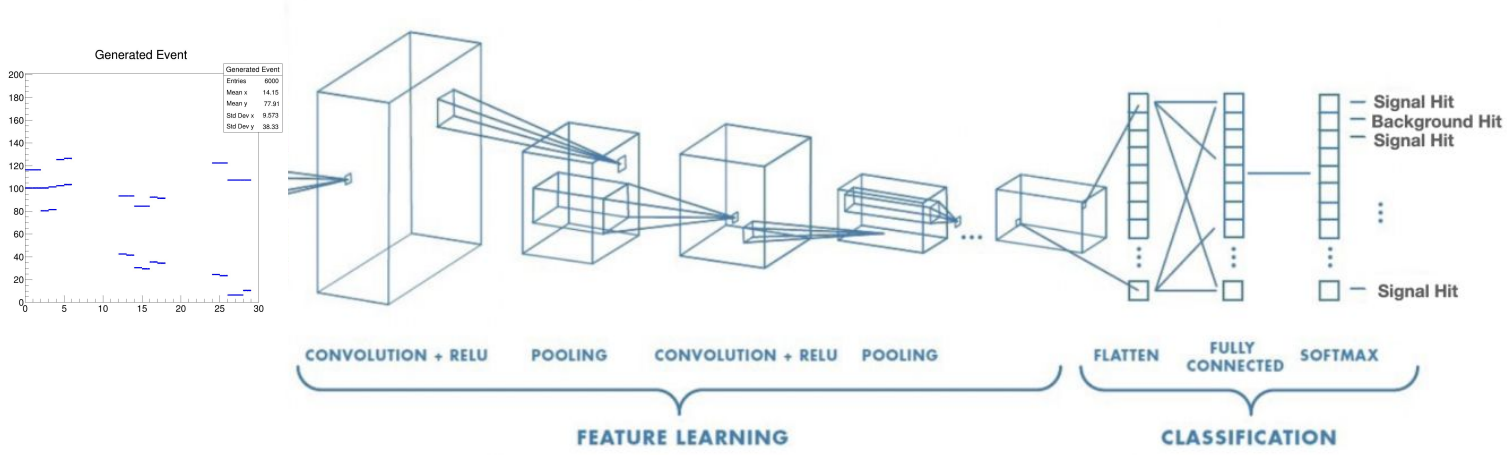
Update 5/3

Arthur Conover

Machine Learning Track Finding

- Using cleaned data with noise hits removed, detect which detectors hits correspond to full track.
- Corresponds to track finding in KTracker, attempts to do in one step.
- Not using full background, previous neural network filter removes hits.



Single Muon Case

- Inject single muon track into event, have convolution neural network find which hits correspond.
- Much simpler than finding two tracks
- Works very well, more than ⅔ of hits are accurately assigned, “misses” are generally 1-2 element IDs off.

Detector	**1**	**2**	**3**	**4**	**5**	**6**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**
Truth	99	99	71	71	84	85	35	34	29	28	40	39	12	12	5	5	19	20	0	0	0	0	0	0
Output	99	98	71	72	84	86	35	34	29	29	40	39	13	13	5	5	19	20	0	0	0	0	0	0

Dimuon Case

- Inject dimuon tracks into event, have convolution neural network determine which hits correspond.
- More complicated as network has to separate two tracks.
- Doesn’t work quite as well, around 50% accuracy, but generally no more than 1-2 element ids off.

Detector	1	2	3	4	5	6	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Pos Truth	83	83	62	63	83	84	41	41	31	30	38	37	34	34	15	15	16	16	0	0	0	0	0	0
Pos Output	85	85	61	63	83	83	41	40	31	30	38	37	31	31	13	12	14	14	0	0	0	0	0	0
Neg Truth	95	95	79	79	105	105	80	80	70	70	76	76	0	0	0	0	0	0	101	101	86	86	88	87
Neg Output	94	95	79	79	105	106	80	79	70	69	76	76	0	0	0	0	0	0	104	103	88	88	90	89

Speed and acceleration

- Without acceleration, my computer calculates 20,000 events in 1.8 seconds
- After converting neural network to ONNX framework, able to perform the operation in just 0.38 seconds.
- Need to compare accuracy to KTracker algorithm on same MC Data, but if comparable accuracy, would mark a big difference in speed of track finding.