

Wire-Cell PR selection: plot updates

Matteo Vicenzi (mvicenzi@bnl.gov)

SBND Wire-Cell Meeting
Sep 22nd, 2026



Updates

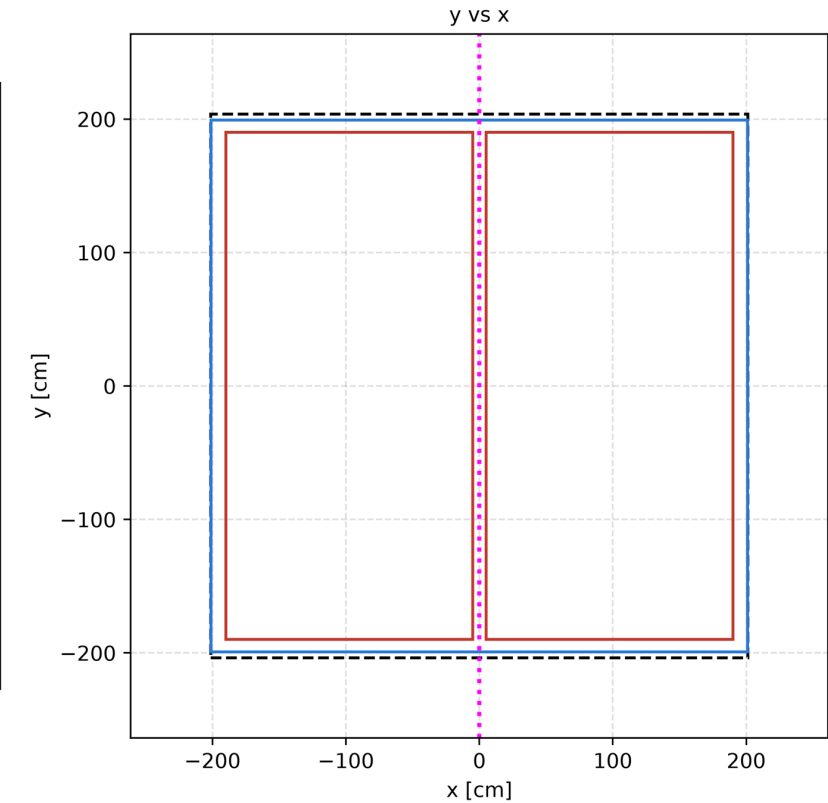
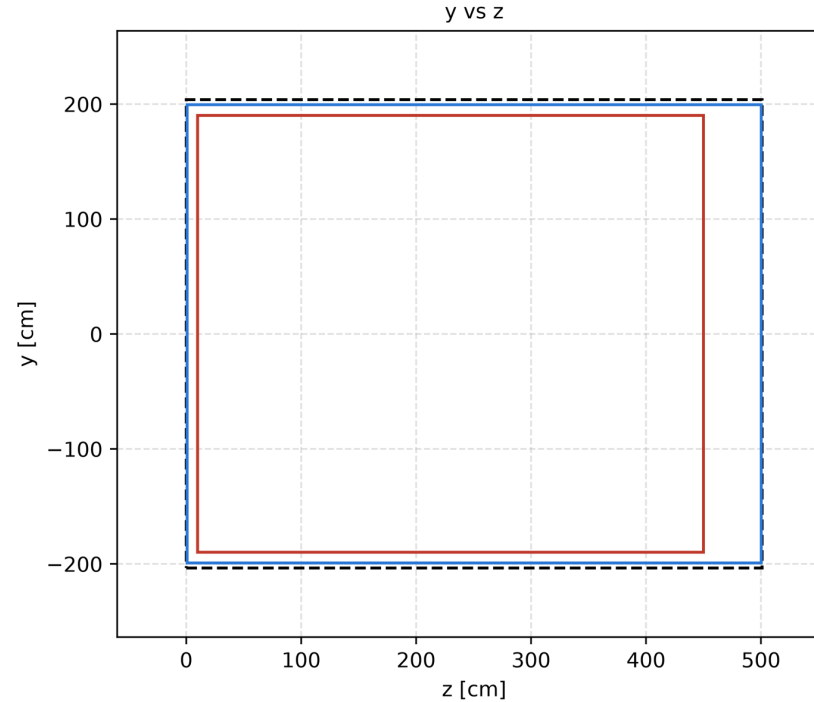
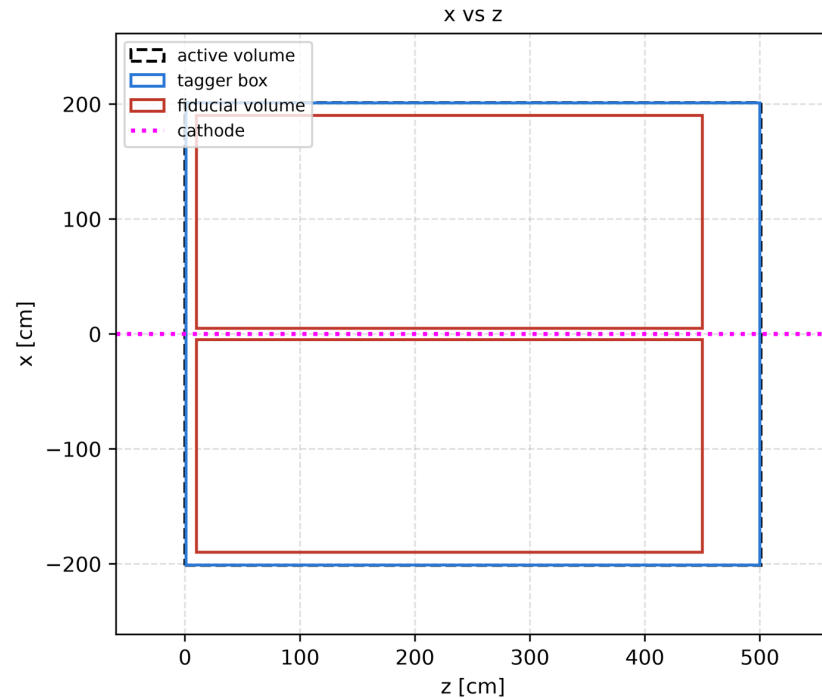
Previously:

- Signal = true ν_μ CC (ν_e CC) with $\text{edep} > 0$ MeV and vertex in AV
- Selection:
 - PR selects neutrino candidate (implicit)
 - BDT numu (nue) score

New:

- Signal = true ν_μ CC (ν_e CC) with $\text{edep} > 100$ MeV (75 MeV) and vertex in FV
- Selection:
 - PR selects neutrino candidate (implicit)
 - Reconstructed vertex in FV
 - BDT numu (nue) score

Fiducial volume



10 cm inward from each wall.
50 cm inward from the downstream z wall
5 cm away from each side of the cathode

Efficiency/Purity

$$\text{Efficiency} = \frac{\text{signal } \nu \text{ of class X, whose best-matched}^* \text{ candidate passes the selection for X}}{\text{all signal } \nu\text{'s of class X}}$$

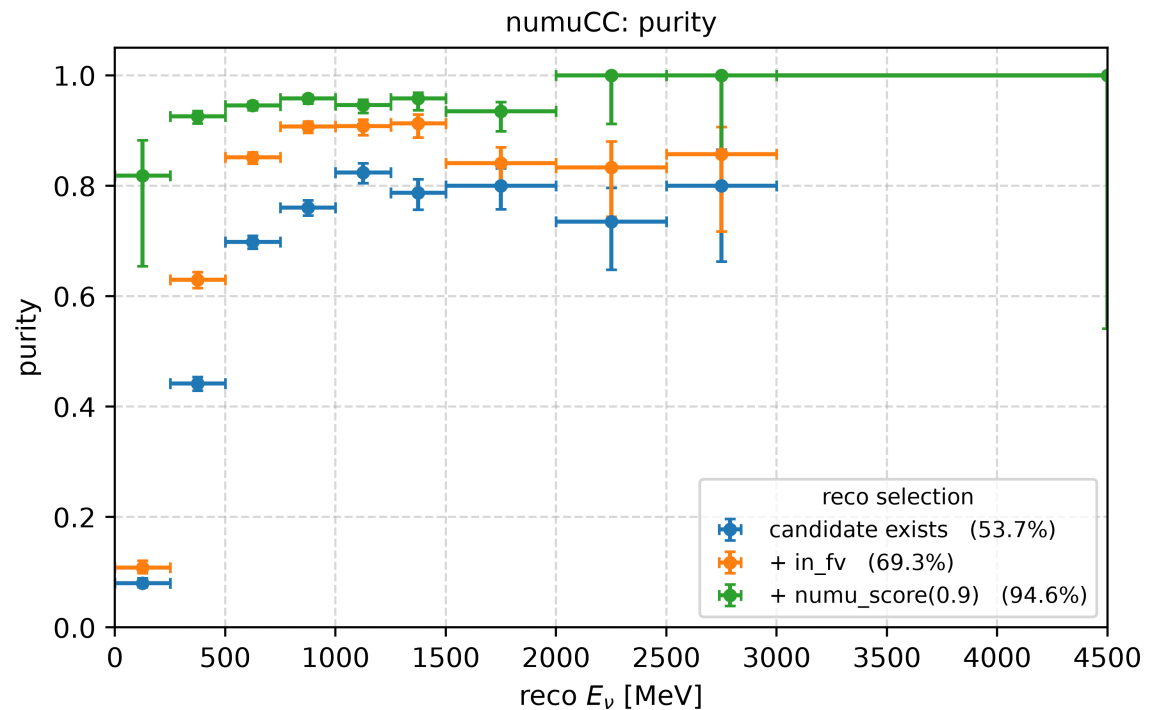
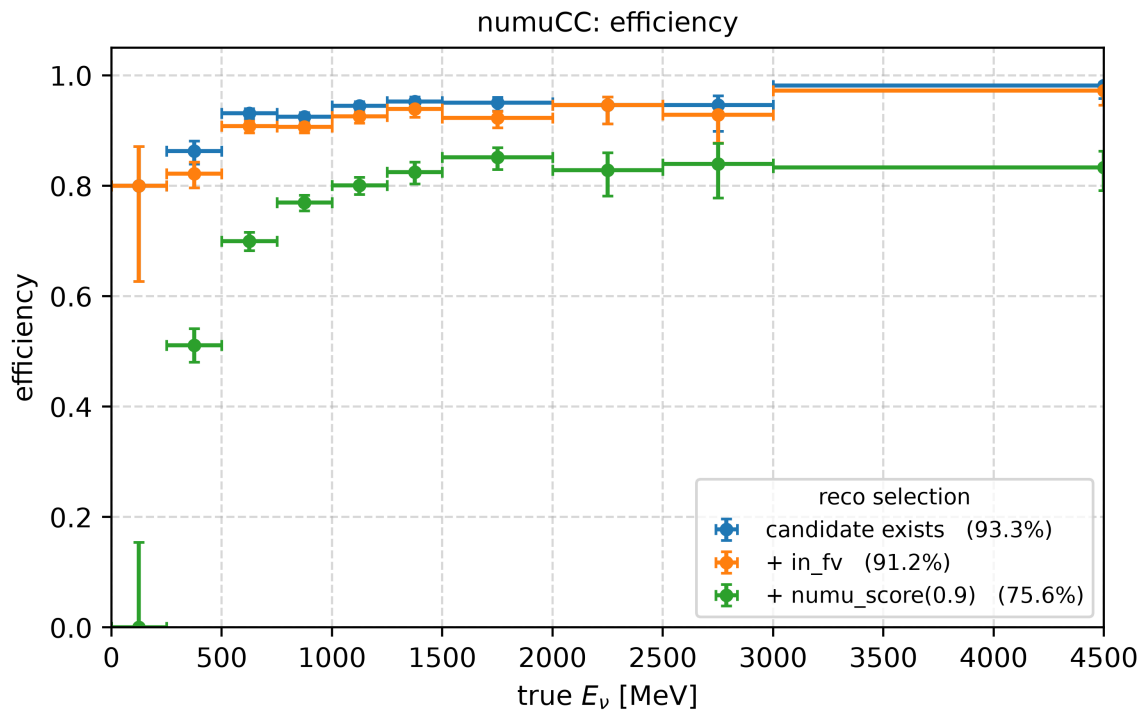
$$\text{Purity} = \frac{\text{selected candidates of class X whose best-matched}^* \text{ true } \nu \text{ is class X}}{\text{all selected candidates of class X}}$$

* both definitions require **matching** between true neutrinos and reco candidates

Selection cuts ($\nu_\mu CC$)

Assuming the updated signal definition (edep>100 MeV && vertex in FV), looking at the progressive selection cuts.

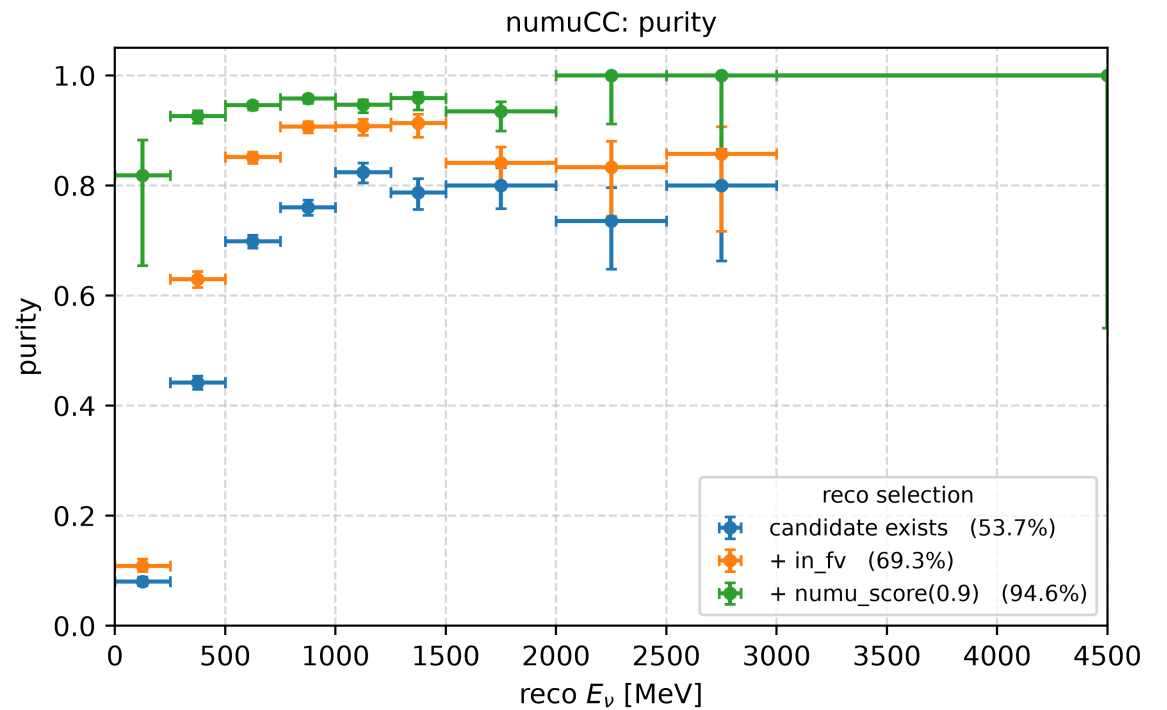
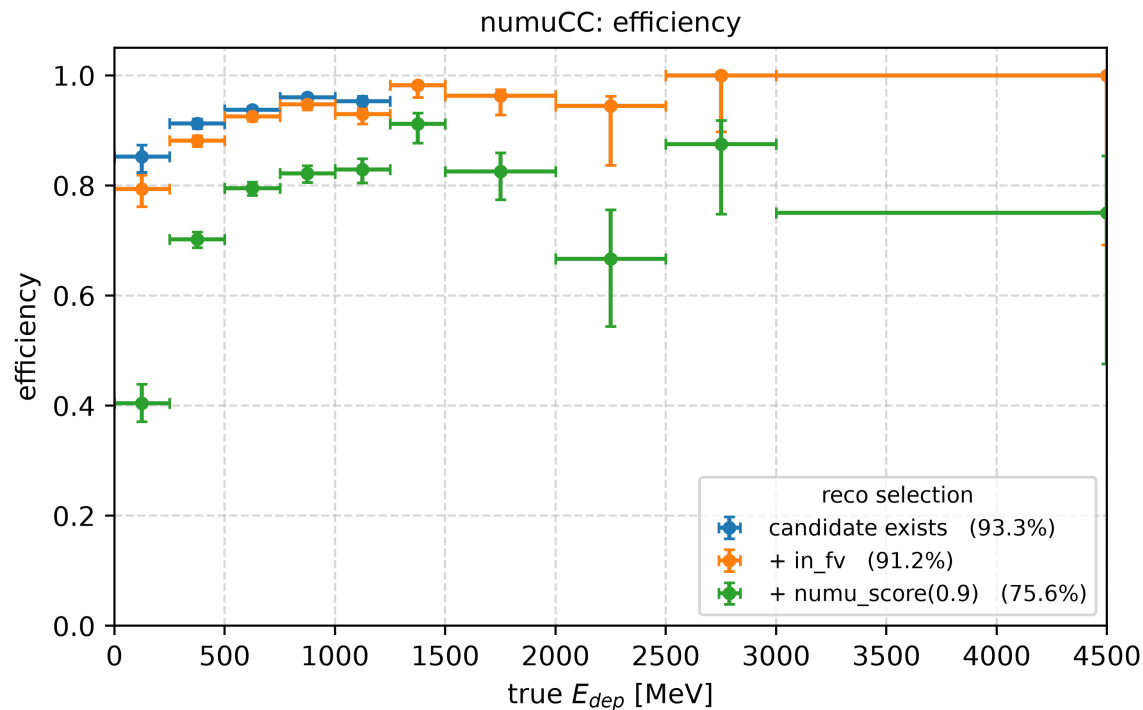
numuCC: signal = true: in_fv & edep_min(100) & numucc



Selection cuts ($\nu_\mu CC$)

Assuming the updated signal definition (edep>100 MeV && vertex in FV), looking at the progressive selection cuts.

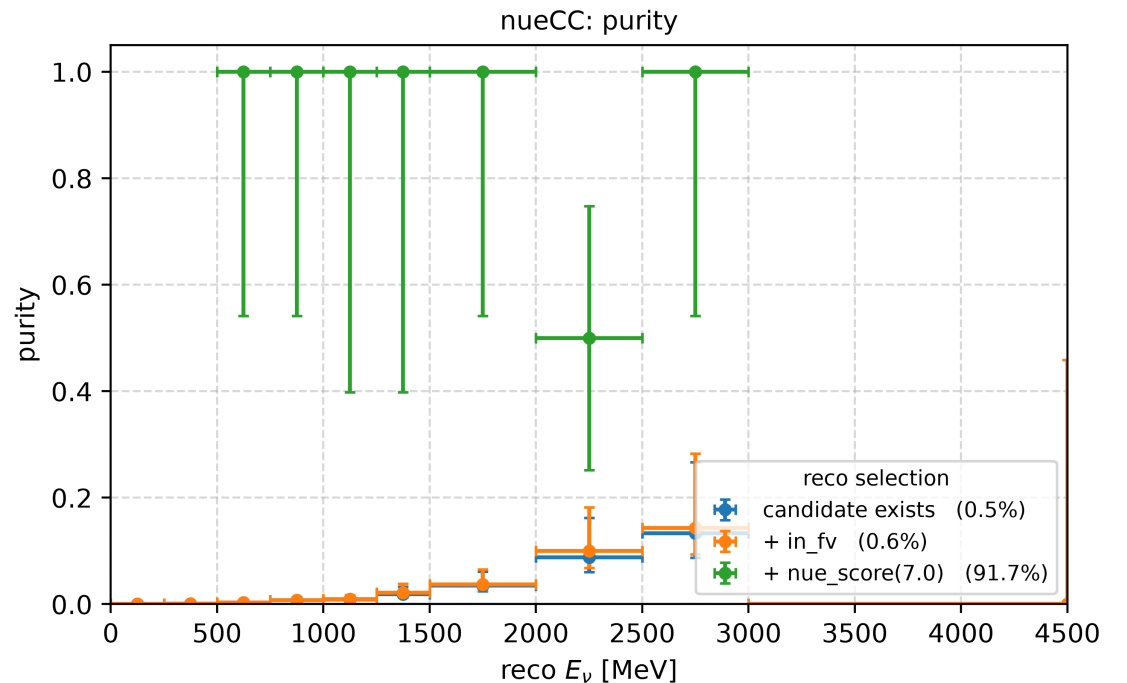
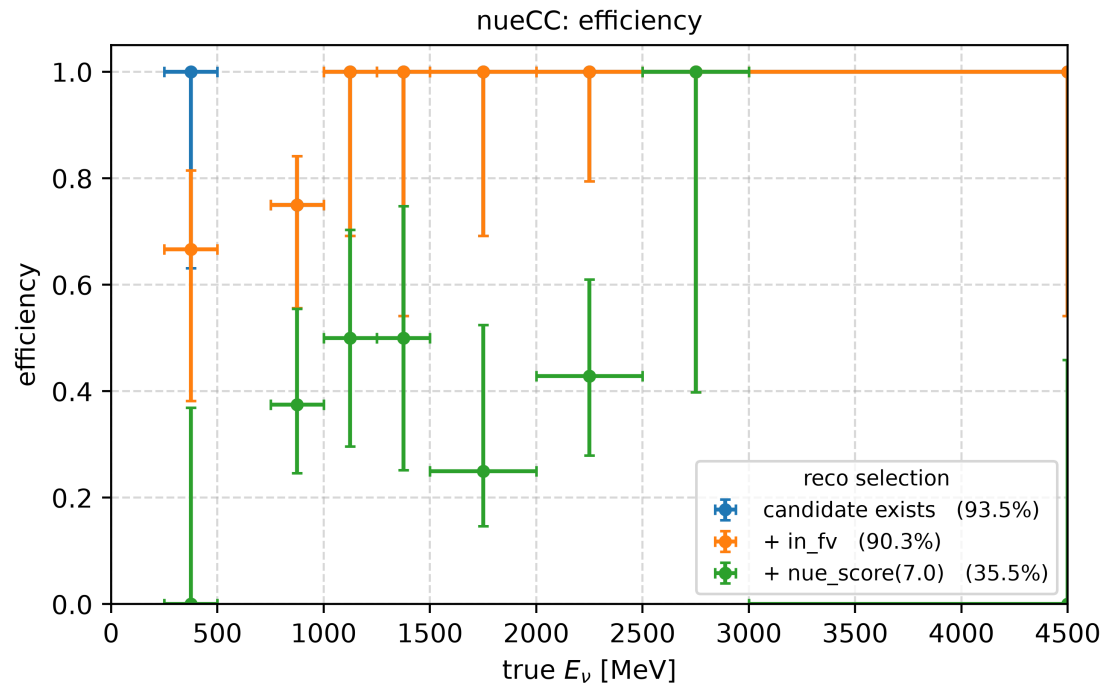
numuCC: signal = true: in_fv & edep_min(100) & numucc



Selection cuts ($\nu_e CC$)

Assuming the updated signal definition (edep>75 MeV & vertex in FV), looking at the progressive selection cuts.

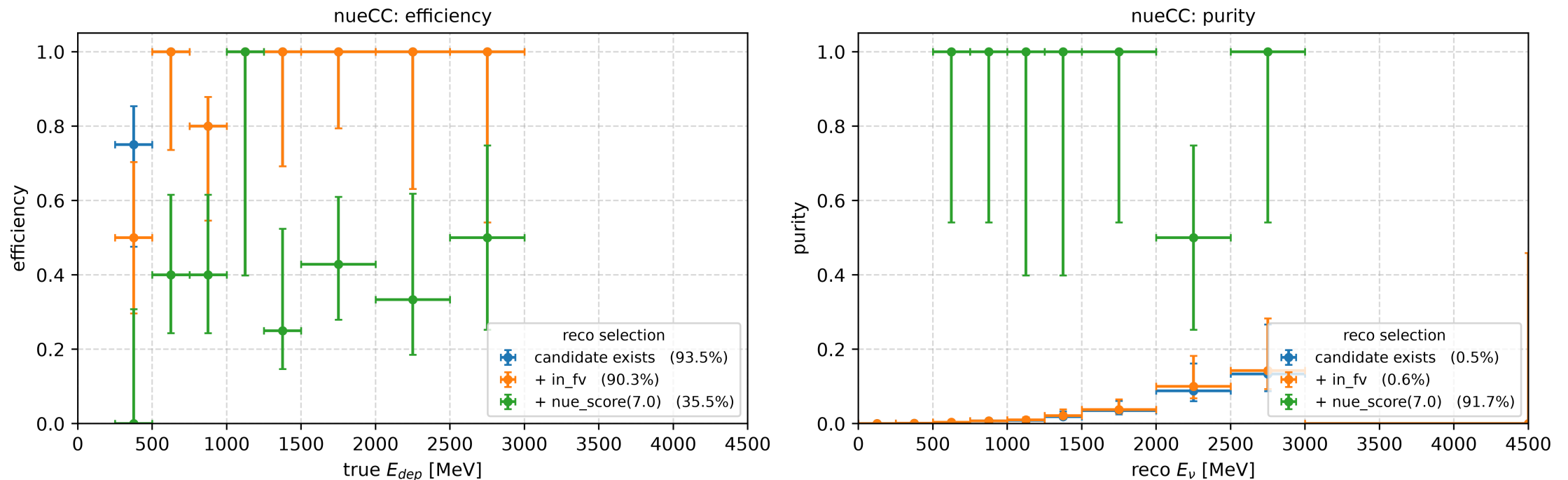
nueCC: signal = true: in_fv & edep_min(75) & nuecc



Selection cuts ($\nu_e CC$)

Assuming the updated signal definition (edep>75 MeV && vertex in FV), looking at the progressive selection cuts.

nueCC: signal = true: in_fv & edep_min(75) & nuecc



Selection cuts

Signal for $\nu_\mu CC$: edep > 100 MeV && vertex in FV

Progressive cuts	Efficiency	Purity
Candidate exists	3,265/3,500 = 93.3%	3,265/6,076 = 53.7%
+ vertex in FV	3,192/3,500 = 91.2%	3,192/4,605 = 69.3%
+ numu_score > 0.9	2,647/3,500 = 75.6%	2,647/2,799 = 94.6%

Signal for $\nu_e CC$: edep > 75 MeV && vertex in FV

Progressive cuts	Efficiency	Purity
Candidate exists	29/31 = 93.5%	29/6,076 = 0.5%
+ vertex in FV	28/31 = 90.3%	28/4,605 = 0.6%
+ nue_score > 7.0	11/31 = 35.5%	11/12 = 91.7%

Things to do:

1. Beam spill time cut?

- Candidate bundle already includes [0.2, 2.2] us cut --> possibly cutting too much if sim is slightly shifted (spill appears to be [0, 1.6] us?)
- Wire-Cell PR might need to treat data/MC differently

2. Understand truth/reco matching with distance cut

- It mixes reco quality (vertex) with truth matching --> noticed SPINE also uses charge cluster overlap instead

3. Looking how to include data for proper comparisons + off-beam for purity.

Back-up

Matching strategy

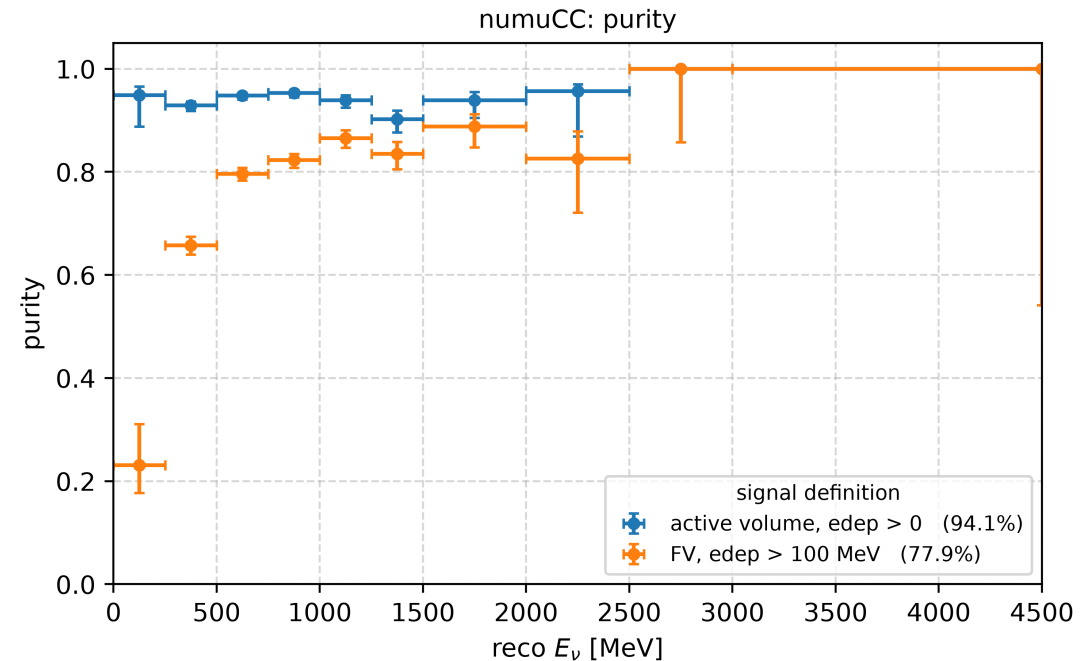
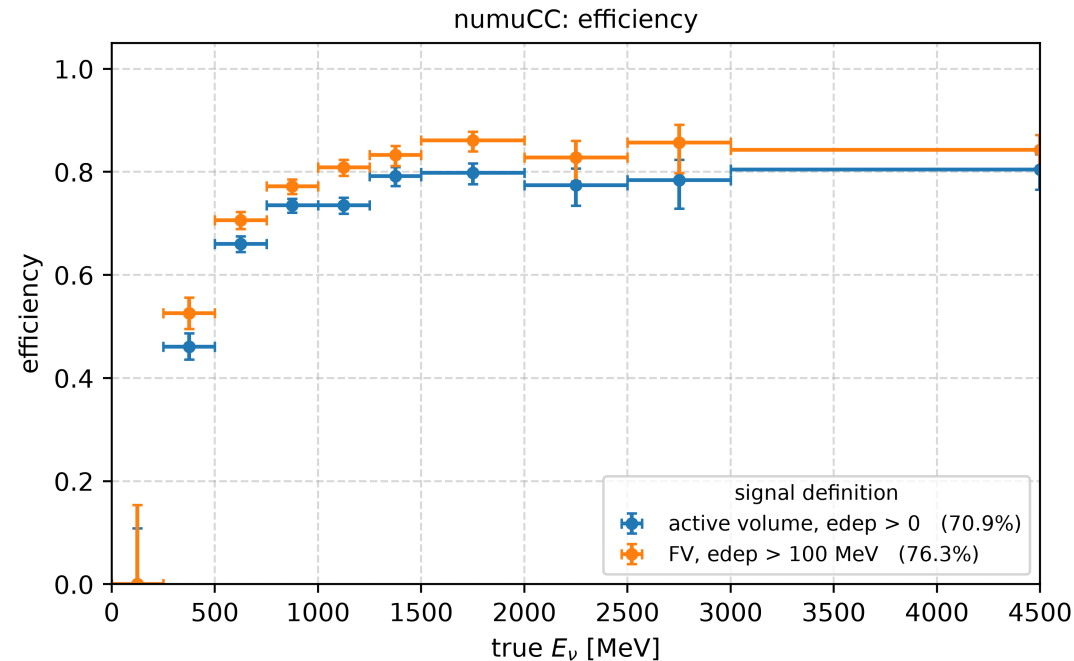
- Is this reco candidate correct? → **purity**
 - Take the true signal neutrino closest to it (vertex distance).
 - The candidate is correct only if that neutrino also points back to it and it belongs to the same class.
 - Two-way match, so one neutrino doesn't get matched to 2 bundles
 - If the event has no signal neutrino, it is not correct.
- Is this true signal neutrino reconstructed? → **efficiency**
 - Take the reco candidate closest to it.
 - The neutrino counts as reconstructed only if that candidate also points back to it and passes the nuscore selection.
 - Two-way match, so one candidate doesn't get matched to 2 ν 's
 - If the event has no reco candidates, it is a miss.

Signal definitions ($\nu_\mu CC$)

Keeping the previous reco selection (candidate exists + numu_score), with different signal definitions.

Signal	Efficiency	Purity
Previous (edep>0 && inAV)	3,226/4,551 = 70.9%	3,226/3,427 = 94.1%
New (edep>100 && inFV)	2,752/3,610 = 76.2%	2,752/3,427 = 80.3%

numuCC: reco selection = reco: numu_score(0.9); match = nearest matchable ν

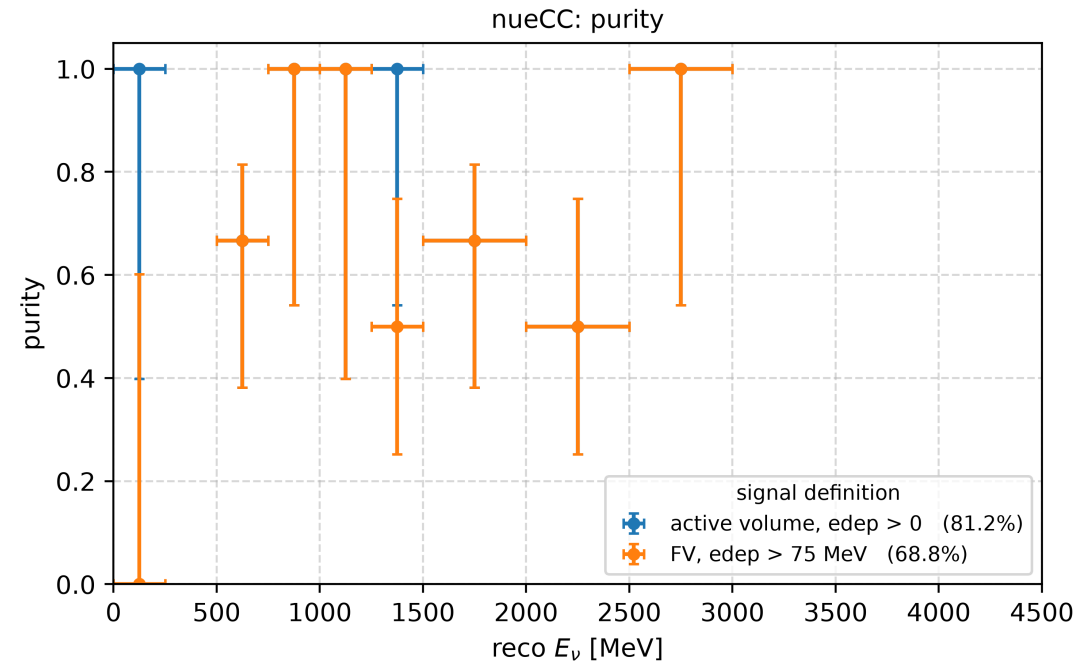
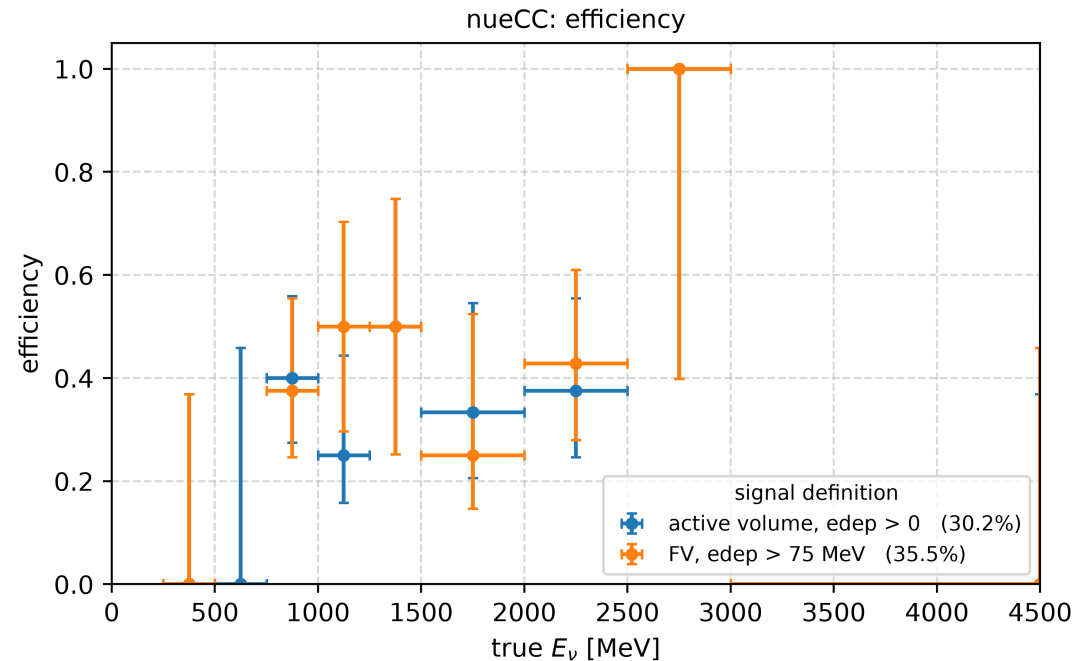


Signal definitions ($\nu_e CC$)

Keeping the previous reco selection (candidate exists + nue_score), with different signal definitions.

Signal	Efficiency	Purity
Previous (edep>0 && inAV)	13/43 = 30.2%	13/16 = 81.2%
New (edep>100 && inFV)	12/34 = 35.3%	12/16 = 75.0%

nueCC: reco selection = reco: nue_score(7.0); match = nearest matchable ν

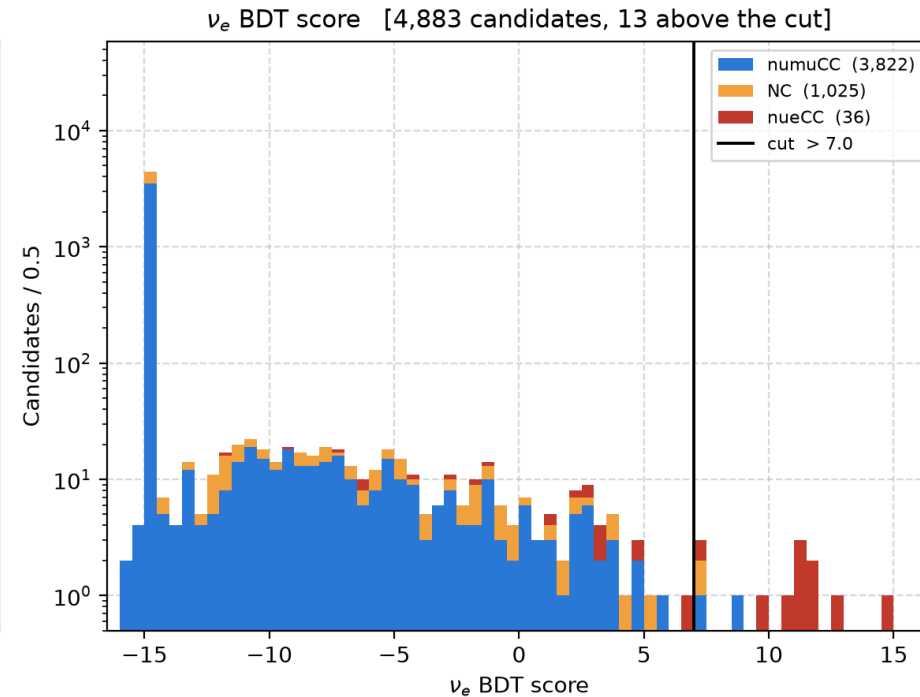
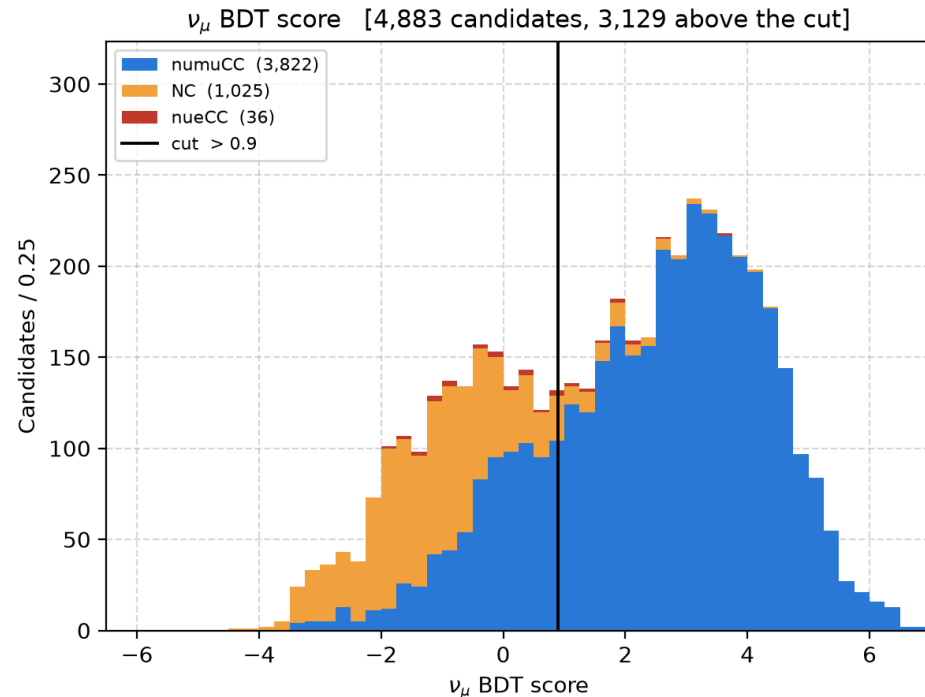


Numu/nue BDT scores

- Given a candidate, reco class is assigned according to the score.
 - These are independent selections, not a classifier.
 - A candidate can pass both or neither

For these plots: using
unambiguous truth-reco matches
(1 signal nu, 1 reco candidate)

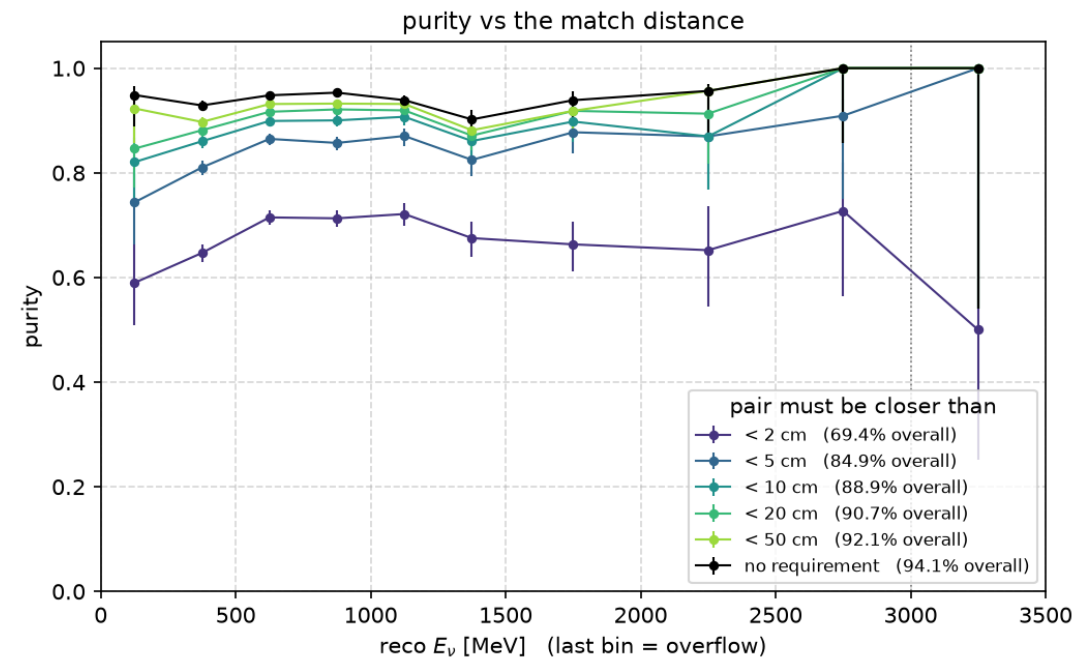
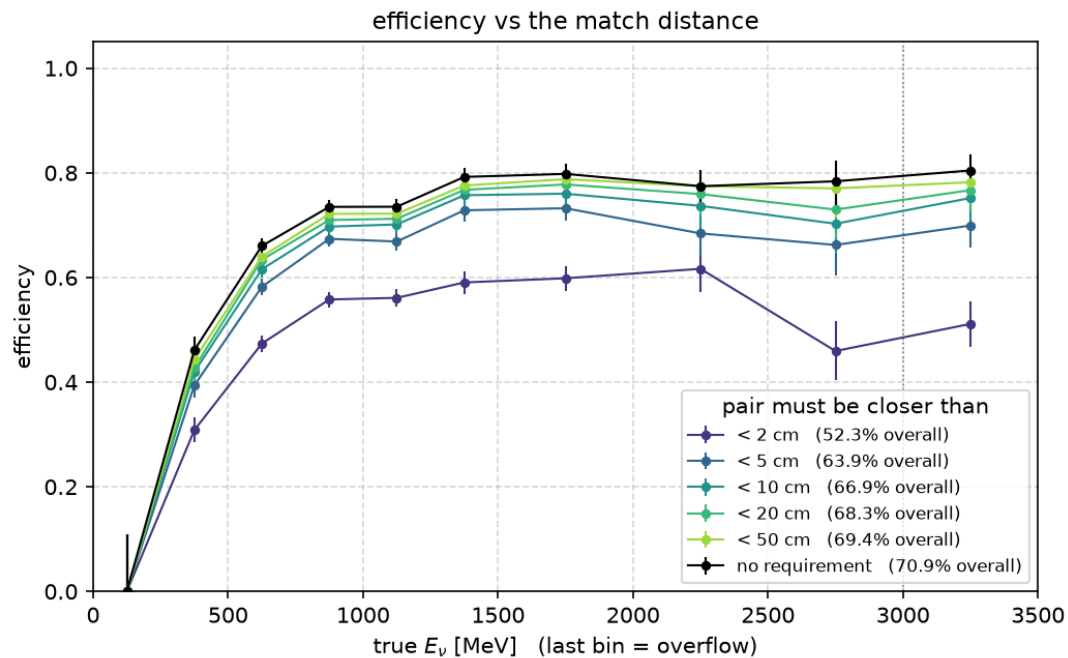
BDT scores by true class — one signal ν , one candidate, matched one-to-one



Aside: match quality

- If you define a max match distance, efficiency/purity drop (flat over energy) → not a selection cut, rather a definition.

ν_μ selection (numu_score > 0.9)



True neutrino time

- Flat in beam spill [0,1.6] μ s, except for ~5 peaks?
- (No 18ns bunch structure, but we don't really care)

