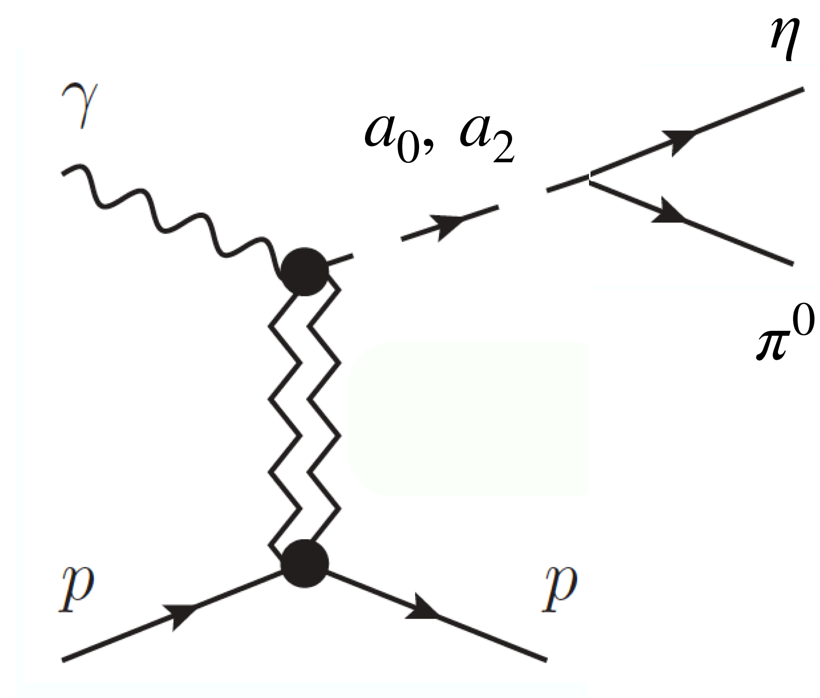
Typical Analysis Workflow

Justin Stevens

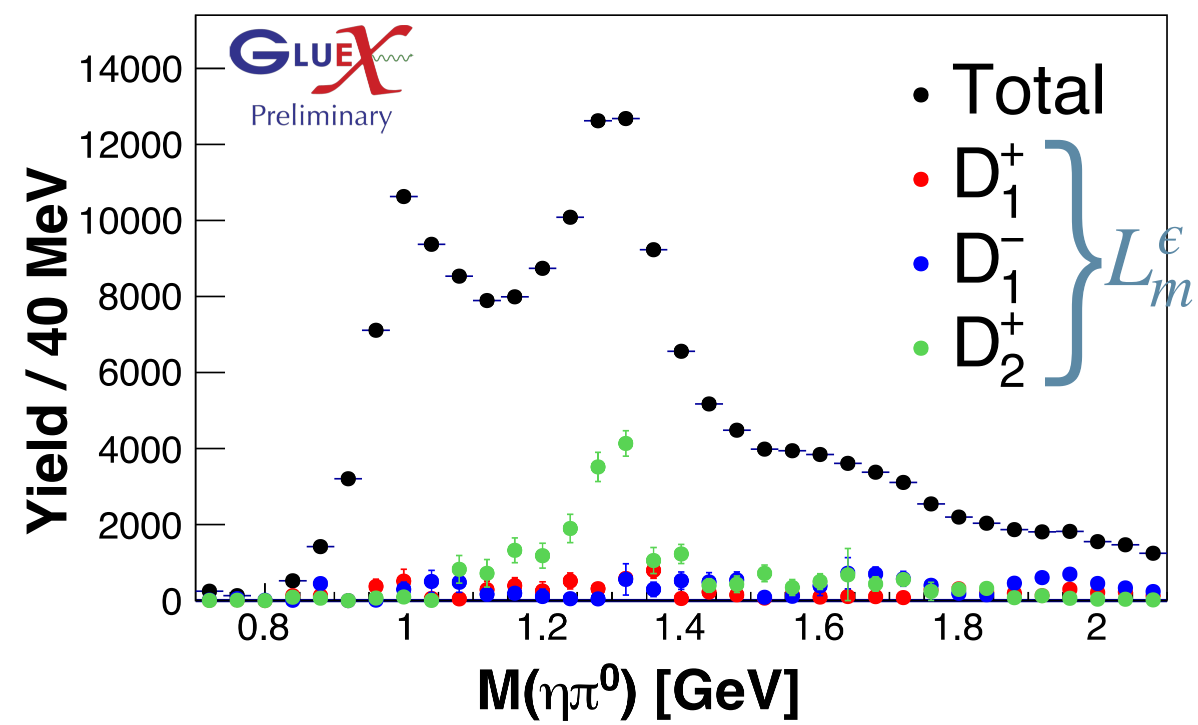
William & Mary
CHARTERED 1693

Big picture analysis strategy

- **Goal:** obtain pure sample of $\gamma p \to \eta\pi^0 p$ to study contributing amplitudes

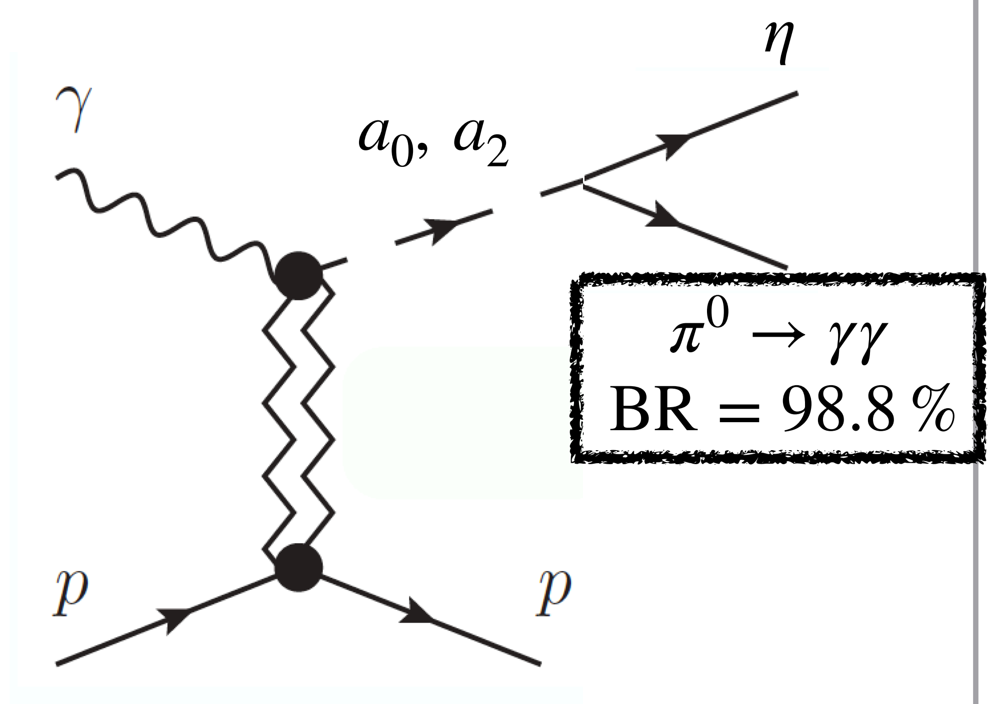


- **Necessary steps:**
  - Choose appropriate ReactionFilter and Kinematic Fit options
  - Apply selection criteria (i.e. cuts) which efficiently reject background but keep signal of interest
  - Statistically subtract remaining background, not removed by cuts
  - Measure yield for cross section or fit angular distributions for beam asymmetry or amplitude analysis



How to reconstruct your final state?

	Mode	Fraction (Γ_i / Γ)
Neutral modes		
Γ_1	neutral modes	$(72.12 \pm 0.34)\%$
Γ_2	2γ	$(39.41 \pm 0.20)\%$
Γ_3	$3\pi^0$	$(32.68 \pm 0.23)\%$
Charged modes		
Γ_8	charged modes	$(27.89 \pm 0.29)\%$
Γ_9	$\pi^+\pi^-\pi^0$	$(22.92 \pm 0.28)\%$
Γ_{10}	$\pi^+\pi^-\gamma$	$(4.22 \pm 0.08)\%$



- Exclusive: if possible, reconstruct all final state particles!
- Decay modes:
  - Typically lower efficiency for each final state gamma, pi, K, proton
  - Common question: should you mass constrain in the kinematic fit?
- Comparison of multiple decay modes provides systematic cross check (major strength of GlueX)

ReactionFilter + Kinematic Fit

- Reminder that KinFit will force mass peaks, even when they don't exist…
- What other ways can you get 4 photons?
  - Generate gp -> eta pi0 p and gp -> pi0 pi0 p and look at mass spectra
- General: leave at least one mass un-constrained, so you can fit or side-band subtract from the peak

Event selection 101

- Reminder of cuts applied in ReactionFilter (Beni's talk)?
  - Loose mass windows, loose timing cuts, KinFit convergence
- What additional cuts are common and why are they needed?
  - KinFit Chi2/NDF, PID, beam energy, unused showers…
- How do you know what cuts to use?
  - Study in simulation! Efficiency vs Purity

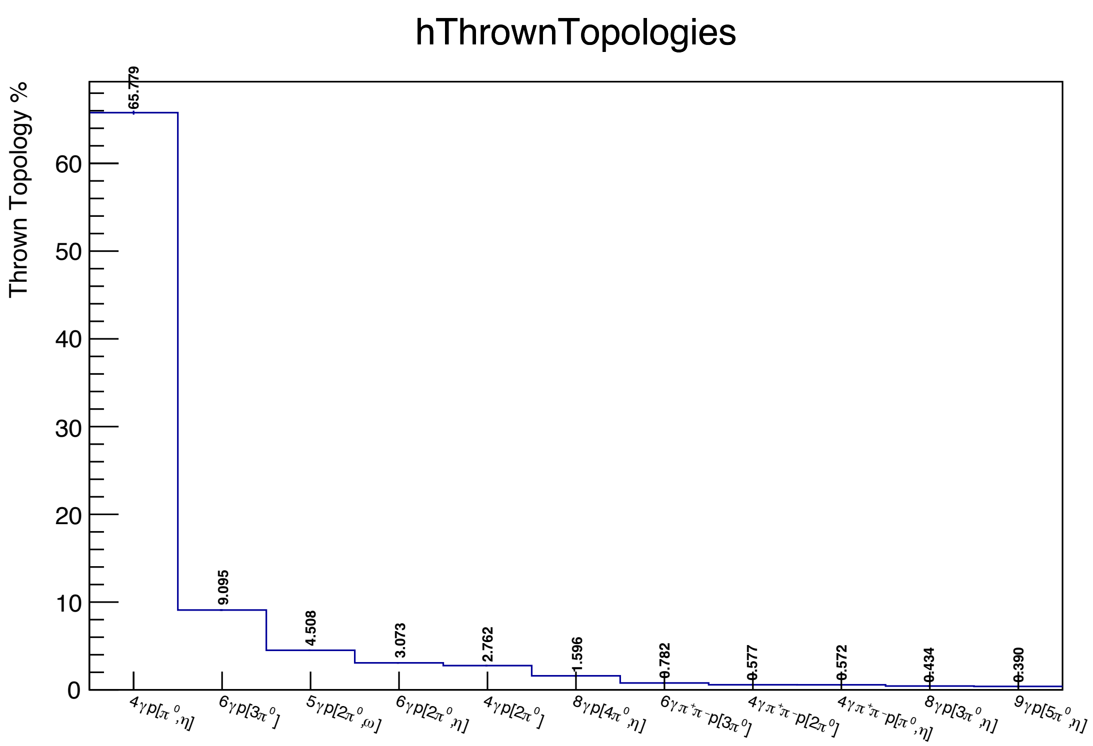
Studying backgrounds with bggen

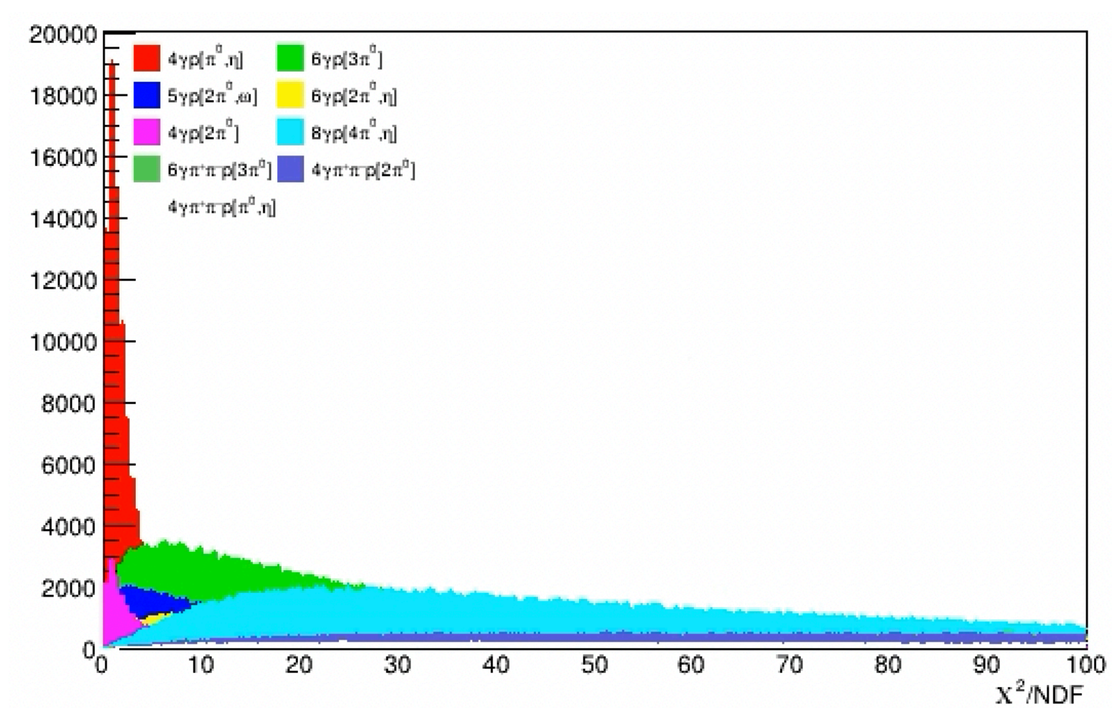
- What is bggen?
  - Inclusive MC generator for “all” photoproduction processes…
- In simulation we can cheat and sort events by topology
- Some example plots to show topologies are different: 2g masses (shuffled g’s), KinFit Chi^2, unused showers

- With large bggen MC samples need to tag events by their “thrown topology” to identify background sources
- Information already in Analysis TTree format:
  - DSelector library creates a unique TString: **NumFinalState[Decaying]**
  - e.g. $\gamma p \to \pi^0 \eta p$ with $\eta \to \gamma\gamma$ and $\pi^0 \to \gamma\gamma$ corresponds to the TString: **$4\gamma p[\pi^0,\eta]$**

Studying backgrounds with bggen

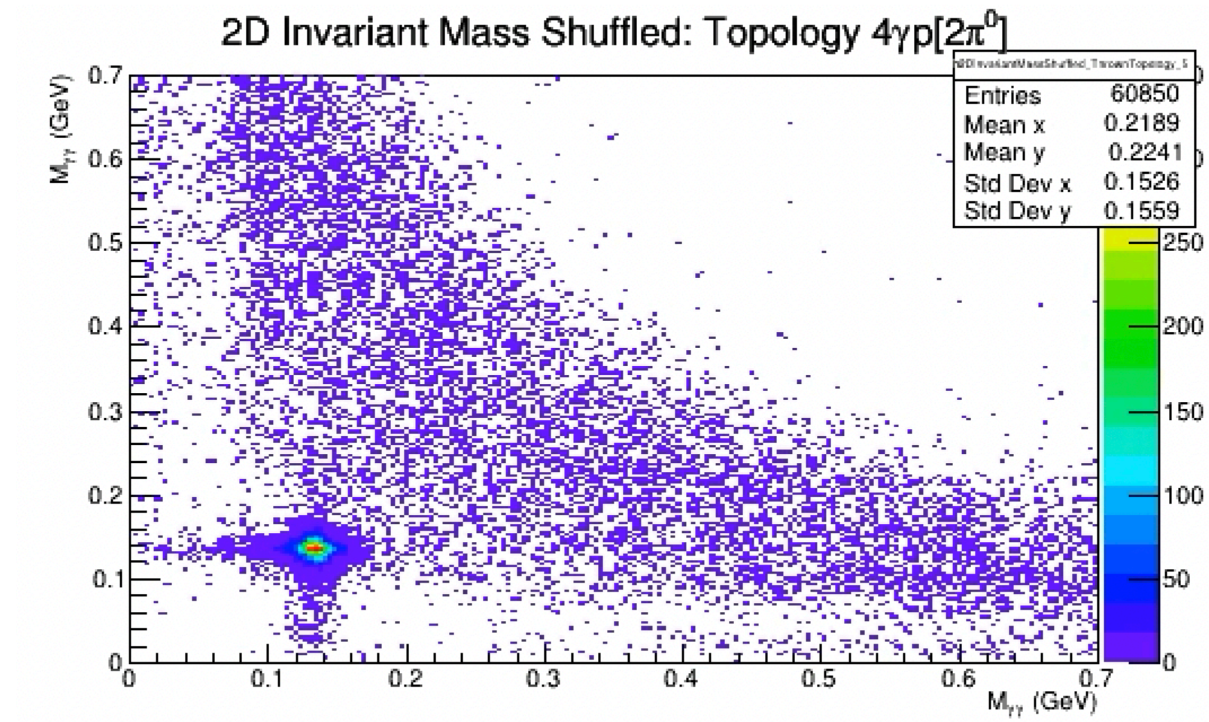
- What is bggen?
  - Inclusive MC generator for “all” photoproduction processes…
- In simulation we can cheat and sort events by topology
- Some example plots to show topologies are different: 2g masses (shuffled g’s), KinFit Chi^2, unused showers

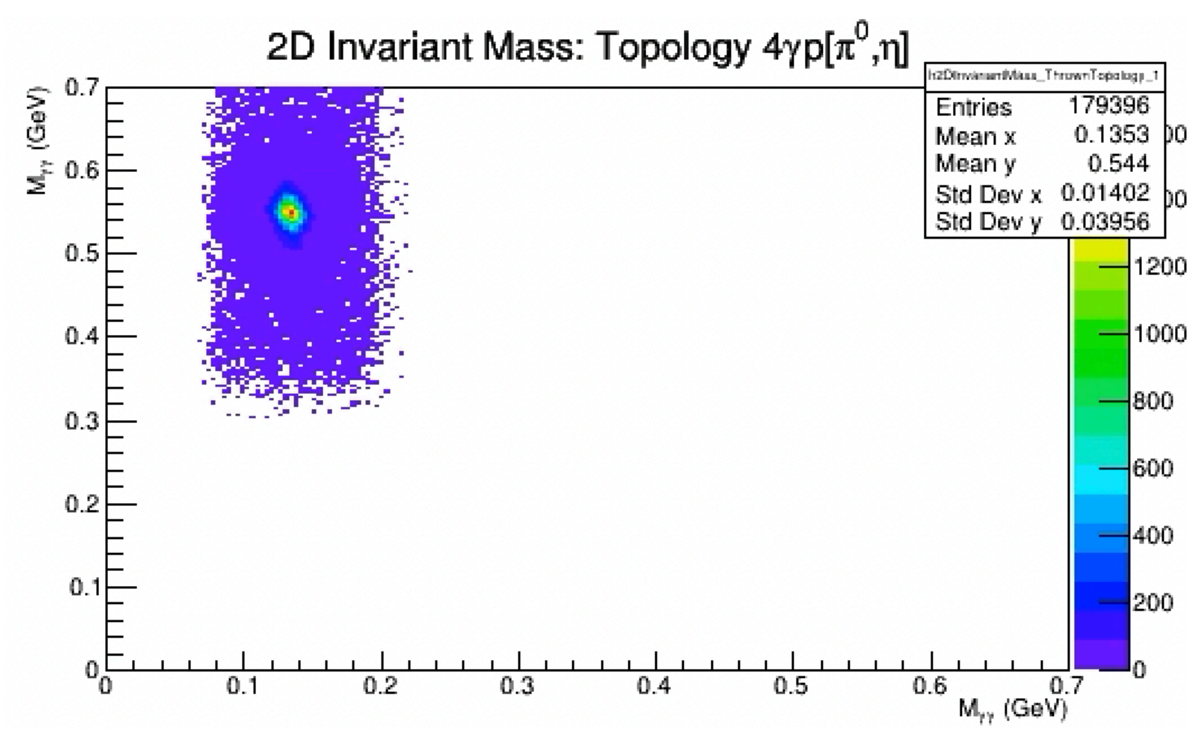


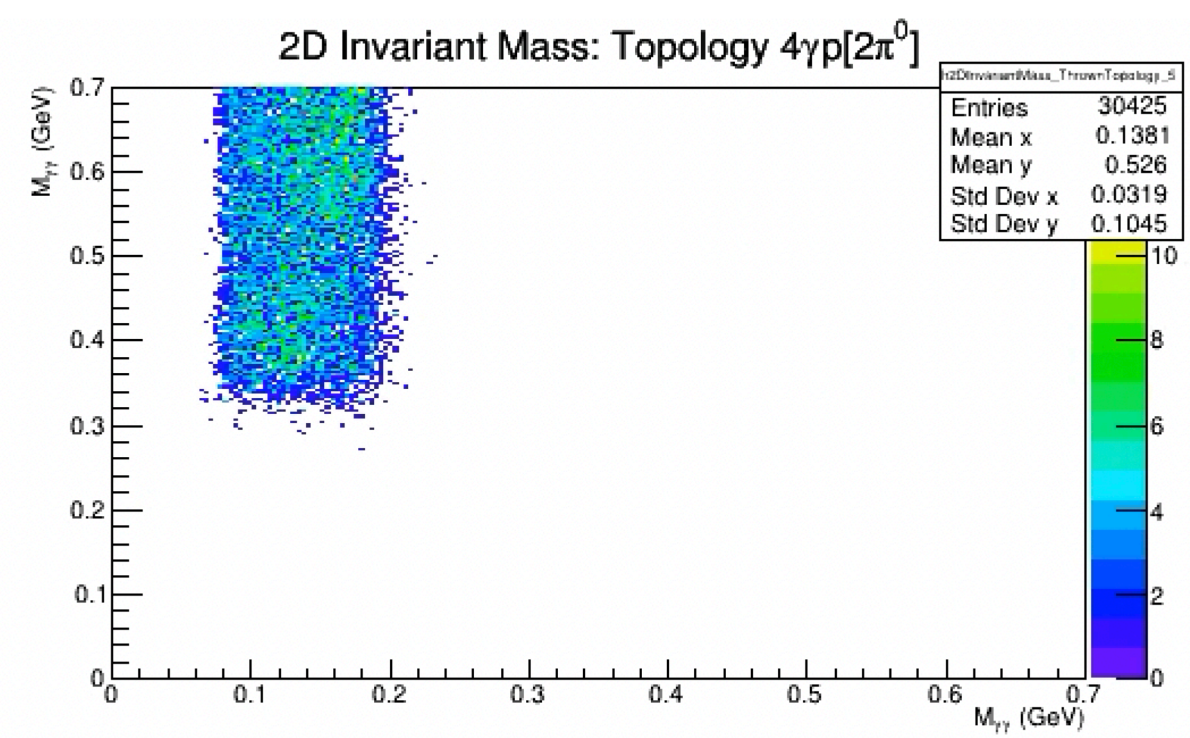


Common background topologies

- **Shuffled photons** or charged particles
- Non-exclusive:
  - (Final state of interest) + ɣ or pi0
  - (Final state of interest) + pi+ pi-
- Mis-identified charged particle (K <— > pi)

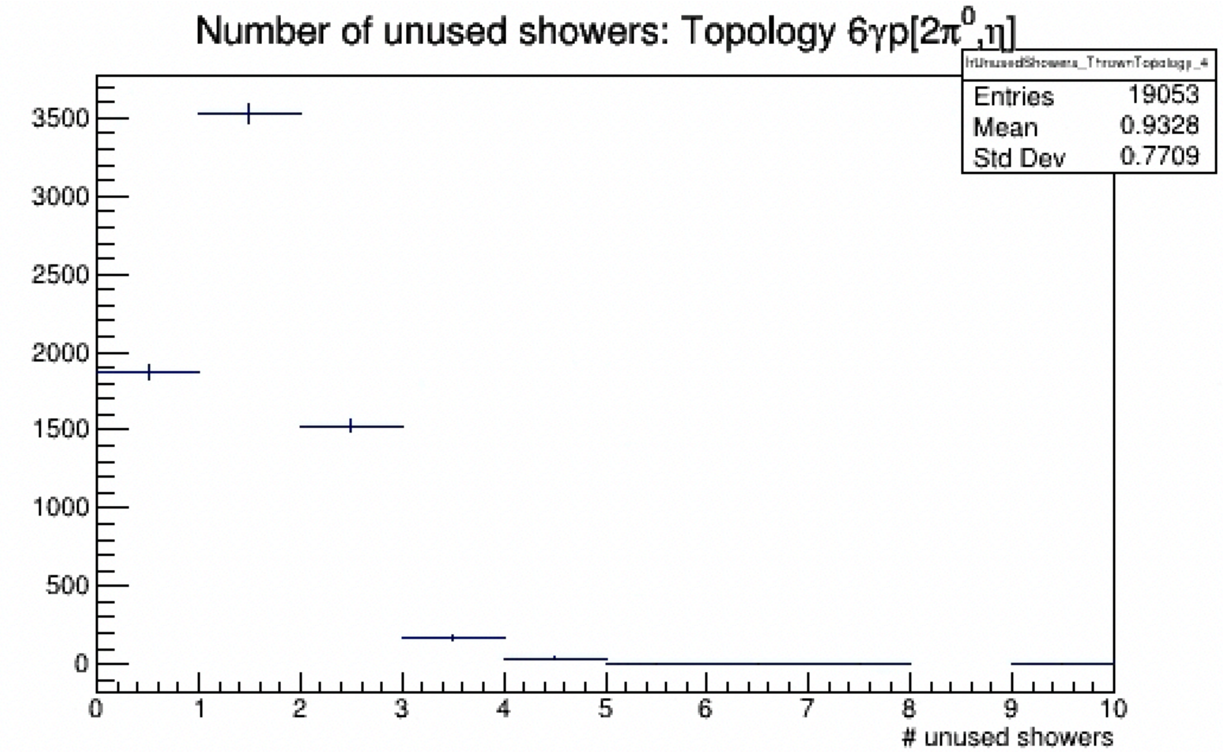


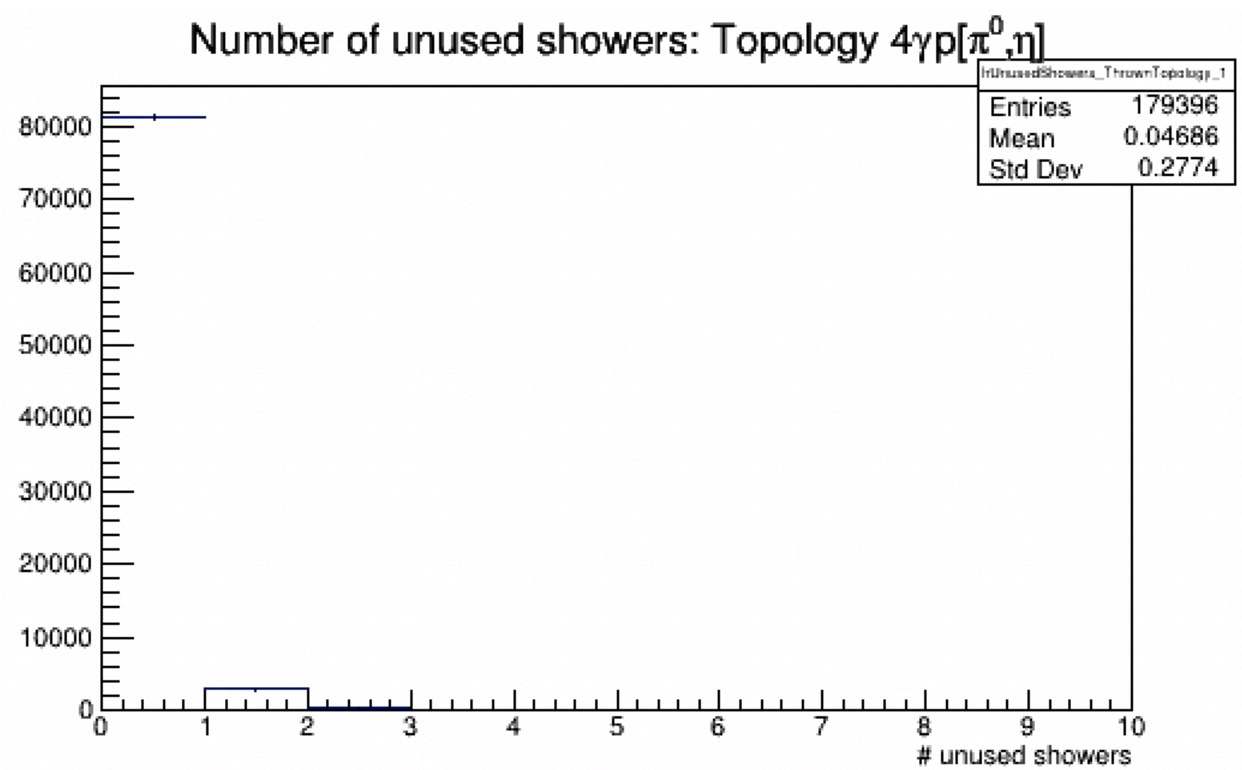


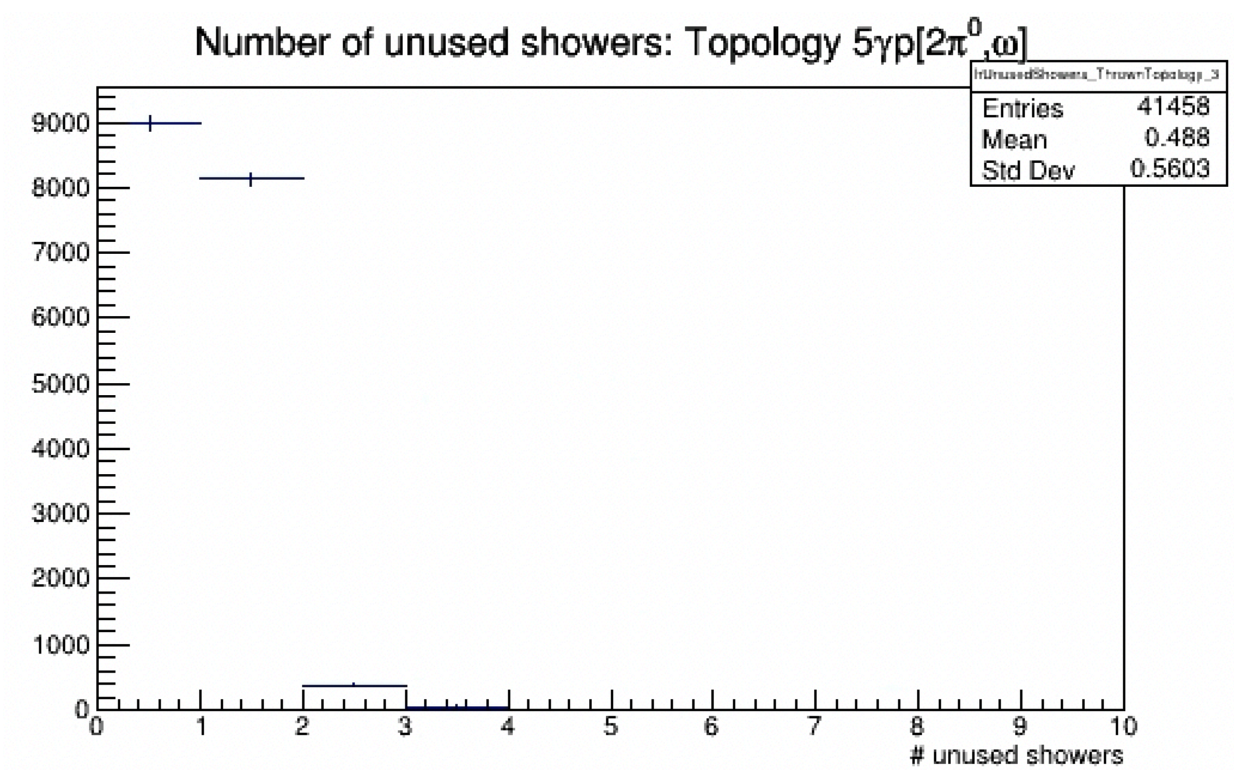


Common background topologies

- Shuffled photons or pi+/pi-
- **Non-exclusive:**
  - (Final state of interest) + ɣ or pi0
  - (Final state of interest) + pi+ pi-
- Mis-identified charged particle (K <— > pi)



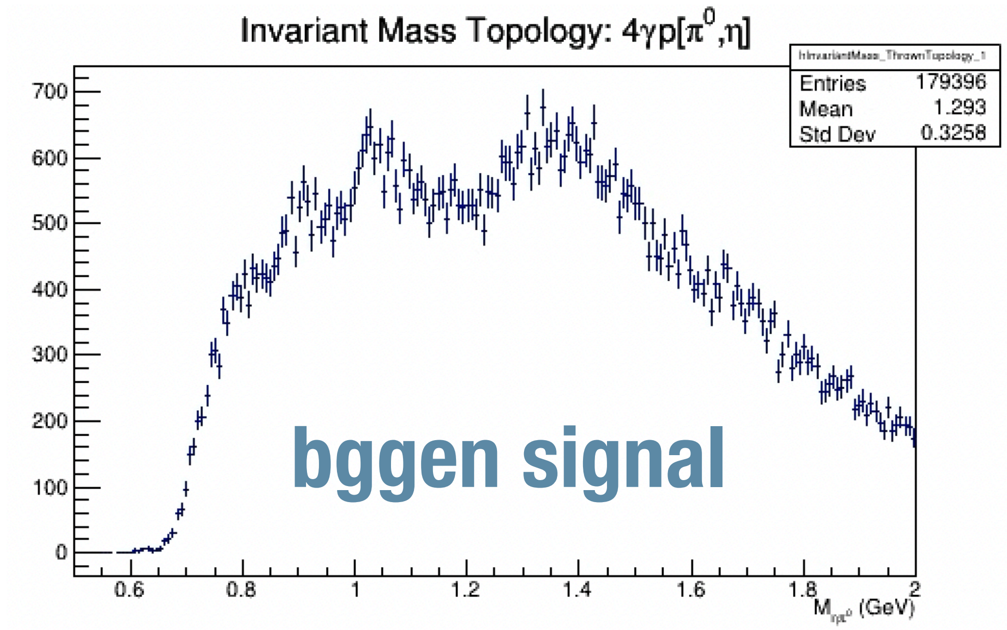


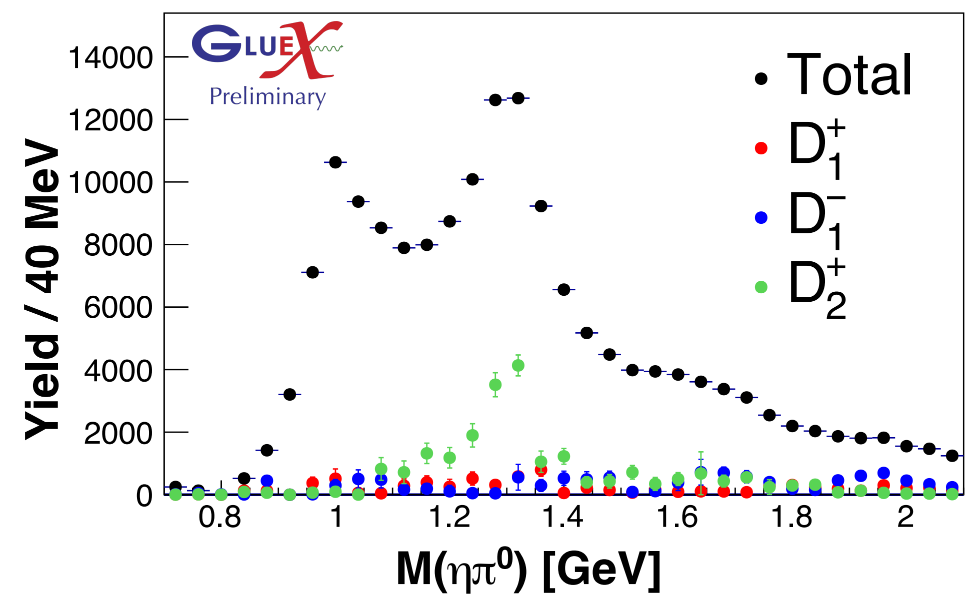


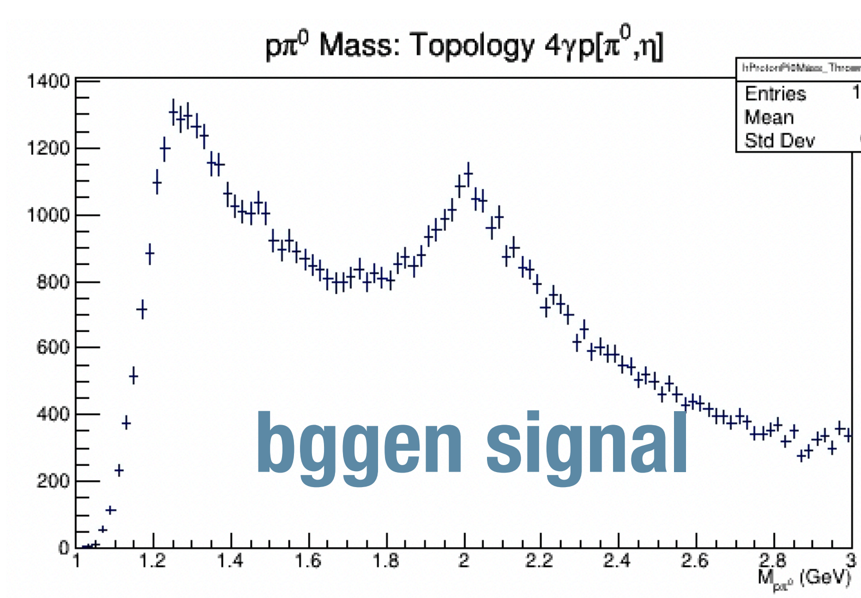
What's might bggen be missing?

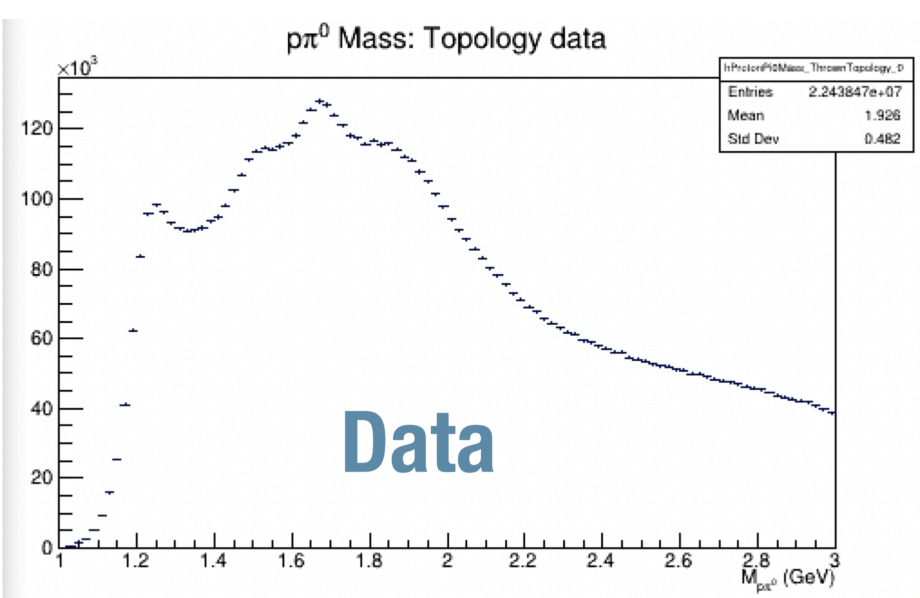
- Meson resonances: a's, b's, f's, h's, etc.

- Baryon excitations:









Simulating specific backgrounds

- Assuming you know what relevant background topologies are, how can you more accurately simulate them?
- Generate 10k events (interactively) and study distributions (gen_amp or genr8?)
    - e.g. gp -> b1 p, b1 -> omega pi0 -> 5g
    - e.g. gp -> Delta+ eta

Simulation of signal process

- Generating specific signal MC instead of using what's in bggen
  - better physics model t-slope, beam energy dependence, etc.
- Efficiencies…
- Mass resolutions…
- Compare kinematic distributions (p vs theta) for each particle…

Subtracting remaining backgrounds

- Basic premise, what are the assumptions…
- Accidental subtraction
- Sideband subtraction
- Etc.

- After subtraction, compare signal MC and data

What type of trees should I use?

- PART format tree (1 entry per event)
    - Output of ReactionFilter, input to DSelector
- Flat tree (1 entry per combo)
- FSRoot (1 entry per combo): see Malte's talk

Some general suggestions

- Reduce dataset footprint whenever possible
  - Write subset of analysis trees with first pass event selection with DSelector or FlattenFSRoot
  - Goal is to iterate and make plots quickly
- Systematic variation: try two analysis options which you expect to give the same result and compare (e.g. 0/90 vs -45/45, different decay modes, etc.)
- Others?

Backup

Thrown Topology in DSelector

- With large bggen MC samples need to tag events by their “thrown topology” to identify background sources

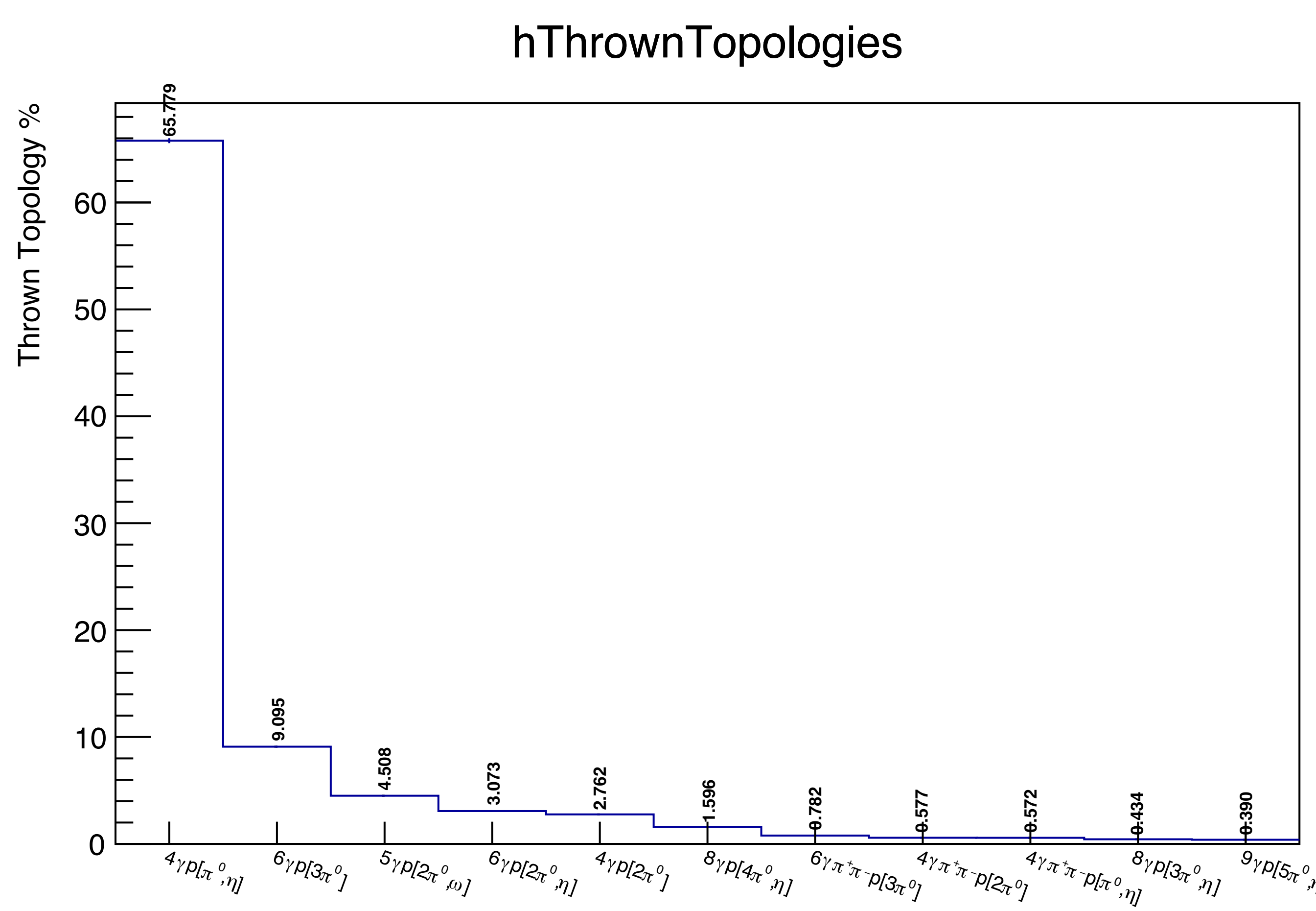
- Information already in Analysis TTree format:

    - Recent update to DSelector library create a unique TString: **NumFinalState[Decaying]**

    - e.g. $\gamma p \rightarrow \pi^0 \eta p$ with $\eta \rightarrow \gamma\gamma$ and $\pi^0 \rightarrow \gamma\gamma$ corresponds to the TString: **$4\gamma p[\pi^0,\eta]$**

Fill histogram with events that pass DSelector cuts corresponding to different “ThrownTopology”

Identify dominant backgrounds topologies



https://github.com/JeffersonLab/hd_utilities/tree/master/AnalysisHowTo/ThrownTopology

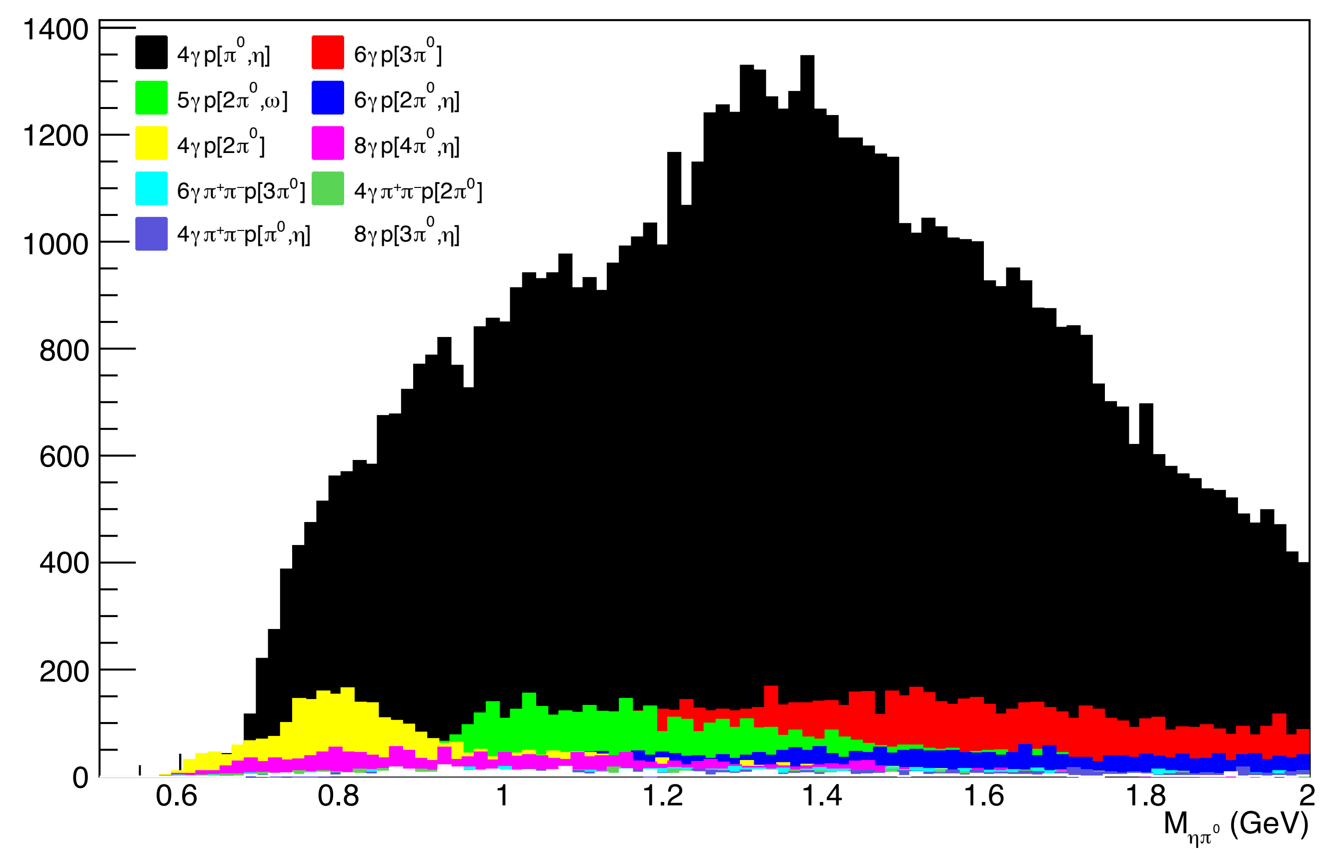
Thrown Topology in DSelector

- With large bggen MC samples need to tag events by their "thrown topology" to identify background sources
- Information already in Analysis TTree format:
  - Recent update to DSelector library create a unique TString: **NumFinalState[Decaying]**
  - e.g. $\gamma p \rightarrow \pi^0 \eta p$ with $\eta \rightarrow \gamma\gamma$ and $\pi^0 \rightarrow \gamma\gamma$ corresponds to the TString: **$4\gamma p[\pi^0,\eta]$**

Once Identified dominant backgrounds topologies, fill invariant mass histogram

Can study how modifying cuts changes topologies and mass dependence

$\eta\pi^0$ Invariant Mass

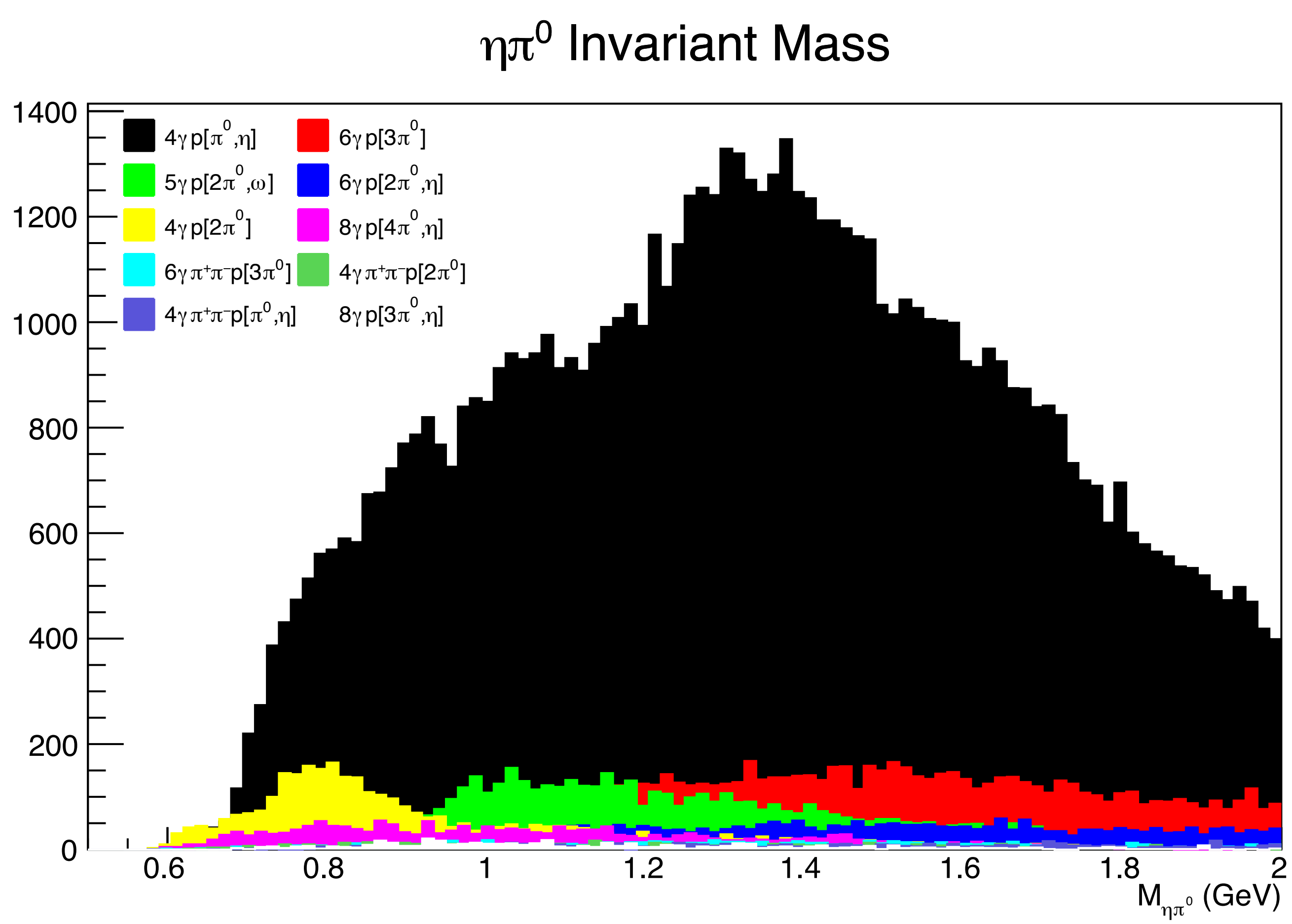


https://github.com/JeffersonLab/hd_utilities/tree/master/AnalysisHowTo/ThrownTopology

Thrown Topology in DSelector

- **Note:** Currently for decaying particles only π^0 keeps how many decayed, for others only flag if ≥ 1 in topology (eg. η, η’, etc.)

- Please try out **Analysis How To** and provide feedback, next step will be to create DHistogramAction to simplify



https://github.com/JeffersonLab/hd_utilities/tree/master/AnalysisHowTo/ThrownTopology