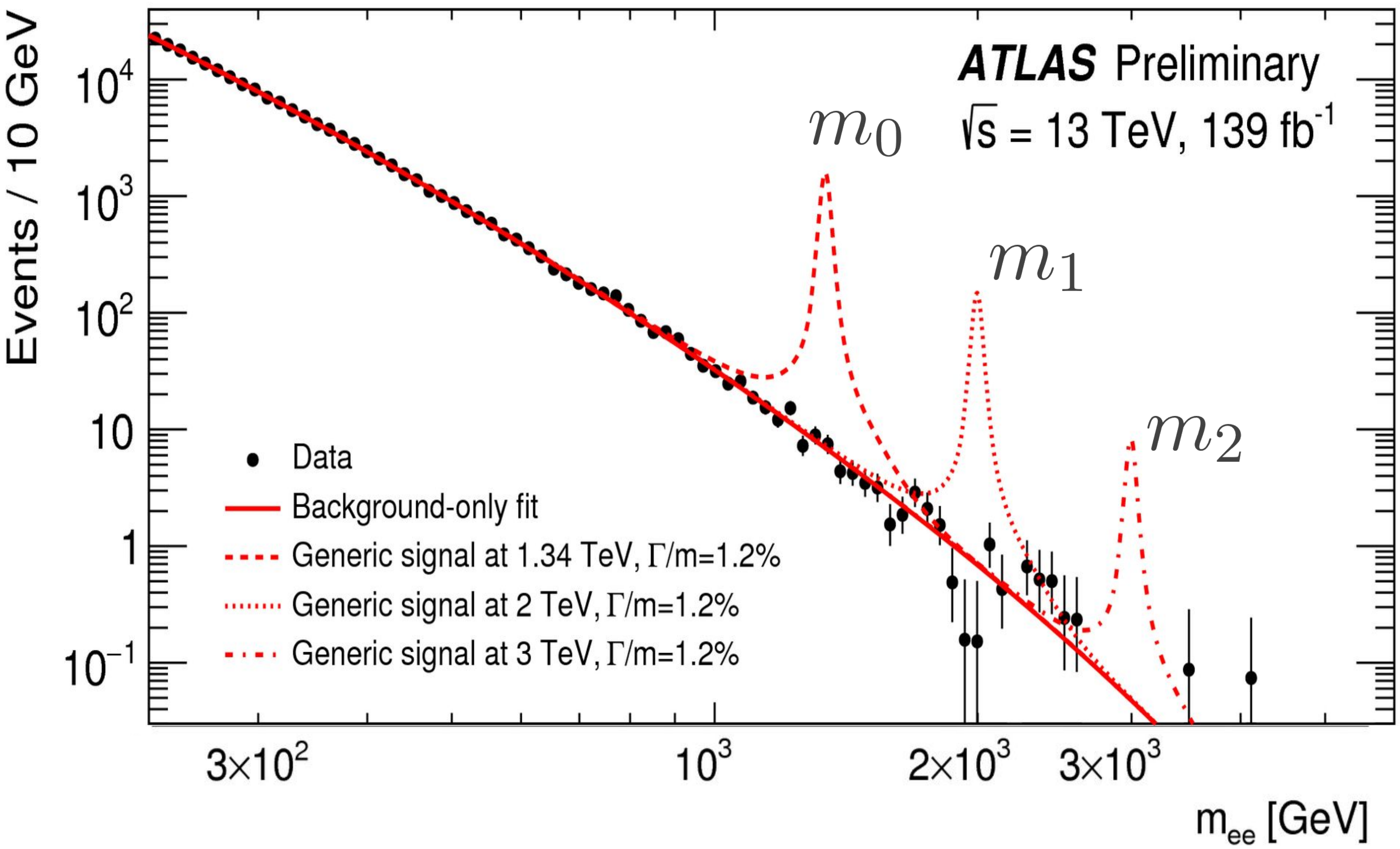
New statistical analysis techniques in RooFit

Robin Syring

Supervisor: Jonas Rembser, Lorenzo Moneta

23/09/2024



ATLAS Preliminary
$\sqrt{s}$ = 13 TeV, 139 fb^{-1}
m_0
m_1
m_2
Data
Background-only fit
Generic signal at 1.34 TeV, Γ/m=1.2%
Generic signal at 2 TeV, Γ/m=1.2%
Generic signal at 3 TeV, Γ/m=1.2%
Events / 10 GeV
10^4
10^3
10^2
10
1
10^{-1}
3×10^2
10^3
2×10^3
3×10^3
m_{ee} [GeV]

Likelihood vs Probability

- hypothesis mass m_0
- measured data with mass m_i

$$\mathcal{L}(m_0) = \prod_i p(m_i|m_0)$$

- Likelihood $\mathcal{L}$
- probability density function p

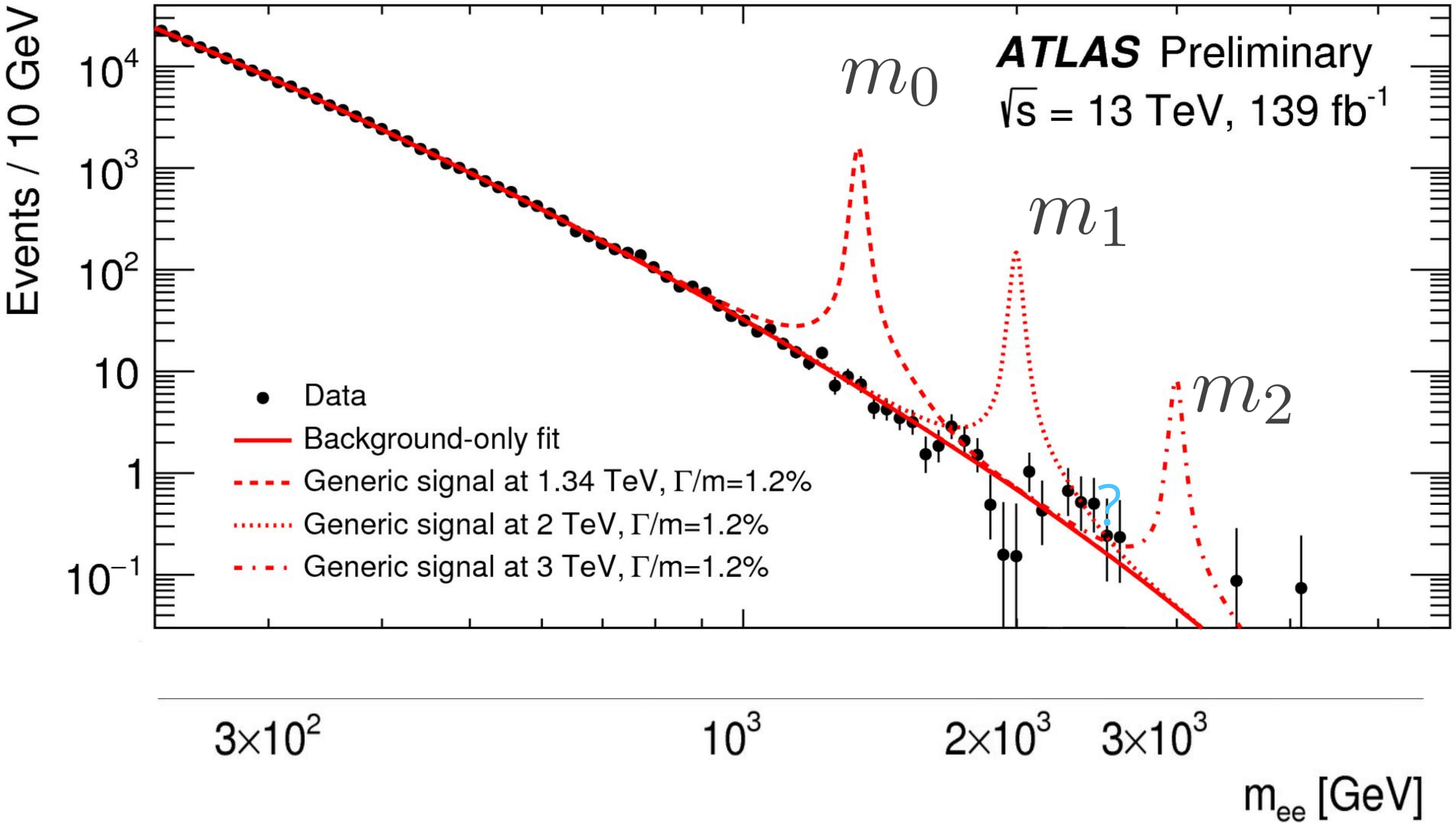
Likelihood Function in Practice

$$\mathcal{L}_{ratio}(m_0) = \prod_i \frac{p(m_i|m_0)}{p_{ref}(m_i)}$$

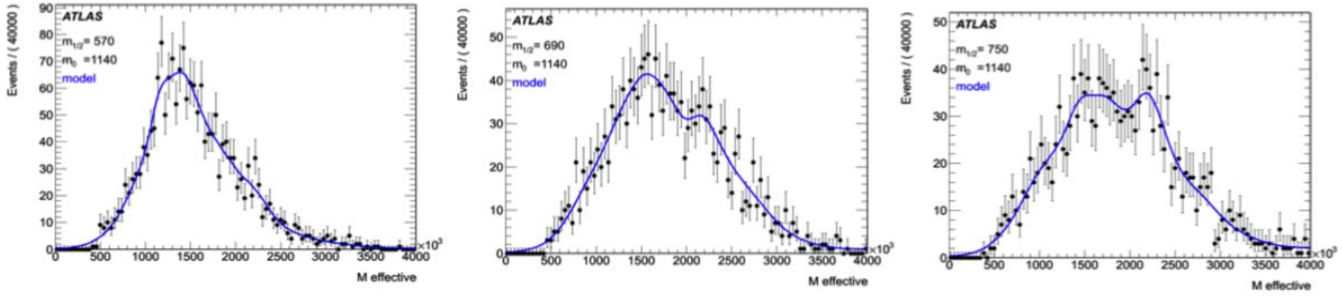
$$p(m_i|m_0) = \int dz_d \int dz_s \int dz_p p(m_i, z_d, z_s, z_p|m_0)$$

- Integrals cannot evaluated directly
  - Detector simulations take time

Morphing = Way of interpolating between different template Histograms


ATLAS Preliminary
√s = 13 TeV, 139 fb-1
m_0
m_1
m_2
?
Events / 10 GeV
Data
Background-only fit
Generic signal at 1.34 TeV, Γ/m=1.2%
Generic signal at 2 TeV, Γ/m=1.2%
Generic signal at 3 TeV, Γ/m=1.2%
3×10²
10³
2×10³
3×10³
m_ee [GeV]

Morphing



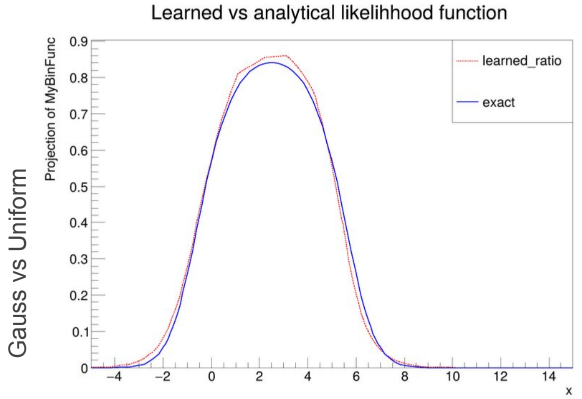
- based on a linear combination of input templates (Moment Morphing)
- Morphing was already implemented in RooFit, but no documentation ⇨ Lots of forum questions
- Tutorial

SBI = Use of ML to learn the likelihood function

Likelihood trick

- train parameterized classifier to
  - discriminate between samples $m_i \sim p(m_i|m_0)$ $m_i \sim p_{ref}(m_i)$
  - transform output of classifier $\hat{s}(m_i|m_0)$ to estimator of likelihood ratio function by

$$\mathcal{L}_{ratio}(m_0) = \prod_i \frac{1-\hat{s}(m_i|m_0)}{\hat{s}(m_i|m_0)}$$



- [SBI]
- [Tutorial]

Power of this method

- compared to morphing, really easy to implement (don’t care of binnings, …)
- higher accuracy due to no binning effects
- reduction of necessary data samples i.e. detector simulations, especially in higher dimensions
- “trivial” sampling of MC Data (cont. sampling possible, less interpolation error)
- but: beware of overtraining
- no sophisticated method to deal with uncertainties

- in some cases, the pdf is a sum of different pdfs $p(m|\mu) = \sum_c w_c(\mu) p_c(m|\mu)$

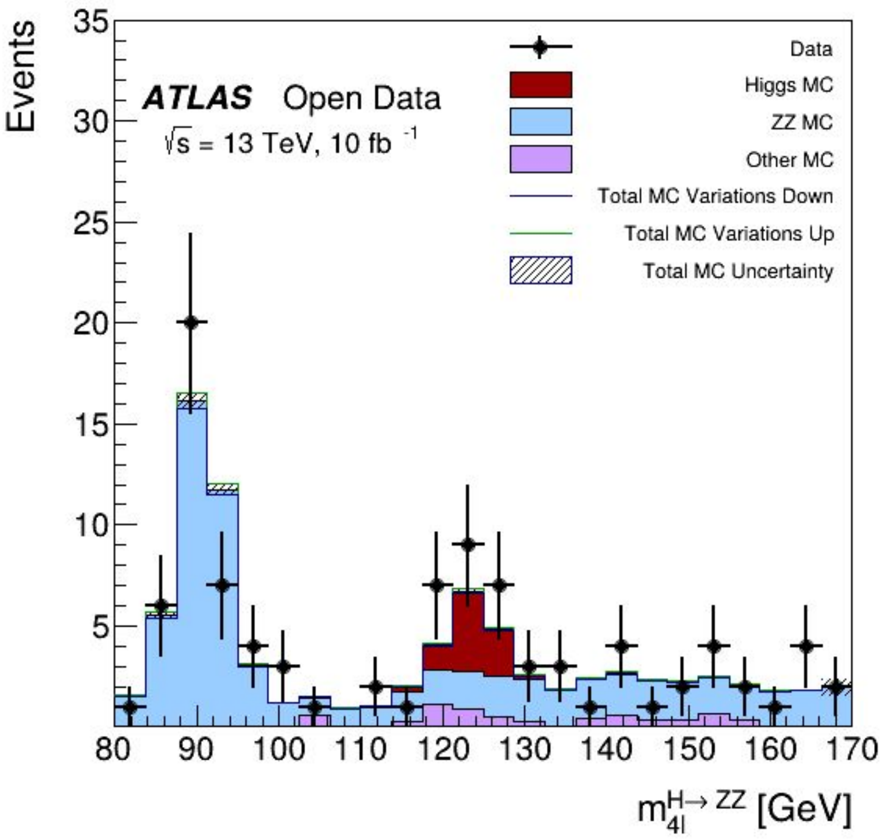
$$
\begin{aligned}
\frac{p(m|\mu=0)}{p(m|\mu)} &= \frac{\sum_c w_c(0) p_c(m|0)}{\sum_{c'} w_{c'}(\mu) p_{c'}(m|\mu)} \\
&= \sum_c \left[\sum_{c'} \frac{w_{c'}(\mu)}{w_c(0)} \frac{p_{c'}}{p_c} \right]^{-1} \\
&= \sum_c \left[\sum_{c'} \frac{w_{c'}(\mu)}{w_c(0)} \frac{p_{c'}(\hat{s}_{c,c'})}{p_c(\hat{s}_{c,c'})} \right]^{-1}
\end{aligned}
$$

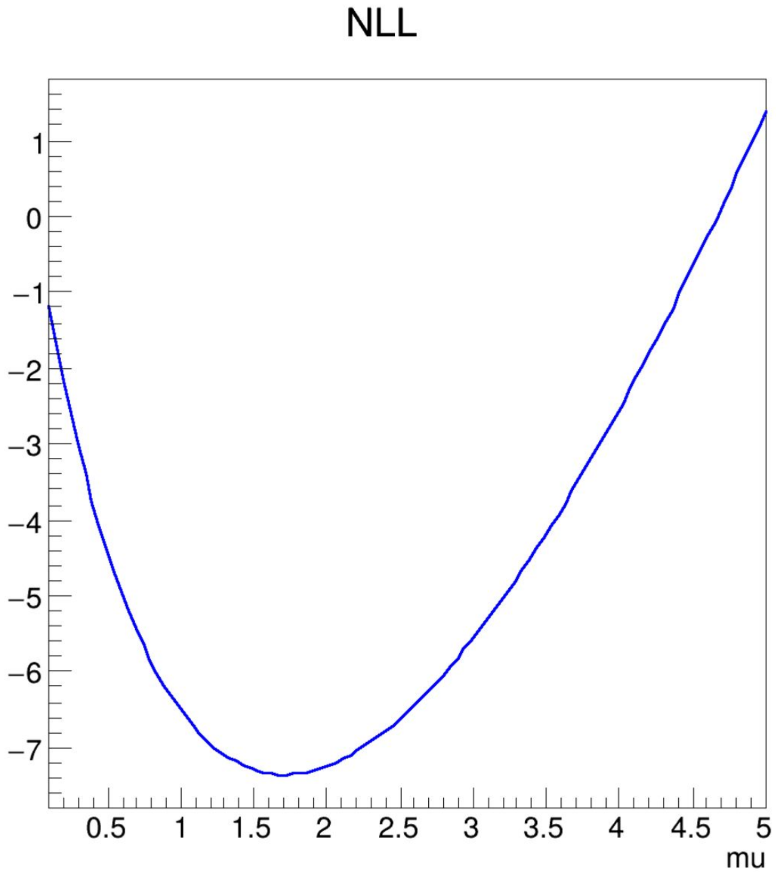
μ = signal strength

$c, c' \in \{zz, higgs\}$

- Allows the classifier to focus on simpler subproblems (unparameterized classifier) (Mixture models)
- Tutorial

Mixture Models





Conclusion

Conclusion

- First RooFit tutorial based on open data, directly connecting RDF tutorial
- Generalized RooFit interface for wrapping python functions to arbitrary number of inputs
- Total: rf615, rf616, rf617, rf618

Next Steps

Open tasks

- Integration of the mixture model formular as a RooFit object
- improve support inside RooFit for external likelihoods (correct plotting, …)
- Inverse functionality: pythonic wrapper to use RooFit functions in other scientific python code
- Prepare CHEP conference presentation

Thank you!

In particular Jonas for your guidance and patience

syring.robin@gmx.de