A’ Analysis Update

Elizabeth Berzin
Collaboration Meeting
October 20, 2026

U.S. DEPARTMENT OF ENERGY
Office of Science
Stanford University
SLAC NATIONAL ACCELERATOR LABORATORY

Outline

- A’ analysis is challenging—very low signal counts, requires a zero-background approach.
- This talk describes some preliminary work exploring new approaches for removing backgrounds:
  - Data-driven “vertex quality score” applied on top of existing tight selection.
  - Initial attempts at a signal/background discriminator combining BH/WAB, vertex quality, and displacement features.
- All results use pass5_v9 1% data.

Current Tight Selection Strategy

- Current tight-selection approach (Sarah): grid search over selected variables, optimizing the following significance metric:

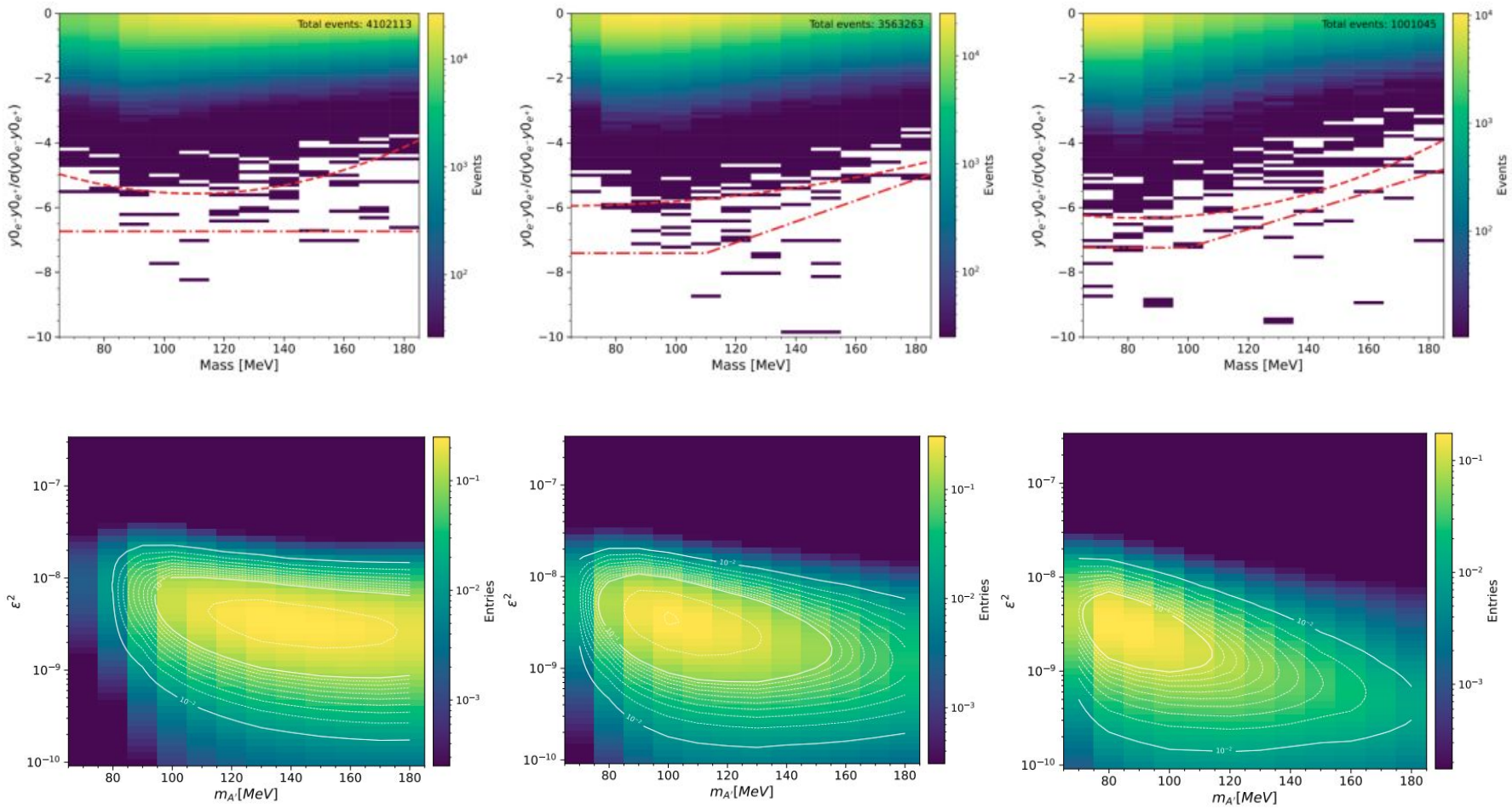
$$Z(N_{\text{sig}}, N_{\text{bkg}}, \sigma^2_{\text{bkg}}) = \sqrt{2N_{\text{tot}} \ln\left(\frac{N_{\text{tot}}(N_{\text{bkg}} + \sigma^2_{\text{bkg}})}{N^2_{\text{bkg}} + N_{\text{tot}}\sigma^2_{\text{bkg}}}\right) - \frac{N^2_{\text{bkg}}}{\sigma^2_{\text{bkg}}} \ln\left(1 + \frac{\sigma^2_{\text{bkg}} N_{\text{sig}}}{N_{\text{bkg}}(N_{\text{bkg}} + \sigma^2_{\text{bkg}})}\right)}$$

variable	L1L1 initial	L1L1 refined	L1L2 initial	L1L2 refined	L2L2 initial	L2L2 refined
$\theta_{\text{recoil}} > \theta^{\text{cut}}_{\text{recoil}}$	300 mrad	325 mrad	300 mrad	375 mrad	300 mrad	350 mrad
$p_{\text{vtx}} > p^{\text{cut}}_{\text{vtx}}$	2.95 GeV	3.0 GeV	3.15 GeV	3.15 GeV	3.1 GeV	3.15 GeV
$p^{e^-}_z / p^{e^+}_z < (p^{e^-}_z / p^{e^+}_z)^{\text{cut}}$	2.0	2.0	1.75	1.75	1.75	1.75
$N\sigma_{x,\text{vtx}} < (N\sigma_{x,\text{vtx}})^{\text{cut}}$	1.25	1.0	1.25	1.0	1.25	1.5
$N\sigma_{y,\text{vtx}} < (N\sigma_{y,\text{vtx}})^{\text{cut}}$	1.5	0.75	2.0	1.25	2.0	1.5
fixed cuts	$\lvert m_{\text{vtx}} - 120\,\text{MeV}\rvert < 1.7\sigma_m(120\,\text{MeV})$, $z_{\text{vtx}} > z_{\text{target}}$, $\frac{y^{e^-}_0 y^{e^+}_0}{\sigma(y^{e^-}_0 y^{e^+}_0)} < -2$					

Table 46: Cut values determined and used as fixed cuts in the first and second iteration of cut optimization. All listed cut values are given for the $m_{A'} = 120$ MeV mass window.

Current Tight Selection Strategy

- Unblinded results on 10% show low signal yields in the presence of non-zero backgrounds.

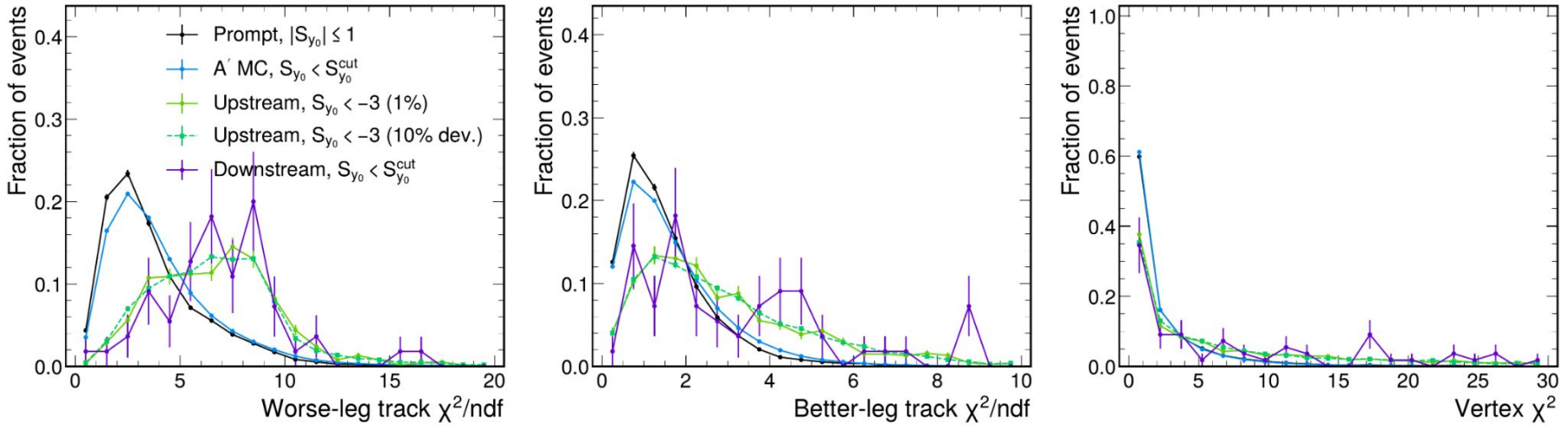


L1L1 L1L2 L2L2

Part 1: Vertex Quality Score

Vertex Quality Score

- Idea: Use data to train a classifier to distinguish between “good” and “bad” (mis-reconstructed) vertices, without significant dependence on MC mismodeling.
  - “Bad” vertices: Vertices reconstructed upstream of the target (z_vtx < −1.1 mm), with y0p < −3, after standard tight selection kinematic cuts, in mass search region (~70–180 MeV).
  - “Good”: Downstream vertices with |y0p| ≤ 1, after standard tight selection kinematic cuts, with masses ~70–180 MeV.
  - Training for each hit category (L1L1, L1L2, L2L2) is done independently:
    - At 1% there are 420 / 500 / 216 bad events and 6168 / 5130 / 1263 stored good events.

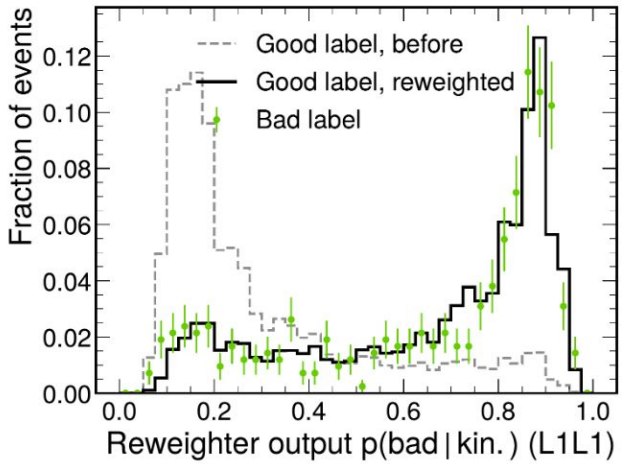


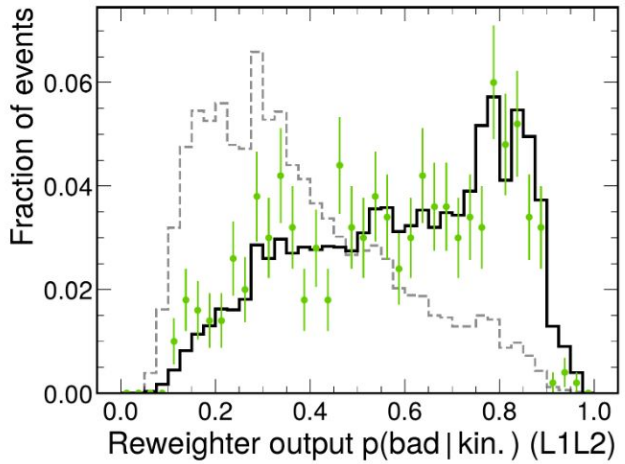
Training Scheme

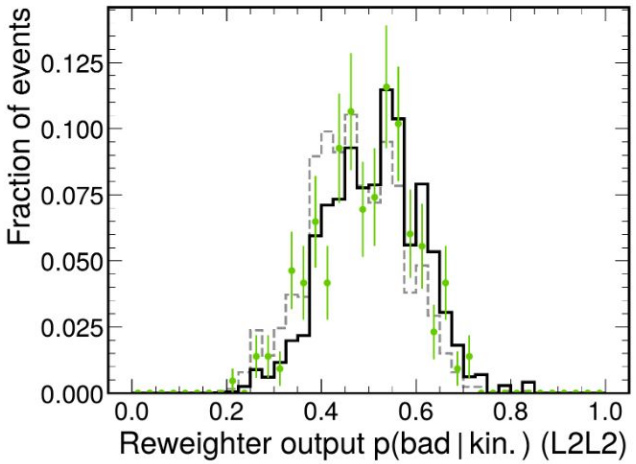
- Given the small number of events available in the 1% sample, models are evaluated through a **5-fold validation scheme**: total of 3 (hit category) x 5 (folds) = 15 models
- Each run is randomly assigned to a fold, balanced such that each fold has the same number of “bad” events.
  - For fold k, the model trains on the runs of the other four folds.
  - A random quarter of those training runs (run-grouped) is held aside as a validation set, used for stopping criteria.
- Model k scores only fold k's events—every 1% event therefore gets exactly one out-of-fold score.
  - Runs are not split between folds, meaning that the held-out scores for each run come from a model that never saw that run in its training. We know that data has some run-dependent reconstruction quality features, this should help prevent learning of run-based features.
- Signal MC and tritrig + WAB MC are scored by all five folds to evaluate fold stability—**no MC in training.**
  - Removes dependence on MC modelling.

Kinematic Reweighting

- While this upstream “bad” population appears representative of the displaced tail in reconstruction quality variables, we do not want to rely too much on mis-reconstructed kinematic features.
- To limit kinematic dependence, events in the “good” class are reweighted based on the output of small BDT trained to separate “good” from “bad” vertices using only kinematic variables: invM, psum, pratio, p(e−), p(e+), recoil_theta
  - Classifier outputs a probability p that a vertex is “bad”, given only kinematic features.
  - The weights of are then normalized such that each class sums to the number of bad events.

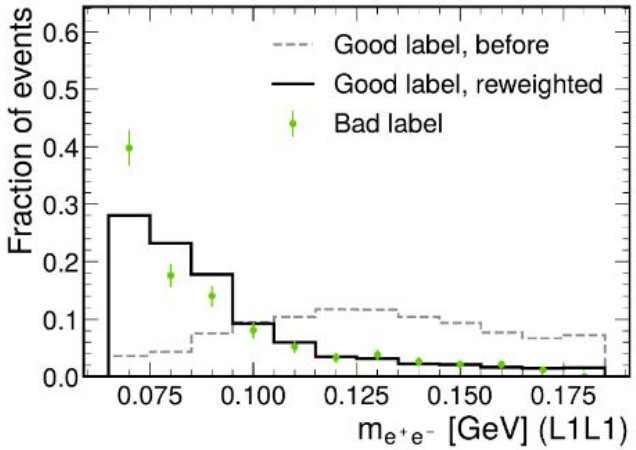


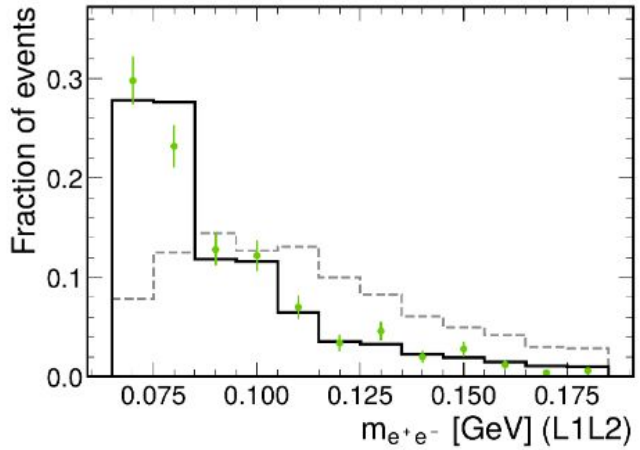


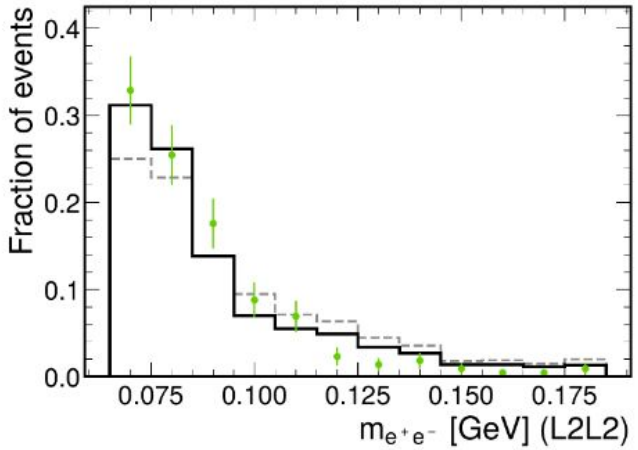


Kinematic Reweighting

- While this upstream “bad” population appears representative of the displaced tail in reconstruction quality variables, we do not want to rely too much on mis-reconstructed kinematic features.
- To limit kinematic dependence, events in the “good” class are reweighted based on the output of small BDT trained to separate “good” from “bad” vertices using only kinematic variables: invM, psum, pratio, p(e−), p(e+), recoil_theta
  - Classifier outputs a probability p that a vertex is “bad”, given only kinematic features.
  - The weights of are then normalized such that each class sums to the number of bad events.





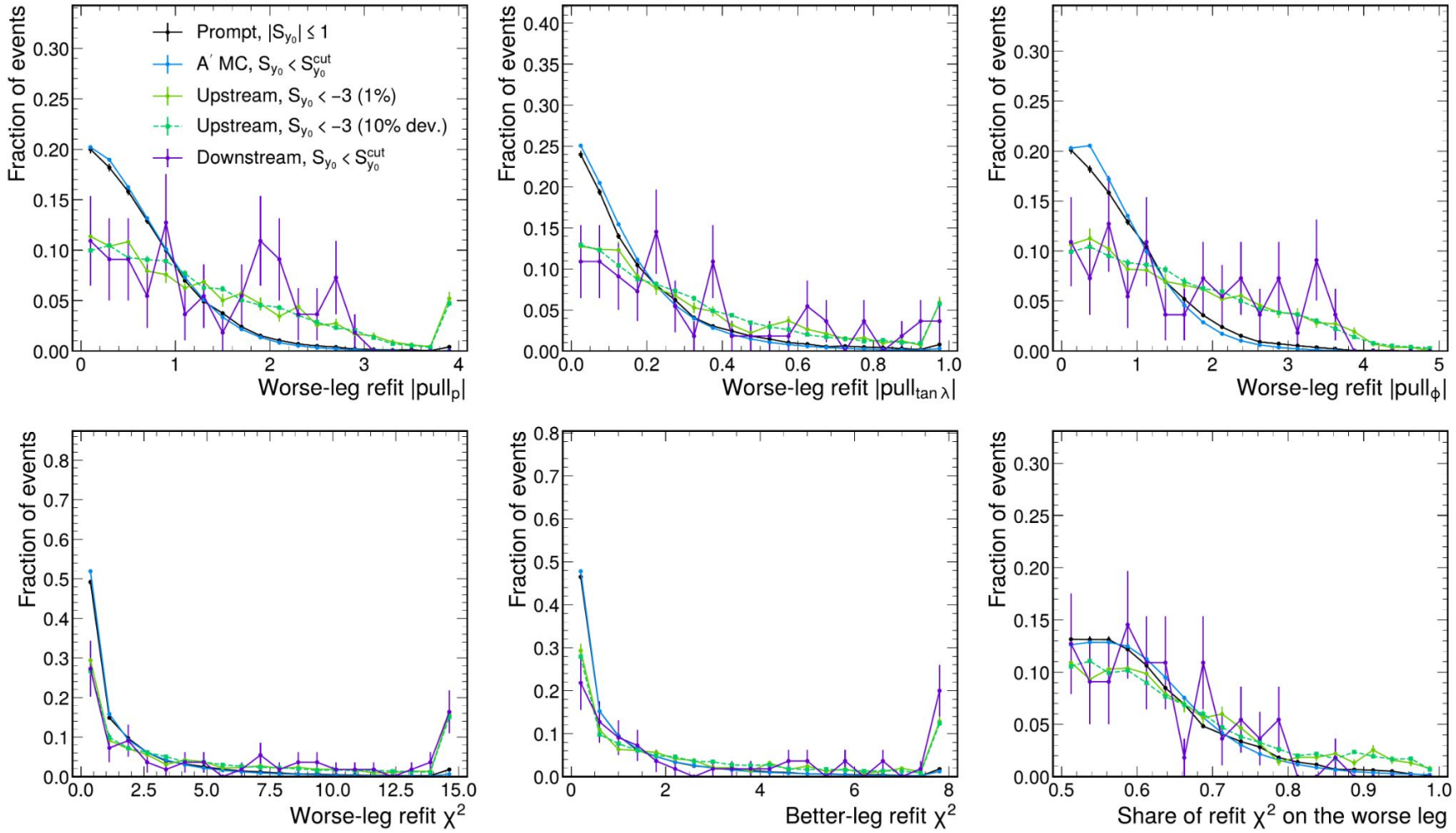


Vertex Quality Score Features

- Trained a BDT on reconstruction quality-features:
  - Track and vertex fit quality: **max_chi2ndf, min_chi2ndf, vertex_chi2**
  - Vertex refit pulls: **max_abs_refit_pull_p**, **max_abs_refit_pull_tanl**, **max_abs_refit_pull_phi**, **max_refit_chi2, min_refit_chi2**
    - Measured how much track parameters change after vertex fit.
  - Interior holes: **ele_interior_holes**, **pos_interior_hole**
  - Timing: **abs_dt_tracks**, **abs_dt_pos_trk_clu**, **abs_dt_ele_trk_pos_clu**
  - Track-cluster matching: **match_dx, match_dy**
- Constraint: the score can only increase with every χ^2, pull, $|\Delta t|$ and hole feature.
- In the following slides thresholds are chosen to preserve 90% signal efficiency.

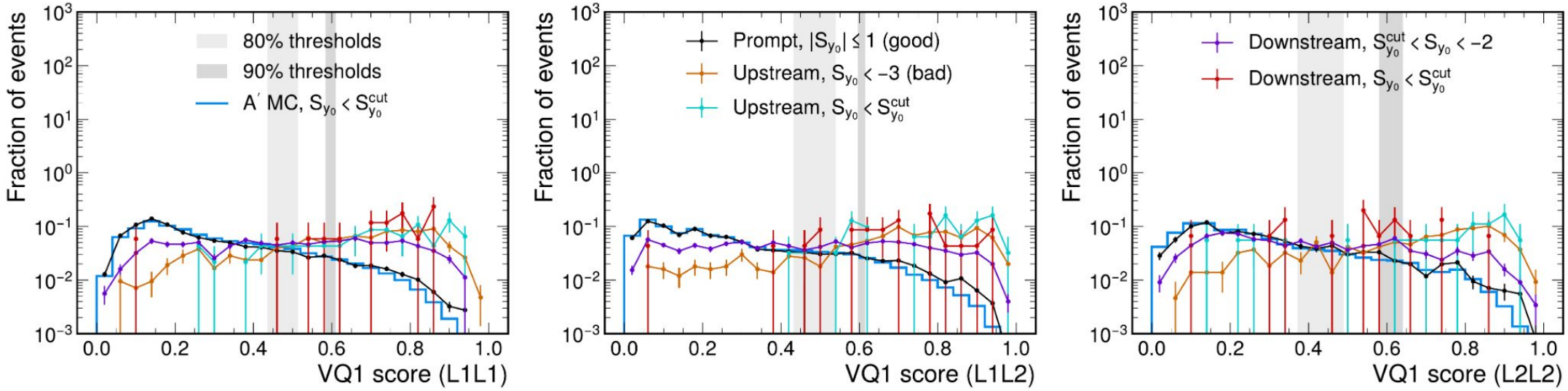
Vertex Quality Score Features

- Upstream tail has similar reconstruction-quality features as downstream tail.



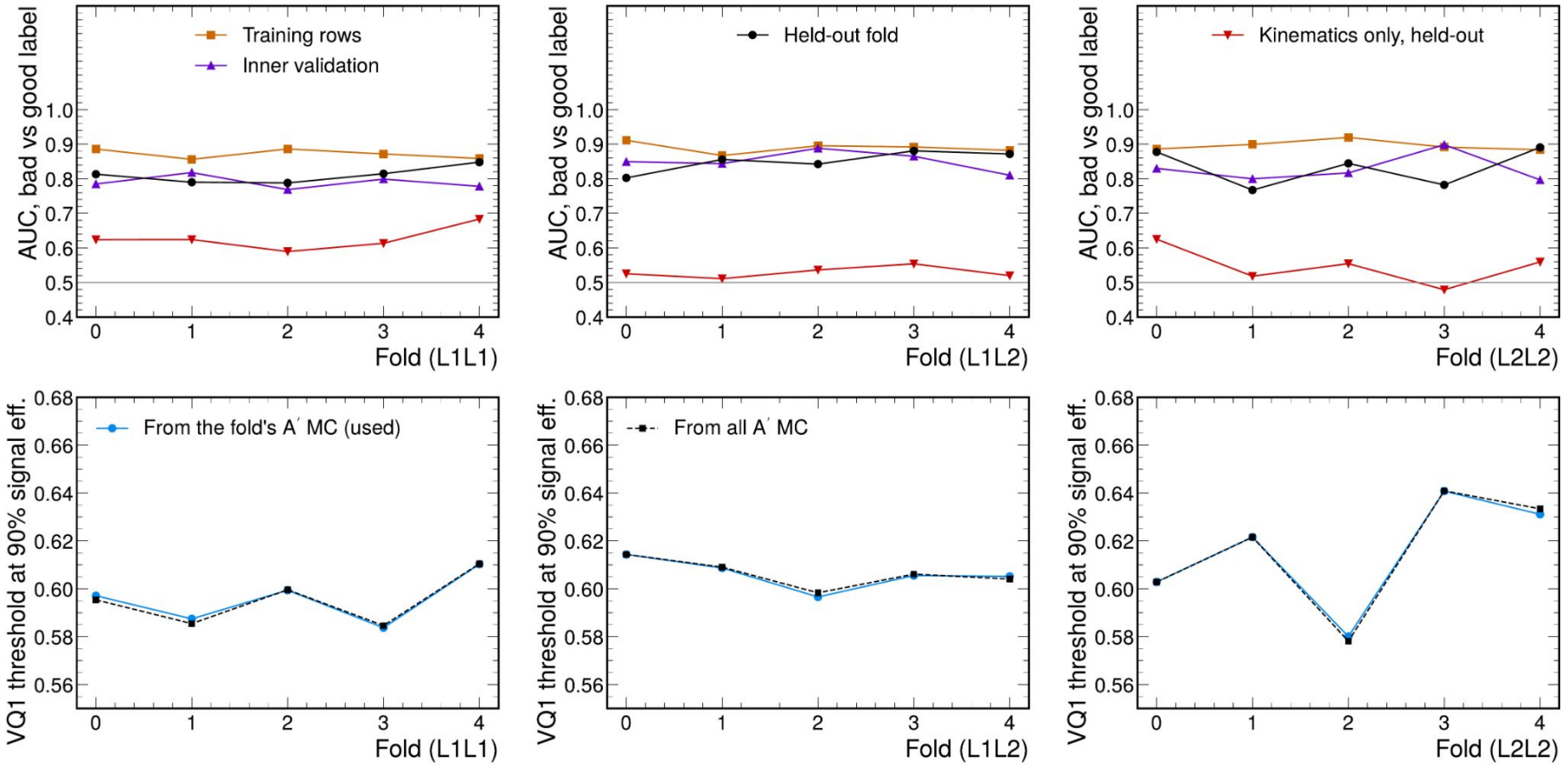
Score Distributions

- **Higher score = worse vertex**
- Model score A’ MC and the prompt code very similarly, despite never having seen MC in training, in all hit categories.
- Mirrored upstream/downstream distributions also score very similarly.
- Intermediate downstream distribution (y0p < -2 but > tight cut), scores intermediately between prompt core and displaced tails.



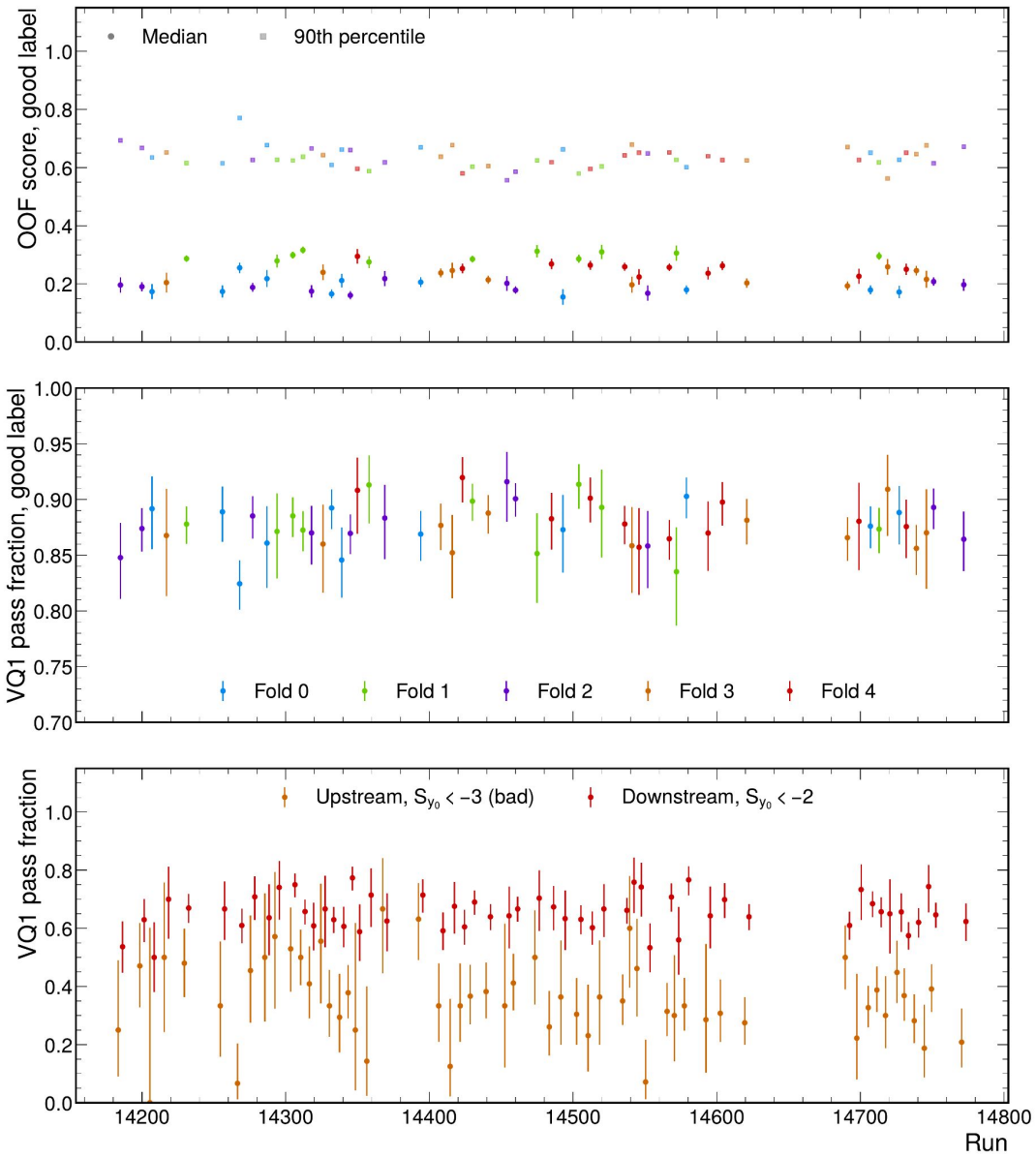
Fold Stability

- Performance is consistent between folds, with more variation in L2L2.
- L1L1 category retains some kinematic dependence.



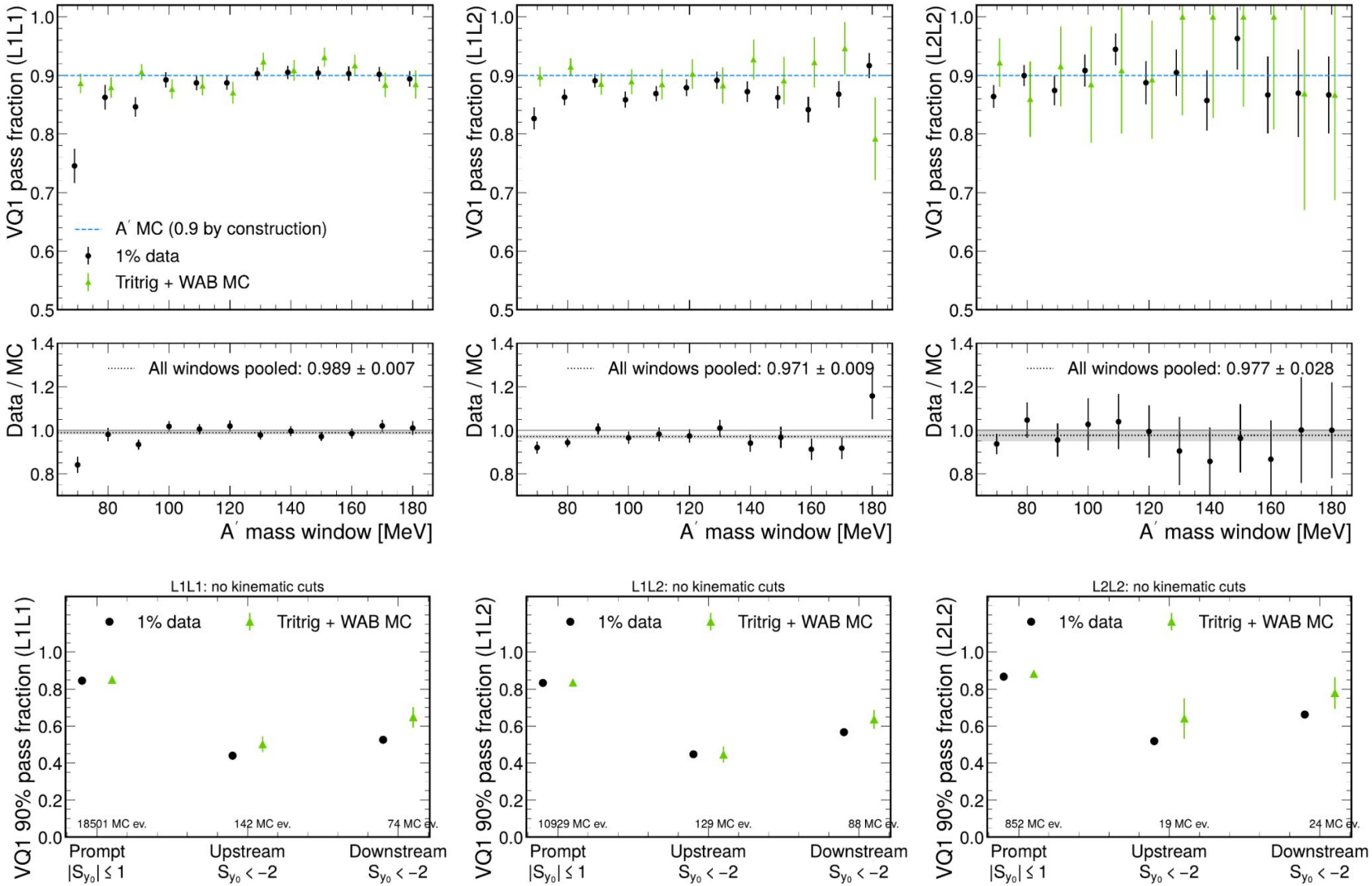
Score Run Stability

- Scores are stable over the run period, among both “good” and “bad” vertices.



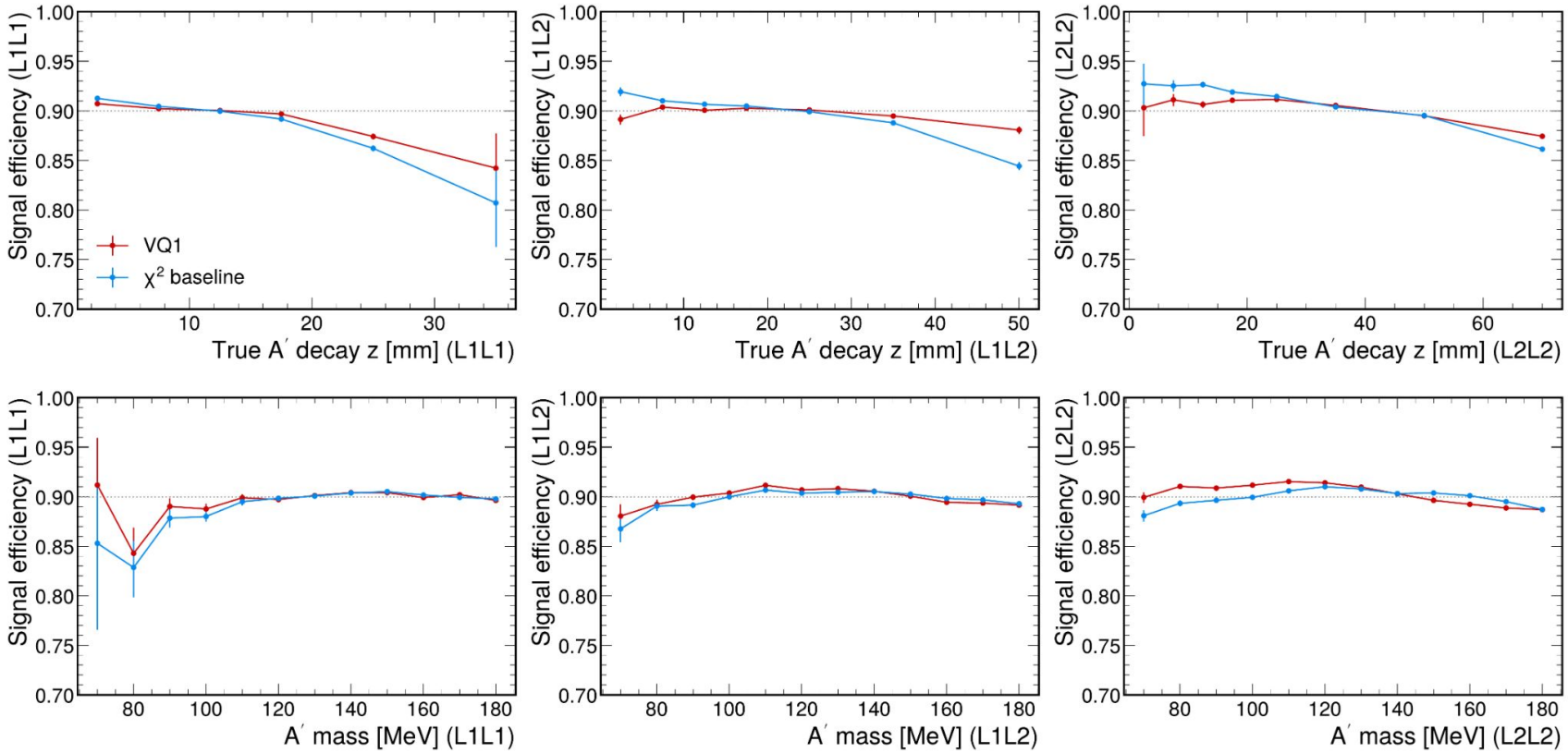
Data/MC Performance

- Performance on tritrig + WAB MC are in agreement with prompt data.
- MC efficiency is slightly higher in the tails.



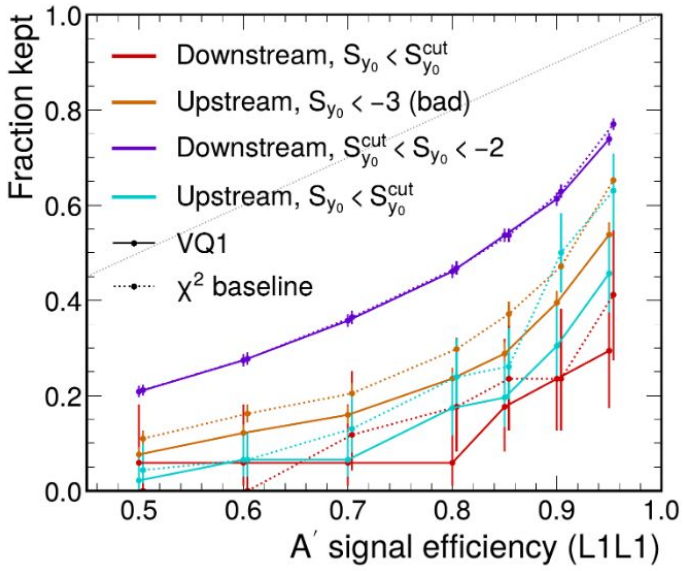
Signal Efficiencies

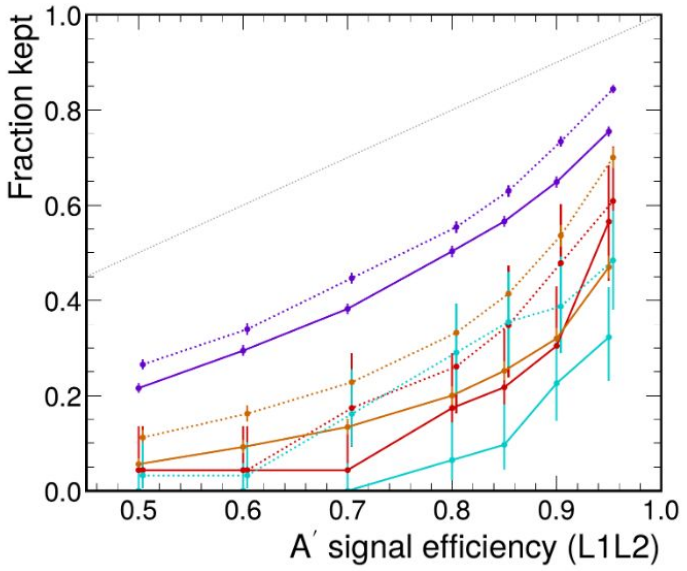
- Thresholds are chosen to have 90% signal efficiency, inclusively.
- Efficiencies are stable with respect to mass, drop by ~5% for very displaced vertices.
- Better stability than a simple χ^2 cut.

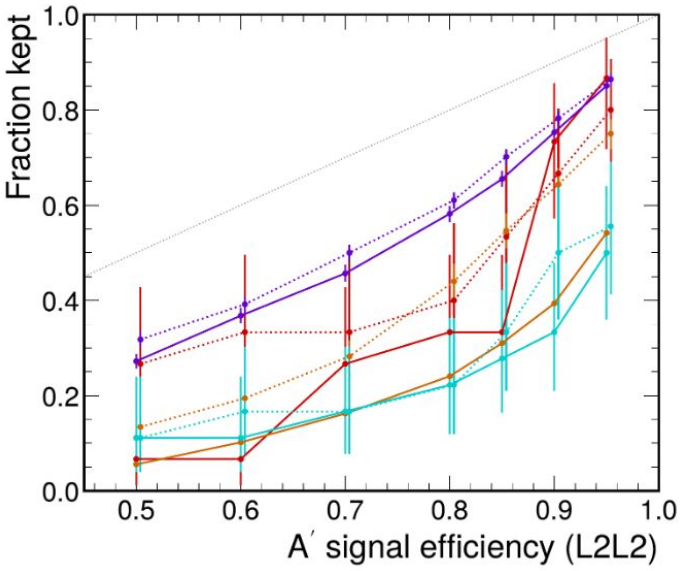


Signal Efficiencies

- Vertex quality score outperformed χ^2 baseline in all displacement regions, more significantly for L1L2, L2L2 categories.

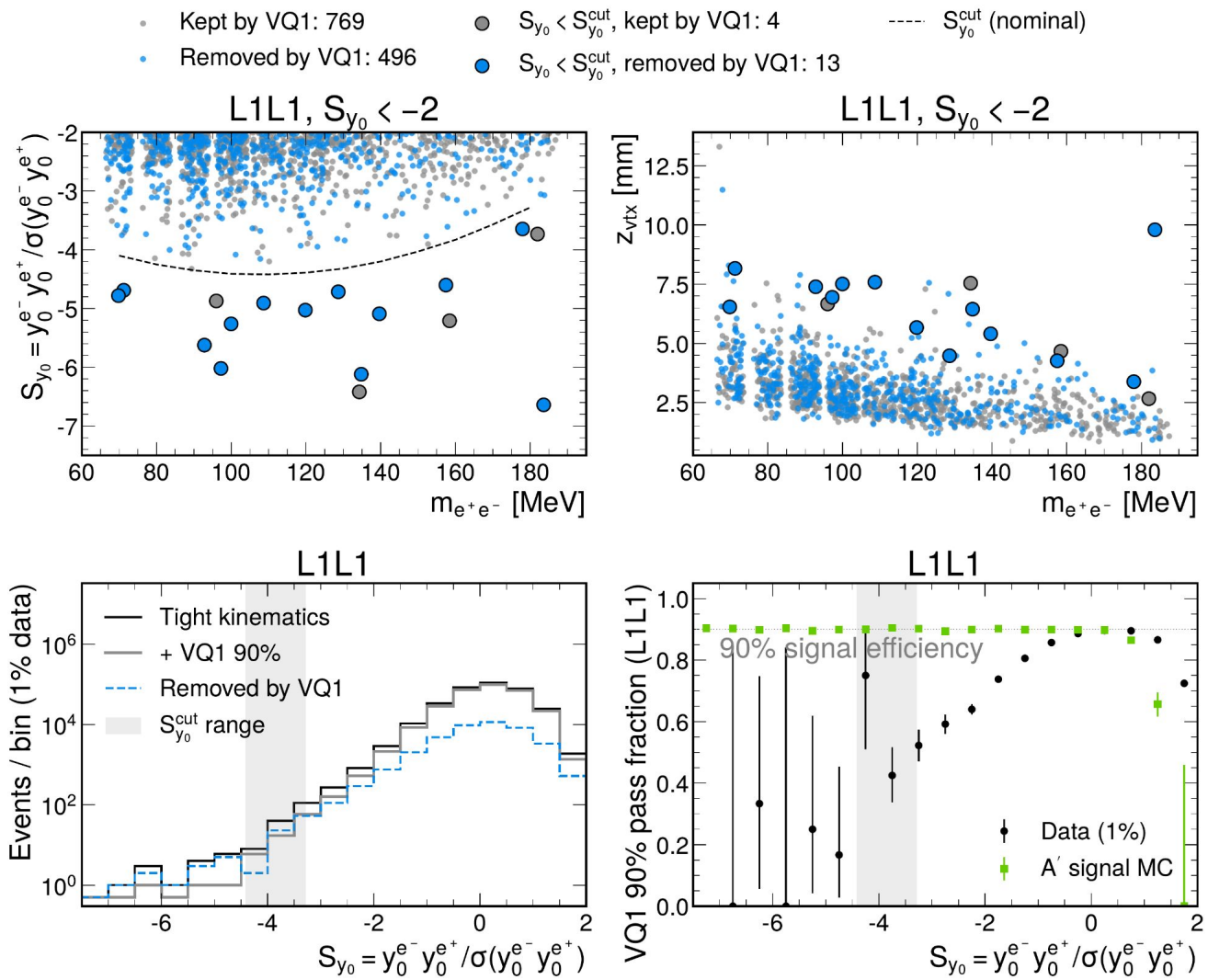






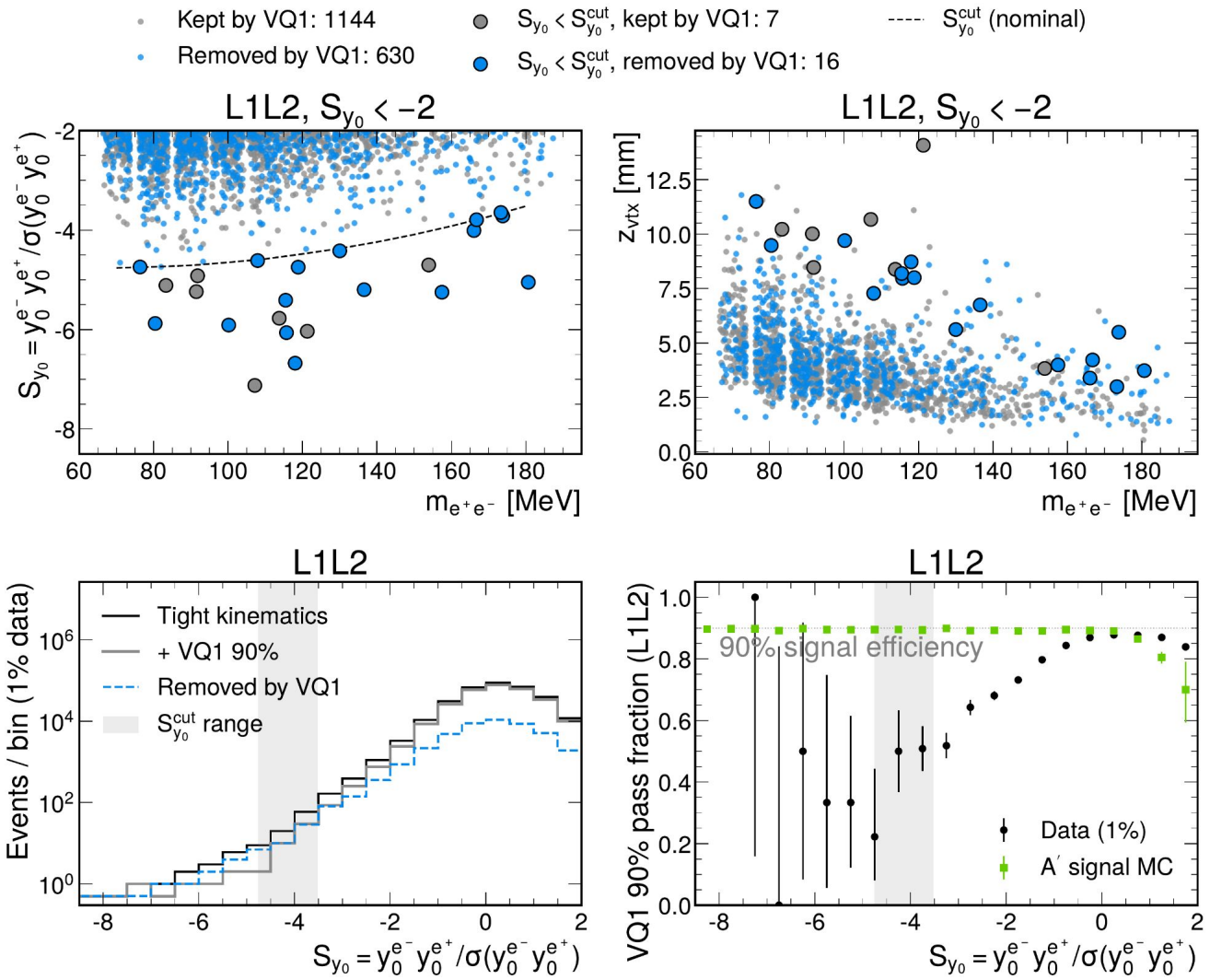
Background Rejection

- Vertex score appears to clean up tail in y0p.
- Signal efficiency remains high with increasing y0p.
- Behavior is consistent between hit categories.



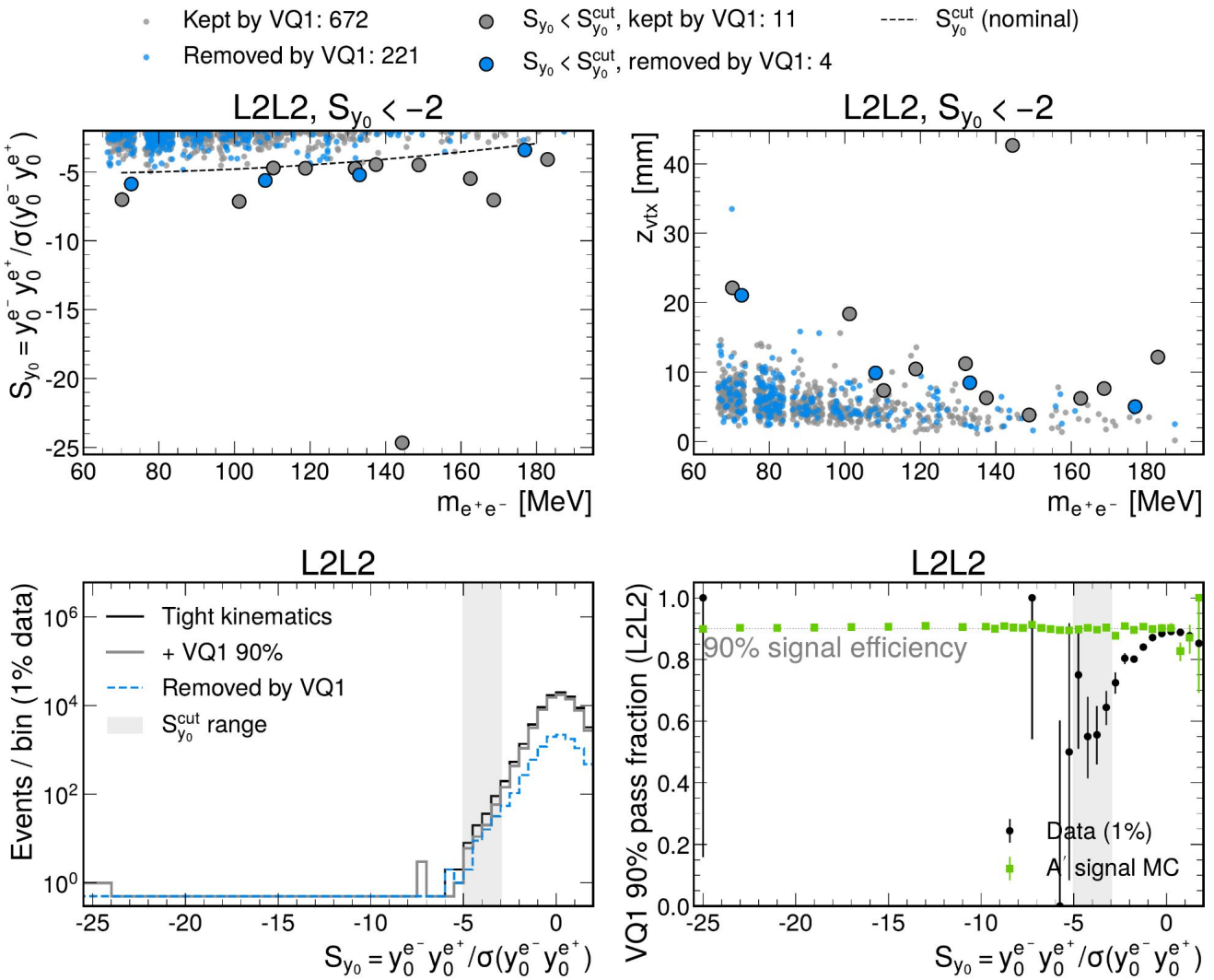
Background Rejection

- Vertex score appears to clean up tail in y0p.
- Signal efficiency remains high with increasing y0p.
- Behavior is consistent between hit categories.



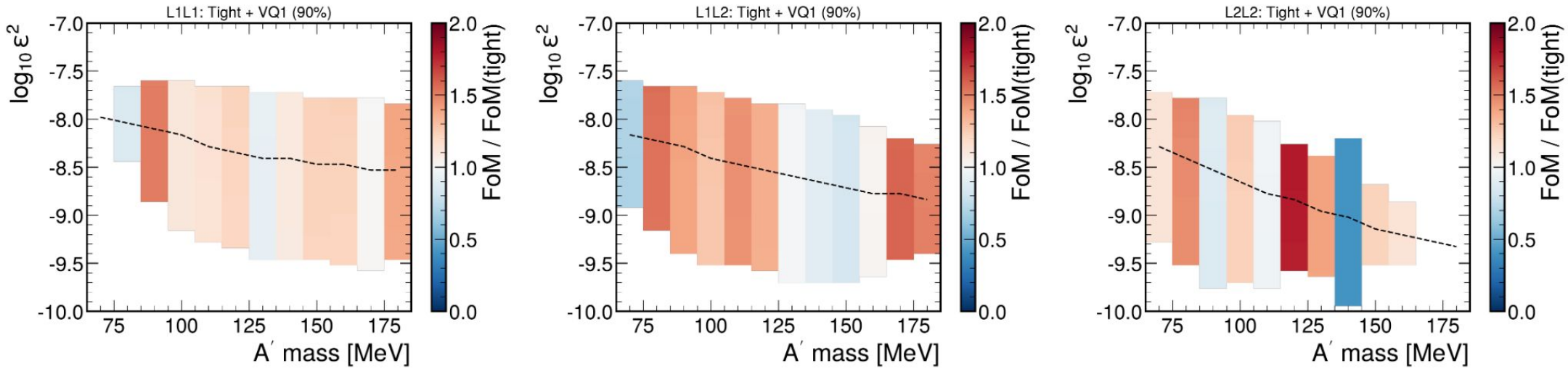
Background Rejection

- Vertex score appears to clean up tail in y0p.
- Signal efficiency remains high with increasing y0p.
- Behavior is consistent between hit categories.



Summary

- A vertex quality score trained only on data is able to remove backgrounds in y0p tail, while maintaining signal efficiency (behavior is stable at ~90% signal efficiency).
- Good agreement between data and MC.
- Improves upon stand-alone tight selection (using Punzi FoM for comparison).
- Outstanding questions: behavior when transferring to 10%?
  - Can test both direct application of 1% trained score and independent training on a subset of the unblinded 10% data.
- Likely some power in hit information (isolation variable, kinks, remove worst hit and refit, etc.)
  - Will test on new pass6_v10 pass, which has saved hit information.



Part 2: Signal/Background Classifier

Combining Features into Single Classifier

- Previous updates have shown marginal improvement using features largely orthogonal to displacement:
  - Kinematic features (see previous BDT background classification talks)
  - Vertex quality features (see part 1)
- In this part I am exploring an initial attempt at combining all of these features, the tight-selection variables, and our y0 displacement variable into a simple MLP, trained on signal vs. background.
  - Potential Issues:
    - **Data/MC invariance:** If features are poorly modelled in MC, a classifier may learn the difference between data/MC rather than signal/background, potentially resulting in an overestimate of our signal efficiency.
    - **Mass invariance:** Training is performed using discrete mass samples in signal. We want to avoid sculpting the background invariant mass distribution, especially if mass will be used in background estimation.
    - **Stability:** For a zero-background approach, we will need to cut tightly on background, in tail regions where we have little MC, low statistics in data. How much can we trust the model here?

Invariance

- Mass and data/MC invariance is imposed through "flatness" loss terms (https://arxiv.org/pdf/2010.09745, https://cms-ml.github.io/documentation/training/Decorrelation.html, https://arxiv.org/pdf/1410.4140),

$$\mathcal{L} = \mathcal{L}_{\text{BCE}} + \alpha \mathcal{L}_{\text{mass}} + \beta \mathcal{L}_{\text{data/MC}}$$

- Goal is to impose invariance in the background selection efficiency at the relevant cut points.
- These penalty terms are an average of the squared difference in pass rate (p) in a given mass bin (b) and cut quantile (q), compared to the average pass rate, divided by the binomial variance. The quantiles are determined separately for each hit category, and can be configured as needed—currently using 50 / 75 / 90 / 99 / 99.9% quantiles.

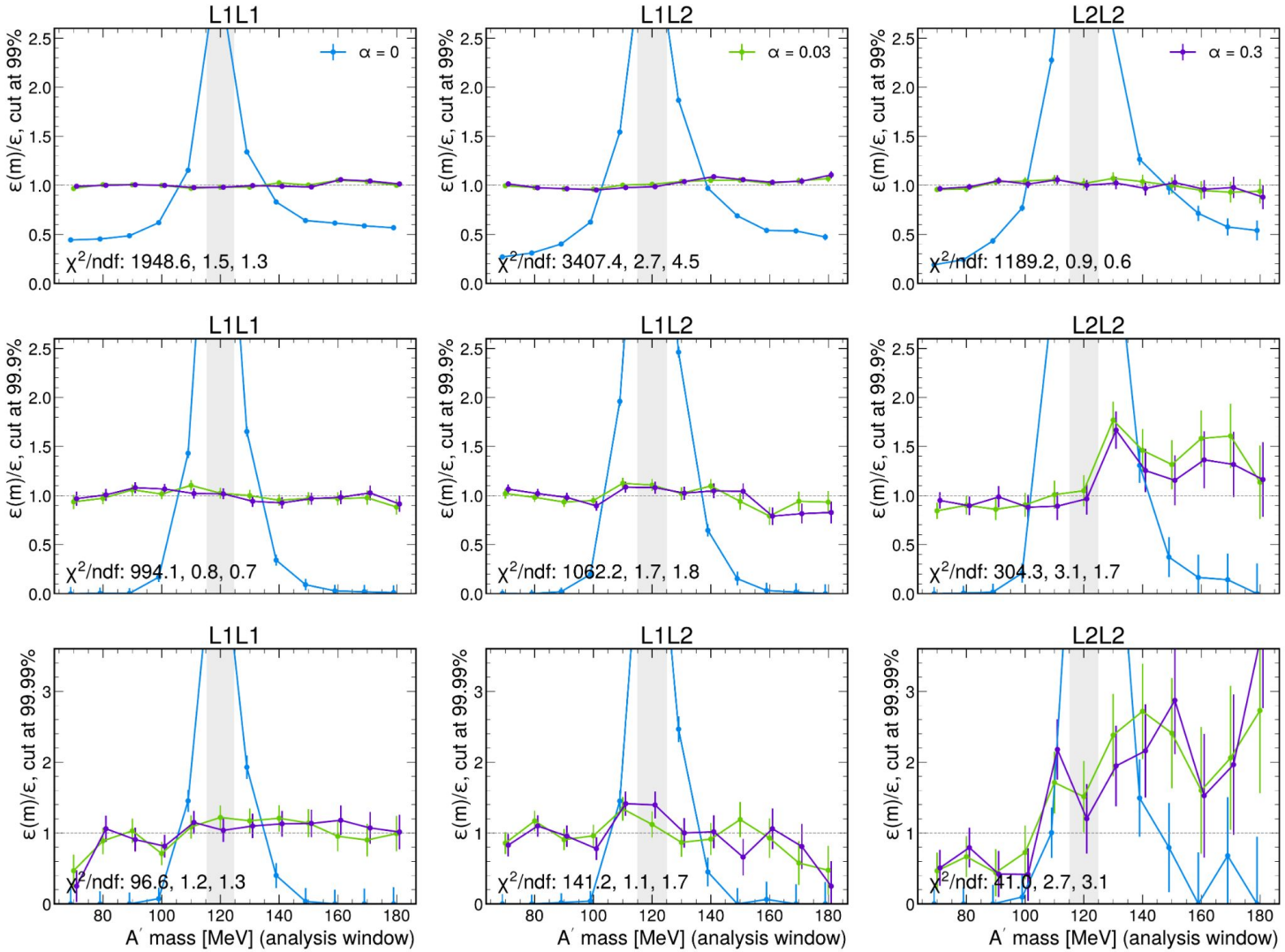
$$\mathcal{L}_{\text{mass}} = \frac{1}{|T|} \sum_{(b,q)\in T} \frac{N_b^{\text{eff}} \left(p_{b,q} - p_{c(b),q}\right)^2}{p_{c(b),q} \left(1 - p_{c(b),q}\right)}$$

- Data/MC invariance is enforced similarly, with the same quantiles—can be noisy given low MC stats.

$$\mathcal{L}_{\text{data/MC}} = \frac{1}{3} \sum_c \frac{1}{5} \sum_q \frac{\left(p_{c,q}^{data} - p_{c,q}^{MC}\right)^2}{\bar{p}_{c,q} \left(1 - \bar{p}_{c,q}\right) \left(1/N_{data}^{\text{eff}} + 1/N_{MC}^{\text{eff}}\right)}$$

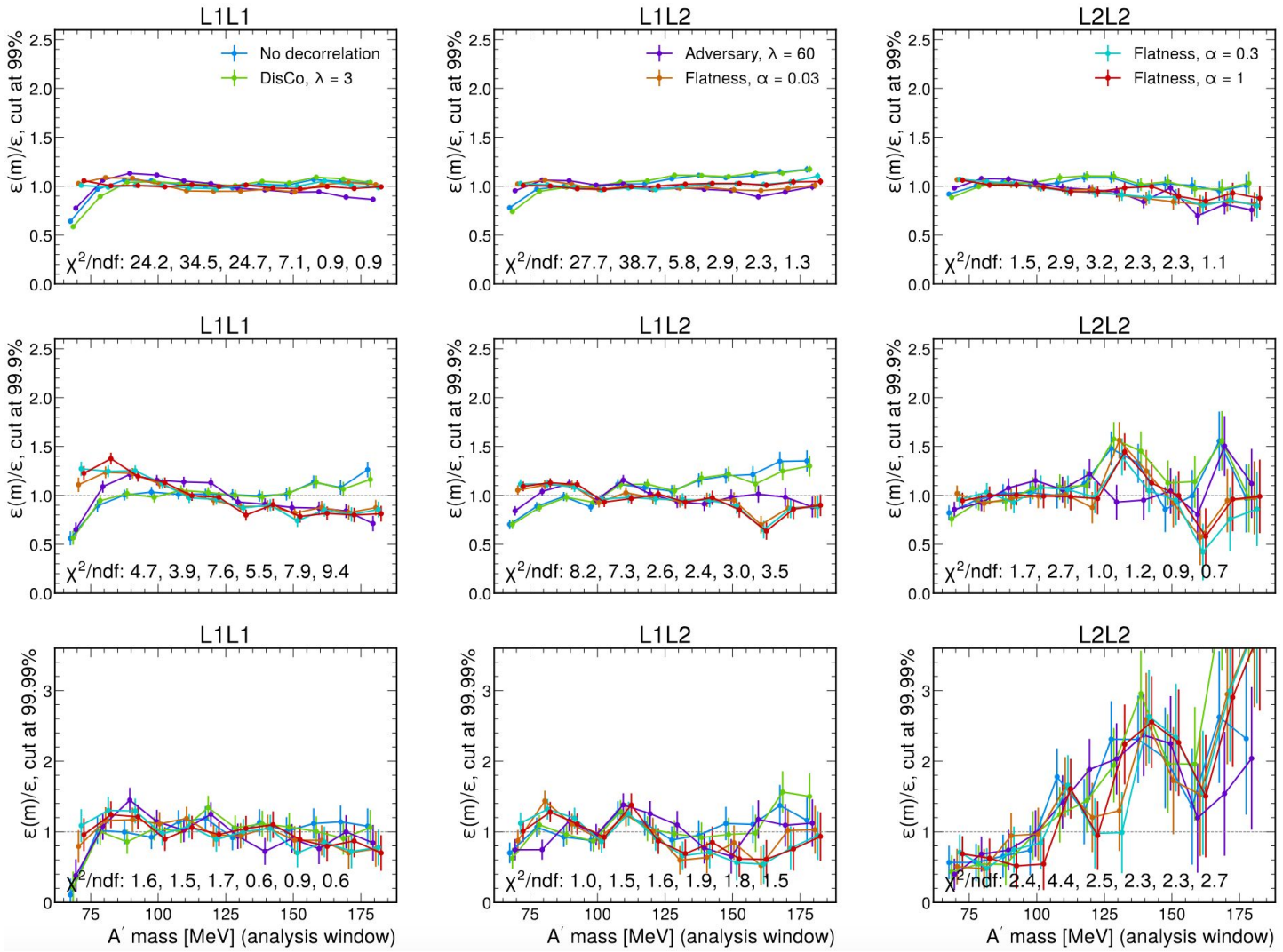
Examples of Mass Decorrelation

- Example training on a single mass point at 120 MeV.
- Without the penalty term, the network learns the signal mass, constructing a peak at 120 MeV in the background.
- Some structure remains in L2L2 category:
  - Lower statistics.
  - Reproduced from displacement cut alone.



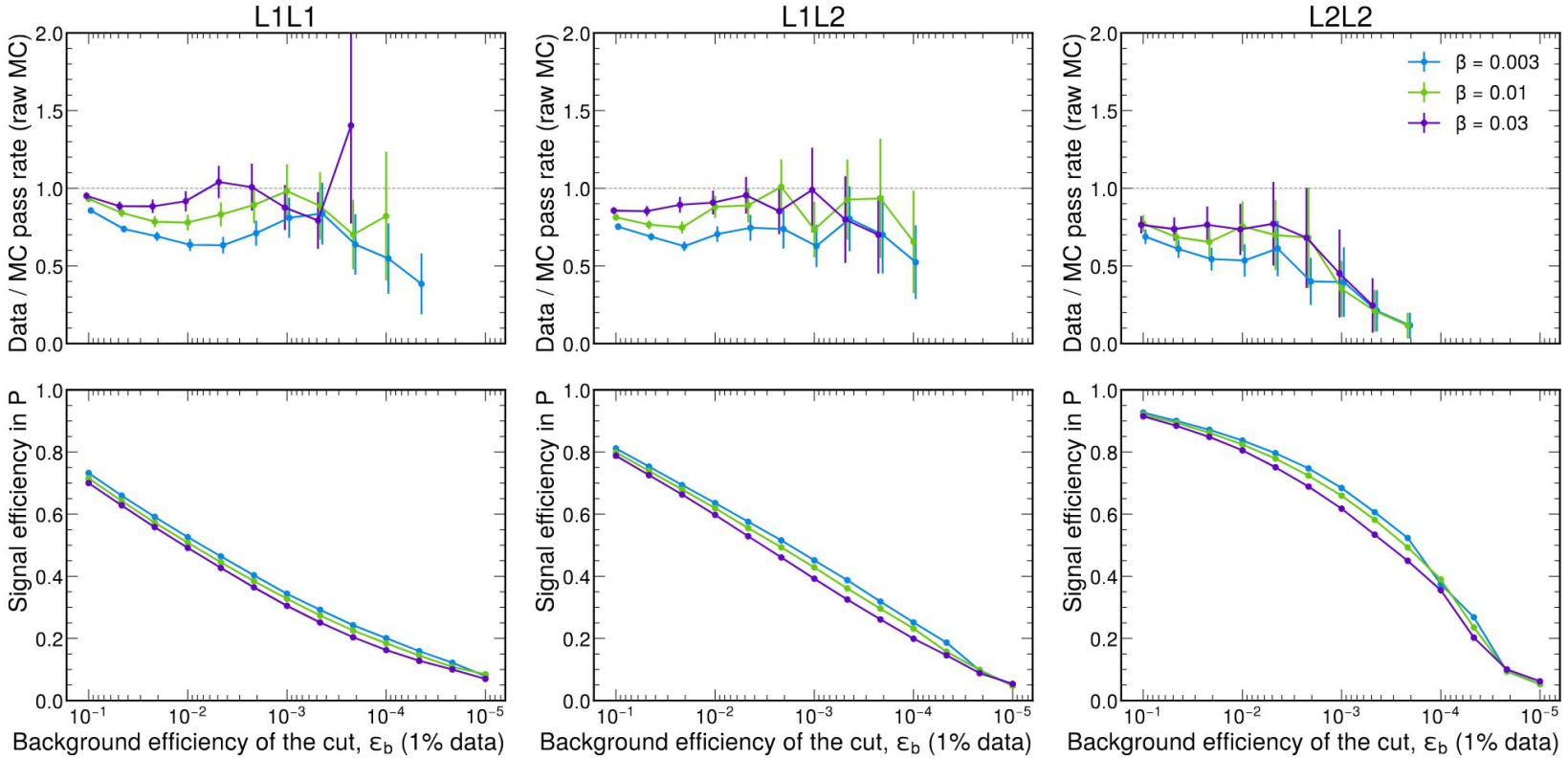
Comparisons to Other Decorrelation Methods

- Tested a few different decorrelation methods on a signal vs. background sample, training on all mass points simultaneously.
- The ANN and DisCo approaches work similarly, ANN achieves similar performance.
- For now am proceeding with the flatness loss.



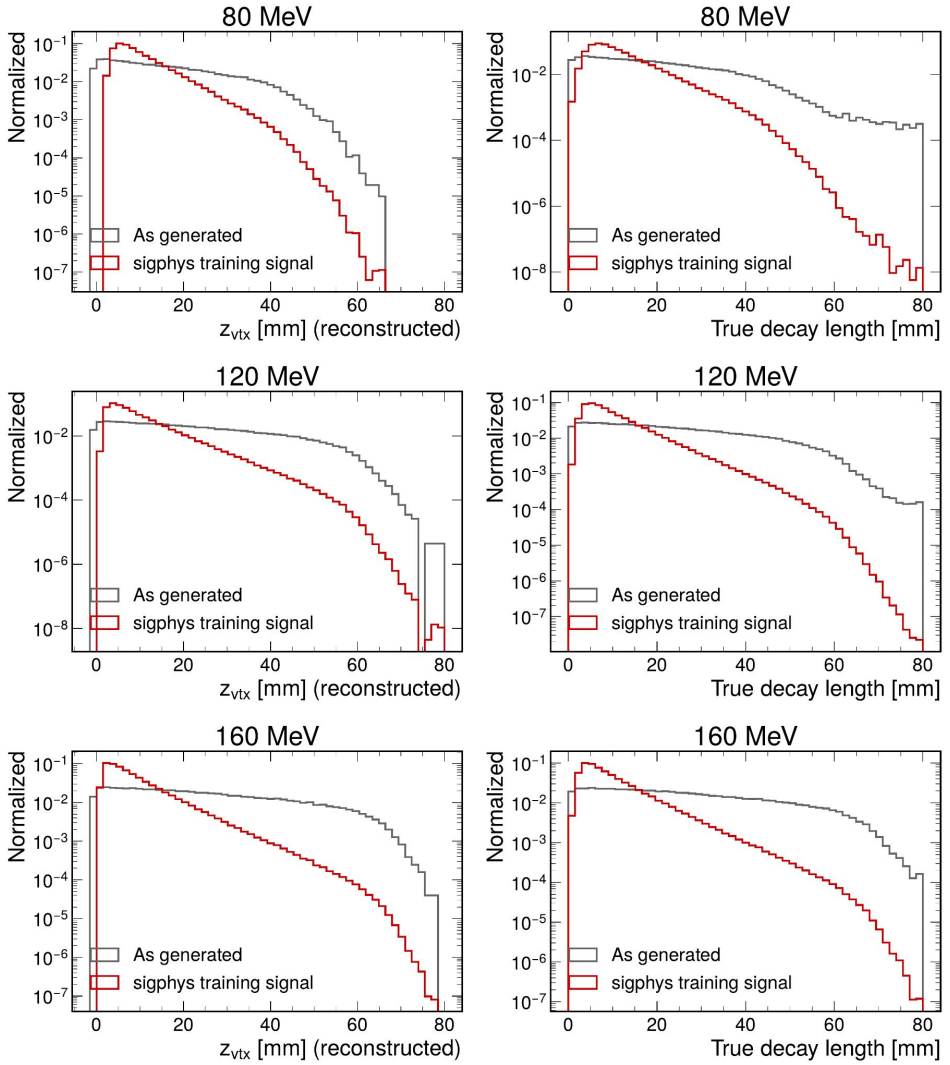
Examples of Data/MC Decorrelation

- If data/MC decorrelation term is too weak, background efficiency is higher in tritrig + WAB MC than data by ~30%, indicating that the network has learned to discriminate based on data/MC features.
- Efficiencies start to equalize with higher penalty weight, with a corresponding decrease in signal efficiency, caused by the removal of learned MC mismodelling qualities.



Training

- Training population: preselected vertices in L1L1, L1L2, L2L2 hit categories, downstream of the target.
- Background: 1% data and tritrig + WAB MC.
  - Background is 80% data and 20% MC by weight.
- Signal: A’ MC
  - Events are weighted by expected displaced yield after preselection, averaged across a range of ε^2 values where the significance of the tight selection is maximized—prioritize events at decay lengths consistent with the ε^2 values of interest.
  - Each mass carries equal total weight.
- Using same 5-fold validation technique as used in the vertex quality score.
  - Also explored difference in seed initialization, averaging across multiple seeds, etc.



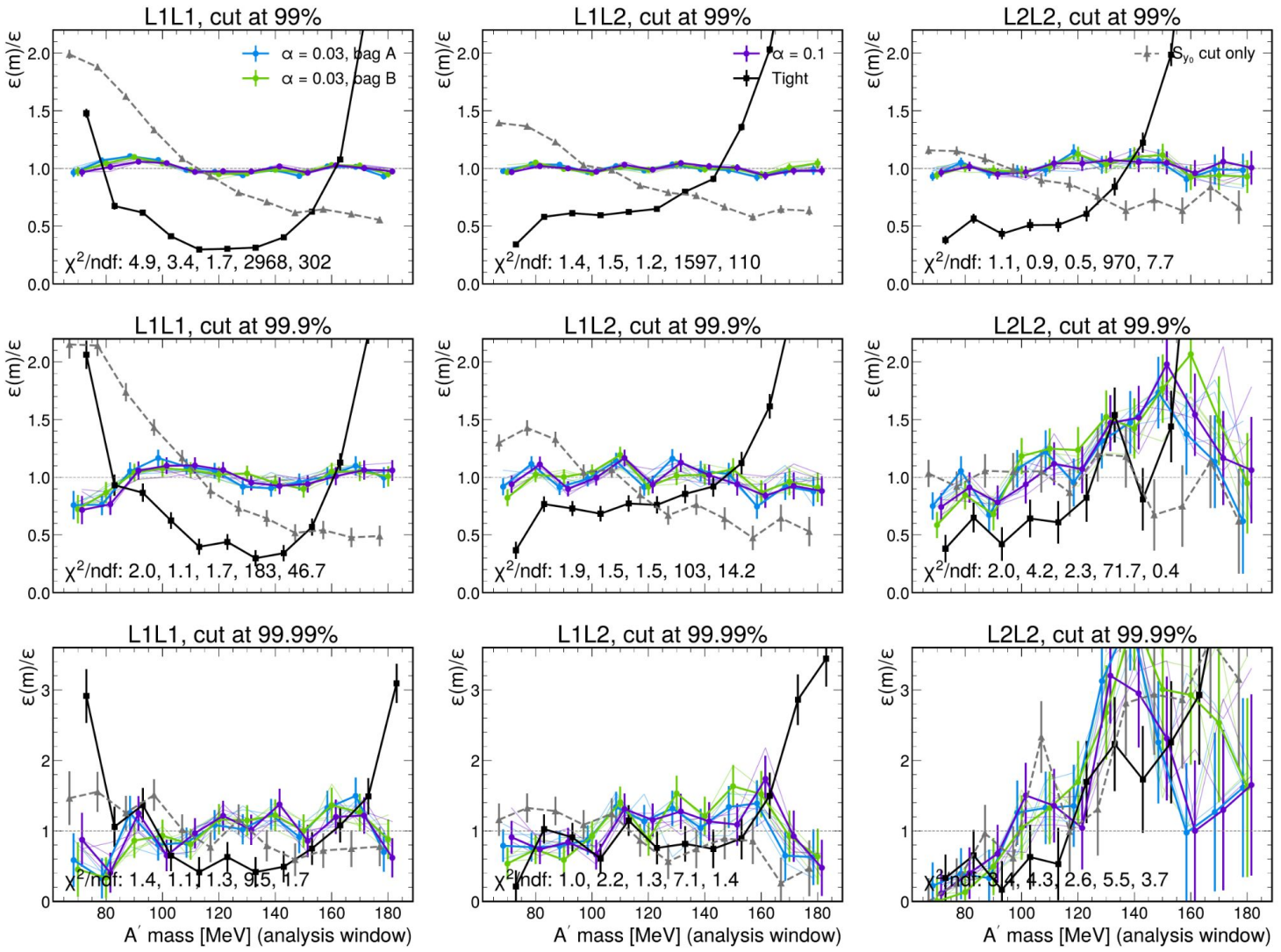
Features

- Combining features previously used in the background classification BDT, the vertex quality score, and standard tight selection variables.
- Many of these are strongly correlated, can be cleaned up.

Group	Inputs
Displacement	**ele_z0**, **pos_z0**, **vertex_pos_z**, **y0_y0err_product**, **ele_sig_z0**, **pos_sig_z0**
Tight-selection kinematics	**psum**, **pratio**, **abs_vtx_proj_x_sig**, **abs_vtx_proj_y_sig**
BDT kinematics	**bdt_ele_px**, **bdt_ele_pz**, **bdt_pos_px**, **bdt_pos_pz**, **bdt_theta_R**, **bdt_min_dTanLambda**, **bdt_min_asym**, **bdt_pos_pt**, **bdt_pt_asym**, **bdt_thetay_diff**, **bdt_d0_asym**, **bdt_phi0_asym**
Vertex quality (the VQ1 inputs)	**max/min_chi2ndf**, **vertex_chi2**, **max_abs_refit_pull_{p,tanl,phi}**, **max/min_refit_chi2**, **ele/pos_interior_holes**, **abs_dt_tracks**, **abs_dt_pos_trk_clu**, **abs_dt_ele_trk_pos_clu**, **pos_match_dx/dy**, **n_preselected_tracks**

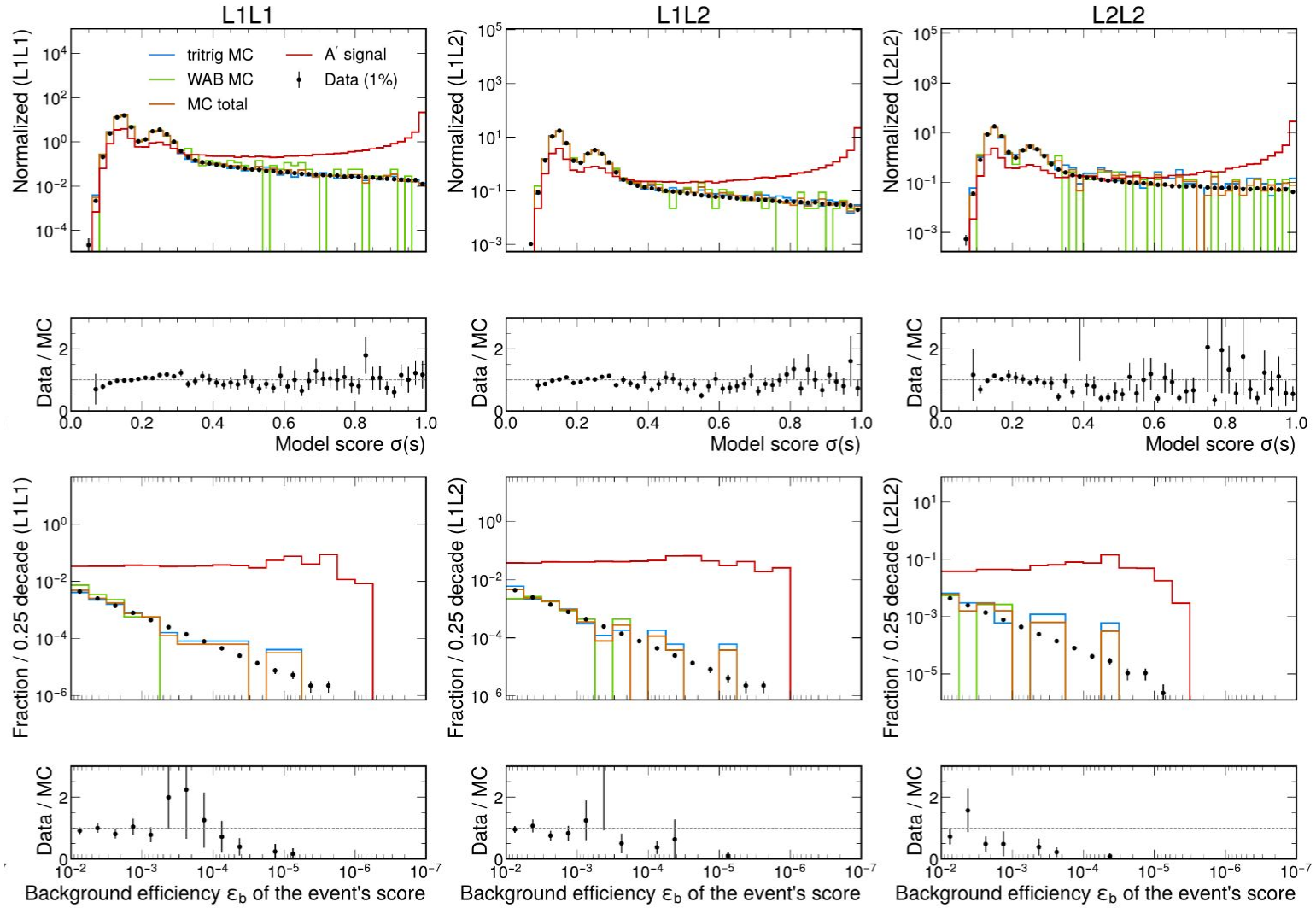
Mass Invariance

- Background efficiencies largely stable in invariant mass at chosen mass penalty weight, and consistent between different seed initializations.
- L2L2 stability starts to break down at tiger selections, where we have lower statistics in training.
  - Trends are consistent with standard tight selection cuts.



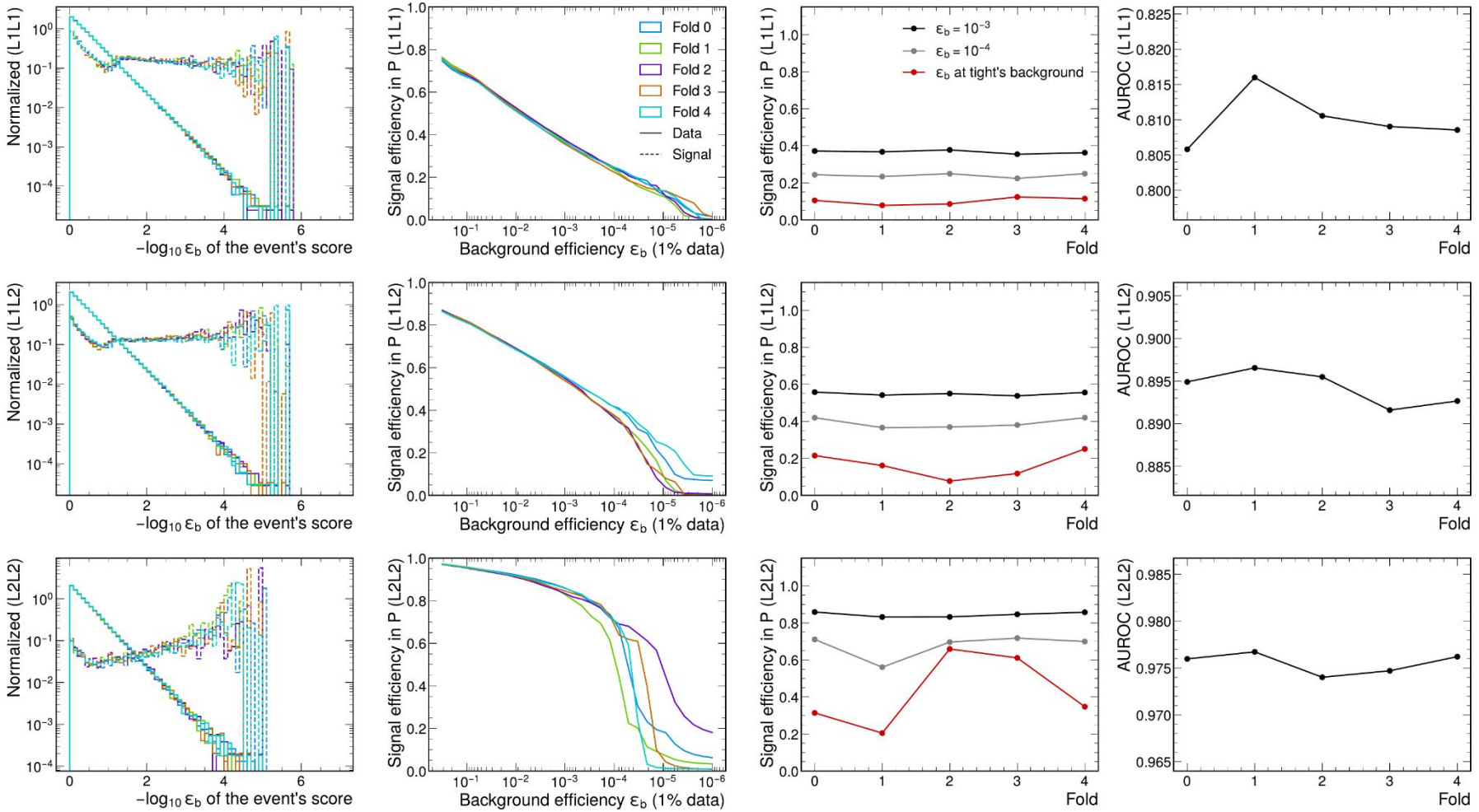
Data/MC Invariance

- Performance on data/MC is stable in core of distribution, and within tail up to MC statistics in L1L1, L1L2 categories.
- Some deviation in L2L2.



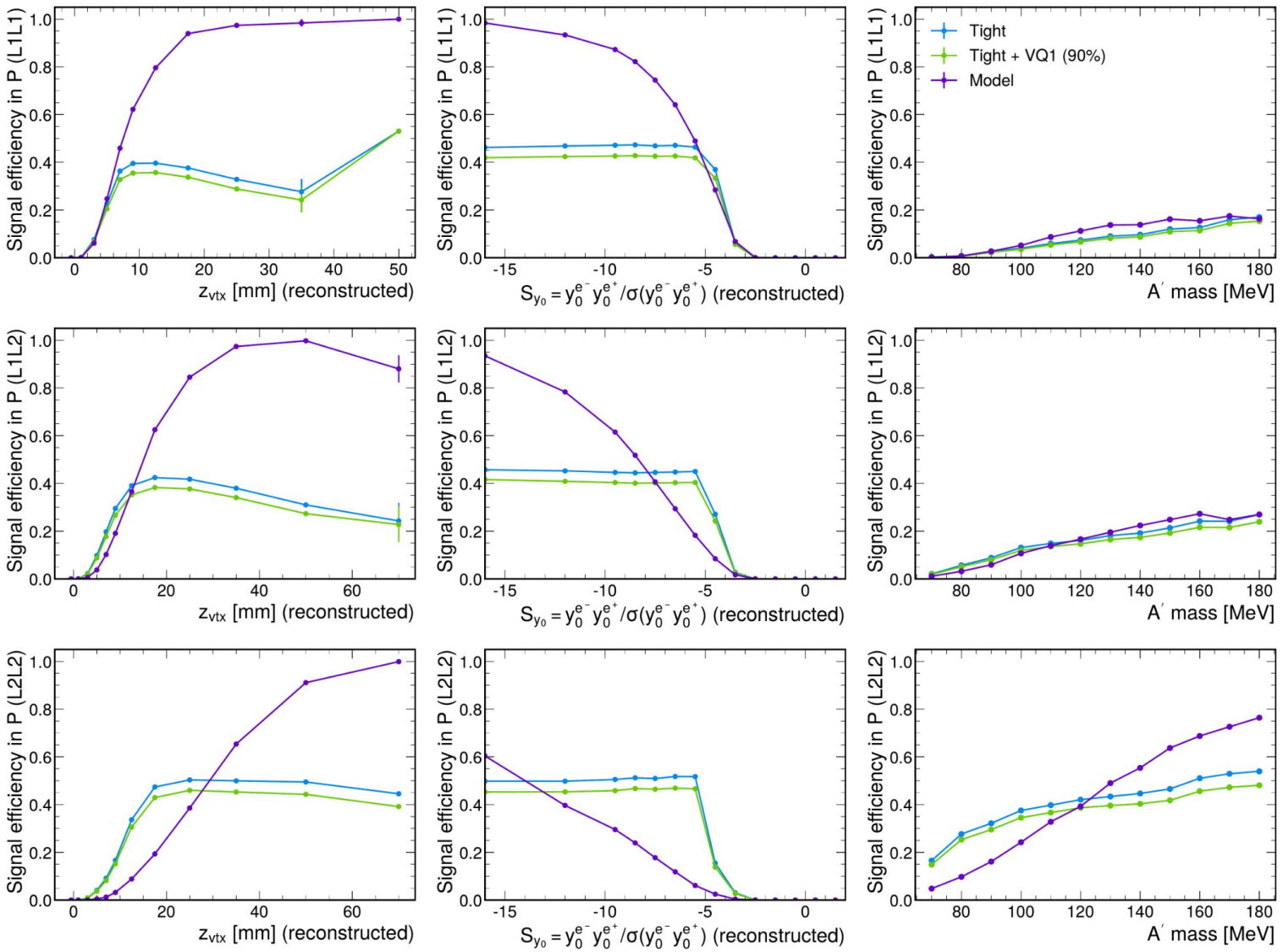
Fold Stability

- Folds deviate in deep tails of the score distribution, most significantly in L2L2.
- Averaging across several seed initializations can help with this.



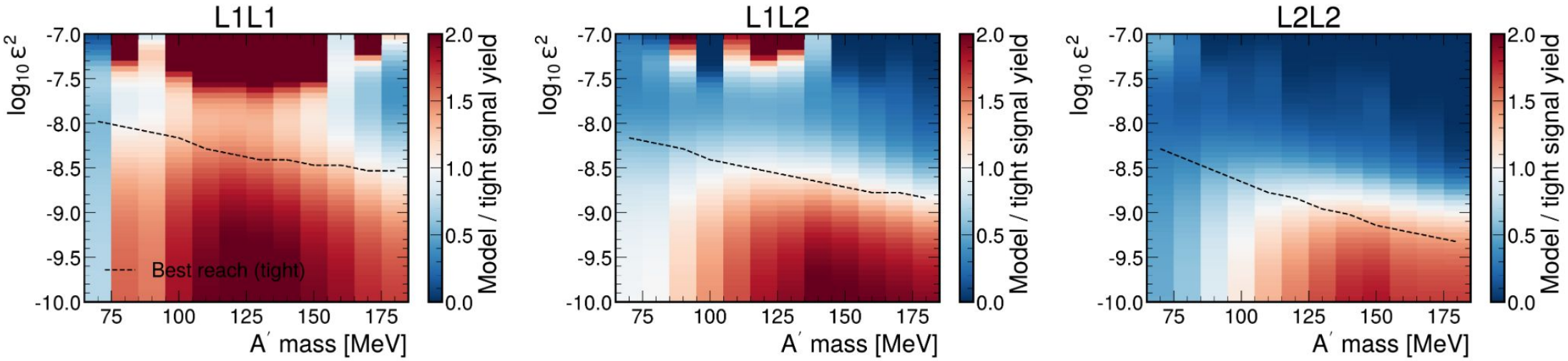
Signal Efficiency

- Signal efficiencies in L1L1 consistently higher in vertex z, y0p, and A' mass.
- Other hit categories have regions of worse signal efficiency.



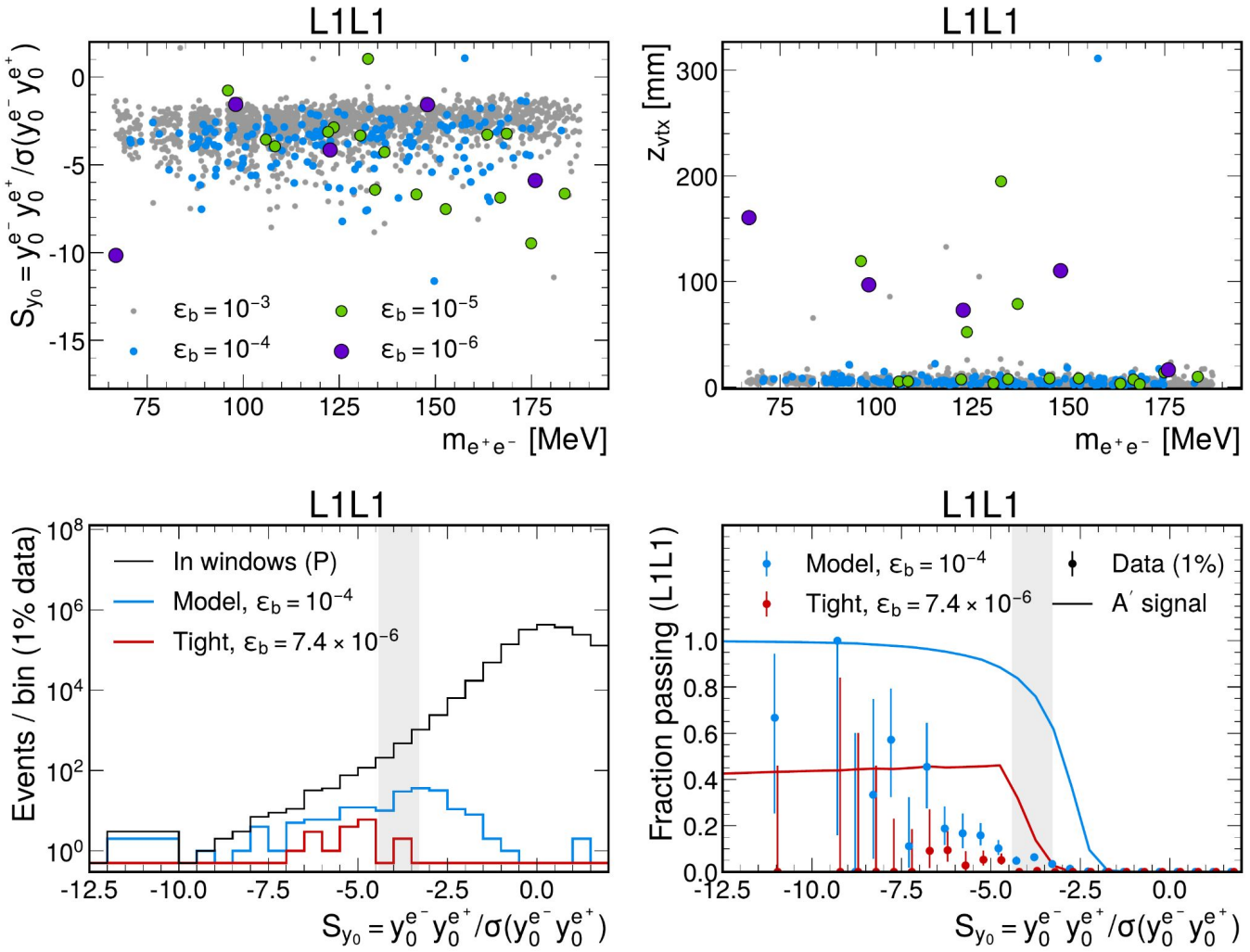
Signal Yields

- At same background as tight selection, signal yields improve by up to 2x at certain values of ε^2 .
  - WIll need to cut tighter for exclusion at 100%, unclear how model behaves there.
- Biggest improvement in L1L1.
- L1L2, L2L2 categories show significantly worse signal efficiencies for some ε^2 values.

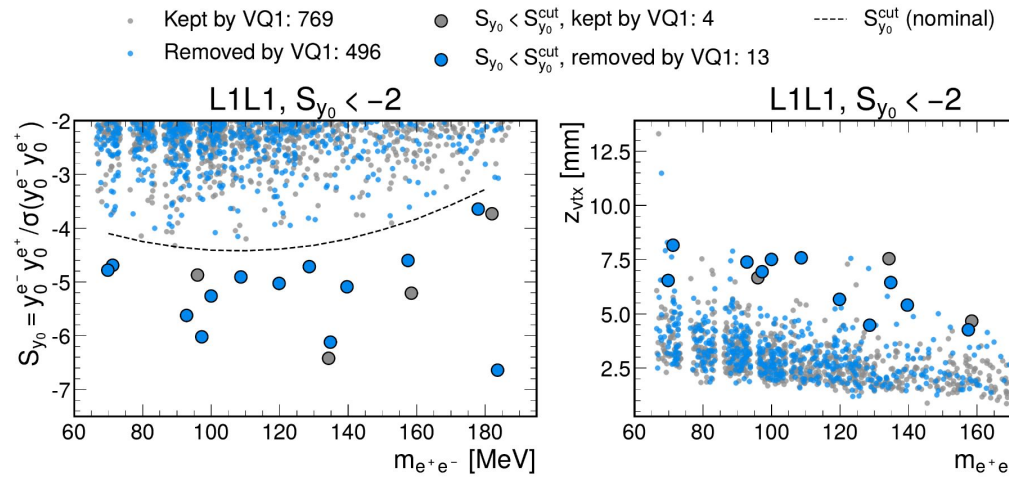


Background Rejection

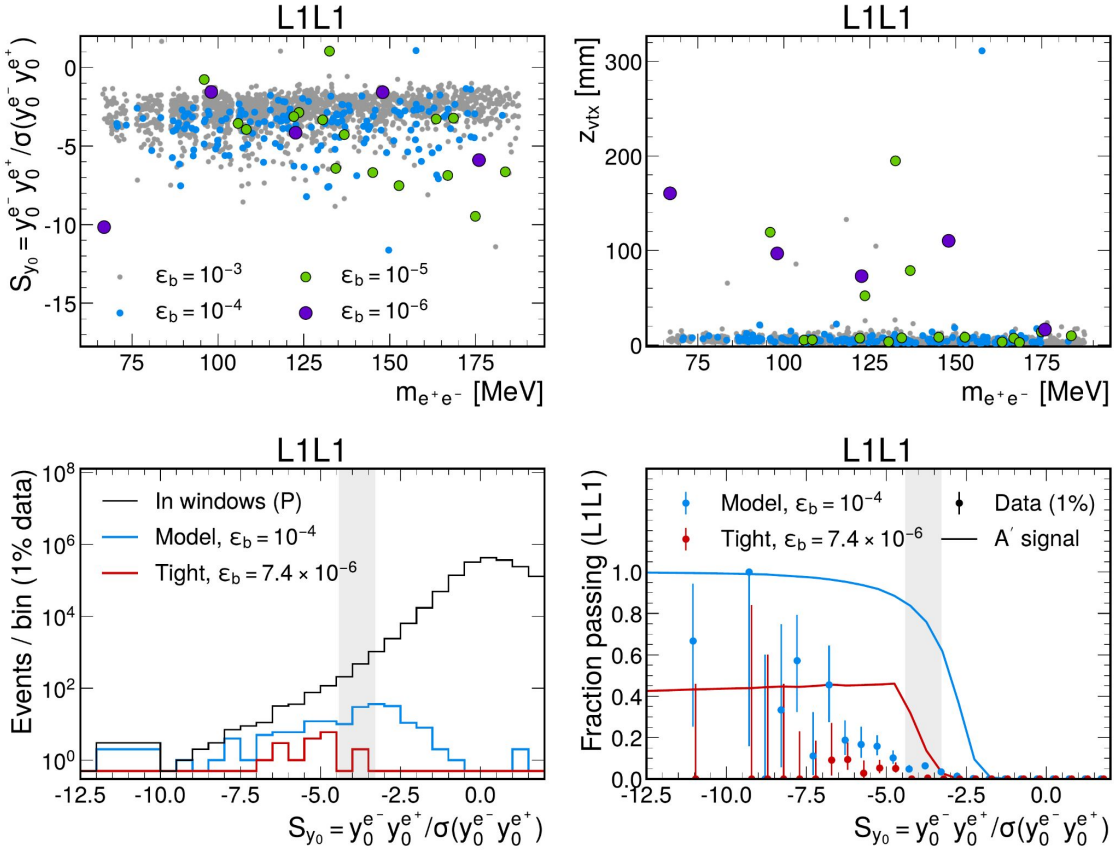
- Discriminator seems to select different events than the standard tight selection—leaves several very displaced background vertices that would otherwise be removed by vertex projection cut.
- Potentially problematic—network can be tuned to be more discriminatory in displaced region by down-weighting prompt background vertices.



Background Rejection



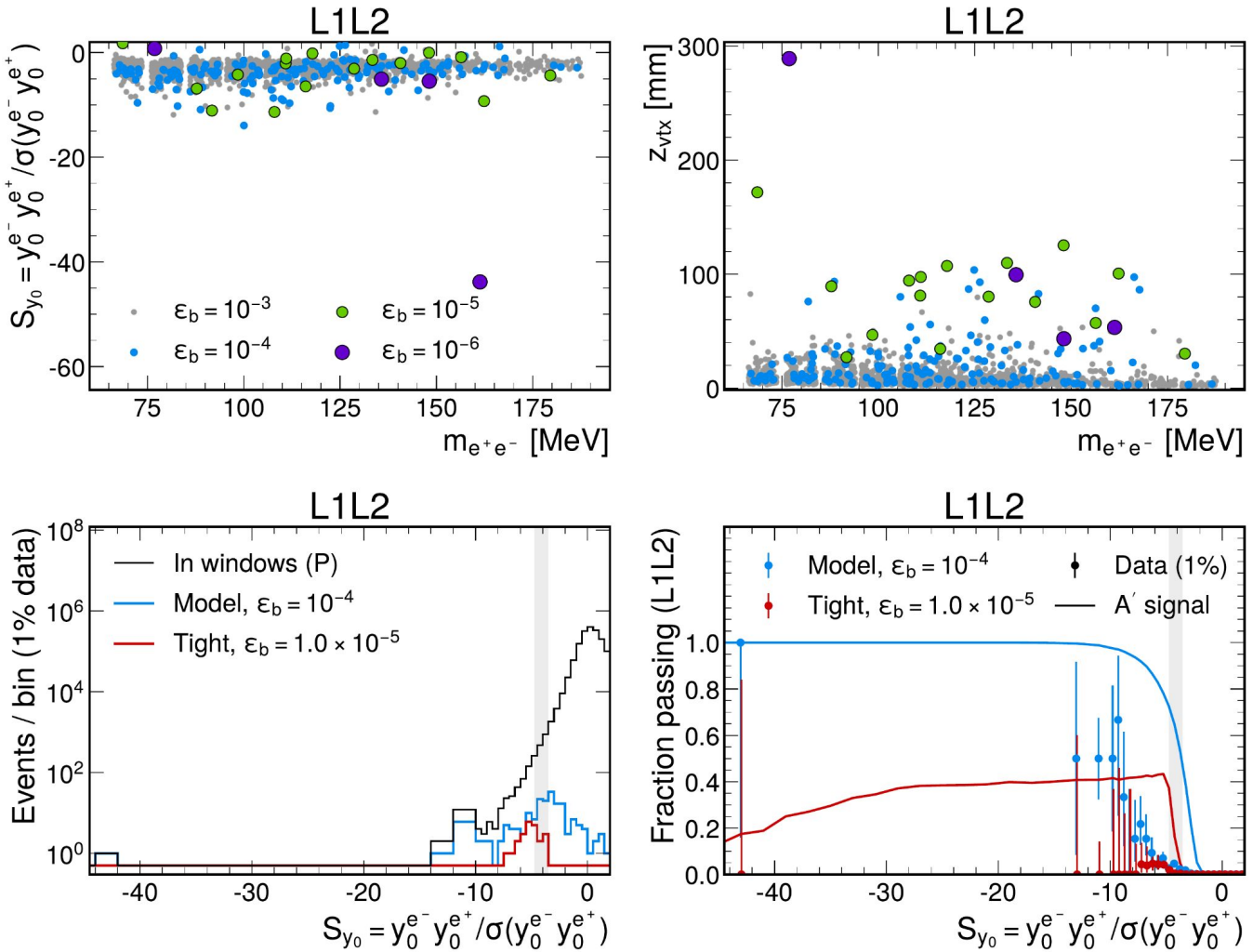
Vertex quality score



Signal/background discriminator

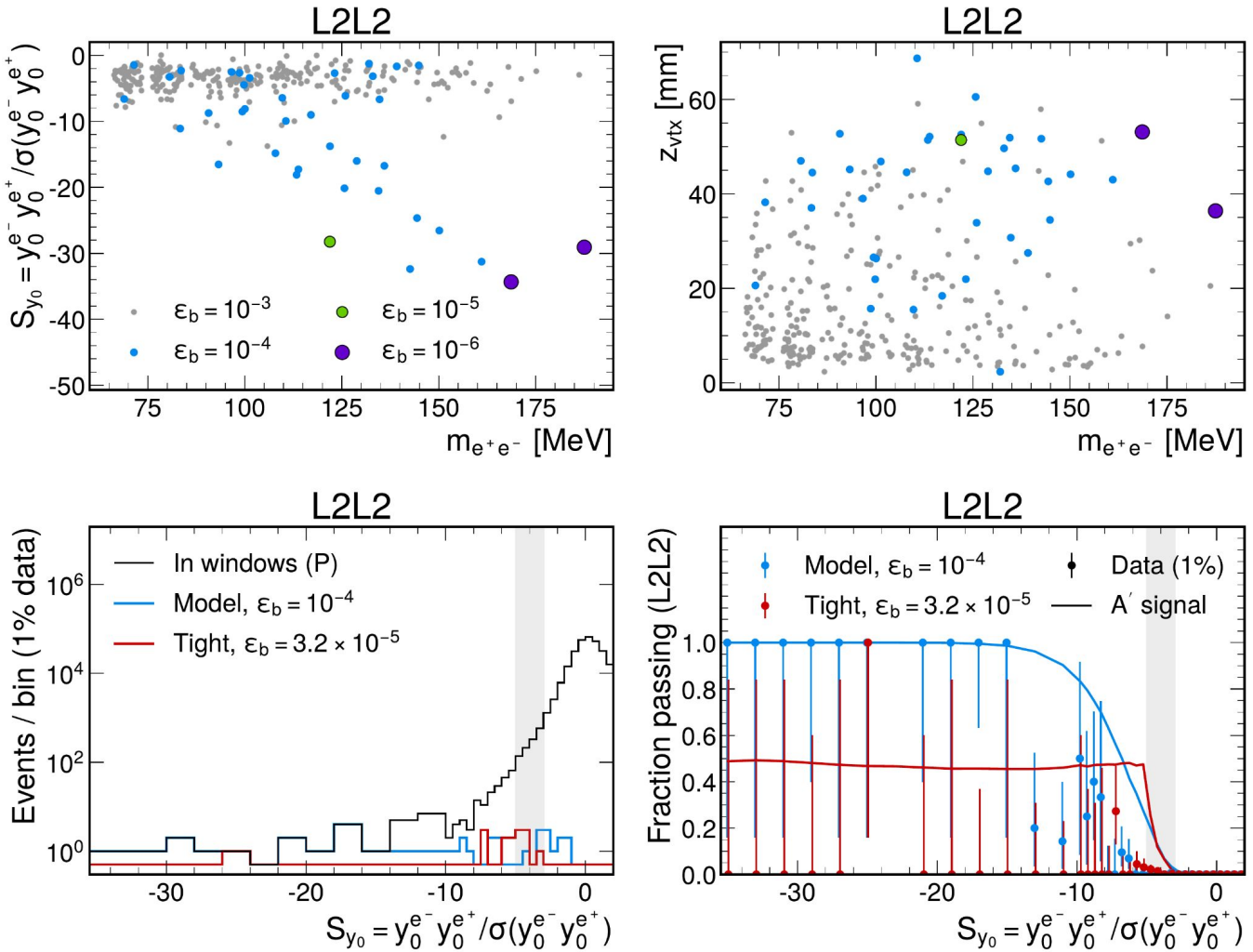
Background Rejection

- Discriminator seems to select different events than the standard tight selection—leaves several very displaced background vertices that would otherwise be removed by vertex projection cut.
- Potentially problematic—network can be tuned to be more discriminatory in displaced region by down-weighting prompt background vertices.



Background Rejection

- Discriminator seems to select different events than the standard tight selection—leaves several very displaced background vertices that would otherwise be removed by vertex projection cut.
- Potentially problematic—network can be tuned to be more discriminatory in displaced region by down-weighting prompt background vertices.



Summary

- Current approach for enforcing data/MC invariance, mass invariance looks promising.
- Need to be careful when cutting tightly on background:
  - Careful validation, looking at stability at different seed initializations.
- Current result leaves displaced vertices that are otherwise rejected by tight-selection:
  - Need to understand how selections differ, how to weigh training samples to achieve desired behavior.
- Other potentially useful techniques: parametric neural networks (pNN): https://cds.cern.ch/record/2842568/files/2202.00424.pdf

