Multivariate Techniques for $t\bar{t}H \rightarrow b\bar{b}$ Analysis in ATLAS

Ziyu GUO
Supervisors: Yann COADOU, Thierry ARTIERES

Top LHC France @ LPSC Grenoble

April 25, 2019

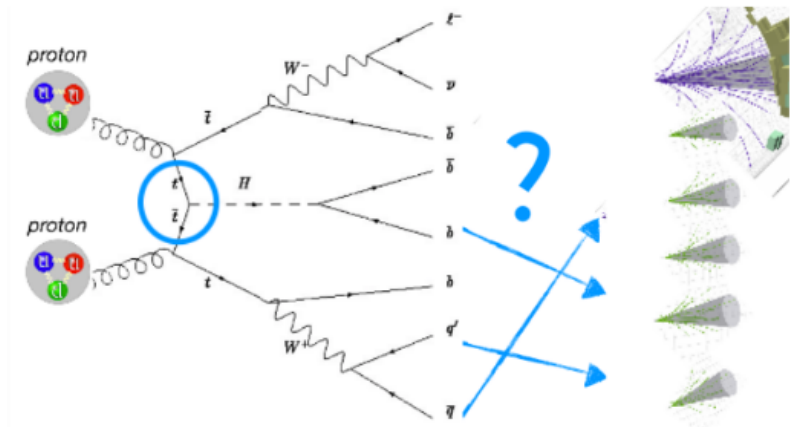
1. $t\bar{t}H \rightarrow b\bar{b}$ analysis

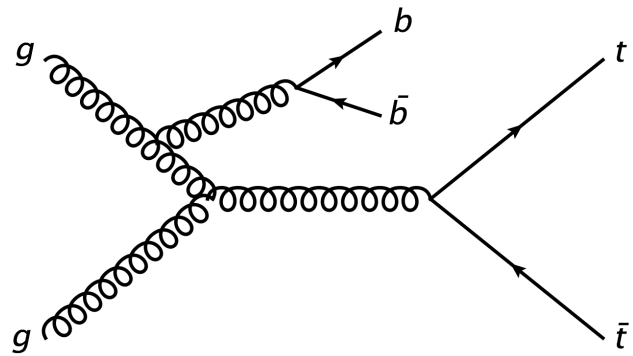
2. MVAs in the analysis with 36.1 fb^{-1} in Run 2

3. Exploring deep learning in $t\bar{t}H \rightarrow b\bar{b}$ analysis
   - Recurrent neural networks
   - Parse tree
   - Adversarial training
   - Multiclassification

$t\bar{t}H \to b\bar{b}$ analysis with 36.1 fb^{-1} in Run 2 PhysRevD.97.072016

- Analysis challenges and strategy:
  - The systematic uncertainty is dominated by the uncertainty in the $t\bar{t}$ + b-jets modeling
    - region categorization including $t\bar{t}$ + b-jets enriched CR, simultaneous fit
  - Low $t\bar{t}H$ signal x-section, and large irreducible background of $t\bar{t}$+jets
    - a BDT to separate signal from bkg, used as the discriminant for the profile likelihood fit in signal-enriched regions
  - Multiple jets and b-jets in the final state, difficult to be matched to partons correctly
    - using MVAs to reconstruct $t\bar{t}H$



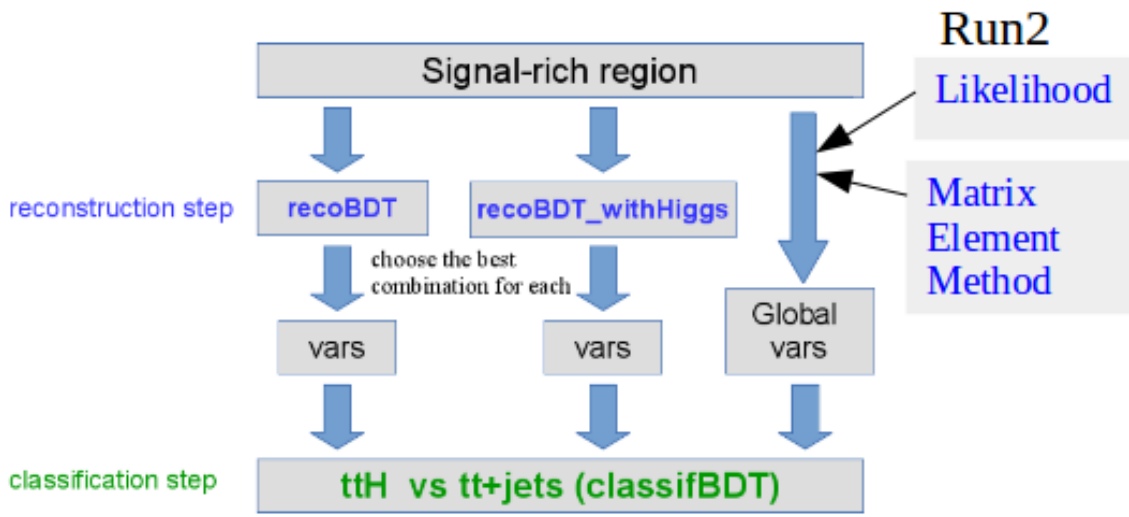


- This study: 1l channel, resolved events
- Pre-selection for training: (5jets, $\geq 4b$-jets), ($\geq$ 6jets, $\geq 4b$-jets)@85%

MVAs in the analysis with 36.1 fb^{-1} in Run 2

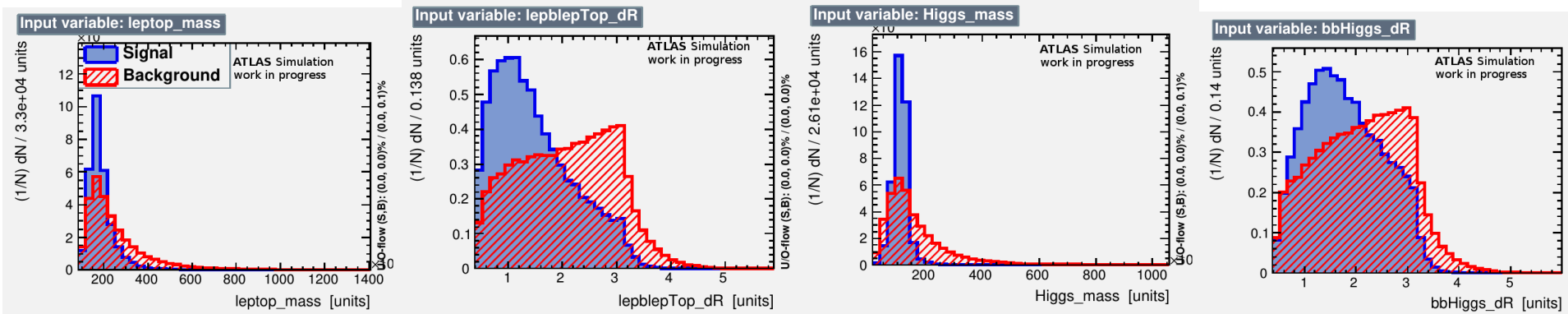
- Reconstruction step: find the correct association between jets and partons which originate from H/top decays
  - Reco. BDT: pick the combination with the highest BDT score as the correct matching among all possible combinations on $t\bar{t}H$ (see next slide)
  - Likelihood discriminant (LHD): build probability distribution function under the signal/background hypotheses using 1D variable distributions from all possible combinations
  - MEM: exploit the full matrix element calculation to separate the signal from the background.

- Classification step
- Such method applied in regions with 5 and ≥ 6 jets separately



Reconstruction BDT

- **Goal**: find the correct association between jets and partons which originate from H/top decays
  - Try all possible jet-parton matches, leading to multiple possible combinations
  - Train on $t\bar{t}H$ sample only
    - Signal: correct matches up to 1 jet from W mis-matched. Bkg: all other improper combinations
- Train two different BDTs:
  - BDT: using vars uncorrelated to Higgs
  - BDT withH: adding Higgs information, e.g. $\Delta R(b, b)$ from Higgs candidate
- Some input variables in $\geq 6j$ region:



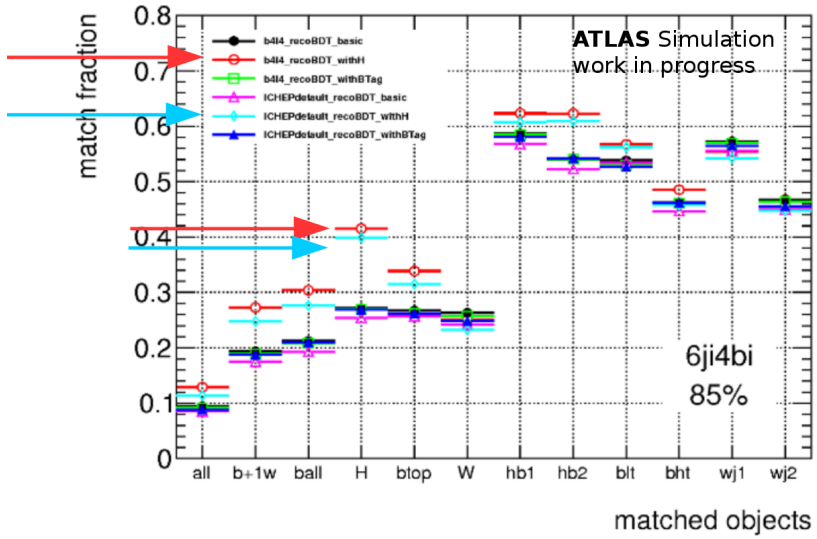
Reconstruction BDT

- To reduce number of combinations, b quarks are only associated to b-jets.
- Previously, b-jets are tagged if passing a tighter fixed b-tagging efficiency threshold (WP: Working Point) $\rightarrow$ low stats, light quark often mis-tagged
- Using the pseudo-continuous (PC) b-tagging, and starting from loose WP, give more correct matchings
  - Sorting jets w.r.t. b-tagging weight, the leading 4 jets are considered as b-jets, and the leading 4 jets cannot be used as light jets.

Reco BDT with Higgs, PC vs. the fixed b-tagging:

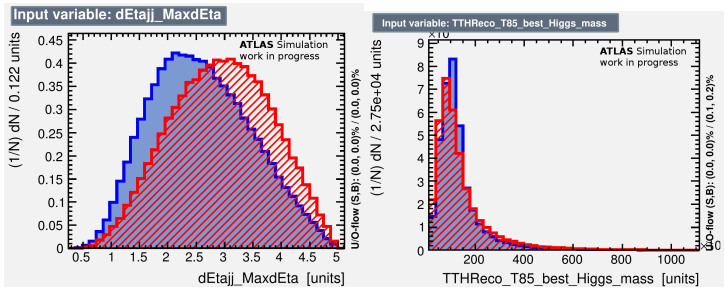
- 42%, two b-jets from Higgs were truth matched
- 5% improvement with PC b-tagging
- Almost 10% improvement for all b-jets truth matching.

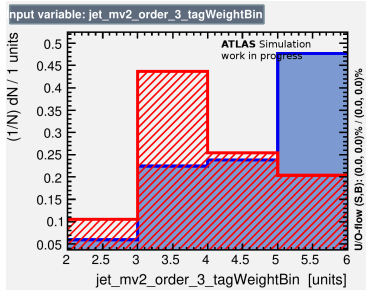
Figure: Reco BDT performance



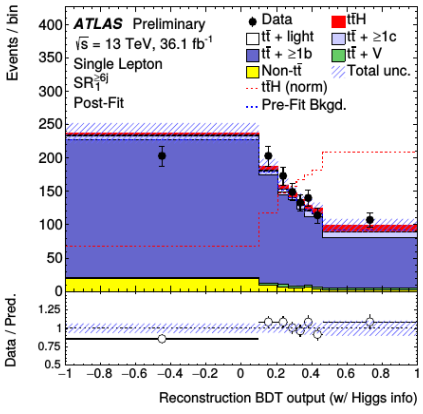
Classification BDT

- 2 BDTs are trained for events having 5 and ≥ 6 jets, both with ≥ 4 b-jets @85% WP, across CRs and SRs
- Signal: $t\bar{t}H$, Bkg: $t\bar{t}$
- Input variables:
  - Global event kinematics
  - PC b-tagging of jets
  - Variables from reconstruction BDT: the vars built based on the combination with the highest Reco BDT score.
  - Outputs of reco BDT, LHD, MEM (where available)

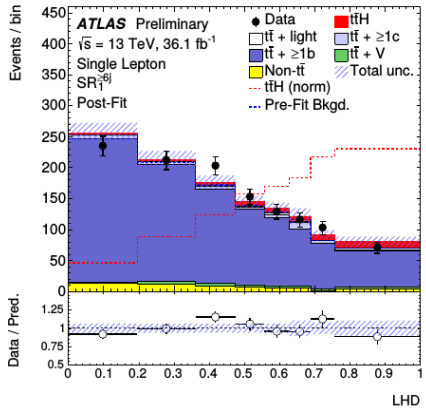




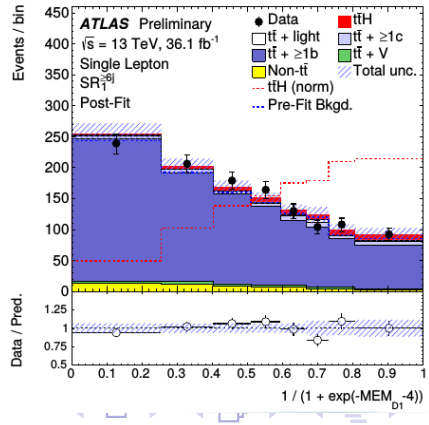
Reconstuction BDT



Likelihood discriminant



MEM discriminant

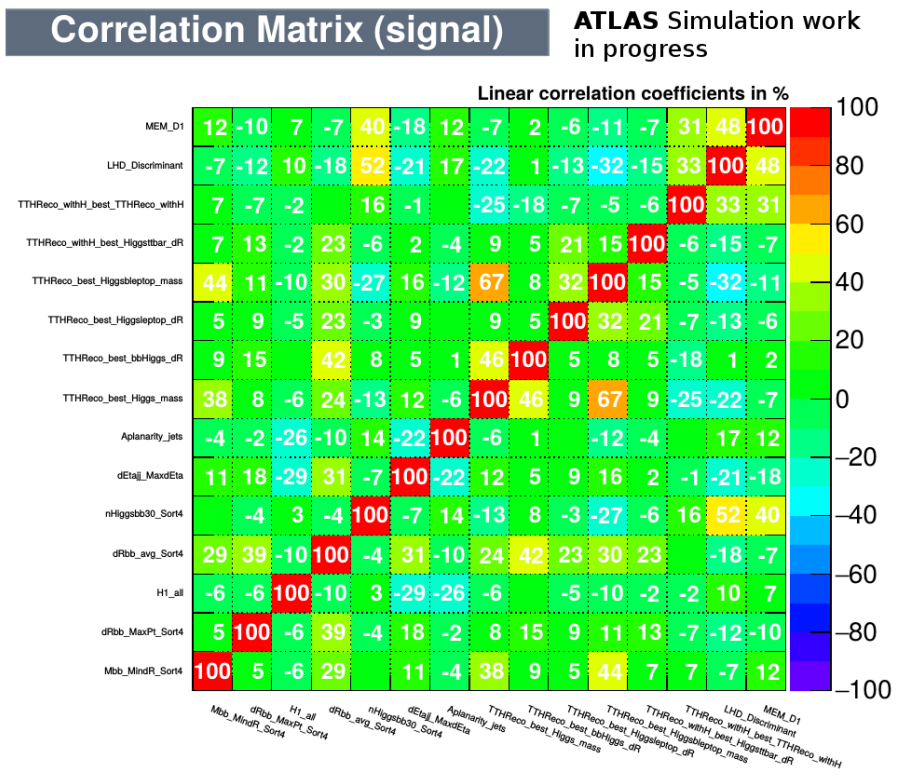


Classification BDT

- Inclusive training: in 4b@85%
- Dedicated training: in 4b@60%, a dedicated BDT is trained with MEM as additional input variable
- Using the PC b-tagging, the inclusive training gives similar performance to the dedicated training. Thus the inclusive training is used in all SRs except in 4b@60%
- Intermediate MVAs output variables are dominant discriminants

	AUC	improve (%)
Kinematics	0.738	-
Kins + RecoBDT	0.756	2.4%
Kins + LHD	0.763	3.4%
Kins + RecoBDT + LHD	0.768	4.1%

- Adding MEM gains a little bit, correlated with LHD

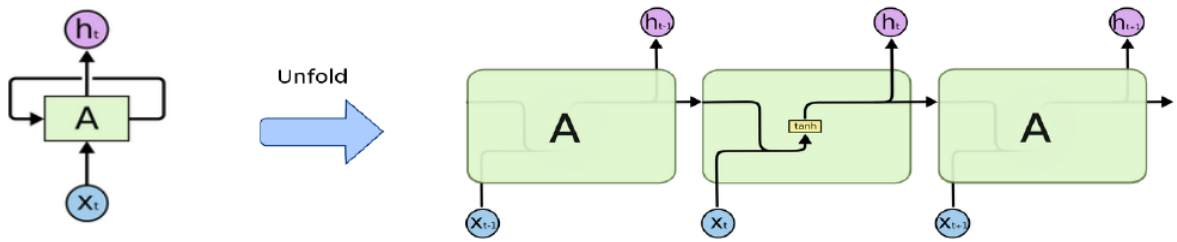


RNN motivation

- Reconstruction step: use similar information but from different aspects
  - MEM: super computationally expensive to run, so only built in one signal rich region.
  - Reco. BDT: takes into account variable correlations. But only use the combination with the highest score, and the truth matching rate is limited.
  - Likelihood discriminant (LHD): use all combinations, but not the correlations between variables.

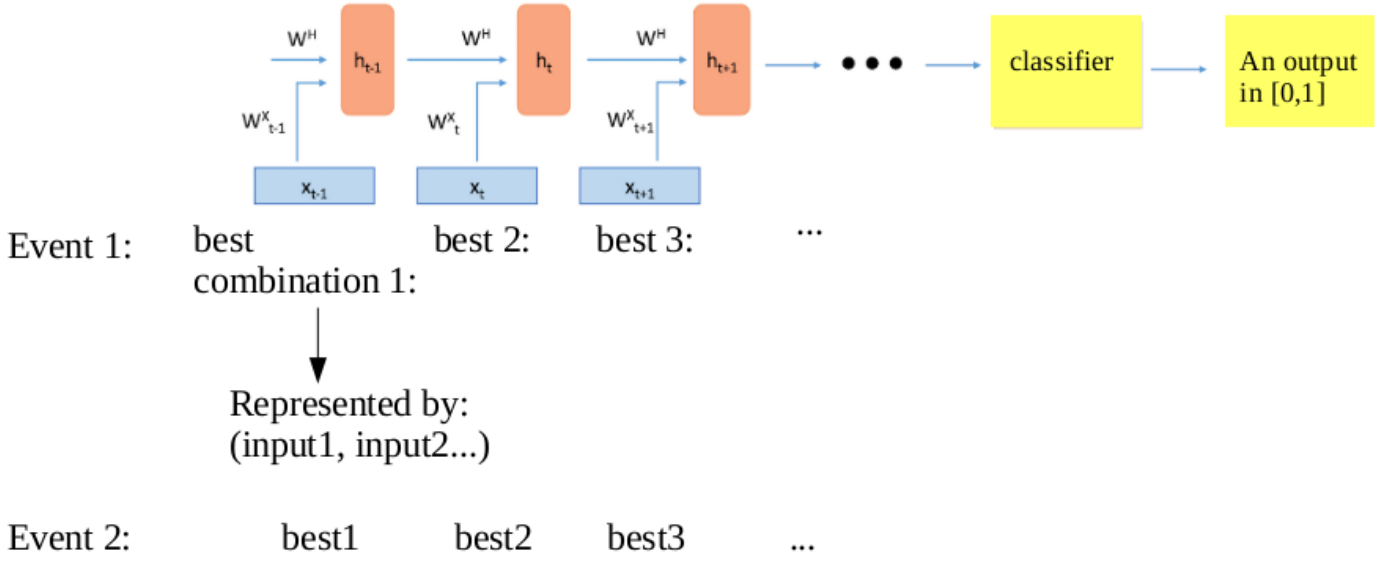
- Classification step: a BDT using info from these reco-level MVAs, classify $t\bar{t}H$ and $t\bar{t}$

- **Goal**: take into account both variable correlations, and more possible combinations



RNN for $t\bar{t}H \rightarrow b\bar{b}$

- Classify $t\bar{t}H$ and $t\bar{t}$ events
- Each event is a sequence, using combinations as frames
- Each frame is represented by the input features of the corresponding combination
- Same input features as reconstruction BDT + classification variables



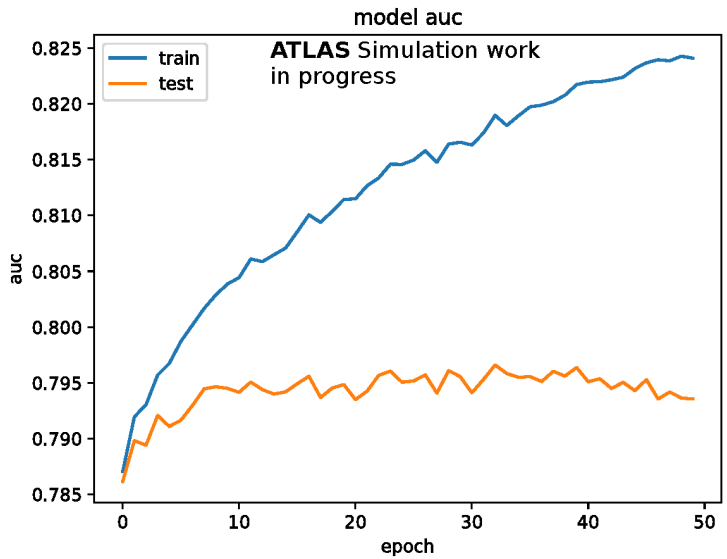
- This way, RNN takes into account more info than reco BDT and LHD: more combinations and proper feature correlations.

Training setup

- Machine learning tools:
  - Keras, Tensorflow, scikit-learn, etc: data science libraries
  - uproot: stream ROOT data into ML libraries
  - Trained on GPU
- Train, validation, test splitting

even		odd
val 20%	learning 80%	test

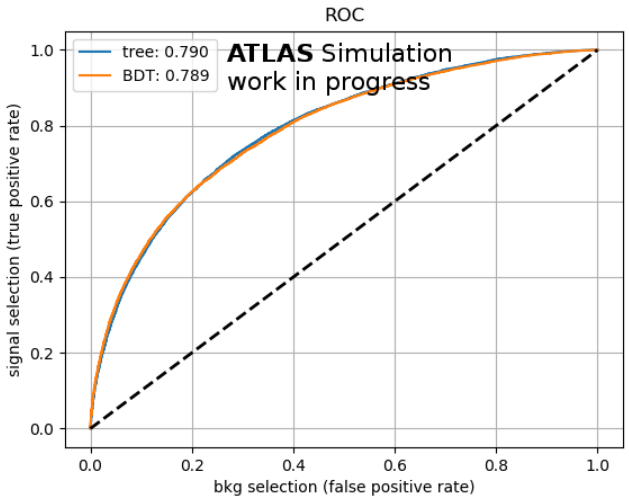
- To avoid over-fitting, monitor the AUC during the training, pick the training epoch with the largest validation AUC.



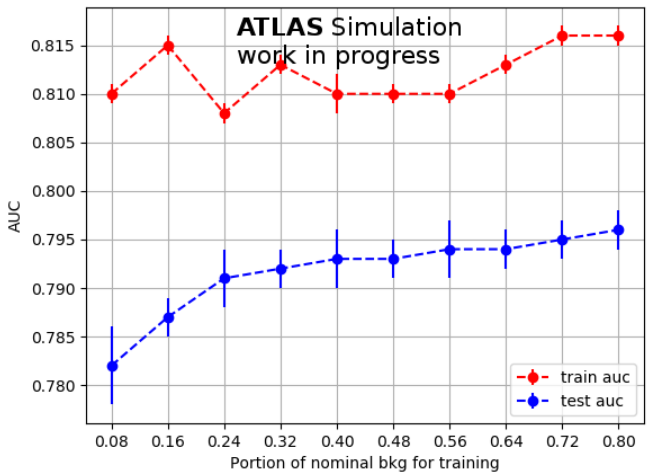
RNN performance

- Models optimized: input $\rightarrow$ 1 RNN layer with 60 cells (dropout values=0.4) $\rightarrow$ output layer with 1 neuron
- Same inputs as Reco BDT (with Higgs info) and classification BDT, w/o intermediate MVAs. Listed in backup .
- RNN (0.790) as good (even slightly better) AUC perf. as BDT (0.789)
- Statistical effect: not much to gain with larger $t\bar{t}$ sample, same for $t\bar{t}H$

ROC



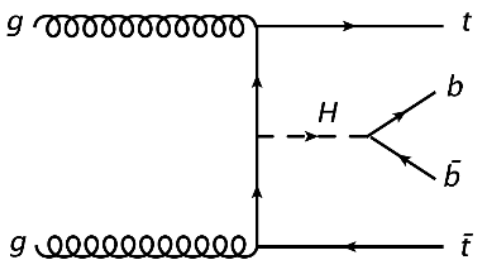
AUC vs. training size

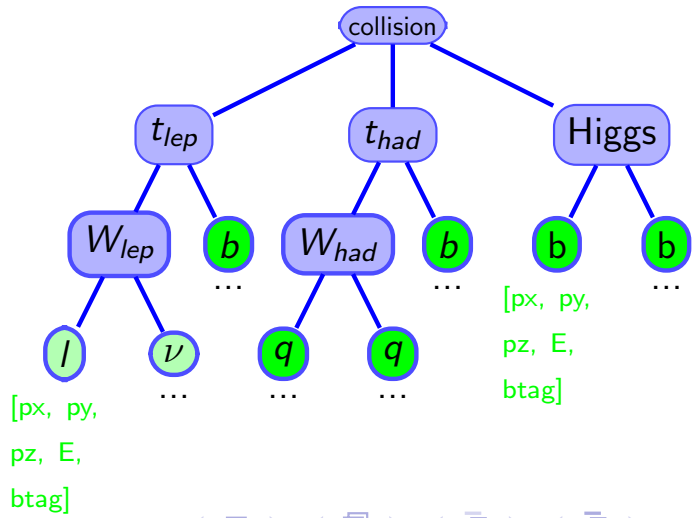


- In 1 step, RNN achieves equivalent performance to 2-step MVAs consisting of various techniques, using the same info.

Parse tree: motivation

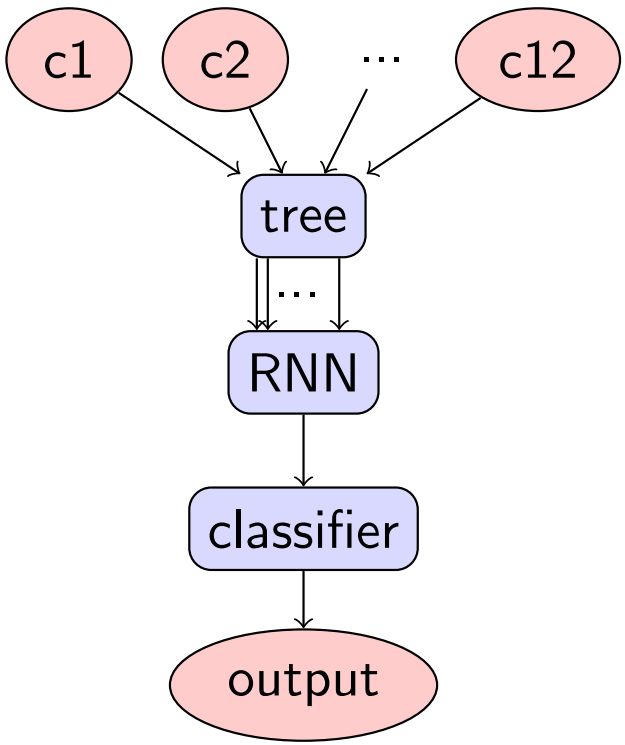
- **Goal**: exploring models trained with low level features
- Combining our domain knowledge + neural networks.
  - Inspired by QCD-Aware Recursive Neural Networks for Jet Physics
- Analogy between parse tree and Feynman diagram
  - Design a tree structure analogous to physical process
  - Use 4-vector and b-tagging of 8 objects as input: 6 jets + lepton and neutrino
  - Go through from the leaves to the collision node, embedding the input space to another n-dimensional space.

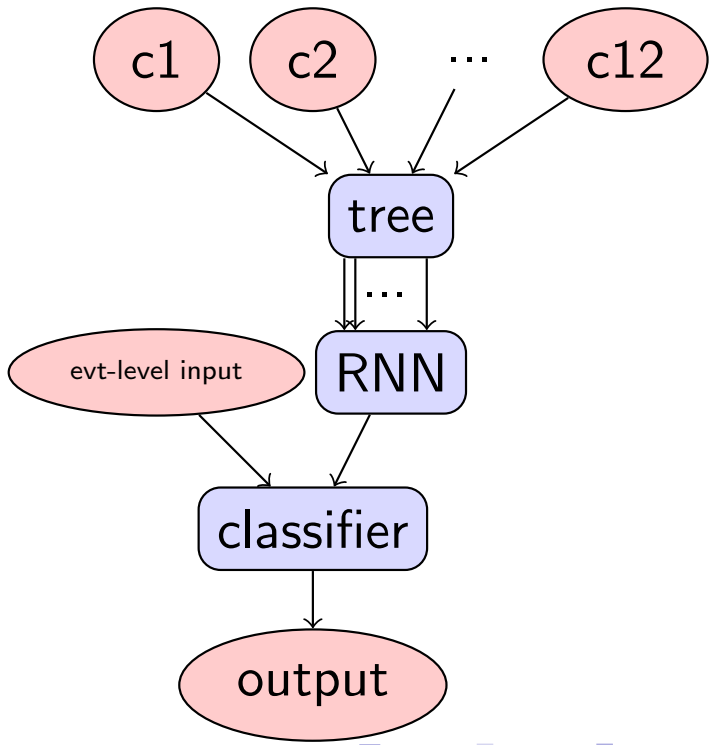




Parse tree model

- Use the tree embedding representation for each combination, making up the sequence input for RNN.
- Also add in high-level inputs: calculated features without combination info, used by BDT: ΔR_{bb}^{arg}, $\Delta R_{bb}^{max,pT}$, $\Delta \eta_{jj}^{max\Delta\eta}$, $m_{bb}^{min\Delta R}$, N_{30}^{Higgs}, $H1$, Aplanarity

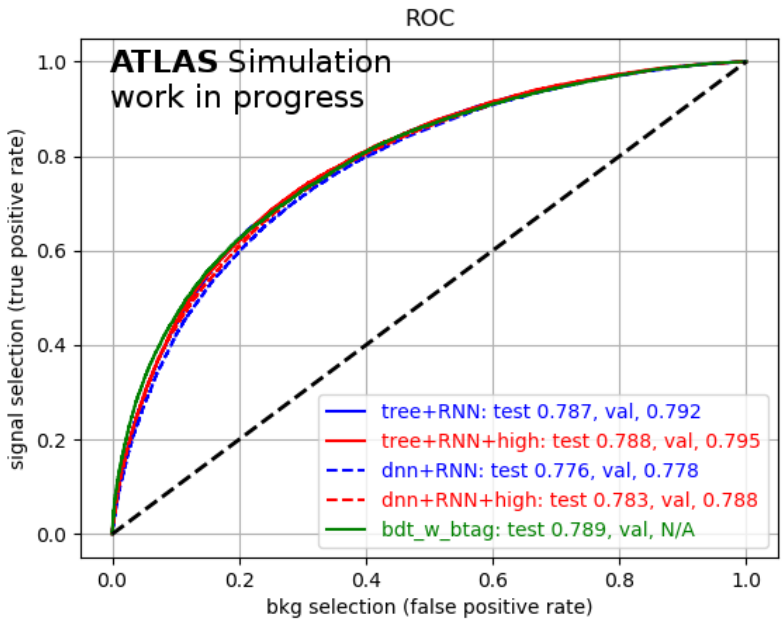




Tree+RNN performance

- Previous BDT and simple RNN are trained with high level features (e.g. combination based and global kinematics)
  - Comparable AUC: BDT (0.789), simple RNN (0.790)
- RNN with (4-vectors, PC *b*-tagging), worse AUC (0.781)
- Tree+RNN could learn useful info from low level features, almost no gains from high level vars
  - Tree+RNN (0.788) with (4-vectors, PC *b*-tagging) as good as BDT/RNN
  - Tree+RNN (0.789) with (4-vectors, PC *b*-tagging, global kinematics), as good as BDT/RNN.

- Replace the tree embedding with a classical DNN. AUC:
  - 0.776 with (4-vectors, PC *b*-tagging)
  - 0.783 with (4-vectors, PC *b*-tagging, global kinematics)
- Tree performance is always better than DNN: tree structure helps to learn from low level features.



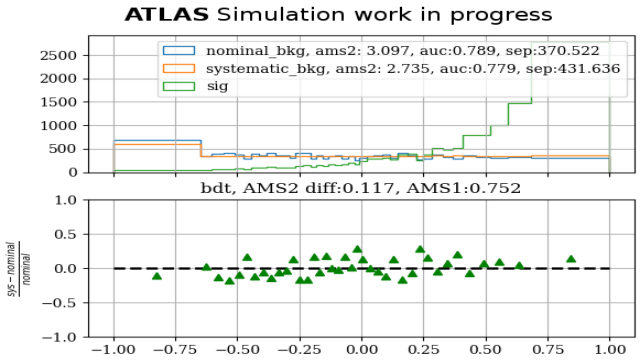
Adversarial training

- Dominant impact on μ is from $t\bar{t} + \geq 1$b shape difference of nominal and systematic samples.
- Difference exists in the nominal and syst samples, but small (but quite large compared to $t\bar{t}H$ presence):

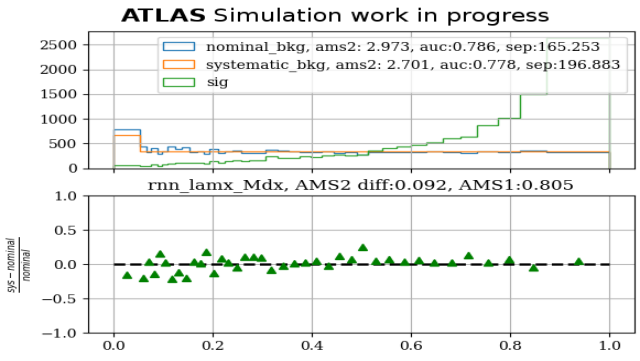
	AUC	Separation
trained on nominal only		
rnn nominal	0.787 ± 0.001	184.416 ± 0.426
rnn syst.	0.778 ± 0.001	222.155 ± 0.895
trained on nominal+syst.		
rnn nominal	0.784 ± 0.001	184.965 ± 0.697
rnn syst.	0.778 ± 0.001	221.890 ± 2.102

- **Goal**: train a classifier insensitive to the difference between nominal and systematic samples. (Following paper: Learning to Pivot with Adversarial Networks)

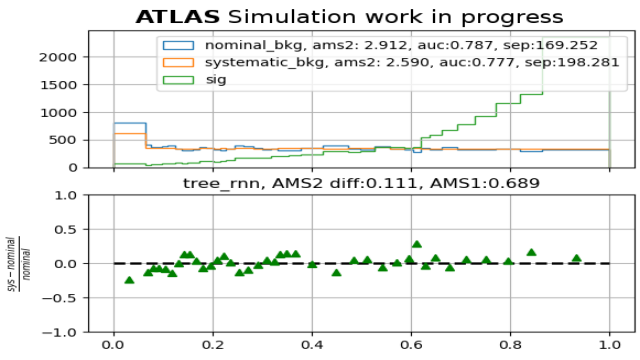
bdt with nominal



rnn with nominal



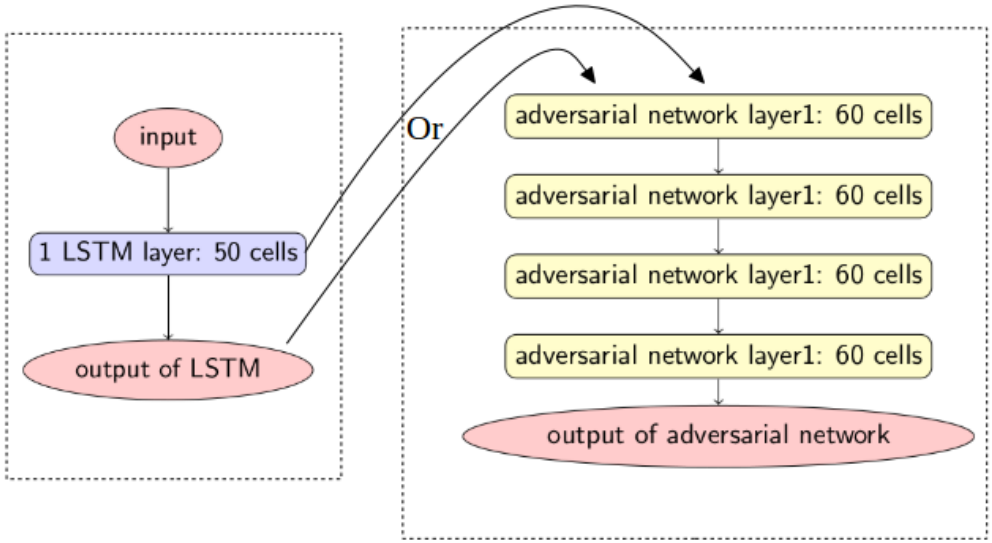
tree+rnn with nominal



Adversarial training

- **Idea**: train a discriminator adversarially to restrict the classifier to have similar outputs for two different samples:
  - Minimax solution: $\hat{\theta}_c, \hat{\theta}_d = \arg\min_{\theta_c}\max_{\theta_d}\left(L_c(\theta_c) - \lambda L_d(\theta_c, \theta_d)\right)$
    - L: loss function, measuring inconsistency between the prediction and label
    - θ: networks connection weights, $\lambda(>0)$: to be tuned

Classifier:
- ttH vs. ttbar
- Train on nominal and syst. samples



Discriminator:
- nominal vs. syst.
- Fed in with the output or the hidden layer of classifier

- Alternative training:
  - Train classifier θ_c, θ_d fixed: $t\bar{t}H$ vs $t\bar{t}$, nominal to be close to syst output
  - Train discriminator θ_d, θ_c fixed: nominal vs syst output.
- Repeated till the θ_d unable to discriminate two samples.

Adversarial training

Experience we learn from lots of experiments with our analysis:

- The shape difference between samples is hard to be separated by discriminator.
- The performance metric we use before likelihood fits, AMS1[1] sum
  - Take into account both signal presence and nominal-systematic difference.

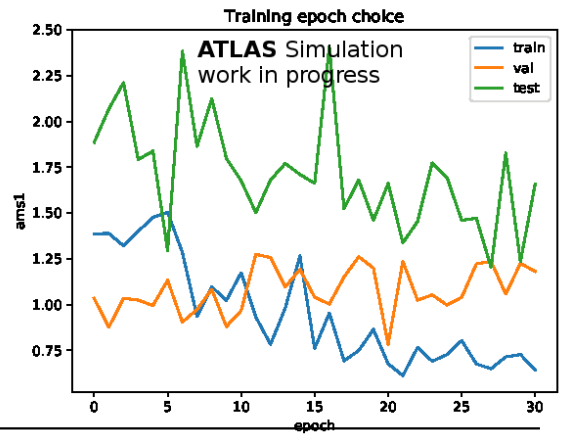
$$\text{AMS1} = \sqrt{\sum_i^N 2((s_i + b_i)\ln(\frac{s_i+b_i}{b0_i}) - s_i - b_i + b0_i) + \frac{(b_i - b0_i)^2}{\sigma_{bi}^2}}, \text{ N=total number of bins}$$

$$\sigma_{bi} = |b0_i - b1_i|$$

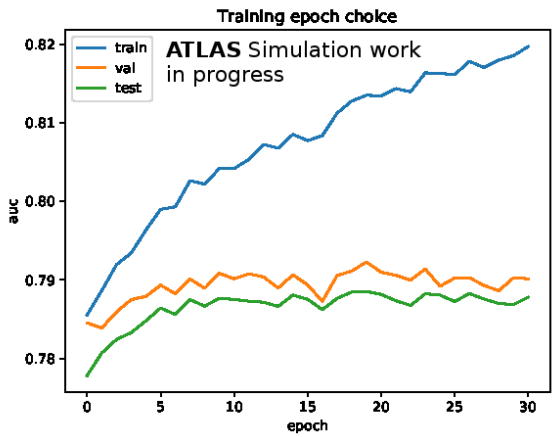
$$b0_i = \frac{1}{2}(bi - \sigma_{bi}^2 + \sqrt{(b_i - \sigma_{bi}^2)^2 + 4(s_i + b_i)\sigma_{bi}^2})$$

  - Unstable value, even after rebinning to have more stats in each bin. Even sensitive to GPU randomness.

AMS1 vs training evolution



AUC vs training evolution



[1] See the reference

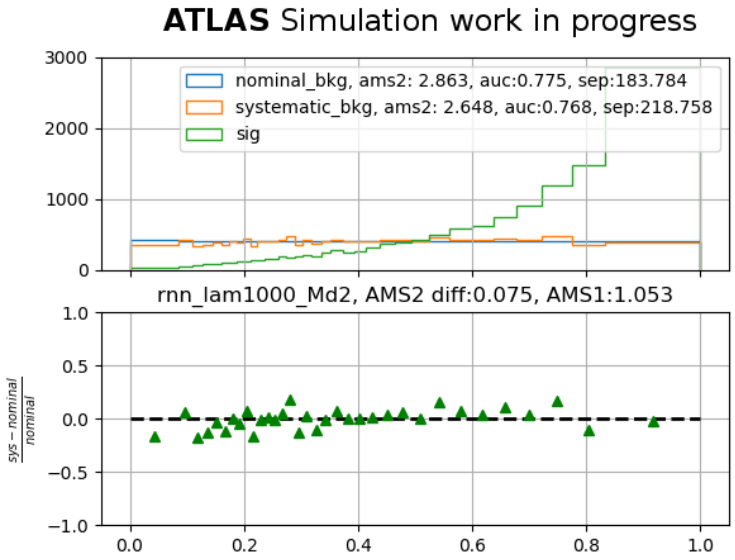
A robust RNN with adversarial networks?

- Not completely clear how much improvement this brings:
  - Output distributions show no clear evidence. Plots below are the RNN output w/ and w/o adversarial training.
  - AMS1 is improved, but with large uncertainty
  - BDT (trained on nominal only) AMS1: 0.752, AUC:0.789

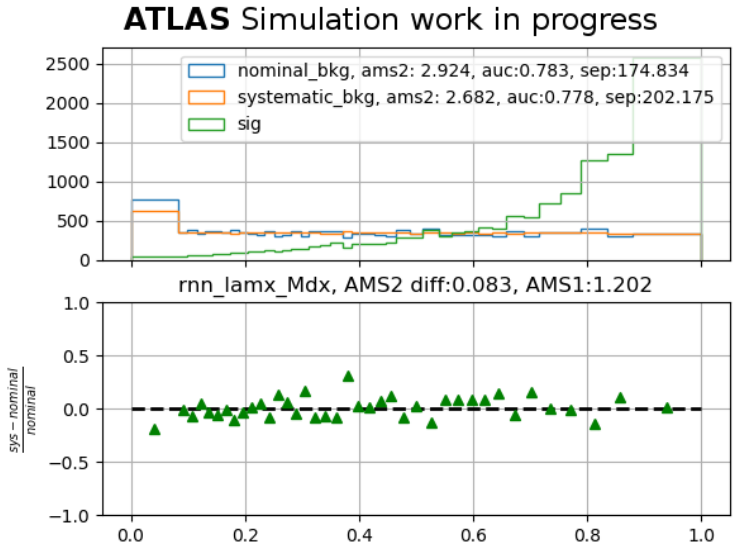
	AUC	AMS1
nominal	0.771 ± 0.004	0.993 ± 0.189
syst.	0.762 ± 0.005	

	AUC	AMS1
nominal	0.784 ± 0.001	0.942 ± 0.149
syst.	0.778 ± 0.001	

w/ adversarial training

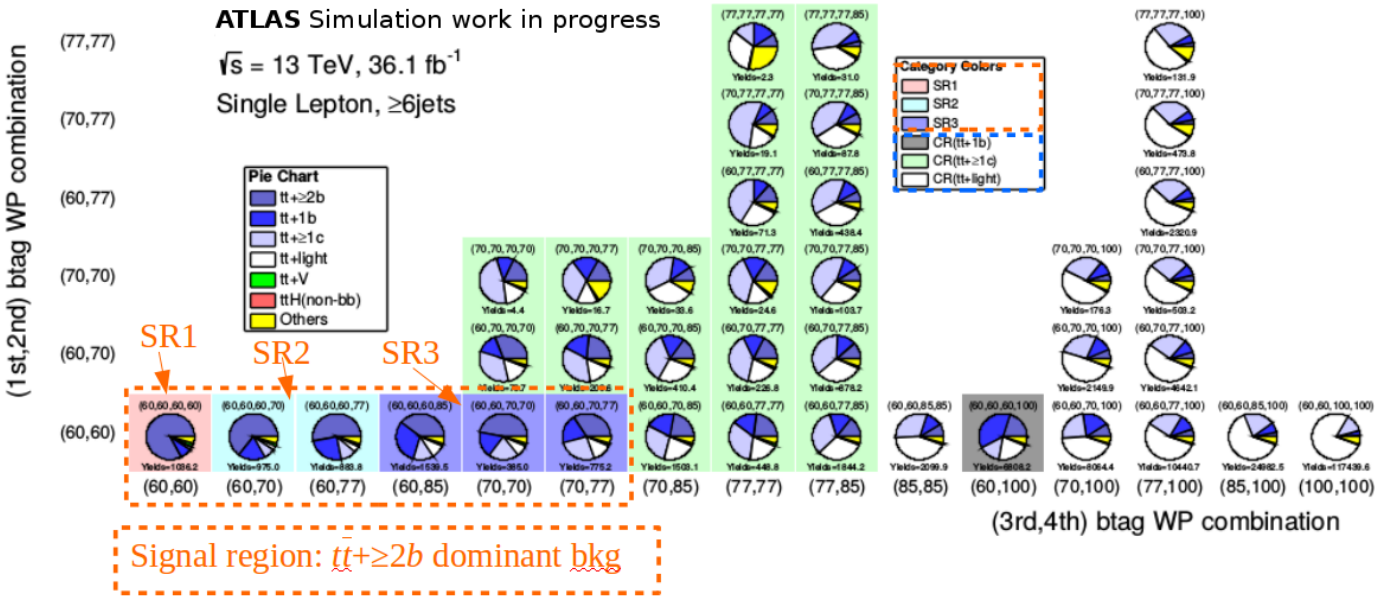


w/o adversarial training



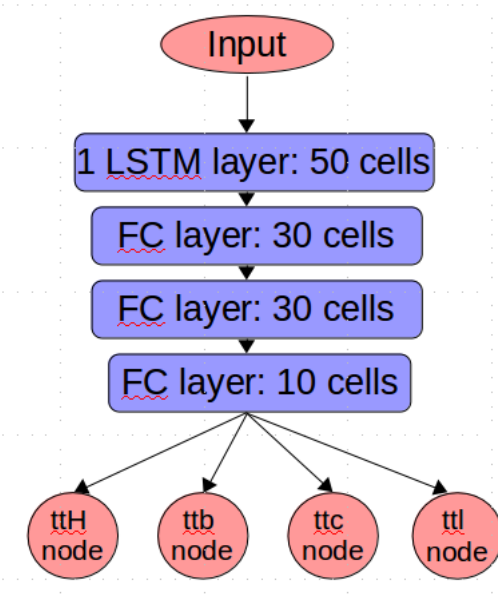
Event categorization

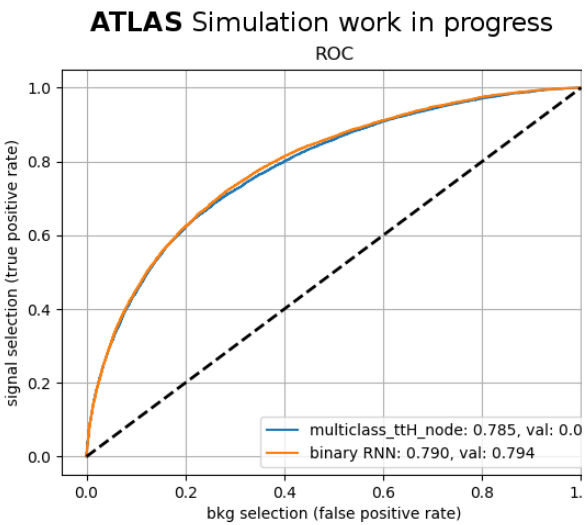
- To improve the significance, events are categorized into orthogonal regions: signal regions (SR) and control regions (CR)
- Latest publication:
  - Splitting events into subsets w.r.t to jets pseudo-continuous b-tag score
  - Examining the bkg composition manually
  - Merge the subsets with similar bkg

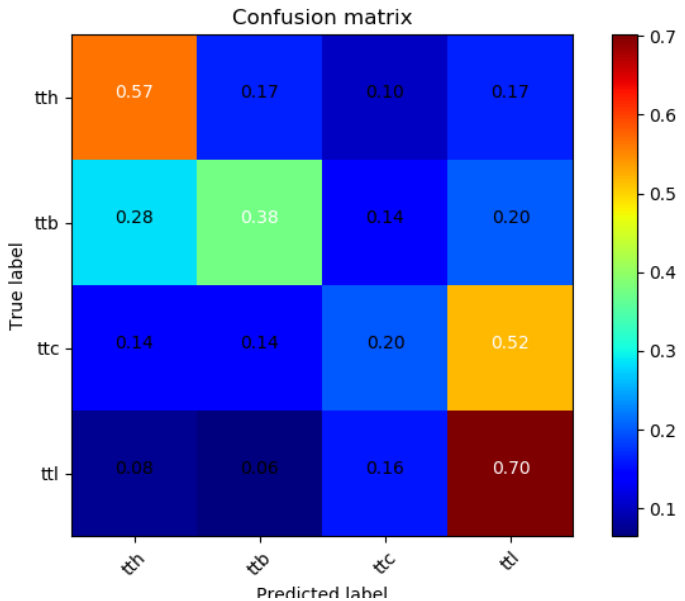


Multi-output classification

- Neural network classifier: a natural way to have multi-output
  - 1 RNN layer (50 cells) + Fully Connected (FC) layers + 4 output nodes ($t\bar{t}H, t\bar{t}b, t\bar{t}l, t\bar{t}c$)
- Model structure not optimized
- Balancing the $t\bar{t}H$, $t\bar{t}b$, $t\bar{t}c$, $t\bar{t}l$ for training
- Could split further the $t\bar{t}b$ category







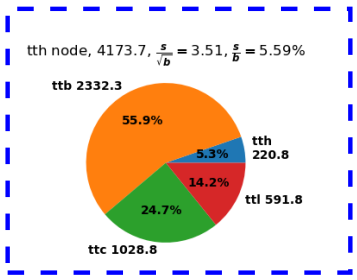
Usage of multi-output

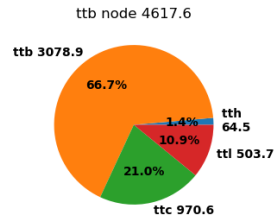
More studies are needed to find an optimal way to use the multi-output.

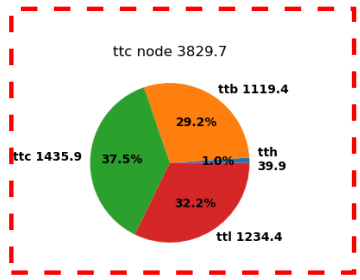
- Multi-dimensional cut
- Foam approach
- Following CMS strategy:
  - Categorize the event as *ttH* if the *ttH* node gives the highest score. Same for $t\bar{t}b$, $t\bar{t}c$, $t\bar{t}l$
  - Using the output cells as region definition:
    - *tth* cell: signal rich region
    - *ttb* cell: ttb rich region
    - ...
  - Use output node associated to category for simultaneous fit

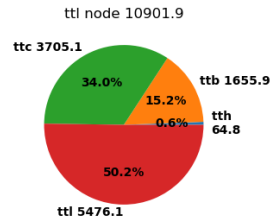
Background decomposition (ATLAS simulation work in progress)

- Region definition based on multiclass output

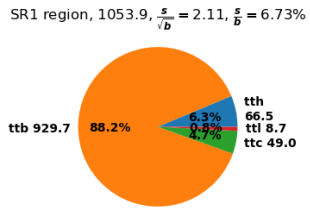


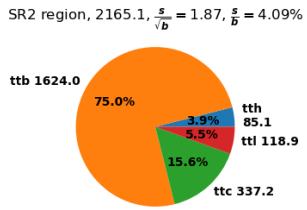


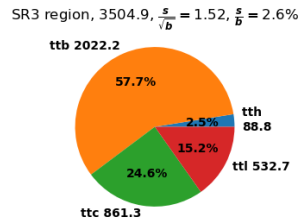


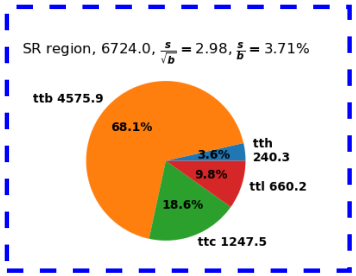


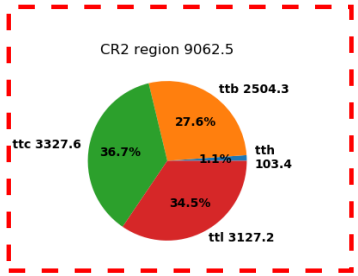
- Default definition

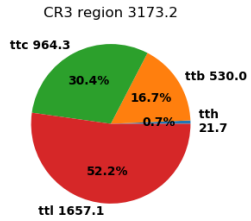












Comparing two methods:

- The tth node gives similar number of $t\bar{t}H$ events, better s/b, s/$\sqrt{b}$ w.r.t to default inclusive SR
- SR1 and SR2 give less ttc & ttl constituents.
- Less events in CR2 (ttc).

Summary

In the analysis with 36.2fb^{-1} of data, two-step MVAs are used

- Event reconstruction step and signal/bkg classification
- PC b-tagging variables are used in both steps

Exploring deep learning techniques for the full Run 2:

- Binary classification:
  - RNN: better explore reconstruction combinatorics due to recurrent structure, one-step classification
  - Parse tree: include physics domain knowledge while designing the neural networks, more efficient to learn from low level features
- Systematic uncertainty reduction:
  - Use an adversarial network that discriminates between $t\bar{t}$ models, to reduce modeling systematics. AUC pays the price, but AMS1 sum gains
- Region definition:
  - More flexible and natural with neural networks to have multi-outputs
- Complete fit studies with these new models are underway, preliminary results of simple RNN and multiclass are promising.

Backup

Reconstruction BDT inputs

Variable	Region	
	≥ 6j	5j
Topological information from $t\bar{t}$		
t_{lep} mass	✓	✓
t_{had} mass	✓	–
Incomplete t_{had} mass	–	✓
W_{had} mass	✓	–
Mass of W_{had} and b from t_{lep}	✓	✓
Mass of W_{lep} and b from t_{had}	✓	✓
$\Delta R(W_{\text{had}}, b \text{ from } t_{\text{had}})$	✓	✓
$\Delta R(W_{\text{had}}, b \text{ from } t_{\text{lep}})$	✓	✓
$\Delta R(\text{lep}, b \text{ from } t_{\text{lep}})$	✓	✓
$\Delta R(\text{lep}, b \text{ from } t_{\text{had}})$	✓	✓
$\Delta R(b \text{ from } t_{\text{lep}}, b \text{ from } t_{\text{had}})$	✓	✓
$\Delta R(q_1 \text{ from } W_{\text{had}}, q_2 \text{ from } W_{\text{had}})$	✓	–
$\Delta R(b \text{ from } t_{\text{had}}, q_1 \text{ from } W_{\text{had}})$	✓	–
$\Delta R(b \text{ from } t_{\text{had}}, q_2 \text{ from } W_{\text{had}})$	✓	–
min. $\Delta R(b \text{ from } t_{\text{had}}, q \text{ from } W_{\text{had}})$	✓	–
min. $\Delta R(b \text{ from } t_{\text{had}}, q \text{ from } W_{\text{had}}) - \Delta R(\text{lep}, b \text{ from } t_{\text{lep}})$	✓	✓
Topological information from Higgs		
Higgs boson mass	✓	✓
Mass of Higgs and q_1 from W_{had}	✓	✓
$\Delta R(b_1 \text{ from Higgs}, b_2 \text{ from Higgs})$	✓	✓
$\Delta R(b_1 \text{ from Higgs}, \text{lep})$	✓	✓
$\Delta R(b_1 \text{ from Higgs}, b \text{ from } t_{\text{lep}})$	–	✓
$\Delta R(b_1 \text{ from Higgs}, b \text{ from } t_{\text{had}})$	–	✓

Input variables to the reconstruction BDT in the single lepton channel. The subscript had(lep) indicates the hadronically (leptonically) decaying W or t and q_i refers to quarks from W.

classification BDT inputs

PhysRevD.97.072016

Variable	Definition	Region	
		≥ 6j	5j
General kinematic variables			
$\Delta R_{bb}^{\text{avg}}$	Average ΔR for all b-tagged jet pairs	✓	✓
$\Delta R_{bb}^{\text{max } p_{\text{T}}}$	ΔR between the two b-tagged jets with the largest vector sum p_{T}	✓	–
$\Delta\eta_{jj}^{\text{max } \Delta\eta}$	Maximum $\Delta\eta$ between any two jets	✓	✓
$m_{bb}^{\text{min } \Delta R}$	Mass of the combination of the two b-tagged jets with the smallest ΔR	✓	–
$m_{jj}^{\text{min } \Delta R}$	Mass of the combination of any two jets with the smallest ΔR	–	✓
$\text{N}_{30}^{\text{Higgs}}$	Number of b-jet pairs with invariant mass within 30 GeV of the Higgs boson mass	✓	✓
$H_{\text{T}}^{\text{had}}$	Scalar sum of jet p_{T}	–	✓
$\Delta R_{\text{lep}-bb}^{\text{min } \Delta R}$	ΔR between the lepton and the combination of the two b-tagged jets with the smallest ΔR	–	✓
Aplanarity	$1.5\lambda_2$, where λ_2 is the second eigenvalue of the momentum tensor [91] built with all jets	✓	✓
$H1$	Second Fox–Wolfram moment computed using all jets and the lepton	✓	✓

Variables from reconstruction BDT output			
BDT	BDT output	✓*	✓*
m_H	Higgs boson mass	✓	✓
$m_{H,b_{\text{lep top}}}$	Mass of Higgs boson and b-jet from leptonic top	✓	–
$\Delta R_{\text{Higgs bb}}$	ΔR between b-jets from the Higgs boson	✓	✓
$\Delta R_{H,t\bar{t}}$	ΔR between Higgs boson and $t\bar{t}$ system	✓*	✓*
$\Delta R_{H,\text{lep top}}$	ΔR between Higgs boson and leptonic top	✓	–
$\Delta R_{H,b_{\text{had top}}}$	ΔR between Higgs boson and b-jet from hadronic top	–	✓*
Variable from Likelihood calculation			
D	Likelihood discriminant	✓	✓
Variable from Matrix Method calculation			
MEM_{D1}	Matrix Method	✓	–
Variables from b-tagging			
w_b^H	Sum of binned b-tagging weights of jets from best Higgs candidate	✓	✓
B_{j^3}	3rd jet binned b-tagging weight (sorted by weight)	✓	✓
B_{j^4}	4th jet binned b-tagging weight (sorted by weight)	✓	✓
B_{j^5}	5th jet binned b-tagging weight (sorted by weight)	✓	✓

Table 8: Input variables to the classification BDT in the single-lepton channel. For variables from the reconstruction BDT, those with a * are from the BDT using Higgs boson information, while those with no * are from the BDT without Higgs boson information. The MEM_{D1} variable is only used in the UPSR, while b-tagging weights are not used in this region (no information as they are all equal, by construction).

LSTM and multiclassification inputs for 6j case

Reconstruction info

	Topological information from $t\bar{t}$
	t_{lep} mass
	t_{had} mass
15 RecoBDT inputs w/o Higgs info	W_{had} mass
	Mass of W_{had} and b from t_{lep}
	Mass of W_{lep} and b from t_{had}
	$\Delta R(W_{had}, b \text{ from } t_{had})$
	$\Delta R(W_{had}, b \text{ from } t_{lep})$
	$\Delta R(\text{lep}, b \text{ from } t_{lep})$
	$\Delta R(\text{lep}, b \text{ from } t_{had})$
	$\Delta R(b \text{ from } t_{lep}, b \text{ from } t_{had})$
	$\Delta R(q_1 \text{ from } W_{had}, q_2 \text{ from } W_{had})$
	$\Delta R(b \text{ from } t_{had}, q_1 \text{ from } W_{had})$
	$\Delta R(b \text{ from } t_{had}, q_2 \text{ from } W_{had})$
	min. $\Delta R(b \text{ from } t_{had}, q \text{ from } W_{had})$
	min. $\Delta R(b \text{ from } t_{had}, q \text{ from } W_{had}) - \Delta R(\text{lep}, b \text{ from } t_{lep})$
RecoBDT inputs Higgs info	Higgs boson mass
	Mass of Higgs and q_1 from W_{had}
	$\Delta R(b_1 \text{ from Higgs}, b_2 \text{ from Higgs})$
	$\Delta R(b_1 \text{ from Higgs}, \text{lep})$

Global classification BDT inputs

General kinematic variables	
$\Delta R_{bb}^{\text{avg}}$	Average ΔR for all b-tagged jet pairs
$\Delta R_{bb}^{\text{max } p_T}$	ΔR between the two b-tagged jets with the largest vector sum p_T
$\Delta\eta_{jj}^{\text{max } \Delta\eta}$	Maximum $\Delta\eta$ between any two jets
$m_{bb}^{\text{min } \Delta R}$	Mass of the combination of the two b-tagged jets with the smallest ΔR
N_{30}^{Higgs}	Number of b-jet pairs with invariant mass within 30 GeV of the Higgs boson mass
Aplanarity	$1.5\lambda_2$, where λ_2 is the second eigenvalue of the momentum tensor [91] built with all jets
$H1$	Second Fox–Wolfram moment computed using all jets and the lepton

B-tagging weights

- MVAreco_b1Higgs_pseudobtag_5bins
- MVAreco_b2Higgs_pseudobtag_5bins
- MVAreco_blepTop_pseudobtag_5bins
- MVAreco_qhadW_pseudobtag_5bins_1
- MVAreco_qhadW_pseudobtag_5bins_2
- MVAreco_bhadTop_pseudobtag_5bins

AMS1 and AUC stability

- Trials with 100 random seeds
- 100 times of experiments. train, val, test fixed, or train and val shuffled for each trial.
- AMS1 of BDT: 1.397.

Table: TransfoD AMS1 (mean/std(%))

	Fixed	Shuffled
train	.756±.146(19.3)	.840±.211(25.1)
val	1.080±.136(12.6)	.918±.187(20.4)
test	1.765±.304(17.2)	1.849±.300(16.2)

Table: Old transform AMS1 (mean/std(%))

	Fixed	Shuffled
train	.616±.109(17.7)	.654±.126(19.3)
val	1.035±.146(14.1)	.840±.241(28.7)
test	1.635±.365(22.3)	1.670±.433(25.9)

Table: TransfoD: AMS2_diff (mean/std(%))

	Fixed	Shuffled
train	.115±0.016(13.9)	.108±.020(18.5)
val	.017±.016(94.1)	.027±.041(152)
test	.029±.009(31.0)	.026±.009(34.6)

Table: Old transform: AMS2_diff (mean/std(%))

	Fixed	Shuffled
train	.084±0.015(17.9)	.080±.018(22.5)
val	.009±.013(144)	.015±.035(233)
test	.019±.006(31.6)	.019±.007(36.8)

Table: AUC0 (mean/std)

	Fixed	Shuffled
train	.814±.005	.813±.006
val	.793±.001	.792±.004
test	.789±.001	.788±.001

Table: AUC1 (mean/std)

	Fixed	Shuffled
train	.792±.002	.792±.003
val	.789±.002	.788±.007
test	.778±.001	.778±.001