2-LEVEL GRAPHS FOR MUON-TOMOGRAPHY

Giles Strong

5th IML Workshop, CERN - 13/05/22

giles.strong@outlook.com

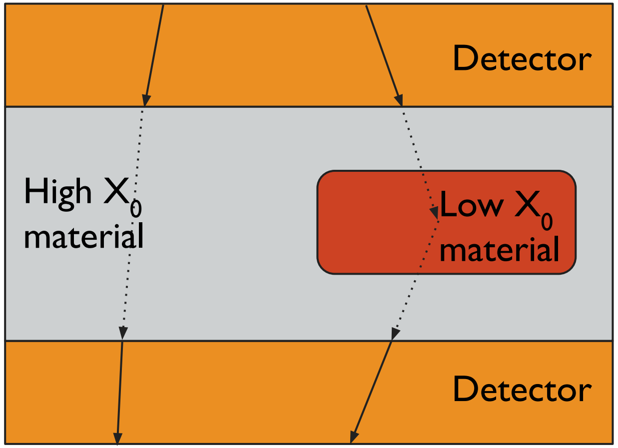
twitter.com/Giles_C_Strong

gilesstrong.github.io/website

github.com/GilesStrong

TOMOGRAPHY VIA MULTIPLE SCATTERING

- Consider a volume with unknown composition
  - E.g. Shipping container, archeological site, nuclear waste, industrial machinery
  - Want to infer properties of the volume:
    - E.g. build a 3D map of elemental composition
- Cosmic muons scattered by volume according to radiation-length (X_0 [m]) of elements in material
  - Measure muons above and below volume
  - Kinematic changes provide info on material composition



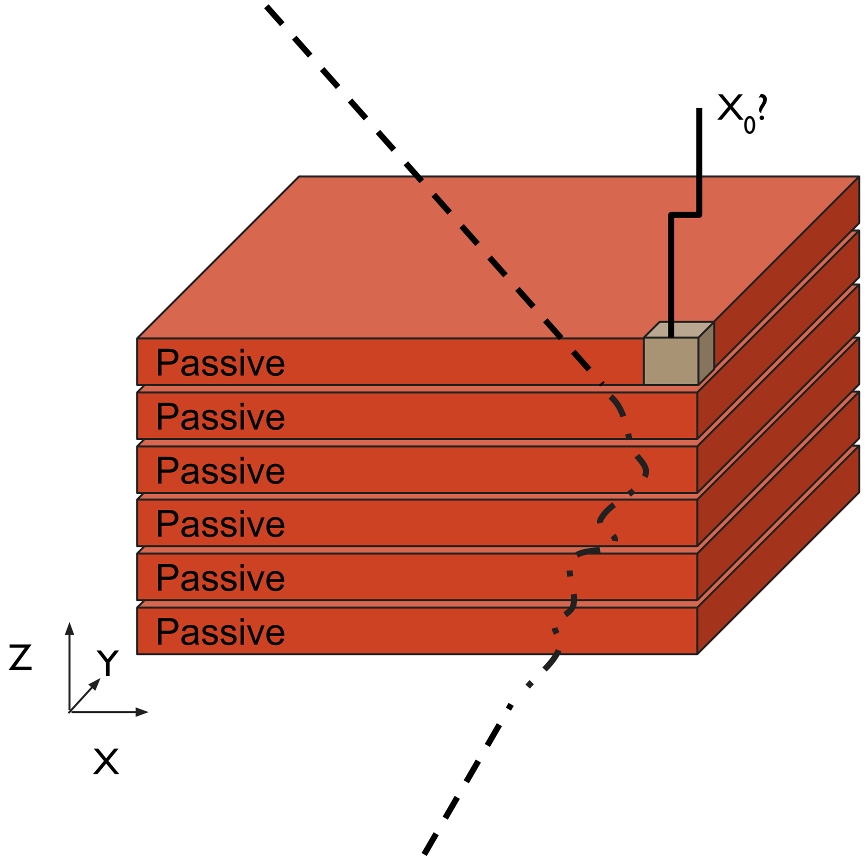
High X_0 = low scattering

Low X_0 = high scattering

X_0 = average distance between scatterings

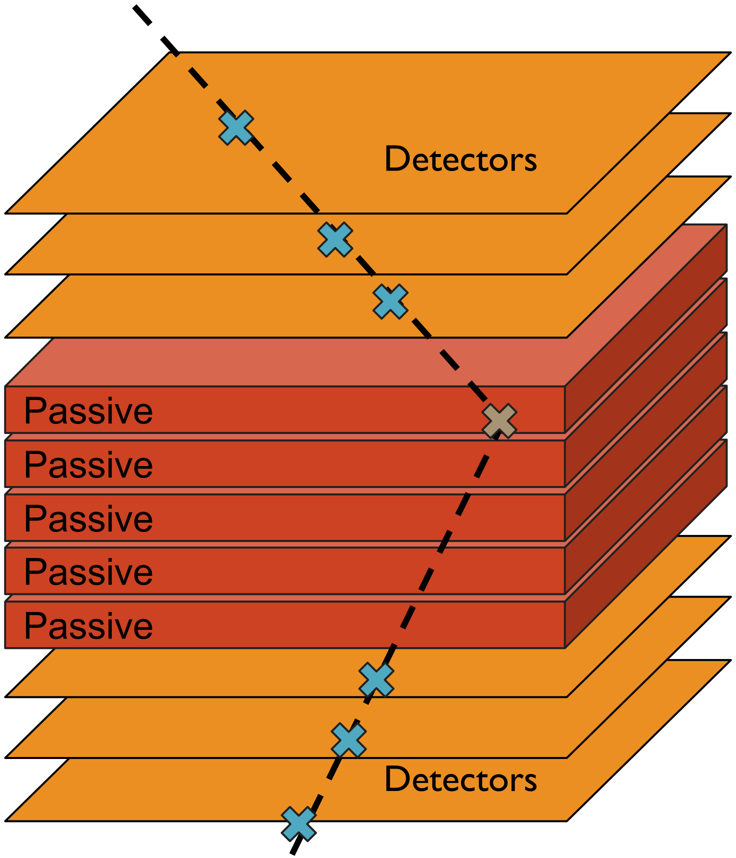
VOXELISED VOLUMES

- Typically, we can split the unknown volume into voxels
- We can then observe the scatterings of many muons through the volume
- Our aim is then to estimate the X_0 of every voxel



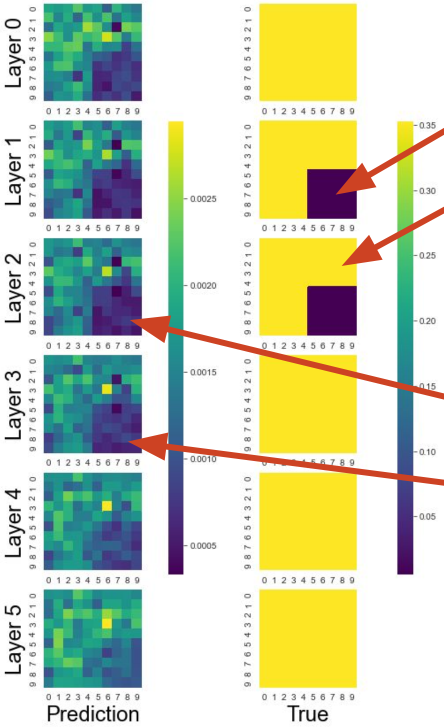
INFERENCE METHODS

- We can fit straight-line trajectories to muon hits
  - These provide the initial & final muon kinematics
- Point of Closest Approach (POCA) method assigns the entire scattering to a single voxel
  - The X_0 can be computed by inverting analytic scattering models (e.g. PDG)
  - The voxel is chosen by extrapolating the trajectories inside the volume and finding their point of closest approach



POCA BIAS

- We know, though, that the muon scattering results from multiple interactions throughout the volume
- Assigning the whole scattering to a single point inherently leads to underestimating the X_0
- Can deep-learning approaches be used for inference?
  - Could do UNet-style image segmentation, or [bin/cluster hits into MVA]
  - But these methods lose out on low level information



- Block of lead (X_0=0.005612m)
- Surrounded by beryllium (X_0=0.3528m)
- Predictions highly biased to underestimate X_0
- Lead block clearly visible
- but high z uncertainty in scatter location causes 'ghosting' above and below

PROBLEM BREAKDOWN

- We have a population of muons, $\underline{\mu}$, and know their start & end trajectories
  - We can also compute high-level features, e.g. delta angles, POCA X_0, etc.
- We have a set of voxels, $\underline{V}$, know their (xyz) positions, and want to know their X_0s
- The final muon-trajectories have a stochastic dependence on the voxels
- But each muon only passes through a few voxels
- How can we map $\underline{\mu} \rightarrow \underline{X}_0$?

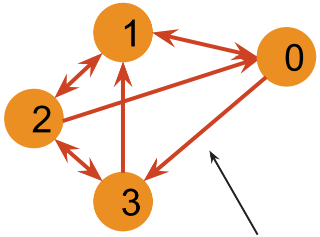
$\Delta\theta_x$	$\Delta\theta_y$	Δx	Δy	X0	X	Y	Z
2.4e-03	2.4e-03	1.3e-03	1.4e-03	2.6e-01	-1.9e-02	-3.49-02	1.0e-01
7.3e-04	4.9e-03	-8.8e-04	1.3e-04	1.2e-01	3.0e-02	-2.4e-02	1.4e-01
…	…	…	…	…	…	…	…

$$\downarrow \; \underline{X}_0 \cong f(\underline{\mu}, \underline{V})$$

X	Y	Z	X0
0	0	0	?
0.1	0	0	?
…	…	…	?

GRAPH MOTIVATION

- Consider each voxel as a node in a graph
  - Our task would be to predict the X_0 of each node
  - The muons provide features for each voxel
- But how can we best express the muons as voxel features?
  - None of the muons pass through all voxels:
    - The muon representation should be tailored to each voxel
  - We only have start & end features for the muons:
    - Voxels need to be aware of other voxels, in order to properly "transport" the muons through the volume

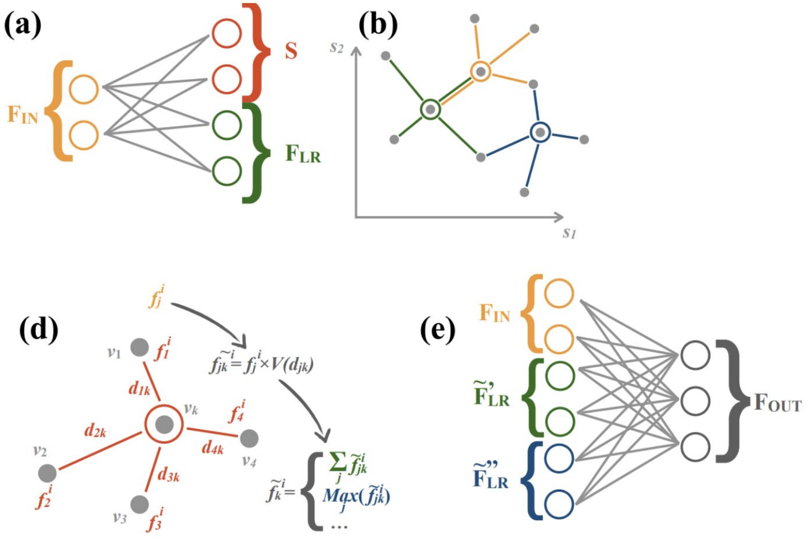


Voxel 0 has features dependent on some of the muons and the some of the other voxels. From this we predict its X_0

We need an architecture that can automatically learn to connect nodes

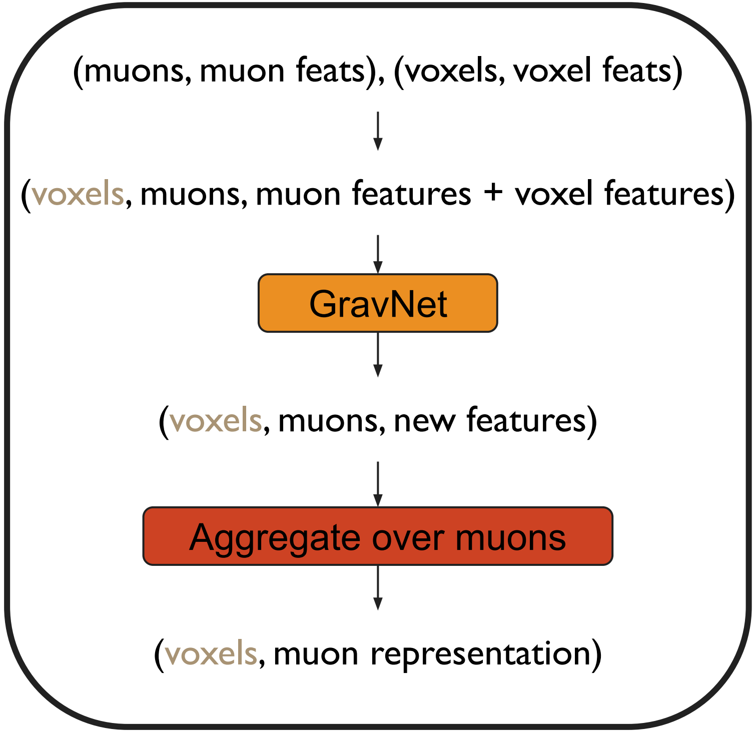
GRAVNET

- Quasim, Kieseler, Iiyama, & Pierini, 2019
- Rewires graphs by learning node coordinates *S* in latent space and selecting *k* nearest-neighbours
- Computes new features per node, which depend on:
  - The node's original features and only the features of the k-nearest nodes
  - A potential term $V(d_{jk})$ allows the neighbour features to be augmented according to separation distance in clustering space *S*
- GravNet fulfills our requirement information sharing along learnable connections



MUON REPRESENTATION

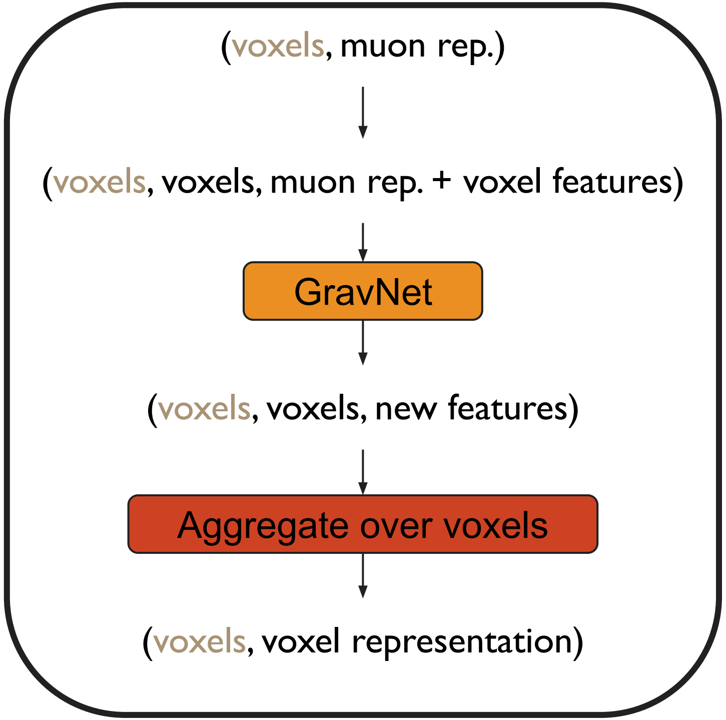
- Aim: learn a dedicated representation of the muons for each voxel
  - Each voxel has its own graph in which muons are the nodes
- Augment the muons by adding the xyz position of the voxel
  - These can be combined with muon features (entry/exit points, POCA locations, and angles)
- Use GravNet to pass information between muons
  - "Selects" muons relevant to the given voxel
- Finally, aggregate the graph by taking permutation-equivariant operations (mean, max, etc.)
  - NB: can also include extra DNNs and self-attention in the aggregation



GravNet is broadcast over the voxels "batch" dimension

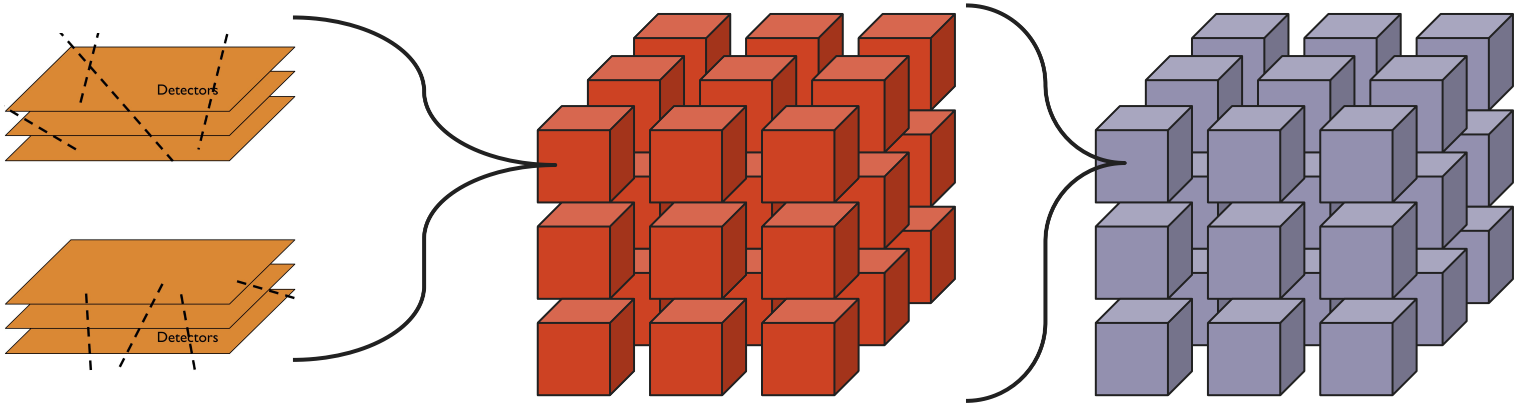
VOXEL REPRESENTATION

- Aim: learn a dedicated representation of the voxels for each voxel
  - Each voxel has its own graph in which voxels are the nodes
  - Allows voxels to adjust their muon representations to account for muon transport through the volume
- Again, augment each voxel's muon representation by adding the xyz distances of the voxels to the voxel in question
- Again, use GravNet and aggregation to arrive at a voxel representation per voxel



GravNet is broadcast over the voxels "batch" dimension

REPRESENTATION SUMMARY



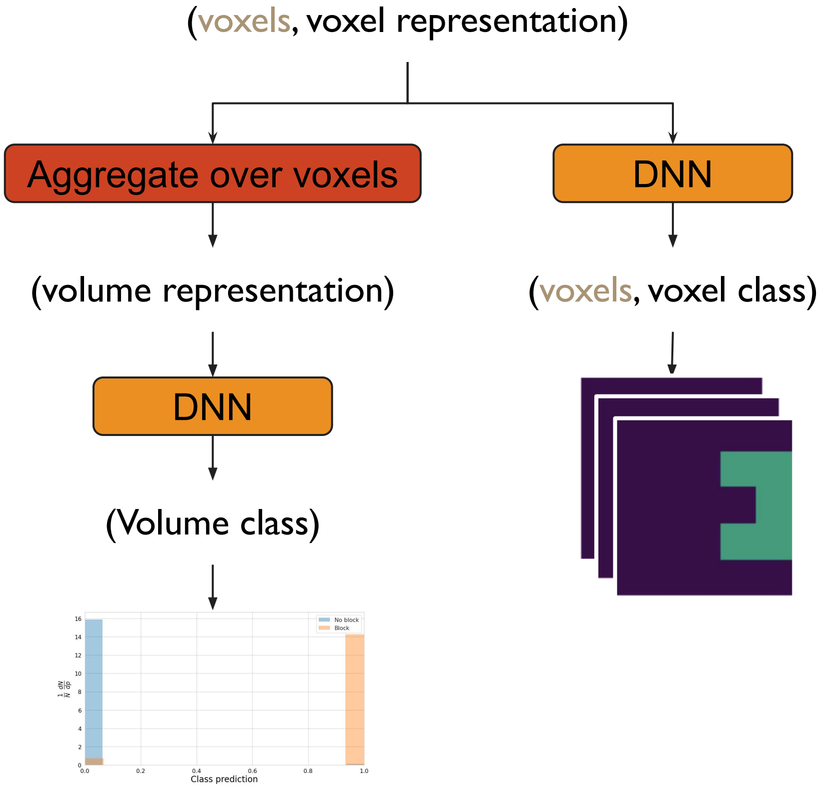
1. Pass many muons through volume

2. For every voxel, construct a latent representation of the relevant muons.

3. Every voxel then refines its representation based on the surrounding voxels.

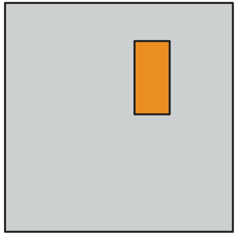
VOLUME PREDICTION

- At this point, we have a representation per voxel.
- We can transform these into X_0 predictions (class/value) with a DNN
- But sometimes we don't need to produce a 3D image
    - Consider scanning a shipping container for dangerous materials (e.g. uranium): we are more interested in determining **presence** rather than **location** - binary classification
- We can easily aggregate over the voxels to produce a volume representation.
    - This can then be further transformed into the appropriate prediction shape

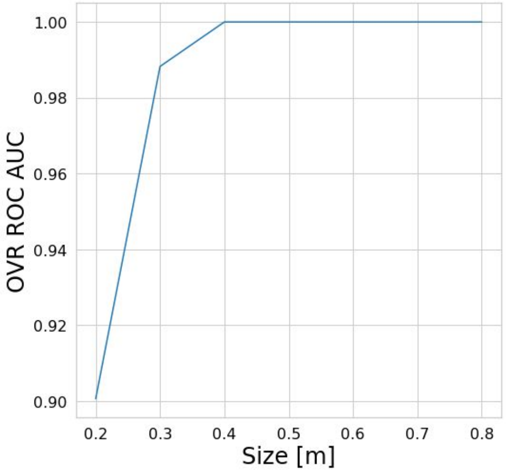


TESTING: BINARY CLASSIFICATION

WORK IN PROGRESS!



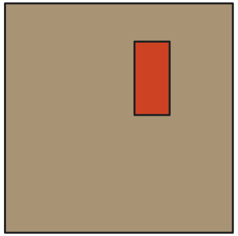
Beryllium volume (1m^3)
may contain lead block
(random size and position)
<u>Is a block of lead present?</u>



GNN archives excellent classification power, even for small blocks

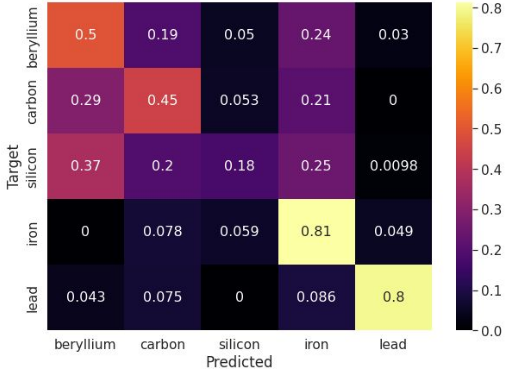
WORK IN PROGRESS!

TESTING: MULTICLASS CLASSIFICATION

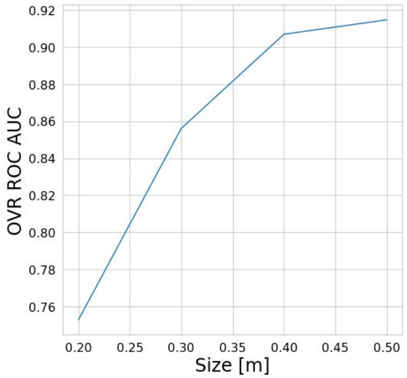


Volume (1m^3) of random material, contains a block of a different random material (random size and position)

<u>What material is the block made from?</u>



Better performance for denser materials, presumably due to more scattering.
(Results average over volume material and block size)

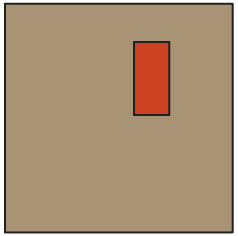


Performance again increases with block size
(Results average over block/volume material pairs)

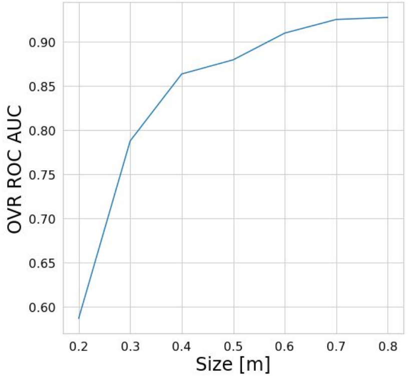
We could also treat this as an X_0 regression…

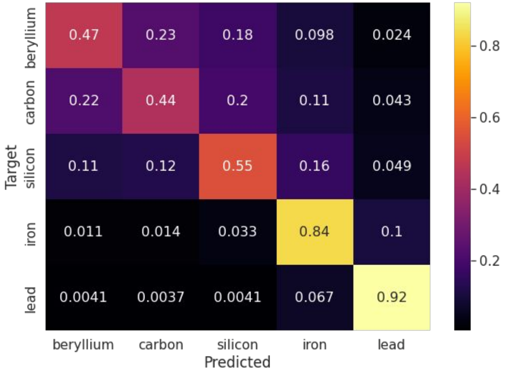
WORK IN PROGRESS!

TESTING: IMAGING



Volume (1m^3) of random material, contains a block of a different random material (random size and position)
<u>What material is each voxel made from?</u>





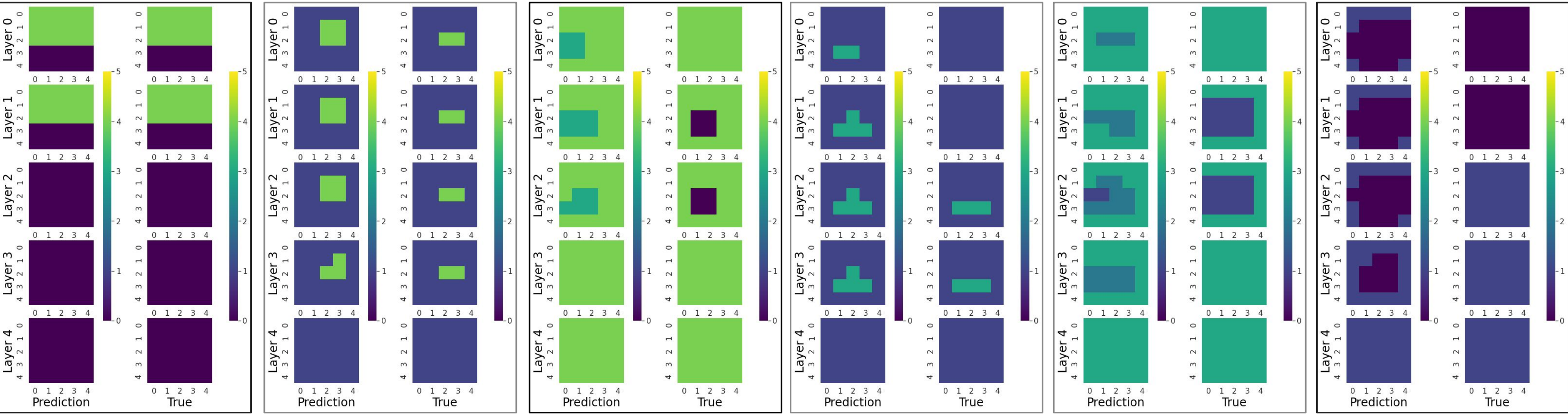
Improved symmetry for confusion matrix.
(Results average over volume material and block size)

Performance again increases with block size. Block/volume material pairs)

We could also treat this as an X_0 regression…
Results shown are computed using only voxels inside the blocks. Predictions for volume voxels is much easier.

WORK IN PROGRESS!

IMAGING BEHAVIOUR EXAMPLES



Perfect prediction

Good position & materials, but slightly too large

Good position but wrong materials

Low z-position accuracy

Predicts > 2 materials

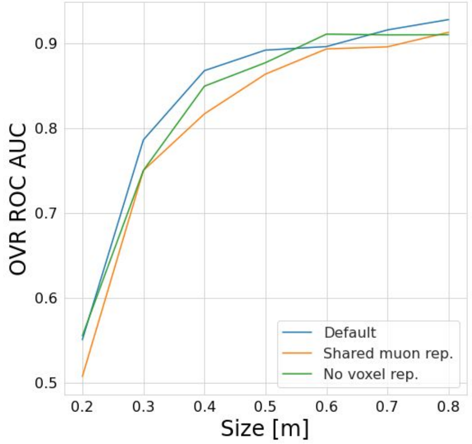
Predicts irregular block shapes

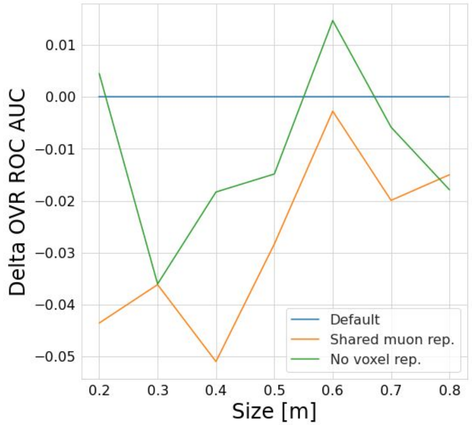
Both imply that the GNN isn’t exploiting too much knowledge about the training scenario

WORK IN PROGRESS!

PRELIMINARY ABLATION

- Shared muon representation:
  - Only compute one muon representation and pass it to all voxels
  - Slight, but consistent, drop in performance = useful for voxels need to have their own dedicated reps.
- No voxel representation:
  - Voxel predictions are based solely on their muon representations
  - No mechanism for voxels to refine their features based on other voxels
  - Generally slightly lower performance = voxel refinement somewhat useful, but could be improved
  - NB: No attempt to recover parameter count to match default model.



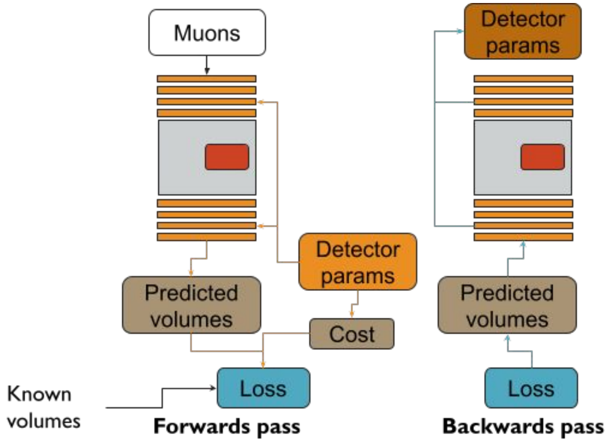


SUMMARY

- Demonstrated that the use of problem-specific intermediate latent-representations can allow us to fully exploit low-level information
    - Such a representation can be easily adapted to suit a variety of end-goals
- In muon-tomography, this allows us to achieve promising performance on several examples
    - The 2-level graph also allows us to include high-level features at both the muon and voxel level; the GNN is fully complementary to existing approaches
    - It also allows us to differentiably compute class probabilities rather than relying on X_0 float predictions or task-specific summary statistics

THE WIDER PICTURE

- Simulation for this work used the MODE TomOpt package
  - Being developed to provide differential optimisation of muon tomography detectors
  - See Section 4.3 of the MODE whitepaper for more details arXiv:2203.13818
  - Contributors:
    - Giles Strong, Tommaso Dorigo, Andrea Giammanco, Pietro Vischia, Jan Kieseler, Maxime Lagrange, Federico Nardi, Haitham Zaraket, Max Lamparth, Federica Fanzago, Oleg Savchenko, Nitesh Sharma, & Anna Bordignon.
  - Interested in helping? giles.strong@outlook.com



Aim is to learn optimal detector layouts given a task and budget.
Inference algorithms must be both differentiable and detector-agnostic…

BACKUPS

GRAPH NEURAL NETWORKS

- Graph = collection of *nodes* with features connected along *edges*
  - <u>Nodes have no assumed ordering</u>
- Potential tasks:
  - Predict target features per node
  - Predict whether edges exist
  - Predict target features of entire graph
- Simple approach: apply same DNN to each node to learn target features
  - Compute average of features over nodes for graph-level targets
- Better approach: DNN also takes into account connected nodes
  - GNNs have a *message passing* mechanism to allow node-level predictions to be influenced by other nodes