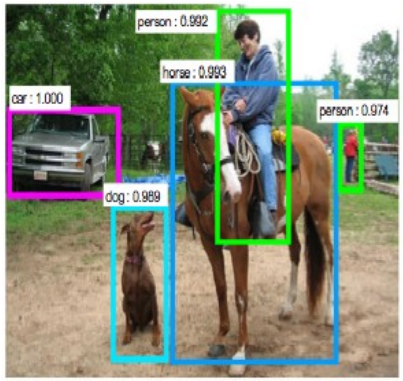
Deep learning applications in Physics

周凯 (FIAS, Frankfurt U., Germany)

Long-Gang Pang, Ling-Xiao Wang, Yin Jiang, Lian-Yi He, Nan Su, Yi-lun Du, Xin-Nian Wang, Gergely Endroedi, Manjunath Omana Kurtan, Jan Steinheimer, Volker Koch, Jorgen Randrup, Anton Motornenko, Horst Stoecker

Introduction



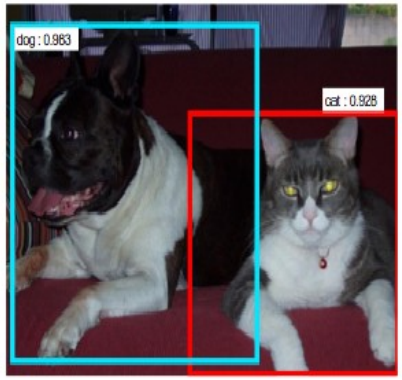
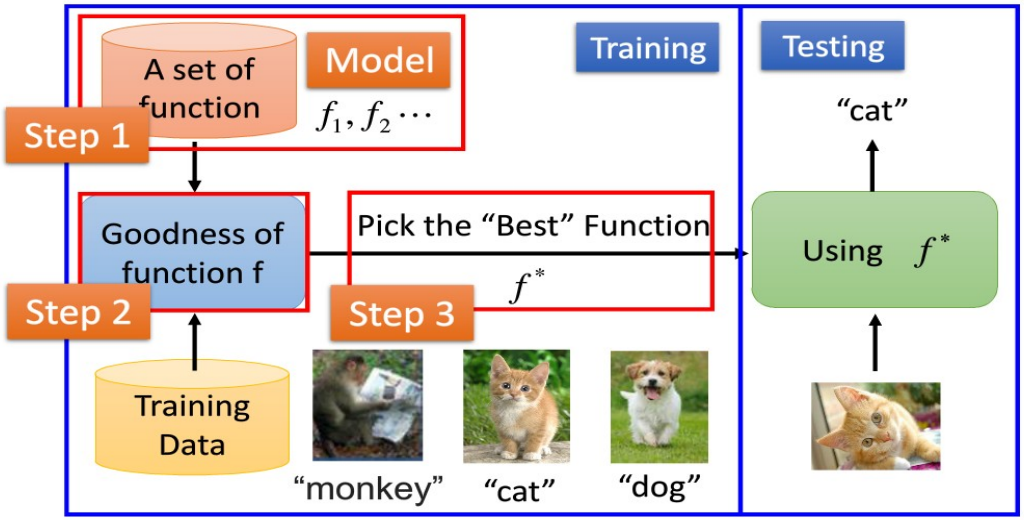


Image Recognition:

$f($  $) =$ "cat"

Framework



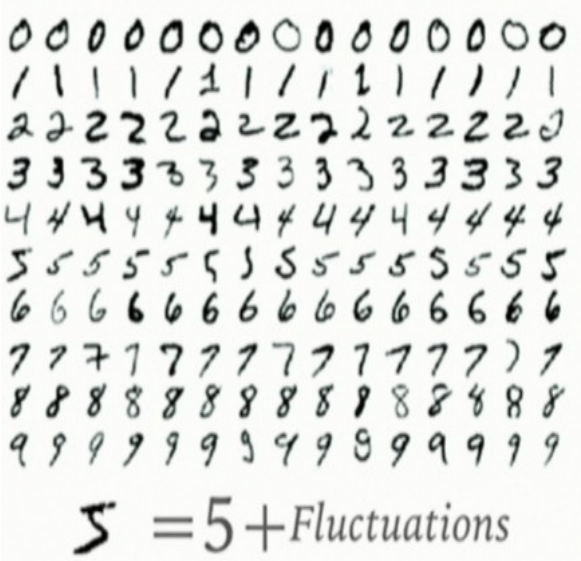
Find and **Decode** the mapping/representations into Deep Neural Network

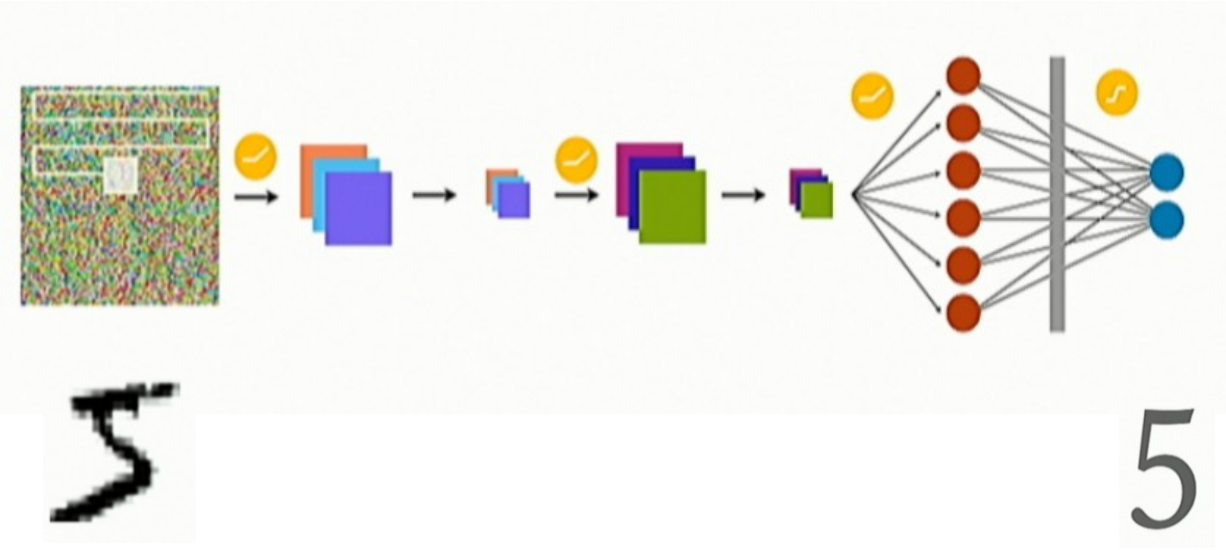
→ **Function approximator**

Universal approximator (Hastad et al 86 & 91)

Introduction

- Convolutional Neural Network has proved to be extremely powerful in Pattern Recognition, Image Classification





- Discriminative learning (prediction) : Classification, Regression
  Generative modelling (generation) : RBM, VAE, GAN

Identifying physics in Heavy-Ion-Collisions (from Hydrodynamics using deep learning)

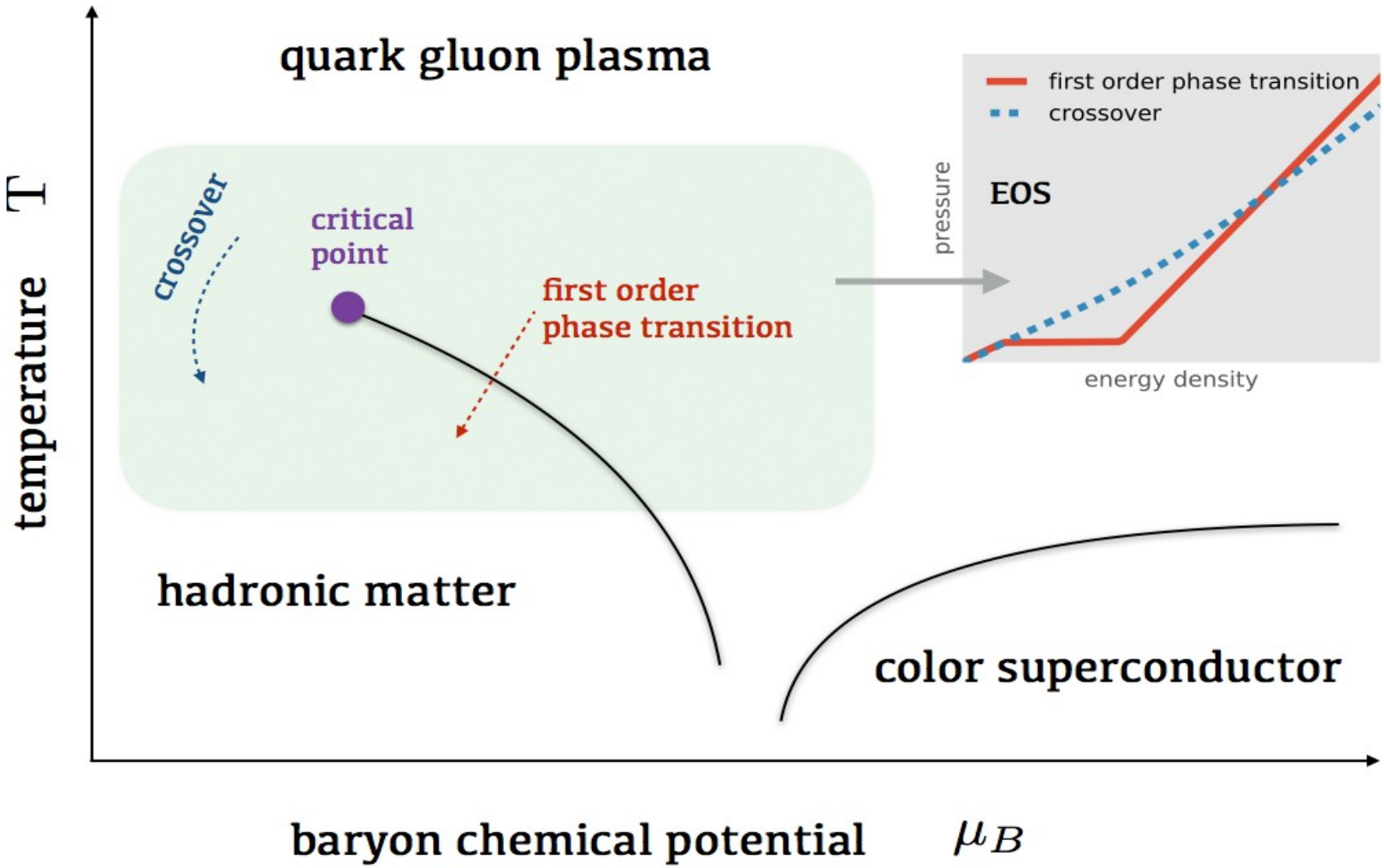
Nature Commun. 9 (2018) no.1, 210

Eur.Phys.J. C80 (2020) no.6, 516

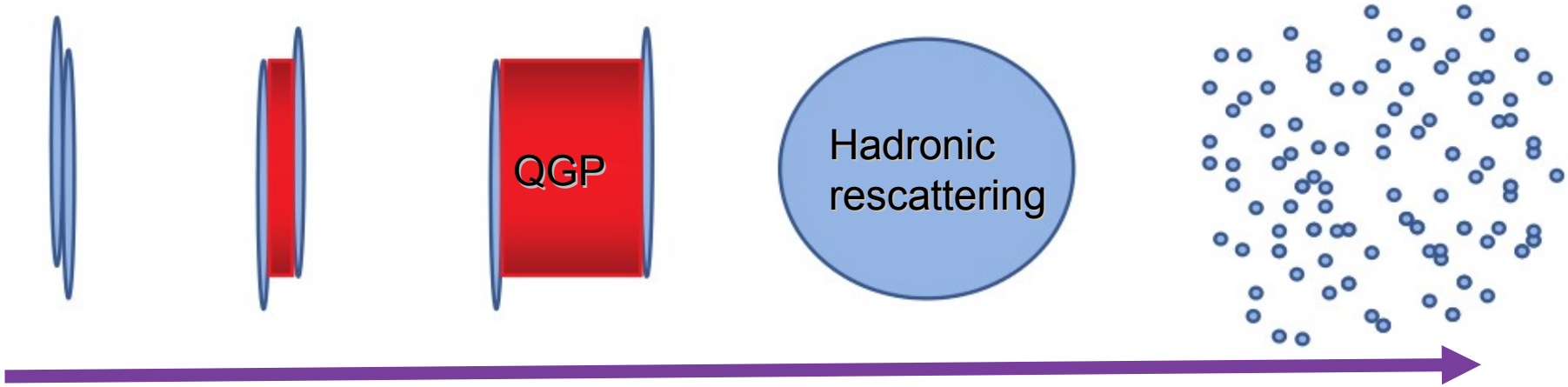
JHEP 1912 (2019) 122

Phys. Lett. B 811 (2020) 135872

QCD Transition: first order / crossover



Standard HIC model - uncertainties



Initial Stage

Fluctuation?
Pre-equilibrium?
Baryon Stopping?
Initial flow?

$$\nabla_\mu T^{\mu\nu} = 0$$
$$\nabla_\mu N^\mu = 0$$

EoS ?
Viscosity?
q/h chemistry?
T_freeze-out?

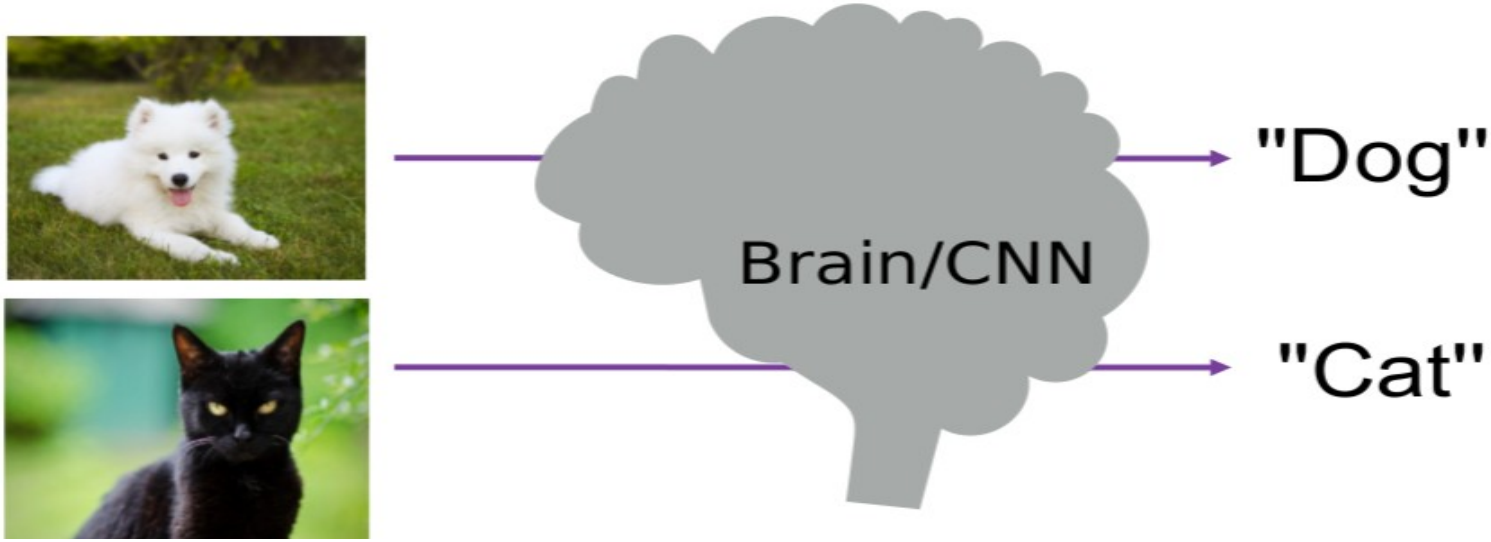
$$p^\mu \partial_\mu f + F \cdot \partial_p f = C$$

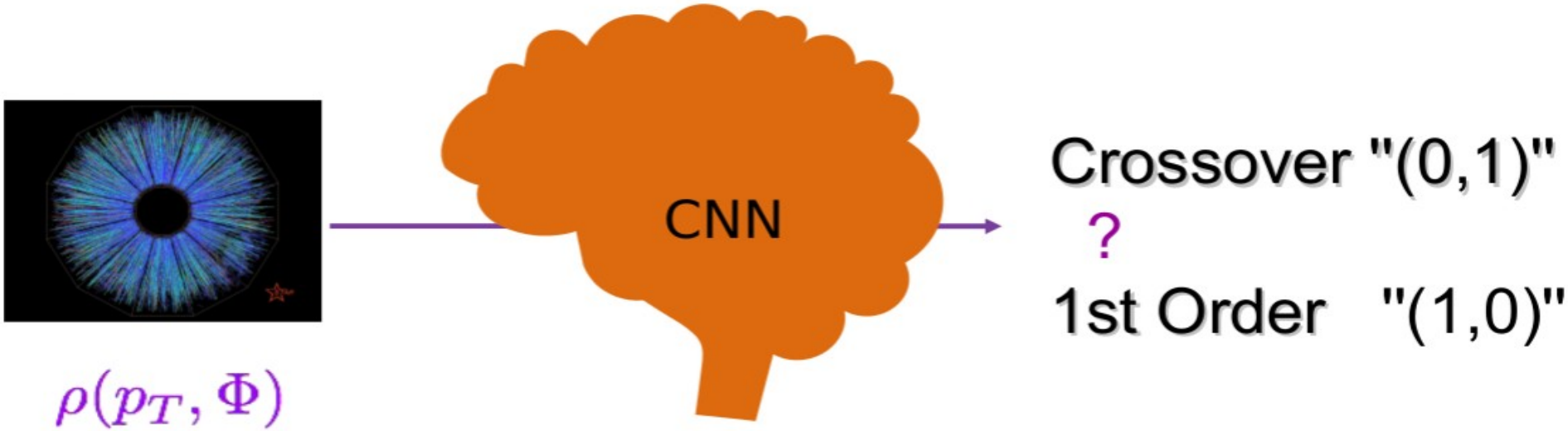
Scattering Cross Section?
...?

Final State

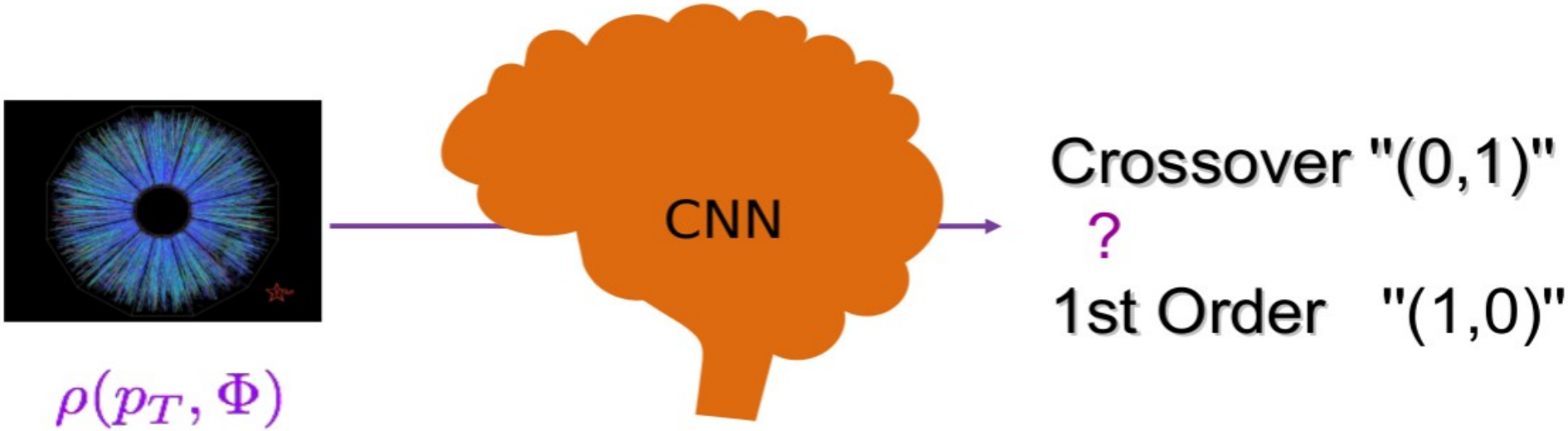
Interference Uncertainties
when confronting Exp. data

Inspried from Brain/CNN





Inspired from Brain/CNN



Supervised learning using deep Convolution Neural Network

with huge amount of labelled training data :(spectra, EoS type)

from event-by-event relativistic Hydrodynamic simulations.

Training dataset from simulation

Final Spectra for charged pions at mid-rapidity :

$$\rho(p_T, \Phi) \equiv \frac{dN_i}{dY p_T dp_T d\Phi} = g_i \int_\sigma p^\mu d\sigma_\mu f_i$$

	TRAINING DATASET	$\eta/s = 0$		$\eta/s = 0.08$	
		EOSL	EOSQ	EOSL	EOSQ
RHIC	Au-Au $\sqrt{s_{NN}} = 200\,\mathrm{GeV}$	7435	5328	500	500
LHC	Pb-Pb $\sqrt{s_{NN}} = 2.76\,\mathrm{TeV}$	4967	2828	500	500

- CLVisc 3+1 D viscous hydrodynamics with AMPT initial conditions
- τ_0 is 0.4 fm for Au-Au and 0.2 fm for Pb-Pb
- T_freeze-out is 137 MeV

22000 events, further doubled by left-right flipping along \phi,
10% are for validation during the training

Testing dataset from different models

TESTING DATASET GROUP 1 : iEBE-VISHNU + MC-Glauber						
Centrality:	$\eta/s \in [0, 0.05]$		$\eta/s \in (0.05, 0.10]$		$\eta/s = (0.10, 0.16]$	
10-60%	EOSL	EOSQ	EOSL	EOSQ	EOSL	EOSQ
Au-Au $\sqrt{s_{NN}} = 200$ GeV	650	850	900	750	200	950
Pb-Pb $\sqrt{s_{NN}} = 2.76$ TeV	500	650	600	644	499	150

TESTING DATASET GROUP 2 : CLVisc + IP-Glasma		
Au-Au $\sqrt{s_{NN}} = 200$ GeV	EOSL	EOSQ
b$\lesssim$8 fm & $\eta/s = 0$	4165	4752

- iEBE-VISHNU hydro packag with a different numerical solver and with different initial condition (MC-Glauber)

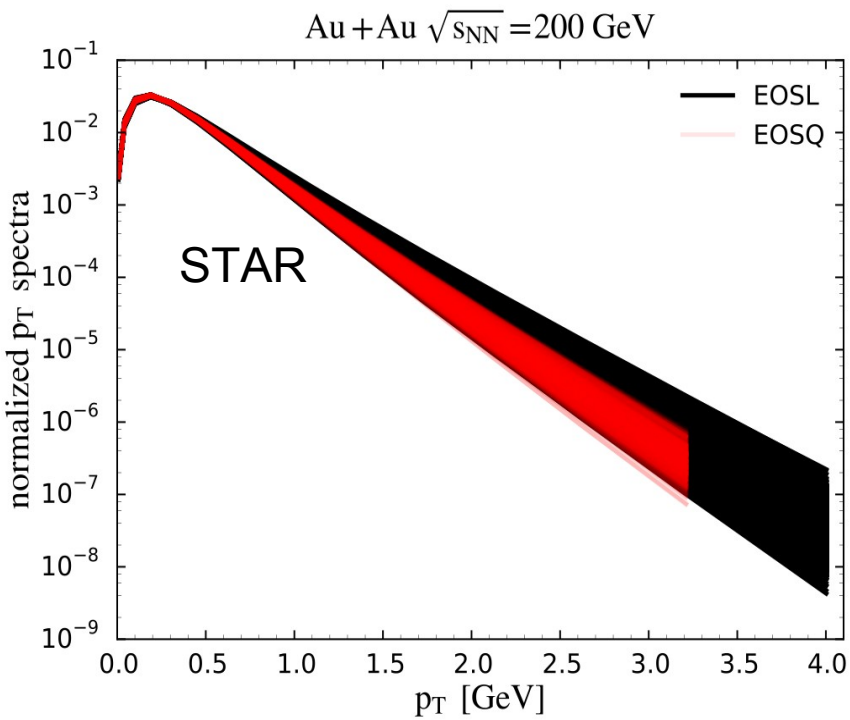
  C. Shen, et.al., Comput. Phys. Commun. 199, 61 (2016)

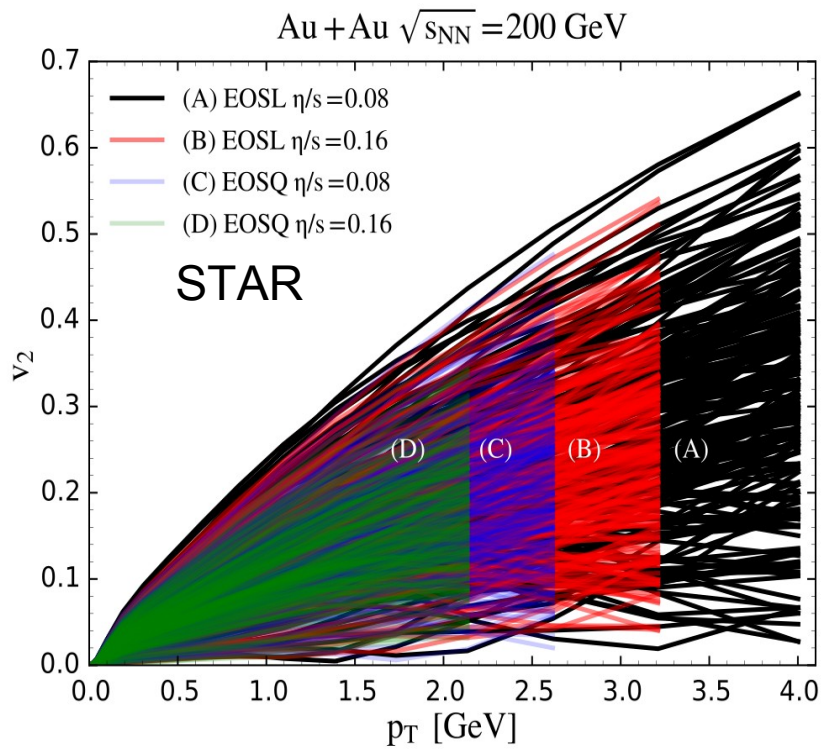
- τ_0 is 0.6 fm , eta/s changing within [0, 0.16]

- T_freeze-out in [115, 142] MeV for iEBE-VIS, 137 MeV for CLVisc

  --- 16000 events

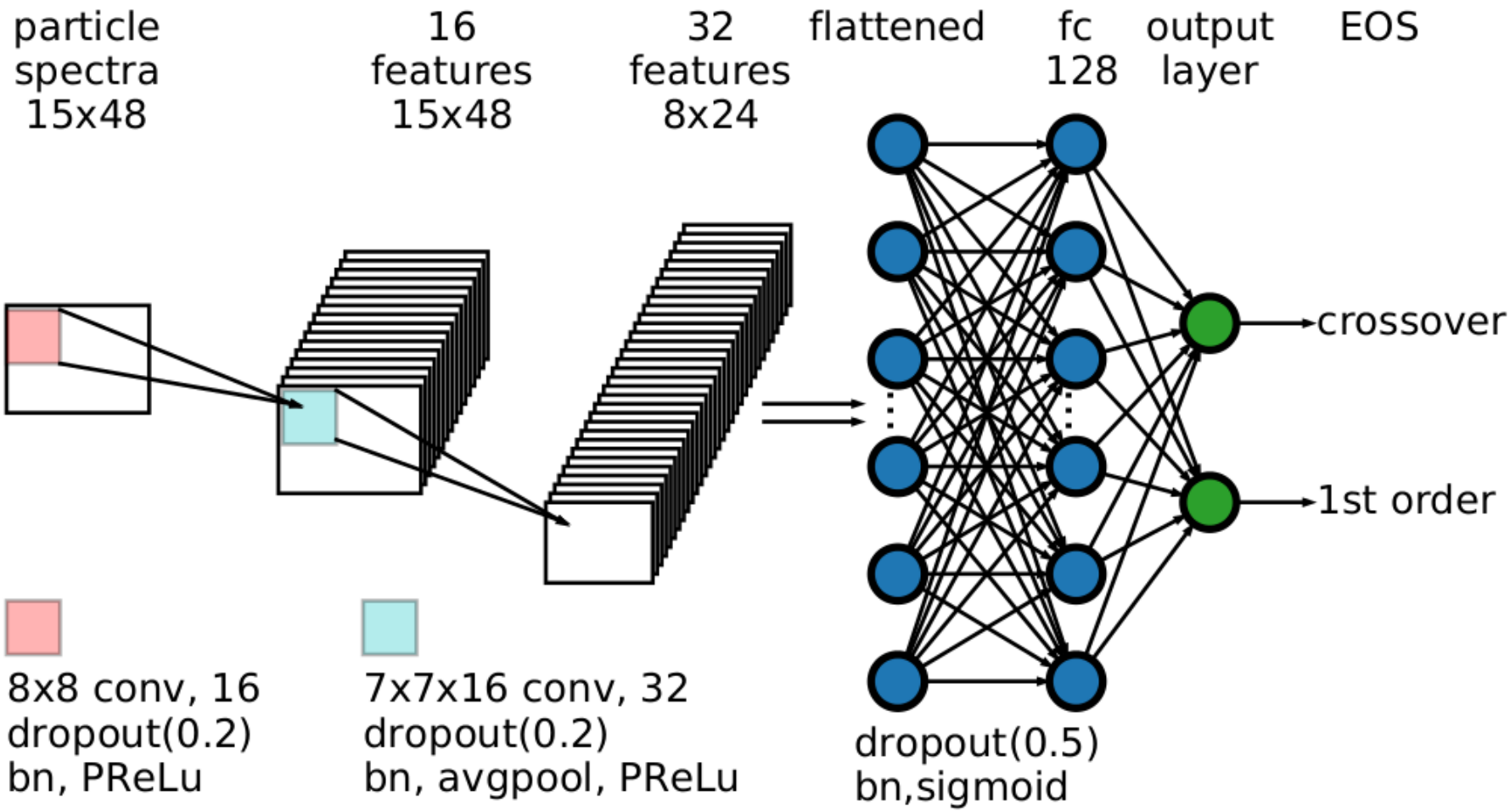
Conventional observables : failed





- Strongly depends on initial fluctuations and other uncertainties !

Convolutional Neural Network achitecture



Results on testing set

TESTING DATA	GROUP 0	GROUP 1	GROUP 2
Number of events	4000	7343	8916
Accuracy	$99.88 \pm 0.04\%$	$93.46 \pm 1.35\%$	$95.12 \pm 3.08\%$

- On average ~95% prediction accuracy, the trained CNN model identifies the type of QCD transition solely from the raw spectra

- The performance is robust against : initial conditions, $\eta/s, \tau_0, T_{fo}$
  model independent!

Take-to-home perspectives

- There do exist "Encoder" (mapping, in a hydro system) of the QCD Transition → final state raw spectra (pT, phi)
  Although it maybe NOT intuitive for conventional observables

  It is CLEAN –- **immune** to other parameters uncertainties

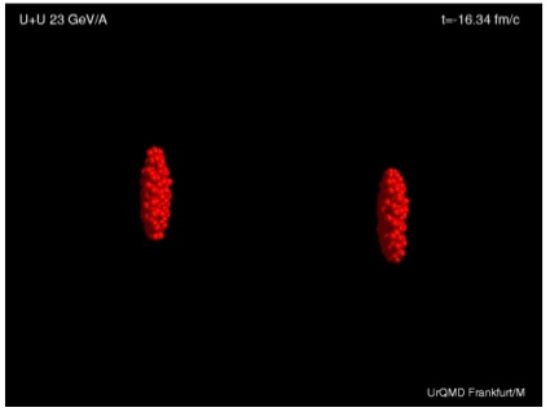
- Deep CNNs can provide "Decoder" for this "Encoder", which can act as **"EoS-Meter"**

- Deep CNNs might help to **directly connect** HIC experiments with QCD properties, help searching for **CEP**.

Nature Commun. 9 (2018) no.1, 210

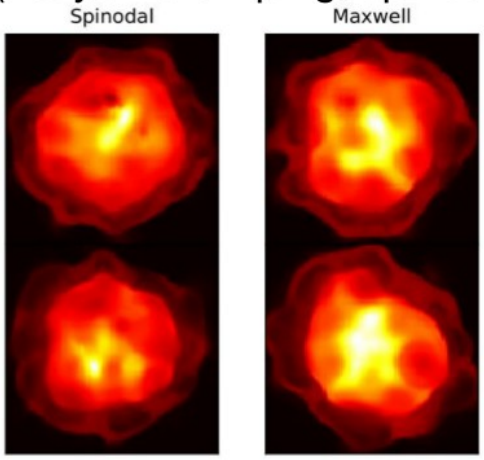
Further deepened works

Hadronic cascade
(UrQMD/FIAS considered)



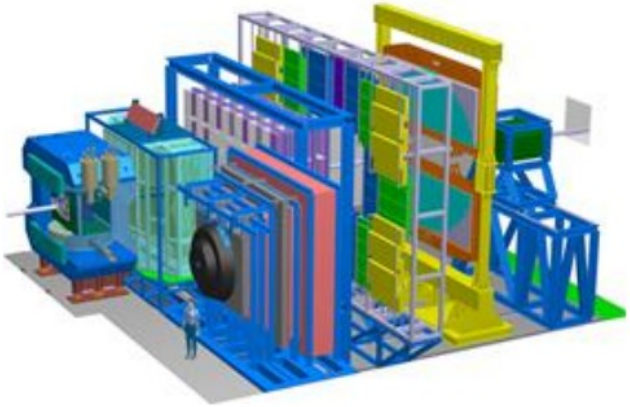
Eur.Phys.J. C80 (2020) no.6, 516

Non-equilibrium transition
(Baryon Clumping, spinodal)



JHEP 1912 (2019) 122

Detector simulation
(Tracks/Hits, PointNet)



Phys. Lett. B 811 (2020) 135872

Hadronic cascade included

- Couple 2+1D viscous hydro (VISHNew) with hadronic cascade model (UrQMD) : stochastic scatterings and resonance decay are contained

  histogram distributions from the simulation

  - Event-by-event spectra, Tsw = 137 MeV
  - Cascade-coarse-grained spectra, Tsw = 137 MeV
  - Events-fine-averaged spectra, Tsw = 137 MeV

  - Event-by-event spectra, Tsw > 150 MeV
  - Cascade-coarse-grained spectra, Tsw > 150 MeV
  - Events-fine-averaged spectra, Tsw > 150 MeV

Eur.Phys.J. C80 (2020) no.6, 516

Training Data

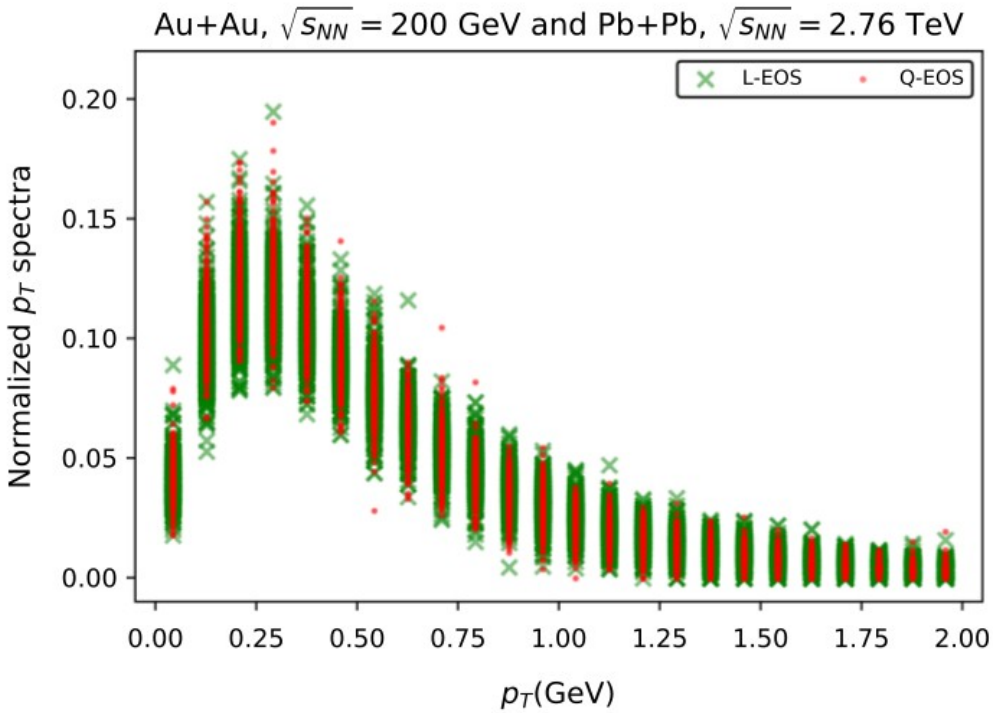
Table 1 Training datasets 1: numbers of event-by-event spectra $\rho(p_T, \Phi)$ computed by the iEBE-VISHNU hybrid model with the MC-Glauber initial conditions in the centrality range 0–60%. The ratio of shear viscosity to entropy density $\eta/s = 0.08$. The equilibration time $\tau_0 = 0.5$ fm/c. The switching temperature $T_{sw} = 137$ MeV. The collision system is Pb + Pb at $\sqrt{s_{NN}} = 2.76$ TeV

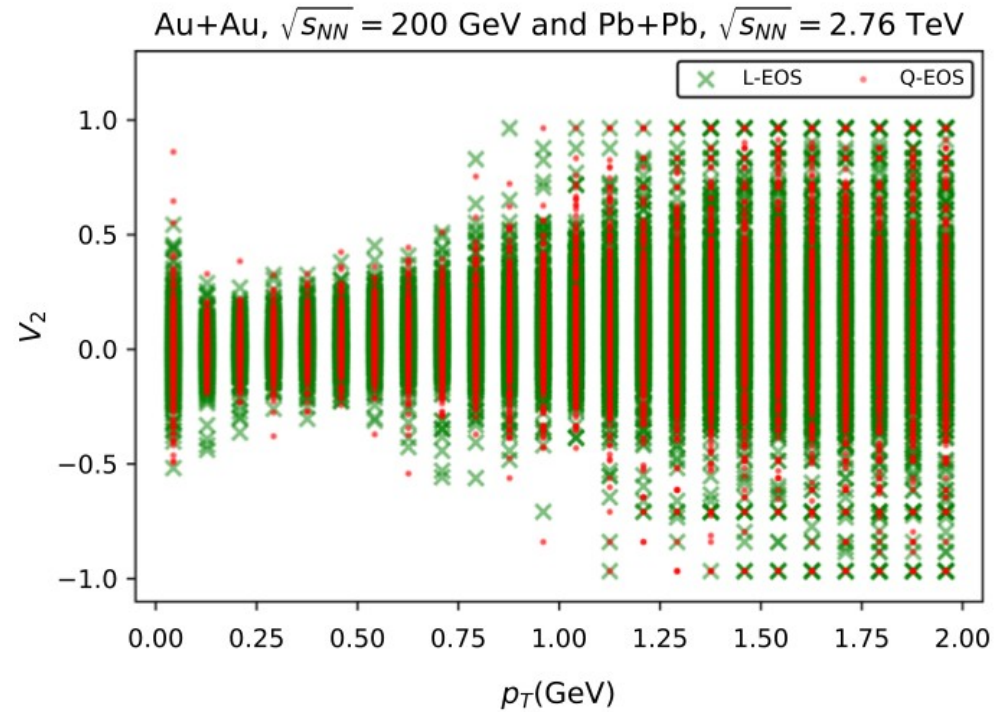
Training datasets 1		
Centrality bin	L-EOS	Q-EOS
4%–5%	2539	2540
14%–15%	1022	1024
20%–21%	2814	2816
30%–31%	2560	2560
40%–41%	1024	1024
50%–51%	896	1024

Table 2 Training datasets 2: numbers of event-by-event spectra $\rho(p_T, \Phi)$ computed by the iEBE-VISHNU hybrid model with the MC-Glauber initial conditions in the centrality range 0–60%. The ratio of shear viscosity to entropy density $\eta/s = 0.00$. The equilibration time $\tau_0 = 0.4$ fm/c. The switching temperature $T_{sw} = 137$ MeV. The collision system is Au + Au at $\sqrt{s_{NN}} = 200$ GeV

Training datasets 2		
Centrality bin	L-EOS	Q-EOS
0%–1%	979	1024
10%–11%	2560	2560
20%–21%	1024	1024
30%–31%	1024	1024
40%–41%	2560	2560
50%–51%	2816	2816

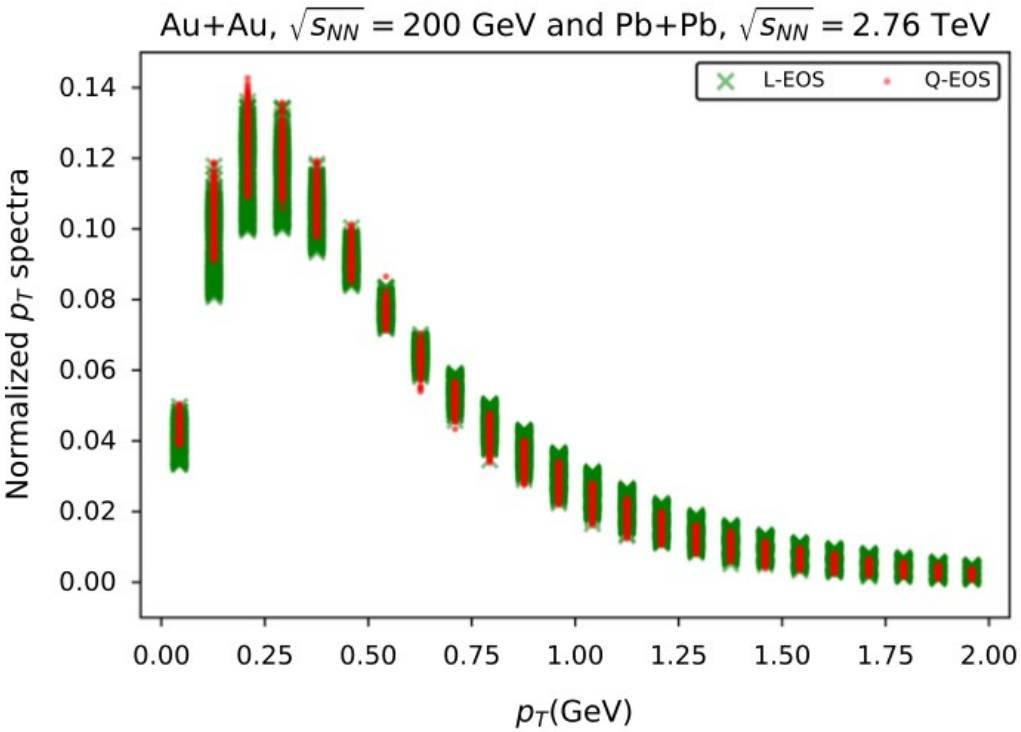
Event-by-event spectra and flow

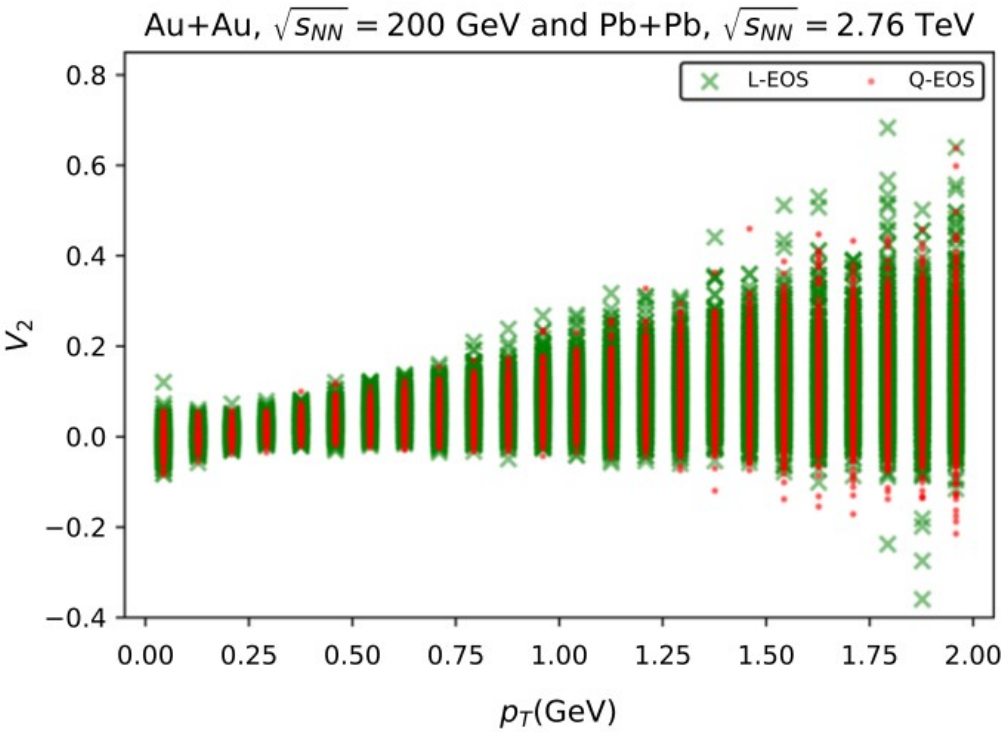




Events generated in different centrality bins with Tsw = 137 MeV from two colliding systems

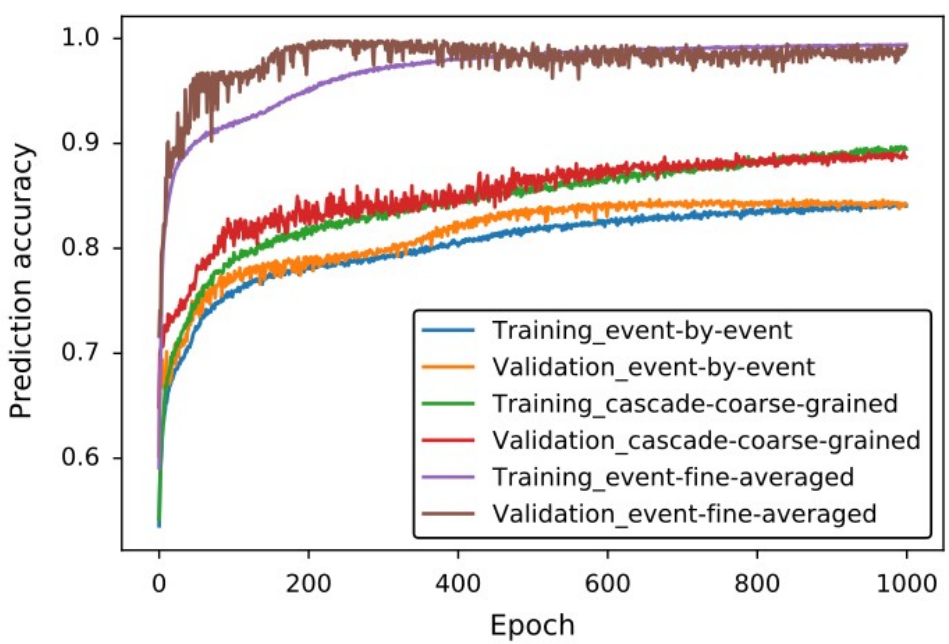
30event-fine-averaged spectra and flow

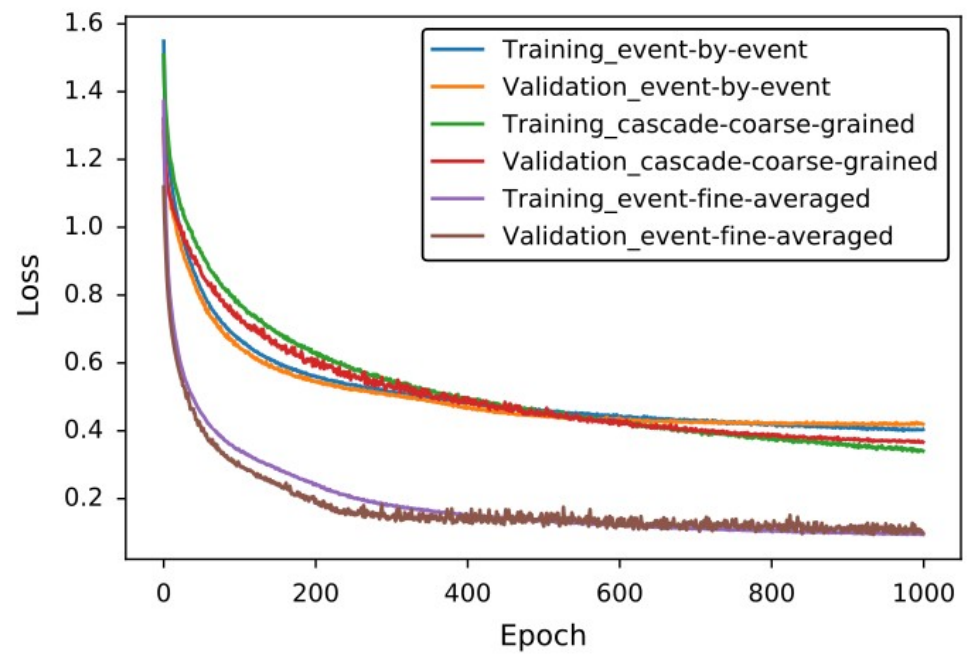




Events generated in different centrality bins with Tsw = 137 MeV from two colliding systems

Learning curve (accuracy,loss)

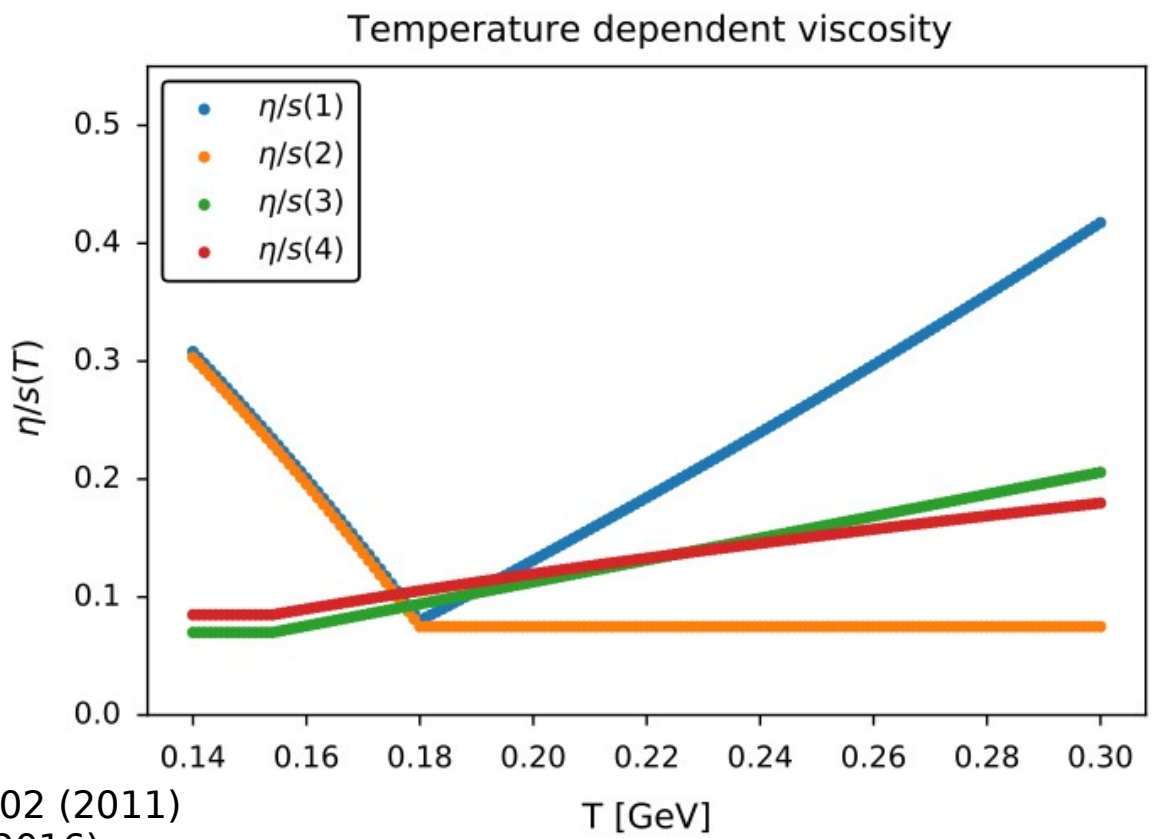




Clear hierarchy shows up in the prediction accuracy when using 3 different spectra as input for the net, ~around 80%, 90%, 99%, respectively

Robust to eta/s ?

T-dependent shear viscosities are employed in the testing dataset, which are also with different initial condition, switching temperature and tau0.



Phys. Rev. Lett. 106, 212302 (2011)
Phys. Rev. C 94, 024907 (2016)
Nat. Phys. 15, 1113 (2019)

Tesing performance

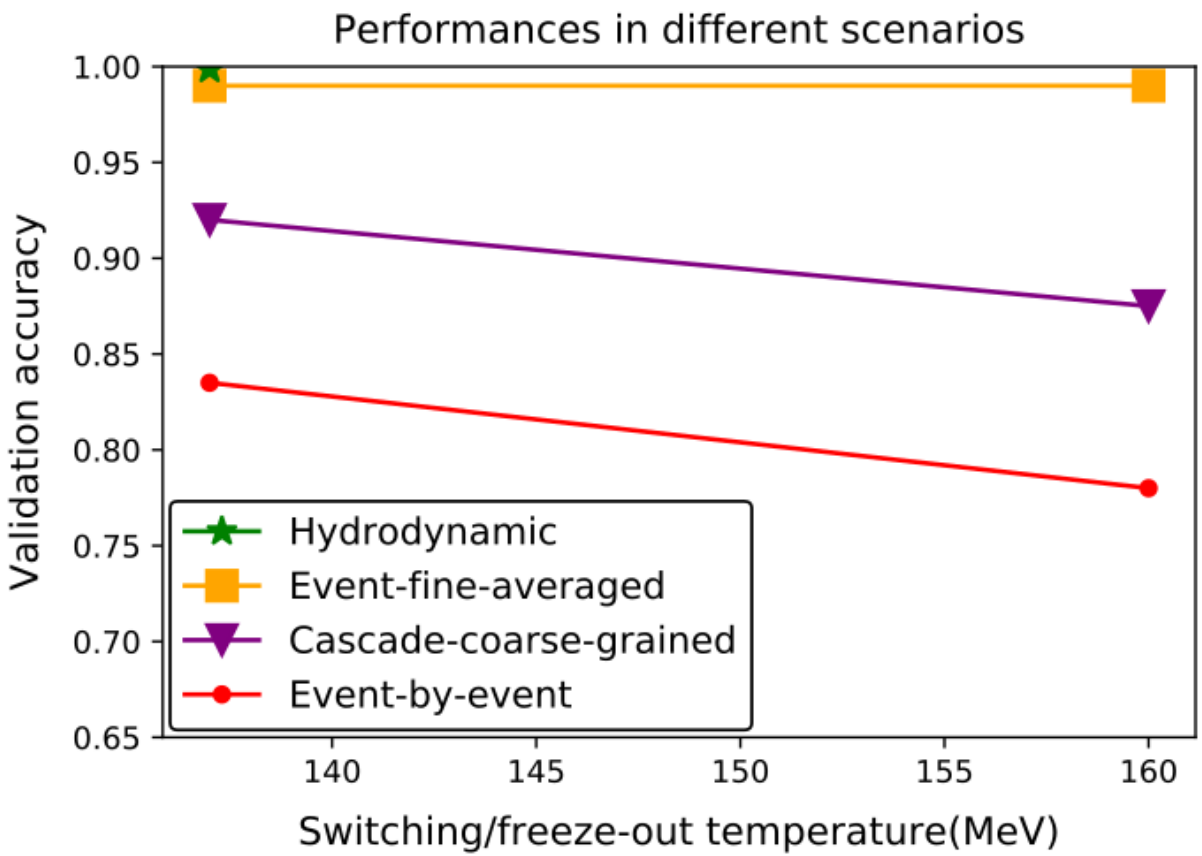
Predictive accuracy for testing datasets

Centrality bin	$\sqrt{s_{NN}}$ [TeV]	Ini. Cond.	τ_0 (fm/c)	$\eta/s(T)$	T_{sw} [MeV]	L-EOS	Q-EOS	Accuracy
10%–11%	Au + Au 0.2	MC-G	0.4	1	160	512	512	100%
10%–11%	Au + Au 0.2	MC-G	0.4	2	160	512	512	100%
10%–11%	Au + Au 0.2	MC-G	0.4	3	160	512	512	100%
10%–11%	Au + Au 0.2	MC-G	0.4	4	160	512	512	100%
15%–16%	Au + Au 0.2	MC-G	0.4	4	160	512	512	99.51%
20%–21%	Au + Au 0.2	MC-G	0.4	3	160	512	512	98.34%
10%–11%	Au + Au 0.2	MCKLN	0.6	3	160	512	512	98.04%
10%–11%	Pb + Pb 2.76	MC-G	0.6	3	155	512	512	99.80%
25%–26%	Pb + Pb 2.76	MC-G	0.6	4	160	512	512	99.90%
35%–36%	Pb + Pb 2.76	MC-G	0.6	3	155	512	512	86.72%

Predictive accuracy for testing datasets 4

Centrality bin	$\sqrt{s_{NN}}$ [TeV]	Ini. Cond.	τ_0 (fm/c)	η/s	T_{sw} [MeV]	L-EOS	Q-EOS	Accuracy
15%–16%	Au + Au 0.2	MCKLN	0.6	0.16	160	640	640	95.59%
10%–11%	Au + Au 0.2	MC-G	0.4	0.12	160	2560	2560	100%
15%–16%	Au + Au 0.2	MC-G	0.4	0.12	160	2560	2560	99.8%
20%–21%	Au + Au 0.2	MC-G	0.4	0.12	160	2560	2560	94.9%
15%–16%	Au + Au 0.2	MC-G	0.4	0.00	155	2560	2560	74.86%
20%–21%	Au + Au 0.2	MC-G	0.4	0.12	155	1792	1792	88.8%
15%–16%	Pb + Pb 2.76	MC-G	0.6	0.08	155	2560	2560	99.99%
20%–21%	Pb + Pb 2.76	MC-G	0.6	0.08	155	2560	2560	99.78%

Overall results



Take-to-home perspectives

- With hadronic cascade included, predictive power of CNN for EoS classification with EbE spectra decreases down to 80% (compared with pure hydro case 99%), since the **stochastic particlization, hadronic cascade and resonance decays**.

- Statistics (cascade-coarse-grained or **events-fine-averaged spectra**) help improving the CNN prediction performance with good generalizability.

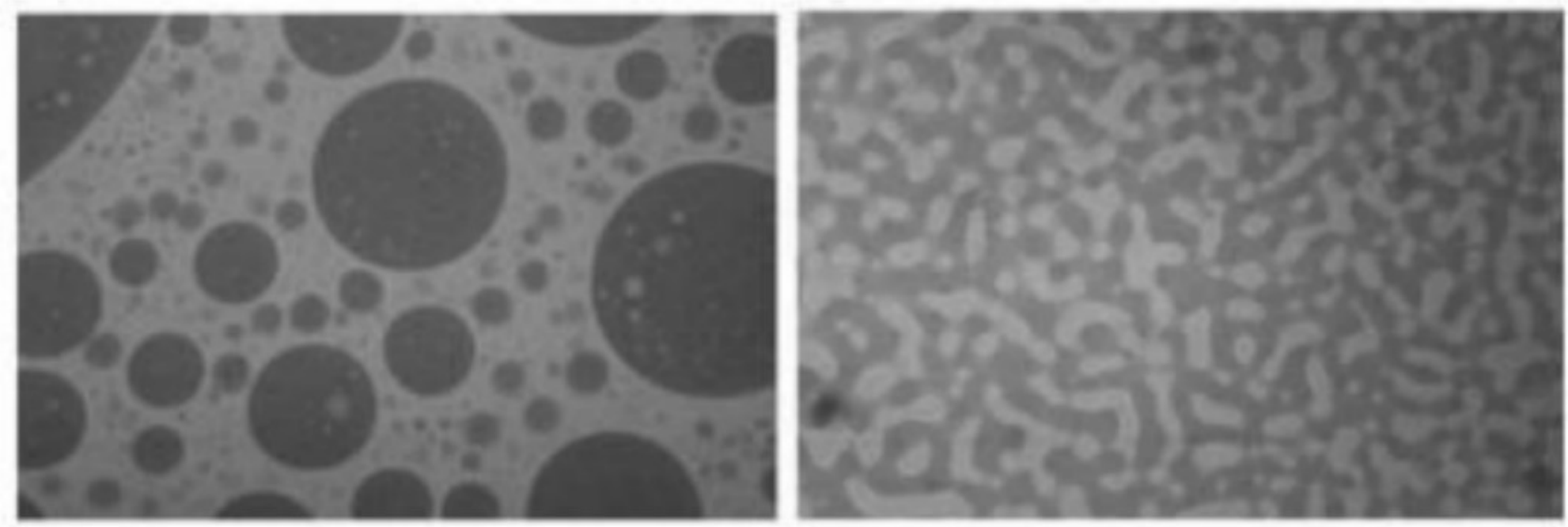
- With freeze-out T increasing, more hadronic cascade are involved and thus the CNN prediction performance with EbE and cascade-coarse-grained spectra decrease slightly

Eur.Phys.J. C80 (2020) no.6, 516

Non-equilibrium transition

Fluctuations carry important information for phase transition !

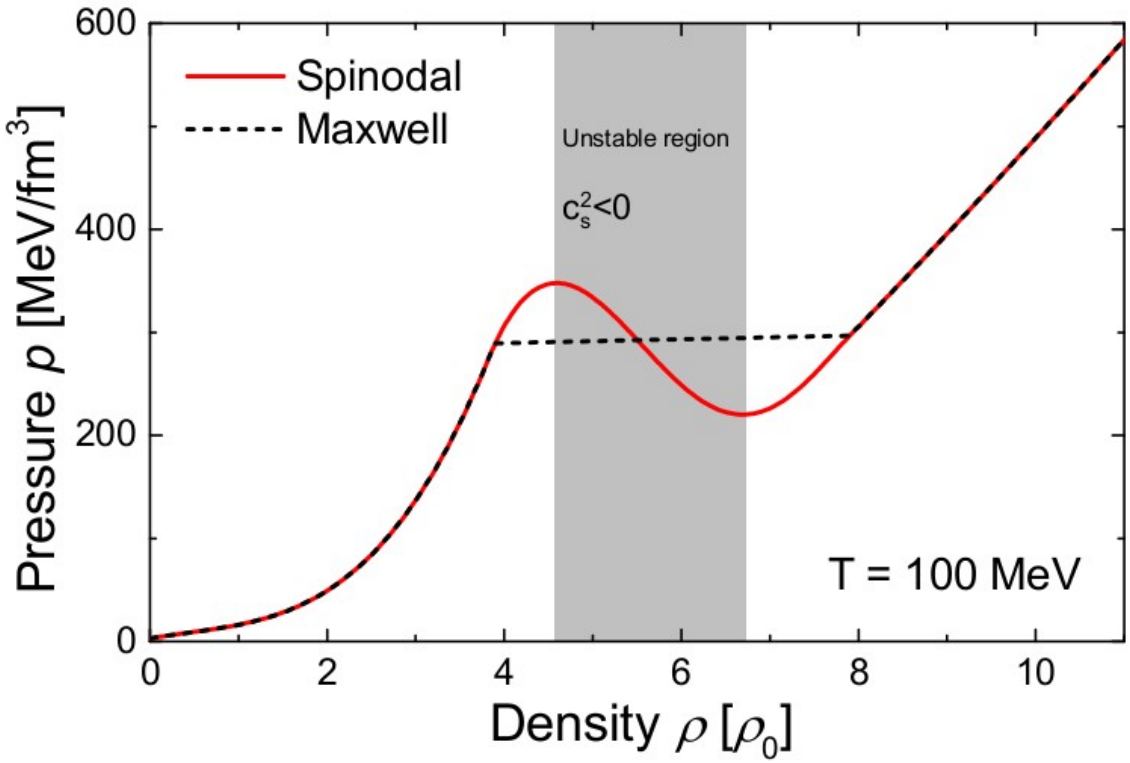
Nucleation (bubble, slow) & **Spinodal decomposition** (instabilities, fast)



JHEP 1912 (2019) 122

Spinodal EoS

- Maxwell construction serve as baseline
- Spinodal EoS has mechanically **unstable region**



Is Spinodal Clumping detectable?

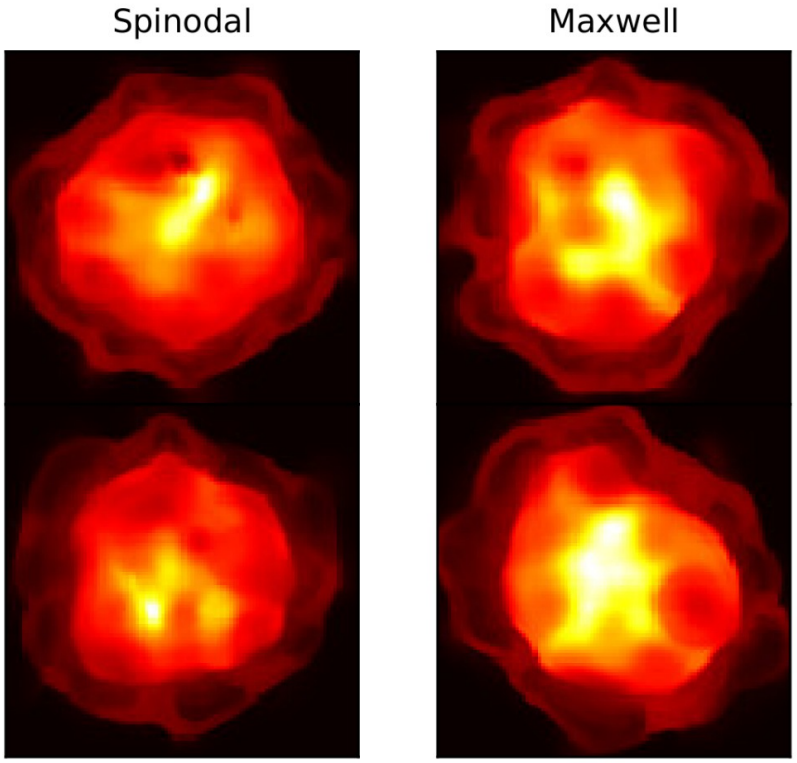
- The two EoS have similar collective radial expansions : same work
- They have significant differences in baryon number distribution
- Is there spinodal clumping in every event ?
- Will the coordinate clumping generate measurable signal in mom.space?

characteristic features in every event?

- Binary Classification of events from hydro with :
  Non-equilibrium spinodal Eos vs. equilibrium Maxwell EoS

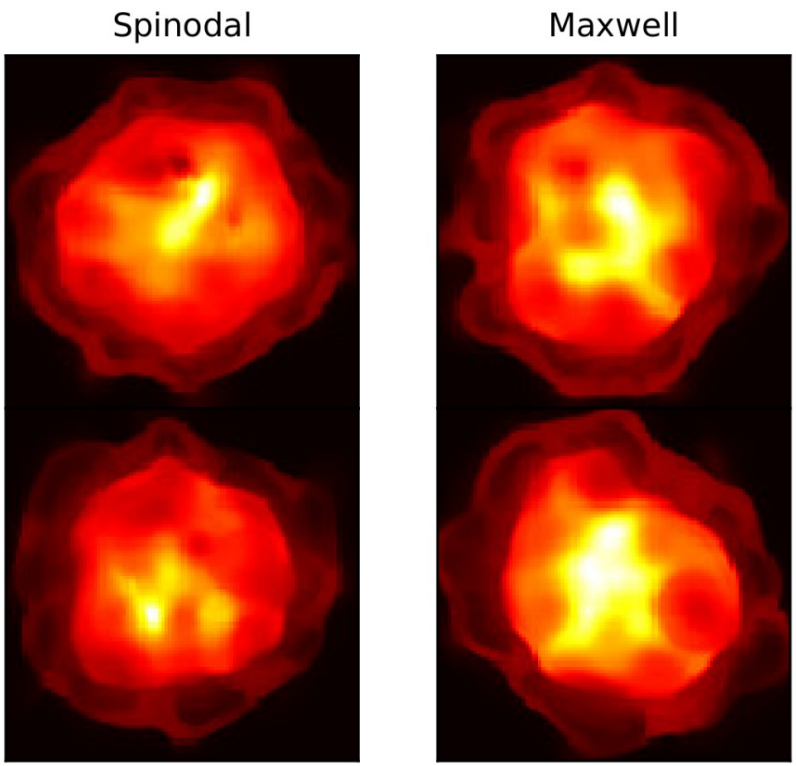
- About 20000 Pb+Pb collision events are generated at (FAIR/GSI) beam energy Elab=3.5 AgeV at t=3 fm/c

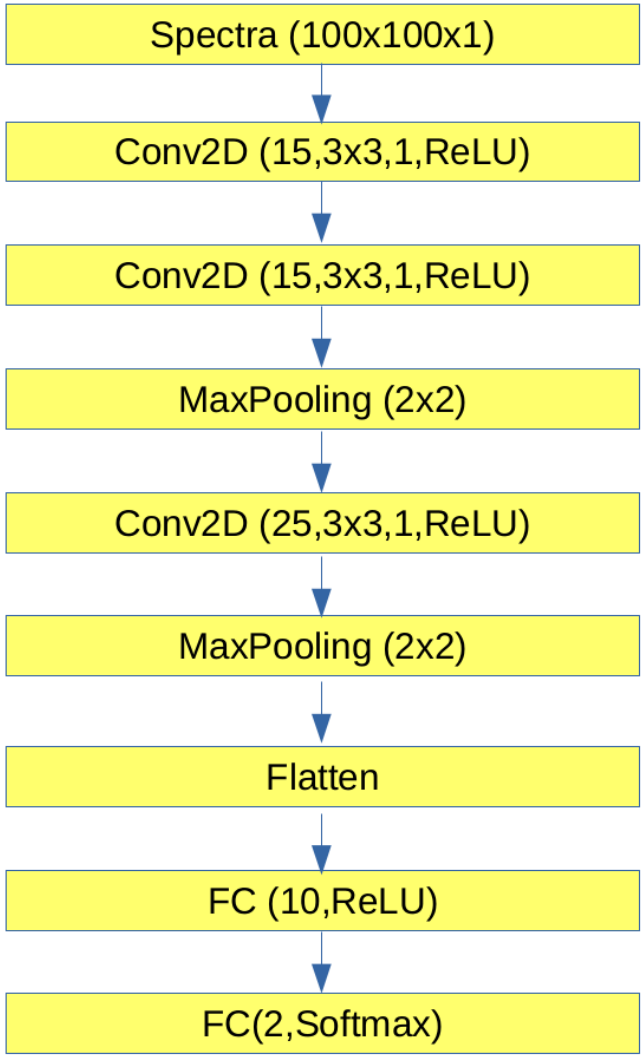
- Each provide an image(normalized, 100x100) of the net baryon number density in X-Y coordinate space at Z=0



characteristic features in every event?

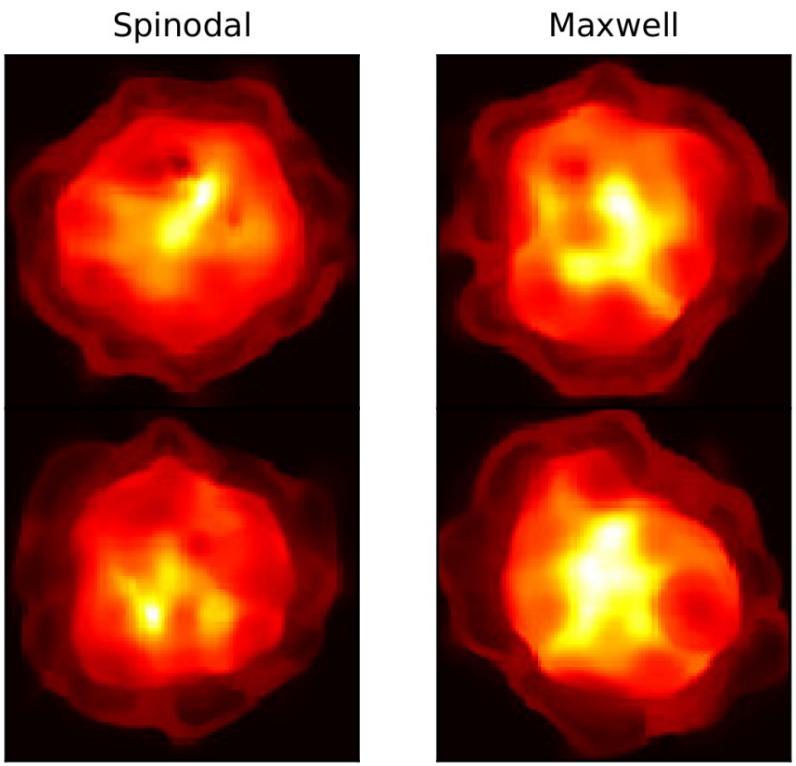
- spinodal / Maxwell EoS binary classification use CNN :

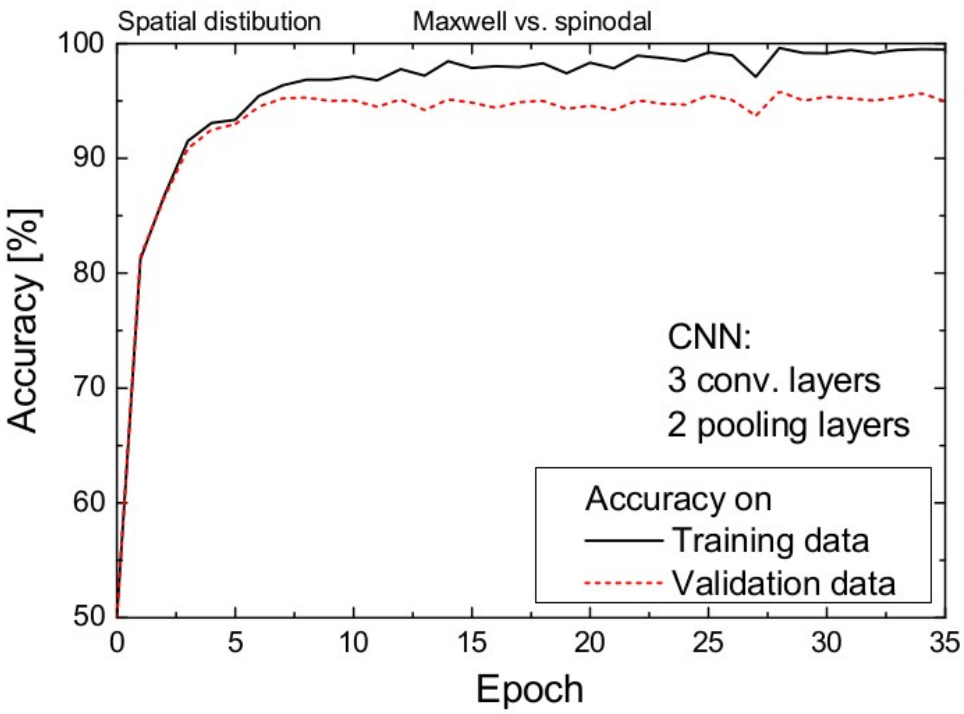




characteristic features in every event?

- 95% accuracy obtained for spinodal identification on independent events
- Yes ! DL suggests : spinodal EoS leads to characteristic feature in almost every event in coordinate space

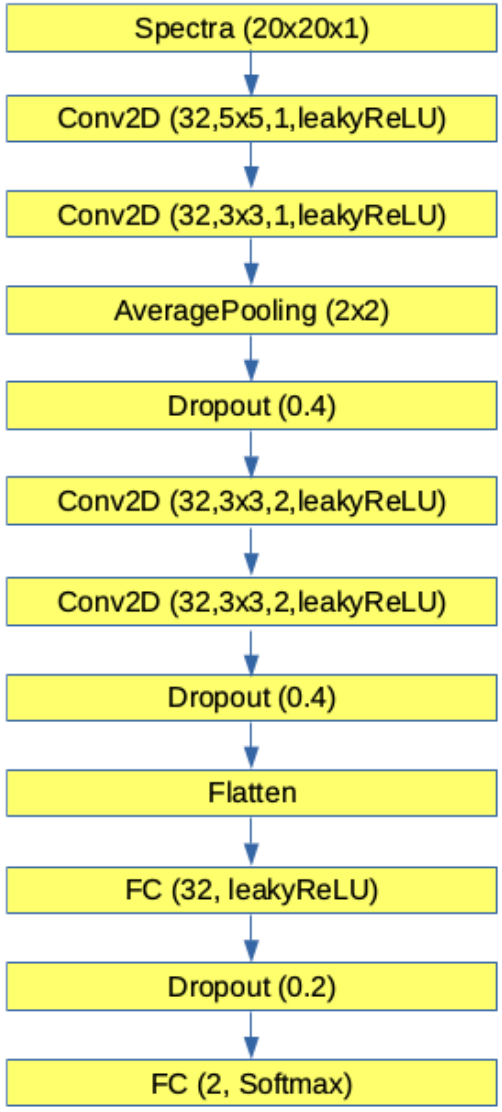


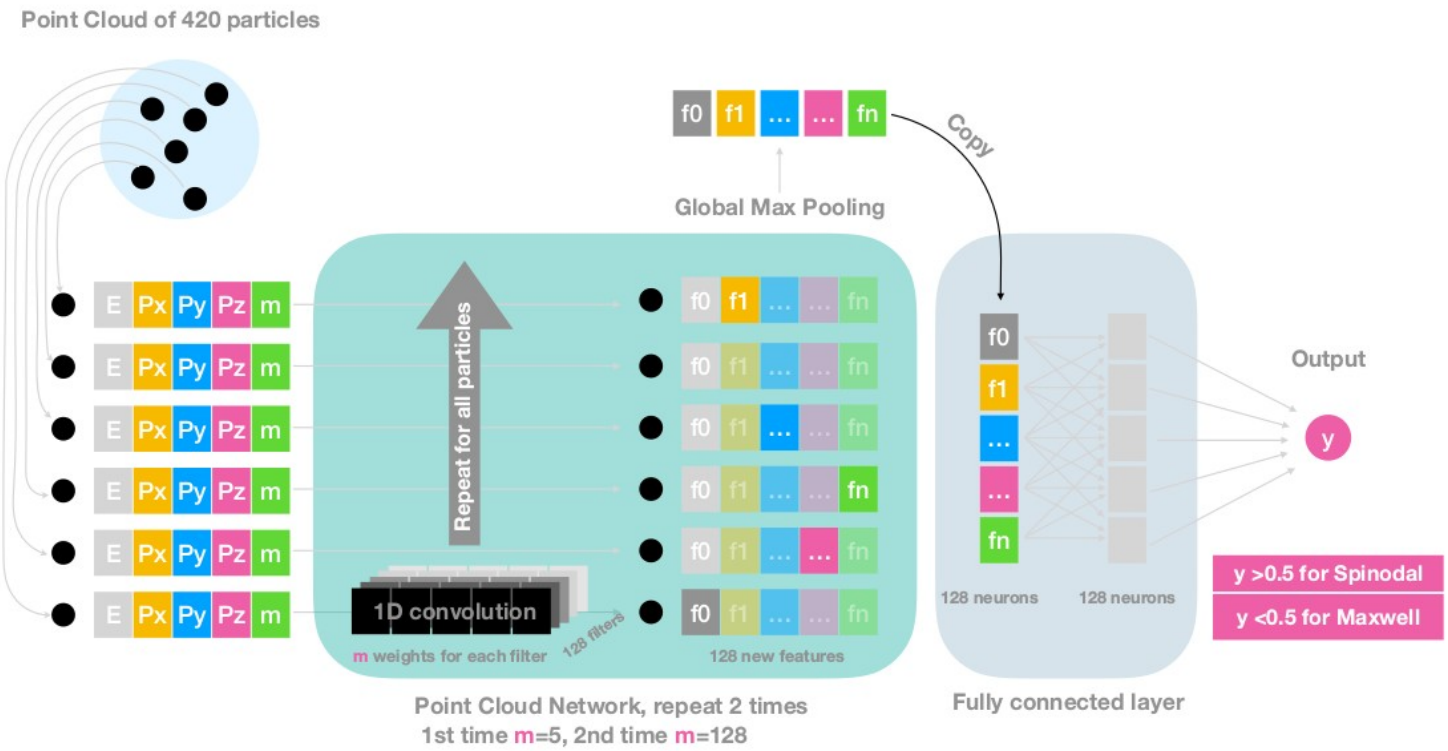


Move to momentume space

- Expansion may transform coordinate cluster to momentum space correlations
- Clusters of quark matter anyhow cannot be detected by experiment : baryons
- When particle is produced, clumping in xy space disappeared
- Cooper-Frye with TP (test particle) =1 and 20
- Resonance decays are performed (via UrQMD model)
- can generate 2D histogram of all baryons in transverse momentum space
- To use the full info of all discrete particles, point could network can be used

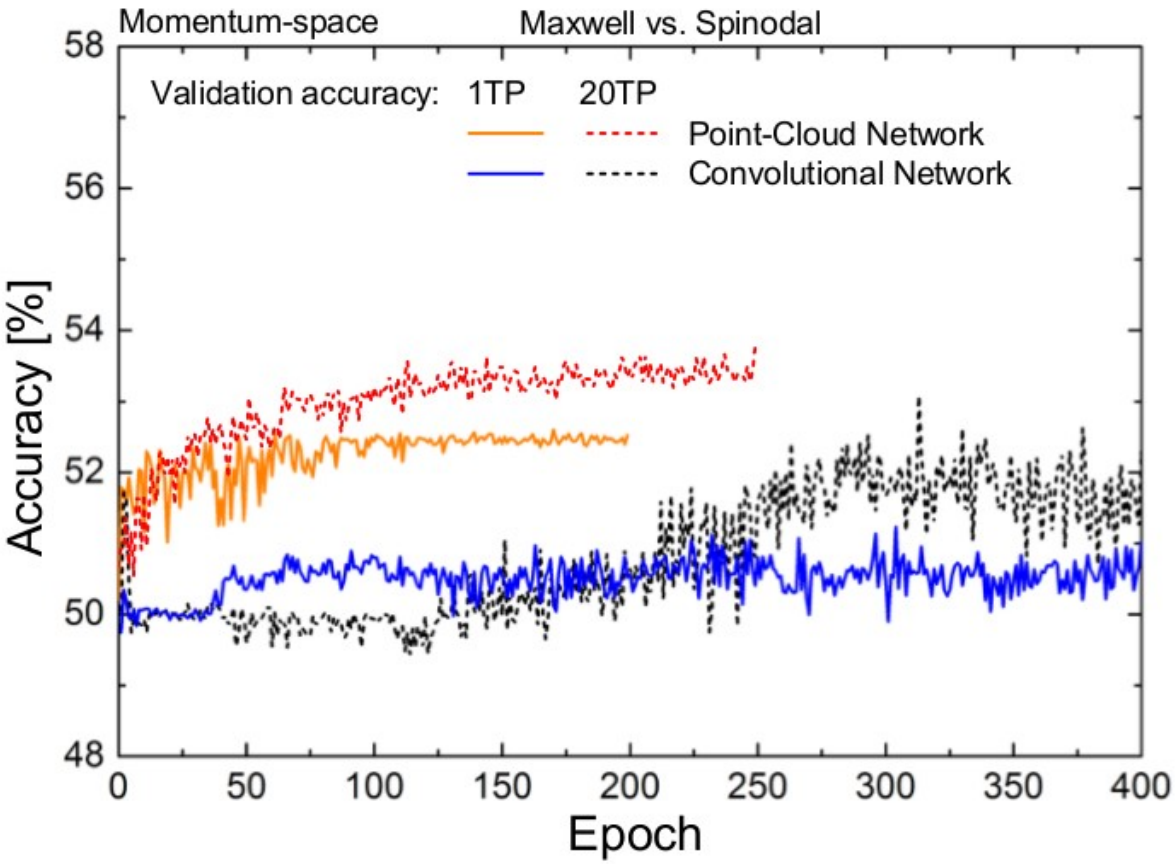
CNN, or, Point Cloud Net





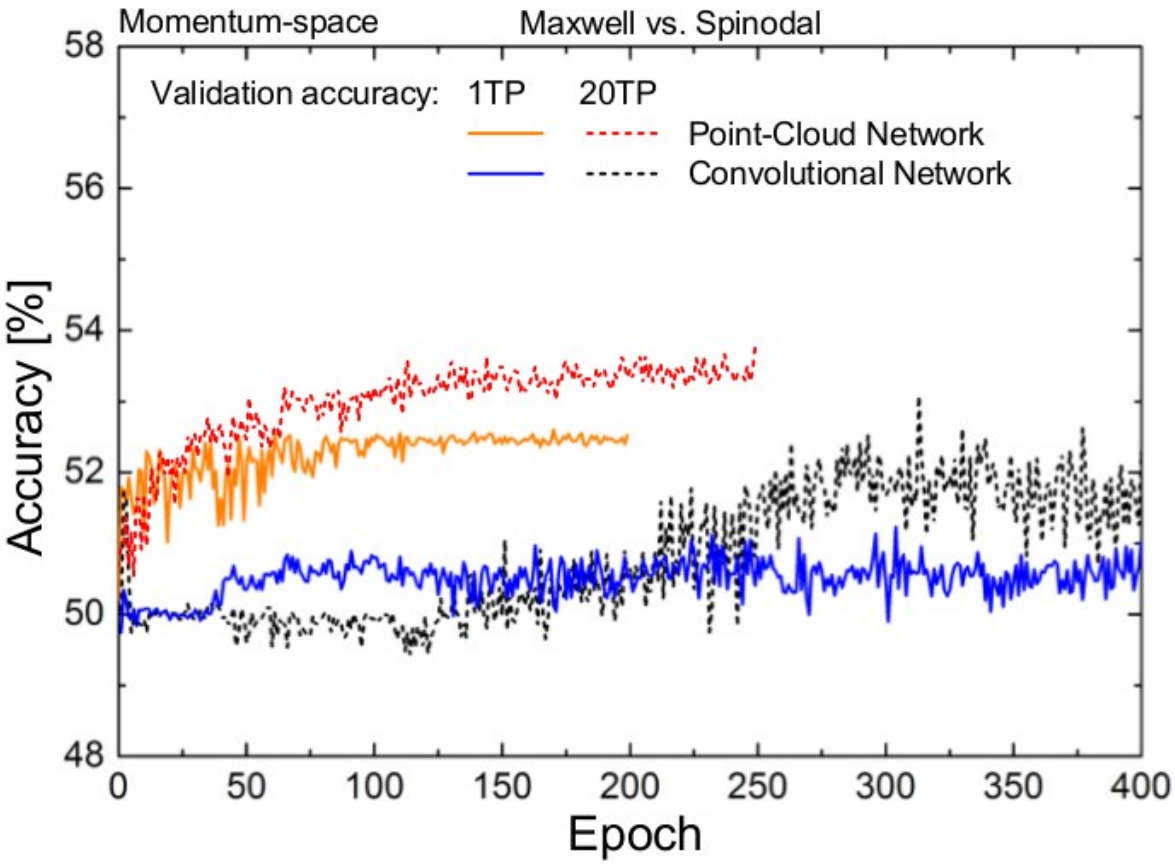
CNN failed : small EbE accuracy

- CNN failed to find significant features to discern spinodal/Maxwell EoS
  TP=20 : ~52 % accuracy, TP=1 : ~50% just random guessing !



Point Cloud Net (E, px, py, pz, m)

- Point Cloud Net does an overall slightly better job for the classification TP=20 : ~54 % accuracy, TP=1 : ~52% just random guessing !



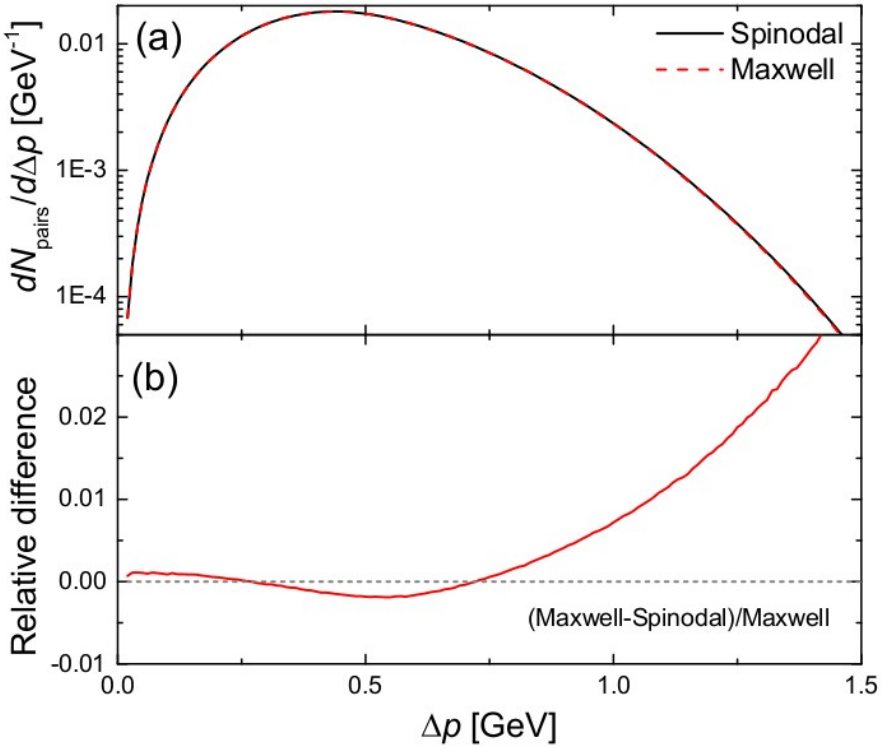
Hand crafted observables

- Can the 'signal' be encoded in the correlations?

  2-P Momentum difference distribution

$$\Delta p_{ab} \equiv \frac{1}{2}|\mathbf{p}^a - \mathbf{p}^b|$$
$$= \frac{1}{2}\left[(p_x^a - p_x^b)^2 + (p_y^a - p_y^b)^2 + (p_z^a - p_z^b)^2\right]^{\frac{1}{2}}$$

- Relative diff. small but appears systematic

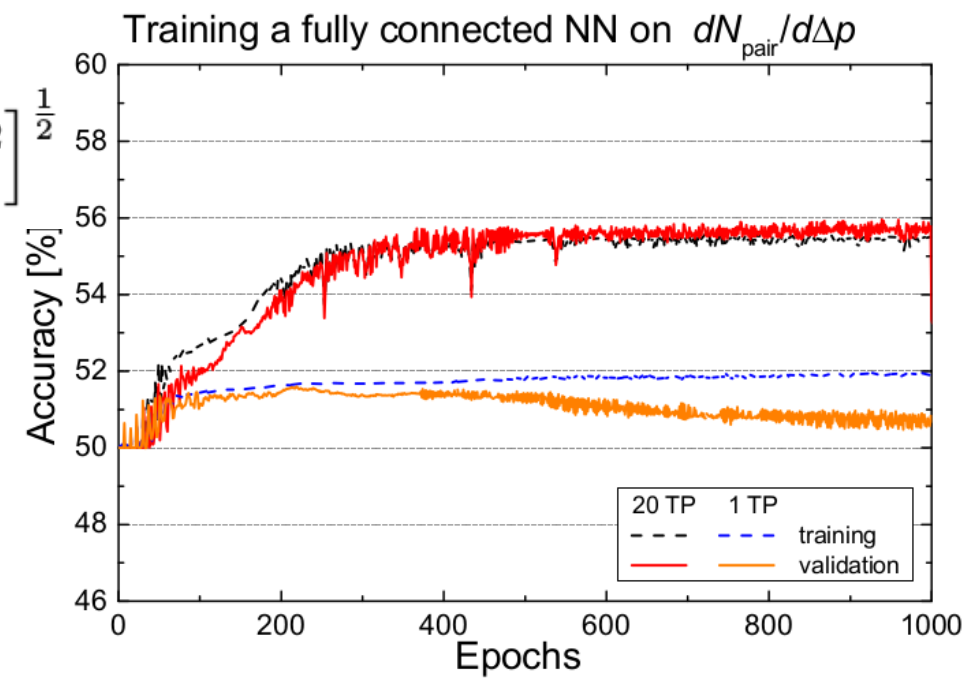


Hand crafted observables

- Can the 'signal' be encoded in the correlations?

  2-P Momentum difference distribution

$$\Delta p_{ab} \equiv \frac{1}{2}|\mathbf{p}^a - \mathbf{p}^b|$$
$$= \frac{1}{2}\left[(p_x^a - p_x^b)^2 + (p_y^a - p_y^b)^2 + (p_z^a - p_z^b)^2\right]^{\frac{1}{2}}$$



- With it, already a simple fully-connected network can get accuracy 55% (TP=20)

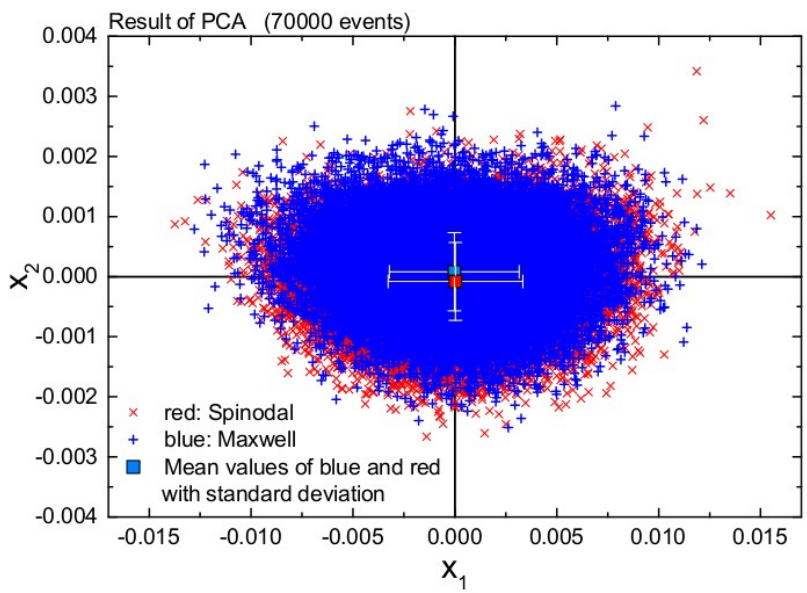
  Better than using the pure spectra !

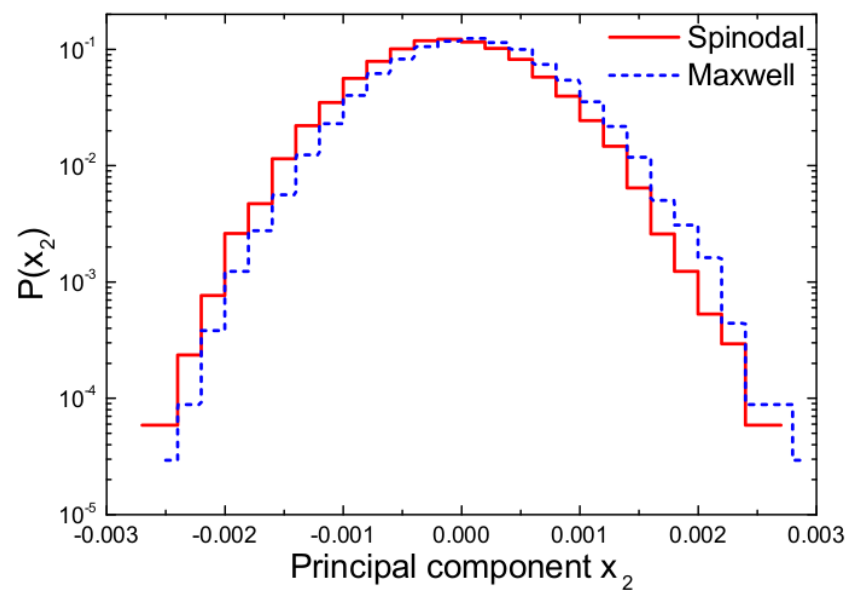
Unsupervised learning

- How to understand why only poor EbE discrimination was obtained here ?
  Do PCA on previous mom. diff. Distribution, and take the first two compoents

- Overlapped ! The mean values are similar but with small systematic shift.

- Spinodal prefer negative x2 : can produce events diff than average ones, but rare !
  – – this is why EbE analysis fails (only few events display characteristic features).

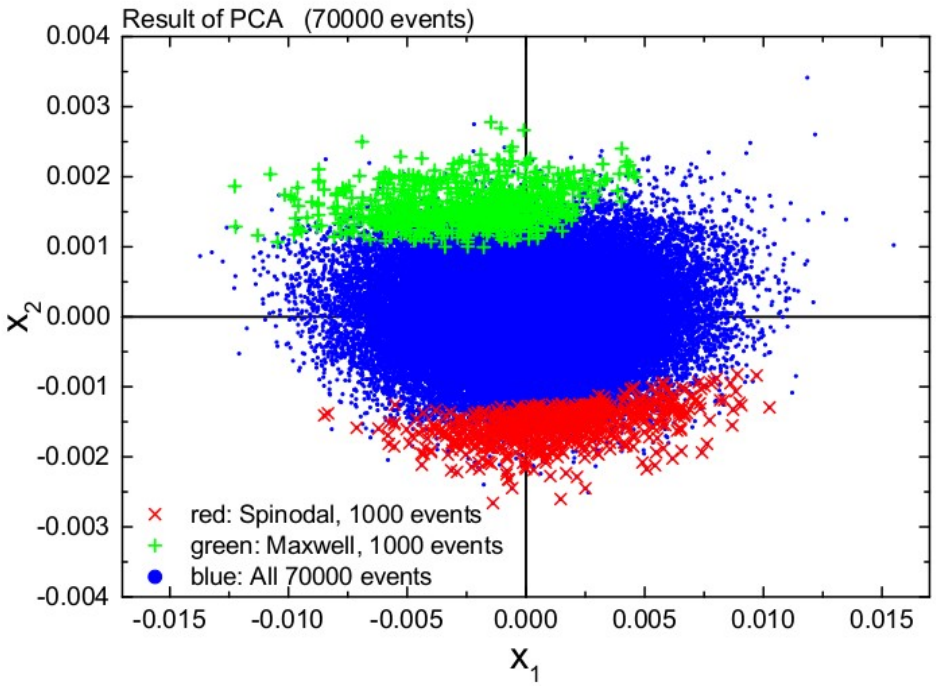




Unsupervised learning

- We can verify this suspicion by relating the network decision (probability) with the PCA components

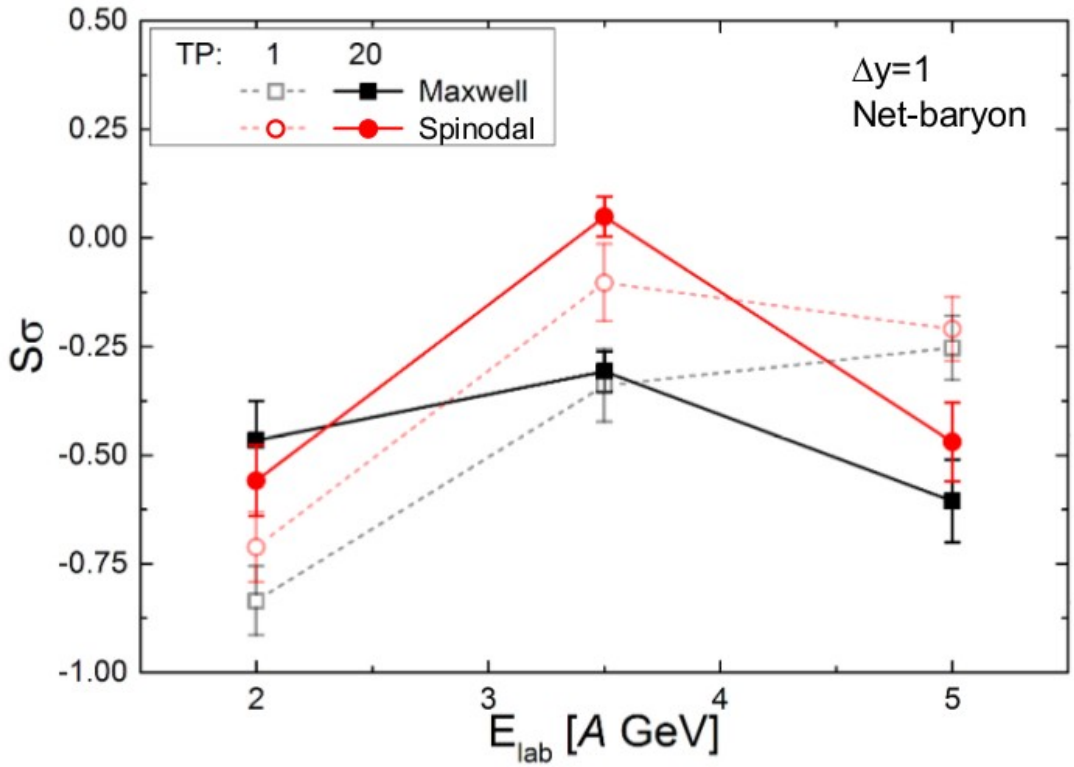
  The network kinds of separates events mostly according to x2 feature : focus on outliers (so, poor accuracy got for all events)



Need cumulative observables

- Cumulants helps for discriminating :

  Skewness for net baryons show enhancement when strongest clustering happened



Take-to-home perspectives

- phase transition non-equilibrium dynamics will give almost all events baryon clumping in coordinate space.
- Transformation from coordinate to momentum space will smear out most of the correlations, making event-by-event discrimination difficult.
- Unsupervised learning helps for understanding
- Point network can take more information for discrete particle list

JHEP 1912 (2019) 122

Further consider detector simulation

- To a more realistic situation, particles from high energy experiments will go through detector. To use really the raw experimental data for extracting physics, detector simulation is needed.

- We first try a simple task here :

  Can DL methods be used to determine the impact parameter (b) of a collision in CBM experiment from raw experimental output?

Phys. Lett. B 811 (2020) 135872

Training data simulation

Gathering data

1 UrQMD 3.4

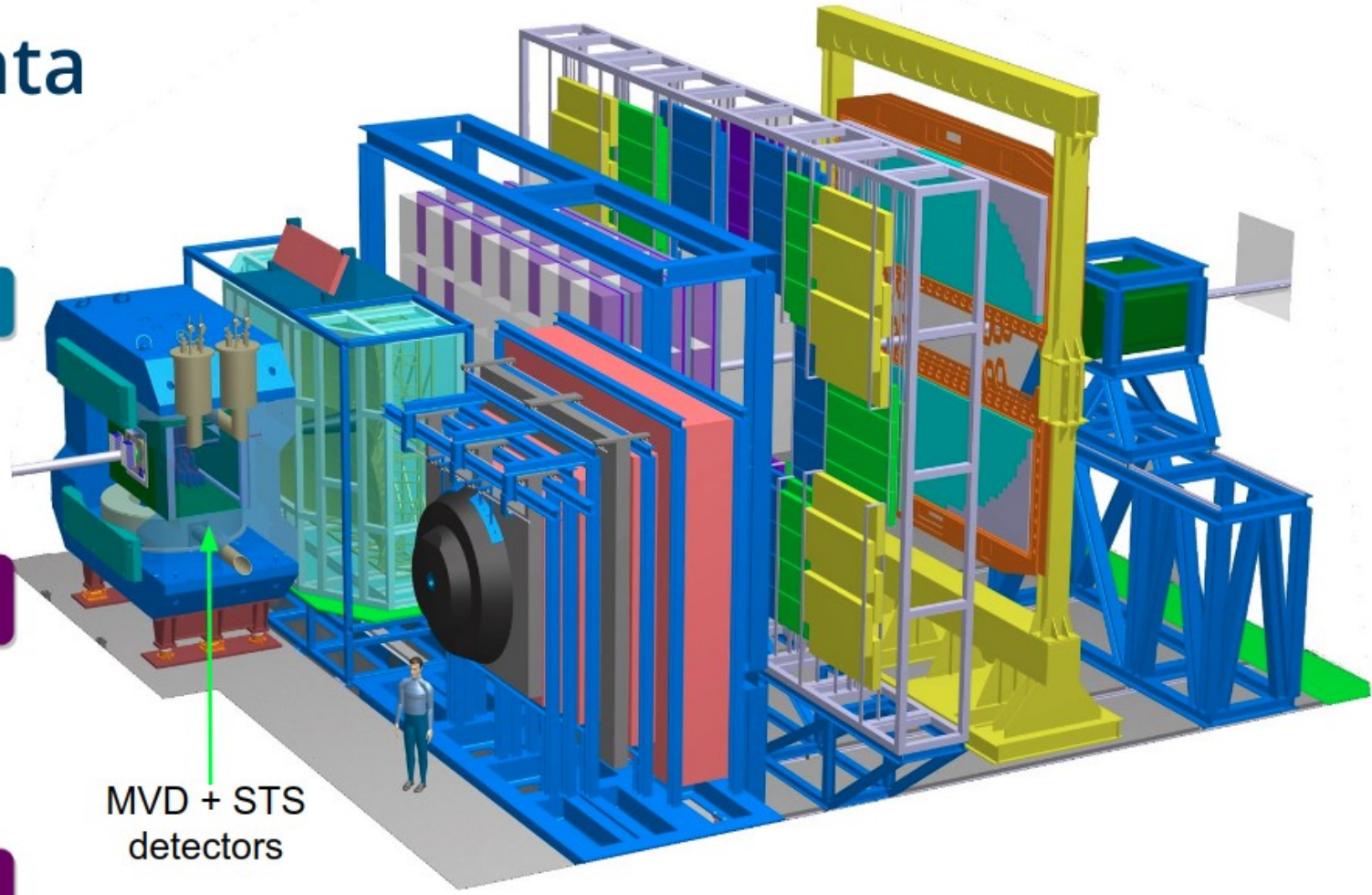
2 Unigen

3 CBMroot : Oct 2019 release

3 Macro: run_transport.C

3 Macro: run_digi.C

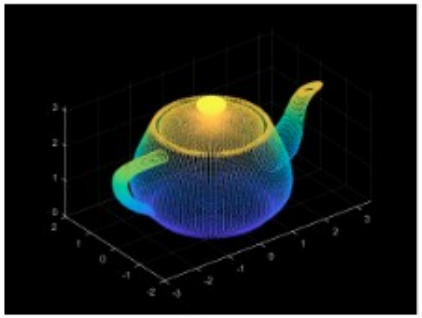
3 Macro: run_reco_event.C

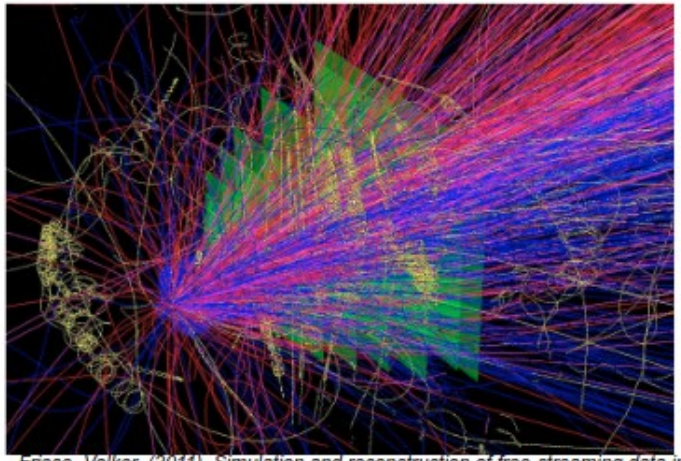


- Au+Au @ 10 AGeV

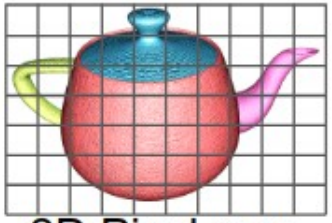
Point Clouds data

- Experimental data: tracks or hits of particles
  - Each particle is a point in a point cloud
  - The order of input shouldn't affect the output





Friese, Volker. (2011). Simulation and reconstruction of free-streaming data in CBM. Journal of Physics: Conference Series. 331. 032008. 10.1088/1742-6596/331/3/032008.



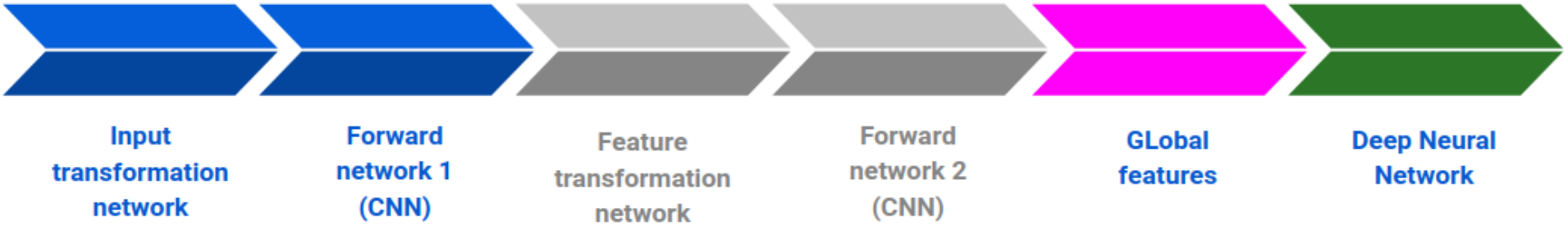
2D Pixel map

- 3D Voxels map
  - voluminous!
- Pointcloud: Unordered 2D array of (x,y,z) coordinates (or other point attributes) of each point
  - Efficient representation for higher dimensions

X1,y1,z1,.....
X2,y2,z2,.....
X3,y3,z3.....
. . .
xn,yn,zn,.....

PointNet based DL model

- Pointnet: Deep learning model for point clouds
  - Unordered
  - Invariant to transformations

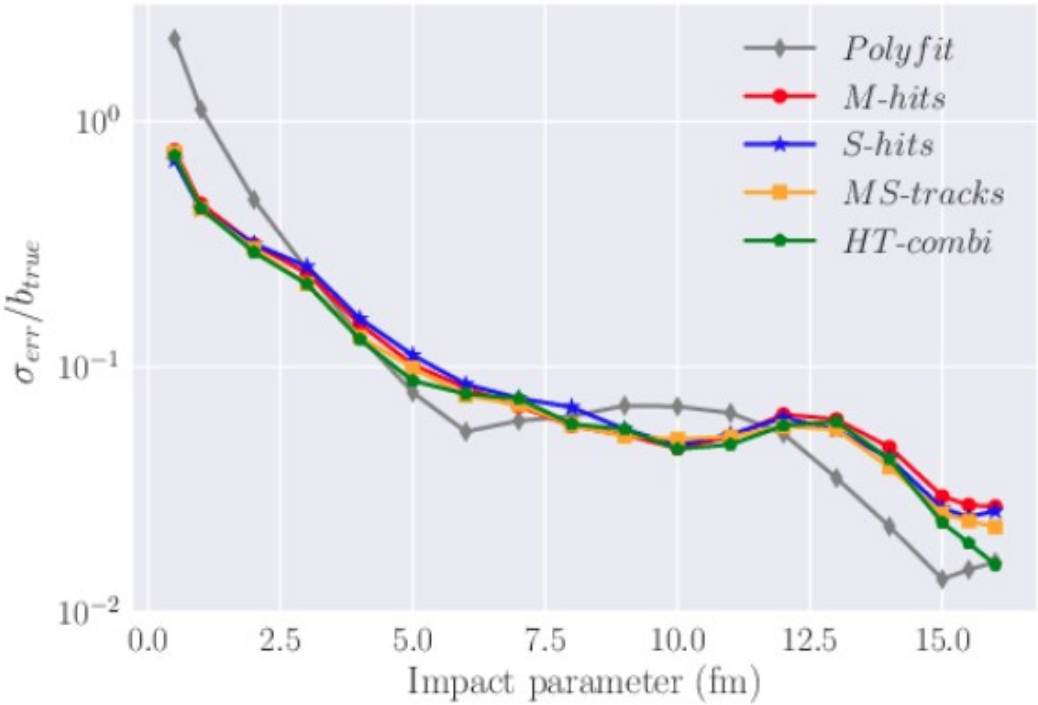


- We have developed pointnet based models which can reconstruct b from 4 different kinds of data

Models

01	M-hits	• (x,y,z) of all Hits in MVD
02	S-hits	• (x,y,z) of all hits in STS
03	MS-tracks	• (x,y,z,dx/dz,dy/dz,q/P) of all tracks in first and last plane from MVD+ STS
04	HT-combi	• Hits from MVD and tracks from MVD+STS
05	Polyfit (non-ML baseline)	• Third order polynomial fit to multiplicity vs. impact parameter curve

Results : relative precision

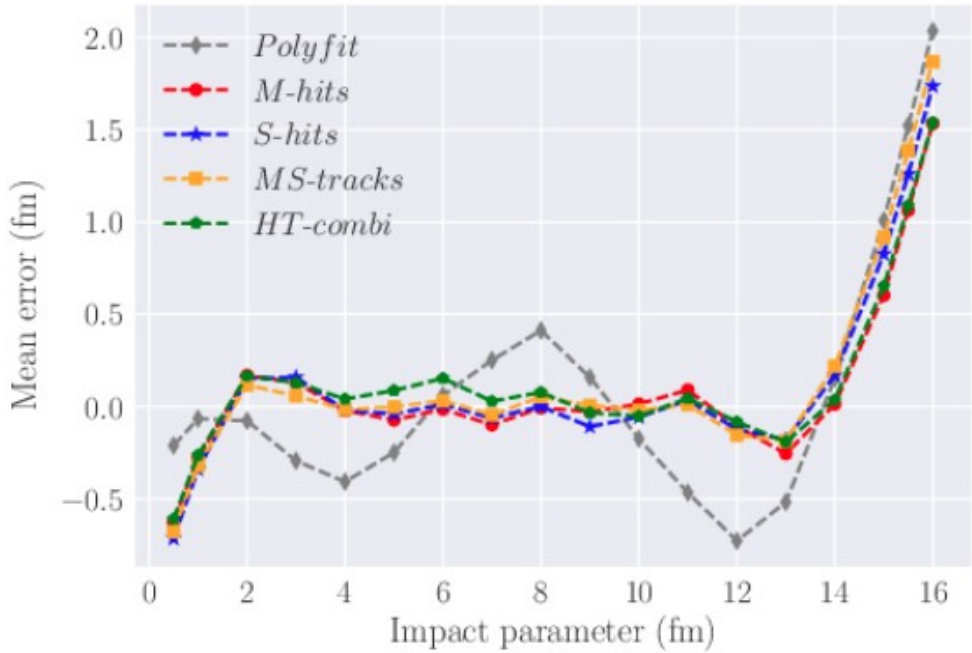


σ_{err} = standard deviation of error in predictions (err= true-predicted)

- Quantifies precision in predictions
- Polyfit fails for central events!
- Similar precision for b>3 fm

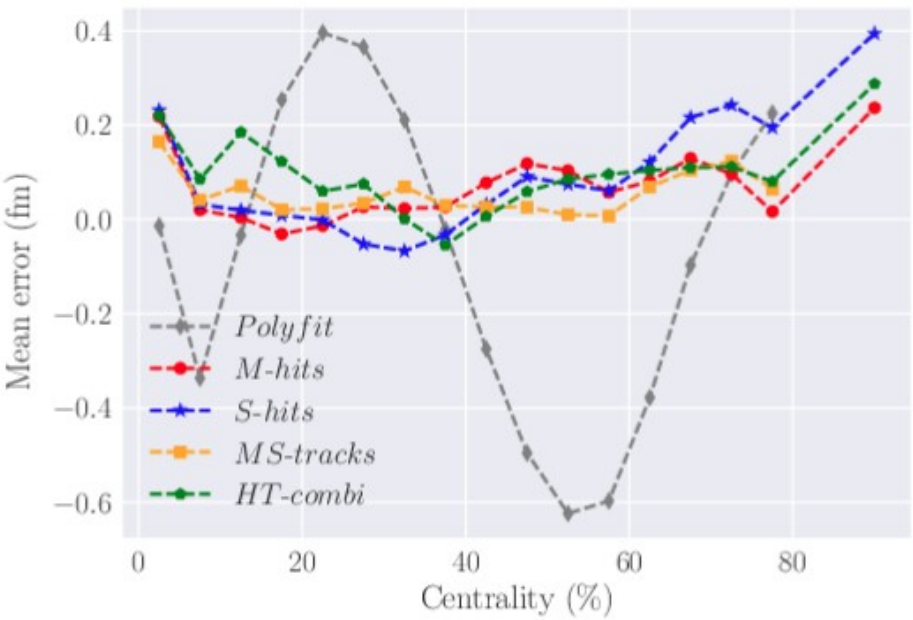
However, the predictions are accurate only if the mean error is close to zero!

Further try Point Cloud Network



- Quantifies accuracy in predictions
- DL models have mean error between -0.3 and 0.2 fm for b= 2- 14 fm
- Polyfit is highly fluctuating
- All models are less accurate for peripheral events

Further try Point Cloud Network



- Data: b=0-16 fm, 1 million events , bdb distribution
- Simulates the realistic b distribution in experiments
- Different from the b distribution of training data
- 5% Centrality bins defined on STS track multiplicity
- DL models have mean error close to zero for most centrality classes

Take-to-home perspectives

The deep learning models outperforms conventional methods for impact parameter determination

- Reconstruct the impact parameter on an **event by event** basis
- Prediction speed upto 1000 events/ s on single GPU
  - Real time analysis of collected data
- Reconstruct the impact parameter using the **hit information alone**
  - Run time AI event selector
  - Detect faults in detector during data taking
- Robust to small changes in physics model in comparison to conventional models
- General framework which can be used for other tasks (e.g. flow extraction)

Phys. Lett. B 811 (2020) 135872

Deep learning in many-body theory study

Phys.Rev. D100 (2019) no.1, 011501

arXiv: 2005.04857

ArXiv: 2007.01037

1+1d $\lambda\phi^4$ (prepare training set)

regularization of continuum Action:

$$S^{\rm lat} = \sum_x \left\{ (4+m^2)\phi^*(x)\phi(x) + \lambda[\phi^*(x)\phi(x)]^2 - \sum_{\nu=1,2} \left[e^{\mu\delta_{\nu,2}}\phi^*(x)+\hat{\nu}) + e^{-\mu\delta_{\nu,2}}\phi^*(x)\phi(x-\hat{\nu}) \right] \right\}$$

Partition sum :

$$\mathcal{Z} = \int D[\phi] exp\left(-S^{\rm lat}[\phi]\right)$$

Dualization approach :

$$\mathcal{Z} = \sum_{\{k,\ell\}} \prod_n \left\{ e^{\mu k_t(n)} \cdot W[s(n)] \cdot \delta[\nabla \cdot k(n)] \cdot \prod_\nu A[k_\nu(x), \ell_\nu(x)] \right\}$$

$$W[s(n)] = \int_0^\infty {\rm d}r\, r^{s(n)+1}\, e^{-(4+m^2)r^2-\lambda r^4}$$

$$s(n) = \sum_\nu [|k_\nu(n)| + |k_\nu(n-\hat{\nu})| + 2(\ell_\nu(n) + \ell_\nu(n-\hat{\nu})]$$

$$A[k_\nu(x), \ell_\nu(x)] = \frac{1}{(\ell_\nu(n) + |k_\nu(n)|)!\ \ell_\nu(n)!}$$

***C. Gattringer and T. Kloiber*, Nucl. Phys. B869 (2013) 56–73**

O. Orasch and C. Gattringer, Int. J.Mod.Phys.A33(2018) no.01,1850010,

Dualization approach for 1+1d $\lambda\phi^4$

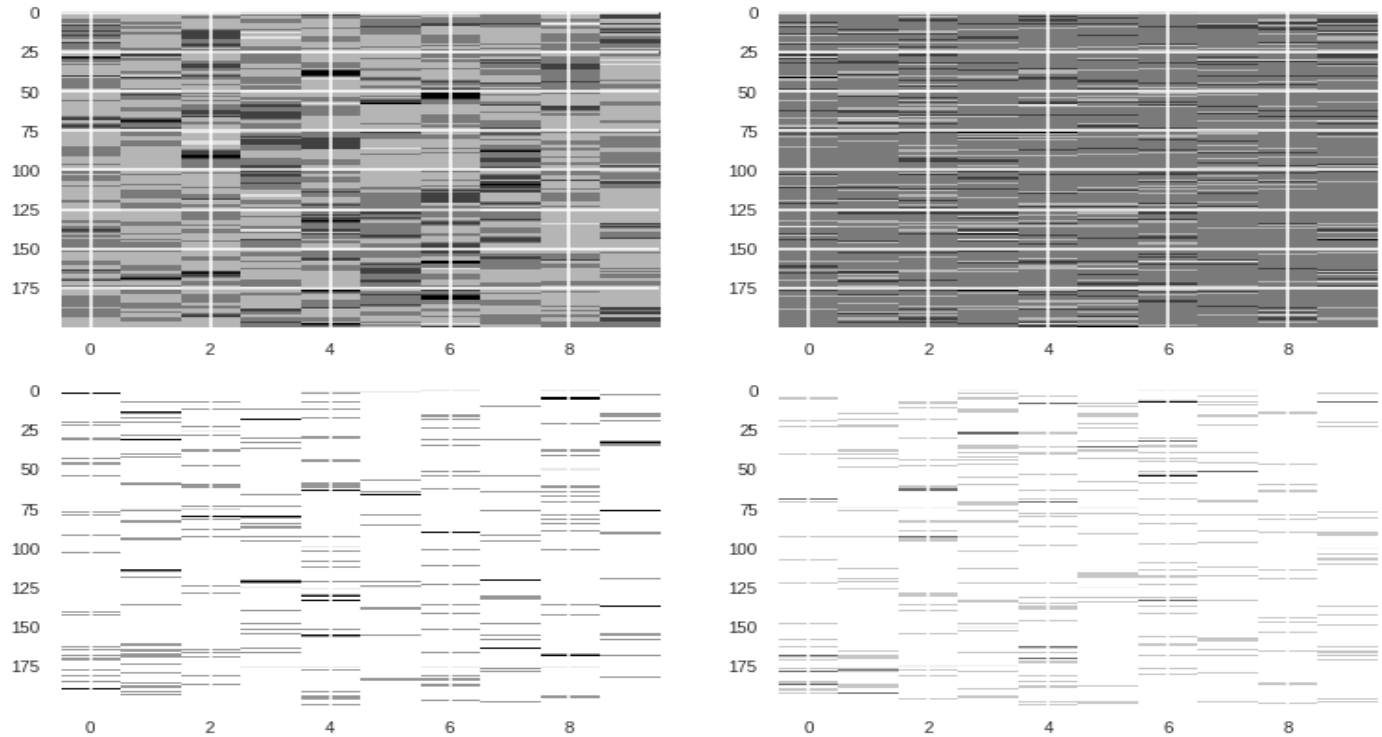
configurations – 4 integer-valued variables : $k_t, \quad k_x, \quad l_t, \quad l_x$

$$m = 0.1$$

$$\lambda = 1.0$$

$$N_t = 200$$

$$N_x = 10$$



C. Gattringer and T. Kloiber, Nucl. Phys. B869 (2013) 56–73

O. Orasch and C. Gattringer, Int. J.Mod.Phys.A33(2018) no.01,1850010,

Divergence constraint : $$\nabla \cdot k(n) = \sum_{\nu} [k_\nu(n) - k_\nu(n - \hat{\nu})] = 0$$

Observables : n and $|\phi|^2$

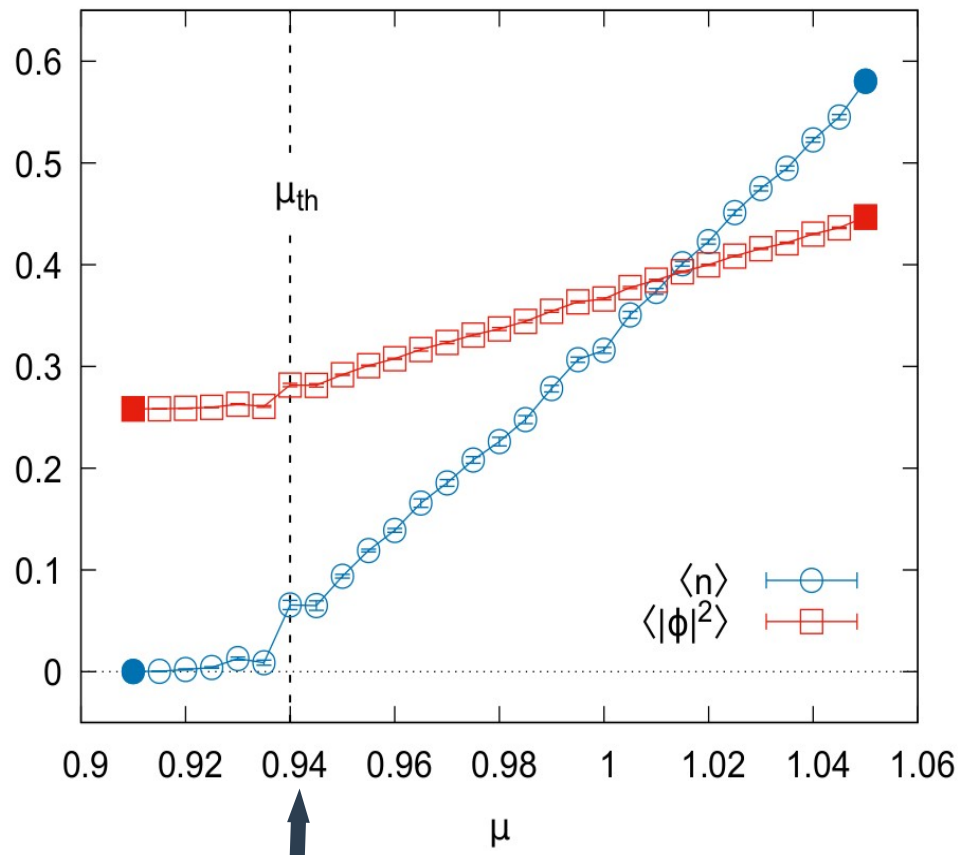
Grand canonical ensemble

$$\langle n \rangle = \frac{T}{L}\frac{\partial \log \mathcal{Z}}{\partial \mu}$$

$$n = \frac{1}{N_x N_t a}\sum_n k_t(n)$$

$$\langle |\phi|^2 \rangle = \frac{T}{L}\frac{\partial \log \mathcal{Z}}{\partial (m^2)}$$

$$|\phi|^2 = \frac{1}{N_x N_t}\sum_n \frac{W[s(n)+2]}{W[s(n)]}$$



Condensation sets in at $\mu_{th} \sim m_{phys} \sim 0.94$

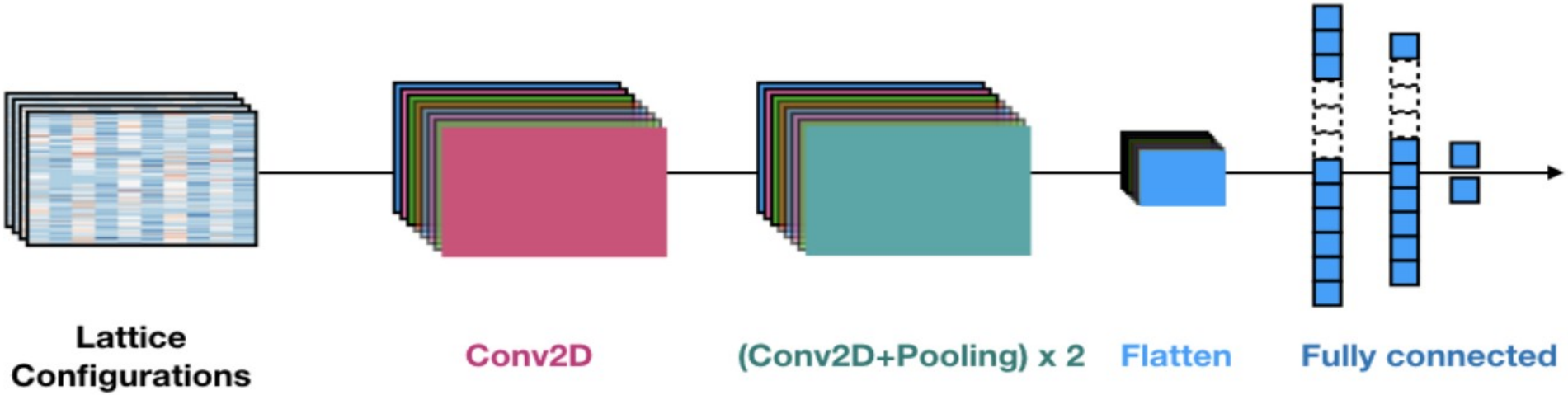
Exploring NN application here

(1) Classification : detect 'phase transition' status based on configurations
(identify order parameter)

(2) Regression : physical observables regression
(identify thermodynamics)

(3) GAN(generate) : * Learn to generate new configurations
* Generate configs with proper distribution
(identify partition function)

DCNN Architechture – Classification

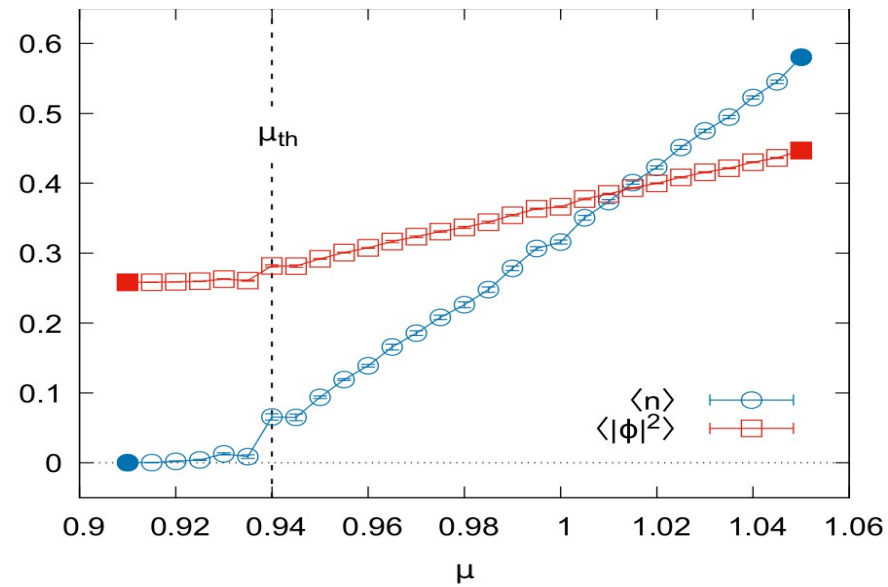


Training set consist two ensembles of configs at

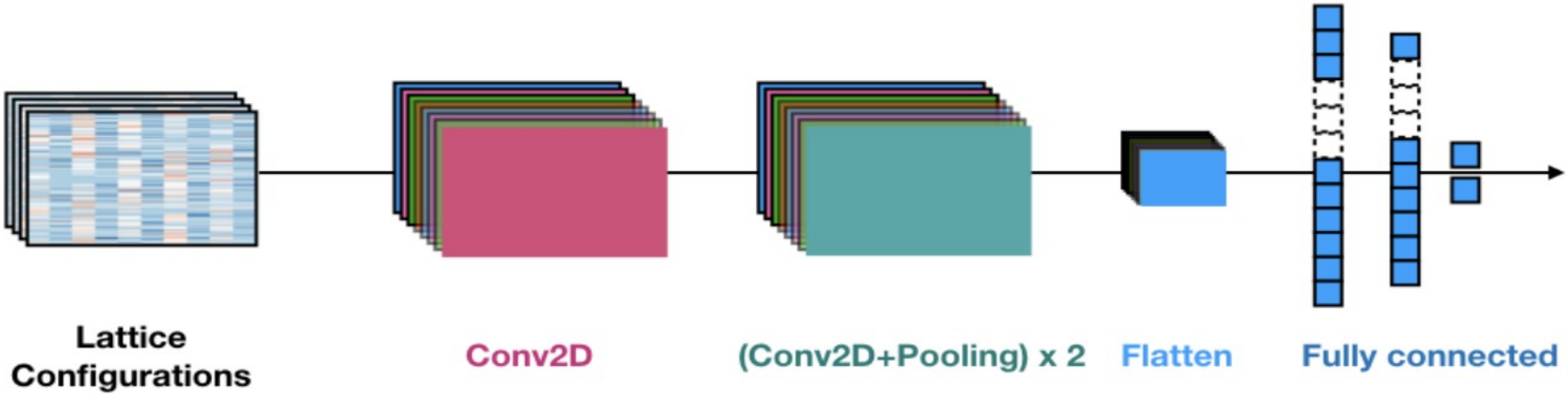
$$\mu = 0.91 \quad \text{with label} \quad y = (0, 1)$$

and

$$\mu = 1.05 \quad \text{with label} \quad y = (1, 0)$$



DCNN Architechture – Classification



Training set consist two ensembles of configs at

$$\mu = 0.91 \quad \text{with label} \quad y = (0, 1)$$

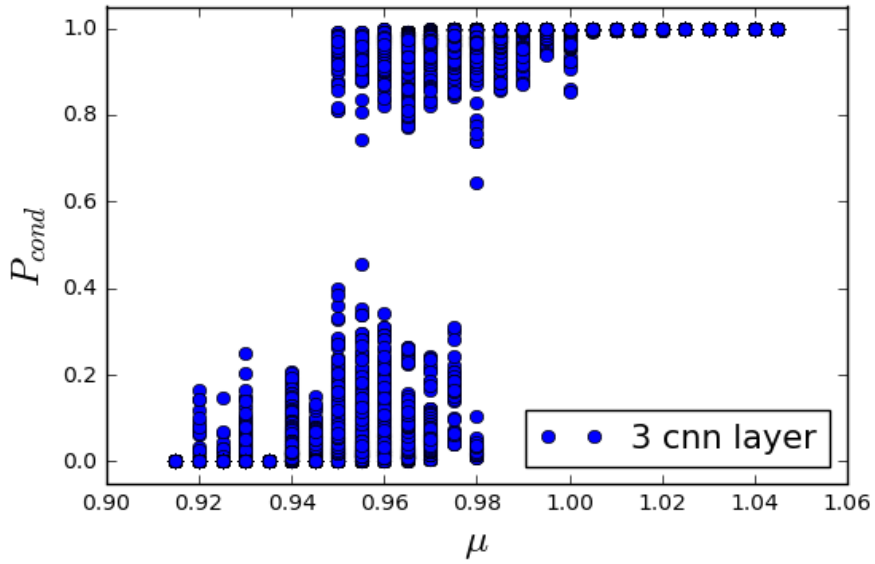
and

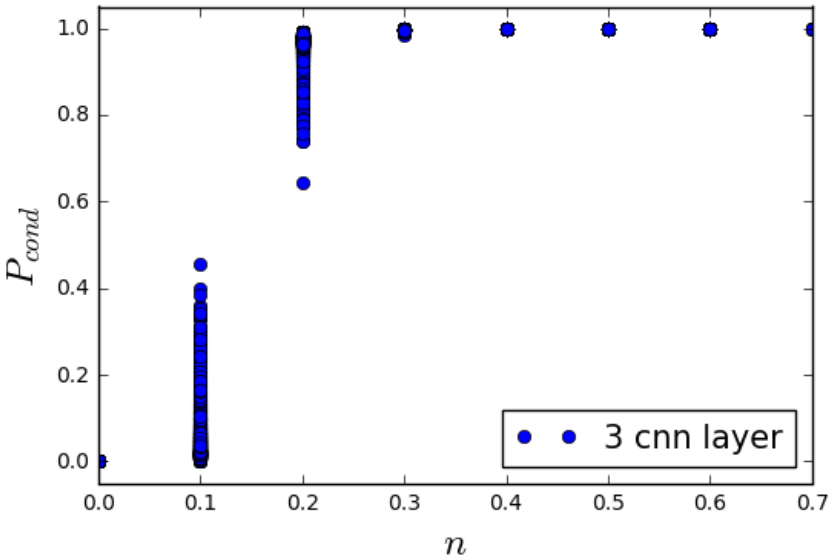
$$\mu = 1.05 \quad \text{with label} \quad y = (1, 0)$$

Testing set consist of different ensembles of configurations at different chemical potential

$$0.91 < \mu < 1.05$$

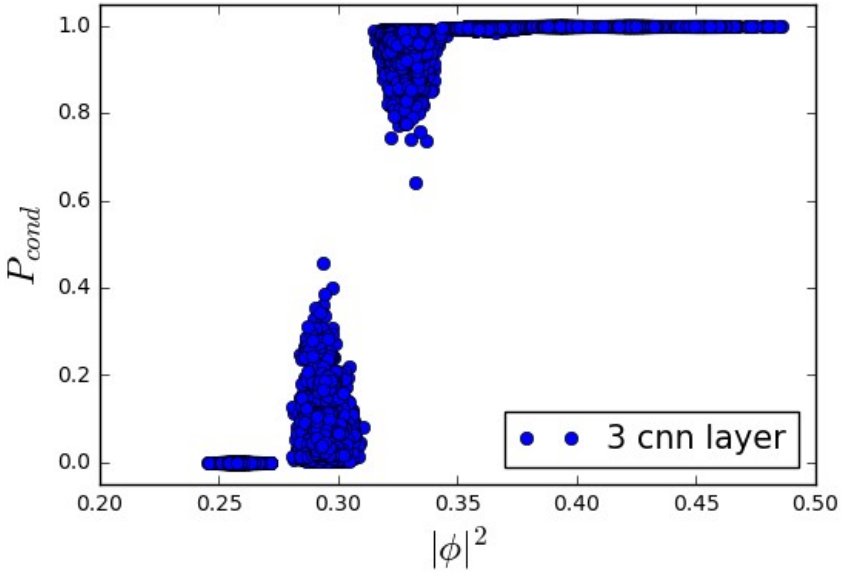
Condensation probability from DCNN





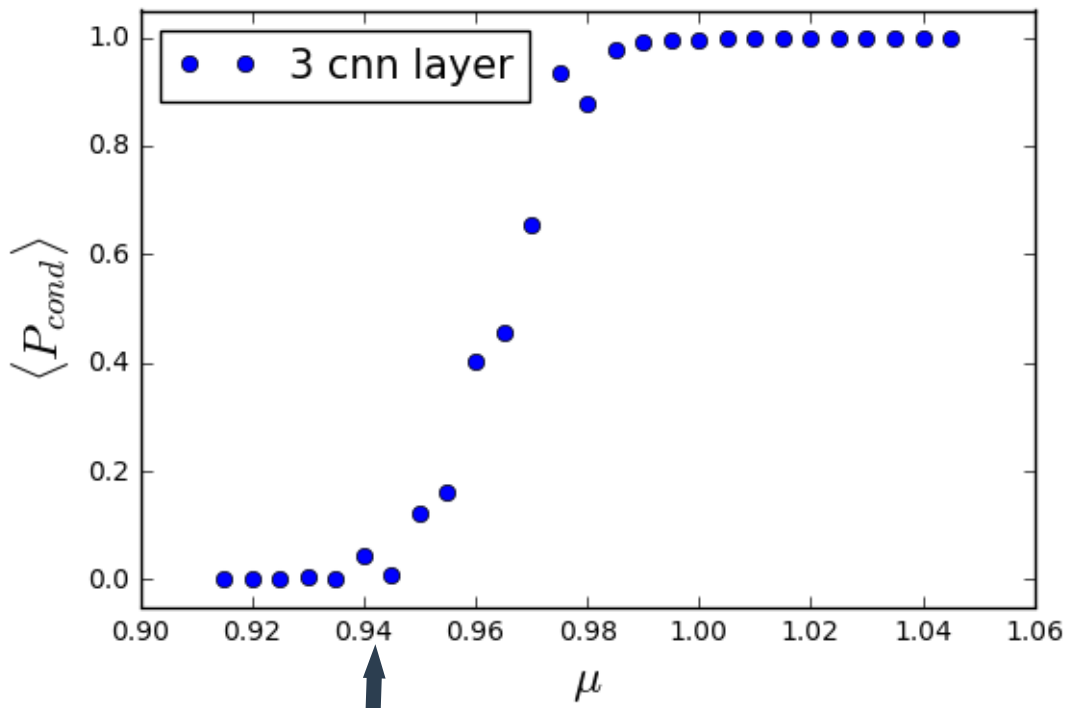
Strong correlation between P_cond and observables :

n / squared field



Ensemble average cond-probability

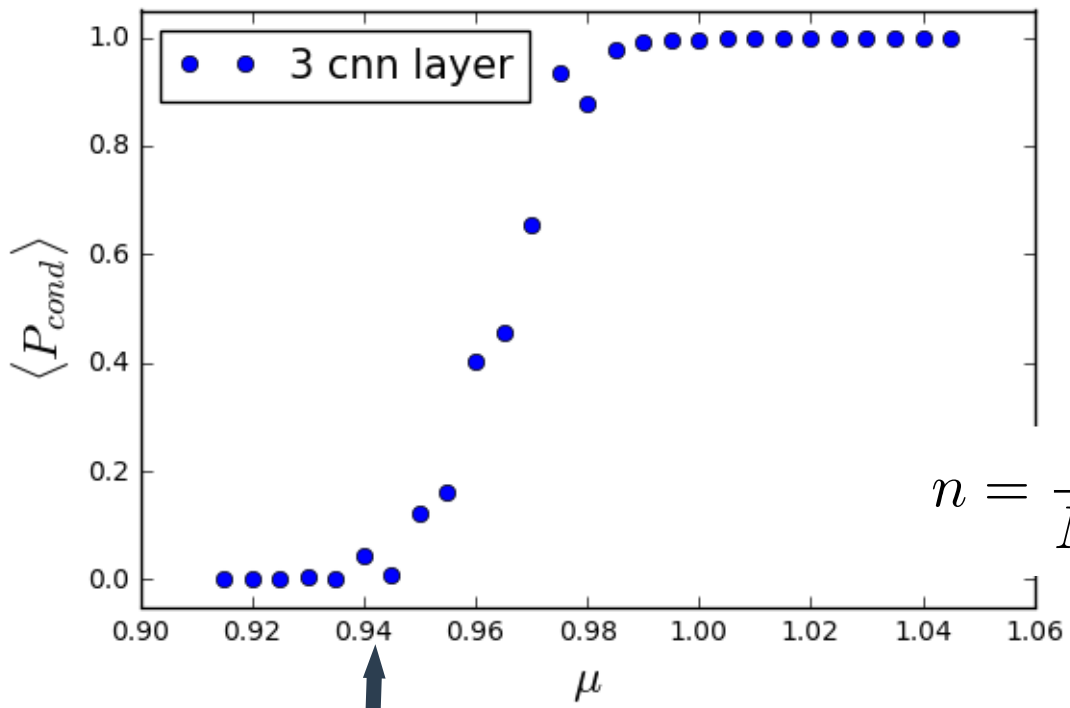
Classifier of the phases : $\langle n \rangle = 0 \quad \text{and} \quad \langle n \rangle \neq 0$



$$\mu_{th}(\langle P_{cond} \rangle > 0) \sim \mu_{th}(\langle n \rangle > 0)$$

Ensemble average cond-probability

Classifier of the phases : $\langle n \rangle = 0 \quad \text{and} \quad \langle n \rangle \neq 0$



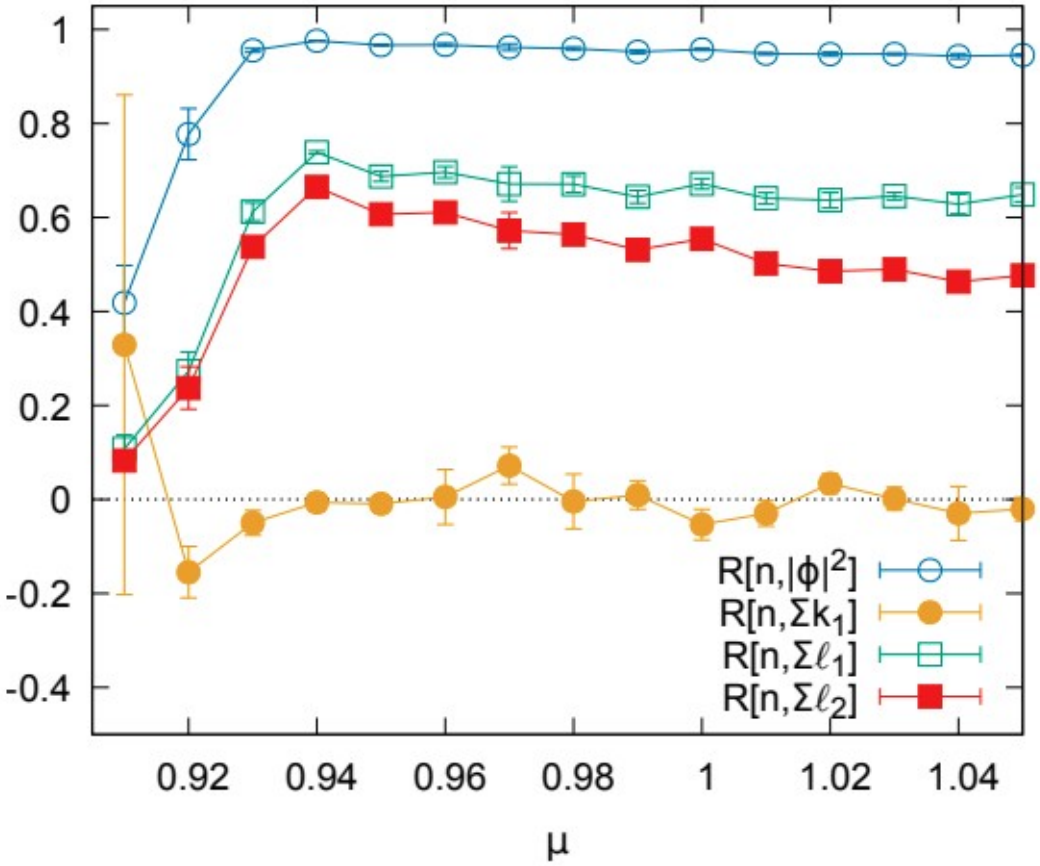
$$n = \frac{1}{N_x N_t a} \sum_n k_t(n)$$

$$\mu_{th}(\langle P_{cond} \rangle > 0) \sim \mu_{th}(\langle n \rangle > 0)$$

Discard kt information

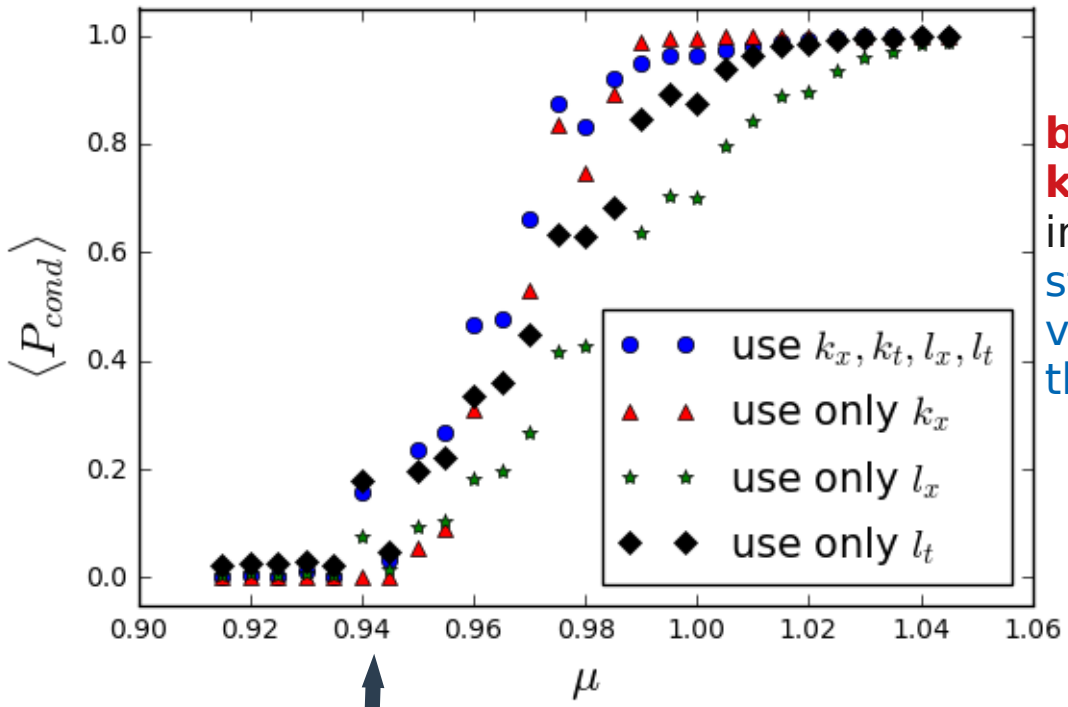
There's finite correlation between kt and l variable, but, no correlation bet. kt and kx

$$R[A,B] \equiv \frac{\langle AB\rangle - \langle A\rangle\langle B\rangle}{\sqrt{\langle A^2\rangle - \langle A\rangle^2}\sqrt{\langle B^2\rangle - \langle B\rangle^2}}$$



Try different field component variables

The same transition point, even use only kx!

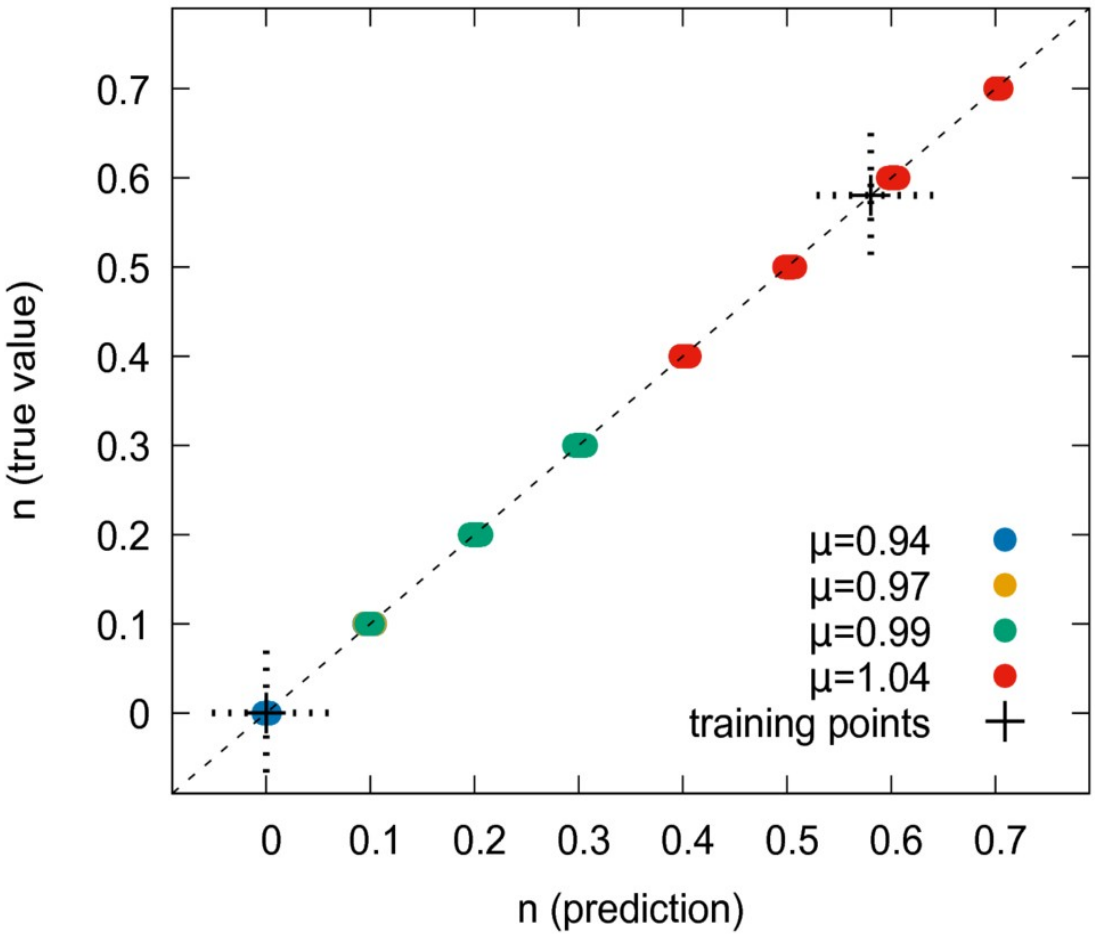


beyond conventional knowledge : indicating hidden structures in the kx variables and not only in the kt variables.

$$\mu_{th}(\langle P_{cond} \rangle > 0) \sim \mu_{th}(\langle n \rangle > 0)$$

regression for particle density n

Note, for training, only used $\mu = 0.91 \quad \text{and} \quad \mu = 1.05$

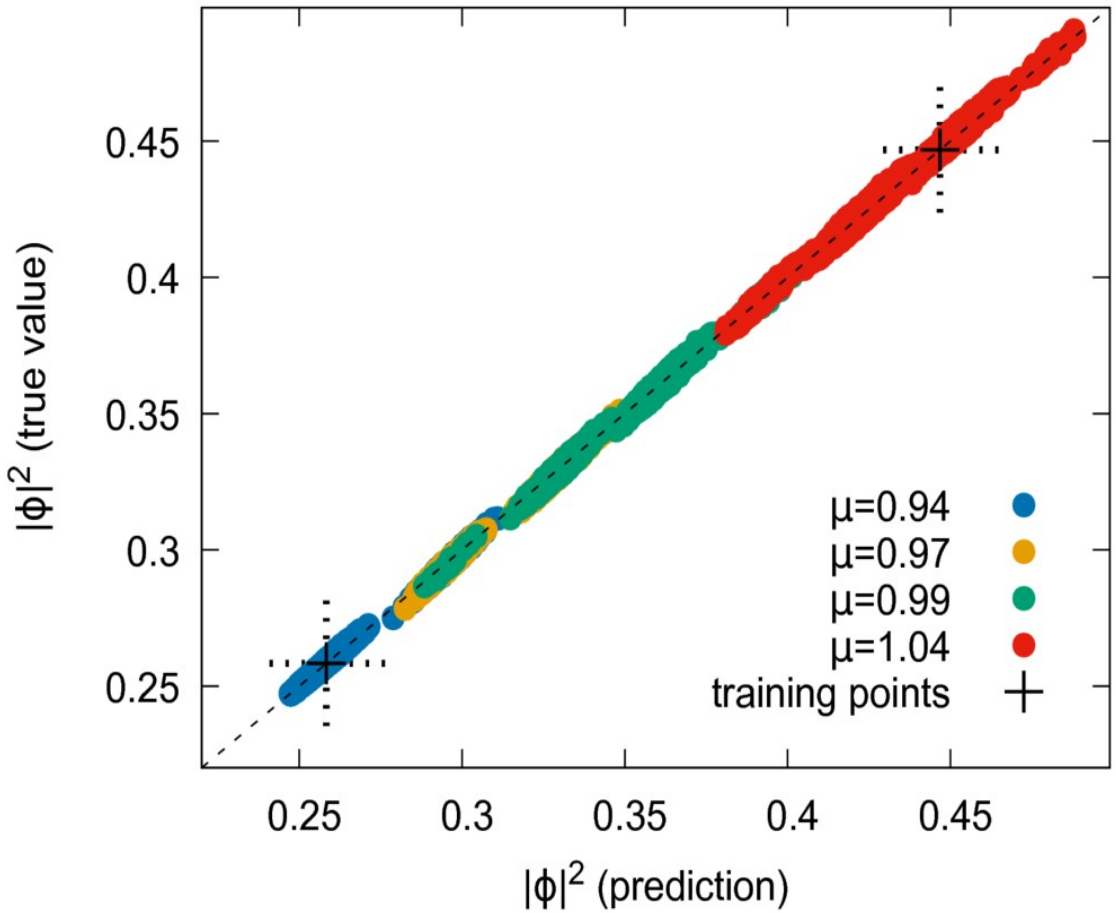


$$n = \frac{1}{N_x N_t a} \sum_n k_t(n)$$

$$RMSE < 0.003$$

regression for squared field ϕ^2

Note, for training, only used $\mu = 0.91 \quad \text{and} \quad \mu = 1.05$



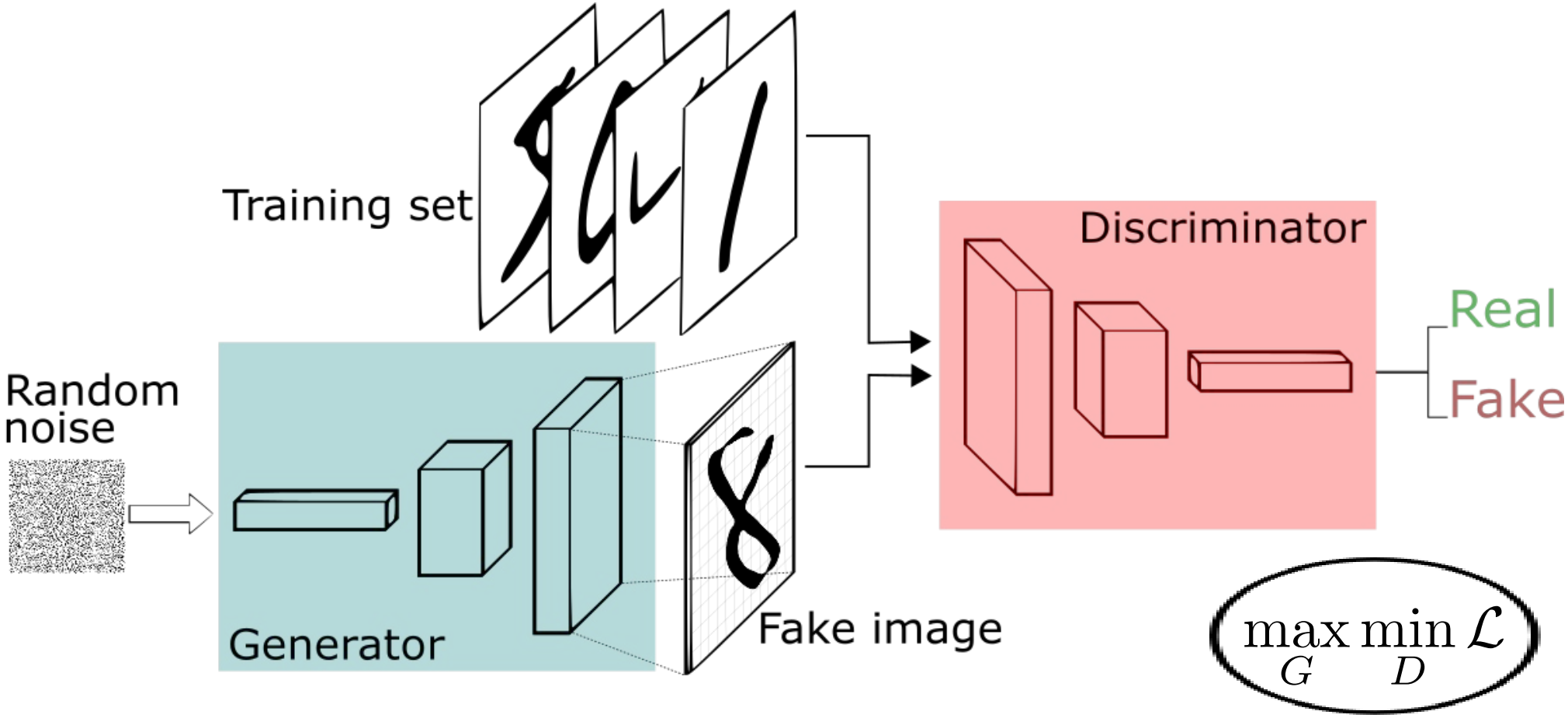
$$|\phi|^2 = \frac{1}{N_x N_t} \sum_n \frac{W[s(n)+2]}{W[s(n)]}$$

$$W[s(n)] = \int_0^\infty \mathrm{d}r\, r^{s(n)+1}\, e^{-(4+m^2)r^2 - \lambda r^4}$$

$$s(n) = \sum_\nu [|k_\nu(n)| + |k_\nu(n-\hat{\nu})| + 2(\ell_\nu(n) + \ell_\nu(n-\hat{\nu})$$

$$RMSE < 0.005$$

Gnerative Adversarial Network



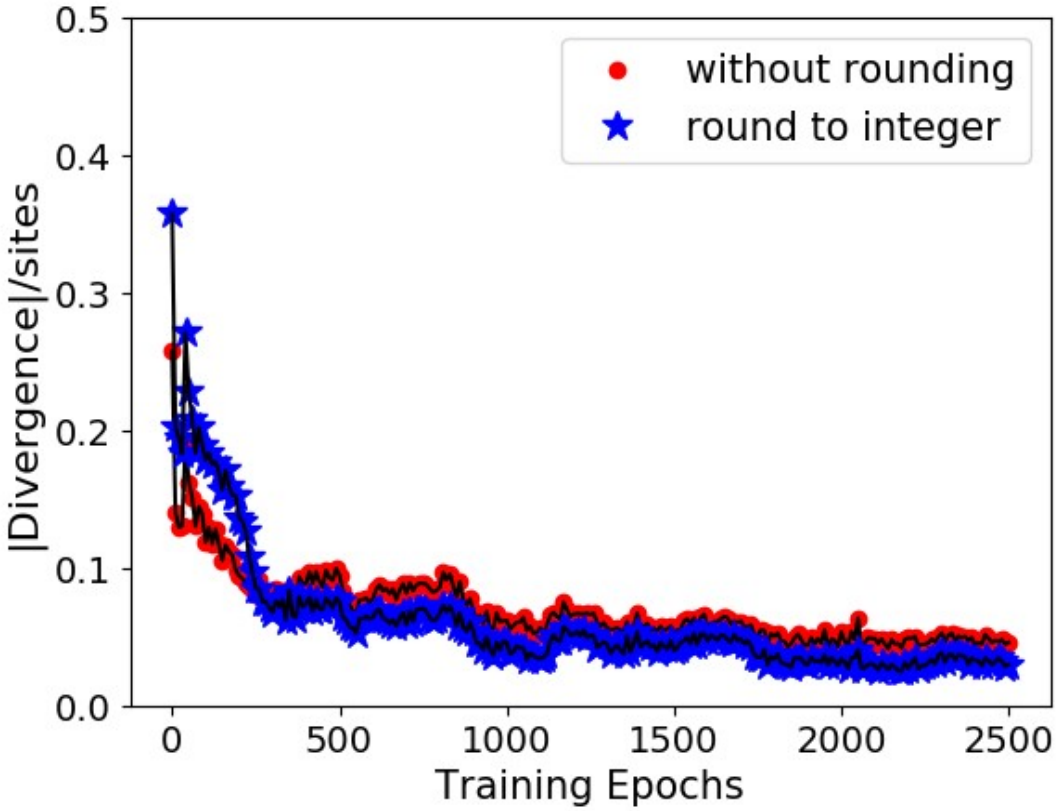
$$\mathcal{L} = -\mathbb{E}_{\hat{x}\sim p_r(\hat{x})}[\log(D(\hat{x}))] - \mathbb{E}_{z\sim p(z)}[\log(1 - D(G(z)))]$$

GAN – generate proper configurations

The divergence condition automatically get learned :

'Physical' configs can be generated

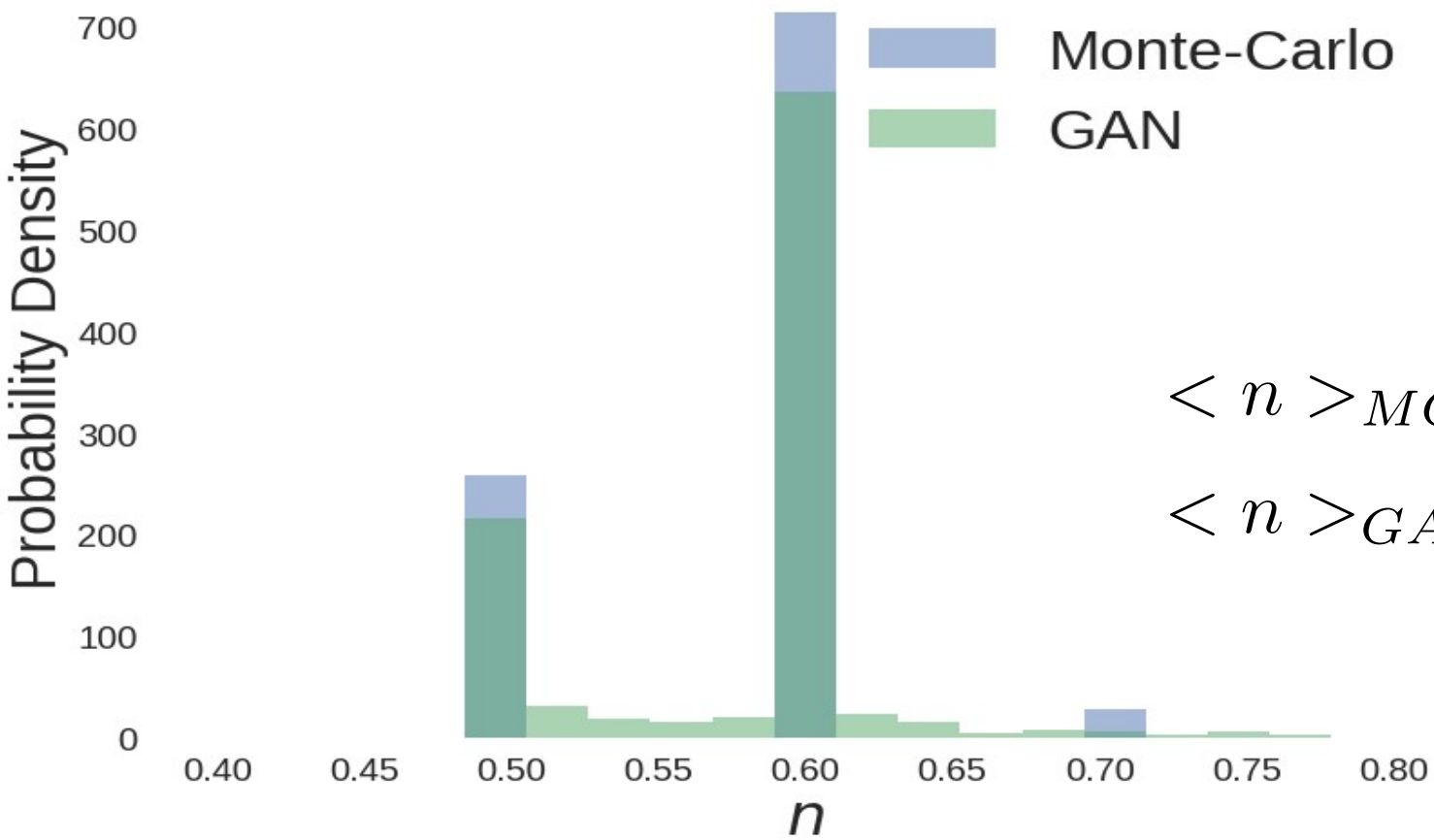
$$\nabla \cdot k(n) = \sum_{\nu} [k_\nu(n) - k_\nu(n - \hat{\nu})] = 0$$



Automatically capture the implicit physical constraint!

Fine tune GAN

Number density n distribution with 1k configs (after 6k trainig epochs):

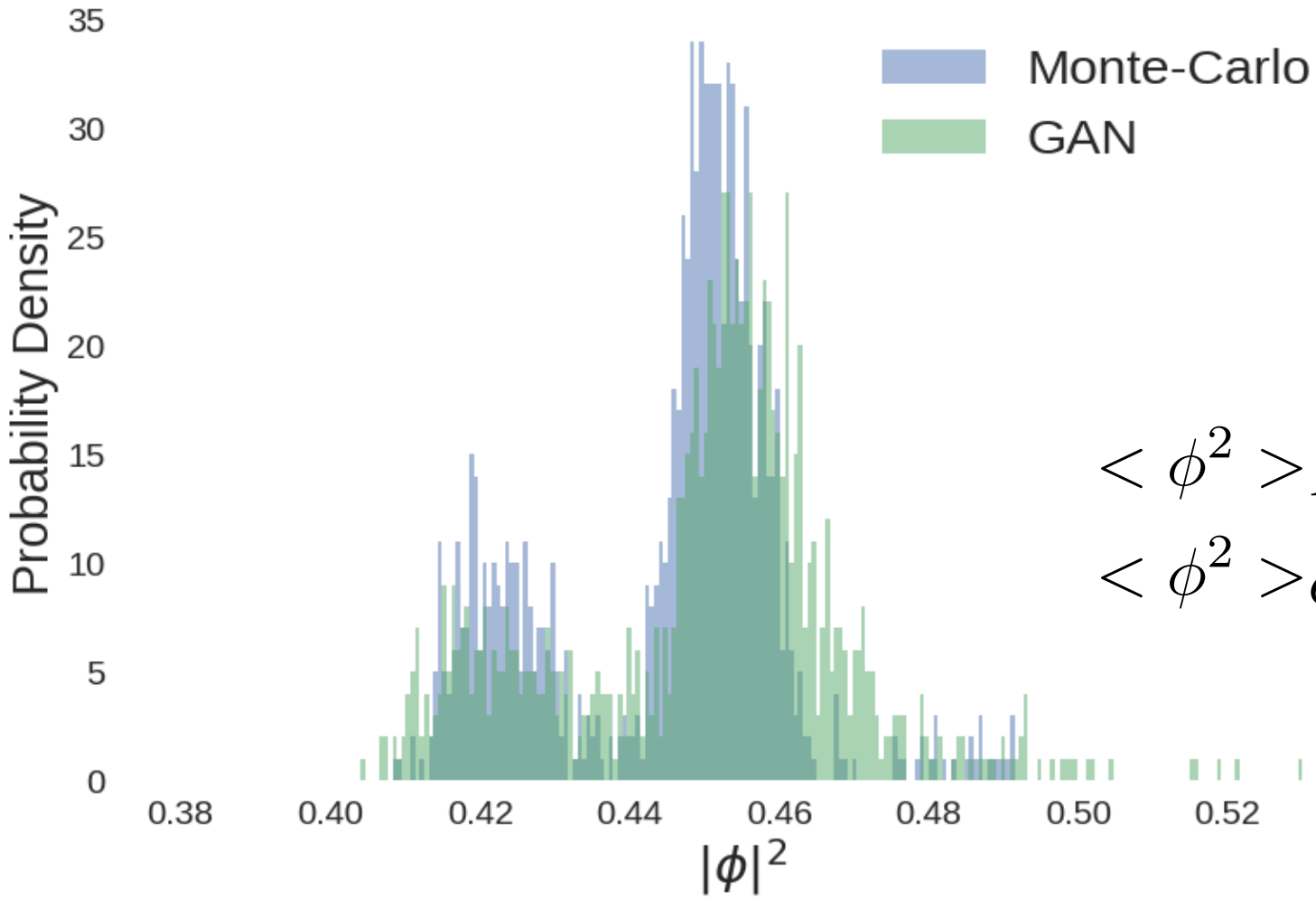


$$< n >_{MC} = 0.580$$

$$< n >_{GAN} = 0.578$$

Fine tune GAN

Squared field distribution with 1k configs (after 6k training epochs):



$$< \phi^2 >_{MC} = 0.447$$

$$< \phi^2 >_{GAN} = 0.449$$

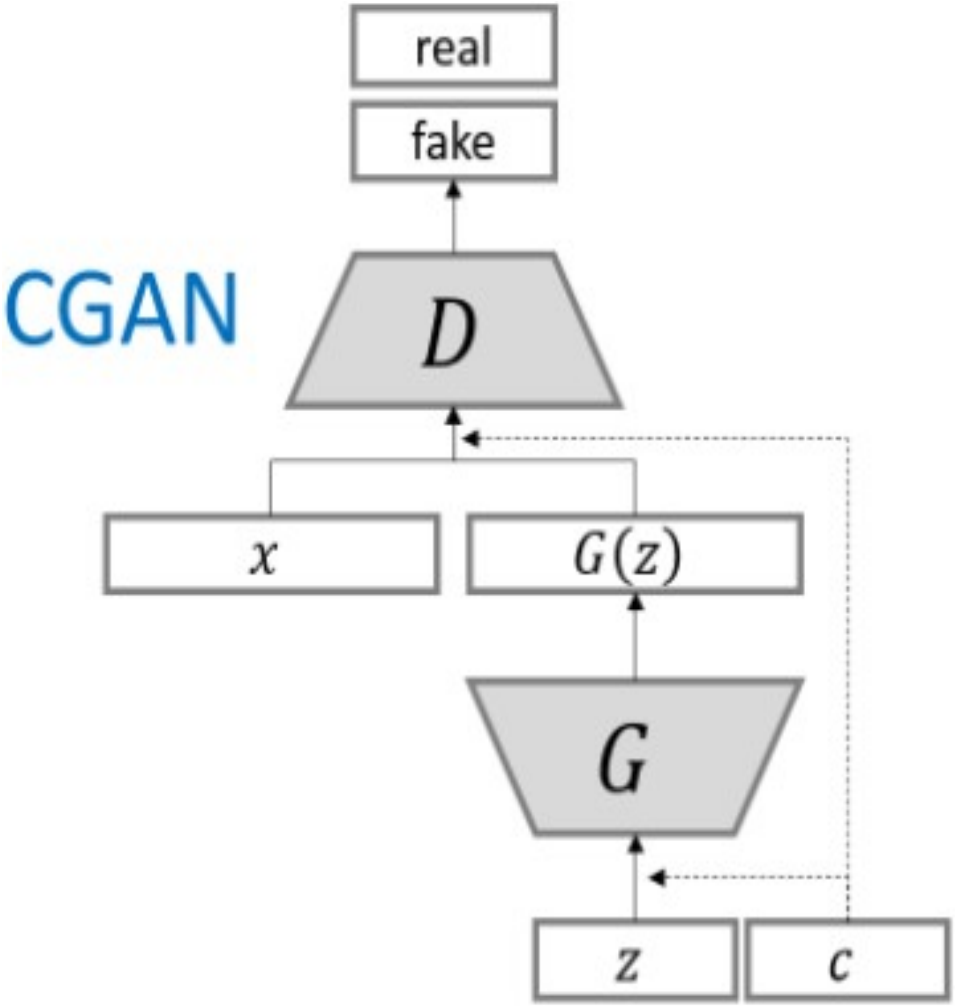
add conditional information n

make GAN conditional on particle density n :

We train GAN using one esemble with $\mu = 1.05$ labeled as well with n (including <u>n=0.4, 0.5, 0.6, 0.7</u>),

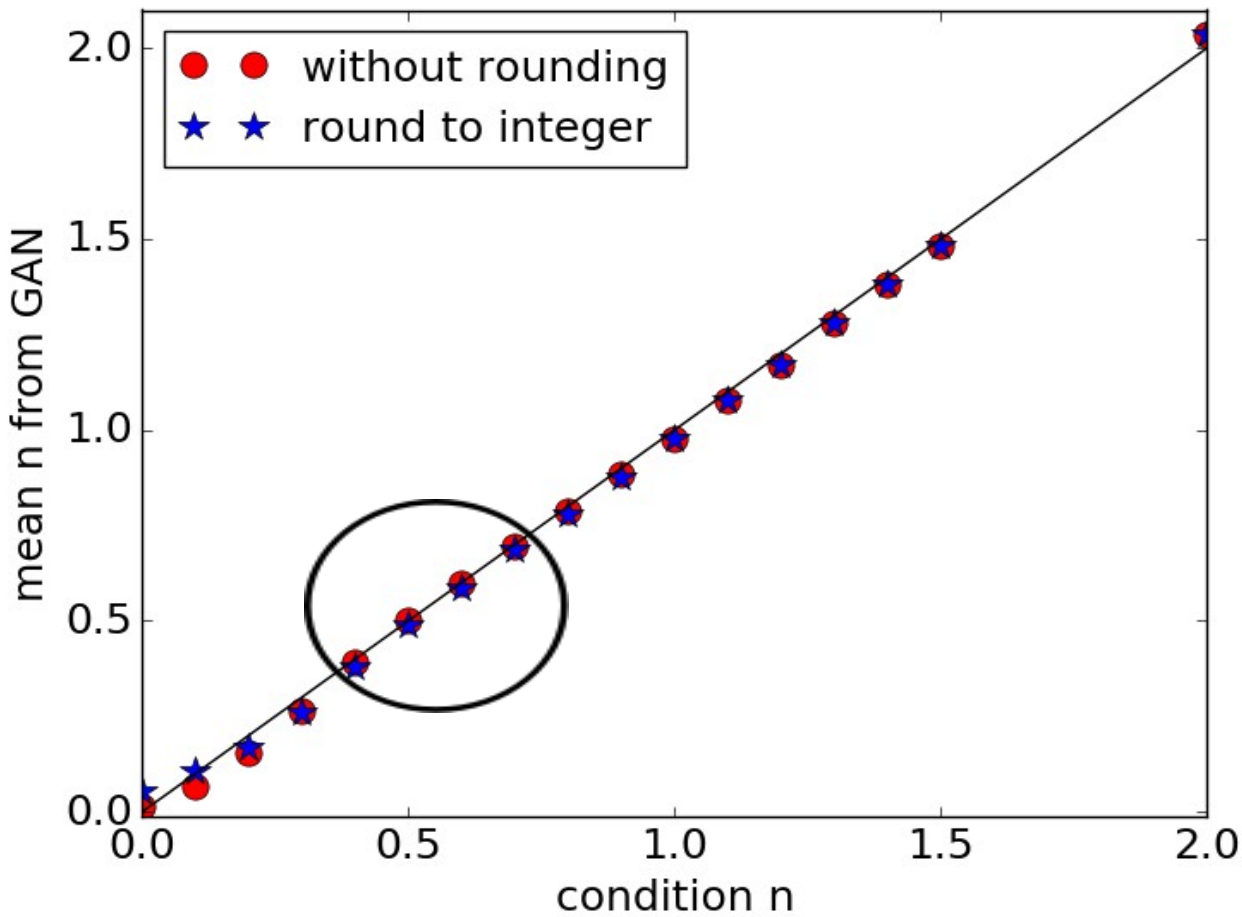
Once trained, in generating stage,

We specify different n values.



Conditional GAN

mean value for n and squared field of generated ensemble is controlled by condition in c-GAN.



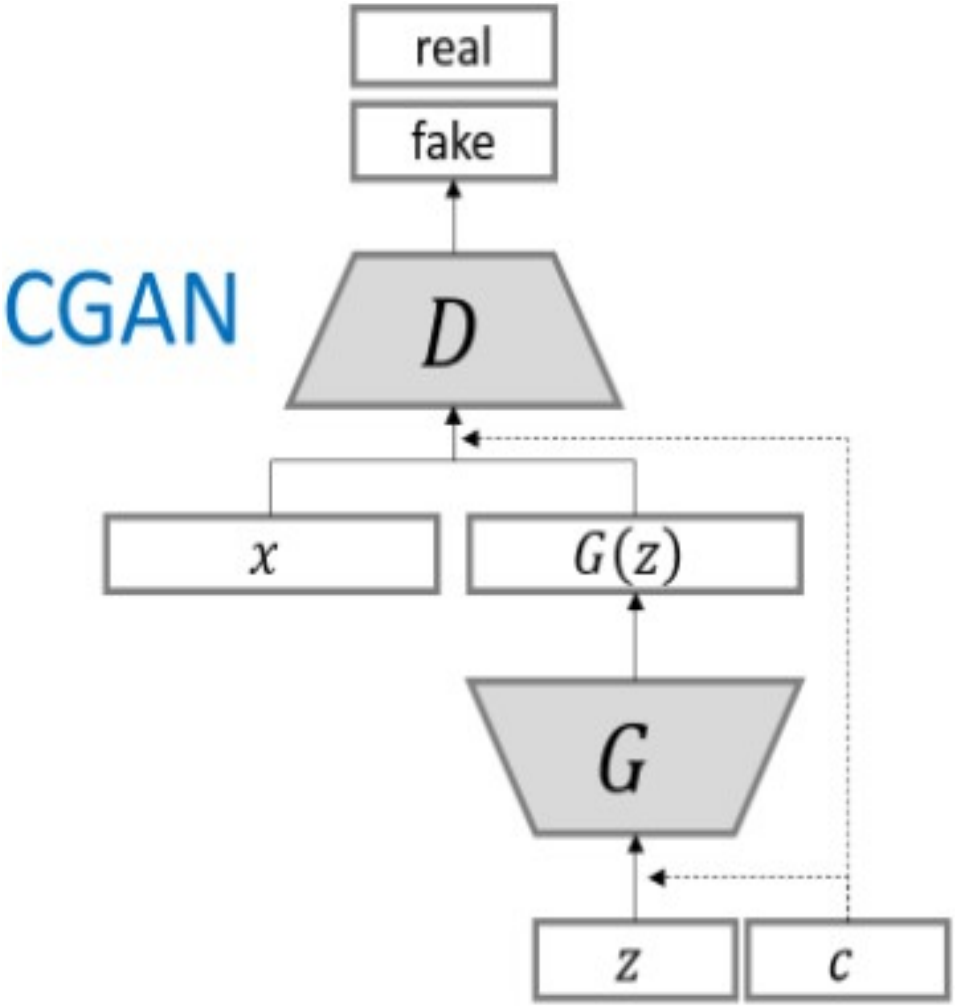
add conditional information mu

make GAN conditional on chemical potential mu :

We train GAN using one esemble with $\mu = 0.91, 0.98, 1.05$ **labeled with corresponding mu for each configuration**

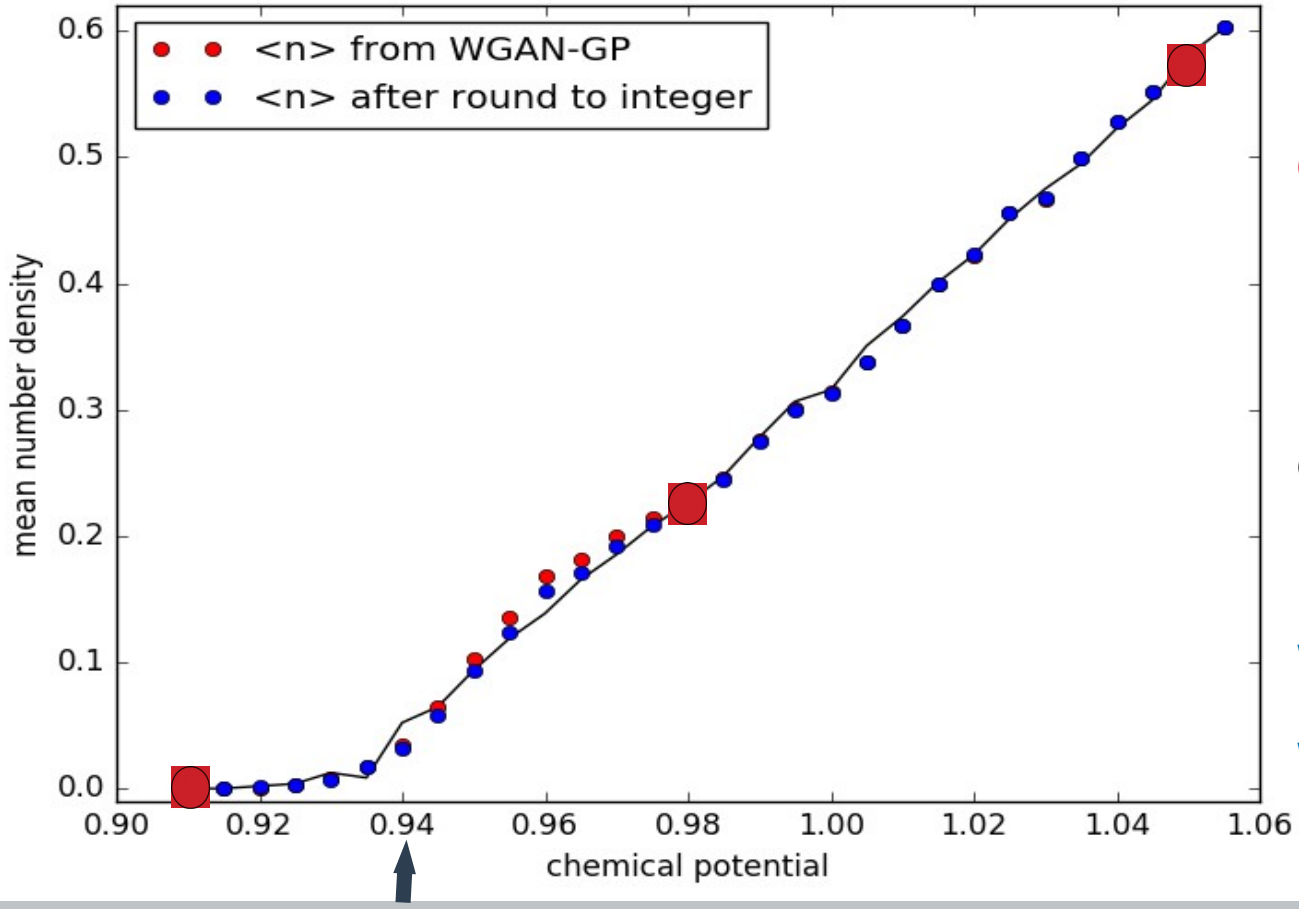
Once trained, in generating stage,

We specify different chemical potentials : generalize to different parameter range!



Conditional GAN

phase diagram generated by c-GAN on mu with limited ensemble of training set:



Got the partition sum:

how much can we reconstruct the partition information from several Monte-Carlo generated ensembles of configs ?

Well phase diagram

Well transition point

Results on lattice 1+1d scalar field

(1) Classification 2: pin down phase transition point

(2) Regression: learn physical observable (non-linear regression)

(3) Generative model : use GAN to generate physical configs.

limited grand canonical –-> canonical ensemble

limited configs. –-> explore phase diagram

Generative model study for XY model

- **Variational Autoregressive Network ---**

  Dian Wu, Lei Wang and Pan Zhang, PRL122,080602(2019)

- Boltzmann distribution for statistical system (e.g. Ising model)

$$p(\mathbf{s}) = \frac{e^{-\beta E(\mathbf{s})}}{Z}$$

- Variational approach (e.g. mean-field-theory) : min. variational Free energy

$$D_{\mathrm{KL}}(q_\theta \,\|\, p) = \sum_{\mathbf{s}} q_\theta(\mathbf{s}) \ln\left(\frac{q_\theta(\mathbf{s})}{p(\mathbf{s})}\right) = \beta(F_q - F)$$

$$F_q = \frac{1}{\beta} \sum_{\mathbf{s}} q_\theta(\mathbf{s}) \left[\beta E(\mathbf{s}) + \ln q_\theta(\mathbf{s})\right]$$

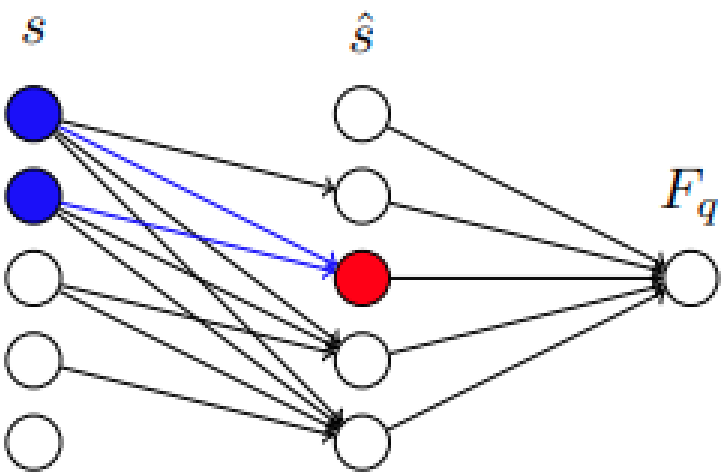
Variational autoregressive network

- Variational Autoregressive Network ---

  Dian Wu, Lei Wang and Pan Zhang, PRL122,080602(2019)

- Introduce more expressive Ansatz with neural network : autoregressive net

$$q_\theta(\mathbf{s}) = \prod_{i=1}^{N} q_\theta(s_i \mid s_1, \ldots, s_{i-1})$$



$$\hat{s}_i = \sigma\left(\sum_{j<i} W_{ij} s_j\right)$$

$$= q(s_i = +1 \mid \mathbf{s}_{<i})$$

$$q(s_i \mid \mathbf{s}_{<i}) = \hat{s}_i \delta_{s_i,+1} + (1 - \hat{s}_i)\delta_{s_i,-1}$$

pixelCNN

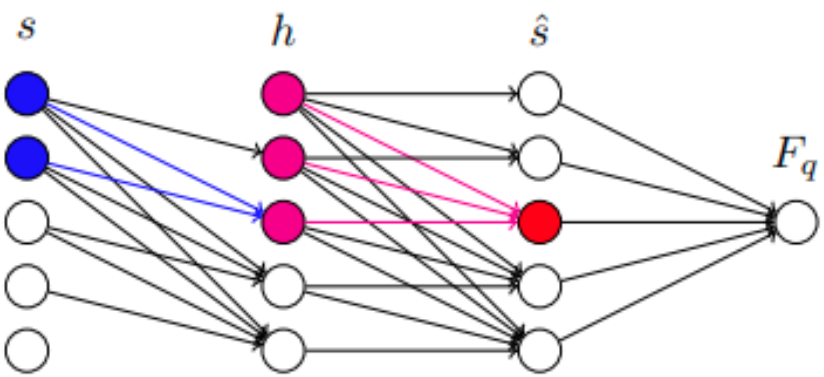
- Variational Autoregressive Network ---

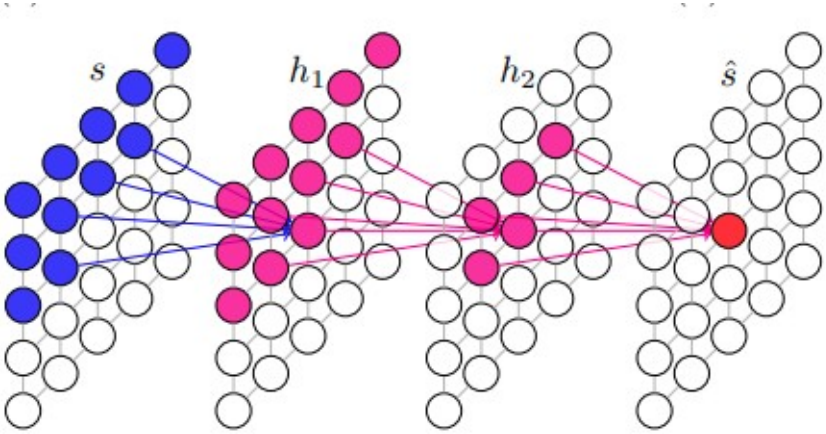
  Dian Wu, Lei Wang and Pan Zhang, PRL122,080602(2019)

- Introduce more expressive Ansatz with neural network : autoregressive net

$$q_\theta(\mathbf{s}) = \prod_{i=1}^{N} q_\theta(s_i \mid s_1, \ldots, s_{i-1})$$

- The network can be made deep, or convolutional, pixelCNN :





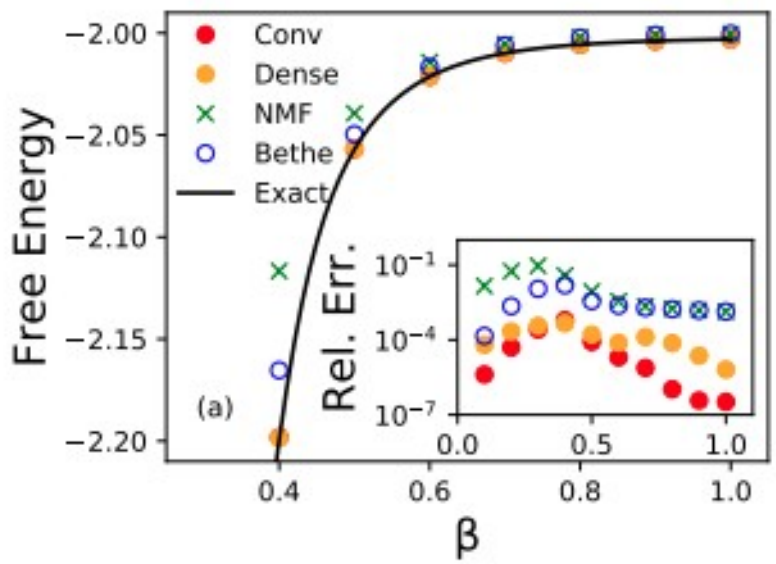
Ising model

- Variational Autoregressive Network ---

Dian Wu, Lei Wang and Pan Zhang, PRL122,080602(2019)

- Gradient descent to minimize the variational free energy

$$\beta \nabla_\theta F_q = \mathbb{E}_{\mathbf{s}\sim q_\theta(\mathbf{s})} \left\{ \left[\beta E(\mathbf{s}) + \ln q_\theta(\mathbf{s}) \right] \nabla_\theta \ln q_\theta(\mathbf{s}) \right\}$$



Continuous-mixture autoregressive net

- Variational Autoregressive Network ---

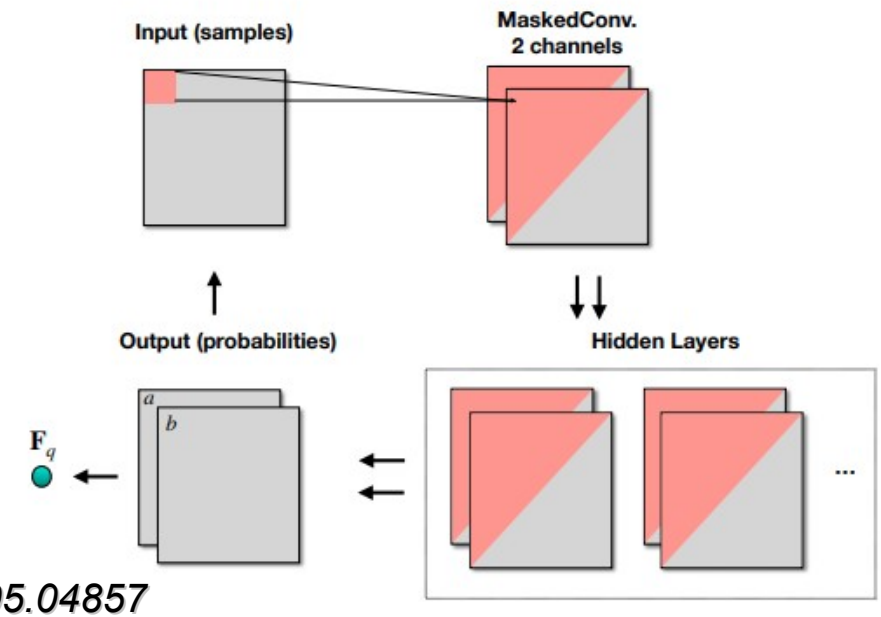
  Dian Wu, Lei Wang and Pan Zhang, PRL122,080602(2019)

- How to generalize to continuous variable systems? Like, for XY model

$$H = -J \sum_{<i,j>} s_i s_j = -J \sum_{<i,j>} \cos(\phi_i - \phi_i)$$

- Continuous-mixture autoregressive network

$$f_\theta(s_i | s_1, ..., s_{i-1}) = \frac{\Gamma(a_i + b_i)}{\Gamma(a_i)\Gamma(b_i)} s_i^{a_i - 1} (1 - s_i)^{b_i - 1}$$

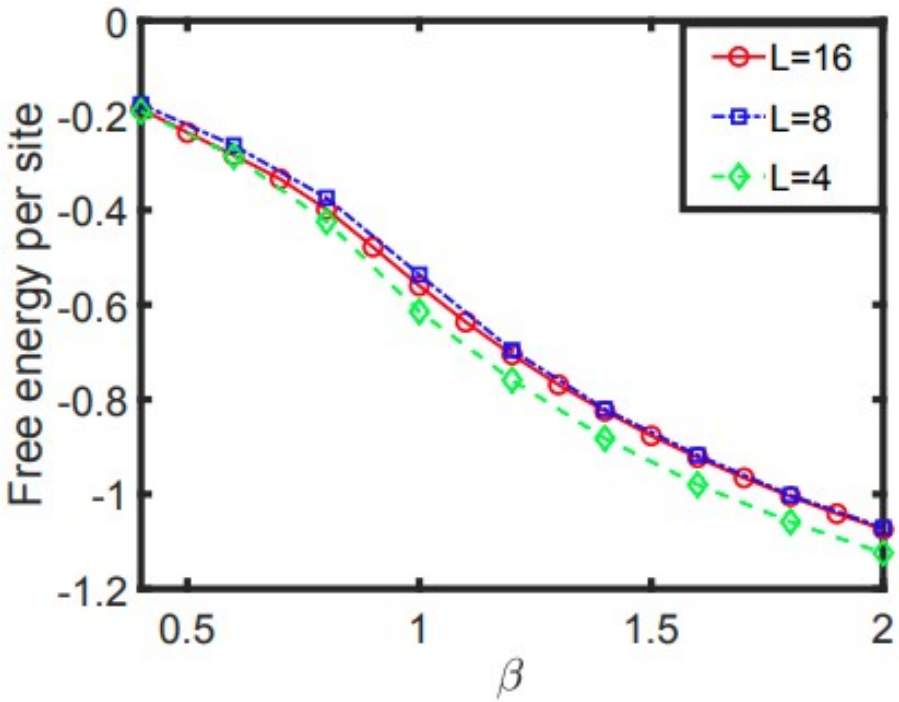


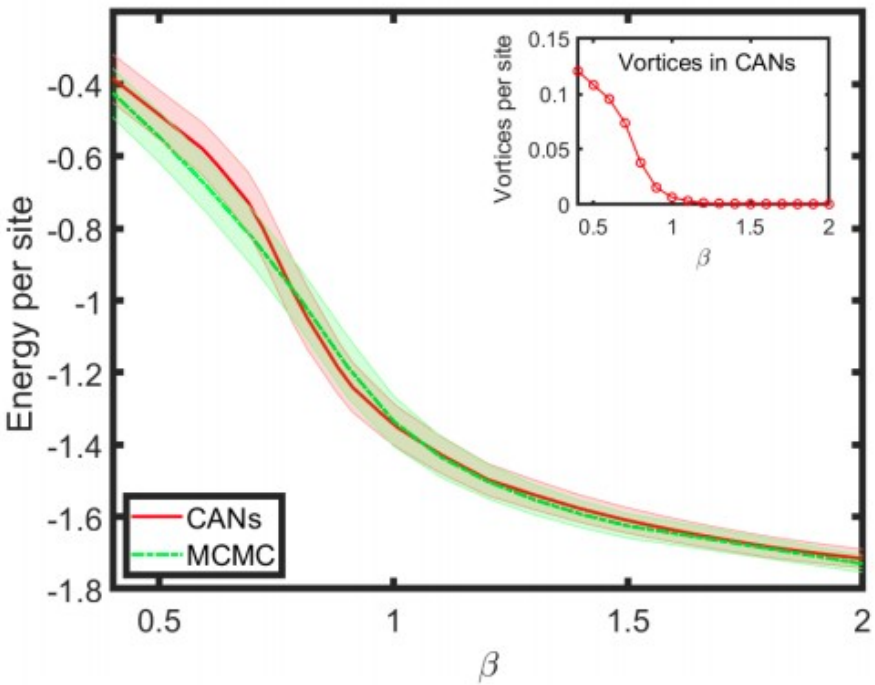
Lingxiao Wang, Yin Jiang, Lianyi He and Kai Zhou, arXiv:2005.04857

CAN for 2D XY model

- Continuous-mixture autoregressive network for XY model

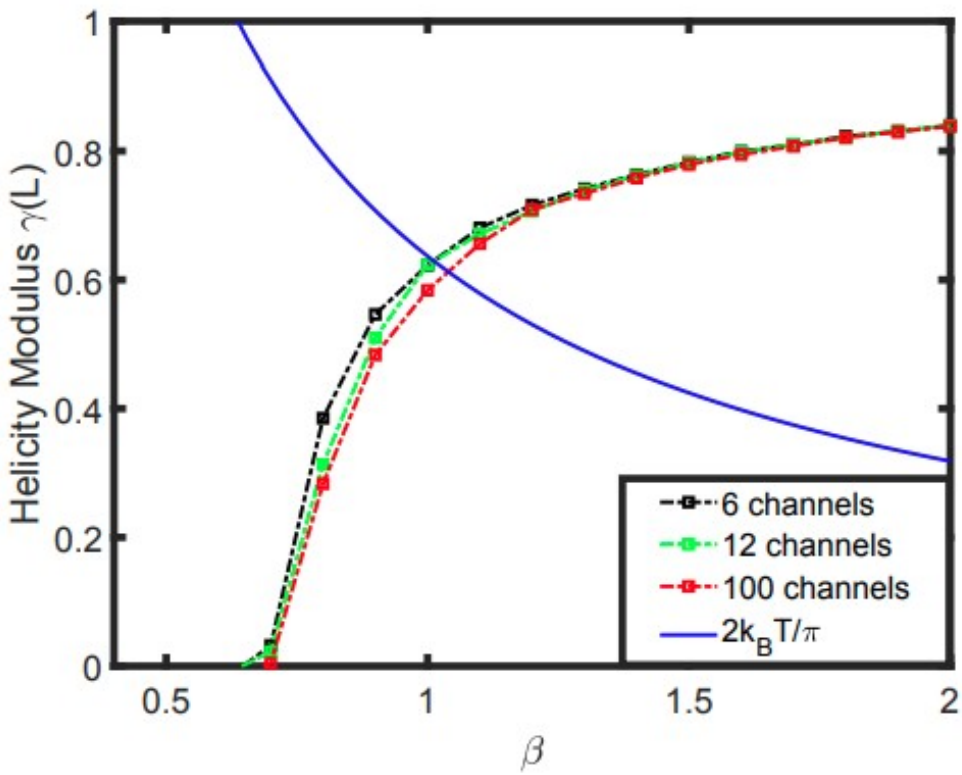
Lingxiao Wang, Yin Jiang, Lianyi He and Kai Zhou, arXiv:2005.04857





KT transition point determination

- To recognize the KT transition point quantitatively, introduce spin stiffness



$$\rho_s = [\partial^2 F(\delta\phi)/\partial(\delta\phi)^2]|_{\delta\phi=0}$$

$$\gamma(L) = -\frac{E}{2L^2} - \frac{J\beta}{L^2}\left\langle \left(\sum_{<i,j>} \sin(\phi_i - \phi_j)\mathbf{e}_{ij}\cdot\mathbf{x}\right)^2 \right\rangle$$

Take-to-home perspectives

- Continuous-mixture autoregressive (CAN) net within variational approach can slove many-body systems with continuous degrees of freedom, like 2D XY model

- Vortices emergy automatically from CAN for 2D XY model with efficient micro states sampling, different from MCMC

- A straightforward determination of the KT transition point from CAN is shown be consistent with standard MCMC eavluation.

Hamiltonian modeling from exp.

- Two major components for predicting thermodynamics of a many-body system :

  1, micro-state distribution (w.r.t. environment params.) – Boltzmann factor

  $$\exp(-A/T)$$

  2, interaction details of the system – Hamiltonian

  $$A = H$$

- Connect the experimental data with proper Hamiltonian models, conventionally :

  1, suitable degrees of freedom

  2, concise model (symbolic beauty, tractability, heuristics)

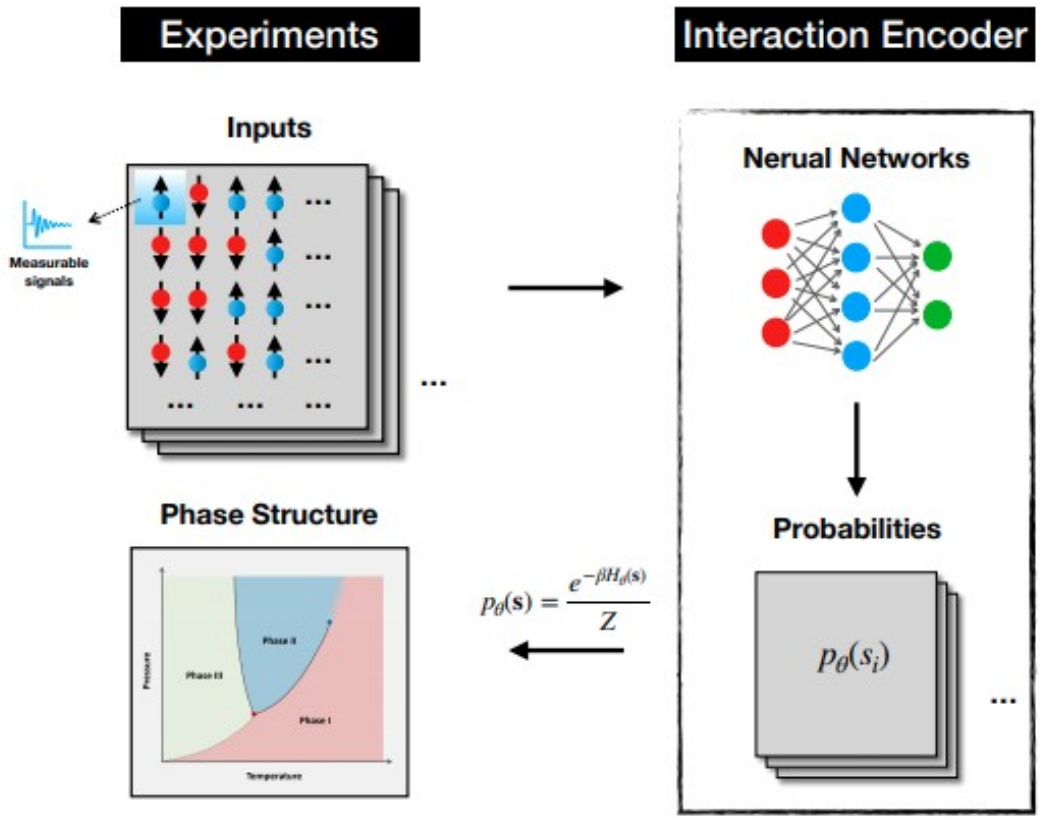
- Can neural network help constructing Hamiltonian (model-independent)?

  Hard, since the growing Hilbert space given finite measurement !

Experiment-to-prediction paradigm

- If one ensemble of ‘micro-states’ could be obtained from experiment measurements :

  measurable signals from TEM/SEM/SPM, or 1st-principle simulations (MCMC)



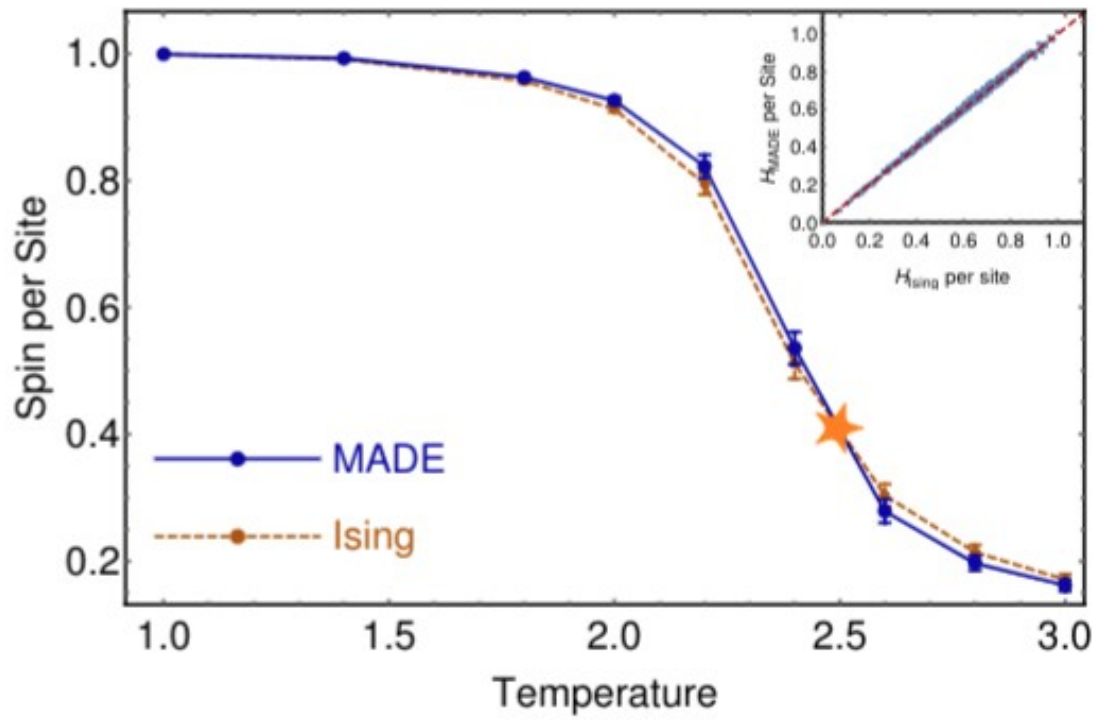
Ising model demonstration

- 2000 Ising configurations (at T=2.5) from MCMC to mock Exp. Data
- Autoregressive network can provide a good probability estimator, thus its Hamiltonian

$$\mathcal{L} = -\sum_{\mathbf{s}\sim q_{data}} \log(p_\theta(\mathbf{s}))$$

$$\mathcal{L} = -\sum_{\mathbf{s}\sim q_{data}} \sum_{d=1}^{N} \log(q_\theta(s_d|\mathbf{s}_{<d}))$$

$$H_\theta(\mathbf{s}) = -T \log p_\theta(\mathbf{s})$$

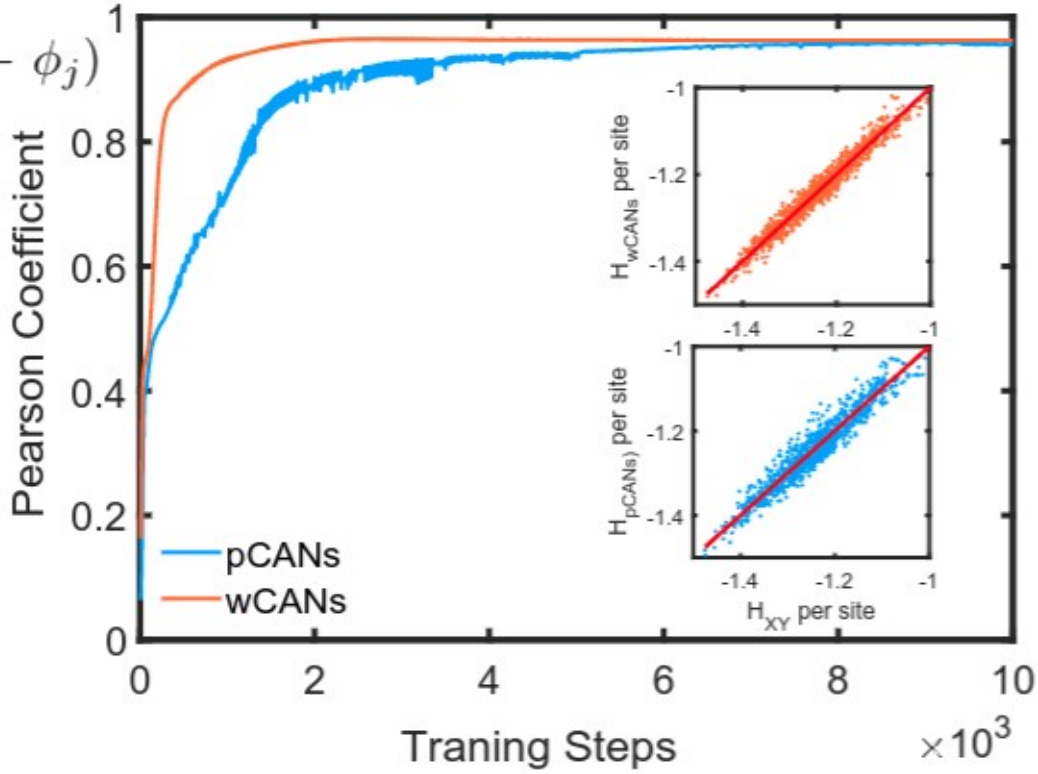


XY model demonstration

- Also works for continuous case with continuous-mixture autoregressive net :

$$H(\mathbf{s}) = -\sum_{<i,j>} s_i s_j = -\sum_{<i,j>} \cos(\phi_i - \phi_j)$$

$$r = \frac{\mathbb{E}\left[\left(H_{CANs} - \mu_{CANs}\right)\left(H_{XY} - \mu_{XY}\right)\right]}{\sigma_{CANs}\sigma_{XY}}$$



Take-to-home perspectives

- A new paradigm to extract microscopic interaction in many-body systems given exp. data
- Key : the autoregressive network can learn the micro-state distribution unsupervisedly
- The unsupervisedly extracted Hamiltonian is suitable to uncover the effective interaction in experiments with measurable degrees of freedom
- The approach can effectively bridge experimental to theoretical research.

Opportunities as physicists

"Computers will not completely replace human, at least for one kind, which is those who can set the objective function. If you are able to take a real-world problem and formulate it into a mathematical form for the objective function, you are going to be a master of the future AI system"

– **Yang Qiang**, HKUST

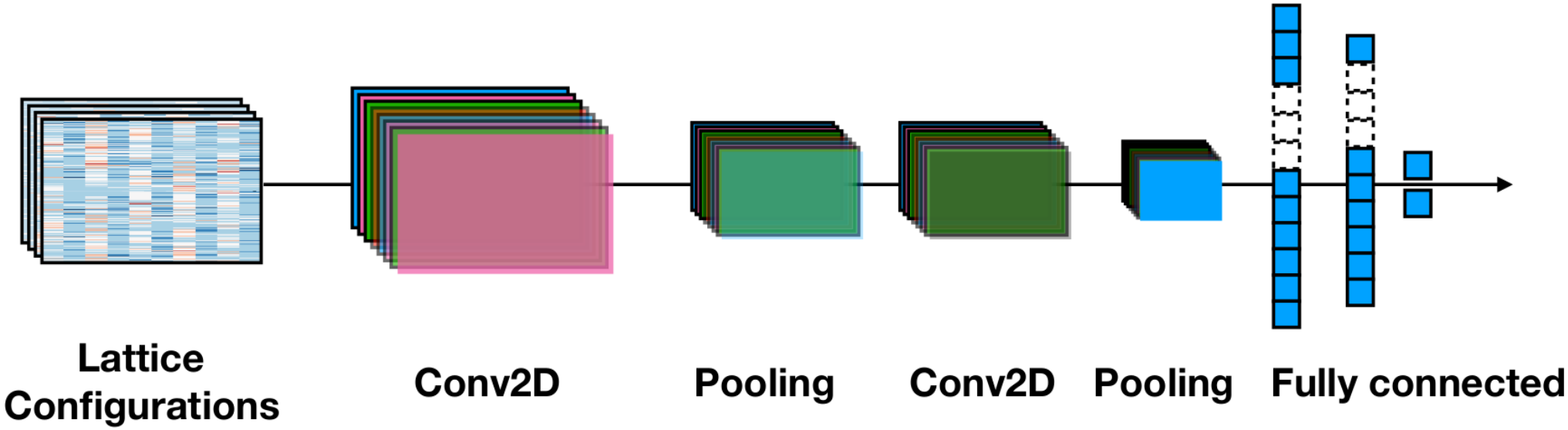
(1) **physics** and related (e.g. chemistry, engineering) problems are much better defined than conventional deep learning ones (e.g. image/natural language processing) – much more economic and efficient in tackling

(2) **deep learning** is a black box – simple physical systems as benchmark

*Renormalization Group *Statistical Physics *collective modes

Thanks!

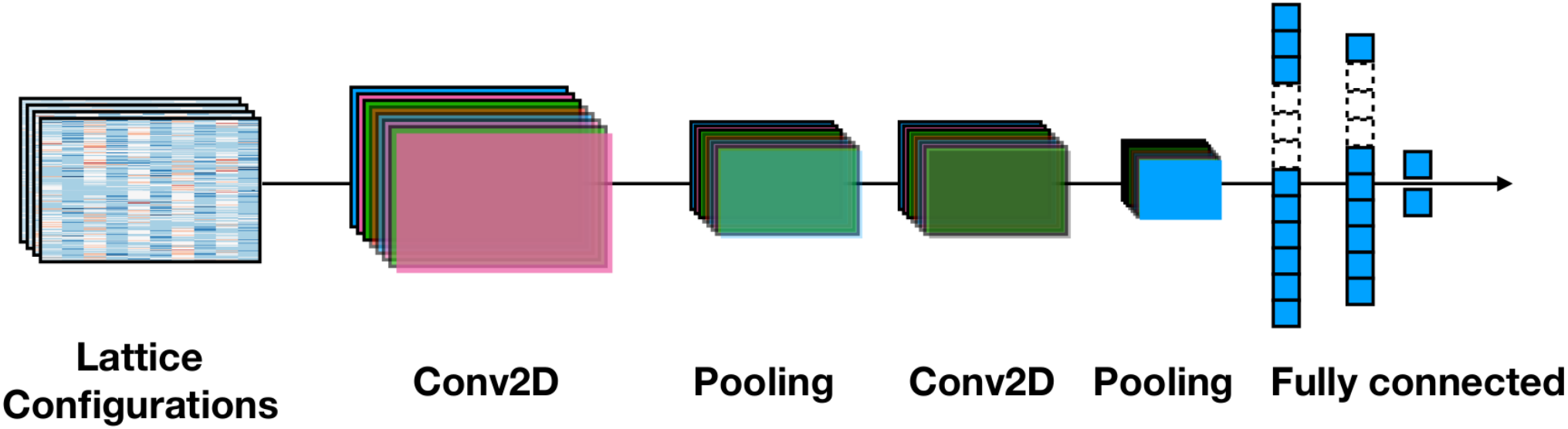
DCNN Architechture – Classification



$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log \hat{y}_i + (1-y_i)\log(1-\hat{y}_i)] + \lambda\|\theta\|_2^2$$

$\alpha_{lr} = 0.0001$ $AdaMax$ **optimization scheme**

DCNN Architechture – Regression



$$\mathcal{L} = -\frac{1}{2N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 + \lambda\|\theta\|_2^2$$

$$\boxed{\text{Training set}: \quad \mu = 0.91 \quad \text{and} \quad \mu = 1.05}$$

GAN – distribution

Zero-sum game – Nash equilibrium

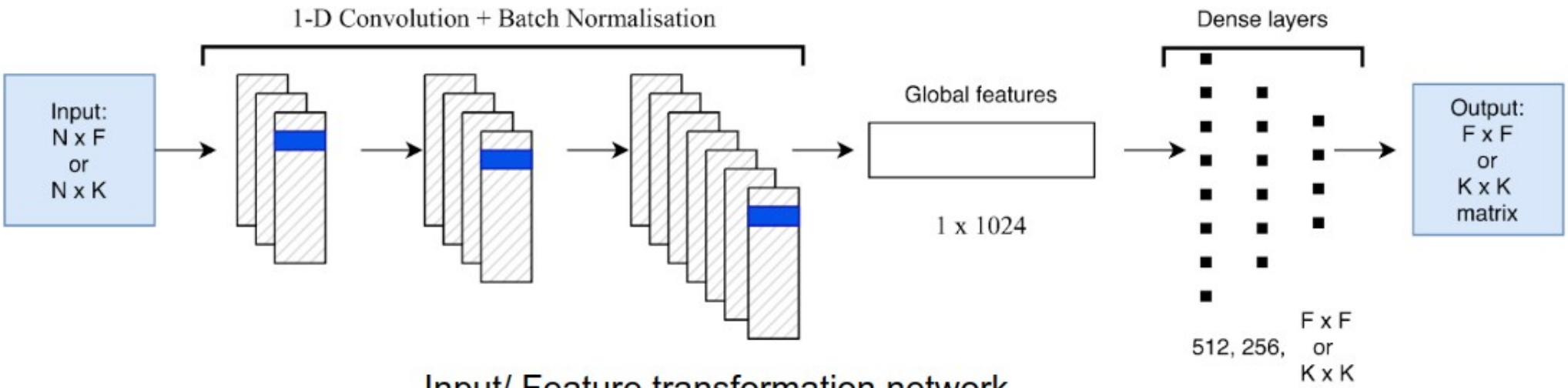
$$G^* = \arg\min_G \max_D (-\mathcal{L}_D(G, D))$$

$$\mathcal{L}_D = -\mathbb{E}_{\hat{x}\sim p_r(\hat{x})}[\log(D(\hat{x}))] - \mathbb{E}_{z\sim p(z)}[\log(1 - D(G(z)))]$$

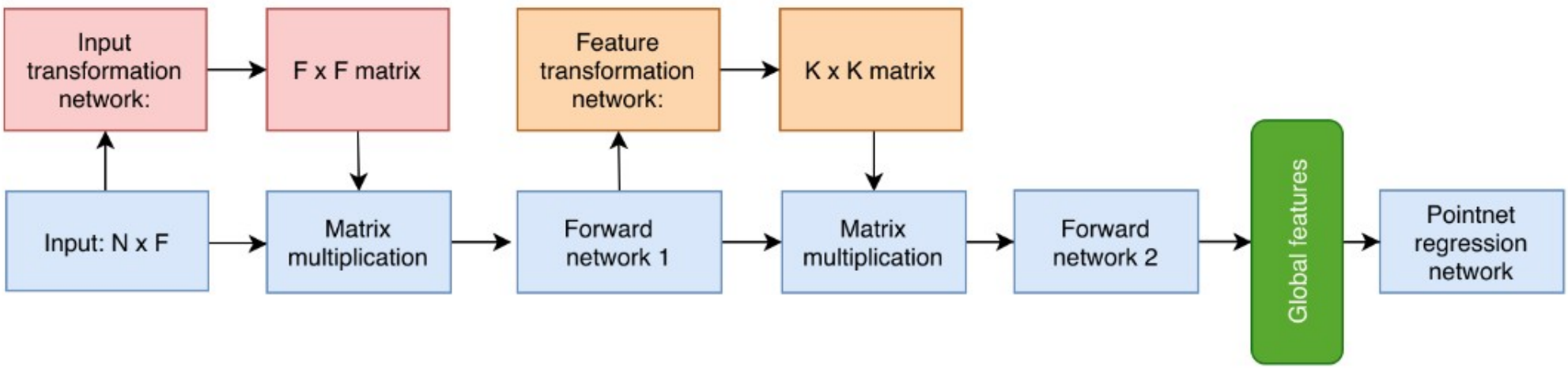
$$\mathcal{L}_G = \mathbb{E}_{z\sim p(z)}[\log(1 - D(G(z)))]$$

$$D^*(\hat{x}) = \frac{p_r(\hat{x})}{p_r(\hat{x}) + p_g(\hat{x})}$$

PointNet

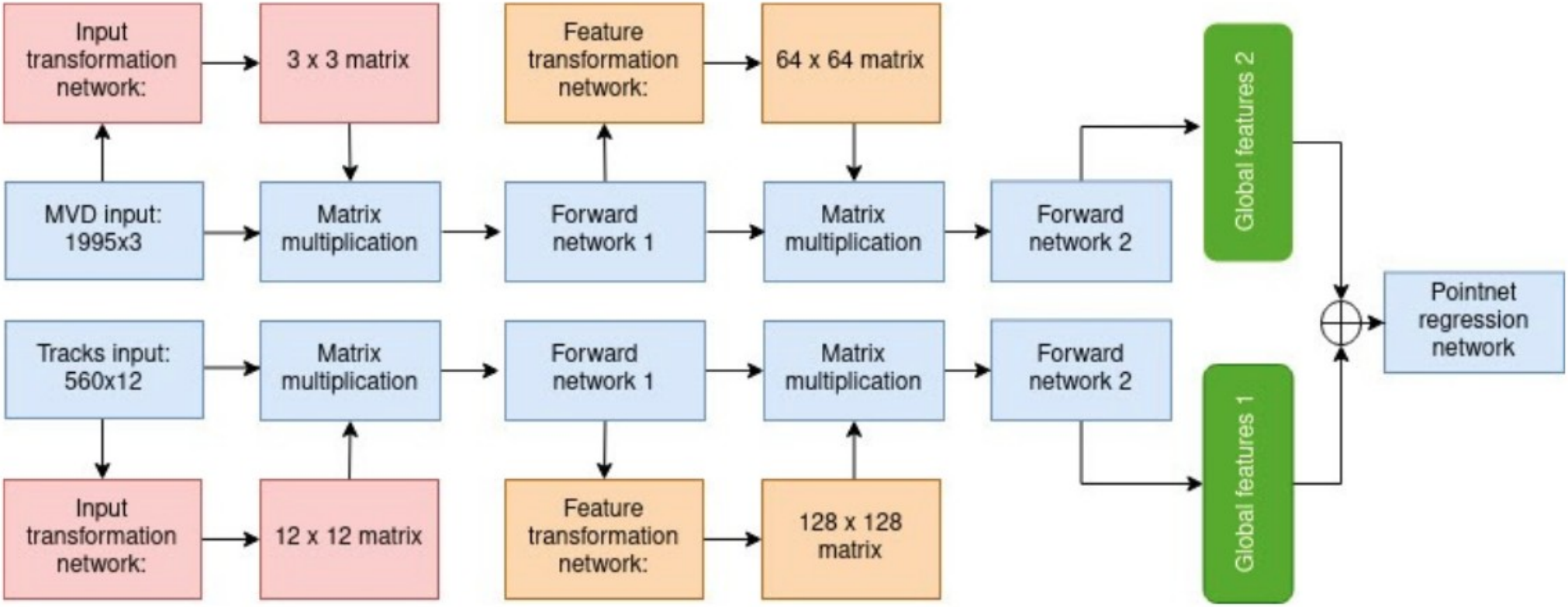


Input/ Feature transformation network



General structure of Mhits, Shits and MStracks models

PointNet



Structure of HT-combi model

PointNet - speed

Model	Speed (events/ s)
M-hits	660
S-hits	159
MS-tracks	1092
HT-combi	435

- Tested on a **Nvidia Geforce RTX 2080 Ti** card with a graphics processing memory of 12 GB
- More room for optimization