TrackOpt vertex with GNN

V. Kostyukhin
Siegen university

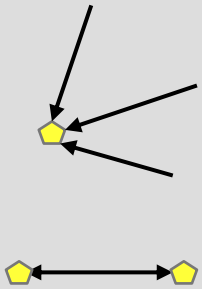
Edge weights with DGL

Universität Siegen

Edge weights estimation is implemented in DGL

1) Node_hidden_state = NN{Concat(Node_features, Mean(Edge_features)}

2) Edge_weight = NN{ Concat(Node_hidden_state_i, Edge_features, Node_hidden_state_j)}



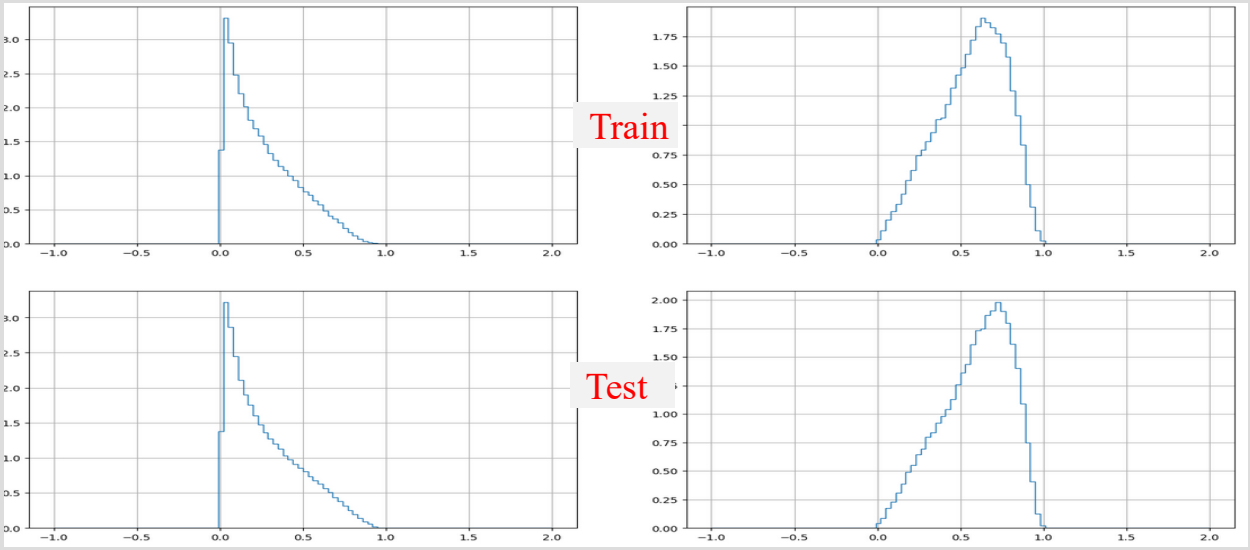
3) Node_hidden_state = NN{Concat(Node_features, Weighted_Mean(Edge_features)}
4) Upd_Edge_weight = NN{ Concat(Node_hidden_state_i, Edge_features, Node_hidden_state_j)}

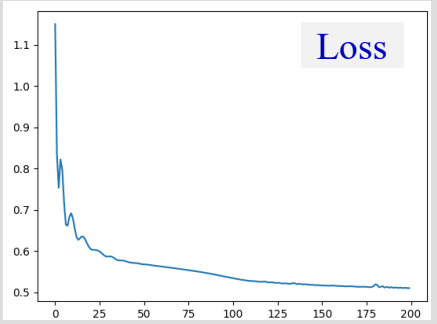
Steps (3),(4) happened to be not needed, final weights are practically the same.

Loss – binary cross-entropy (classification)
Activations – Mish + Sigmoid to get probability

DGL results (preliminary)

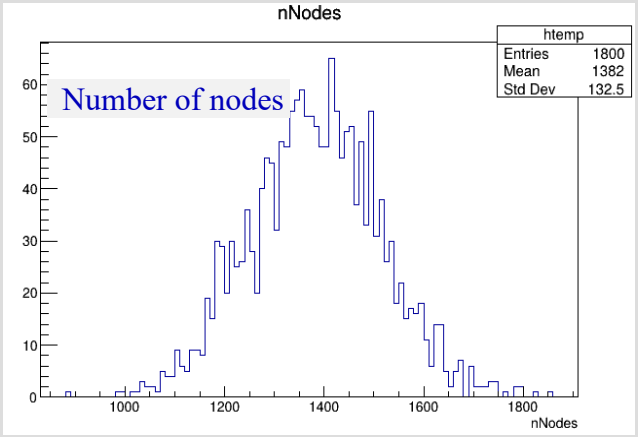
Training: 200 epoch, 100 graphs(events) batch train/test
loss= 0.5098(train), 0.5080(test) – no overtraining

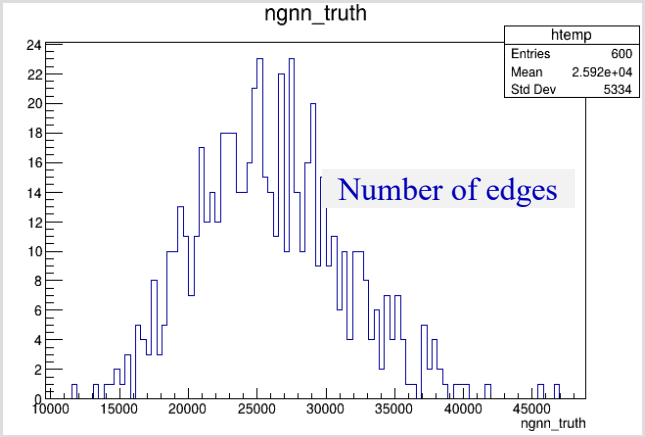




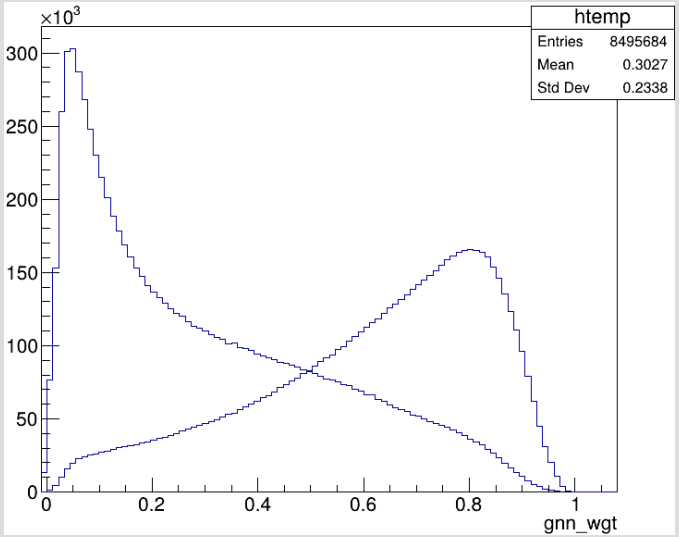
AUC(train)= 0.8265. AUC(test)= 0.8294 for primary+secondary edges. AUC(SVonly)=0.848

For comparison - XGBoost classification of edges based on the same edge features: AUC ~ 0.75



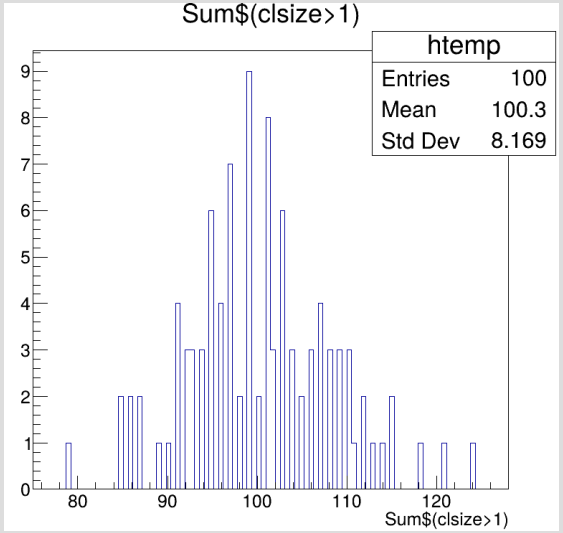


LMC after DGL

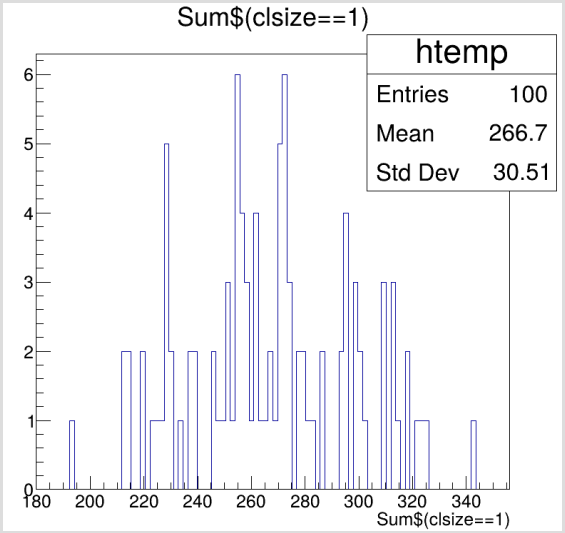


For LMC: weight=WGT_{GNN}-0.5

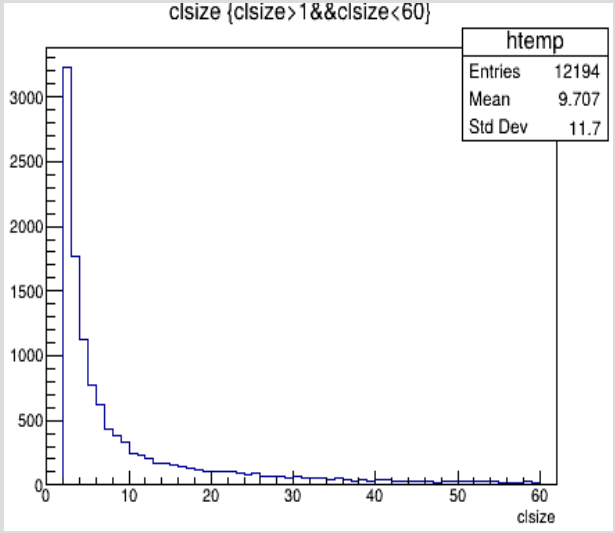
$N_{cluster}$ (size>1)



N_{single_track}

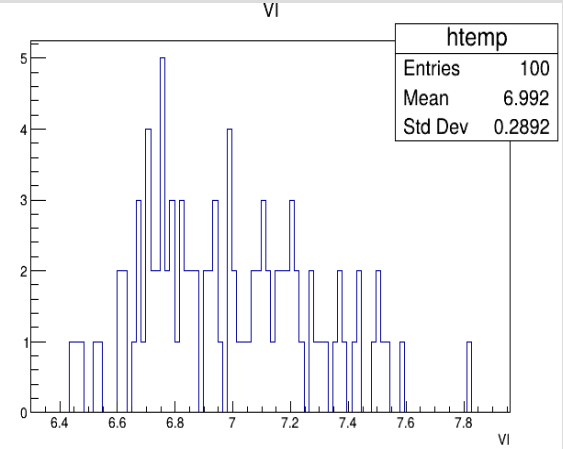


Cluster size

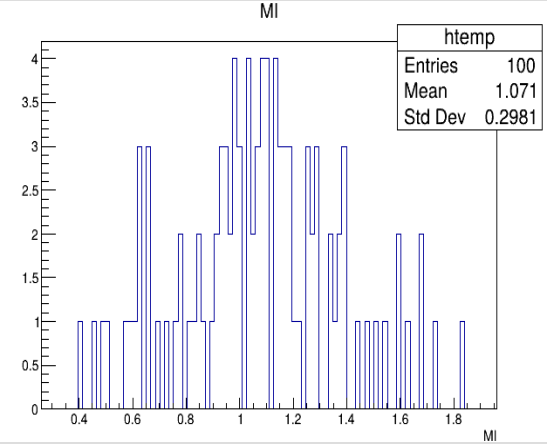


LMC metrics

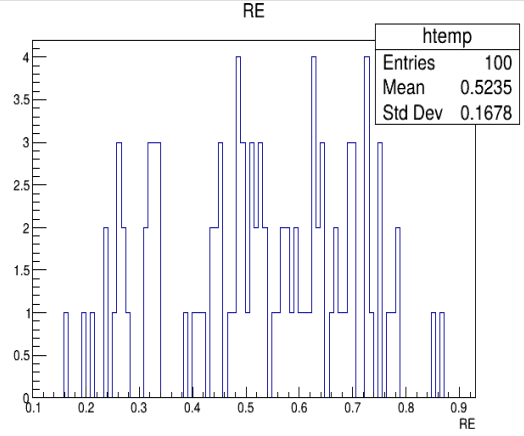
Variation of information



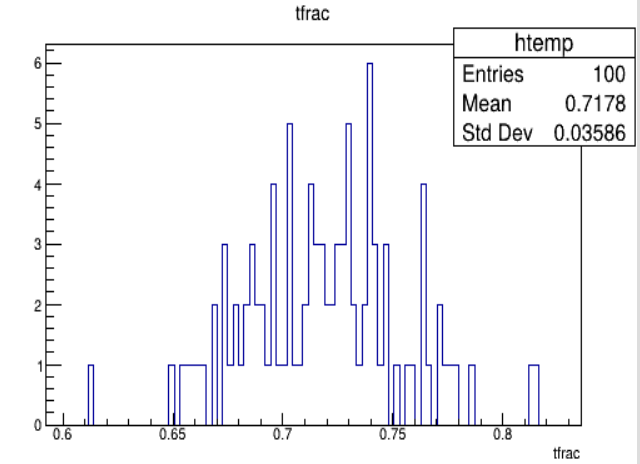
Mutual information



Rand index



Fraction of correctly resolved true edges



Conclusions and next steps

1) Basic machinery for data processing is created and works, although requires polishing/optimisation
2) R&D on GNN layer versions, LMC parameters, constraints, informative problem-dependent metrics, etc. can be started.
3) Constraints – identify fake edges connecting tracks from PV and SV.
   - ❖ First try – bad AUC=0.55. More work is needed.
4) Current technical problems(work in progress):
   a. DGL doesn't export GNN models directly (ONNX?).
      PyTorch backend/export?
   b. UPROOT buffer sizes – limit number of graphs for saving
   c. …

LMC problem

Starting energy: 2097.73

Iter	Total decrease	Pair updates	New sets	Num. of sets
1	2287.43	2286.65	0.784414	515
2	242.296	241.122	1.17428	379
3	55.2082	54.6484	0.559751	348
4	12.1379	11.8518	0.286146	347
5	11.8605	11.5254	0.33507	345
6	12.7867	12.4791	0.307552	323
7	4.79566	4.50951	0.286146	320
8	1.23263	0.94648	0.286146	320
9	2.78869	2.4156	0.373089	320
10	4.56306	4.27692	0.286146	320
11	1.23263	0.94648	0.286146	320
12	2.78869	2.4156	0.373089	320
13	4.56306	4.27692	0.286146	320
14	1.23263	0.94648	0.286146	320
15	2.78869	2.4156	0.373089	320
16	4.56306	4.27692	0.286146	320
17	1.23263	0.94648	0.286146	320
18	2.78869	2.4156	0.373089	320
19	4.56306	4.27692	0.286146	320
20	1.23263	0.94648	0.286146	320
21	2.78869	2.4156	0.373089	320
22	4.56306	4.27692	0.286146	320
23	1.23263	0.94648	0.286146	320
24	2.78869	2.4156	0.373089	320
25	4.56306	4.27692	0.286146	320
26	1.23263	0.94648	0.286146	320
27	2.78869	2.4156	0.373089	320
28	4.56306	4.27692	0.286146	320
29	1.23263	0.94648	0.286146	320
30	2.78869	2.4156	0.373089	320
31	4.56306	4.27692	0.286146	320
32	1.23263	0.94648	0.286146	320
33	2.78869	2.4156	0.373089	320
34	4.56306	4.27692	0.286146	320
35	1.23263	0.94648	0.286146	320
36	2.78869	2.4156	0.373089	320
37	4.56306	4.27692	0.286146	320
38	1.23263	0.94648	0.286146	320
39	2.78869	2.4156	0.373089	320
40	4.56306	4.27692	0.286146	320
41	1.23263	0.94648	0.286146	320
42	2.78869	2.4156	0.373089	320
43	4.56306	4.27692	0.286146	320
44	1.23263	0.94648	0.286146	320
45	2.78869	2.4156	0.373089	320
46	4.56306	4.27692	0.286146	320
47	1.23263	0.94648	0.286146	320
48	2.78869	2.4156	0.373089	320
49	4.56306	4.27692	0.286146	320
50	1.23263	0.94648	0.286146	320
51	2.78869	2.4156	0.373089	320
52	4.56306	4.27692	0.286146	320
53	1.23263	0.94648	0.286146	320
54	2.78869	2.4156	0.373089	320
55	4.56306	4.27692	0.286146	320
56	1.23263	0.94648	0.286146	320
57	2.78869	2.4156	0.373089	320
58	4.56306	4.27692	0.286146	320
59	1.23263	0.94648	0.286146	320
60	2.78869	2.4156	0.373089	320
61	4.56306	4.27692	0.286146	320
62	1.23263	0.94648	0.286146	320
63	2.78869	2.4156	0.373089	320
64	4.56306	4.27692	0.286146	320
65	1.23263	0.94648	0.286146	320
66	2.78869	2.4156	0.373089	320
67	4.56306	4.27692	0.286146	320
68	1.23263	0.94648	0.286146	320
69	2.78869	2.4156	0.373089	320
70	4.56306	4.27692	0.286146	320
71	1.23263	0.94648	0.286146	320
72	2.78869	2.4156	0.373089	320
73	4.56306	4.27692	0.286146	320
74	1.23263	0.94648	0.286146	320
75	2.78869	2.4156	0.373089	320
76	4.56306	4.27692	0.286146	320
77	1.23263	0.94648	0.286146	320
78	2.78869	2.4156	0.373089	320
79	4.56306	4.27692	0.286146	320
80	1.23263	0.94648	0.286146	320
81	2.78869	2.4156	0.373089	320
82	4.56306	4.27692	0.286146	320
83	1.23263	0.94648	0.286146	320
84	2.78869	2.4156	0.373089	320
85	4.56306	4.27692	0.286146	320
86	1.23263	0.94648	0.286146	320
87	2.78869	2.4156	0.373089	320
88	4.56306	4.27692	0.286146	320
89	1.23263	0.94648	0.286146	320
90	2.78869	2.4156	0.373089	320
91	4.56306	4.27692	0.286146	320
92	1.23263	0.94648	0.286146	320
93	2.78869	2.4156	0.373089	320
94	4.56306	4.27692	0.286146	320
95	1.23263	0.94648	0.286146	320
96	2.78869	2.4156	0.373089	320
97	4.56306	4.27692	0.286146	320
98	1.23263	0.94648	0.286146	320
99	2.78869	2.4156	0.373089	320
100	4.56306	4.27692	0.286146	320