Using Machine Learning to Increase the significance of our measurement of WW$_{\gamma}$

Caroline Doctor

OU Summer REU 2020

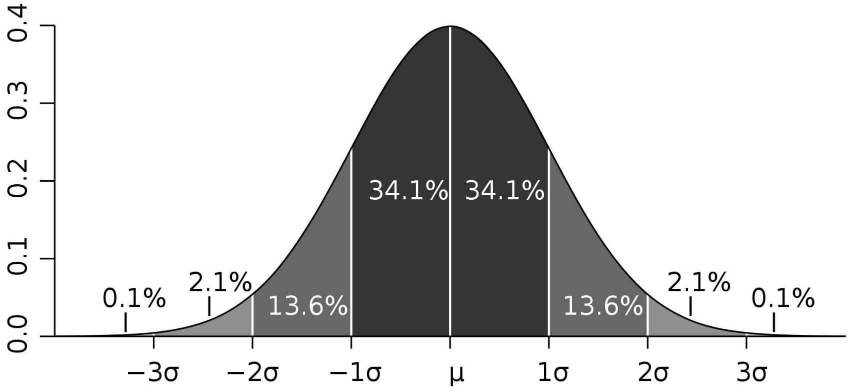
QUICK REMINDER

- significance = sigma value

  significance of 5 σ = 5 standard deviations away from mean; 1 in 3.5 million chance of being a fluctuation

- significance = number of most likely signal events / square root of number the most likely background events

$$\sigma = \frac{s}{\sqrt{b}}$$



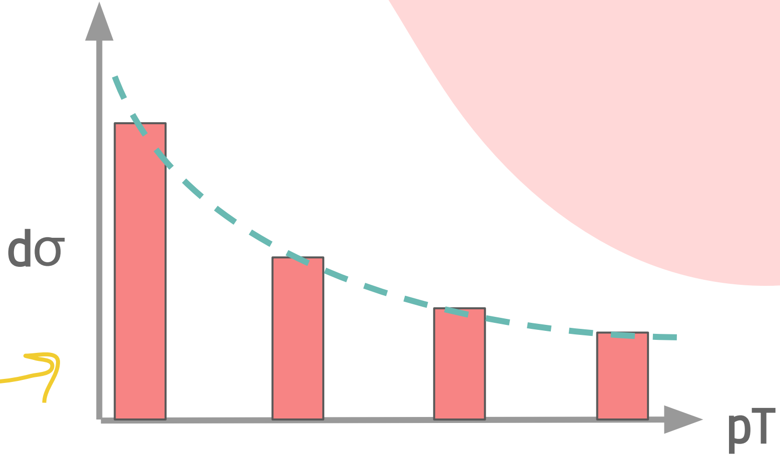
WHY INCREASE SIGMA?

$$N = \sigma L \rightarrow \sigma = \frac{N}{L}$$

N = # of events
σ = cross-section
L = luminosity

Example:
100 signals total

- 60 with pT between 20-30
- 20 with pT between 30-40
- 10 with pT between 40-50
- 10 with pT > 50



MY PROJECT

- I give the BDT variables and set the parameters
- It sorts the collision events
- a second code will translate the BDT results into a variable called “classifier” ranging from 1 to -1

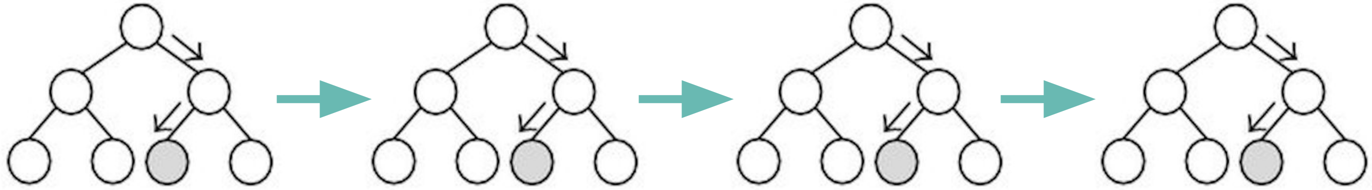
  -1 = most likely background

  1 = most likely signal

- We can cut on the classifier variable

BOOSTED DECISION TREES

- boosted decision trees are sequential and seek to improve in the next tree what the previous tree didn’t do well
- “Each successive tree uses the residuals of the previous tree”



NOT ALL BACKGROUNDS ARE EQUAL

- 2 types of backgrounds: ones containing real photons, and ones containing ‘fake’ photons

FAKE PHOTON BACKGROUNDS

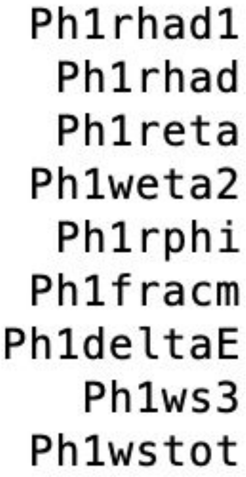
- ZZ
- Diboson
- Zjets
- Wjets
- ttbar
- tW

REAL PHOTON BACKGROUNDS

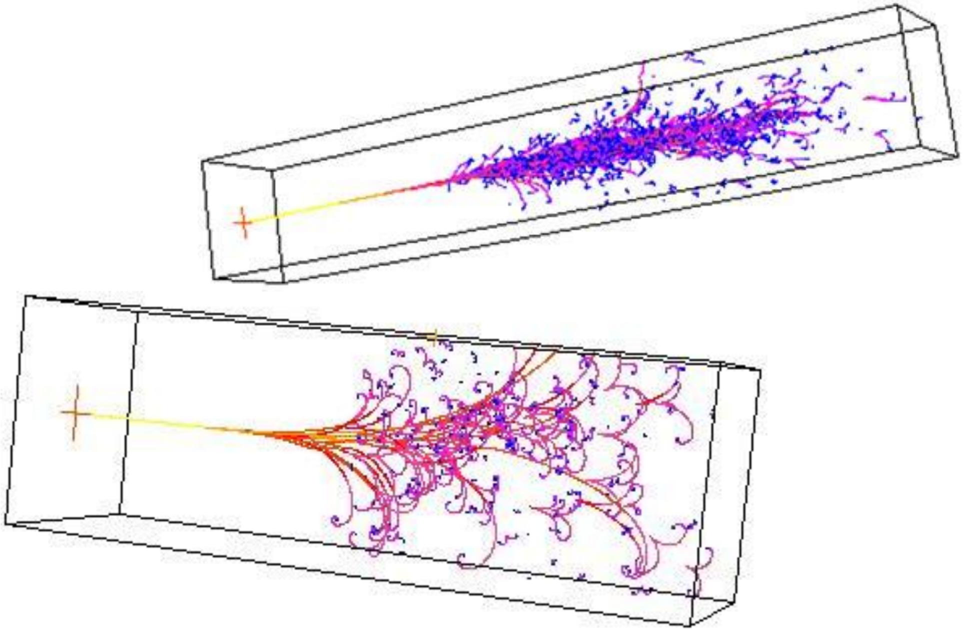
- Wy
- tty
- Zy

QUICK VARIABLE RUNDOWN

- Photon ID Variables:
- just various shower shapes in the calorimeter.



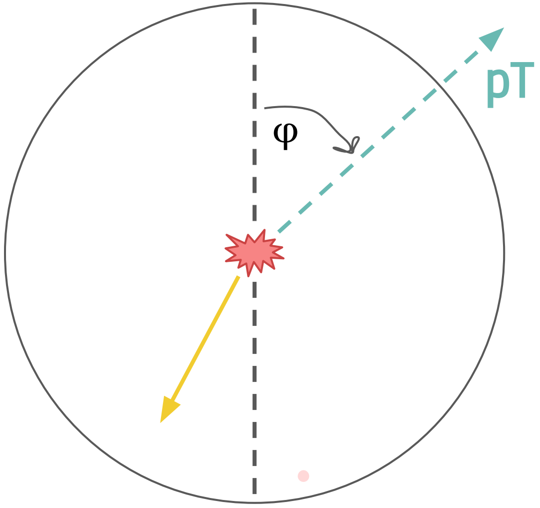
```
Ph1rhad1
 Ph1rhad
 Ph1reta
Ph1weta2
 Ph1rphi
Ph1fracm
Ph1deltaE
  Ph1ws3
Ph1wstot
```



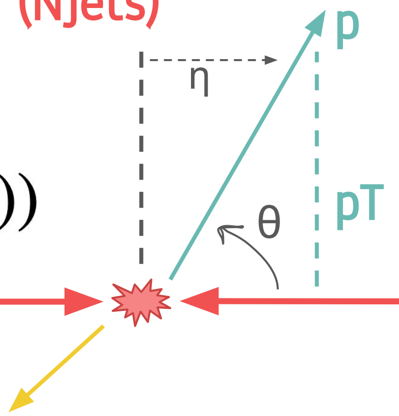
QUICK VARIABLE RUNDOWN

- Physics variables the detector measures:
- missing transverse energy (MET)
- invariant mass of the two leptons (mll)
- Eta (η)
- transverse momentum (pT)
- number of jets produced (Njets)

transverse view



$$\eta = -\,ln(tan(\tfrac{\theta}{2}))$$



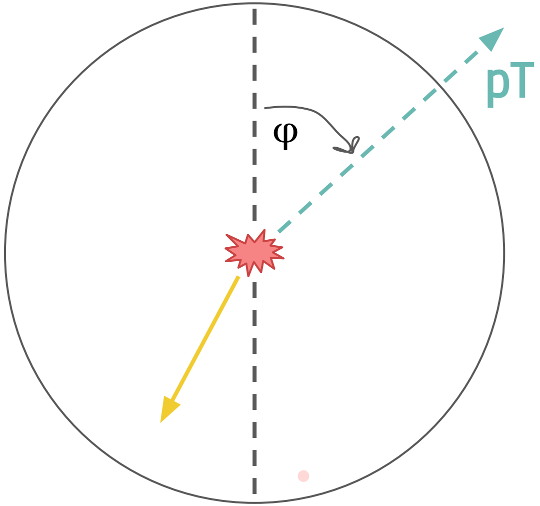
$$p = \frac{pT}{sin\theta}$$

beam view
/side-view

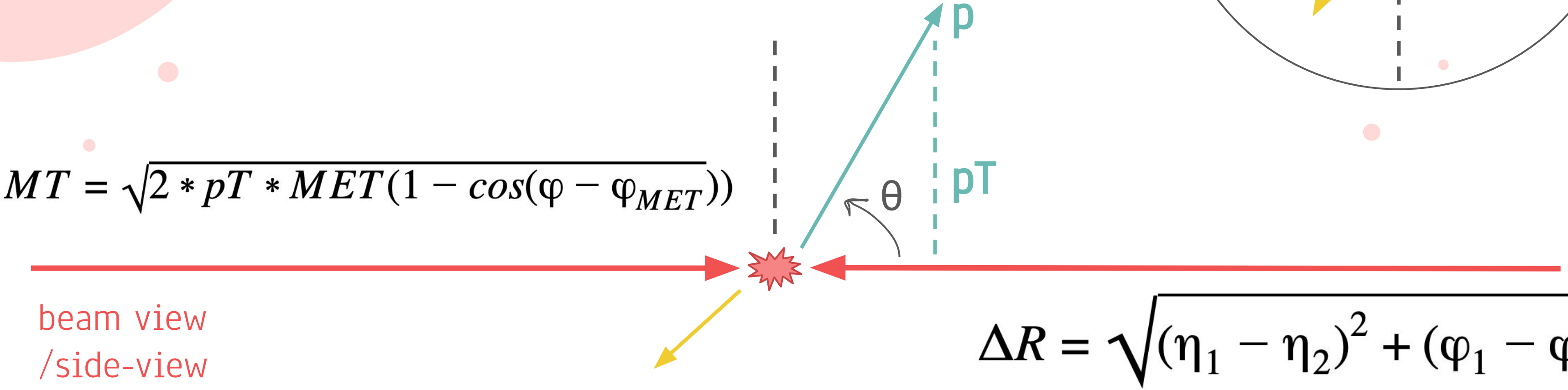
QUICK VARIABLE RUNDOWN

- Physics variables that are functions of others:
  - theta of each particle (θ)
  - momentum of each lepton and the photon (p)
  - the transverse mass (MT)
  - the Δ R between all three particles

transverse view



$$MT = \sqrt{2 * pT * MET(1 - cos(\varphi - \varphi_{MET}))}$$



beam view
/side-view

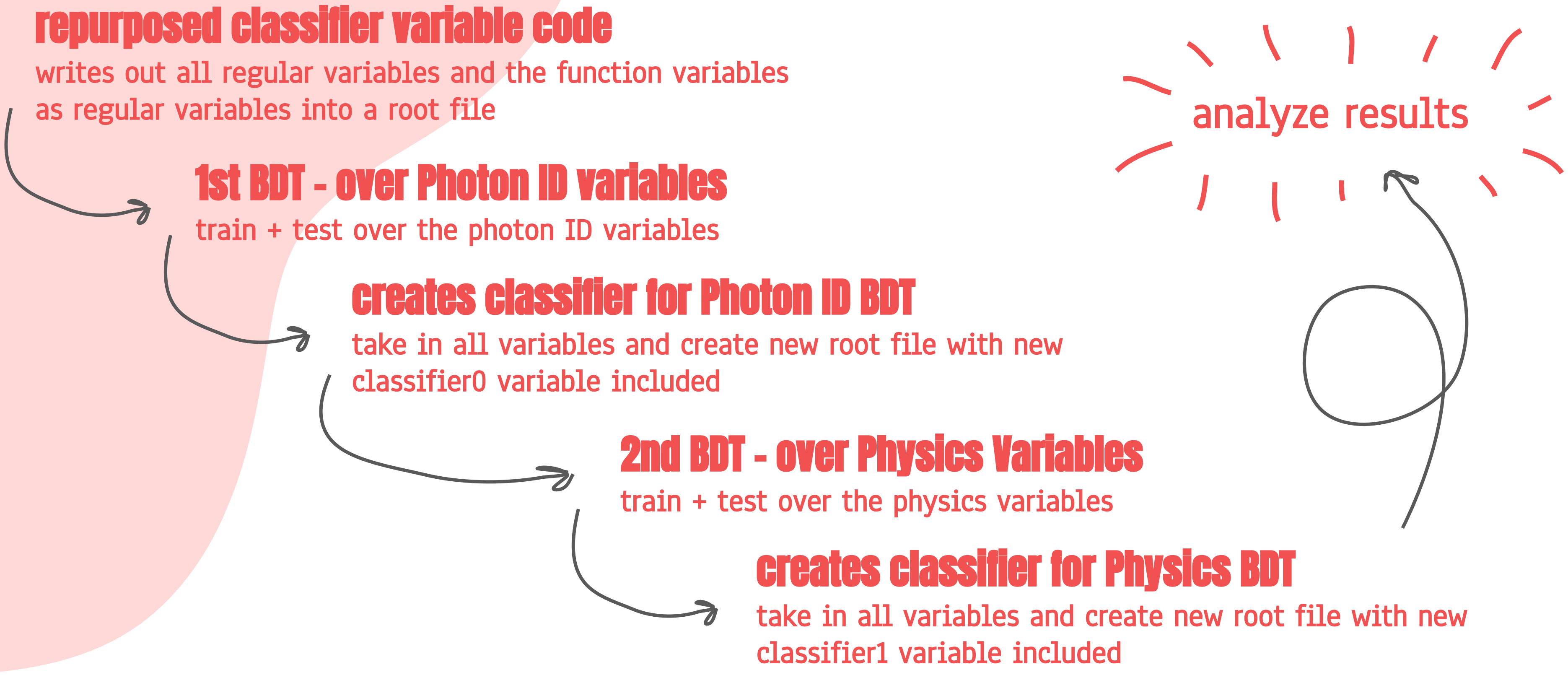
$$\Delta R = \sqrt{(\eta_1 - \eta_2)^2 + (\varphi_1 - \varphi_2)^2}$$

HICCUPS ALONG THE WAY

1) Typos/errors in one code affecting the next code.
2) BDT code allowed “function” variables but the code creating our classifier variable did not.
3) Simple variables weren’t enough to differentiate between signal and background so the function variables had to be included somehow if we wanted results.

Solution: repurpose the classifier variable code

RUNNING PROCESS



repurposed classifier variable code

writes out all regular variables and the function variables as regular variables into a root file

1st BDT - over Photon ID variables

train + test over the photon ID variables

creates classifier for Photon ID BDT

take in all variables and create new root file with new classifier0 variable included

2nd BDT - over Physics Variables

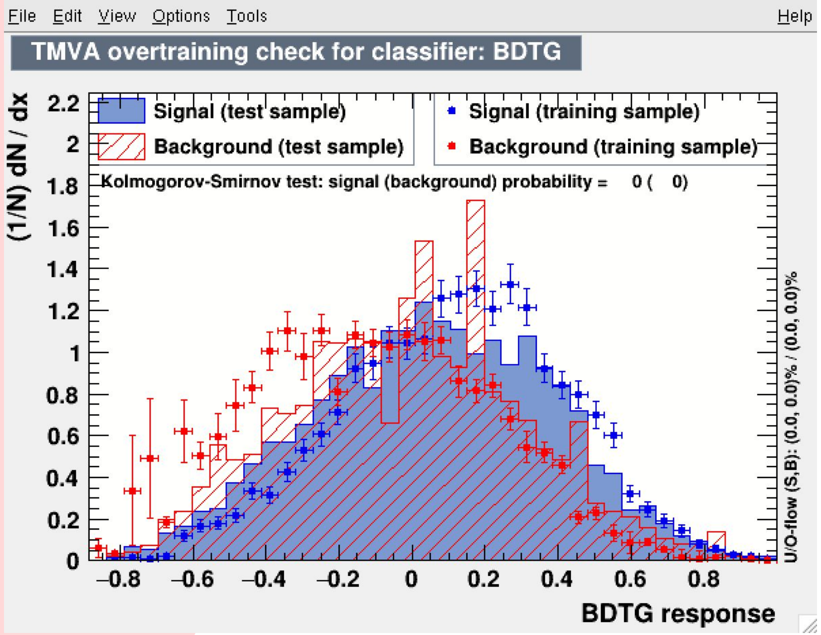
train + test over the physics variables

creates classifier for Physics BDT

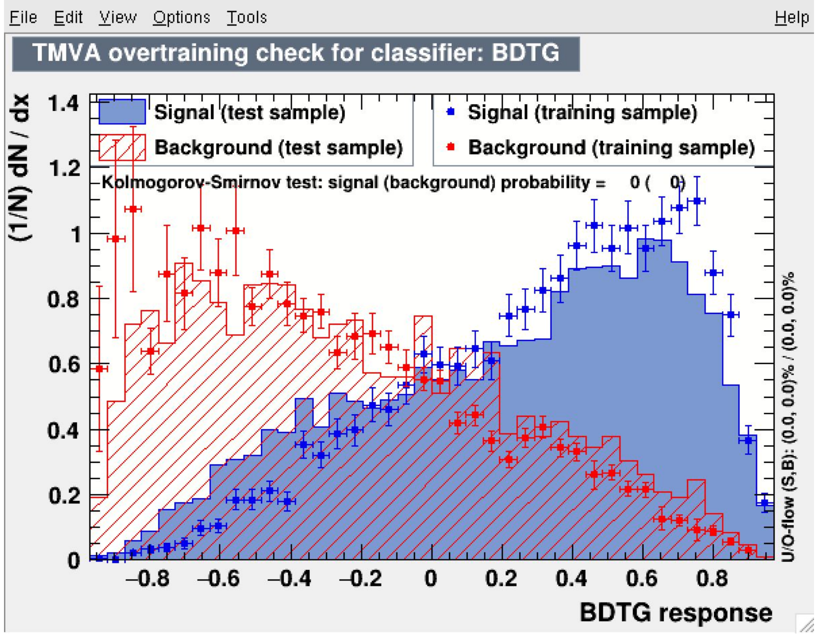
take in all variables and create new root file with new classifier1 variable included

analyze results

BDT RESULTS

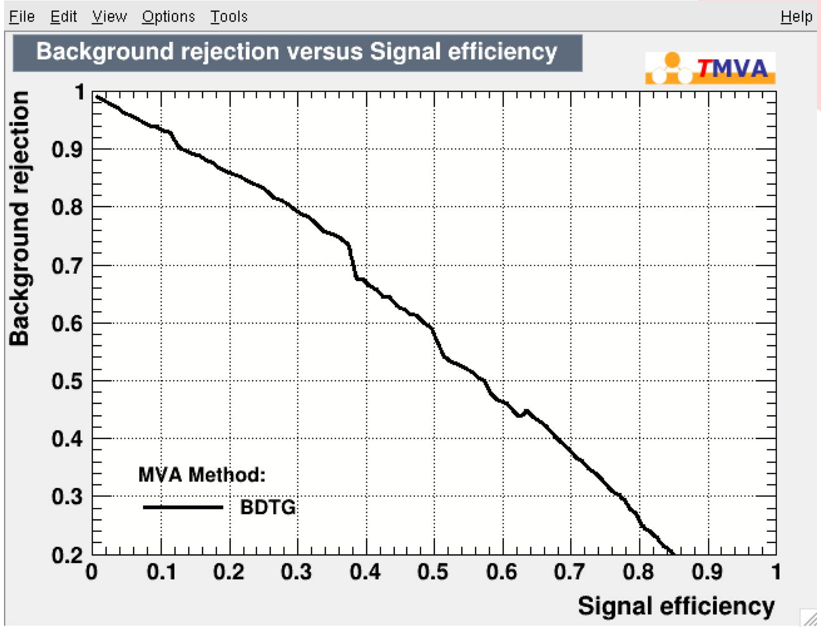


Photon ID

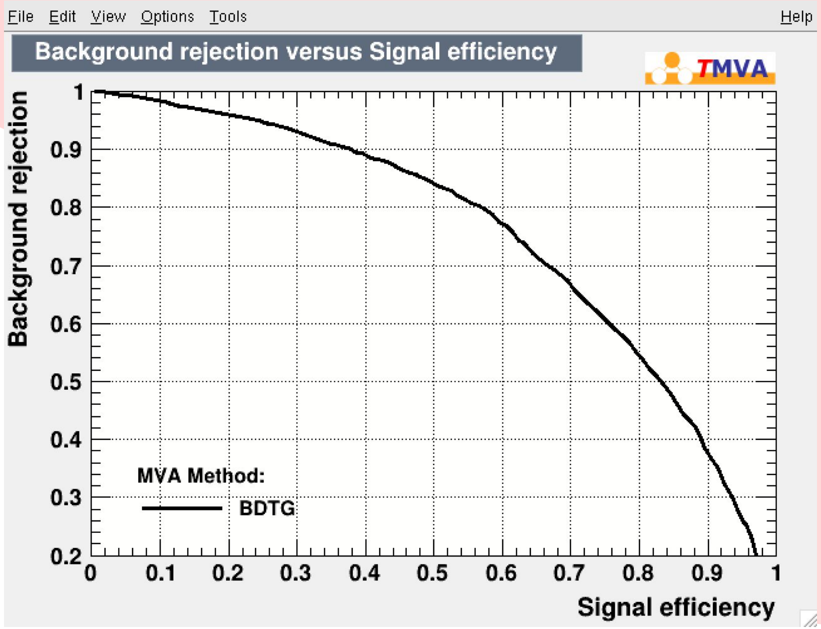


Physics

ROC CURVES

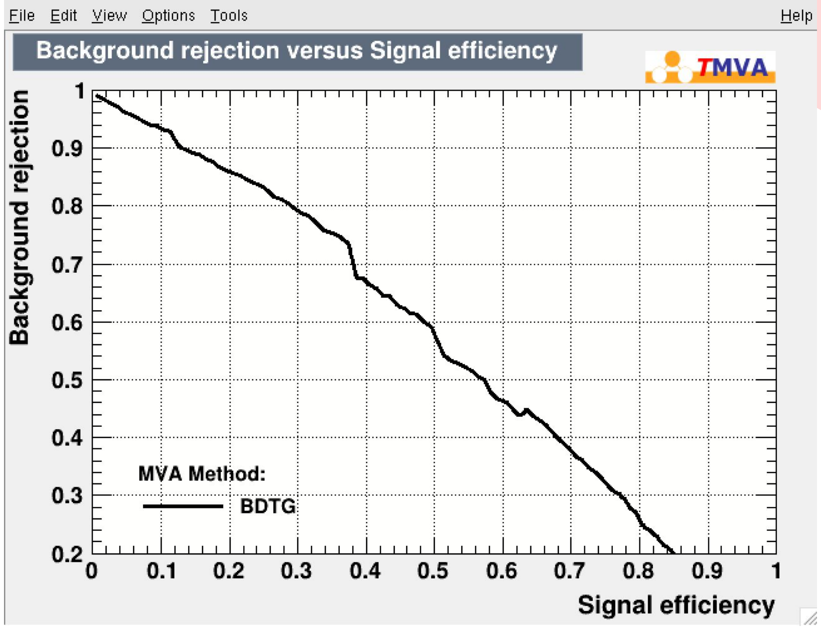


Photon ID

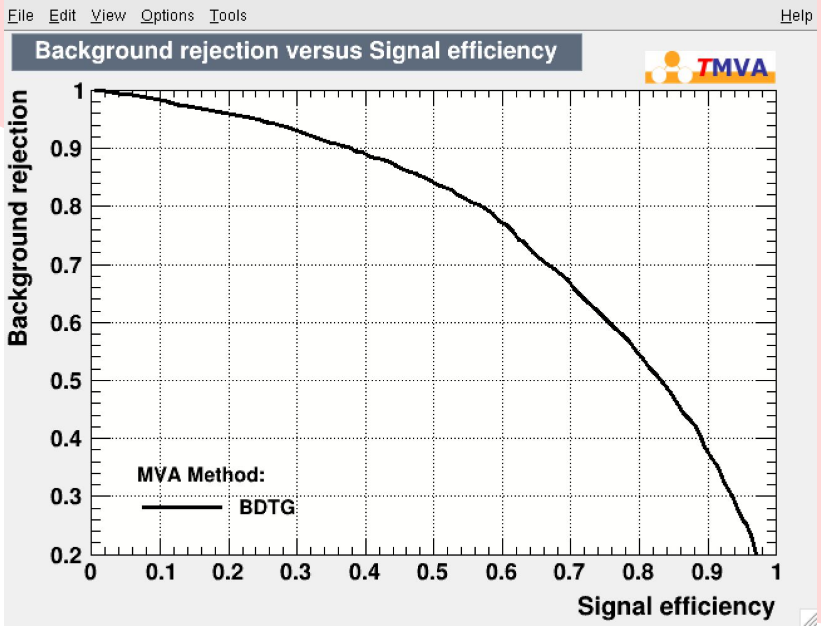


Physics

ROC CURVES



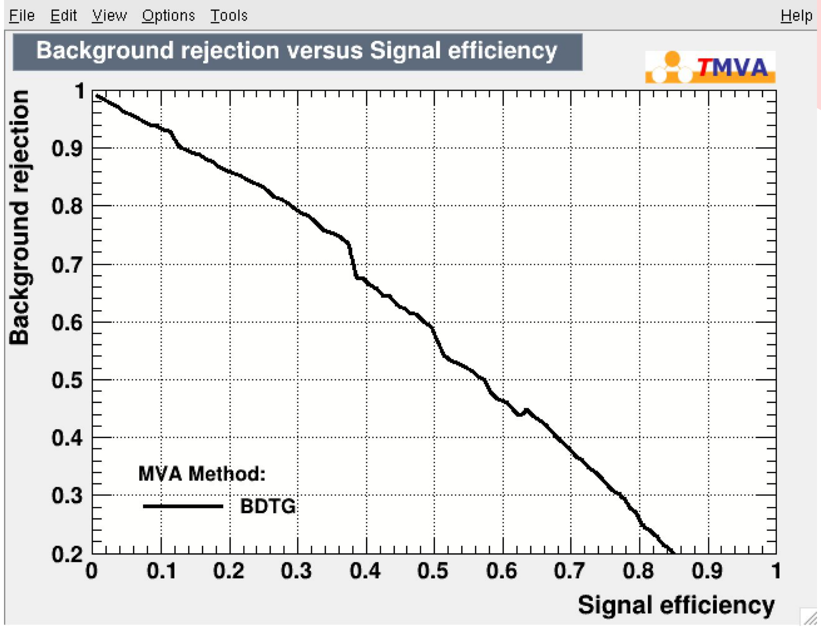
Photon ID

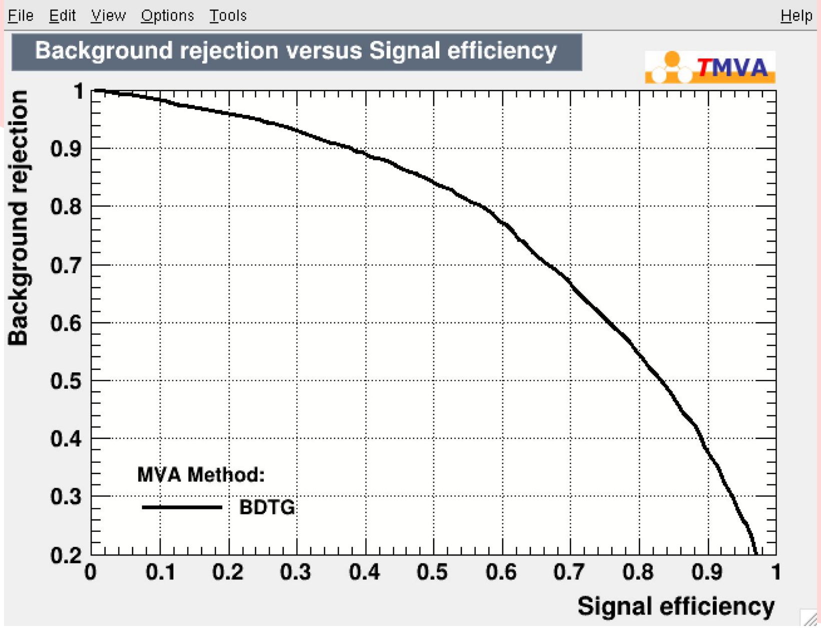


Physics

$$\sigma = \frac{s}{\sqrt{b}} = \frac{0.5}{\sqrt{0.4}} \approx 0.79 \approx -\ 20\%$$

ROC CURVES





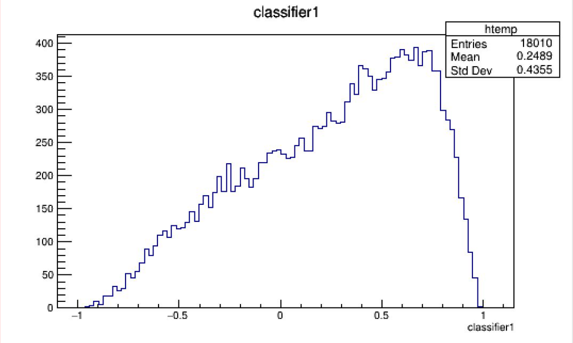
Photon ID

$$\sigma = \frac{s}{\sqrt{b}} = \frac{0.5}{\sqrt{0.4}} \approx 0.79 \approx -20\%$$

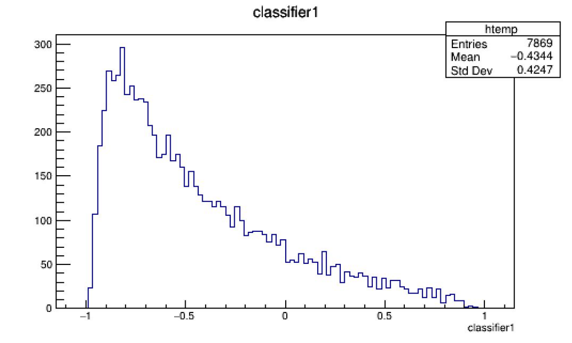
Physics

$$\sigma = \frac{s}{\sqrt{b}} = \frac{0.5}{\sqrt{0.15}} \approx 1.29 \approx 30\%$$

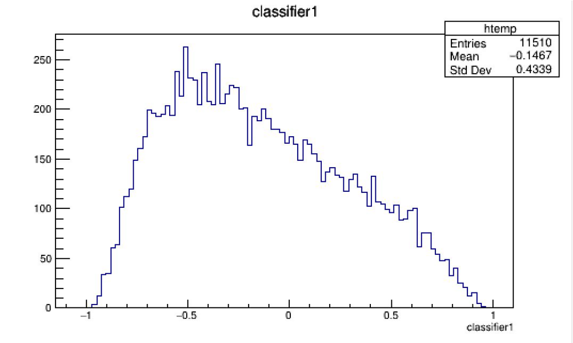
RESULTS



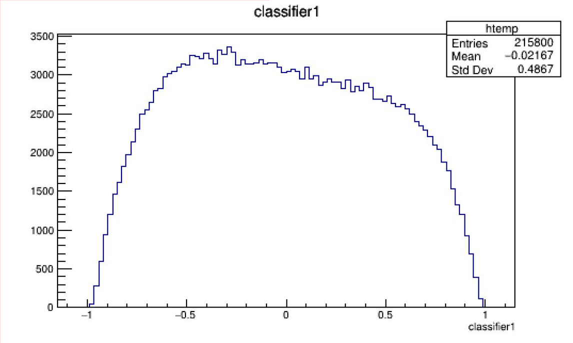
WWy



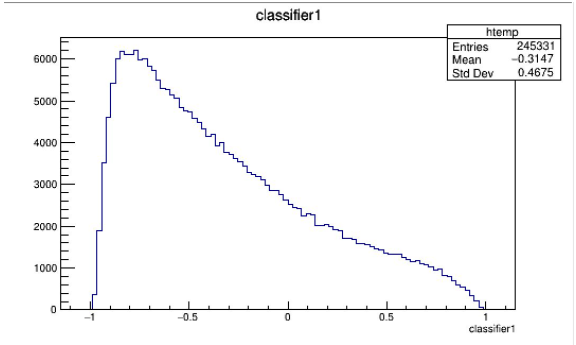
ZZ



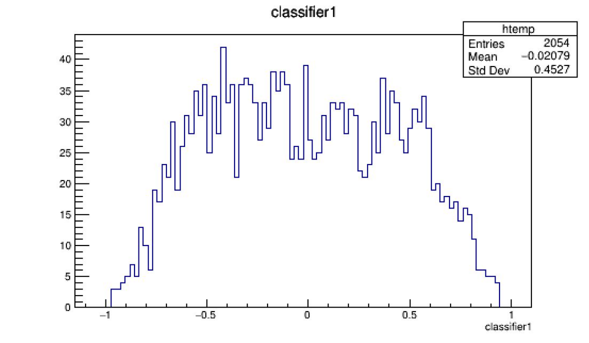
ttbar



Diboson

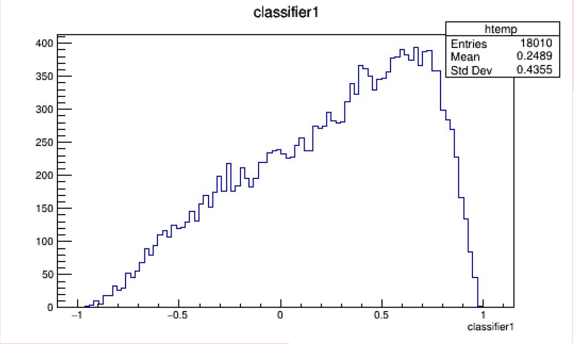


Zjets

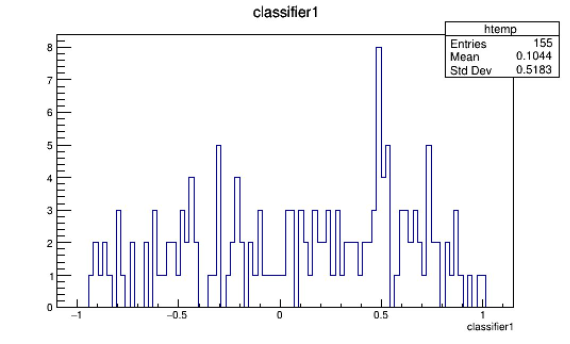


tW

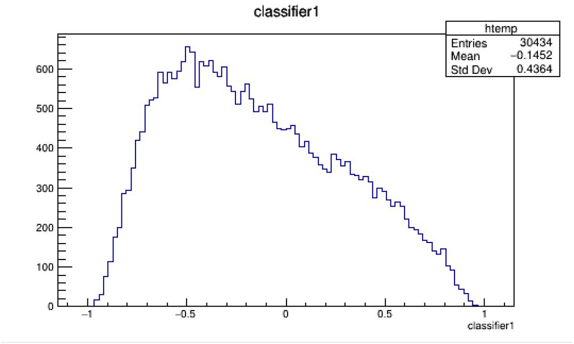
RESULTS



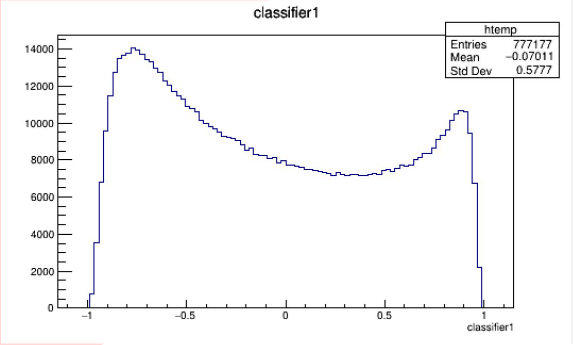
WWy



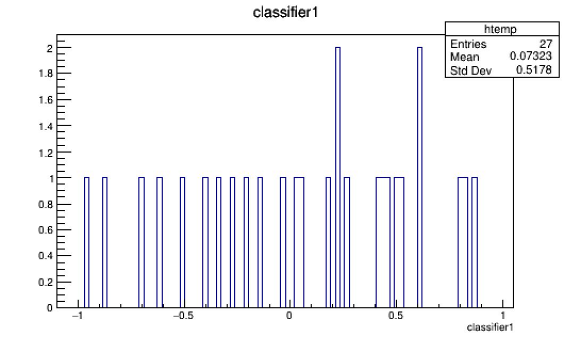
Wy



tty



Zy



Wjets

SO WHAT DID WE GAIN?

1) We know which variables when run through a BDT will enable us to increase the significance by 30%
2) Know the parameters required for this result
3) Know which backgrounds the BDT will most likely mistake as WWy
4) Have codes ready to go to run the real data through instead of the Monte Carlo simulated data

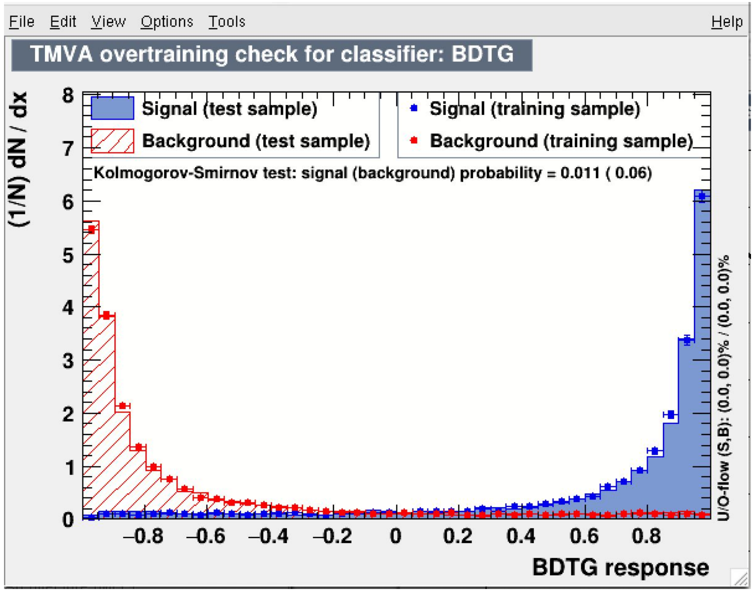
QUESTIONS?

thanks for listening!

credit to Slidesgo again for the slides template and a big thank you to OU for having me (virtually) for the summer!

1st CODE : TMVAClassification.C

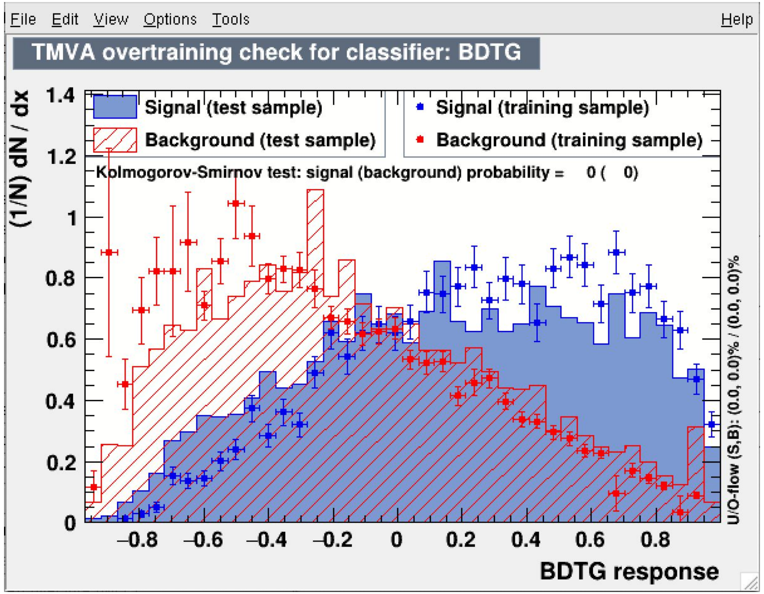
- 1st Code I worked with and was able to begin adjusting.
- It hasn't been used in 2 years so had to swap out the variables for updated versions and make sure it ran.
- Eventually got it to run and had a great plot:



Blue is signal, Red is background.
The dots are from the training and the bars are from the test.

1st CODE : TMVAClassification.C

- 1st Code I worked with and was able to begin adjusting.
- It hasn’t been used in 2 years so had to swap out the variables for updated versions and make sure it ran.
- Eventually got it to run and had a great plot:

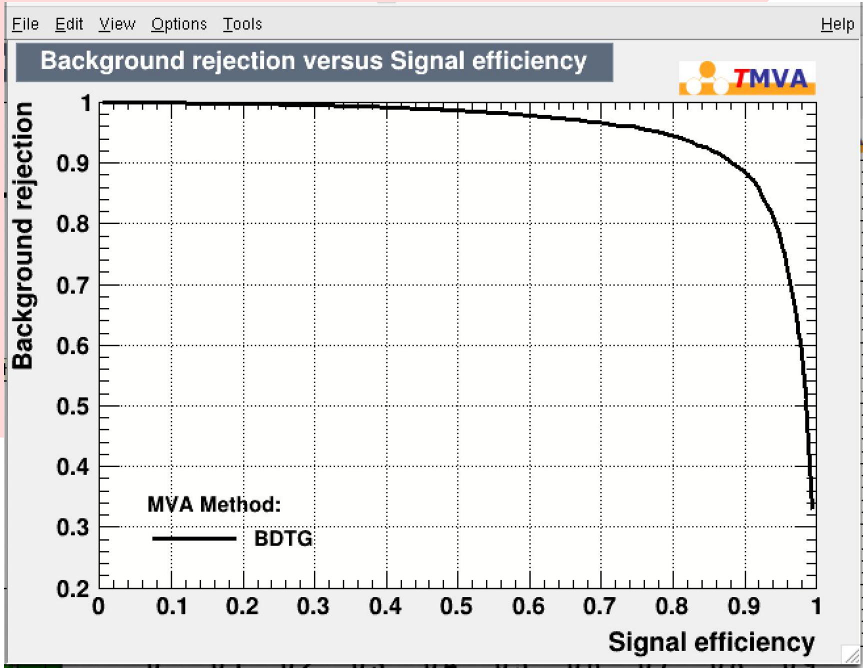


File Edit View Options Tools
Help
TMVA overtraining check for classifier: BDTG
Signal (test sample)
Background (test sample)
Signal (training sample)
Background (training sample)
Kolmogorov-Smirnov test: signal (background) probability = 0 (0)
(1/N) dN / dx
0
0.2
0.4
0.6
0.8
1
1.2
1.4
−0.8
−0.6
−0.4
−0.2
0
0.2
0.4
0.6
0.8
BDTG response
U/O-flow (S,B): (0.0, 0.0)% / (0.0, 0.0)%

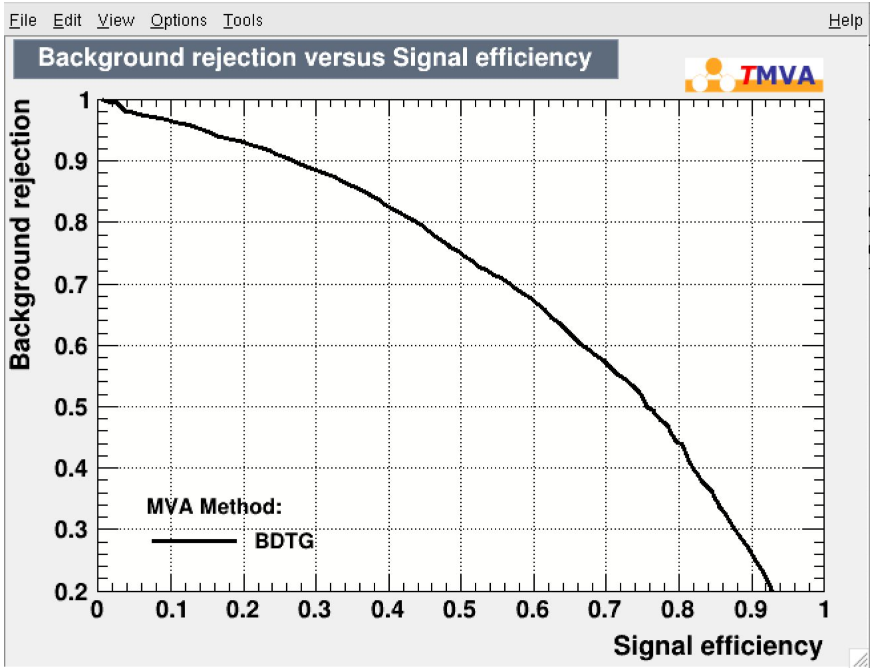
Add in cuts that there must be one electron, one muon and at least one photon:

ROC CURVES

without cuts



with cuts



1st CODE : TMVAClassification.C

- What Changed?
- The size of sample was now much smaller.
- less events for the BDTG to train on therefore worse results.

SOLUTION?

- try splitting up the BDTG into training on two sets of variables: the ones relevant to the physics and photon shower shapes; allowing it to focus on one set at a time instead of dividing its attention and doing a poor job.

2nd CODE : ApplicationCreateCombinedTree.C

- This code basically would just rewrite all the variables into a new .root file while also creating a new variable “classifier” which was the likelihood of an event being signal or background on a scale of -1 to 1.

- -0.9 being a high likelihood of being a background event,
  0.9 being a high likelihood of being a signal event,
  0 being it’s equally likely to be background or signal based off the BDTG’s assessment.

2nd CODE : ApplicationCreateCombinedTree.C

Since we split into photon ID specific variables and physics related variables and are running two BDTGs, we are actually getting two sets of classifier variables:

- classifier0 is an event's likelihood of being signal based off the photon ID variables.
- classifier1 is an event's likelihood of being signal based off the physics variables.

MY PROJECT

- Using Boosted Decision Trees (BDTs) to increase σ
- TMVA: Toolkit for Multivariate Data Analysis
- Multivariate Analysis = analysis that takes into account multiple measurements made on each experimental unit and the relations among those measurements.