# Rivet analysis preservation & recasting

*including observations from Gambit, TopFitter, Les Houches, LPCC forum…*

Andy Buckley

University of Glasgow

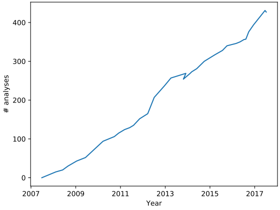
ATLAS data-reinterpretation workshop,
CERN, 21 July 2017

THE ROYAL SOCIETY · University of Glasgow · MCnet

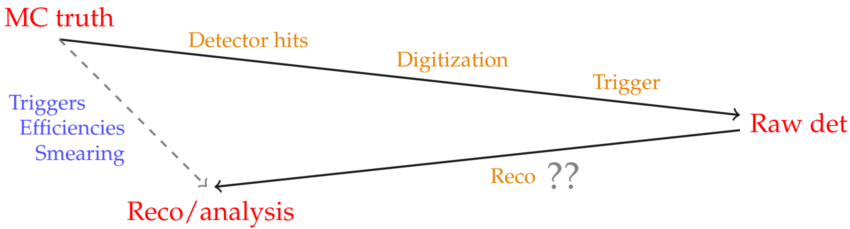
# Rivet background

**Rivet is an analysis system for MC events, and *lots* of analyses**

- Easy & powerful tool to get many physically meaningful plots from many MC gens
- "Physicist-friendly" code interface
- LHC standard for archiving *unfolded* data analyses
- Well-established in ATLAS: 172 analyses and official support manpower
- Athena interface and standalone tools in ATLAS software
- Since version 2.5.0, includes detector-smearing tools for BSM preservation/recasting




# Detector effects in 'fast-sim' vs. 'smearing'




- Explicit fast-sims work hard to get to the "Raw det" stage, but the big reco step is not known in detail
- In practice, kinematic smearing is subleading: we calibrate well! (And where it matters, is a fast-sim accurate enough?!)
- ∼All det effects in search analyses are lepton/tagging efficiencies
- ⇒ Rivet takes the "short route" of simple smearing, plus tabulated efficiencies cf. Delphes (+ CP notes & analysis papers)
- Effs (and resolutions) can be *analysis-specific*: many different WPs/effs of taggers, calibration & isolation across analyses/runs

# Detector sim in Rivet

- Rivet smearing/efficiency approach based on a refinement of the GAMBIT system, with less "code noise" and more scope for user-defined smearing functions
- Analysis-specific efficiencies and smearings are more precise and allow use of multiple jet sizes, tagger & ID working points, isolations, ... ⇒ **many variations in real analyses**
- Smearing uses established "projection" mechanism: familiar and efficient. Smeared objects are just "wrappers" around truth-level definitions
- BSM developments also produced powerful filtering tools: Rivet `Cut` objects and generalised C++ functions can all be used to apply complex selections. Cut-flow tools are included.

# Rivet smearing examples

## Leptons:

```
// Definition
FinalState es1(Cuts::abseta < 3.2 && Cuts::abspid == PID::ELECTRON);
SmearedParticles es2(es, ELECTRON_EFF_ATLAS_RUN2, ELECTRON_SMEAR_ATLAS_RUN2);
declare(es2, "Elecs");
...
// Usage
Particles elecs = apply<ParticleFinder>(event, "Elecs").particles(10*GeV);
```

## Jets:

```
// Definition
FastJets js1(FinalState(Cuts::abseta < 4.9), FastJets::ANTIKT, 0.4);
SmearedJets js2(js1, JET_SMEAR_PERFECT, JET_BTAG_EFFS(0.7, 0.12, 0.02));
declare(js2, "Jets");
...
// Usage
Jets jets = apply<JetAlg>(event, "Jets").jetsByPt(30*GeV);
```

## Also a `SmearedMET` ...

Standard global functions, plus user-defined. C++11 lambda fns etc. are allowed. Rivet 2.6 allows *chaining* of smearings and efficiencies.

# From analysis preservation to reinterpretation

| Description | #evt | tot.eff | rel.eff | #evt | tot.eff | rel.eff | tot.eff |
|---|---|---|---|---|---|---|---|
| 2jm cut-flow: | Rivet | | | MadAnalysis5 | | | CheckMATE |
| | 31250 | 100% | - | 32150 | 100% | - | 100% |
| Pre-sel+MET+pT1 | 28472 | 91% | 91% | 28478 | 91% | 91% | 91% |
| Njet | 28472 | 91% | 100% | 28477 | 91% | 100% | 91% |
| Dphi_min(j,MET) | 22950 | 73% | 81% | 22889 | 73% | 80% | 73% |
| pT2 | 22950 | 73% | 100% | 22889 | 73% | 100% | 73% |
| MET/sqrtHT | 10730 | 34% | 47% | 10710 | 34% | 47% | 33% |
| m_eff(incl) | 10630 | 34% | 99% | 10609 | 34% | 99% | 32% |

Prelim results from LH2017 BSM (AB, Grellscheid, Fuks, Desai)

Current Les Houches benchmarking study: Rivet analyses reproduce published full-sim & custom-config Delphes fast-sim within a few %

But signal-region counts are just the beginning. Plan to augment Rivet with a statistics suite to turn SR counts into BSM limits: Rivet $CL_s$ implementations in CONTUR and in Rivet contrib — potential for limit setting with combined SM+BSM data!

To streamline, we need to *standardise* distribution of not just observed counts, but also **SM background expectations, efficiency tables, cut-flows + other data for signal validation, and correlations**. Obvious route is HepData. Interest in ATLAS top, CMS, BSM pheno…

# Correlations and simplified likelihoods

Without correlations, reinterpretations have to be conservative: only use the single best-expected-ΔLL SR from each correlated group.
Better: full likelihoods or *simplified likelihoods cf. CMS*

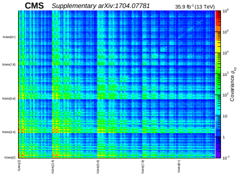
$$\mathcal{L}_S(\mu, \boldsymbol{\theta}) = \prod_{i=1}^{N} \frac{(\mu \cdot s_i + b_i + \theta_i)^{n_i} e^{-(\mu \cdot s_i + b_i + \theta_i)}}{n_i!} \cdot \exp\left(-\frac{1}{2}\boldsymbol{\theta}^T \mathbf{V}^{-1} \boldsymbol{\theta}\right)$$

$$V_{ij} = E[\theta_i \times \theta_j]$$

See CMS NOTE-2017/001

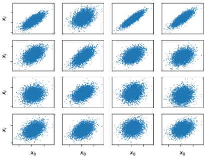
with averaging over elementary bkg nuisance distributions

SL cov



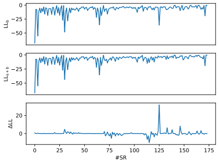

⇒

Corr sampling



⇒

ΔLLs




Technical discussion / implementation needed on whether separated systematic cov matrices, simplified cov matrices, or nuisance param forms best. How to identify dataset types, and match cov indices across observables? Etc.