Incorporation of Systematic Uncertainties in the Training of Multivariate Methods

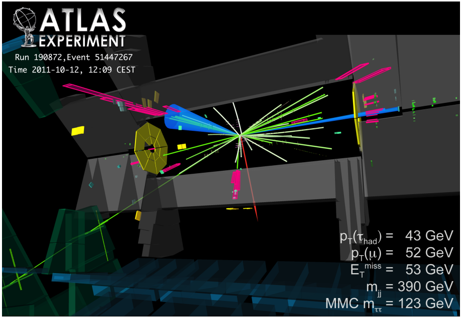
Thomas Alef & Eckhard von Törne, University of Bonn

ACAT 2019
March 13^{th}, 2019

Motivation and Goal

- Modern analysis often limited by systematic uncertainties
  $\Rightarrow$ make multivariate methods robust against systematics
- **Systematic aware Boosted Decision Trees** (**saBDT**) developed during Masterthesis
  - Based on AdaBoost/Gini Index BDTs from TMVA
  - Tested on modified public data from Kaggle Higgs Challenge
- Compared with **Adversarial Neural Networks** (**AdvNN**)
- **AdvNN** based on KERAS

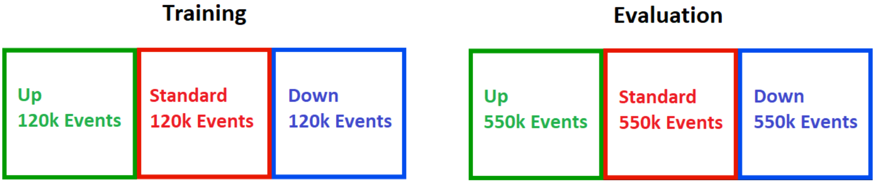
Public Data from Kaggle Higgs Challenge



- Data from Kaggle Higgs Challenge $H \to \tau\tau$
- 30 variables
- Training: 120,000 events (Kaggle challenge public data)
- Evaluation: 550,000 events (Kaggle challenge private data)
- For testing influence of systematics a systematic variation was added

Implementation of Systematic Variation

- Jet Energy Scale chosen as example systematic - standard ATLAS systematic
- Strength of systematic variation: 1% (ATLAS standard value 1-4%)
- Scale jet energies up by 1%
  $\rightarrow$ recalculate all variables based on jet energies with new values
  $\rightarrow$ new systematic varied *Up* dataset
- Repeat with scaling down by 1%
- $\Rightarrow$ 3 Datasets: *Nominal*, *Up*, *Down*



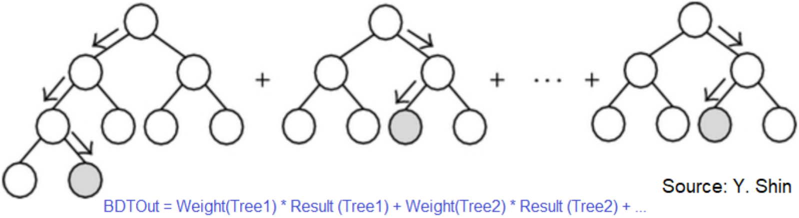
Evaluation Metric: AAMS

- Kaggle Challenge used Approximate Median Significance (AMS)
- Adding systematic uncertainty: Advanced Approximate Median Significance ($AAMS$) (see hal-01208587)
- *Cut and Count* approach: events with higher score than x are classified signal

$$
\begin{aligned}
AAMS &= \sqrt{2\left((s+b)\,ln\frac{s+b}{b_0} - s - b + b_0\right) + \frac{(b-b_0)^2}{\sigma_b^2}} \\
b_0 &= \frac{1}{2}\left(b - \sigma_b^2 + \sqrt{\left(b-\sigma_b^2\right)^2 + 4\,(s+b)\,\sigma_b^2}\right)
\end{aligned}
$$

- s signal events, b background events, σ_b background difference on the different data sets
- Unstable for small $b \rightarrow$ add a regularization term of 10 to b
- Maximum of σ_b for all possible cut values: $\sigma_b^{\max}$
  $\Rightarrow$ if small, method behaves similar on varied datasets

How to make BDTs aware of Systematics



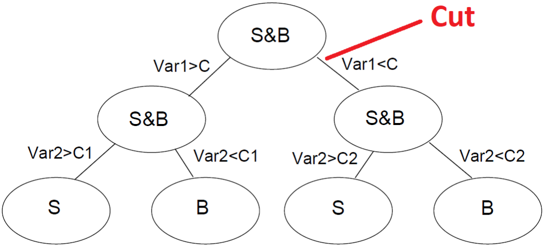
- BDT uses all three datasets during training
- If performance similar on all three datasets - invariant under systematic variations
- Similar behavior checked for:
  - Every single node split
  - Whole tree (Boostweight)
- AdaBoost BDT with Gini Index on ROOT 6.10/06
- NTrees=1000, MinNodeSize=1%, AdaBoost=0.2

BDT: Standard Node Split

- So far: scan through all variables and possible cuts, maximize:

$$Gain = G_{\text{Parent}} - G_{\text{Left}} - G_{\text{Right}}$$

  with Gini Index $G = p \cdot (1 - p)$ (maximal for p=0.5) and $p = \frac{N_{\text{Signal}}}{N_{\text{All}}}$
- Basically: find the cut which improves the purity of the nodes the most



saBDT: Systematic Aware Node Split

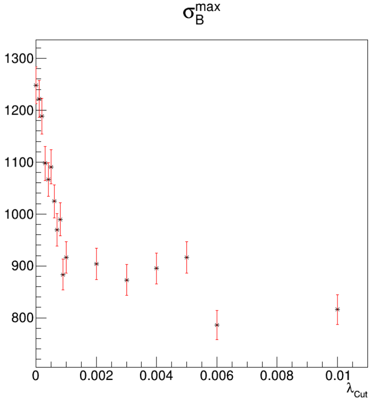
- Modify *Gain* to penalize differing behavior on different data sets
- Modification based on purity to stay consistent
- Subtract a term accounting for purity differences on different data sets:

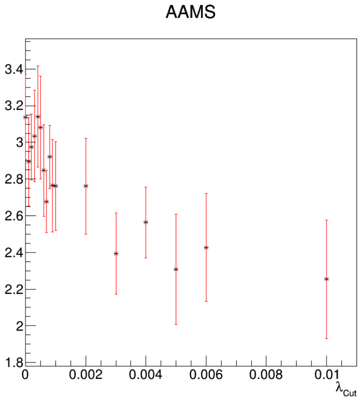
$$NewGain = Gain - \lambda_{Cut} \cdot \frac{1}{8} \cdot \sqrt{\sum_{Left,Right} \left(p_{Reg} - p_{Up,Down}\right)^2}$$

- λ_{Cut} as hyperparameter to control strength of invariance
- Penaltyterm can be between 0 and 0.25

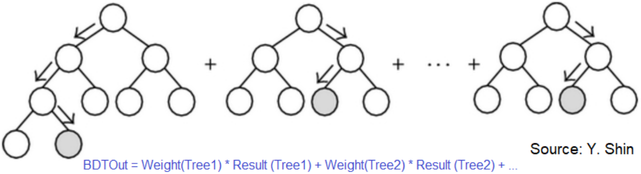
saBDT: λ_{Cut} Hyperparameter Scan Results

- Stable AAMS with possible increase for low λ_{Cut}
  $\Rightarrow$ Algorithm works!
- σ_b^{max} decreases - performance similar on different datasets





saBDT: Systematic Aware Treeweight



- Every decisiontree is weighted according to its error rate:

$$err = \frac{N_{\text{misidentified}}}{N_{\text{All}}} \Rightarrow TW = \log \frac{1 + err}{1 - err}$$

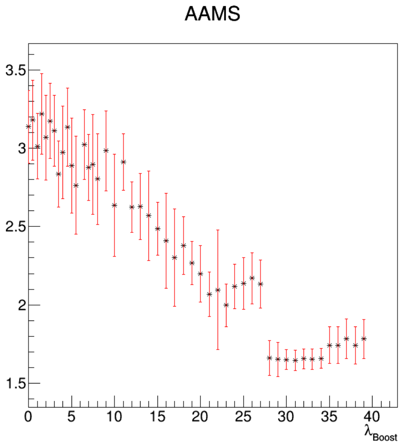
- TW is the boostweight, high when tree performing well
- Multiply factor accounting for differences on systematic varied samples:

$$NewTW = TW \cdot \exp\left(-\lambda_{Boost} \cdot \frac{\sum_{\text{Up,Down}} \left(err_{\text{Reg}} - err_{\text{Up,Down}}\right)^2}{2} \right)$$

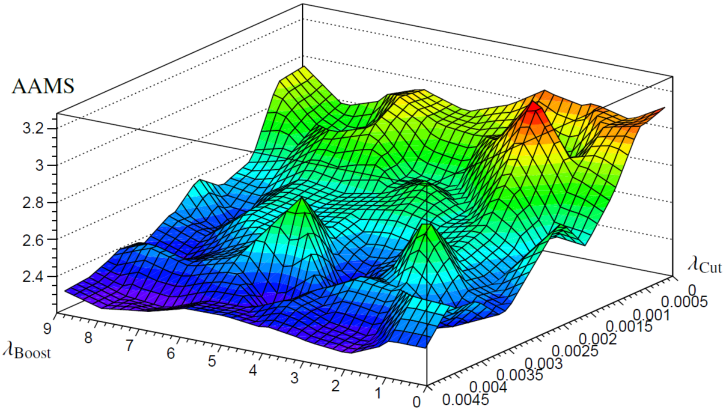
- λ_{Boost} as hyperparameter
- New factor pulls down weight of trees affected by systematic variation

saBDT: λ_{Boost} Hyperparameter Scan Results

- *AAMS* (performance) drops for high λ_{Boost}
- Influence of systematics decreases as well!
- Stable region with possible increase for low values



saBDTs: 2-Dim Hyperparameter Scan



- Scanning through λ_{Cut} and λ_{Boost} reveals increasing *AAMS*
- Confirmed by Bootsstrap: 82.1% chance it is not a statistical fluctuation

saBDT: Different Strength of Systematic Variation

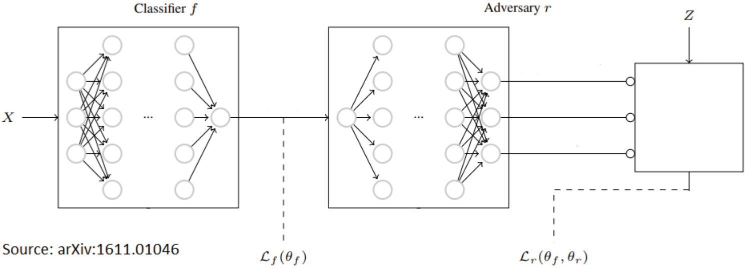
- Different strength in systematic variation of data is applied

Systematic Variation	BDT (*AAMS*)	saBDT (*AAMS*)	% no stat. Fluc.
20%	1.07±0.05	1.52±0.06	98.4%
10%	1.38±0.06	1.94±0.07	99.6%
3%	2.40±0.09	2.64±0.09	92.3%
1%	3.13±0.11	3.22±0.10	82.1%

- 3% and 1% ATLAS standard values
- saBDTs improves result especially well with strong systematic variation
- Result dominated by systematic uncertainty in this region → decreasing systematic uncertainty more valuable

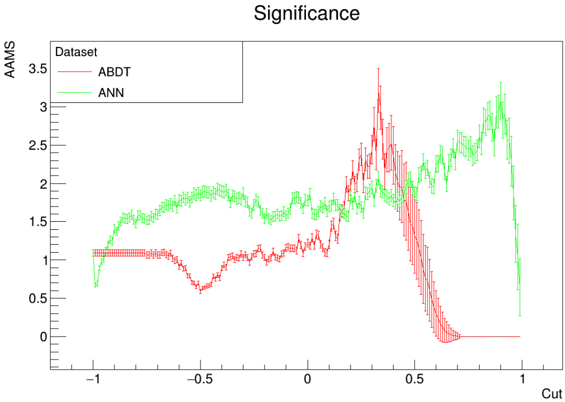
Adversarial Neural Networks

- As comparison AdvNN (see Louppe, Kagan, Cranmer: arXiv:1611.01046)
- Multiple talks during the next days
- *Classifier* able to distinguish signal and background
- *Adversary* penalizing Classifier if it is sensitive to systematic variations
- γ as strength parameter for penalty



Source: arXiv:1611.01046

saBDTs vs AdvNNs



- Comparison for 1% systematic variation
- saBDT performs slightly better!
- Maximal $AAMS(\text{saBDT}) = 3.23 \pm 0.10$, $AAMS(\text{AdvNN}) = 3.08 \pm 0.11$
- AdvNN not fully optimized

Conclusion and Outlook

Conclusion

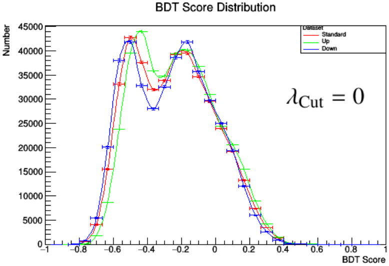
- saBDTs proved capable of reducing systematic uncertainty
- Gain in AAMS was achieved
- AdvNNs were outperformed
- AdvNNs less optimized than saBDTs - difference originating from this?
- Invariance proved to be most valuable for high systematic effects

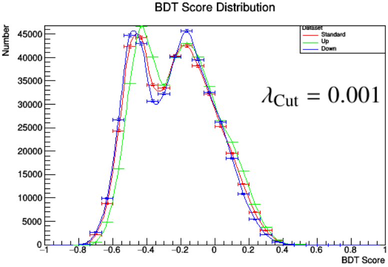
Outlook

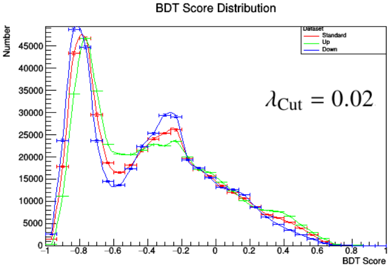
- saBDTs tested with different systematics
- New metrics to test the performance
- Multiple systematics at once?

Backup

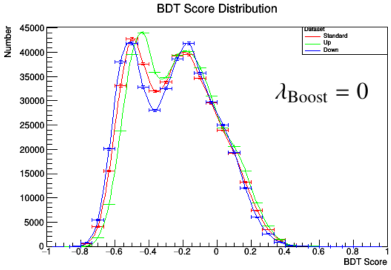
saBDT: Node Split BDT Distribution

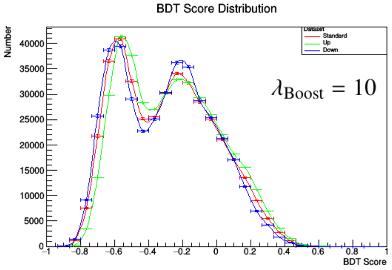






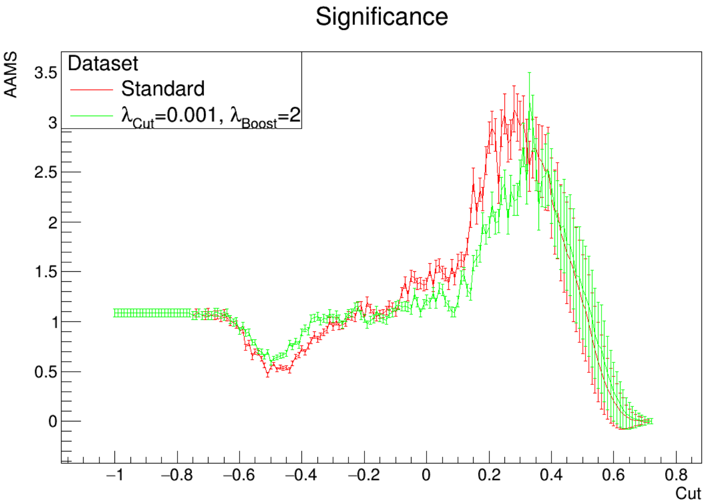
saBDT: Boost BDT Distribution





- Distributions behave similar to λ_{Cut}
- Getting shifted to the left

saBDT: AAMS



Significance
AAMS
Dataset
Standard
λ_{Cut}=0.001, λ_{Boost}=2
3.5
3
2.5
2
1.5
1
0.5
0
−1
−0.8
−0.6
−0.4
−0.2
0
0.2
0.4
0.6
0.8
Cut

Adversarial Neural Networks

Used AvdNN

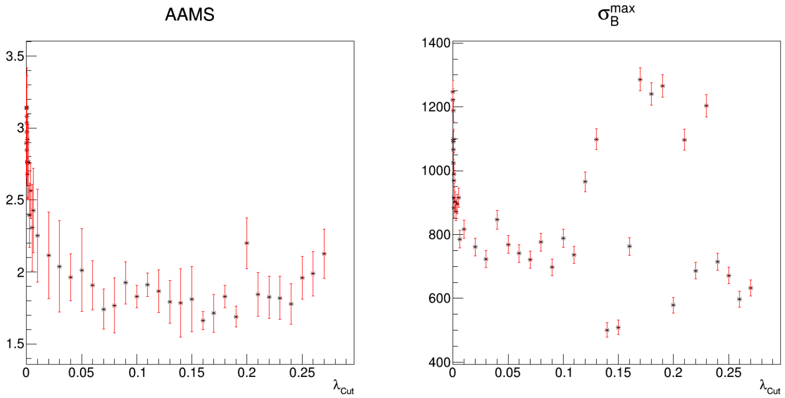
- *Classifier*:
  - 30 input nodes, one for every variable
  - 3 dense hidden layers, regularized by $l1 = 0.0001$ and $l2 = 0.001$
  - 120 nodes each
  - Activation function is *relu* for the hidden layers
  - 1 output note, with *sigmoid* as activation function
  - batch size is 64
- *Adversary*:
  - 1 input node
  - 3 dense hidden layers
  - The first two hidden layers have 30 nodes each and the last with 12
  - Activation function is *relu* for the hidden layers
  - 3 output nodes, with *softmax* as activation

Variables

Variable	Comment
DER_mass_MMC	effect but hard to calculate – neglected
DER_mass_transverse_met_lep	If mEt is affected here as well
DER_mass_vis	not affected
DER_pt_h	If mEt is affected here as well
DER_deltaeta_jet_jet	not affected
DER_mass_jet_jet	directly affected
DER_prodeta_jet_jet	not affected
DER_deltar_tau_lep	not affected
DER_pt_tot	directly affected
DER_sum_pt	directly affected
DER_pt_ratio_lep_tau	not affected
DER_met_phi_centrality	If mEt is affected here as well
DER_lep_eta_centrality	not affected
PRI_tau_pt	not affected
PRI_tau_eta	not affected
PRI_tau_phi	not affected
PRI_lep_pt	not affected
PRI_lep_eta	not affected
PRI_lep_phi	not affected
PRI_met	affected similar to jet energy
PRI_met_phi	not affected
PRI_met_sumet	directly affected
PRI_jet_num	not affected
PRI_jet_leading_pt	directly affected
PRI_jet_leading_eta	not affected
PRI_jet_leading_phi	not affected
PRI_jet_subleading_pt	directly affected
PRI_jet_subleading_eta	not affected
PRI_jet_subleading_phi	not affected
PRI_jet_all_pt	directly affected

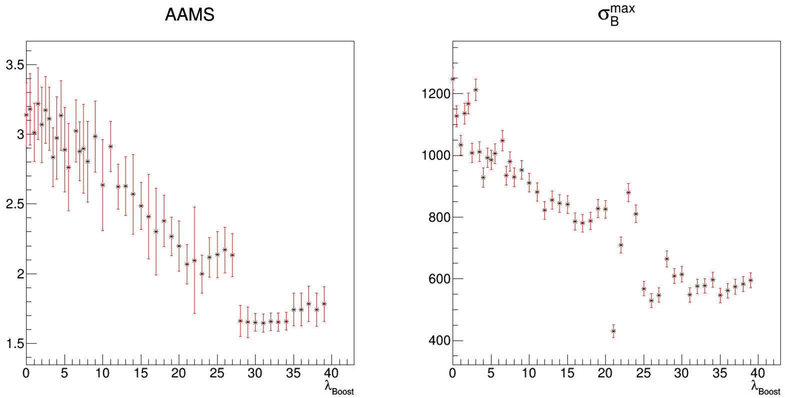
saBDT: systematic aware node split results

- *AAMS* (performance) drops initially with λ_{Cut}
- Influence of systematics decreases as well!
- Breakdown around $\lambda_{\text{Cut}} = 0.01$



saBDT: λ_{Boost} Hyperparameter Scan Results

- *AAMS* (performance) drops for high λ_{Boost}
- Influence of systematics decreases as well!
- Stable region with possible increase for low values



Overview of AMS/AAMS Results

Method	*AMS*	*AAMS*
Kaggle Winner	3.81	NA
Kaggle TMVA	3.50	NA
BDT	3.44	3.13
saBDT	3.35	3.22
NN	3.27	2.88
AdvNN	3.20	3.08

- Including systematic aware training leads to loss in *AMS* and gain in *AAMS*
- Tested methods not as fully optimized as during challenge

Aware Boosted Decision Trees: Bootstrap

- Difference in performance of standard BDT and tuned saBDT tested on bootstrapped samples
- Bootstrap creates new samples with different statistics out of the original sample
- saBDT performs indeed better, but not significant
- $\Delta AAMS = 0.138 \pm 0.150$