Strategies for Delivering POG Corrections to Analysis

Jeffrey Berryhill (FNAL)

Physics Objects Commissioning Workshop

Leap Day, 2008

- Publishers and Customers for Corrections
- Measurement design and output format
- Workflows for use in analysis

POG Measurements and Analysis

- In this workshop, an impressive array of **object calibration, efficiency,** and **fake rate measurements** anticipated by POGs, even with a few pb^{-1}

But how can the average user exploit those to move swiftly to a high-quality published analysis?

- Without communication in advance on how these could be used for analysis, there will likely be confusion, producer and user error, and a large duplication of effort.

(We do not have enough FTEs, e.g., to have every individual analyst measure the Z for their calibrations!)

- Purpose of this discussion: **devise a workflow for the production and consumption of data-driven POG corrections**

Categories of Corrections/Calibrations

"Deep" Corrections:

- **raw-level calibrations of data:** procedures whose improvement to objects can only be realized by (re)reconstruction of the raw data
- **reco-level for data:** requires access to reco
- **SIMU-level for MC:** requires redoing simulation production
- Resource-intensive exercises which will occur as infrequently as possible, by collective agreement, and initiated by a pro.

Fast (re)simulation is somewhere in between

User Corrections:

- **AOD-level for data or MC:** requires access to AOD
- **Post-AOD:** requiring access to PAT layers, or to ntupled AODs of high-level objects
- It may be that valiant efforts at deep-level corrections still result in the need for correction factors to match MC with data at this level
- Can be (re)run by the average user with low overhead
- <u>I will limit this discussion to corrections of this type.</u>

Publishers: Popular POG corrections realizable at the User level

- **Physics object ID efficiency weighting**
  - Offline ID efficiencies in data and in MC
  - Online Trigger efficiencies in data and in MC
  - Fake Rates in data and in MC
  - Realized by re-weighting events (DATA/MC)
  - Planned for (at least) e, gamma, muon, tau, b-tag

- **Physics object 4-vector modification**
  - Energy or momentum scale corrections post-AOD
  - Resolution corrections post-AOD
  - Re-smearing of MC 4-vectors
  - Examples discussed in this workshop: jet-energy scale, ECAL supercluster ET corrections, muon PT scale

- **Physics object MVAs** (common data-driven likelihoods, NN kernels, etc.)

Customers: Analysis Use Cases

- **Object ID weights**:
  - Corrected signal efficiency estimation from signal MC
  - Corrected background rate determination from bkg MC
  - Selection optimization based partially or totally on signal and background corrected MC
- **Object 4-vectors**:
  - Corrected data objects or MC objects (or both) to re-select data candidates or re-estimate signal acceptance
  - To remodel mass peaks or other kinematic PDFs/templates
- **Object MVAs**: ingredients to signal or background PDFs/templates

ID Weighting example: Signal Efficiency for $W \rightarrow e\nu$ Cross Section

1. W selection, electron trigger, and signal extraction method designed

2. Electron efficiency measured in $Z \rightarrow ee$ data with tag-and-probe
   For several bins of electron ET and eta (and perhaps other vars.)
   For the specific electron selection in 1.
   For the electron trigger path chosen in 1.

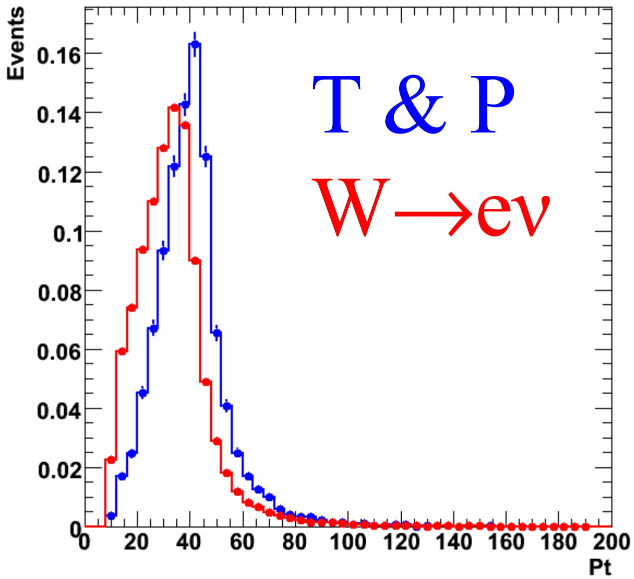
3. Repeat tag-and-probe (or MC truth based) electron efficiency measurement in Z MC

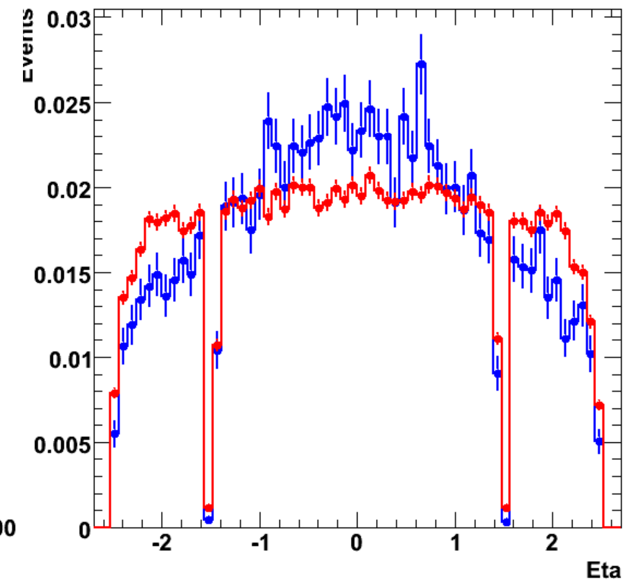
4. For each W signal MC electron candidate, re-weight by w_i = $(DATA/MC)_i$ offline selection and online trigger selection efficiency ratios for that tag-and-probe bin in ET and eta

5. Corrected signal efficiency is ($\Sigma f_i w_i$), f_i the relative W electron acceptance of the ith efficiency bin. Possibly repeat 1-5 to optimize S/sqrt(S+B).
6. Uncertainty determined from W MC stats, and errors in w_i and f_i

Can we partially or fully automate steps 2 through 6?

ID Weighting example: Signal Efficiency for $W \rightarrow e\nu$ Cross Section

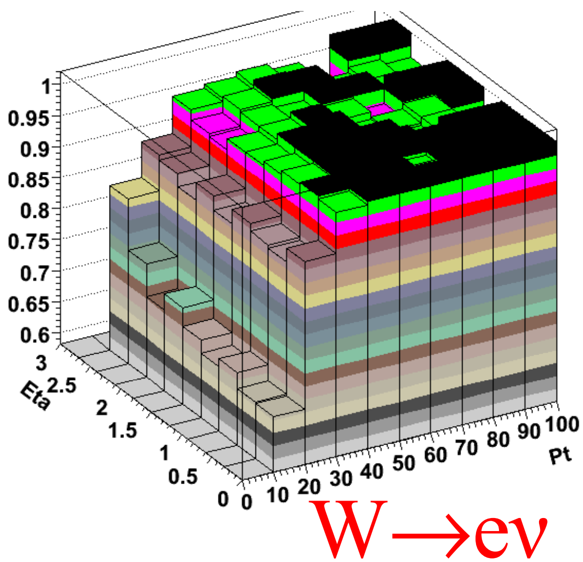




Procedure used in CMS 2007/026.

W and Z electrons have differing PT and eta spectra = different average eff.

At the % level, <u>binned</u> T&P electron efficiencies model W electron efficiency correctly



Cut Efficiency from $W \rightarrow e\nu$ MC truth :
90.98±0.10

Cut Efficiency from T & P applied in $W \rightarrow e\nu$ distributions:
89.4±0.5 (T&P errors estimated with 10 pb^{-1})

Common Tools and Formats for POG Correction Production

- **Tools and samples** for measuring corrections should be **recycled** wherever appropriate
  - Maintains reproducibility and transparency
  - Reduces common and subtle mistakes
  - Reduces software maintenance and computing resource consumption
  - E.g.: electron, muon Z tag-and-probe (and tau use of same) should be a unified package→ in progress
- Tools should be easily configurable to **test a new selection**
- A workflow should be established within the POG to **regularly (re)produce their corrections in the CAF** or the relevant Tier 2.
- **Online and offline DQM** tasks for those samples should be supported as well.

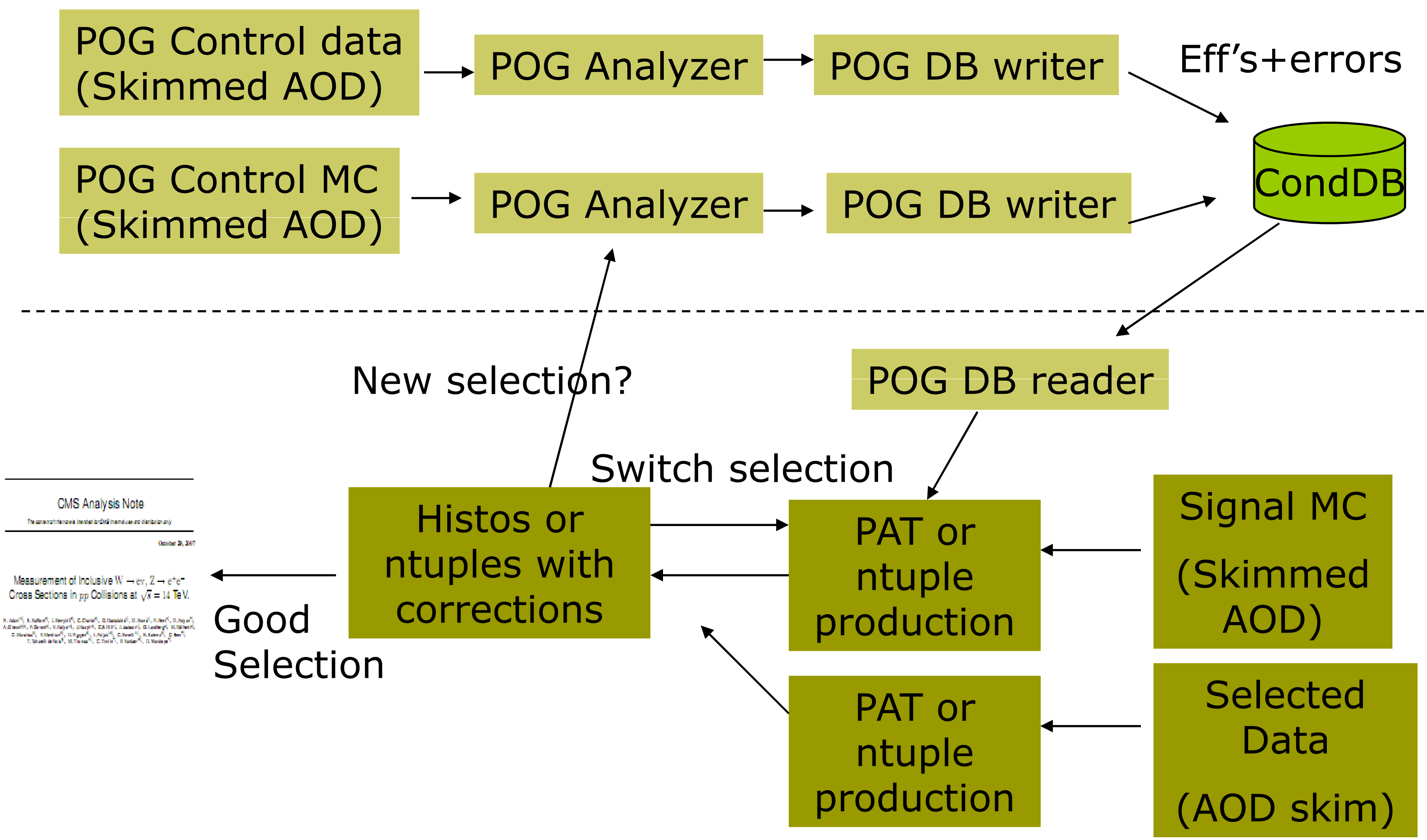
Common Tools and Formats for POG Correction Production

- The output (usually LUTs) should be archived in
  - flat txt files,
  - root files, AND
  - (in the long term) **the offline conditions database.**
  - The measurements from data will in general be binned in time so they would have a natural IOV (the integration period over the control sample used)

- **A DB tool to store the output LUTs, retrieve them, and provide simple plots/views of it needs to be developed.**

- Likely that **a general DB software solution** suiting many of the POG corrections could be found (JetMET JEC?)

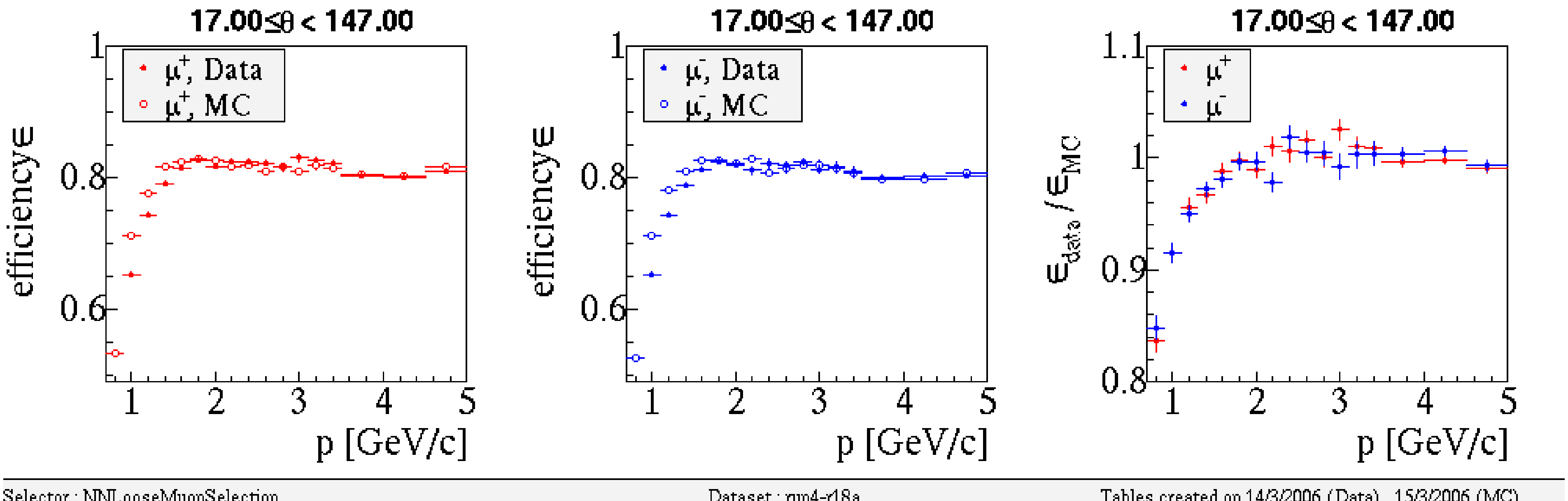
Possible Workflow for AOD-level Corrections



Experience from other experiments

D0 has a systematic electron and muon ID certification system for all proposed lepton selection algorithms, with POG-delivered efficiencies

BaBar-ians will immediately recognize this proposed computing model as the Particle ID ("PID table") system for data-driven efficiency corrections



Atlas is also contemplating how user-level corrections factor into analysis.

Possible Physics Tools Implementation

- Assuming this data can be retrieved from wherever it is archived, pursue a way of introducing it into physics objects, as class data or a class function. Then it would be possible to get (single object) efficiency corrected MC histograms via

  Myhisto.Fill(MyObject.pt(), MyObject.eff_correction())

  where the user need not know all of the interface details to use it correctly
- <u>Object or correction function needs access to:</u>
  Particle ID selection algorithm (using provenance?)
  run period and possibly other run conditions (inst. Lumi)
  Possible topological conditions (DR to a jet?)
  Object 4-vector
- Also need function returning eff_correction error (in general a covariance matrix).

Possible Physics Tools Implementation

- Assuming objects have a way of finding their efficiency correction, then multi-object efficiencies can be corrected for from Event Hypothesis class methods:

Myhisto.Fill(MyHypothesis.mass(), MyHypothesis.eff_correction())

where MyHypothesis.eff_correction() performs the product over constituent single object efficiencies automatically.

- Trigger efficiency (w.r.t. offline ID efficiency) corrections of events will be both selection dependent and trigger path dependent, so the algorithm is more complex, but should be manageable for at least the simplest cases.

Possible Physics Tools Implementation

- POG fake rates: can be implemented in the same fashion as efficiency corrections

- 4-vector modification: either simple PAT 4-vector scaling method, or (better) just have a producer module which makes the corrected object list.

Conclusions

• Data-driven corrections from POGs must be considered as part of a workflow for productive analysis

•Simple, empirical, post-AOD reweighting of events or rescaling of 4-vectors will always be part of that strategy

•POGs should consider:

  an appropriate standardization of their tools to realize this
  a DB format for their output
  a tool for reading and writing that format

Conclusions

•PAGs should consider:

whether and how to use these corrections.

flexible enough? binned with enough dependent variables?

(i.e. **are they "analysis independent" enough?**)

for what preferred control samples and selections?

•PAT should consider:

a **simple Hypothesis-layer method for event reweighting** or candidate 4-vector rescaling, for which the corrections could be entirely transparent to the user

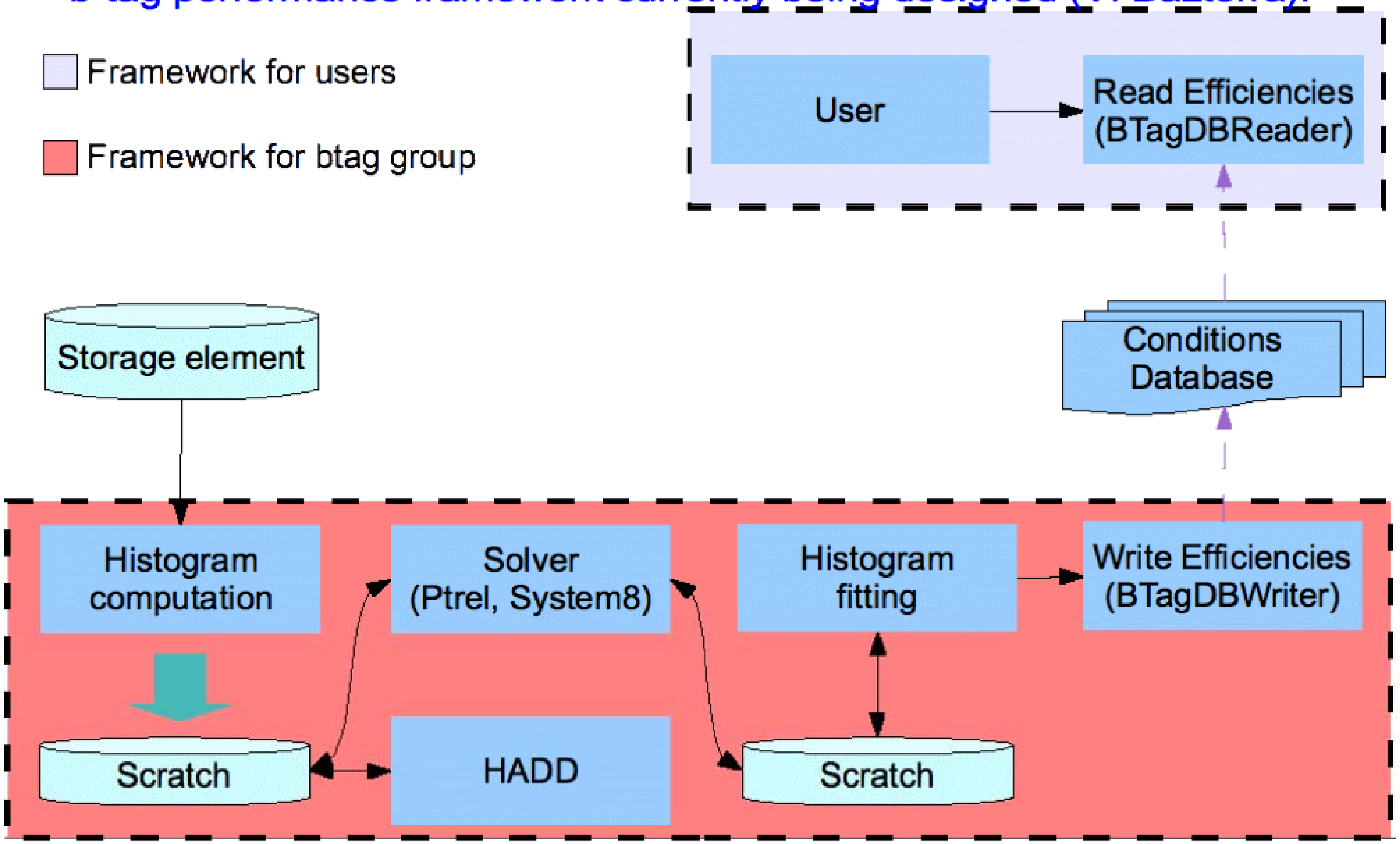
I would like to thank the following colleagues for helpful discussions:
Martijn Mulders, Nicola Amapane, Alexei Safonov, Francisco Yumiceva, Pascal Vanlaer, Kostas Kousouris, Petar Maksimovic, Roberto Tenchini, Victor Bazterra

Backups

-

Backups

- b-tag performance framework currently being designed (V. Bazterra):



Weighting methods

- **Typical use case is estimation of signal efficiency from signal MC, with (semi-)empirical corrections.**
- At least four different schemes:
  - Relative weighting: MC events have non-unit weight. Full simulated selection is applied. Weight is ratio of efficiency in data to efficiency in MC.
  - Absolute weighting: MC events have non-unit weight. MC truth + acceptance selection is applied. Weight is absolute efficiency determined by data (i.e. simulation is ignored).
  - Relative "killing": De-weighted variation of relative weighting. Full selection is applied and random number is compared against relative weight to determine final accept/reject. Only works if relative weight is < 1. Does not commute with any downstream data reduction.
  - Absolute killing: De-weighted variation of absolute weighting. "Look-up table based fast simulation"

Weighting methods

- **Typical use case is estimation of signal efficiency from signal MC, with (semi-)empirical corrections.**
- My opinion:
- Relative weighting is the most likely use case for physics analysis in the long run
- Absolute weighting is the next best option for times or cases where simulation is poorly understood
- Killing options do not commute with any downstream data reduction and possibility for user error is high. They can be supported if there is demand for it, and can be simply derived from weighting schemes.

Requirements for Analysis

- **Scheme details for weighting:**
  - For perfect simulation, efficiency is N(selected)/N(generated)
  - For imperfect simulation, efficiency is (ignoring trigger for the moment)

  Σ_i (Π_j R_j(ET_ij, eta_ij, phi_ij,…))/N(generated)

  where R_j is data/MC efficiency ratio of the jth selected object in the ith selected MC event
  - For poor simulation, efficiency is similar, except R_j is replaced with the efficiency measurement from data
  - Combined uncertainty from MC stats+R_j covariance matrices

Trigger Efficiencies

- Efficiency as we have defined it thus far is for offline selection on a perfectly efficient trigger.
  - For 100% efficient trigger and no unmeasured correlation in object selection, corrections factorize into single object corrections
- Corrections for trigger efficiency do not in general factorize and are **analysis-dependent and trigger-path-dependent**

  e.g. Z -> ee on single electron trigger,

  eff = 1 – (1-eff_1(e+))(1-eff_1(e-))

  Z -> ee on single threshold, two electron trigger

  eff = eff_2(e+e-) = eff_2(e+)eff_2(e-)

  H -> 4e on OR of several triggers = complicated function of various disjoint subsets of events

Trigger Efficiencies

- For this reason, the POGs up to this point have defined "offline efficiency" for objects with no trigger requirements, and "online efficiency" as efficiency of trigger w.r.t. selected offline objects
  - Eff = (∏_j eff_off_j) * eff_on
  - A well designed selection has eff_on close to unity (else offline selection is "too loose" for the object required by the trigger path)
- Eff_on will depend on both the event hypothesis and the HLT path(s). If the HLT path is a logical combination of single object requirements, and if those single object conditions can be measured separately and archived, then automated path-dependent corrections are conceivable as well.

Trigger Efficiencies

If that trigger efficiency function exists, then total efficiency correction is a single function call:

```
Myhisto.Fill(MyHypothesis.mass(), MyHypothesis.eff_correction())
```

where either MyHypothesis knows the trigger path already or the trigger path is an argument to eff_correction. eff_correction retrieves eff_off*eff_on combined correction.

Links to POG activity

Egamma:

http://cms.cern.ch/iCMS/analysisadmin/analysismanagement?ancode=EGM_07_001

https://twiki.cern.ch/twiki/bin/view/Sandbox/JasonHauptElectronEff

Muon:

https://twiki.cern.ch/twiki/bin/view/CMS/MuonTagAndProbe

Btag:

http://cms.cern.ch/iCMS/jsp/analysis/admin/analysismanagement.jsp?ancode=BTV_07_001

https://twiki.cern.ch/twiki/bin/view/CMS/SWGuideBTagPerformanceFromData