TREE BOOSTING (+NEURAL NETWORKS) FOR EFT ANALYSES

R. Schöfbeck (HEPHY Vienna), Oct. 24th, 2023, Area 3 meeting
A CONDITIONAL SEQUENCE

adapted from arXiv:2211.01421

$p(x_{\text{det}}|\theta) = \int d\zeta_{\text{ptl}} \int d\zeta_{\text{p}} \cdots p(x_{\text{det}}|\zeta_{\text{ptl}})$

ML facilitates this inversion by exploiting that simulation samples

$x_{\text{det}}, \zeta_{\text{ptl}}, \zeta_{\text{p}} \sim p(x_{\text{det}}, \zeta_{\text{ptl}}, \zeta_{\text{p}}|\theta)$

“Simulation based inference”

1. Generators run in ‘forward mode’
2. Pick up uncertainties

$\frac{1}{\sigma_{\theta}} \frac{d\sigma_{\theta}}{d\zeta_{\text{p}}} = p(\zeta_{\text{p}}|\theta)$

parton-level differential cross section $\sim$ pdf

$\theta$ NOT stochastic; Frequentist
Event classification $\theta \rightarrow \text{isSig} \in \{0, 1\}$

$$L = \langle f(x)^2 \rangle_{\text{isSig}=1} + \langle (1 - f(x))^2 \rangle_{\text{isSig}=0} = \sum_{\text{isSig}} \int dx \cdots$$

$$f^*(x) = \frac{1}{1 + \frac{p(x|\text{isSig}=1)}{p(x|\text{isSig}=0)}}$$

Learn LR by classification; “Likelihood ratio trick” achieve NP optimality (for x-sec)
We can try to learn EFT effects “on average”

- Sending ‘mixed signals’ to the loss function
  - Averages the training data set - suboptimal when linear effects dominate
  - Classifier does not reflect knowledge on the $\theta$-dependence

1. Back to the drawing board & inject $\theta$ polynomial SMEFT dependence in estimator.
2. Exploit the fact that SMEFT predictions can (optionally) be weighted: Only need one training data set
EXPLOITING SMEFT REWEIGHTING

\[ L = \sum_{\theta \in B} \left( \langle \hat{f}(x; \theta)^2 \rangle_{\theta} + \langle (1 - \hat{f}(x; \theta))^2 \rangle_{SM} \right) \]

\[ = \sum_{\theta \in B} \int dx \, dz \left( p(x, z|\theta) \hat{f}(x; \theta)^2 + p(x, z|SM)(1 - \hat{f}(x; \theta))^2 \right) \]

Let's write this under one integral \( z \ldots \text{latent space} \)

We start with SM and BSM samples

\[ r = \frac{p(x_{\text{det}}, \ldots, z_{\text{prt}}, \ldots, z_p|\theta)}{p(x_{\text{det}}, \ldots, z_{\text{prt}}, \ldots, z_p|SM)} = \frac{p(x_{\text{det}}|z_{\text{prt}}) \cdots p(z_{\text{prt}}|z_p) \cdots p(z_p|\theta)}{p(x_{\text{det}}|z_{\text{prt}}) \cdots p(z_{\text{prt}}|z_p) \cdots p(z_p|SM)} \sim \frac{p(z_p|\theta)}{p(z_p|SM)} \sim \frac{|M(z_p, \theta)|^2}{|M(z_p, SM)|^2} = w(\theta) \]

Change in likelihood of simulated observation \( x \)
with latent "history" \( z \) going from "SM" to \( \theta \)

staged simulation in forward mode:
Intractable factors cancel
re-calculable theory prediction
weighted simulation
PARAMETRIZED CLASSIFIERS

\[ L = \sum_{\theta \in \mathcal{B}} \int dx \, dz \, p(x, z | \text{SM}) \left( r(x, z | \theta) \hat{f}(x; \theta)^2 + (1 - \hat{f}(x; \theta))^2 \right) \]

Similar to S. Chen, A. Glioti, G. Panico, A. Wulzer

*JHEP 05 (2021) 247*

arXiv:2308.05704

\[ \hat{f}(x; \theta) = \frac{1}{1 + \hat{R}(x; \theta)} \]

invert likelihood trick

insert model knowledge: fit universal coefficient functions

\[ \hat{R}(x; \theta) = \left( 1 + \sum_a \theta_a \hat{n}_a(x) \right)^2 + \sum_a \left( \sum_{b \geq a} \theta_b \hat{n}_{ab}(x) \right)^2 \]

MSE or cross entropy

Why would you want to use trees instead?

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]

**PARAMETRIZED CLASSIFIERS**

\[ \quad \]
A SIMPLE TREE ALGORITHM

• A tree is a hierarchical phase-space partitioning ($\mathcal{J}$)
  • the novelty in the Boosted Information Tree is that we associate each region $j$ with a polynomial $F_j(\theta)$
  • Note: A tree algorithm can have an arbitrarily complicated predictive function; here it is a SMEFT polynomial

• Fitting tree: Optimize “node split positions” on some loss. Trained (e.g. greedily) on the ensemble.
Want to regress in $r$, exploiting its the polynomial $\theta$ dependence

$$r(x, z|\theta) = \frac{d\sigma(x, \theta)/dx}{d\sigma(x, SM)/dx}$$

→ will allow to compute the optimal LLR test statistic $q(D)$

$$L = \sum_{\theta \in B} \int dx \, dz \, \rho(x, z|SM) \left( r(x, z|\theta) - \hat{F}(x, \theta) \right)^2$$

Tree ansatz

$F_j(\theta)$ polynomial with const. coeff. (per node)

find optimal partitioning

find optimal predictor

Remove DOF from predictor:

$$F_j(\theta) = \frac{\sum_{i \in j} w_i(\theta)}{\sum_{i \in j} w_i(\theta_0)}$$

sum up event weights within node & divide

No trainable parameters in the predictor

(Regression by means of classification)

Solve for optimal partitioning with greedy CART algorithm

$$L = -\sum_{\theta \in B} \sum_{j \in J} w_j^2(\theta) \left( \sum_{j \in J} w_j(\theta_0) \right) = -\sum_{j \in J} \sum_{\theta \in B} \theta^a \theta^b l_{aj}^{(j)} + O(\theta - \theta_0)^3$$

We’re optimizing the Fisher information!

We’ll find an optimized tree.

→ boost

CONCRETE SOLUTION: TREE BOOSTING

• Boosting: Fit model iteratively to pseudo-residuals of the preceding iteration with learning rate $\eta$

• Ansatz:

$$\hat{F}^{(b)}(x, \theta) = \hat{f}(x, \theta) + \eta \hat{F}^{(b-1)}(x, \theta)$$

• Insert into the loss function:

$$L[\hat{f}^{(b)}] = \sum_{\theta \in \Theta} \int \int dx \, dz \, p(x, z|SM) \left( r(x, z|\theta) - \eta \hat{F}^{(b-1)}(x, \theta) - \hat{f}^{(b)}(x, \theta) \right)^2$$

pseudo-residual, amounting to event-level reweighting

$$w_i^{(b)}(\theta) \rightarrow w_i^{(b-1)}(\theta) - \eta w_i^{(b-1)}(\theta_0) \hat{F}^{(b-1)}(x_i, \theta)$$

.... perform this iteratively “Boosted Information Tree”
1D TOY EXAMPLE

• pdf(x|θ) = 1/(2π) - θ Sin(x)

• We predict the first derivative of the pdf wrt. to the parameter
- **Realistic case**: model of the ZH process
- “Boosted Information Tree (BIT)”
  - 3 WC, 9 DOF, 500k events, ZH
  - 200 trees, D=5, 9 minutes of training
  - also more realistic study, including backgrounds [2107.10859], [2205.12976]
- Learning coefficient functions to compute parametrized optimal observables

\[
\sigma^{SM} + \frac{\theta}{\Lambda^2} \sigma^{\text{int}} + \frac{\theta^2}{\Lambda^4} \sigma^{\text{quad}} \rightarrow \text{parametrized } q_\theta(D)
\]
Obtain parametrized classifiers with 20-40% improvements in 2D toy cases (NOT marginalized!)

No free lunch – Analysis dependent choices are needed
  • Binned analysis: variable binning → background estimation is CPU intensive
  • Systematics treatment for unbinned analyses (beyond Higgs $M_{4\ell}$) less far developed

Is it all worth it in higher dimensions? Yes! More examples: [ML4EFT]; full list of references in backup
Given a phase space region with EFT dependence: NN must select & predict

- In the Boosted Information Tree, the weak learner only selects
  - The prediction ($F_j$) is computed from the boxed events → integrates latent space
  - The regression problem is solved with computational complexity of classification
    - Speed advantage at high operator dimensions!

\[
F_j(\theta) = \frac{\sum_{i \in j} w_i(\theta)}{\sum_{i \in j} w_i(\theta_0)} = \frac{\int dz \frac{d\sigma}{d(x,z)}}{\int dz \frac{d\sigma_{SM}}{d(x,z)}}
\]
SUMMARY!

• Various algorithms predict SMEFT dependence, mostly capitalizing on weighted simulation
  • Provide NP-optimal observables for hypothesis tests
• Trees eliminate the latent space integration and suitable to weight-based SMEFT predictions
• Parametrized classifiers “ask” for unbinned analysis

Get in touch with us for an in-person walkthrough

```
from MultiBoostedInformationTree import MultiBoostedInformationTree
bit = MultiBoostedInformationTree(
    training_features = training_features,
    training_weights = training_weights,
    base_points = base_points,
    feature_names = model.feature_names,
    **model.multi_bit_cfg
)
bit.boost()

test_predictions = bit.vectorized_predict(test_features)
```

+ 500 lines for figs
GOALS FOR MACHINE-LEARNING \textbf{OF EFT}

- SMEFT effects can be
  1. \textbf{in the tails of the distributions} because, e.g.
     4-point functions grow with energy
  2. \textbf{in angular observables & correlations,}
     sometimes encoding CP-violating effects
     - “interference resurrection” \textbf{PLB 2017 11 086}
     “method of moments” \textbf{JHEP 06 (2021) 031}
     - Enhance / single out the linear term
       - Up to triple-angular correlations,
         x5-10 boost in sensitivity
  3. \textbf{on top of “kinematically complex” backgrounds}
     - Def: Usually amenable to classification MVAs
     - Unify the training target with classification

Tree-level SMEFT amplitude of ZH (transverse polarisation):

\[
\mathcal{M}_\lambda = \pm g_Z m_Z \sqrt{s} \left[ \frac{g_Z \sigma}{\bar{s} - m_Z^2} \right] \frac{c_{\theta_\omega}}{\bar{s} - m_H^2} \left( 1 + \frac{\bar{s} - m_H^2}{m_Z^2} \right) \left( \frac{g_Z \sigma c_{\theta_\omega}}{\bar{s} - m_H^2} + \frac{Q_{q_e} e \theta_{q_e}}{\bar{s}} \right) \frac{v^2}{\Lambda^2} C_{\text{HW}}
\]

\[
- 2i \bar{k} \sqrt{\bar{s}} \frac{c_{\theta_\omega}}{m_Z^2} \left( \frac{g_Z \sigma c_{\theta_\omega}}{\bar{s} - m_Z^2} + \frac{Q_{q_e} e \theta_{q_e}}{\bar{s}} \right) \frac{v^2}{\Lambda^2} C_{\text{HW}} + g_Z^2 \sqrt{\bar{s}} \frac{v^2}{\Lambda^2} C_{\text{HQ}}
\]
HOW TO PARAMETRIZE?

- Quantum field theory: Differential cross section predict polynomial SM-EFT dependence:

\[ d\sigma(\theta) \propto |M_{SM}(z) + \theta_a M^a_{BSM}(z)|^2 dz \]

probability = wave function, squared

- additivity of the matrix element \( \rightarrow \) incur a simple (polynomial) dependence in \( \theta \) for fixed configuration \( z \)

\[ \frac{d\sigma(x, \theta)}{dx} = \frac{d\sigma_{SM}(x)}{dx} + \sum_a \theta_a \frac{d\sigma^a_{int.}(x)}{dx} + \frac{1}{2} \sum_{a,b} \theta_a \theta_b \frac{d\sigma^{ab}_{BSM}(x)}{dx} \]

- Neyman-Pearson:

\[ q(D) = \frac{L(D|\theta)}{L(D|SM)} \]

where

\[ L(D|\theta) = P_{L\sigma(\theta)}(N) \times \prod_{i=1}^{N} p(x_i|\theta) \]

Optimality can be achieved with cross-section ratio \( R \) or its universal coefficient functions \( R_a, R_{ab} \)

"normalization" "shape"

NB #1 Curse of dimensionality is lifted!!
15 operators \( \rightarrow \) 136 coefficients

NB #2: \( R \) is positive: Fit universal dependence using the most general quadratic polynomial

\[ R(x|\theta, SM) = \frac{d\sigma(x, \theta)/dx}{d\sigma(x, SM)/dx} = 1 + \sum_a \theta_a R_a(x) + \frac{1}{2} \sum_{a,b} \theta_a \theta_b R_{ab}(x) \]

\[ \cong \left(1 + \sum_a \theta_a \hat{n}_a(x)\right)^2 + \sum_a \left(\sum_{b \geq a} \theta_b \hat{n}_{ab}(x)\right)^2 \]
• Measure the top quark – Z boson coupling

• Train separate “SM vs. EFT” classifiers
  • Single operator $O_{tZ}$, $O_{tW}$, $O_3\phi Q$
  • different trainings for different limits (!)
  • “likelihood trick” for SMEFT effects

• signal extraction with 1D, 2D, and 5D LL fit
  • Sampling of parameter space in the training
  • Targeted signals differ kinematically, but no parametrized training is used
  • Signal mix
    • no large linear terms → OK

• Best current limits

- $tt$: 811 pb
- $tW$: 72 pb
- $tZ$: 1 pb
- $tZq$: 0.088 pb

- $t (t$-channel)$: 217 pb
- $t (s$-channel)$: 10 pb

- $ttZ$: 0.088 pb

- Weak vector coupling (L)
- Weak vector coupling (R)
- Weak dipole interactions

- CMS Preliminary
- JHEP 12 (2021) 083
IMPROVING HIGH DIMENSIONAL LIMITS

- **[ML4EFT]** – study ZH and top quark pairs
- Pheno study with parametrized NN classifiers
- Top quark pairs in low ($N_f=2$) and high feature dimension $N_f=18$
  - Pairs of 2D limits with 6 more ops marginalized
  - Binned vs. unbinned: Some gain w/ unbinned when using 2 features
  - High dimensional observation ($N_f=18$) constraining a high-dimensional ($N_{coef}=8$) model using an SM candle
  - Large improvement for $N_f=18$– mostly in the marginalized limits
- Take seriously constraining power from SM candle
- Whether the sensitivity gain survives systematics in an unbinned detector-level analysis is an open question
TOWARDS UNBINNED ANALYSIS

- Binned parametrized classifiers are impractical for high SMEFT parameter dimension
- What’s missing to go all-in? Systematics.
- \( L = \sum_{\theta} \log[N(x)/N(0)] \)
  \[ x \in [3.2 \times 10^{-2}, 9.2 \times 10^{-1}] \]

- \( \nu_s \)
  \[ \text{linear fit} \]
  \[ \text{quadratic fit} \]
  \[ \text{quartic fit} \]

- A challenge: \( \dim(\theta), \dim(\nu) \sim 20 - 50 \), and high event counts in the profiling
- Divide & conquer #1: Experiments begun machine-learning certain nuisances: \( h_{\text{damp}}, b\)-fragmentation

- Divide & conquer #2:
  Unbinned unfolding for high dimensions
- Consider on the conditional pdf \( p(x_{\text{det}} | z_{\text{ptl}}) \)
  which can be evaluated in the forward mode
- Unfolding algorithms use Bayes’ theorem
  \[ p(x_{\text{det}} | z_{\text{ptl}}) p(z_{\text{ptl}}) = p(z_{\text{ptl}} | x_{\text{det}}) p(x_{\text{det}}) \]
  to learn \( p(z_{\text{ptl}} | x_{\text{det}}) \); GAN & other generative versions
  - Mostly iterative, to remove simulated prior

- Report unbinned unfolded data; then SMEFT analysis
REFERENCES

- **Madminer**: Neural networks based likelihood-free inference & related techniques
  - K. Cranmer, J. Pavez, and G. Louppe [1506.02169]
  - J. Brehmer, K. Cranmer, G. Louppe, J. Pavez [1805.00013] [1805.00020] [1805.12244]
  - J. Brehmer, F. Kling, I. Espejo, K. Cranmer [1907.10621]
  - J. Brehmer, S. Dawson, S. Homiller, F. Kling, T. Plehn [1908.06980]
  - A. Butter, T. Plehn, N. Soybelman, J. Brehmer [2109.10414]
  - established many of the *main ideas & statistical interpretation* in various *NN applications*

- **Weight derivative regression** (A. Valassi) [2003.12853]

- **Parametrized classifiers** for SM-EFT: NN with quadratic structure

- **Boosted Information Trees**: Tree algorithms & boosting
  - S. Chatterjee, S. Rohshap, N. Frohner, R. S., D. Schwarz [2107.10859], [2205.12976]
  - ML4EFT R. Ambrosio, J. Hoeve, M. Madigan, J. Rojo, V. Sanz [2211.02058]

- All approaches are “SMEFT-specific ML” with differences mostly on the practical side

→ talk later today

my practical experience