using pyhf at Belle II

invisible Z' search + tau LFU measurement

Paul Feichtinger

TECHNISCHE UNIVERSITÄT WIEN Vienna | Austria

HEPHY INSTITUTE OF HIGH ENERGY PHYSICS

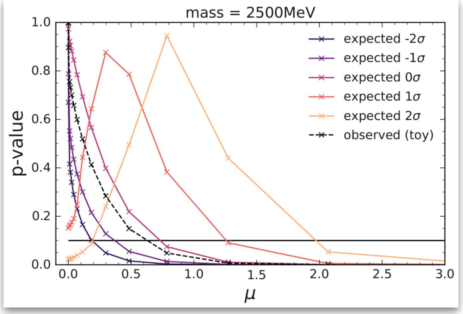
European Research Council
Established by the European Commission

Belle II pyhf workshop

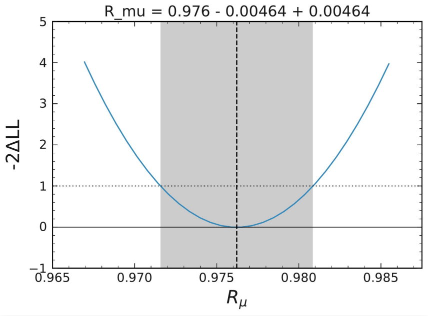
Bonn, 3rd March 2022

Introduction

I used pyhf in two different analysis for

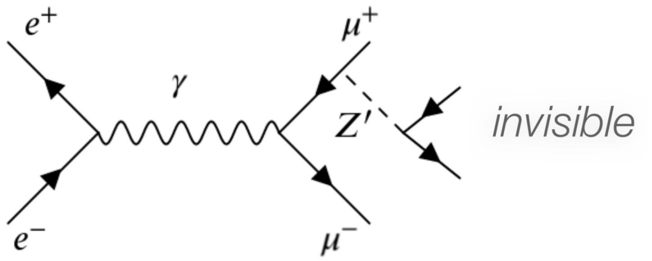


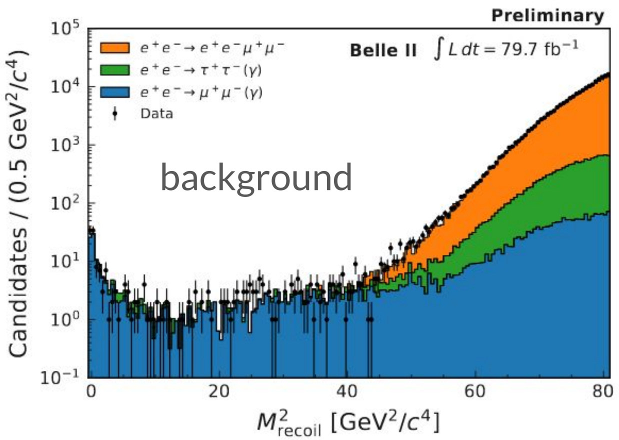
- search for an invisibly decaying Z’ (BSM)
  - hypothesis testing

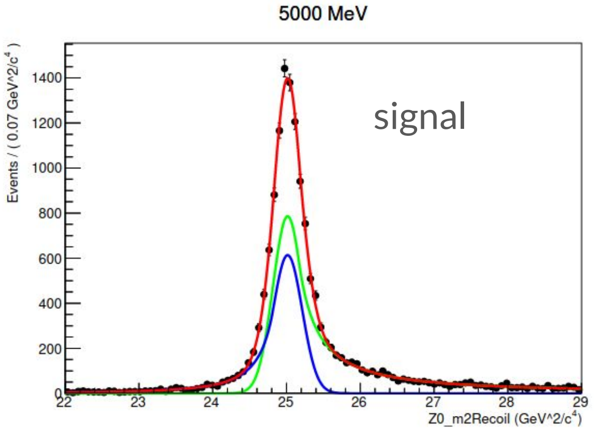


- test LFU in leptonic τ decays (SM)
  - precision measurement

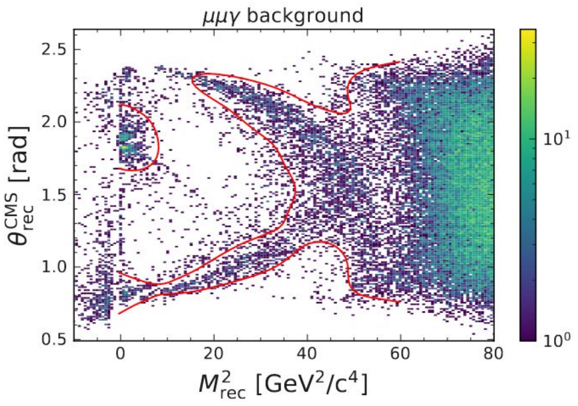
Search for an invisibly decaying Z’

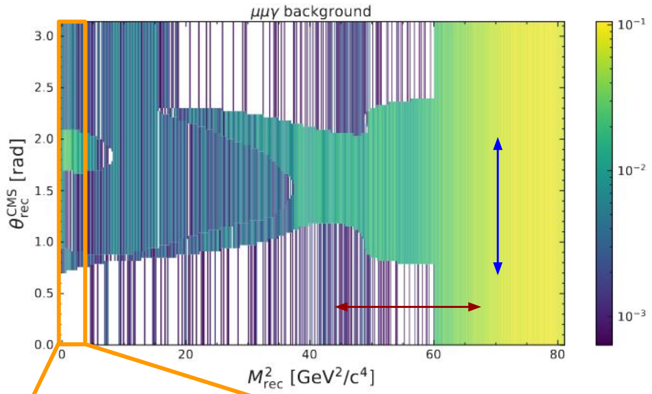






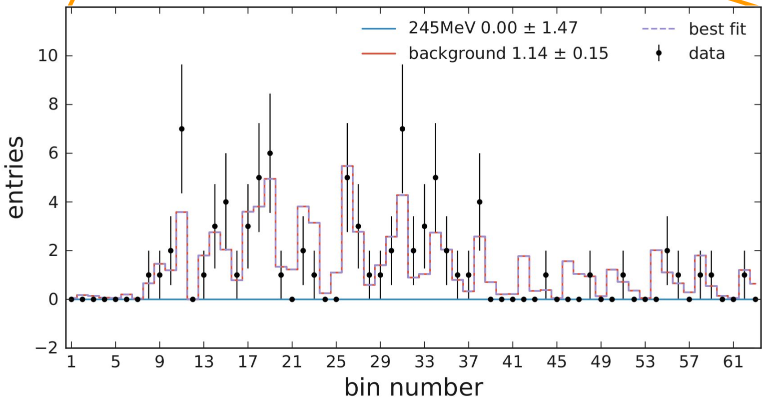
- search is bump hunt in recoil mass distribution
- perform a profile likelihood ratio test with pyhf
- challenges
  - many mass hypotheses to test (~700)
  - low background environment





we derive a 2D template based on the background distribution and the signal resolution

flattened bins for a 245MeV Z’

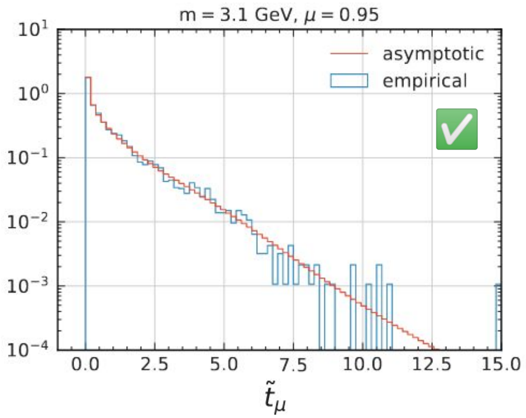


- many zero background bins in 2D template
  - raised error since Pois(n|λ=0) was not defined
  - removed 0 entry bins from the fit
  - should work since pyhf v0.7.0

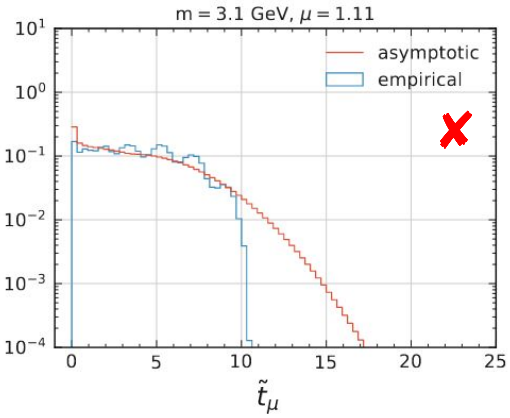
- asymptotic formulae for distribution of test statistic only valid in the large sample limit
- easy to sample from the model to get empirical distribution of test statistic:

  `model.make_pdf(pars).sample()`
- background-like distributions were not accurately described by the asymptotics
  - used toys to derive expected limits

signal-like distribution (true POI = test POI)

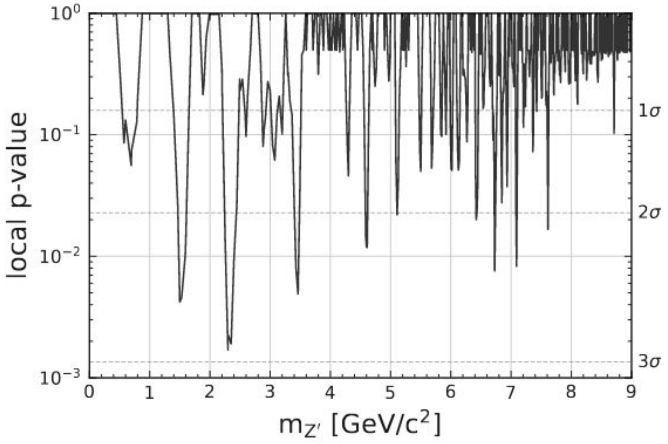


background-like distribution (true POI = 0)

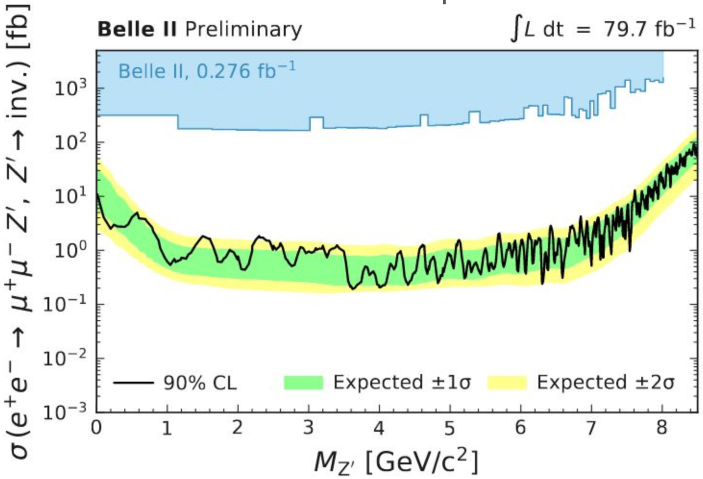


- compute expected + observed confidence intervals on cross section, for each mass hypothesis
- found that jax backend + scipy optimizer was very fast and robust

signal significance



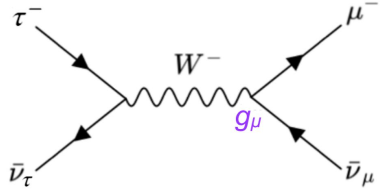
exclusion plot

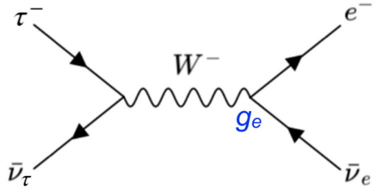


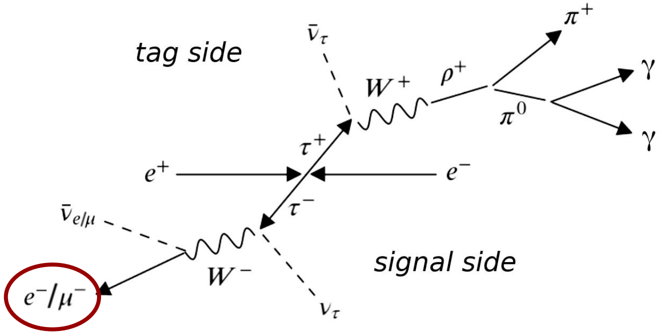
[arXiv:2212.03066](https://arxiv.org/abs/2212.03066)

Testing LFU in leptonic τ decays

- want to measure the ratio $$R_\mu = \frac{\mathcal{B}(\tau^- \to \nu_\tau \mu^- \overline{\nu}_\mu(\gamma))}{\mathcal{B}(\tau^- \to \nu_\tau e^- \overline{\nu}_e(\gamma))}$$
- close to one in SM if weak couplings g_μ and g_e are equal







Why do a template fit?

- ➢ we can achieve very high signal purity (~99%), so why do a template fit?
    - previous experiments only divided the background subtracted yields

1. electron / muon identification is leading systematic
    - lepton ID uncertainties heavily depend on lepton kinematics
    - by performing a fit in bins of lepton momentum we can reduce impact of regions with large uncertainty, without having to remove them
2. reduce uncertainty by providing as much information as possible
3. systematic uncertainties can be included with nuisance parameters directly in the fit
4. can publish full model with pyhf

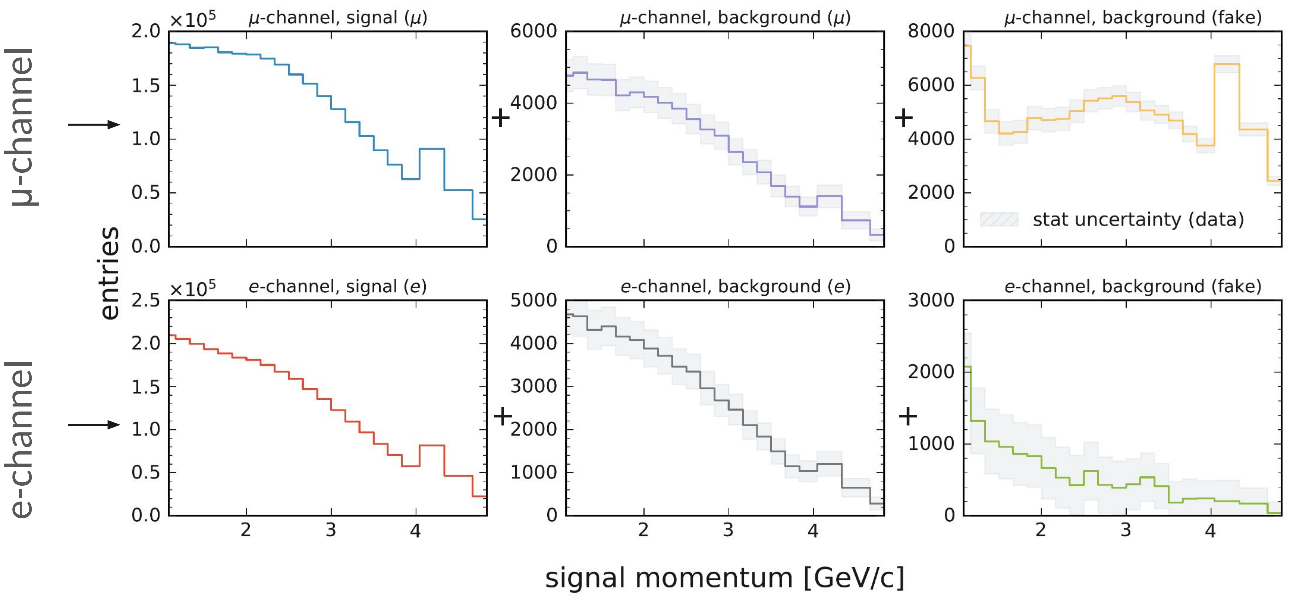
MC templates

separate events into 2 channels (based on lepton ID), each with 3 samples

signal

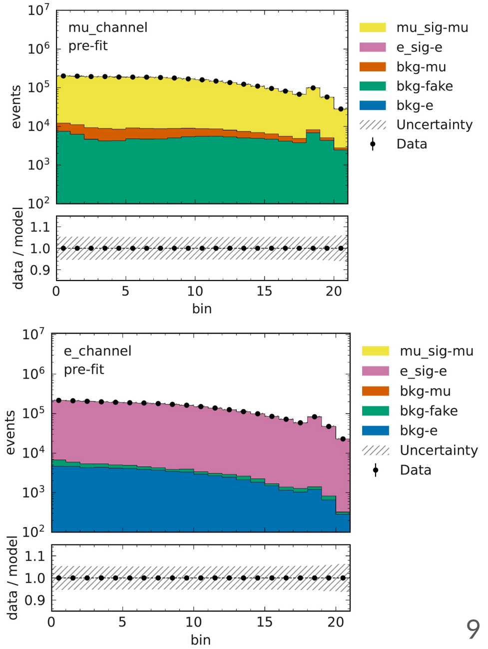
background (correct lepton)

background (mis-id)



=

asimov data



Systematic uncertainties

	uncertainty [%]	parameters	modifier
electron ID		7	histosys
control sample (signal embedding)		51*	shapesys
MC sample size		42	staterror
muon mis-ID		7	histosys
trigger		1+51*	normsys+shapesys
$\pi\pi^0$ modelling		113	shapesys
FSR (20% variation)		2	normsys+histosys
muon ID		7	histosys
electron mis-ID		7	histosys
normalisation		6	normsys
photon efficiency		1	histosys
π^0 efficiency		1	histosys
NN selection		1	histosys
misalignment		51*	shapesys
luminosity		1	lumi
ISR		2	normsys+histosys
decay in flight (5% variation)		1	histosys
BF($\tau \to e\nu\bar{\nu}$)		1	normsys
tracking efficiency		1	normsys
photon energy bias		1	histosys
track momentum		0	histosys
total systematic uncertainty			
statistical uncertainty			
total		253	

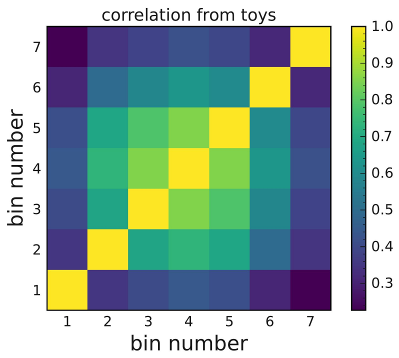
type of modifier depends on the correlation between template bins/samples

- uncorrelated shape: fully independent, one nuisance parameter per bin
- correlated shape: need to provide up/down variation, one nuisance parameter
  - can be shared for different samples
  - often cancels in the ratio
- partial correlation can be modeled with multiple correlated shape modifiers
  - using this for lepton ID

Template variations from covariance

more details in backup

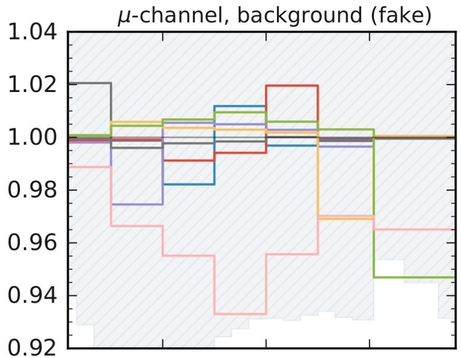
correlation for muon fake rate in 7 momentum bins



assumed correlation

eigendecomposition

$$C^{\mathrm{sys}} = VUV^T$$



$$\Gamma = V\sqrt{U}$$

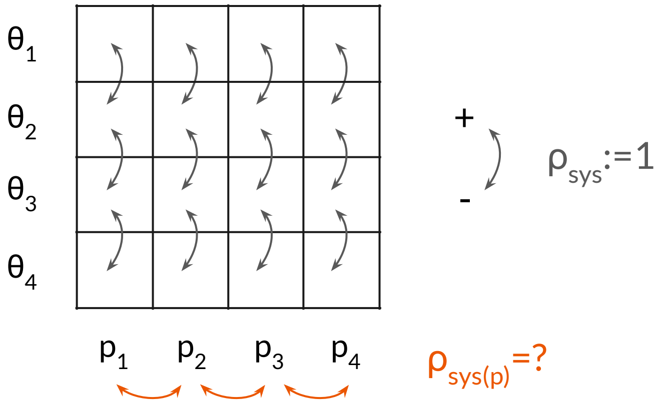
each colour shows a correlated shape variation (histosys)

Correlation of lepton ID systematics

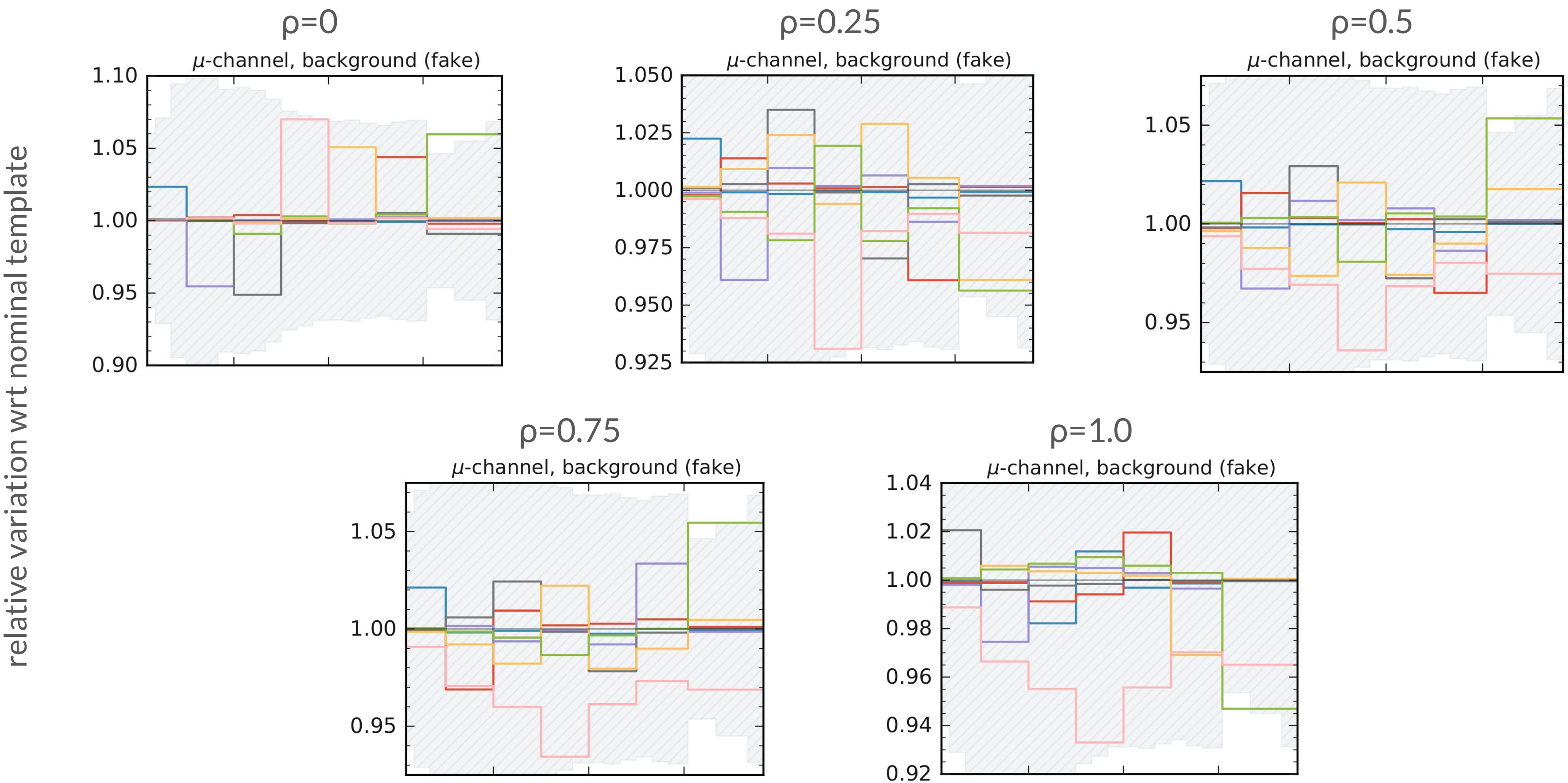
- LID uncertainties (stat + sys) are provided in bins of polar angle, momentum and charge
- statistical component of LID uncertainties is independent across correction bins
- correlation for the systematic component is currently unknown, need to make assumptions
  - want to be conservative and maintain a model that is able to describe the data

assuming correlation across...

- **charge**: fully correlated (conservative)
- **θ**: fully correlated (conservative)
- ***p***: not clear what is conservative, since fit is performed in momentum bins

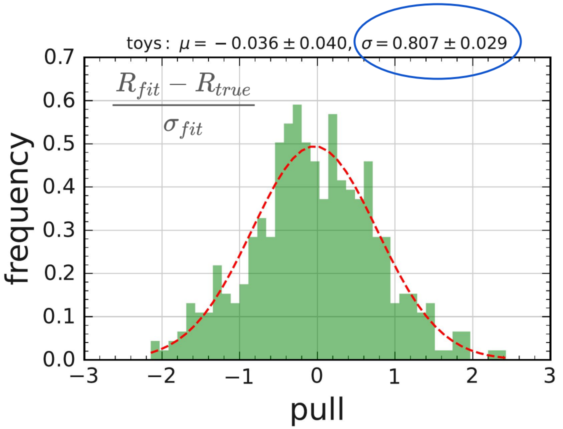


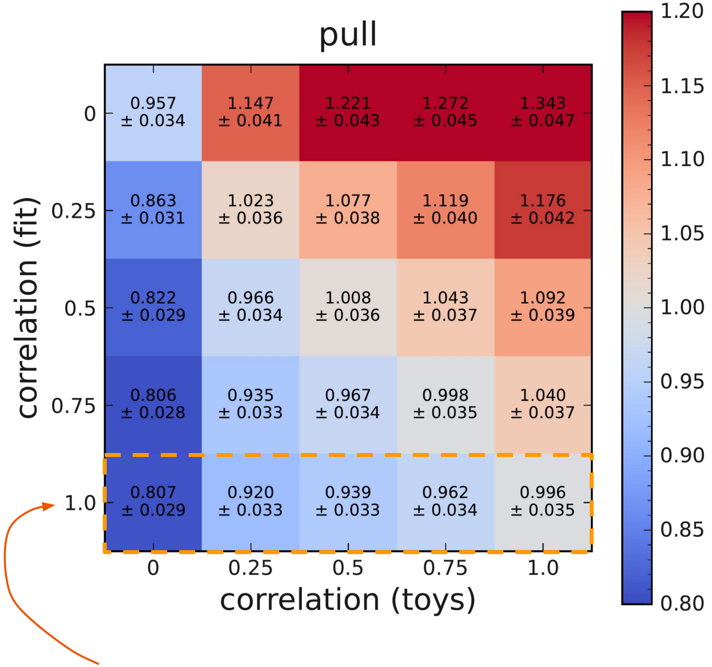
compute template variations for different correlations: $\rho_{sys(p)} = \{0, 0.25, 0.5, 0.75, 1.0\}$



- want to identify the most conservative assumption
- we check the standard deviation of the pull distribution when fitting toys generated with one correlation with a model assuming a different correlation

pull when having ρ=0 in toys an ρ=1 in model





$\rho_{sys(p)}$= 1 seems to be a good choice

independent variations probably ‘absorbed’ by other systematics (shapesys)

Summary (why I use pyhf)

- easy to build a complicated statistical model out of well defined building blocks
- integrates well with different backends
- written in python
  - transparent: easy to read + modify code
  - interactive: can use jupyter
- limited to binned models, but flexible enough for many applications
  - can even model correlation, even though nuisance parameters are always independent

Thank you!

$$f(\boldsymbol{n},\boldsymbol{a}\,|\,\boldsymbol{\eta},\boldsymbol{\chi}) = \underbrace{\prod_{c\in\text{ channels}} \prod_{b\in\text{ bins}_c} \text{Pois}\left(n_{cb}\,|\,\nu_{cb}\left(\boldsymbol{\eta},\boldsymbol{\chi}\right)\right)}_{\substack{\text{Simultaneous measurement}\\ \text{of multiple channels}}} \underbrace{\prod_{\chi\in\boldsymbol{\chi}} c_{\chi}(a_{\chi}|\,\chi)}_{\substack{\text{constraint terms}\\ \text{for "auxiliary measurements"}}}$$

$$\nu_{cb}\left(\boldsymbol{\phi}\right) = \sum_{s\in\text{ samples}} \nu_{scb}\left(\boldsymbol{\eta},\boldsymbol{\chi}\right) = \sum_{s\in\text{ samples}} \underbrace{\left(\prod_{\kappa\in\boldsymbol{\kappa}} \kappa_{scb}\left(\boldsymbol{\eta},\boldsymbol{\chi}\right)\right)}_{\text{multiplicative modifiers}} \left(\nu_{scb}^{0}\left(\boldsymbol{\eta},\boldsymbol{\chi}\right) + \underbrace{\sum_{\Delta\in\boldsymbol{\Delta}} \Delta_{scb}\left(\boldsymbol{\eta},\boldsymbol{\chi}\right)}_{\text{additive modifiers}}\right)$$

Description	**Modification**	**Constraint Term** c_χ	**Input**
Uncorrelated Shape	$\kappa_{scb}(\gamma_b) = \gamma_b$	$\prod_b \text{Pois}\left(r_b = \sigma_b^{-2} \middle|\, \rho_b = \sigma_b^{-2}\gamma_b\right)$	σ_b
Correlated Shape	$\Delta_{scb}(\alpha) = f_p\left(\alpha\middle|\, \Delta_{scb,\alpha=-1}, \Delta_{scb,\alpha=1}\right)$	$\text{Gaus}\left(a = 0\middle|\, \alpha, \sigma = 1\right)$	$\Delta_{scb,\alpha=\pm1}$
Normalisation Unc.	$\kappa_{scb}(\alpha) = g_p\left(\alpha\middle|\, \kappa_{scb,\alpha=-1}, \kappa_{scb,\alpha=1}\right)$	$\text{Gaus}\left(a = 0\middle|\, \alpha, \sigma = 1\right)$	$\kappa_{scb,\alpha=\pm1}$
MC Stat. Uncertainty	$\kappa_{scb}(\gamma_b) = \gamma_b$	$\prod_b \text{Gaus}\left(a_{\gamma_b} = 1\middle|\, \gamma_b, \delta_b\right)$	$\delta_b^2 = \sum_s \delta_{sb}^2$
Luminosity	$\kappa_{scb}(\lambda) = \lambda$	$\text{Gaus}\left(l = \lambda_0\middle|\, \lambda, \sigma_\lambda\right)$	$\lambda_0, \sigma_\lambda$
Normalisation	$\kappa_{scb}(\mu_b) = \mu_b$		
Data-driven Shape	$\kappa_{scb}(\gamma_b) = \gamma_b$		

Fit function

constraints

μ channel

e channel

$$f(\vec{n}|R_\mu, \vec{\chi}) = \prod_{b \in bins} \mathrm{Pois}\left(n_b^\mu \mid \nu_b^\mu(R_\mu, \vec{\chi})\right) \times \prod_{b \in bins} \mathrm{Pois}\left(n_b^e \mid \nu_b^e(R_\mu, \vec{\chi})\right) \times \prod_{\chi \in \vec{\chi}} c_\chi(a_\chi \mid \chi)$$

nuisance parameters

data

parameter of interest

$= \mathcal{B}_\mu$

$$\nu_b^\mu(R_\mu, \vec{\chi}) = R_\mu \times \mathcal{B}_e \times \nu_b^{\mu,\,\mathrm{sig}}(\vec{\chi}) + \nu_b^{\mu,\,\mathrm{bkg}(\mu)}(\vec{\chi}) + \nu_b^{\mu,\,\mathrm{bkg(fake)}}(\vec{\chi})$$

$\mathrm{BF}(\tau \to e\nu\nu)$ - multiplicative modifier

templates

$$\nu_b^e(R_\mu, \vec{\chi}) = \mathcal{B}_e \times \nu_b^{e,\,\mathrm{sig}}(\vec{\chi}) + \nu_b^{e,\,\mathrm{bkg}(e)}(\vec{\chi}) + \nu_b^{e,\,\mathrm{bkg(fake)}}(\vec{\chi})$$

Template variations from covariance

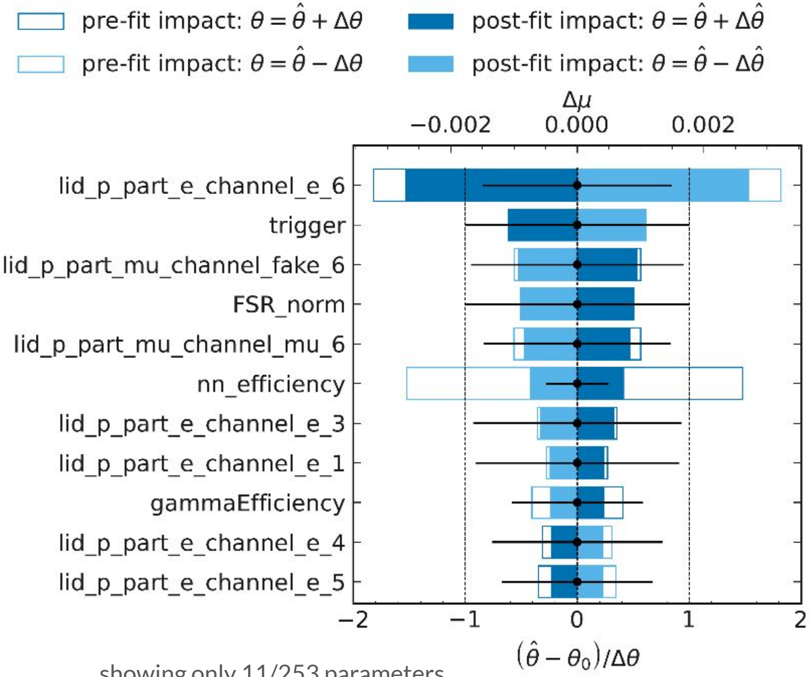
- we can model any covariance matrix (for template bins) with multiple shape variations (see PHYSTAT talk by S. Glazov)
- relation between covariance matrix and template variations: $C_{ik}^{\text{sys}} = \sum_{j=1}^{N_s} \Gamma_{ij}\Gamma_{kj}$
- Γ_{ij} is a variation of bin i associated to systematic source j, in total N_s independent sources
- variations can be obtained from a covariance matrix via eigendecomposition

$$C^{\text{sys}} = VUV^T = (V\sqrt{U})(V\sqrt{U})^T = \Gamma\Gamma^T \implies \Gamma = V\sqrt{U}$$

- this results in as many shape variations as we have template bins
- can model any correlations also across samples

Cabinetry

- cabinetry provides many convenience functions such as `cabinetry.fit.ranking()`
- pre-fit and post-fit impact on POI when varying nuisance parameters up/down one standard deviation
- can be used even if model is not created with cabinetry
- gives impact of each nuisance parameter individually, impact of a systematic source (e.g. lepton ID) can be obtained with toys



showing only 11/253 parameters

- test statistic $\tilde{t}_\mu$ from Asymptotic formulae for likelihood-based tests of new physics (Cowan, Cranmer, Gross, Vitells)
- based on profile likelihood ratio,

$$\tilde{t}_\mu = -2\ln\tilde{\lambda}(\mu) = \begin{cases} -2\ln\Big(L(\mu,\hat{\hat{\vec{\theta}}}(\mu))/L(0,\hat{\hat{\vec{\theta}}}(0))\Big) & \hat{\mu} < 0 \\ -2\ln\Big(L(\mu,\hat{\hat{\vec{\theta}}}(\mu))/L(\hat{\mu},\hat{\vec{\theta}})\Big) & \hat{\mu} \geq 0 \end{cases}$$