Extending SIMP Search to L1L2

Tom Eichlersmith

he/him/his

University of Minnesota

eichl008@umn.edu

August 6, 2024

Premise

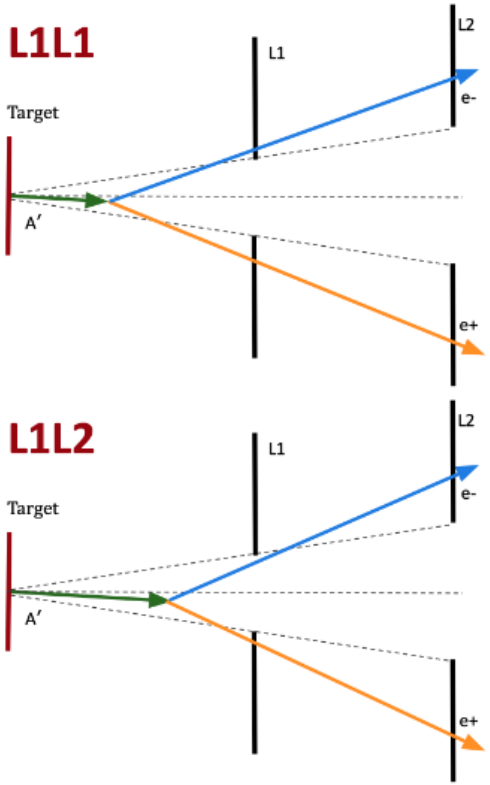


Fig 30 from PRD

Samples

Same data and simulation samples as Alic's SIMP (L1L1) search.

Selections

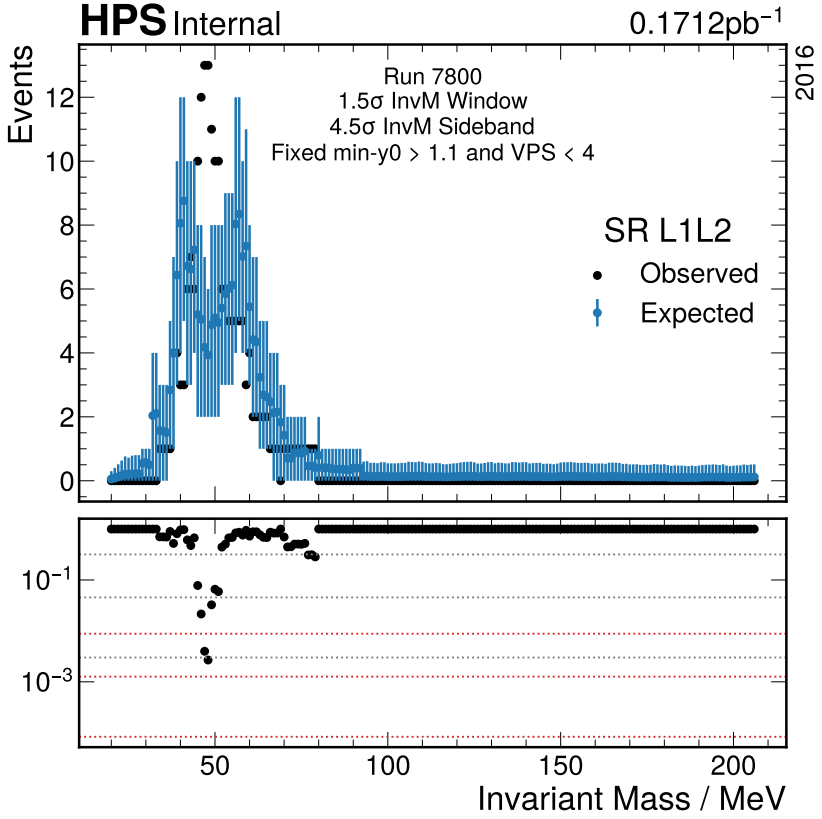
Rely on Alic's thorough study and validation, copy pre-selection and start with same final selection variables.

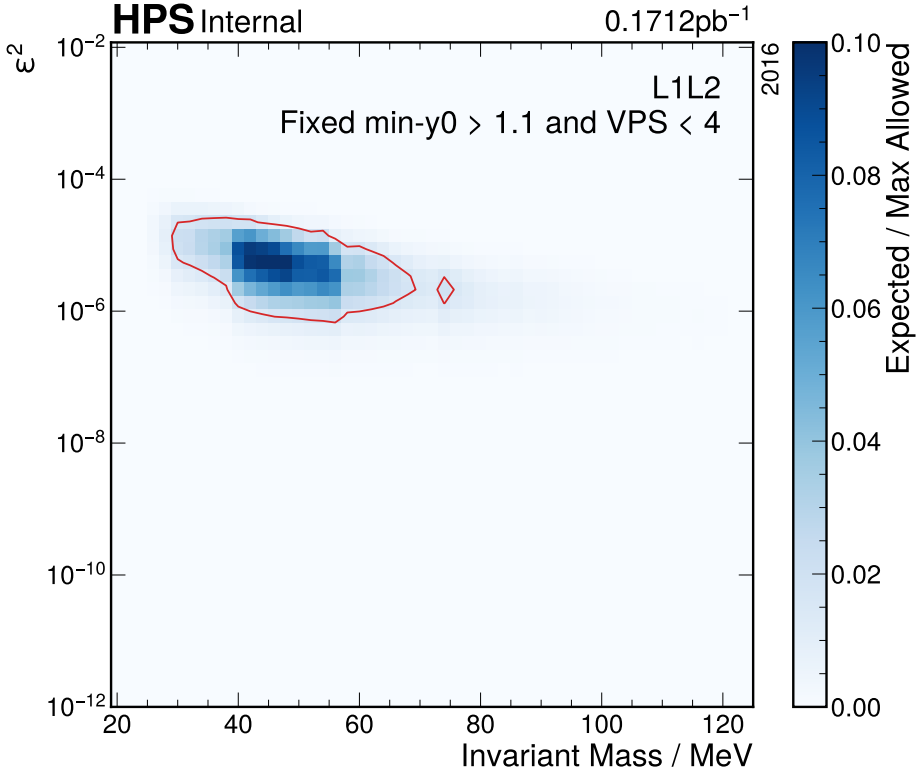
Search and Exclusion

- Search for excess in m_{reco} vs $\min(|y_{0,e^-}|, |y_{0,e^+}|)$ space
- Exclude by using OIM on the z distribution after final selections

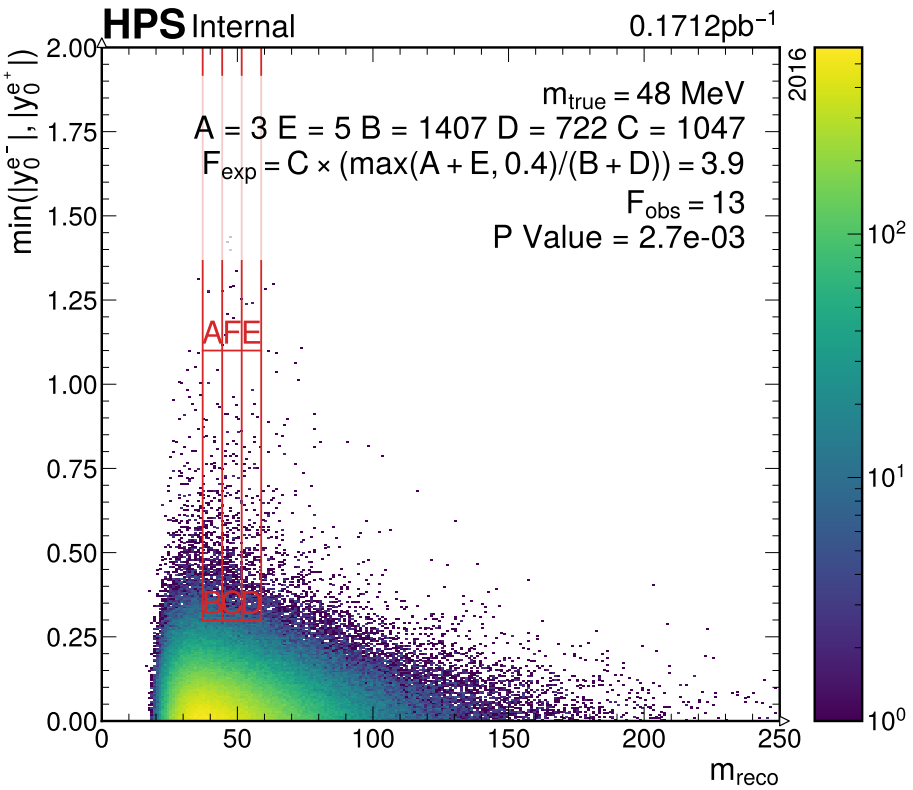
For Context

Using Run 7800 as Test Bed





For Context



1. Fill histogram with data
2. Set mass edges at 1.5σ and 4.5σ (values optimized by Alic)
3. Set upper min-y_0 edge at cut value
4. Lower other min-y_0 edge (a.k.a. y_0 "floor") from the cut value until there are at least 1k events in region C
5. Calculate expected number of events in F and compare to observed number of events
6. Estimate p-value by throwing toy experiments in A+E (Poission), B+D and C (Normal) and re-calculating F from these toys

Punzi Figure of Merit

Basis

Proposed in ▸ PHYSTAT2003 by Giovanni Punzi where a FoM is designed to be maximized while improving *both* search and exclusion potential.

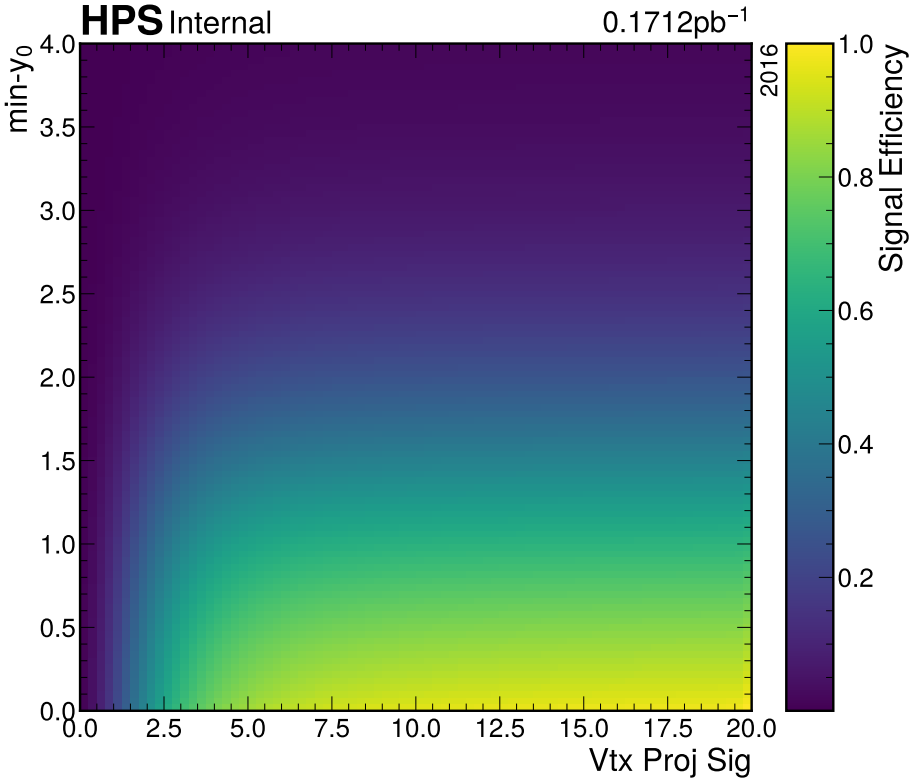
$$f_{\text{punzi}} = \frac{E}{\frac{a}{2} + \sqrt{B}}$$

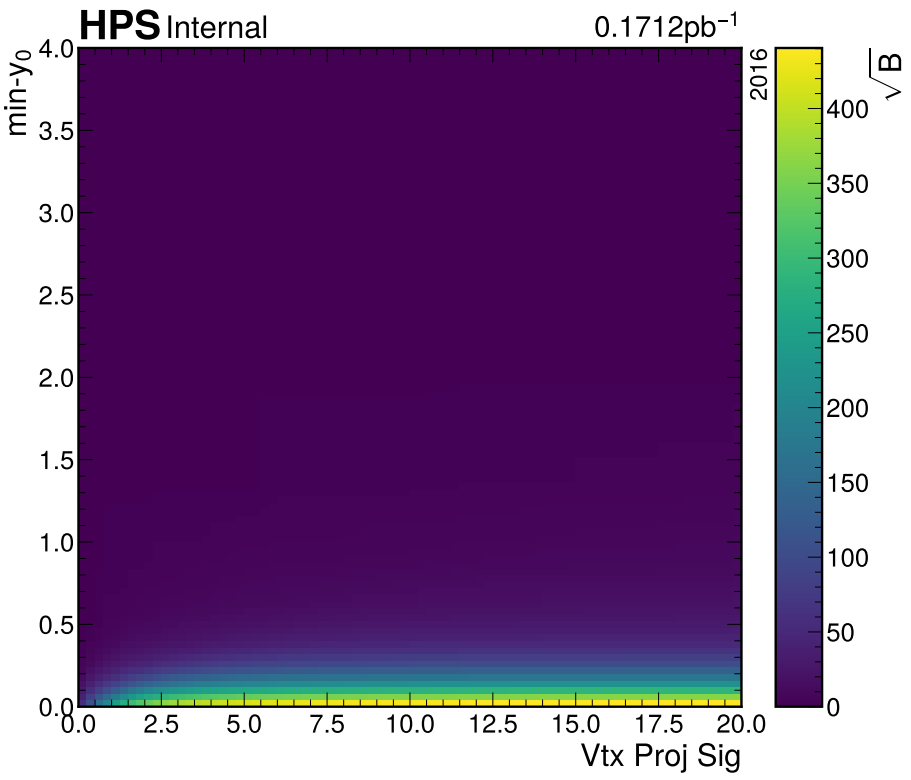
where E is the signal efficiency, B is the background yield, and a is the desired confidence level of search or exclusion (in number of σ, currently using 3).
Two main benefits (from my perspective)

- Does not diverge as $B \to 0$
- Does not require knowledge of absolute rate of signal

Example Calculation

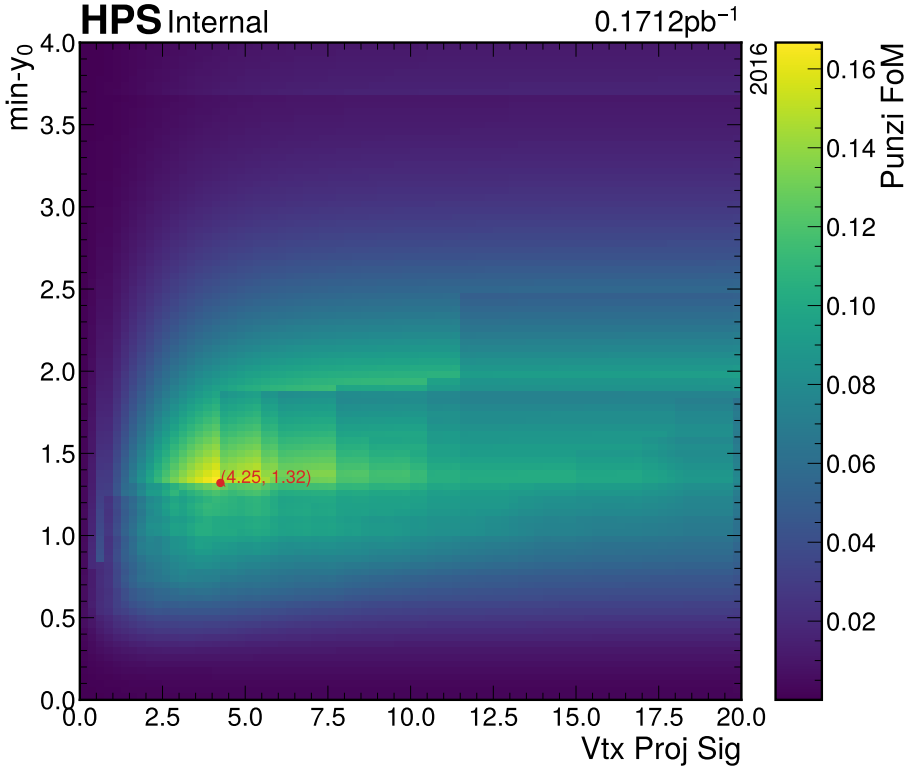
$m_{V_D} = 60$ MeV





Example Calculation

$m_{V_D} = 60$ MeV



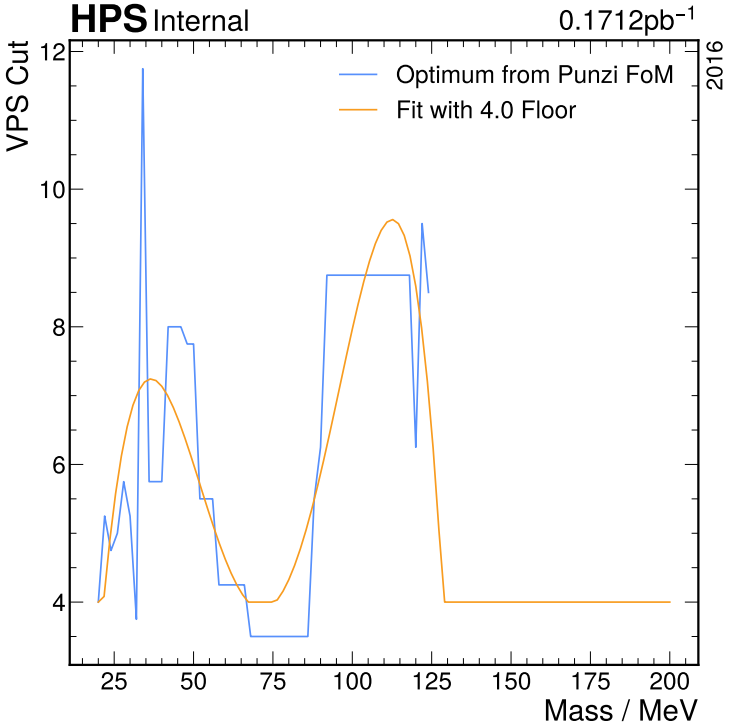
$$f_{\text{punzi}} = \frac{S_{\text{eff}}}{\frac{3}{2} + \sqrt{B}}$$

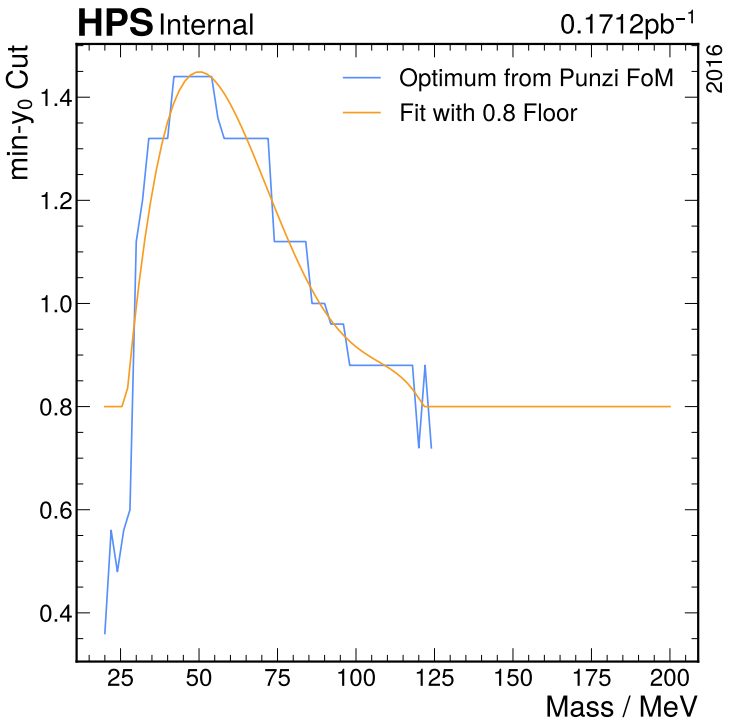
- Choose cuts that maximize f_{punzi}
- Can observe lines originating from single-event bins being crossed
- Seems to be offering reasonable values

All Mass Points

Do this procedure for all signal sample mass points.

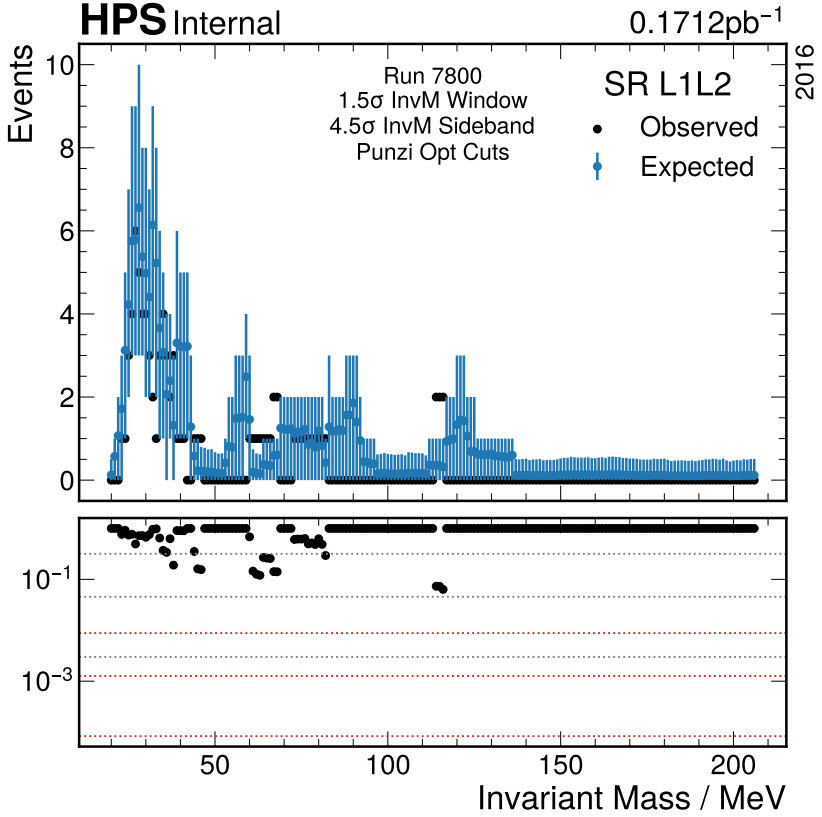
Punzi-Optimized Cuts

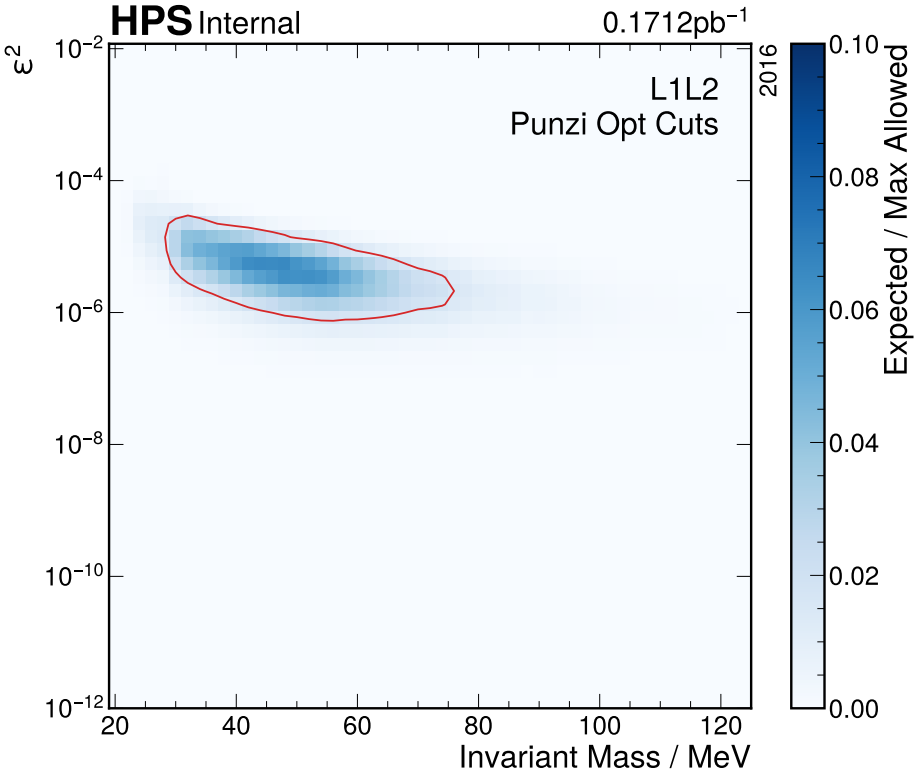




Set "floor" on both cuts to be the cut value outside the fit range.

Reach of Punzi-Optimized Cuts





Future Work

Optimization Strategy

- Punzi FoM does not require choice of ϵ or signal production yield
- L1L1 min-y_0 Cut optimized by maximizing discovery significance at specific ϵ
- Could weight Punzi FoM by decay weighting along z (then need to choose ϵ) or optimize Punzi within each z bin (cuts are functions of z)

Exclusion Estimate

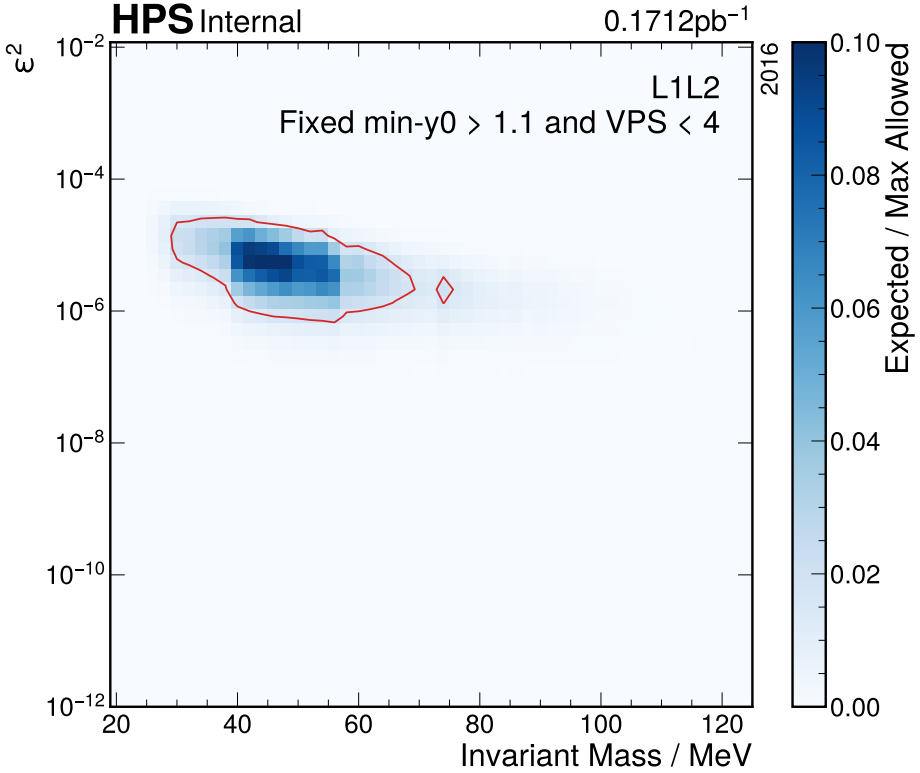
- Alic has been seeing hints that drawing the exclusion estimate at 0.1 in Expected / Allowed for 10% is overly optimistic

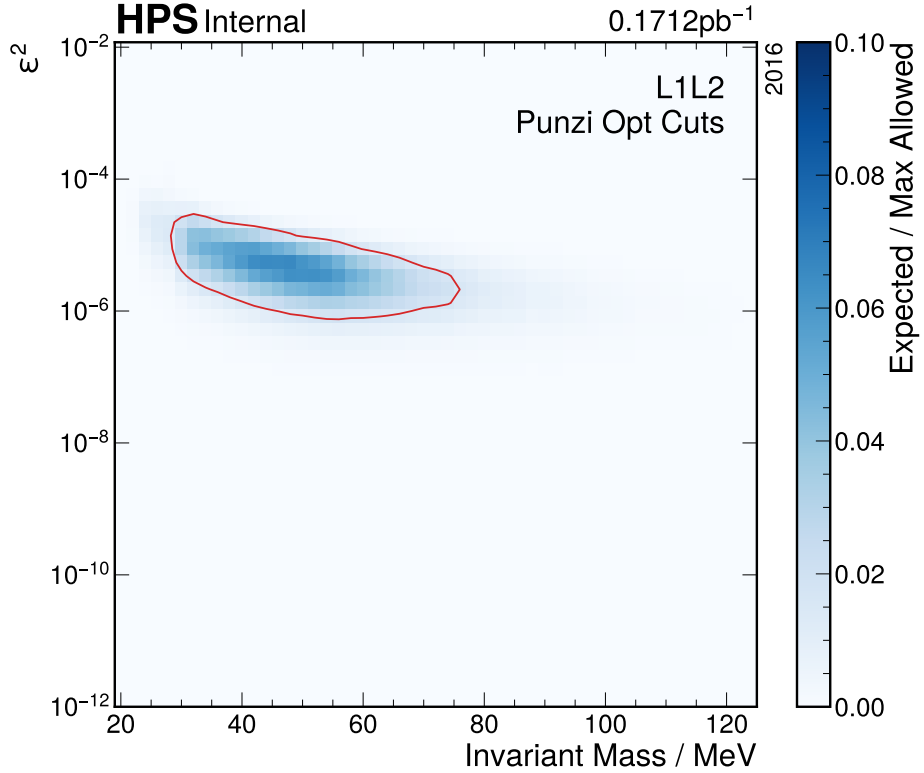
Additional Material

- Look at $|y_0|$ separated by track (L1 or L2) to see if major differences
- Various distributions (mass vs z, z vs min-y0) as selections are made

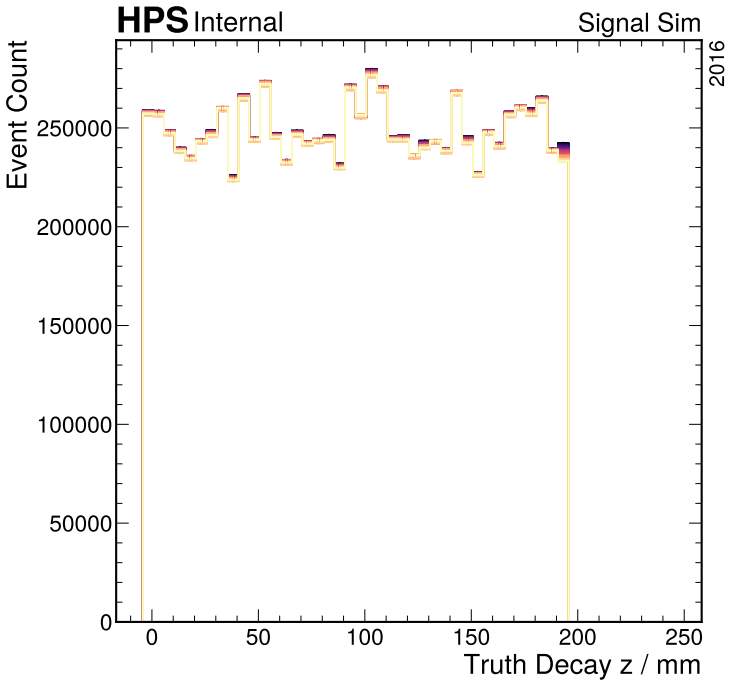
Questions

Comparison of Reaches





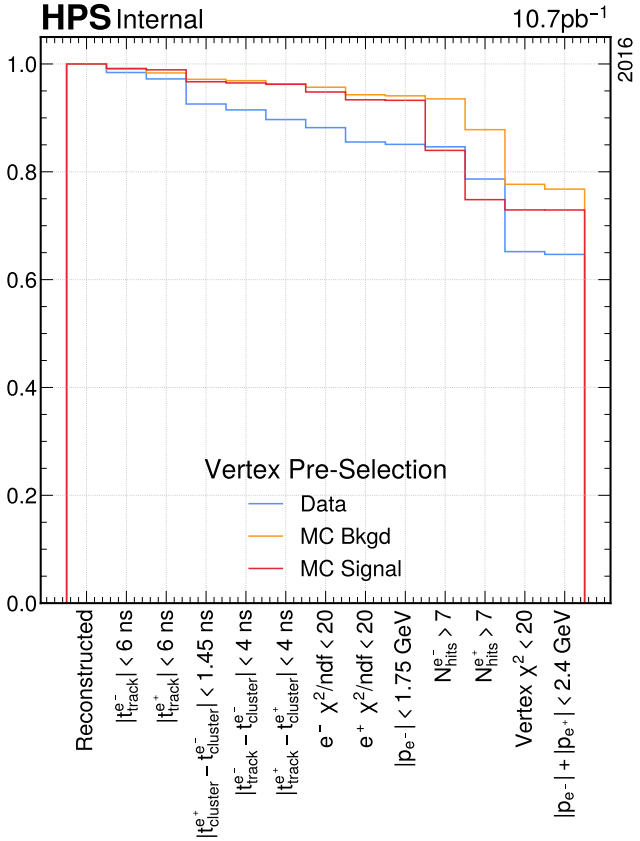
Displacement of SIMP Signal Events



One of the first things I did to make sure samples didn't need to be created.

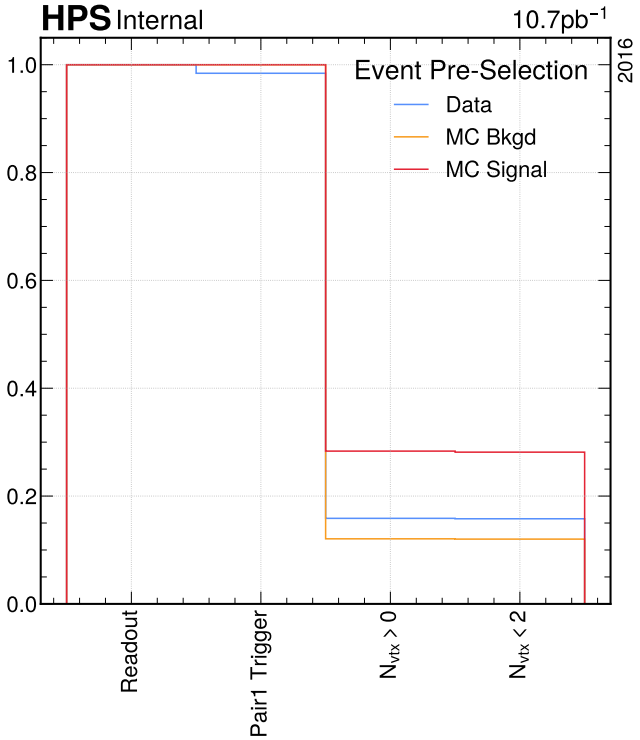
- Different colors correspond to different mass points
- Truth-level decay vertices sampled out until ~ 200 mm
- Close to the same z position as L1, so I think these samples can be faithfully used to study the L1L2 selection
- Idential distribution across colors makes me think that the random seed determining the decay length was not changed, but I think that is okay since we normalize by this distribution during exclusion estimates anyways

Pre-Selection on Vertices



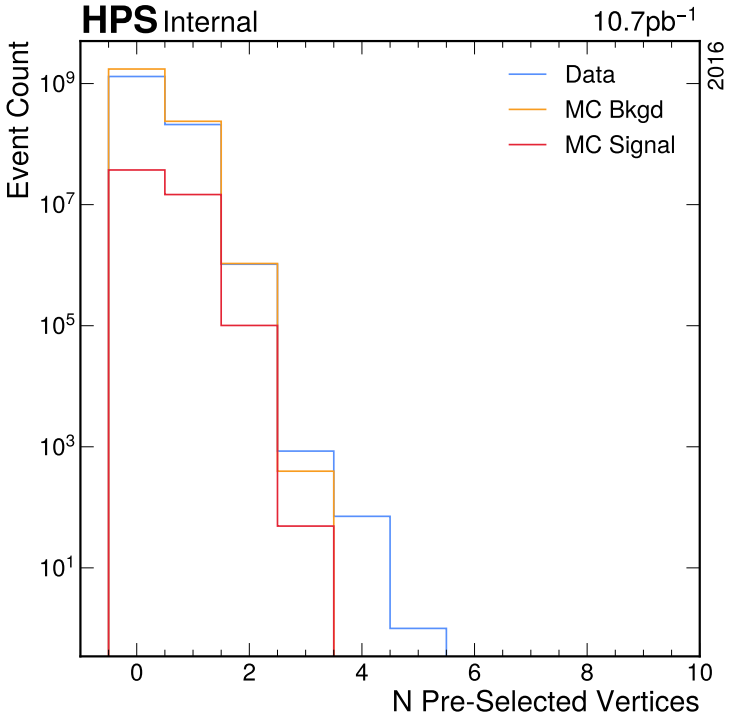
- Same Pre-Selection on vertices as developed and validated by Alic
- Seeing same efficiencies as documented within Alic's SIMP (L1L1) note

Pre-Selection on Events



- Similar to first stage of Alic's event selection, although dropping reconstruction category requirement
- Largest effect is requiring at least one pre-selected vertex

Number of Pre-Selected Vertices per Event



Event Pre-Selection basically amounts to choosing the events falling into the $N = 1$ bin.
Data is the only sample which has the additional requirement of the Pair1Trigger which has a small effect.

Experiments with Decay Weighting

First idea is to simply define a new FoM that includes the decay weighting function.

$$f_{\mathrm{DW}} = \frac{1}{\frac{a}{2} + \sqrt{B(t)}} \int_{z_{\mathrm{target}}}^{\infty} D(z)E(z,t)dz$$

where

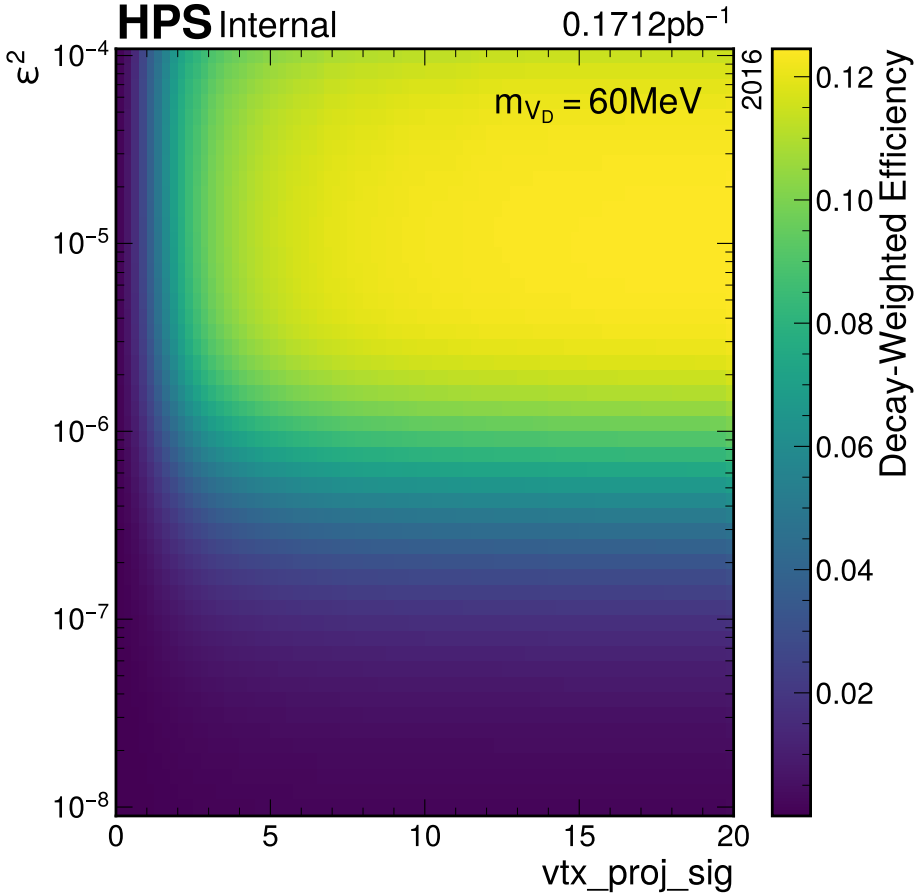
$$D(z) = \sum_{V \in \{\rho_D, \phi_D\}} \mathrm{BR}(A' \to V\pi_D) \frac{\exp((z_{\mathrm{target}} - z)/(\gamma c\tau_V))}{\gamma c\tau_V}$$

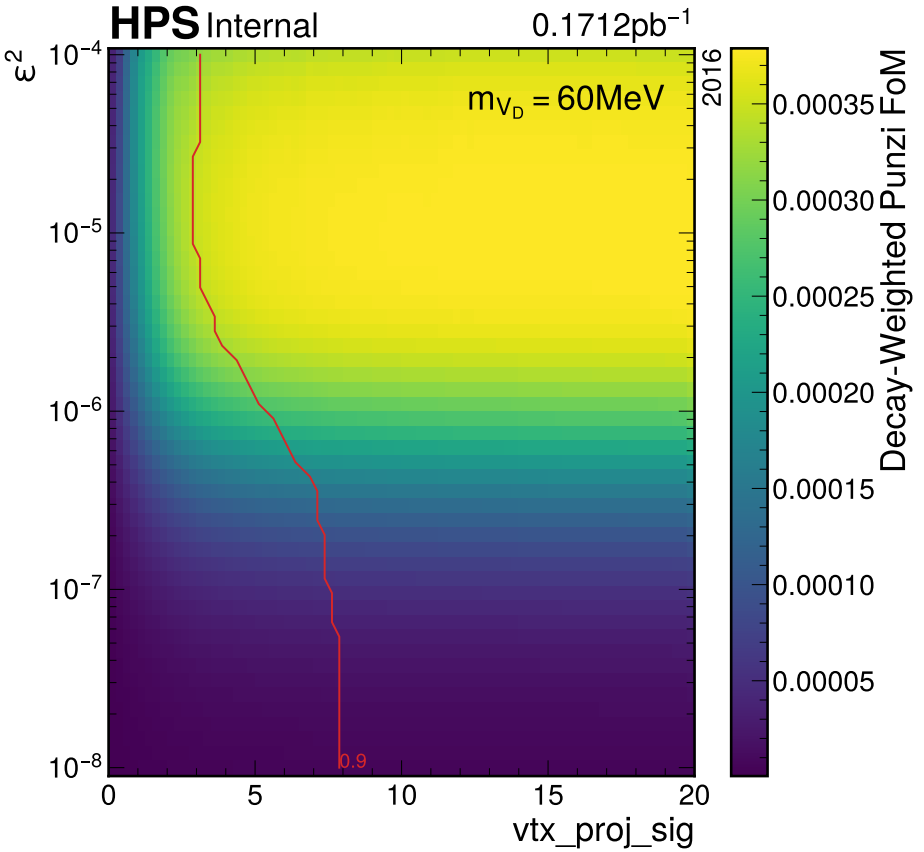
This becomes equivalent to f_{punzi} in the $\epsilon \to 0$ limit where $D(z)$ becomes flat and the events are equally weighted along z. Calling this "Decay-Weighted Punzi FoM" and the integral "Decay-Weighted Efficiency".

Issues

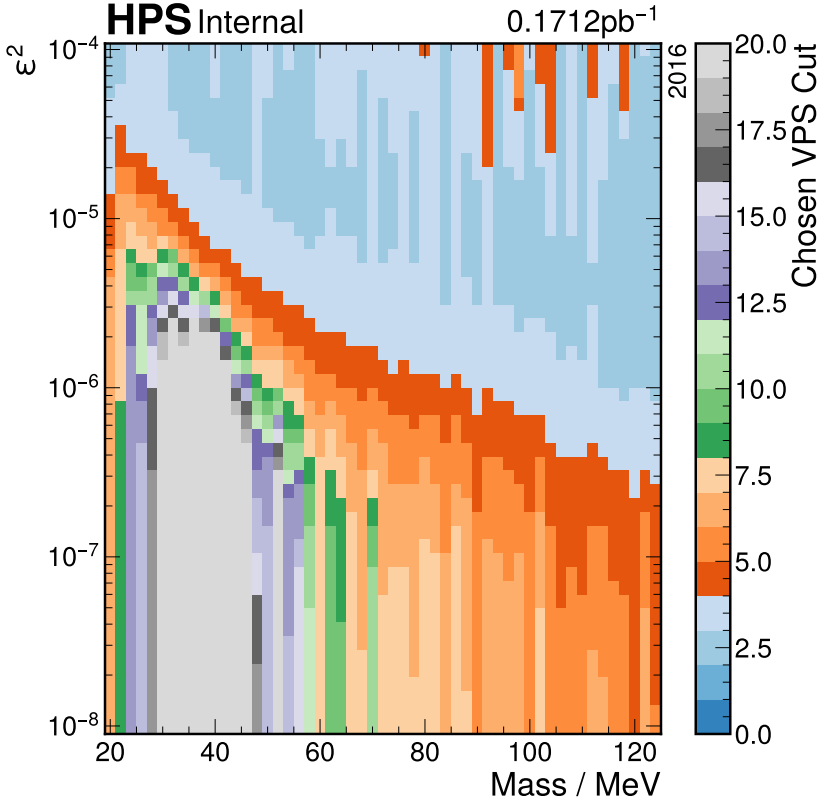
- May have to choose an ϵ^2 value to optimize for
- Pure maximum is often attained by removing the cut $\to$ look for where the FoM "flattens" out (i.e. tigtening the cut does not improve the FoM much anymore) $\to$ choose cut that is the tightest cut getting to 90% of the maximum

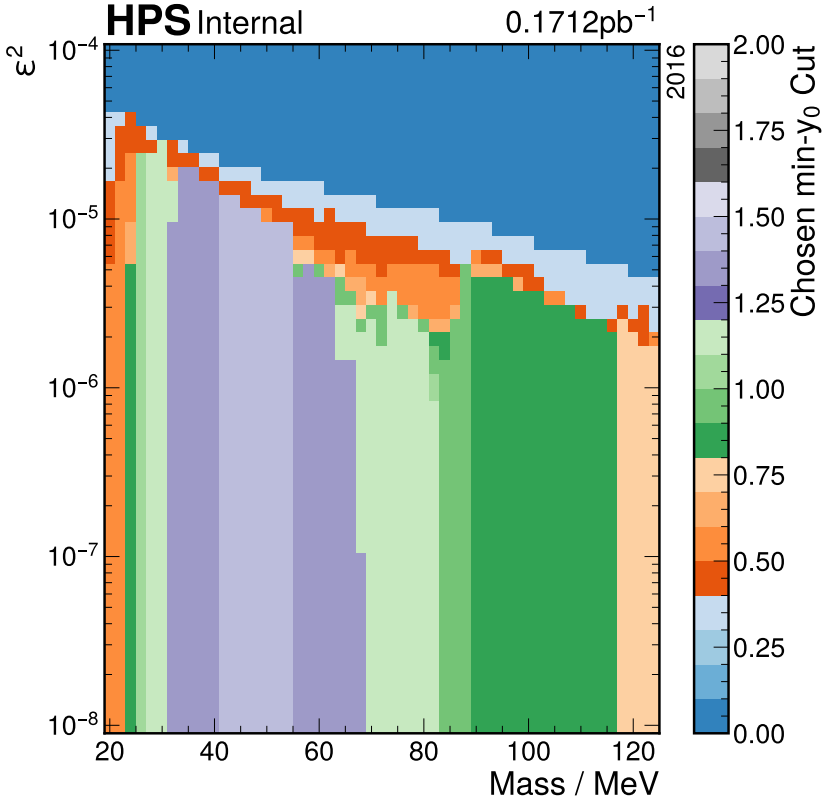
Example Decay-Weighted Punzi Calculation





Cut Choices by ϵ^2 and m_{V_D}





Chose $\mathrm{VPS} < 4$ and then applied it to optimize min-y_0.