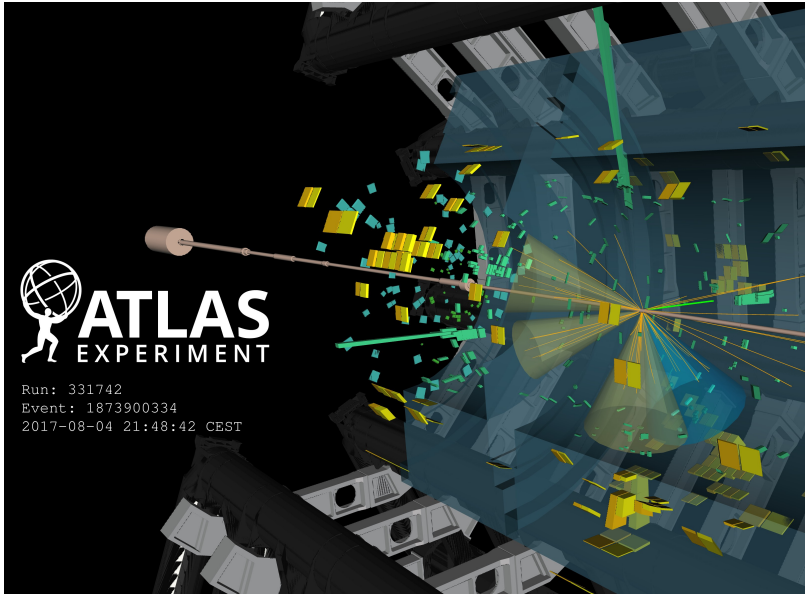
Constraining effective field theories with machine learning

Johann Brehmer[1], Kyle Cranmer[1], Irina Espejo[1], **Alexander Held[1]**,
Felix Kling[2], Gilles Louppe[3], Juan Pavez[4]

[1] New York University
[2] SLAC National Accelerator Laboratory
[3] University of Liège
[3] Federico Santa María Technical University

CHEP 2019, Adelaide
November 7, 2019

The inference challenge

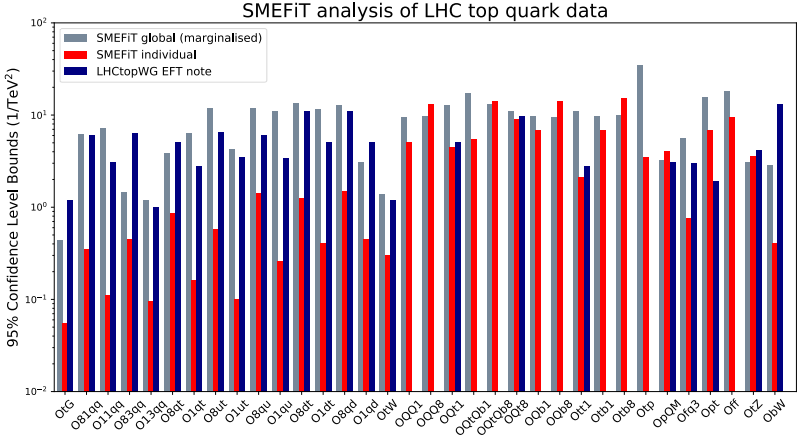


Phys. Lett. B 784 (2018) 173, ATLAS Experiment © 2019 CERN

High-dimensional measurements x
are linked to parameters θ
with **likelihood** $L(\theta) = p(x \mid \theta)$

$$p(x \mid \theta)$$

$$\mathscr{L}_{EFT} = \mathscr{L}_{SM} + \sum_i \frac{\theta_i}{\Lambda^2} \mathcal{O}_i + \ldots$$



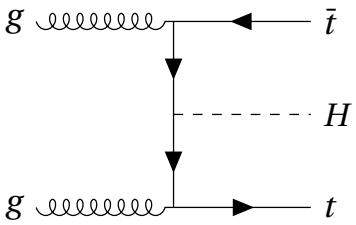
JHEP 1904 (2019) 100

The likelihood (1)

- The full likelihood can be **factorized**:

$$p\left(x\,|\,\theta\right) = \int \mathrm{d}z_P \mathrm{d}z_S \mathrm{d}z_D \; p\left(z_P\,|\,\theta\right) \; p\left(z_S\,|\,z_P\right) \; p\left(z_D\,|\,z_S\right) \; p\left(x\,|\,z_D\right)$$

parton level z_P



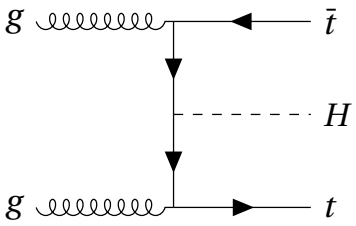
The dependence on parameters θ is here.

The likelihood (2)

- The full likelihood can be **factorized**:

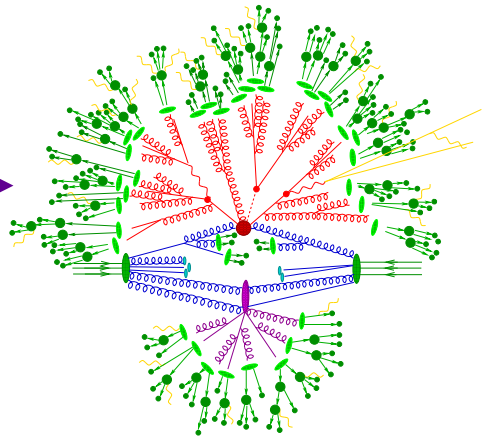
$$p\left(x\,|\,\theta\right) = \int \mathrm{d}z_P \mathrm{d}z_S \mathrm{d}z_D \; p\left(z_P\,|\,\theta\right) \; p\left(z_S\,|\,z_P\right) \; p\left(z_D\,|\,z_S\right) \; p\left(x\,|\,z_D\right)$$

parton level z_P



The dependence on parameters θ is here.

parton shower z_S



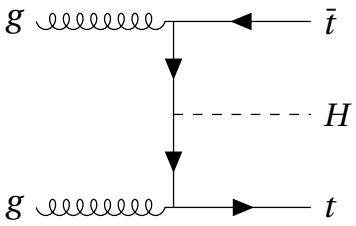
JHEP 0902 (2009) 007

The likelihood (3)

- The full likelihood can be **factorized**:

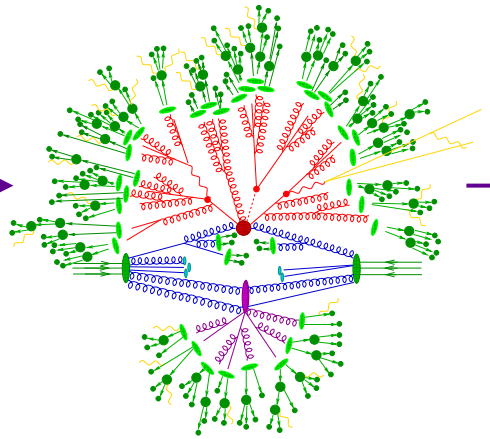
$$p\left(x\,|\,\theta\right) = \int \mathrm{d}z_P \mathrm{d}z_S \mathrm{d}z_D \; p\left(z_P\,|\,\theta\right)\, p\left(z_S\,|\,z_P\right)\, p\left(z_D\,|\,z_S\right)\, p\left(x\,|\,z_D\right)$$

parton level z_P



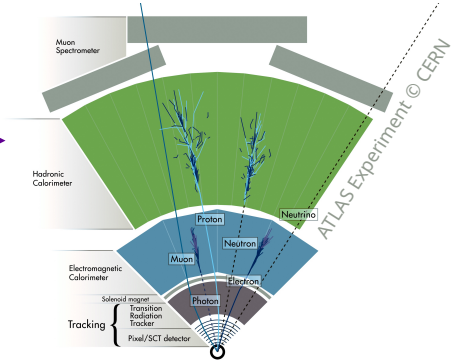
The dependence on parameters θ is here.

parton shower z_S



JHEP 0902 (2009) 007

detector interaction z_D

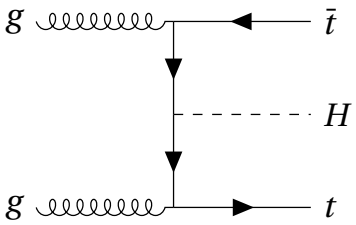


The likelihood (4)

- The full likelihood can be **factorized**:

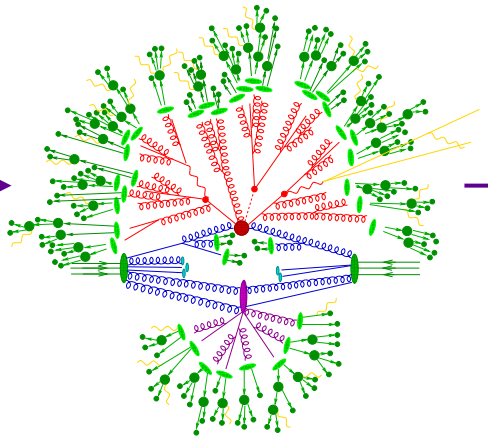
$$p\left(x\,|\,\theta\right) = \int \mathrm{d}z_P \mathrm{d}z_S \mathrm{d}z_D \; p\left(z_P\,|\,\theta\right) \, p\left(z_S\,|\,z_P\right) \, p\left(z_D\,|\,z_S\right) \, p\left(x\,|\,z_D\right)$$

parton level z_P



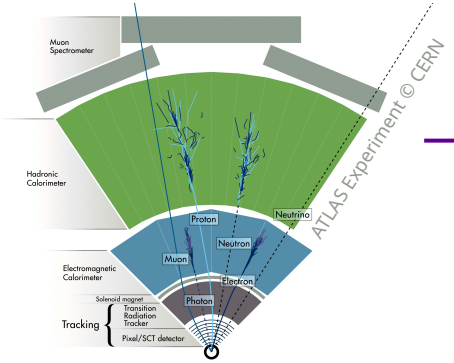
The dependence on parameters θ is here.

parton shower z_S

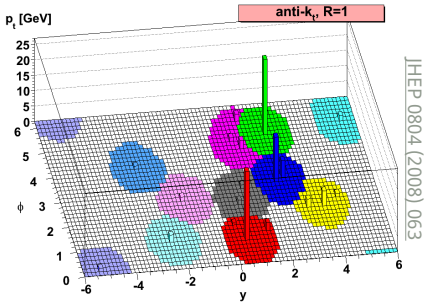


JHEP 0902 (2009) 007

detector interaction z_D



object reconstruction x



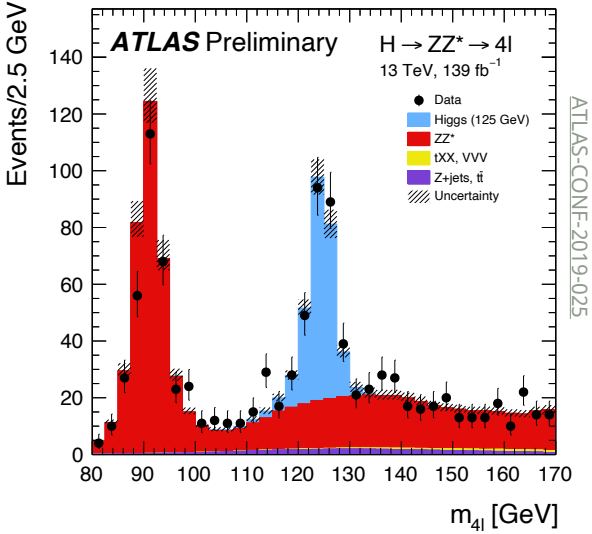
JHEP 0804 (2008) 063

Methods in practice

- Generally **not possible to use the full likelihood** function $L(\theta) = p(x|\theta)$

- **Alternative methods** are used, but have **drawbacks:**
  - Summary statistics are commonly used, at the cost of information loss
  - Matrix element method approximates part of the integral

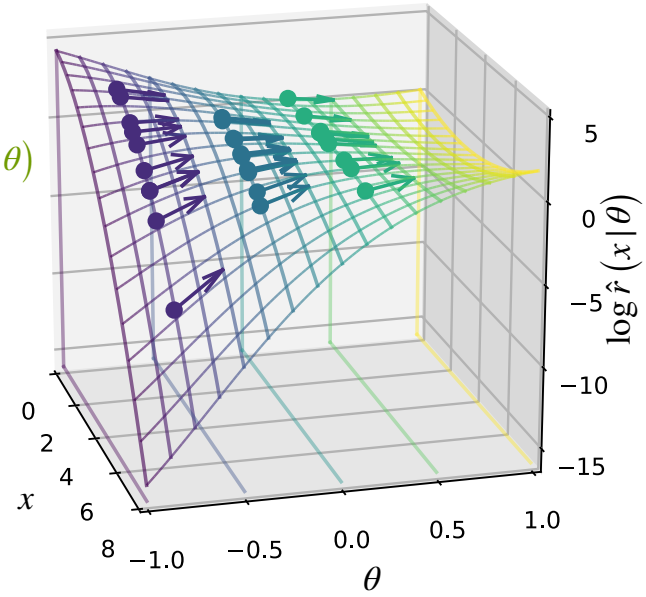
Summary statistic: $p(x'|\theta)$



New opportunities

- **In particle physics, more can be done**
  - Joint likelihood ratio can be calculated: $r\left(x, z \mid \theta, \theta_0\right) = \dfrac{p\left(z \mid \theta\right)}{p\left(z \mid \theta_0\right)}$
    - Depends on reconstructed objects x and parton-level kinematics z
  - Also possible to take derivatives wrt. parameters: $t\left(x, z \mid \theta\right) = \nabla_\theta \log p\left(x, z \mid \theta\right)$
- A **neural network** can **learn** an estimator for the **likelihood ratio** $\hat{r}\left(x \mid \theta\right)$
- **"Mining Gold":**
  - Extract more information from simulator: x, $r\left(x, z \mid \theta\right)$, $t\left(x, z \mid \theta\right)$

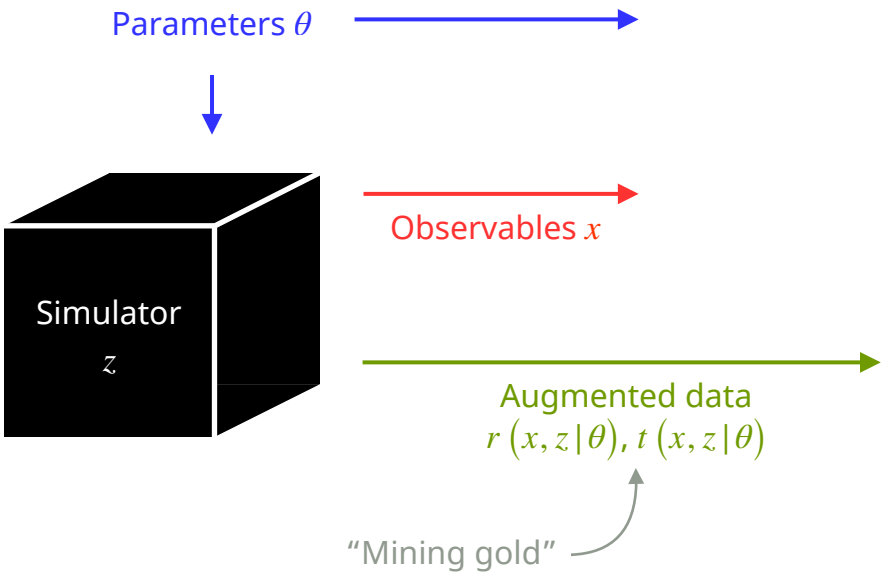
Learning $\hat{r}$ from samples (x, z, θ)



More details: Phys.Rev. D98 (2018) no.5, 052004

General approach (1)

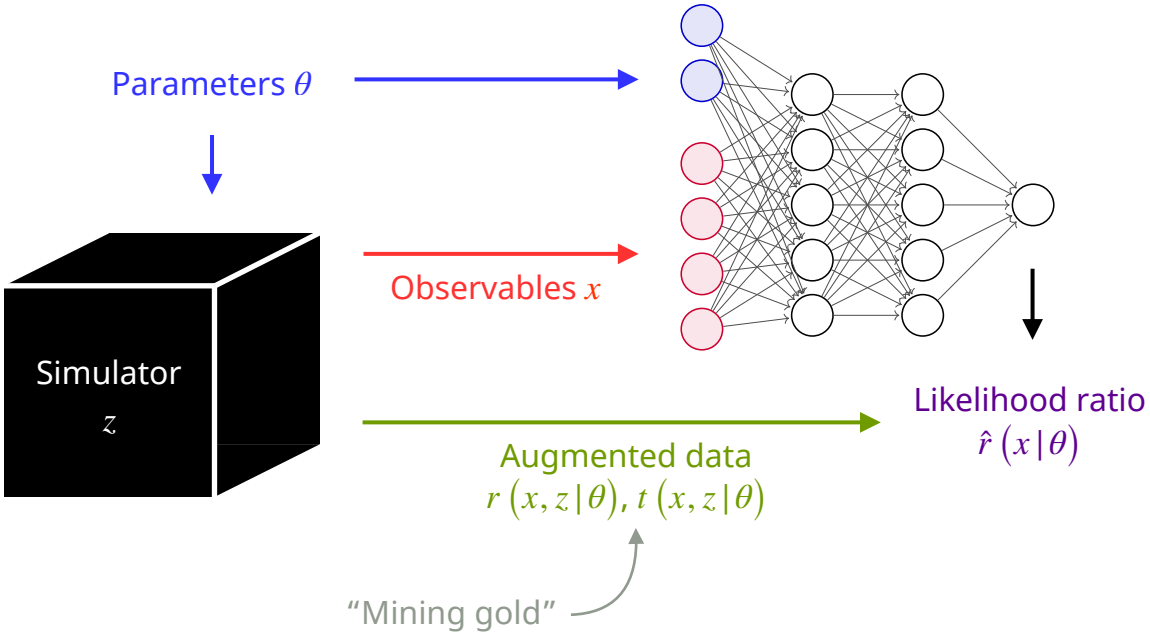
1) Simulate events (x, z, θ)



General approach (2)

1) Simulate events (x, z, θ)

2) Learn the likelihood ratio

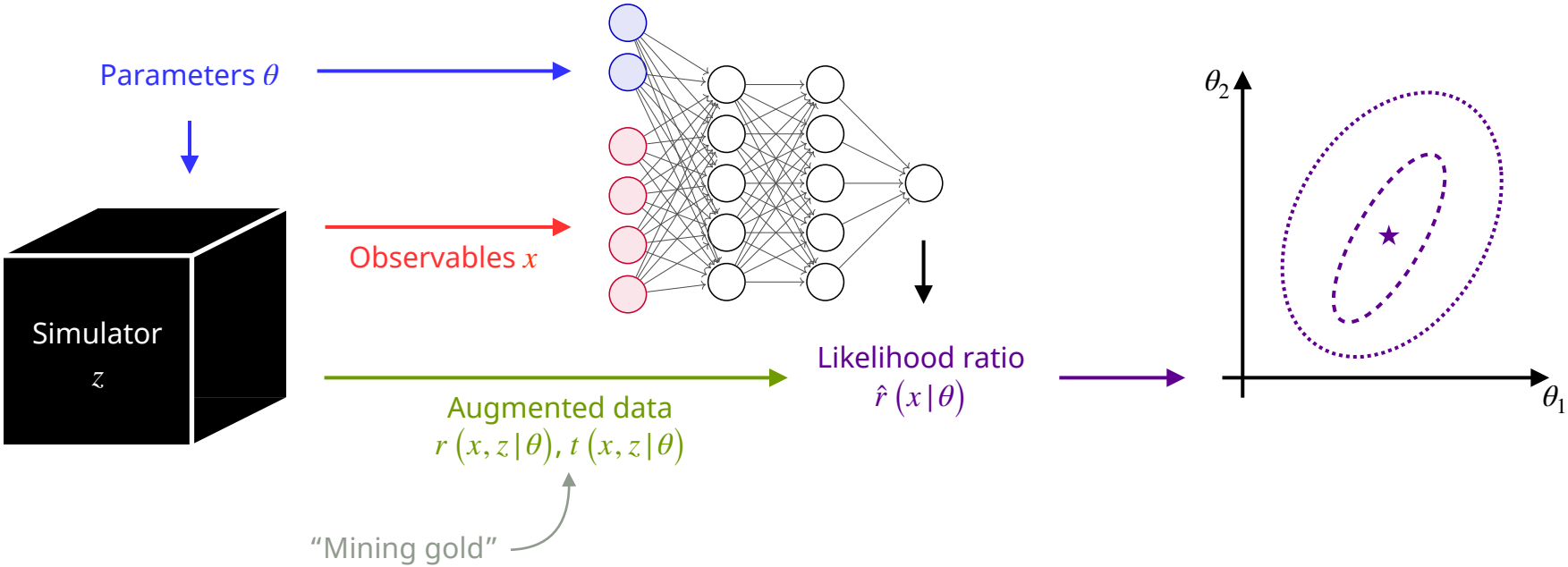


General approach (3)

1) Simulate events (x, z, θ)

2) Learn the likelihood ratio

3) Do inference



The MadMiner package

- **MadMiner** implements this workflow in a ready-to-use python package
  - Standalone solution for phenomenological analyses
  - Modular structure, can replace elements for LHC-style analyses etc.
  - Code is on github: diana-hep/madminer

README.md

MadMiner: Machine learning–based inference for particle physics

By Johann Brehmer, Felix Kling, Irina Espejo, and Kyle Cranmer

pypi package 0.6.0 | build passing | docs passing | chat on gitter | docker pulls 118 | code style black | License MIT | DOI 10.5281/zenodo.1489147

- **Get started!**
  - `pip install madminer`
  - Tutorials: notebooks with example implementations
  - Documentation: readthedocs, arXiv:1907.10621

jupyter notebooks with examples

- 1_setup.ipynb
- 2a_parton_level_analysis.ipynb
- 2b_delphes_level_analysis.ipynb
- 3a_likelihood_ratio.ipynb
- 3b_score.ipynb
- 3c_likelihood.ipynb
- 4a_limits.ipynb
- 4b_fisher_information.ipynb
- A1_systematic_uncertainties.ipynb
- A2_ensemble_methods.ipynb

- **Future plans:**
  - Integrate within frameworks for LHC experiment use

MadMiner
latest
Search docs
SITES
Introduction to MadMiner
Getting started

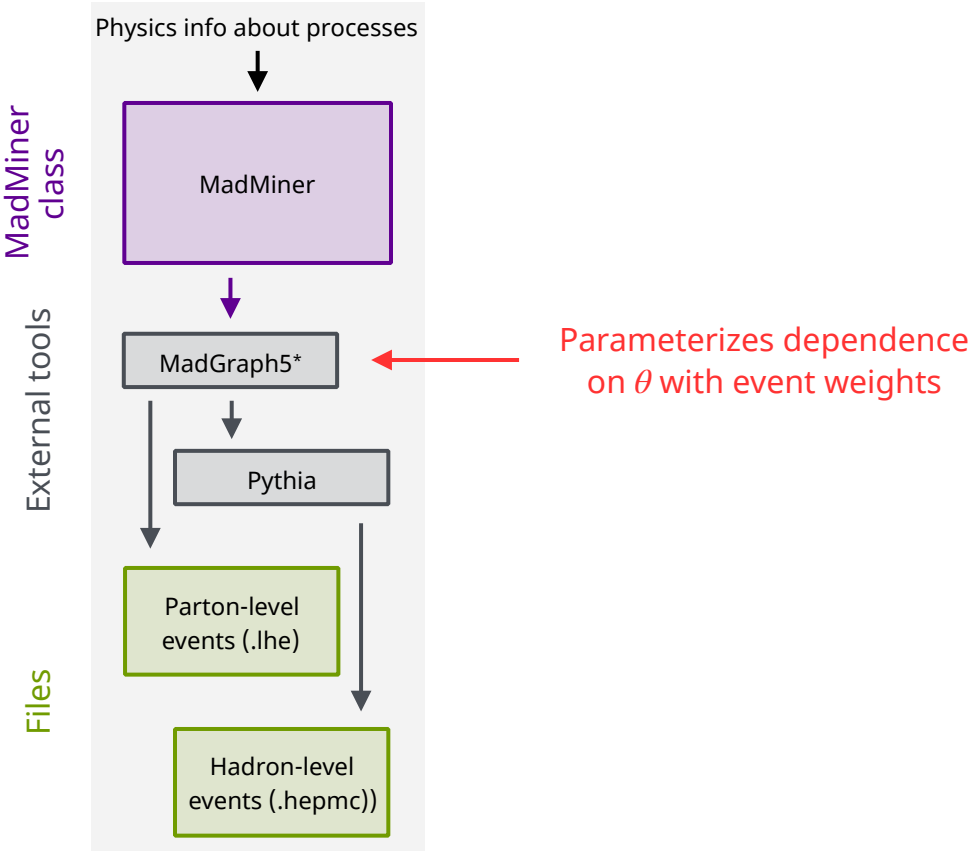
Docs » Getting started

Getting started

Simulator dependencies

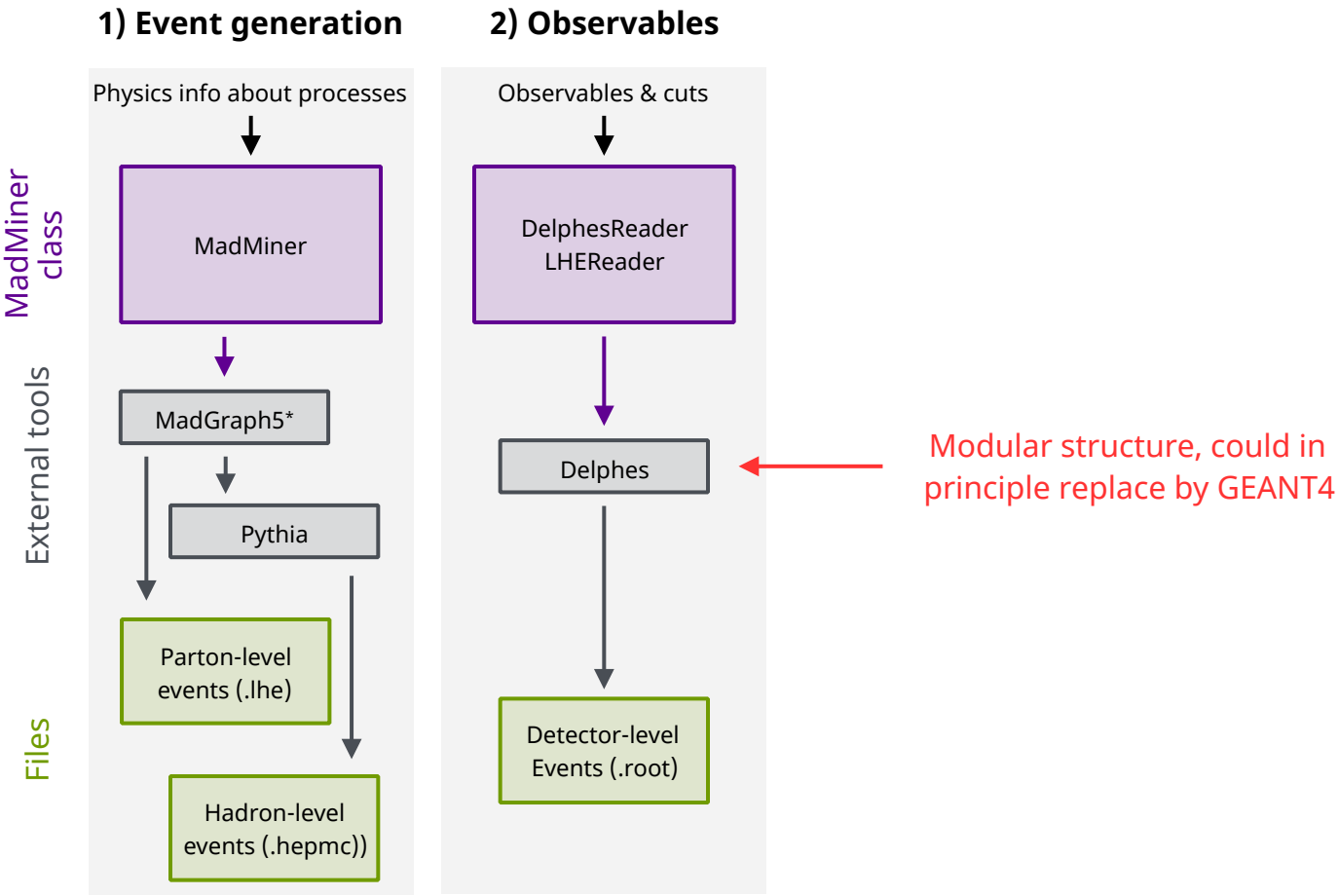
MadMiner structure (1)

1) Event generation



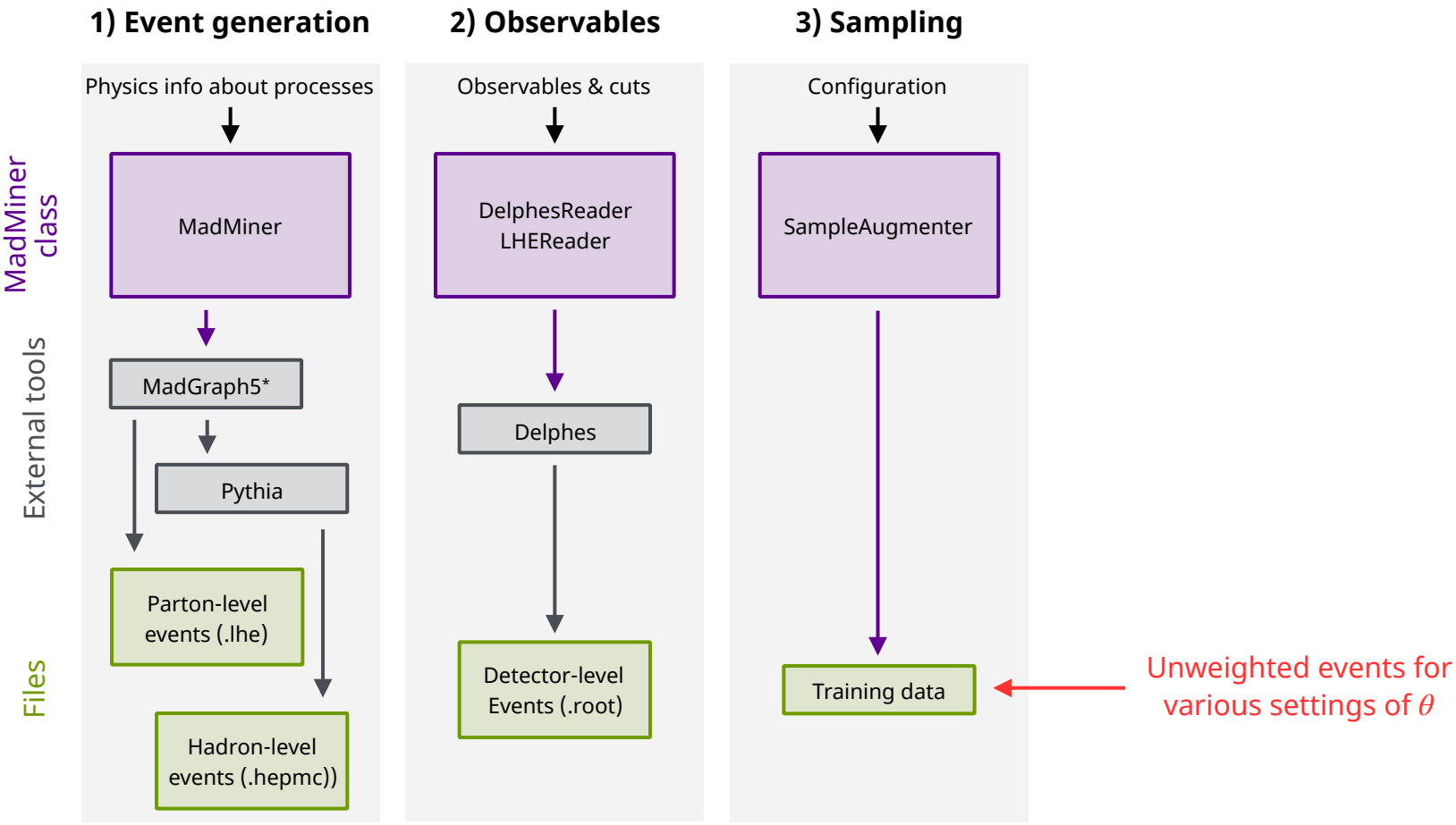
* formally MadGraph5_aMC@NLO

MadMiner structure (2)



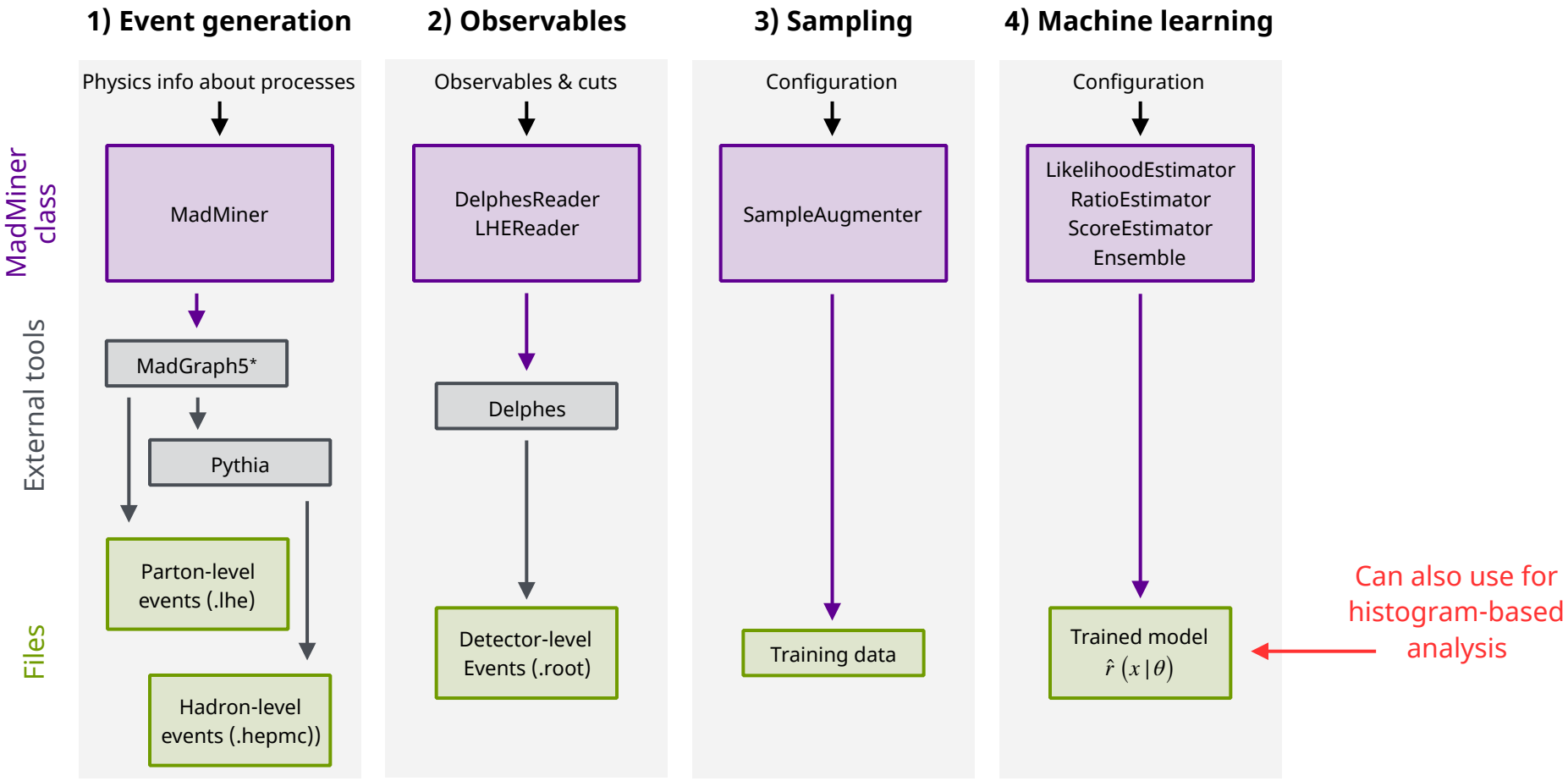
* formally MadGraph5_aMC@NLO

MadMiner structure (3)



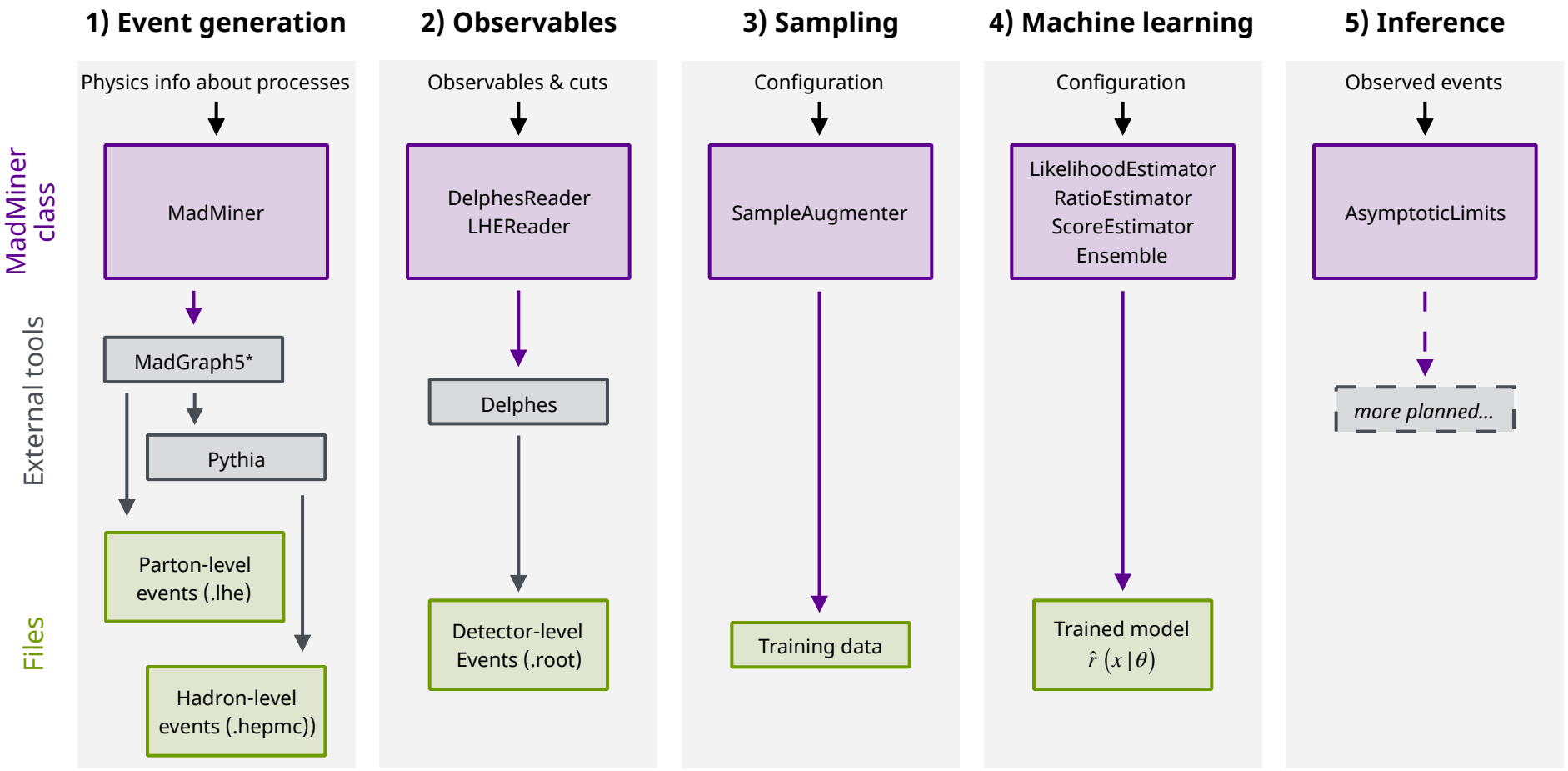
* formally MadGraph5_aMC@NLO

MadMiner structure (4)



* formally MadGraph5_aMC@NLO

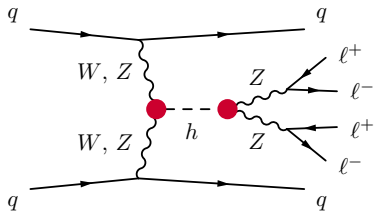
MadMiner structure (5)



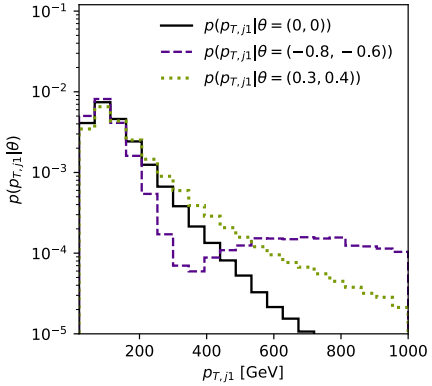
* formally MadGraph5_aMC@NLO

Example application: VBF (1)

- **Example: VBF production**
  - Sensitive to two coefficients f_W, f_{WW}



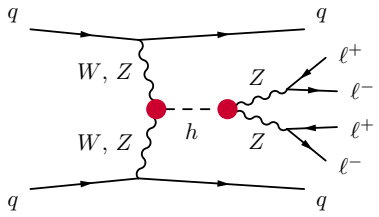
  - Varying coefficients:



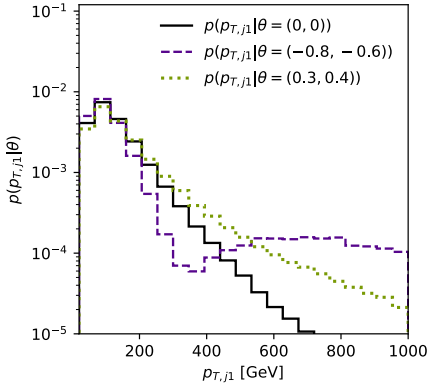
More details: Phys.Rev.Lett. 121 (2018) no.11, 111801, Phys.Rev. D98 (2018) no.5, 052004

Example application: VBF (2)

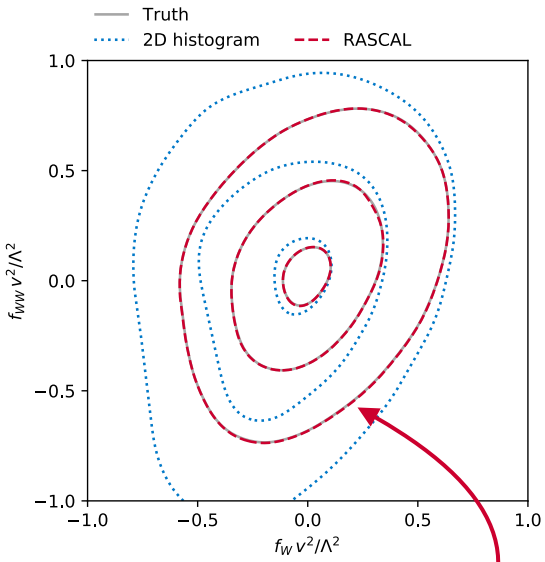
- **Example: VBF production**
  - Sensitive to two coefficients f_W, f_{WW}



  - Varying coefficients:



Learning likelihood ratio as function of 42 features:

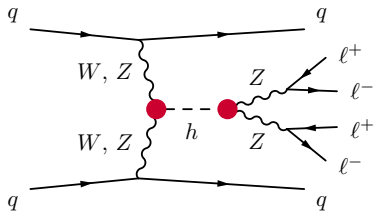


Optimal limits derived with new technique

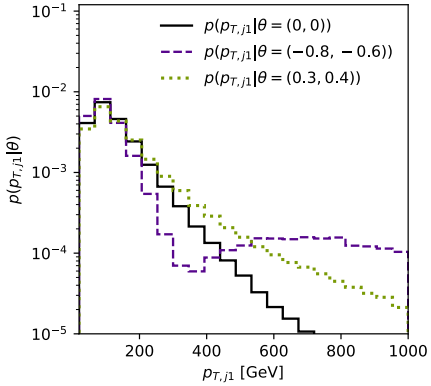
More details: Phys.Rev.Lett. 121 (2018) no.11, 111801, Phys.Rev. D98 (2018) no.5, 052004

Example application: VBF (3)

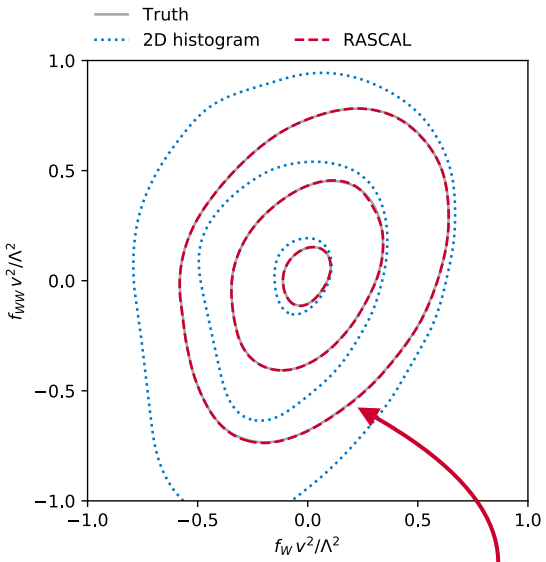
- **Example: VBF production**
  - Sensitive to two coefficients f_W, f_{WW}



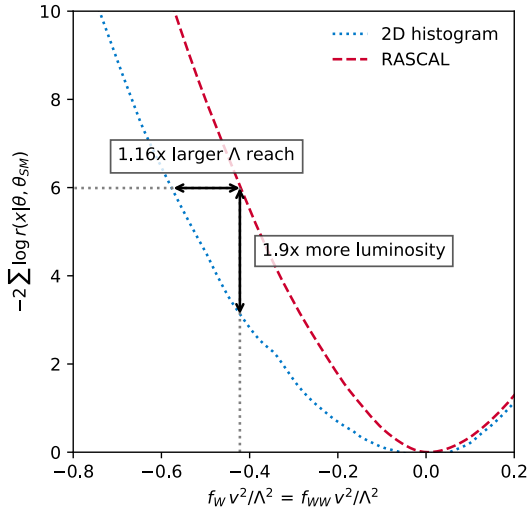
  - Varying coefficients:



Learning likelihood ratio as function of 42 features:



Optimal limits derived with new technique



Extending physics reach

More details: Phys.Rev.Lett. 121 (2018) no.11, 111801, Phys.Rev. D98 (2018) no.5, 052004

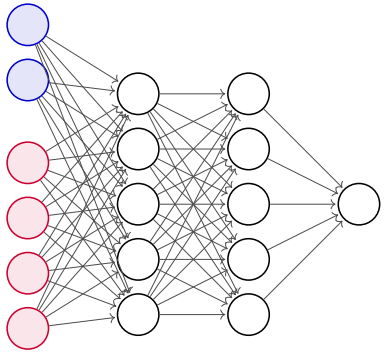
Systematic uncertainties

- **Example: Wilson coefficients with ttH**
  - Using $t\bar{t}H(\gamma\gamma)$ process to measure three parameters: c_{uG}, c_u, c_G
  - Systematic uncertainties used: PDF and scale variations (μ_R, μ_F)

- Neural network learns dependence of $\hat{r}\left(x\,|\,\theta\right)$ on systematic variations

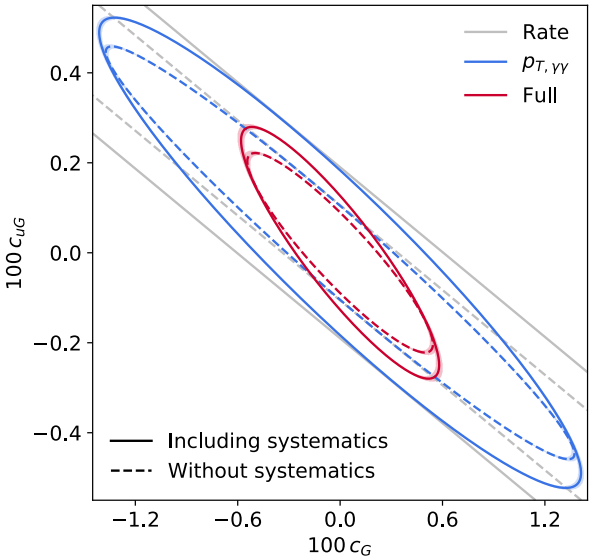
Parameters of interest & nuisance parameters: c_{uG}, μ_R, ...

48 observables: $p_{T,\gamma\gamma}$, η_γ, E_T^{miss}, ...



$\hat{r}\left(x\,|\,\theta\right)$

Expected 68% CL limits

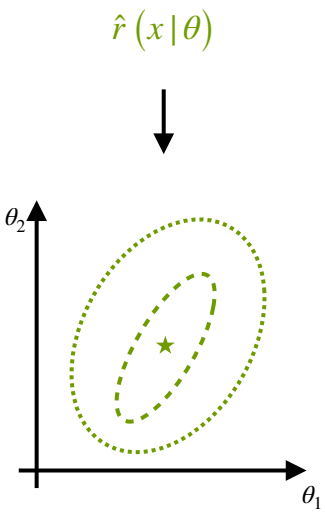


More details: arXiv:1907.10621, parameterized ML: arXiV:1601.07913

Summary

- **Inference with traditional methods does not scale well** to measuring many parameters at once

- A **new family of methods** was developed to address this challenge
  - Combining **machine learning** techniques with matrix element information
  - Provides likelihood ratio $\hat{r}\left(x \mid \theta\right)$ for inference
  - Can define locally optimal observables
  - **Promising performance**, especially for multi-parameter measurements (EFT etc.)
  - Methods applicable beyond HEP (strong lensing example: arXiv:1909.02005)



- **MadMiner** is a python package automating the workflow with these new techniques
  - Implements full chain for phenomenological study
  - Future plans: integrate within frameworks for LHC experiment use
  - `pip install madminer`

Backup

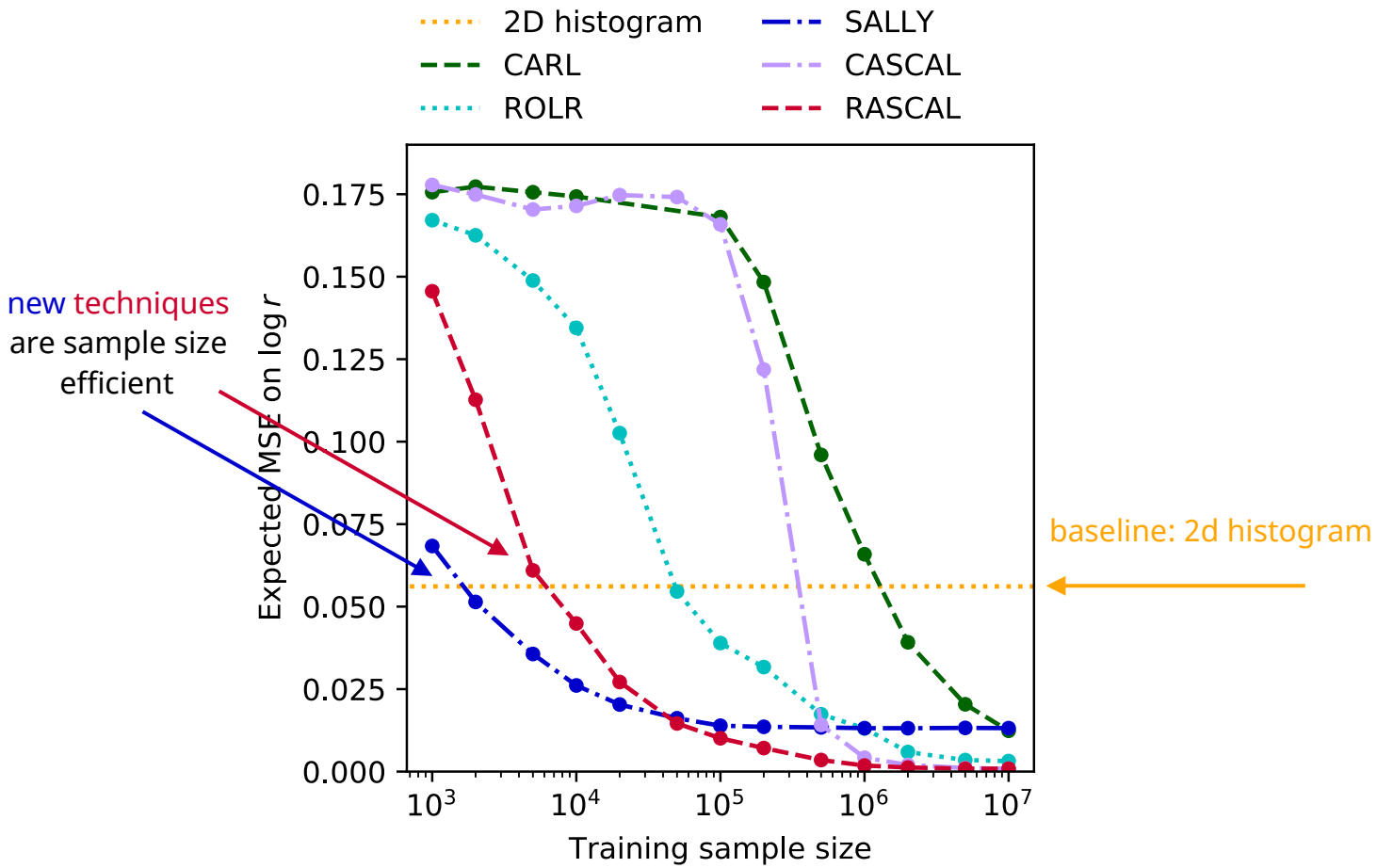
References

- **Techniques:**
  - Johann Brehmer, Kyle Cranmer, Gilles Louppe, Juan Pavez: *Constraining Effective Field Theories with Machine Learning*
    - Phys.Rev.Lett. 121 (2018) no.11, 111801, arXiv:1805.00013
  - Johann Brehmer, Kyle Cranmer, Gilles Louppe, Juan Pavez: *A Guide to Constraining Effective Field Theories with Machine Learning*
    - Phys.Rev. D98 (2018) no.5, 052004, arXiv:1805.00020
  - Johann Brehmer, Gilles Louppe, Juan Pavez, Kyle Cranmer: *Mining gold from implicit models to improve likelihood-free inference*
    - arXiv:1805.12244
  - Markus Stoye, Johann Brehmer, Gilles Louppe, Juan Pavez, Kyle Cranmer: *Likelihood-free inference with an improved cross-entropy estimator*
    - arXiv:1808.00973
  - Johann Brehmer, Felix Kling, Irina Espejo, Kyle Cranmer: *MadMiner: Machine learning-based inference for particle physics*
    - arXiv:1907.10621

- **Applications** (beyond what is already mentioned above)**:**
  - Johann Brehmer, Sally Dawson, Samuel Homiller, Felix Kling, Tilman Plehn: *Benchmarking simplified template cross sections in WH production*
    - arXiv:1908.06980
  - Johann Brehmer, Siddharth Mishra-Sharma, Joeri Hermans, Gilles Louppe, Kyle Cranmer: *Mining for Dark Matter Substructure: Inferring subhalo population properties from strong lenses with machine learning*
    - arXiv:1909.02005

- **MadMiner:** https://github.com/diana-hep/madminer , DOI: 10.5281/zenodo.2574893

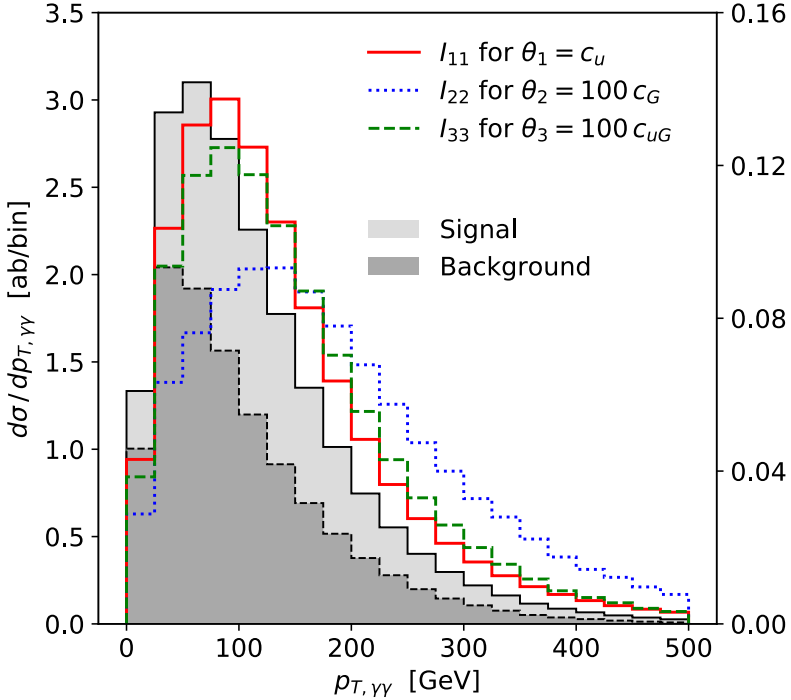
Scaling with training sample size



More details: Phys.Rev. D98 (2018) no.5, 052004

ttH example

- 1.5M signal events, 1M background, 10M unweighted events for training
  - Including background contributions from $t\bar{t} + \gamma\gamma$
- Three-layer fully-connected NN with 100 nodes per layer



More details: arXiv:1907.10621