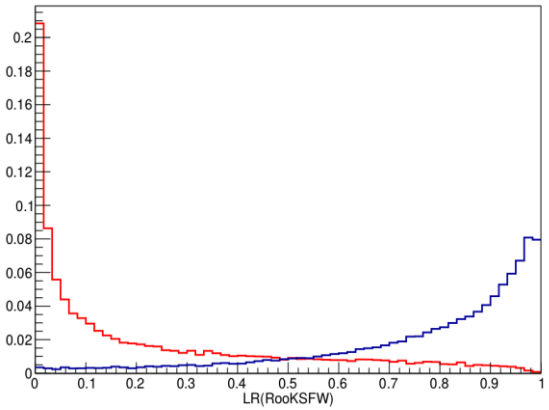
Ratio of the 2nd and 0th Fox Wolfram moments



ΔZ



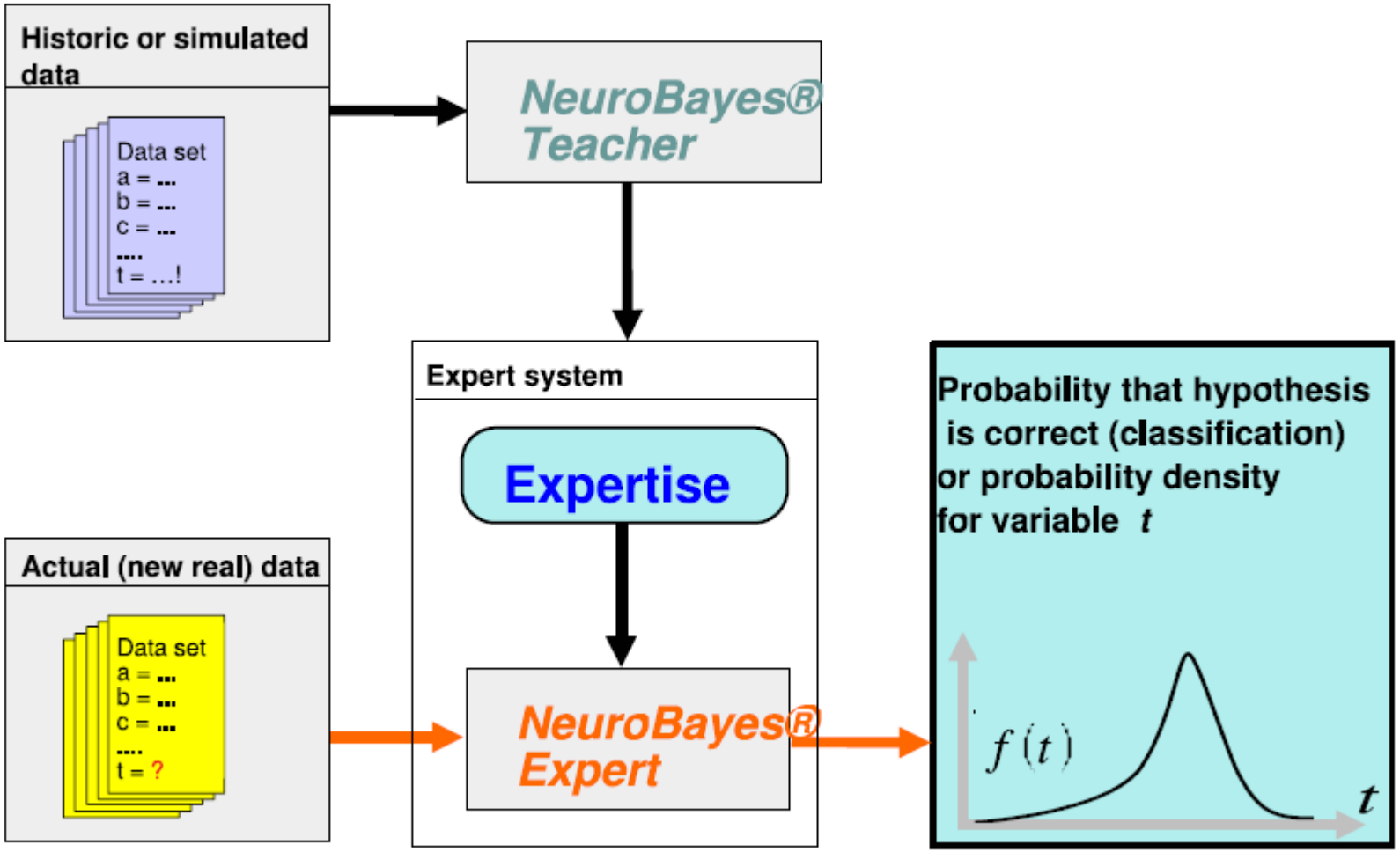
LR(RooKSFW)



Blue : Signal

Red : Continuum background

NN work flow


Historic or simulated data
Data set
a = ...
b = ...
c = ...
....
t = ...!
NeuroBayes®
Teacher
Expert system
Expertise
Actual (new real) data
Data set
a = ...
b = ...
c = ...
....
t = ?
NeuroBayes®
Expert
Probability that hypothesis is correct (classification) or probability density for variable t
f(t)
t

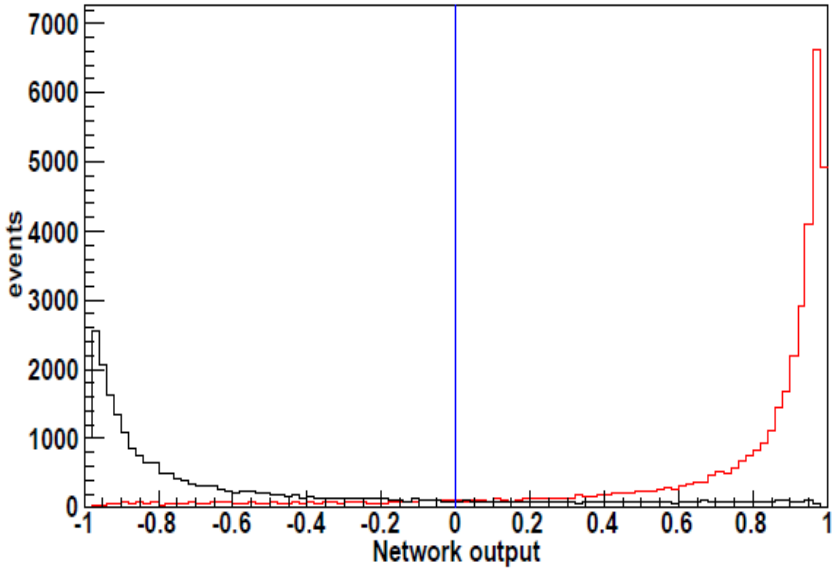
Training

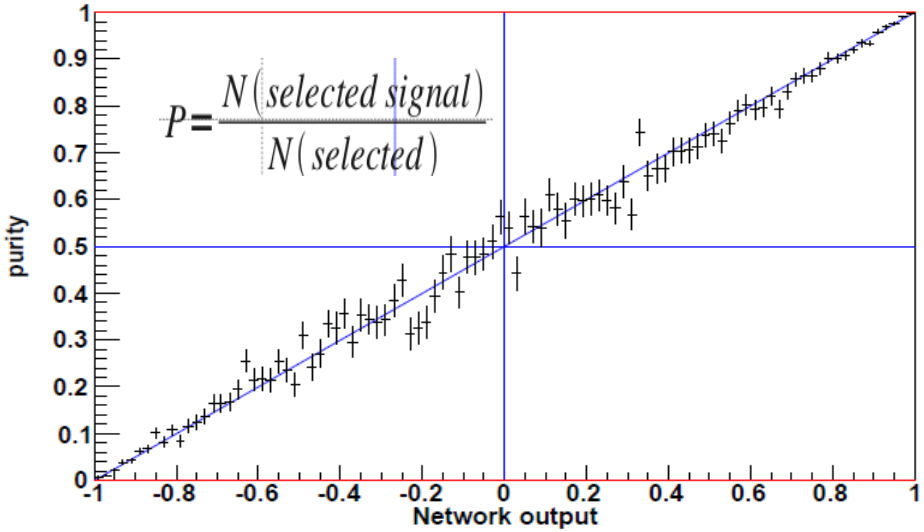
- Before using Neural network, need to find best set of weights which give minimised loss function
- Training is a process where we search for set of weights which minimise loss function.
- Possible loss function : Sum of quadratic deviation or entropy(Maximum likelihood)
- Most important part using neural networks
- Bad training can result in bad or even wrong solution
- Technically need : KEKCC account
  Setup belle environment
- Required working Files are located in this location :
- **/home/belle/subha/public/NB_PostCKMSchool/**
- In **src/nb_training.cc**
  - ❑ Setting of the NeuroBayes (Include header files, provide signal & bkg data sample)
  - ❑ Define the output expertise **.nb** file
  - ❑ Adding Input variables
  - ❑ Filling Input Variables
- Excute **./train.bash**
- Check the result in analysis_sorted.ps file and analysis.ps

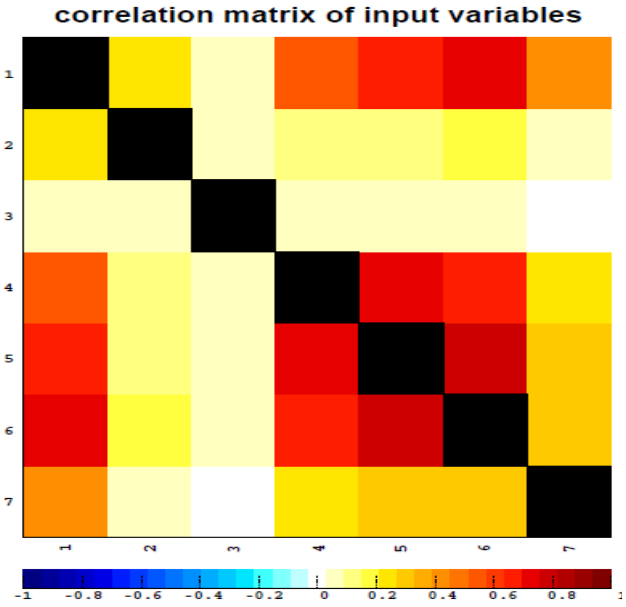
Training

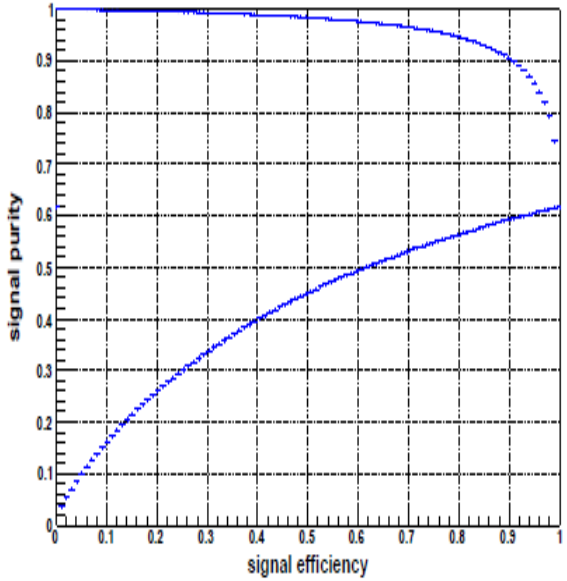
In analysis.ps and analysis_sorted.ps file :

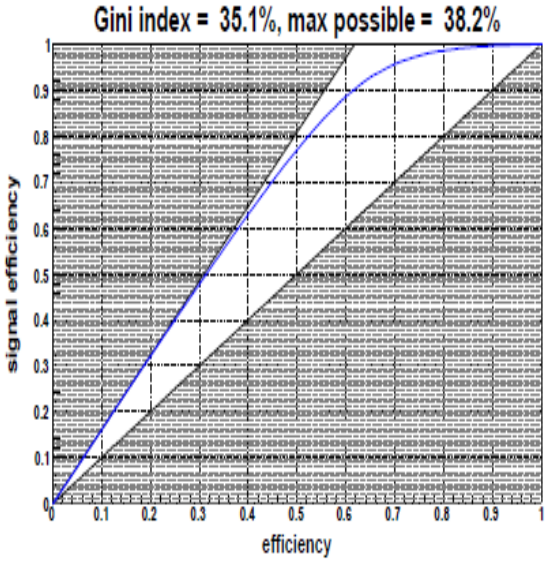
- useful histograms to postscript file for checks
- Loss function dependence on iteration
- NN output for signal and background
- Purity vs. NN output, purity vs. efficiency
- Correlation matrix
- Plots for individual variables


events
Network output


$P = \frac{N(\text{selected signal})}{N(\text{selected})}$
purity
Network output


correlation matrix of input variables


signal purity
signal efficiency


Gini index = 35.1%, max possible = 38.2%
signal efficiency
efficiency

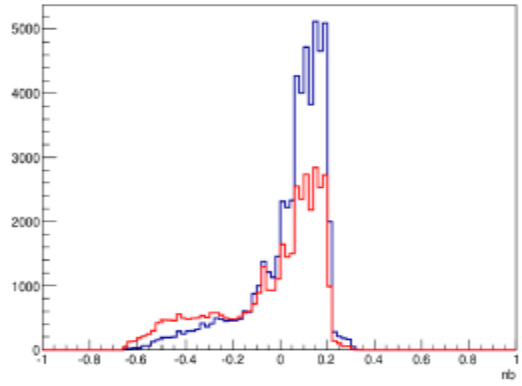
Expert

- Apply selection on independent sample
- In **src/nb_expert.cc**
  - ❑ Load sample on which you wish to apply training
  - ❑ Load the expertise .nb file
  - ❑ Create new root file to save NB Output
- In this process, you can use the same root files in training and also apply the expertise.
- But In first case even events and in second case odd events will use, making them statistically independent.
- Excute **./expert.bash**
- Look at the result for independent sample

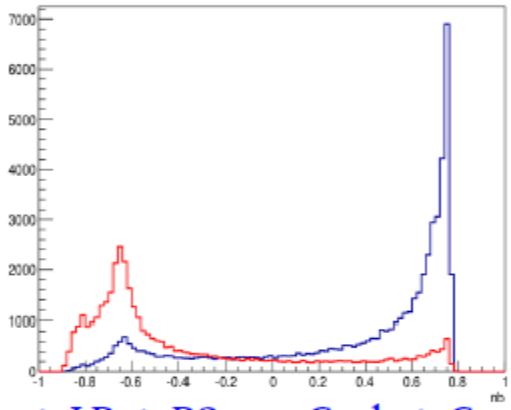
- We can study NN performance with various combination of event shape variables & select the one that gives higher signal efficiency while rejecting most of the continuum background.

NN outputs of different combinations of event shape variables

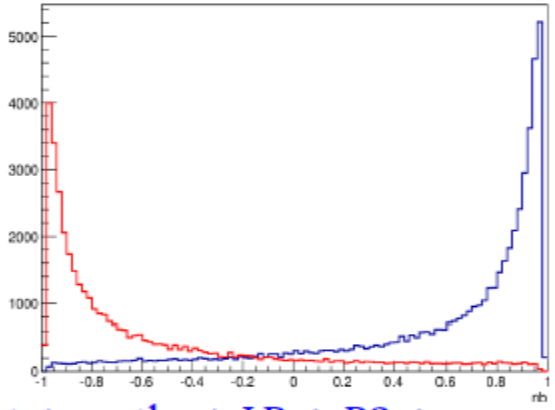
Cosb + Cosbt



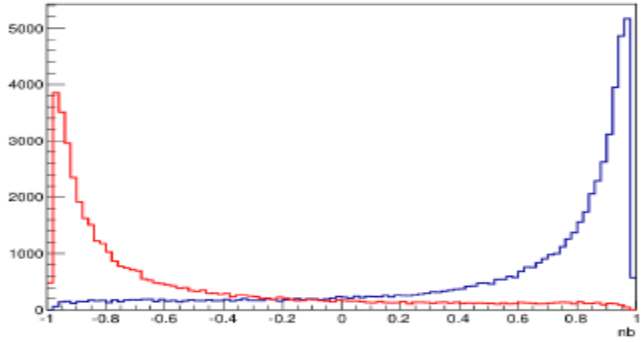
Cosb + Cosbt + costhr



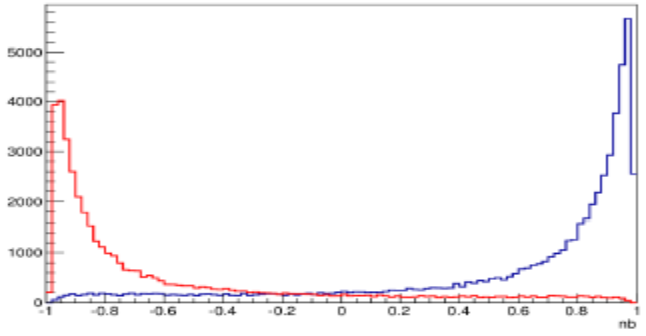
Cosb + Cosbt + costhr + LR



Cosb + Cosbt + costhr + LR + R2



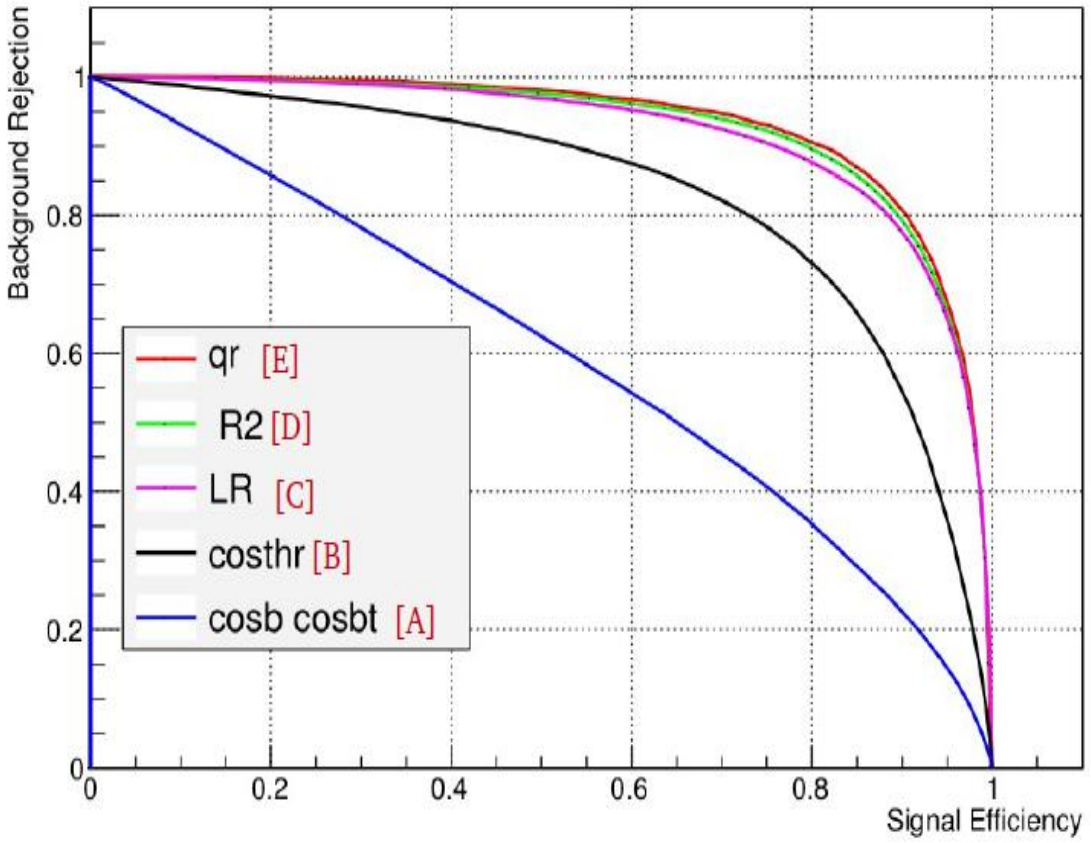
Cosb + Cosbt + costhr + LR + R2 + qr



Blue : SignalMC
Red : Continuum qq-bar MC

8

Signal Efficiency vs Background Rejection



9

- Compare different NN performance with various combination of event shape variables
- select the one that gives higher signal efficiency while rejecting most of the continuum background.

Thank you

Backup