# Perspectives of Deep learning techniques in Lattice 1+1d Scalar Field Theory

**Kai Zhou** (FIAS, Frankfurt U., Germany)

**arXiv:1810.12879**

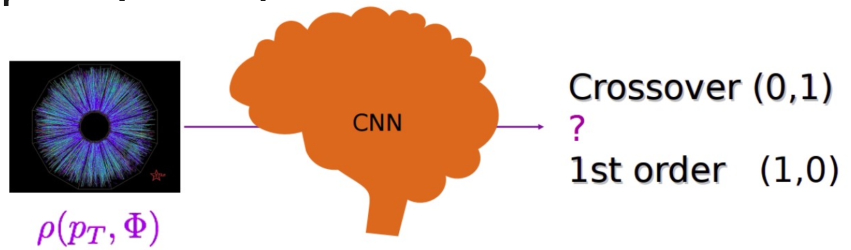
In collaboration with :

**Gergely Endrődi** (ITP, Frankfurt, Germany)
**Long-gang Pang** (UC Berkeley, Berkeley, USA)
**Horst Stoecker** (FIAS, Frankfurt, Germany)

# Deepthinkers group at FIAS

(1) Statistical physics/ lattice configuration analysis

(2) heavy-ion collisions : decode medium property / exp. data



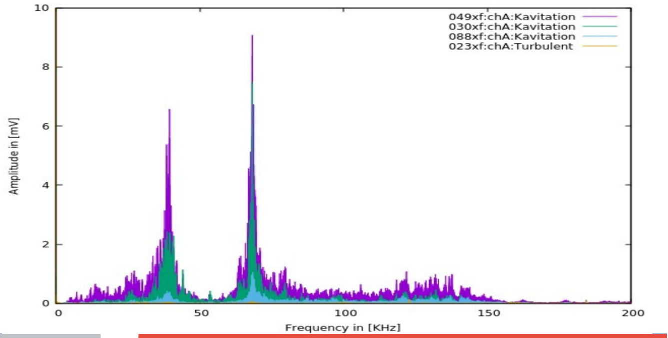

(3) hydrodynamic simulation : speed-up

(4) identify dynamics in Stochastic motion

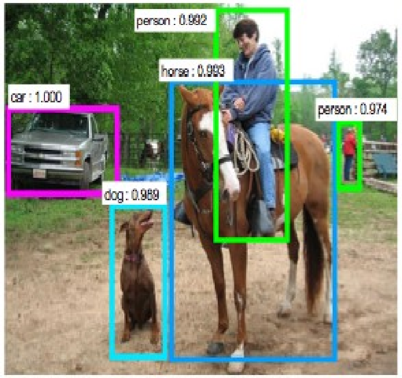
(5) Seismology influence / prediction

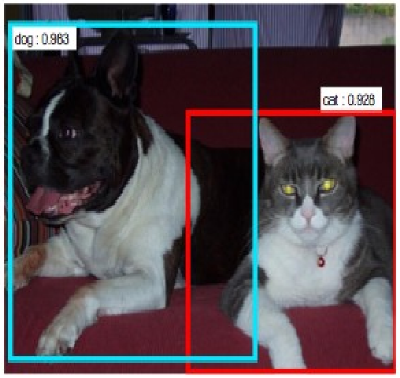
(6) smart-valve: 'hear' the valve flow (leakage? Flow status? Velocity?)

(7) logistic system optimization

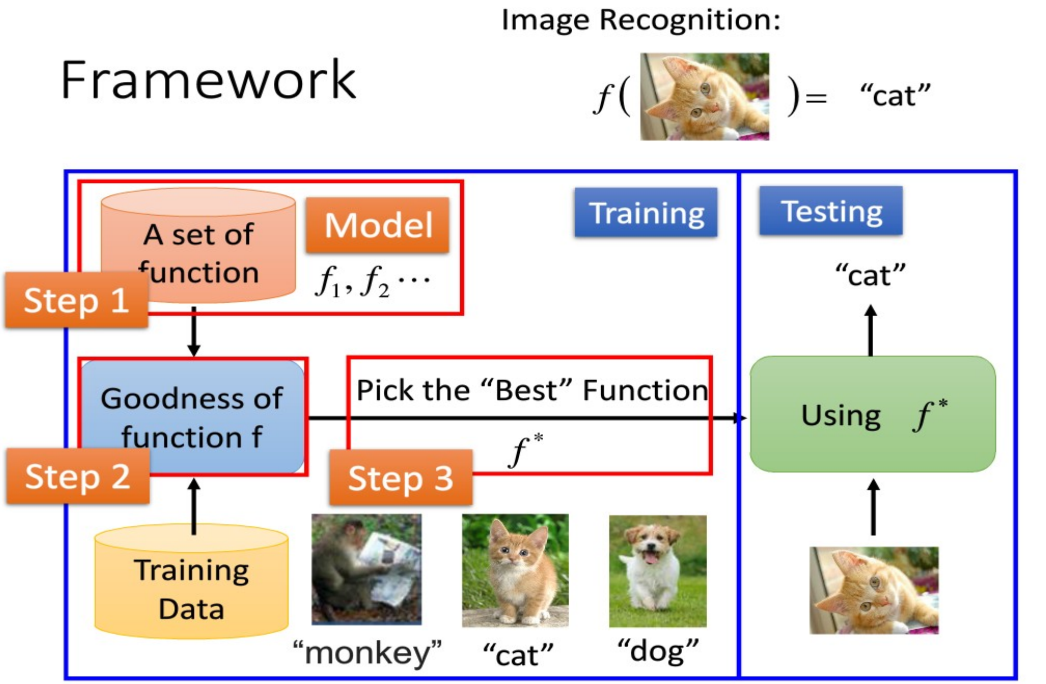


# Introduction


Framework



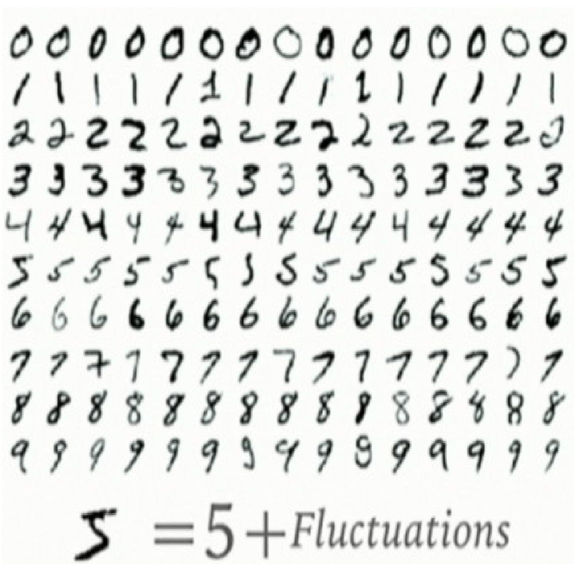

**Find** and **Decode** the mapping/representations into
Deep Neural Network

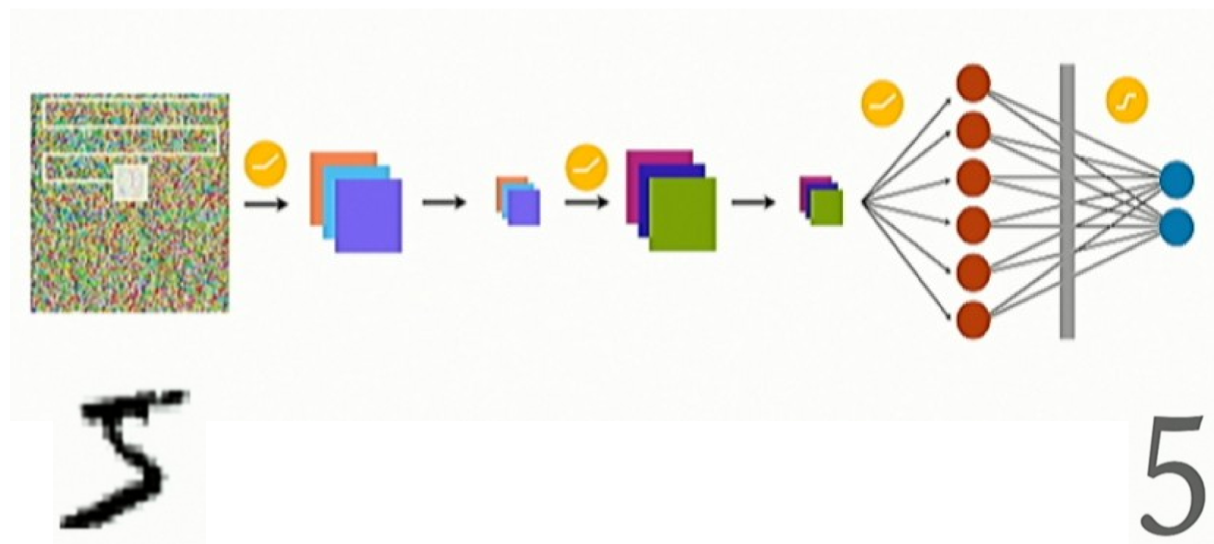
→ **Function approximator**

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

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

**regularization of continuum Action:**

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

**Partition sum :**

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

**Dualization approach :**

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

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

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

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

***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 $\lambda\phi^4$

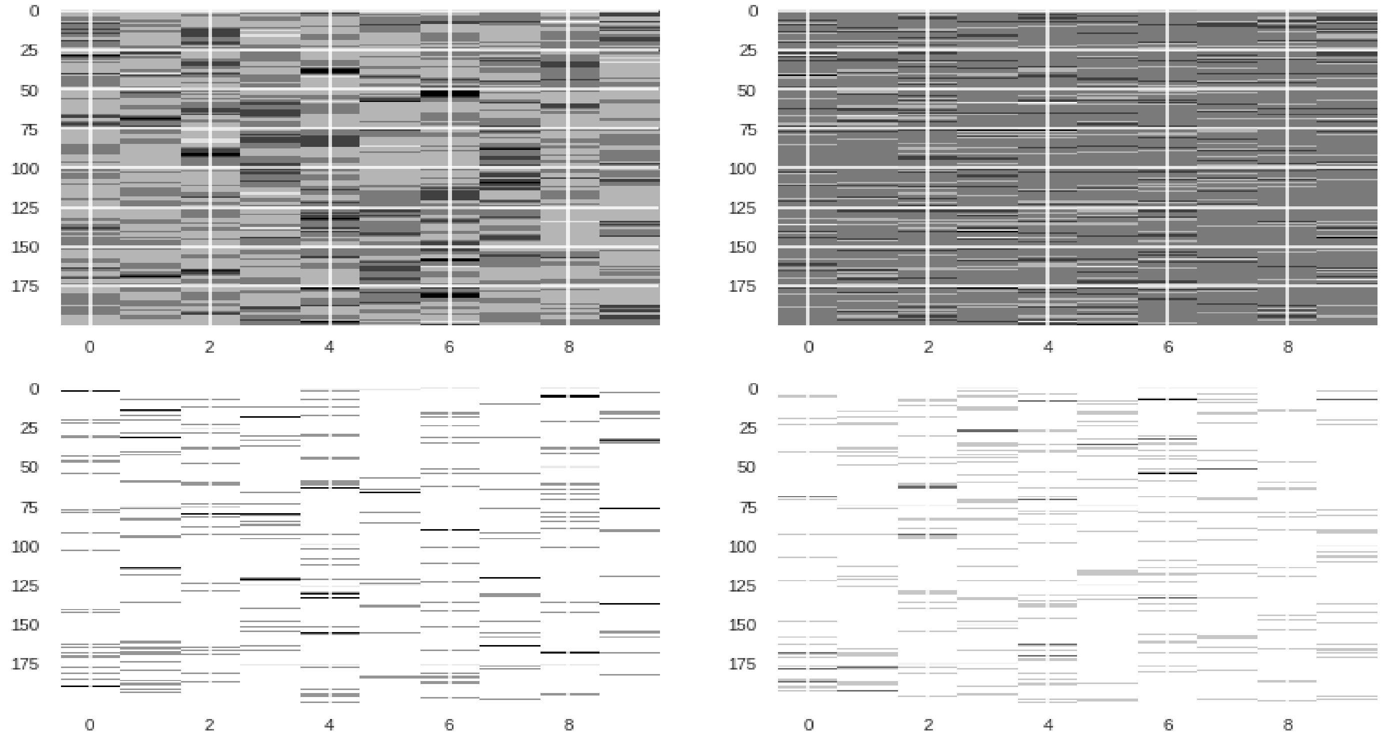
**configurations** – 4 integer-valued variables : $k_t, \quad k_x, \quad l_t, \quad l_x$

$$m = 0.1$$

$$\lambda = 1.0$$

$$N_t = 200$$

$$N_x = 10$$



*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 $|\phi|^2$

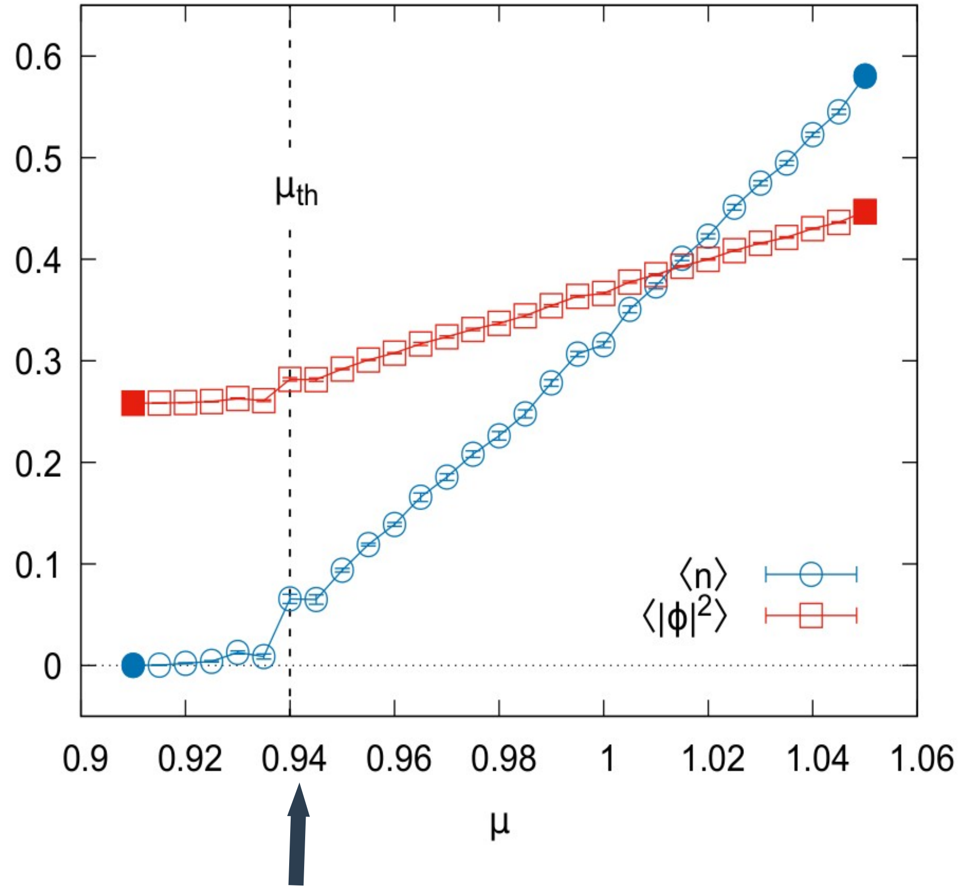
**Grand canonical ensemble**

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

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

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

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




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

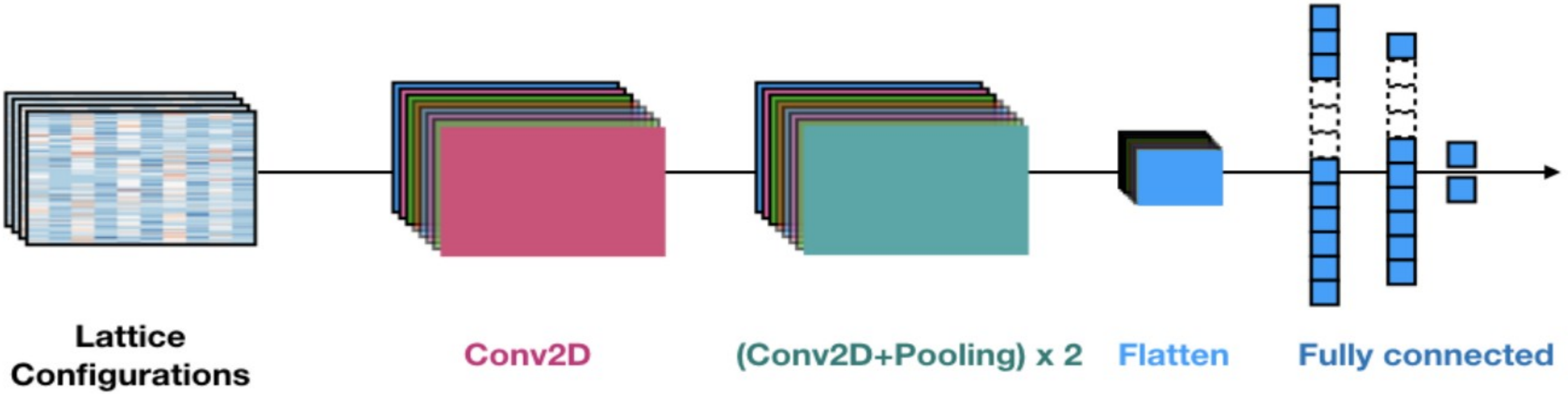
# Exploring NN application here

(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

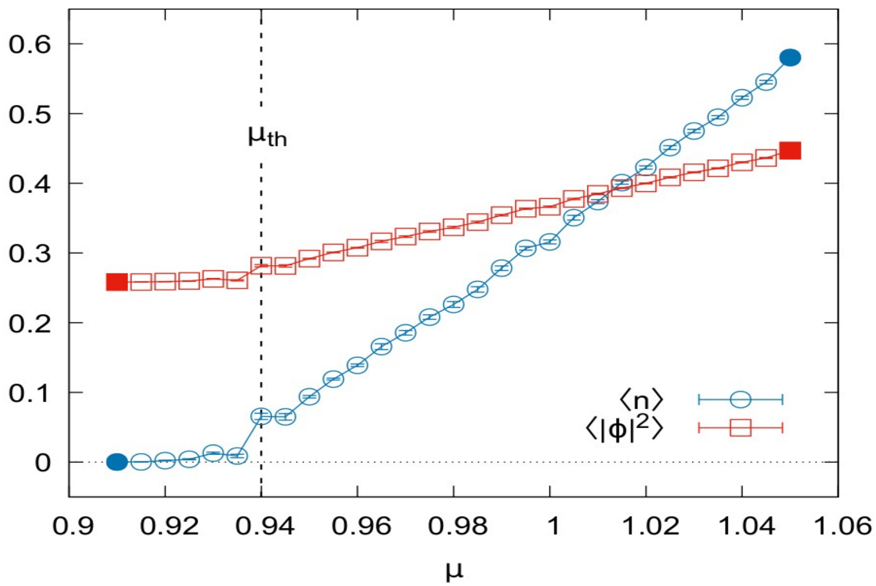



**Training set** consist two ensembles of configs at

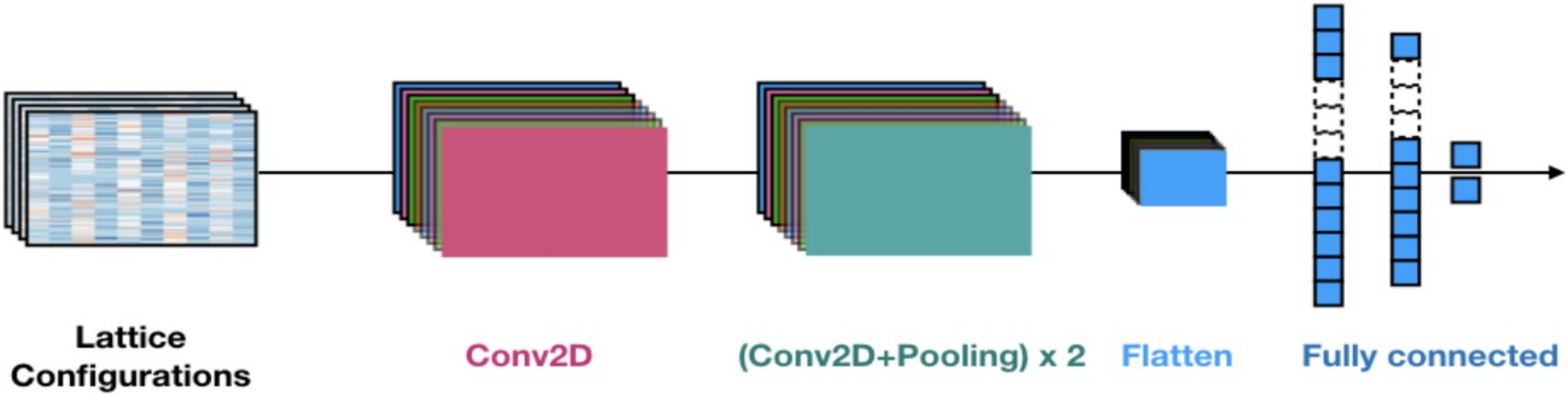
$$\mu = 0.91 \quad \text{with label} \quad y = (0,1)$$

**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

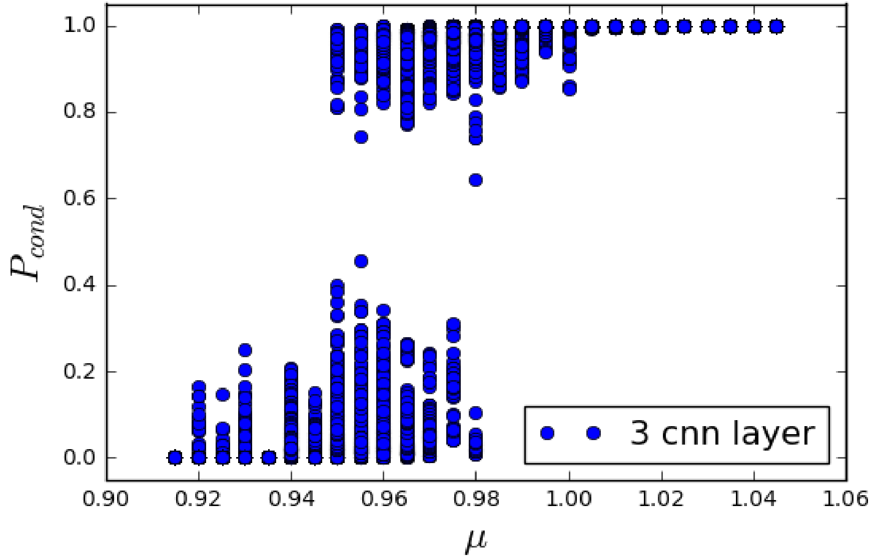
$$\mu = 1.05 \quad \text{with label} \quad y = (1,0)$$

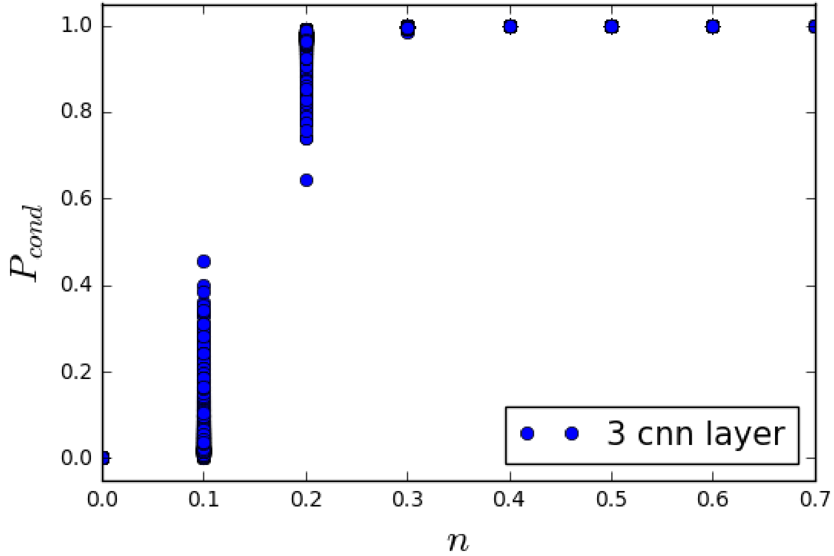
**Testing set** consist of

different ensembles of configurations at different chemical potential

$$0.91 < \mu < 1.05$$

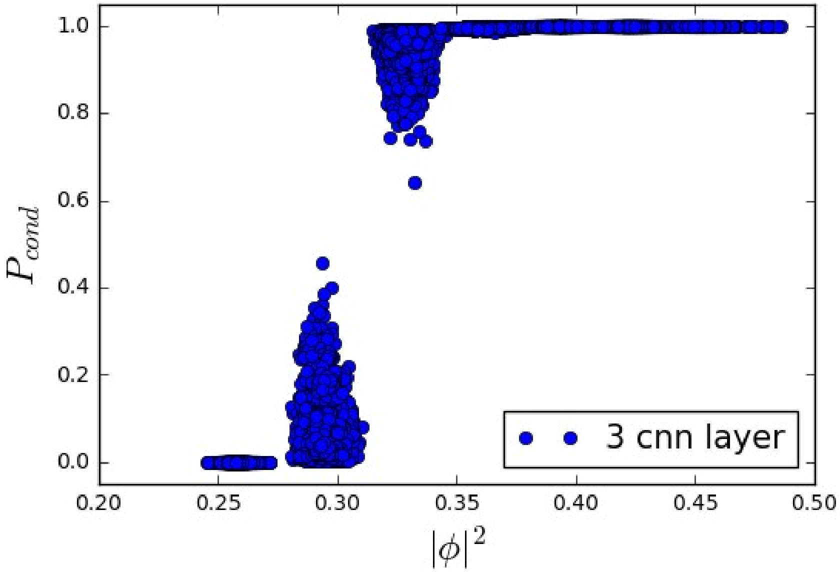
# Condensation probability from DCNN



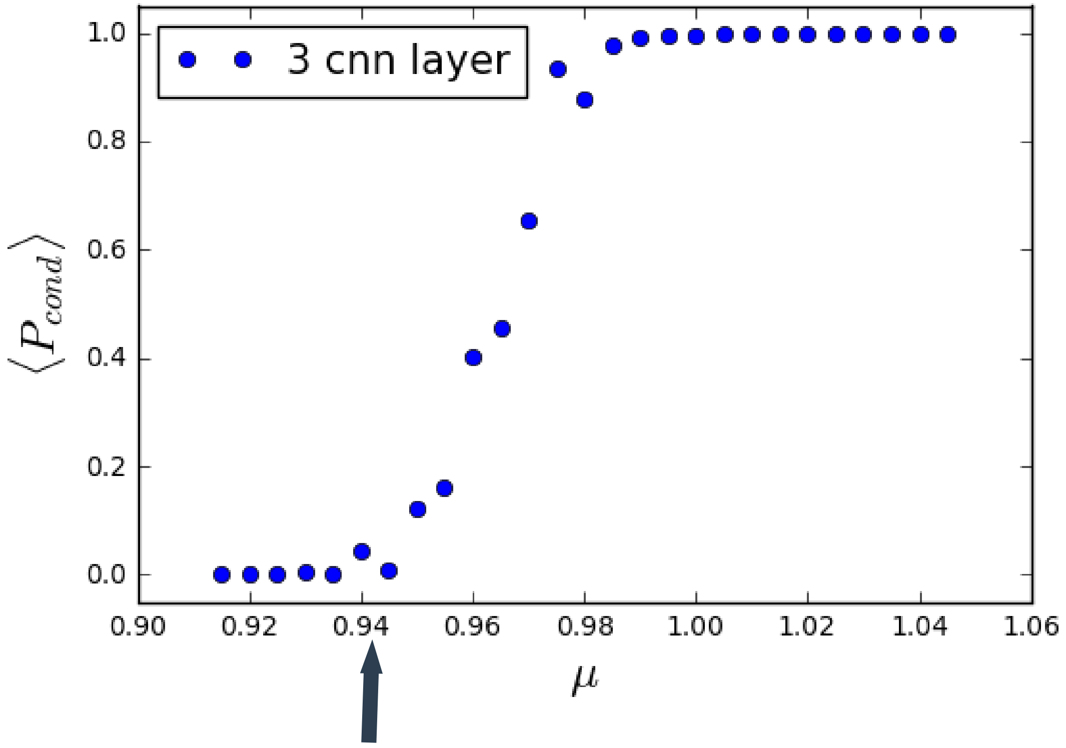

**Strong correlation between**
**P_cond and observables :**

**n / squared field**



# Ensemble average cond-probability

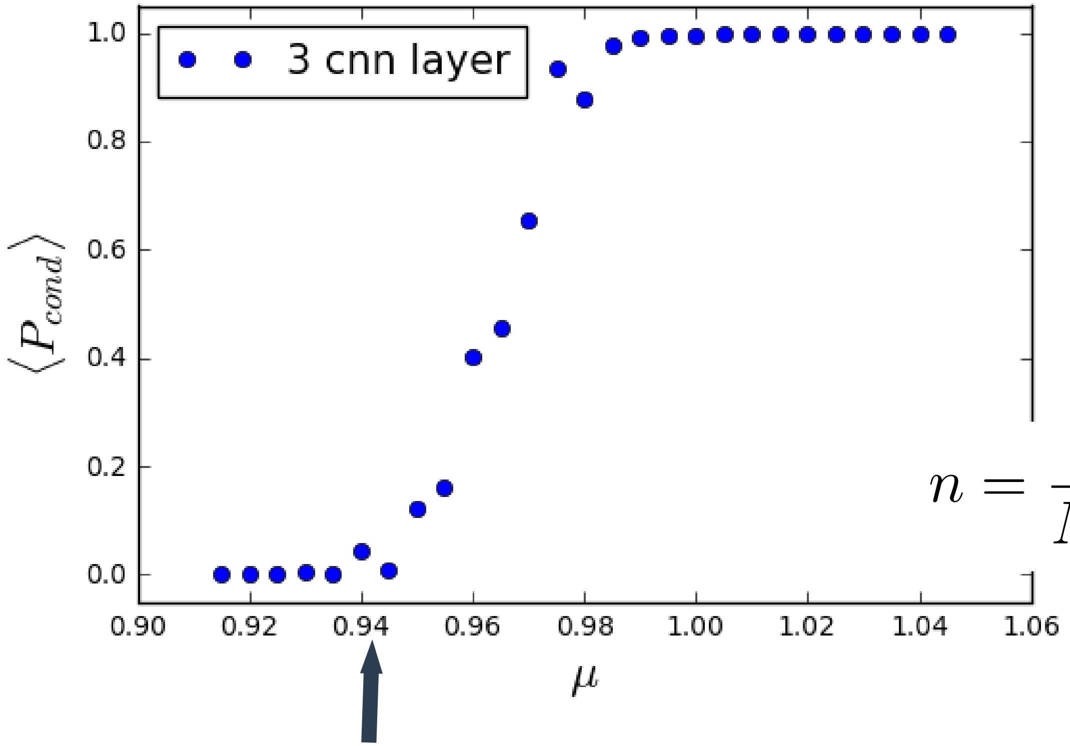
**Classifier of the phases :** $\langle n \rangle = 0$ and $\langle n \rangle \neq 0$




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

# Ensemble average cond-probability

**Classifier of the phases :** $\langle n \rangle = 0$ and $\langle n \rangle \neq 0$

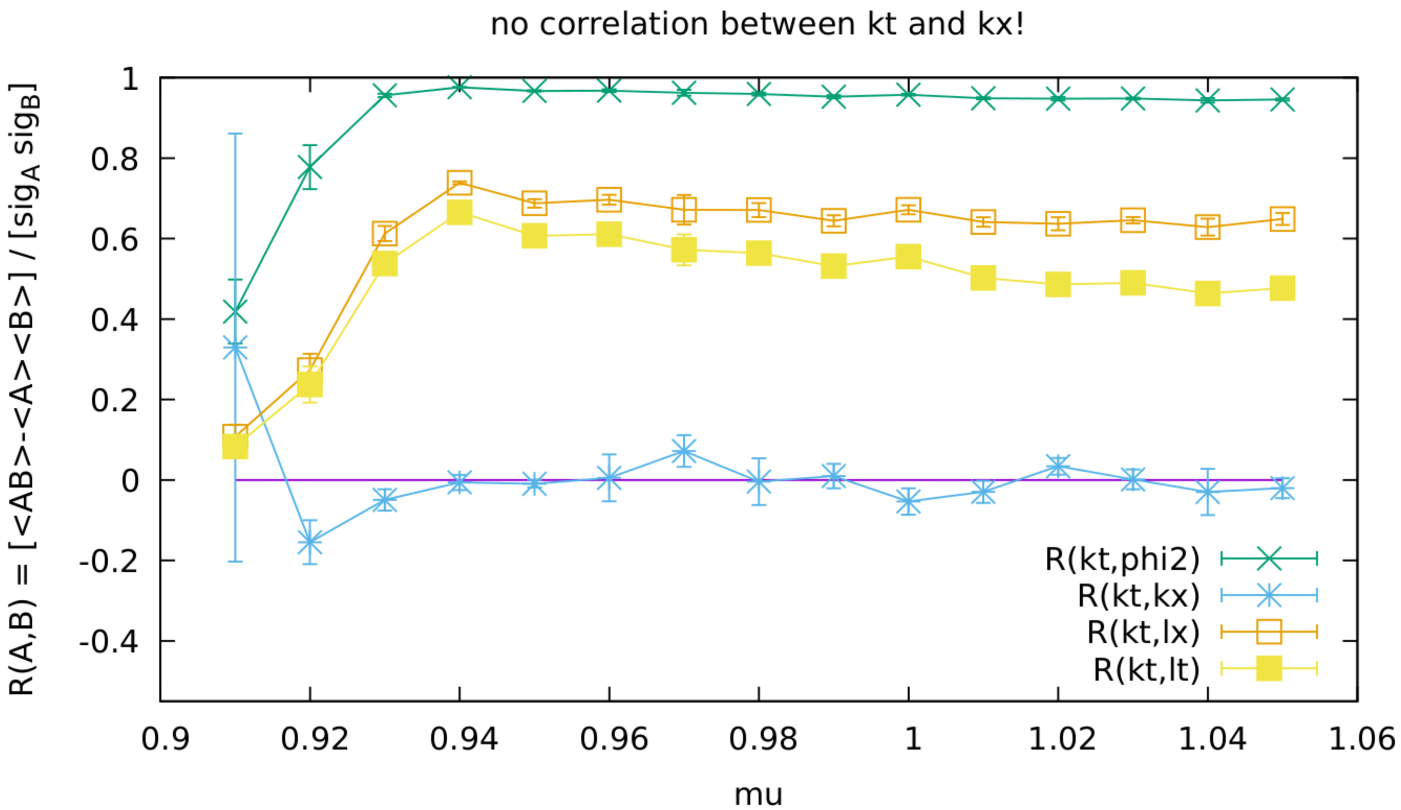



$$n = \frac{1}{N_x N_t a} \sum_n k_t(n)$$

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

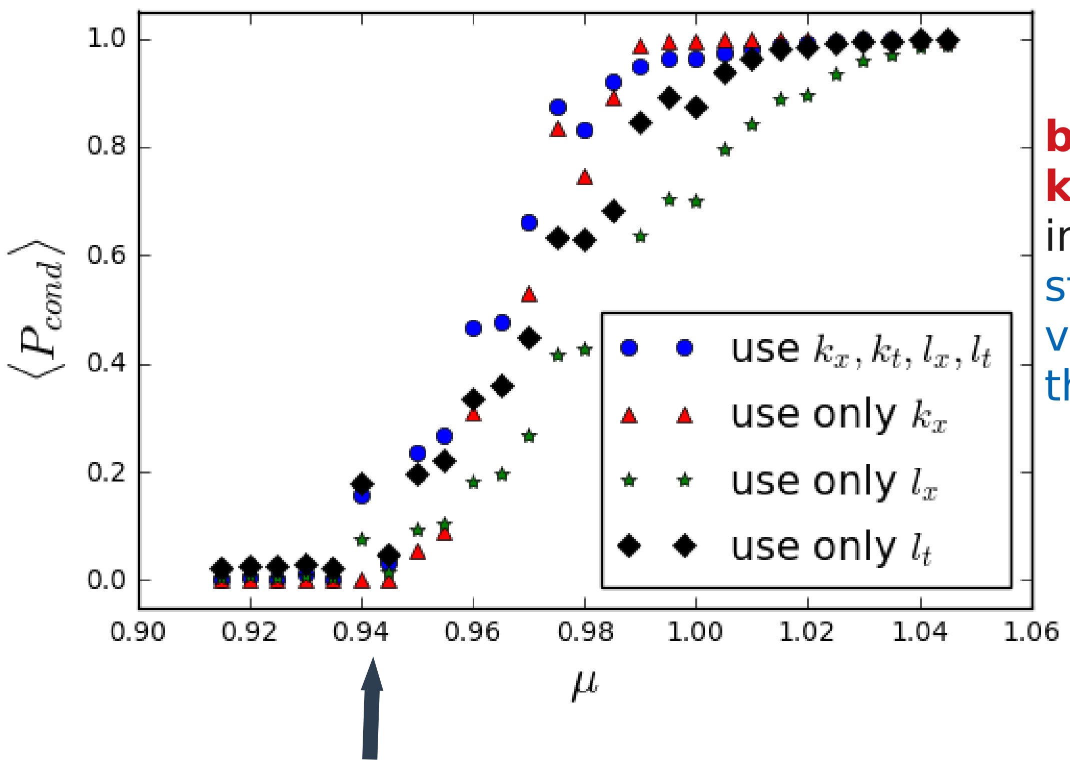
# Discard kt information

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



# Try different field component variables

**The same transition point, even use only kx!**

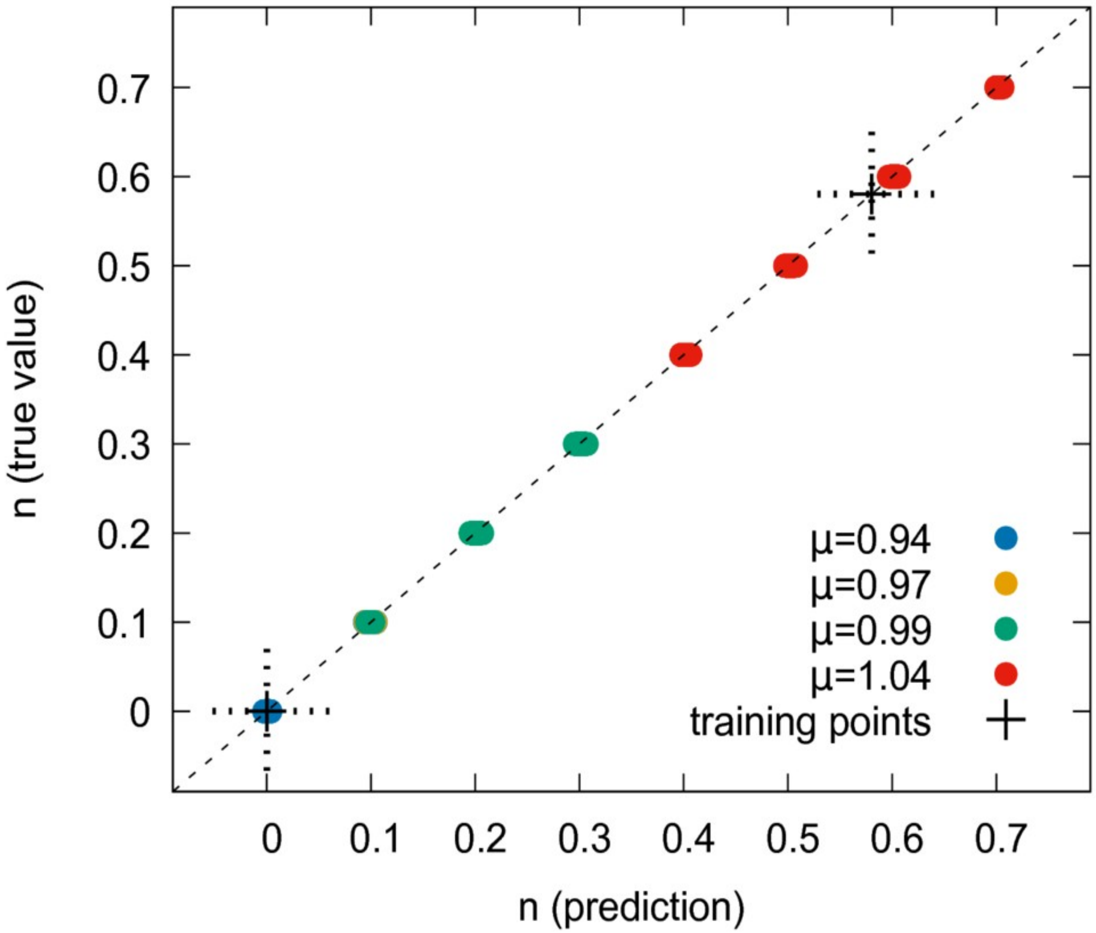



**beyond conventional knowledge :** indicating hidden structures in the kx variables and not only in the kt variables.

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

# regression for particle density $n$

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

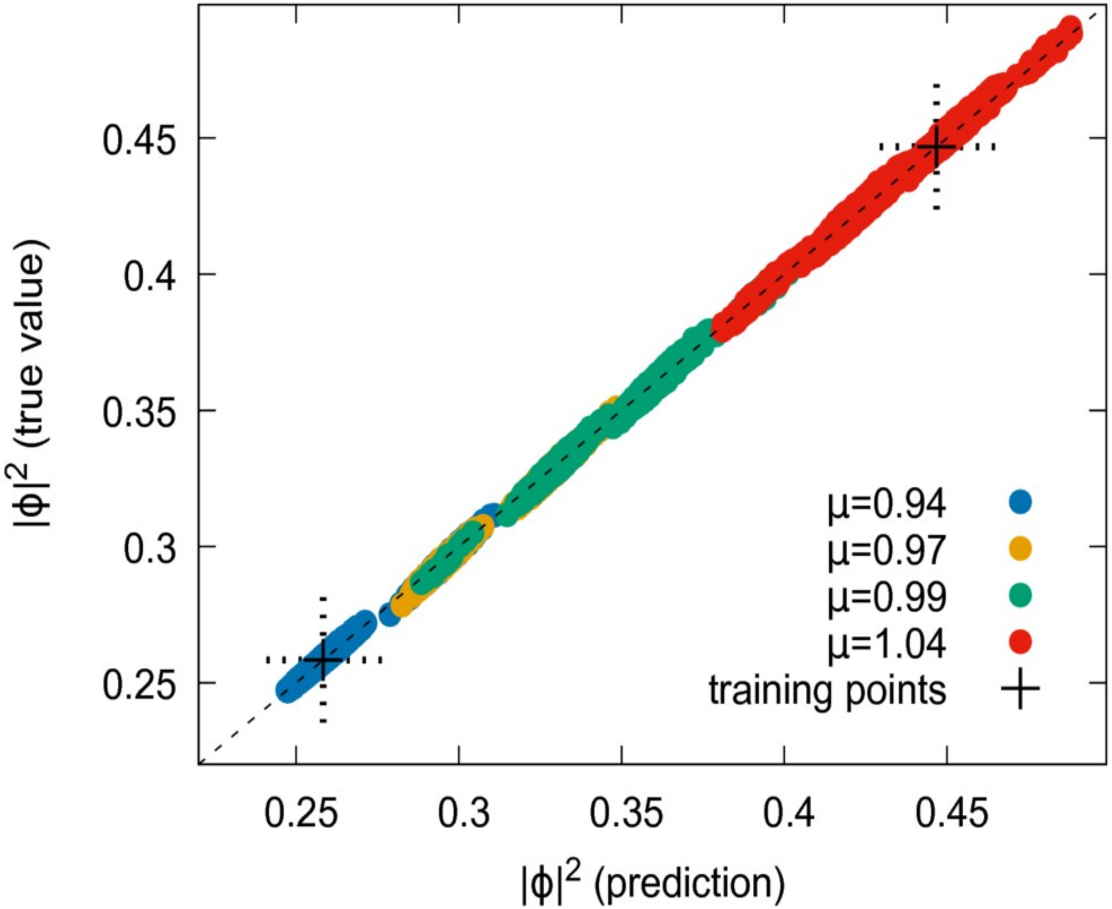



$$n = \frac{1}{N_x N_t a} \sum_n k_t(n)$$

$$RMSE < 0.003$$

# regression for squared field $\phi^2$

**Note, for training, only used** $\mu = 0.91$ and $\mu = 1.05$



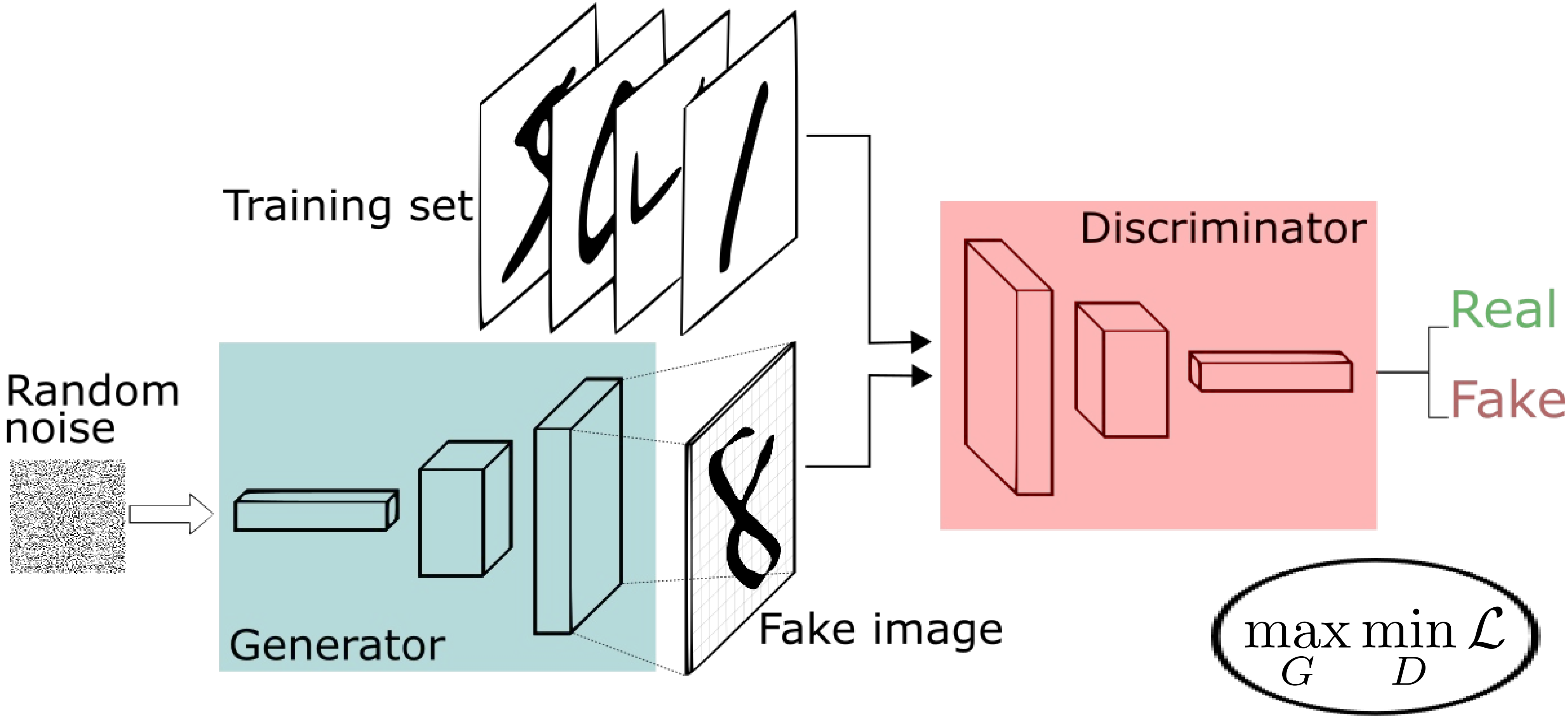

$$|\phi|^2 = \frac{1}{N_x N_t} \sum_n \frac{W[s(n)+2]}{W[s(n)]}$$

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

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

$$RMSE < 0.005$$

# Gnerative Adversarial Network



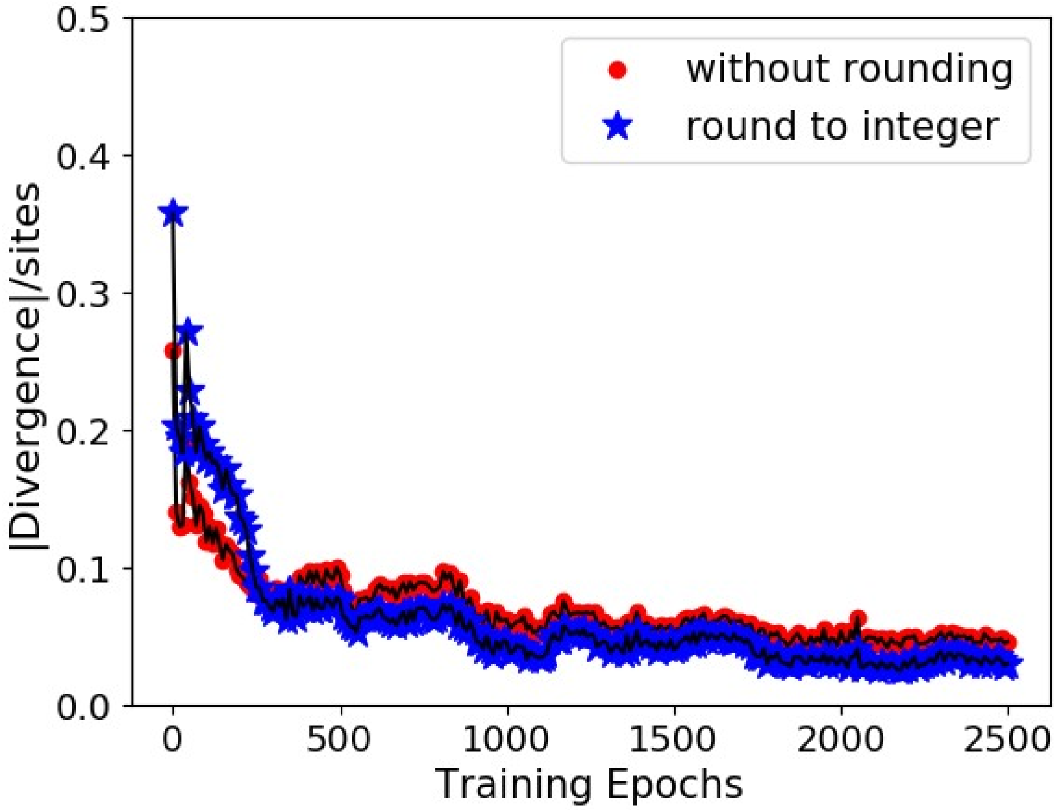

$$\mathcal{L} = -\mathbb{E}_{\hat{x}\sim p_r(\hat{x})}[\log(D(\hat{x}))] - \mathbb{E}_{z\sim p(z)}[\log(1 - D(G(z)))]$$

# GAN – generate proper configurations

**The divergence condition automatically get learned :**

**'Physical' configs can be generated**

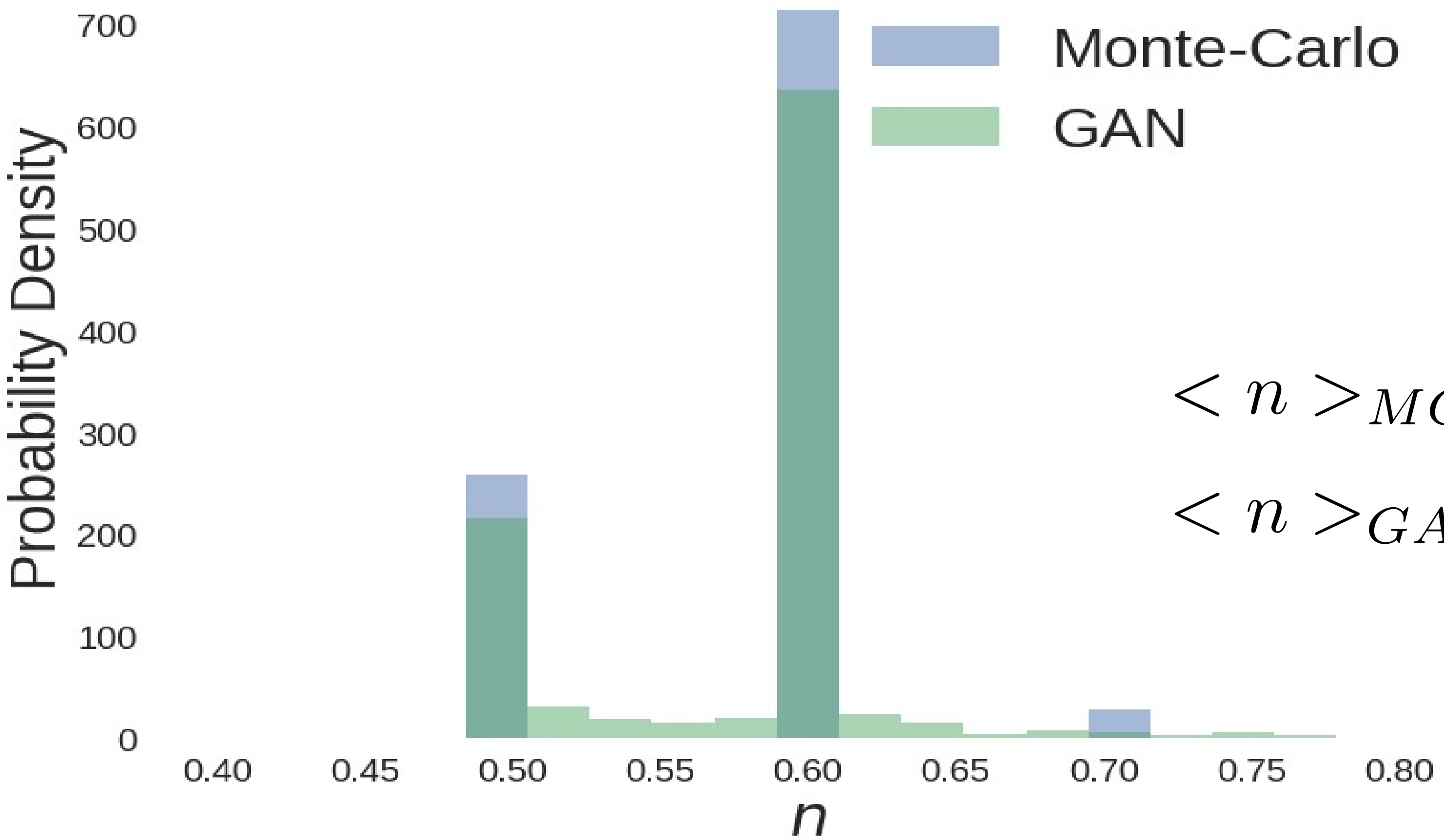
$$\nabla \cdot k(n) = \sum_{\nu} [k_\nu(n) - k_\nu(n - \hat{\nu})] = 0$$




**Automatically capture the implicit physical constraint!**

# Fine tune GAN

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

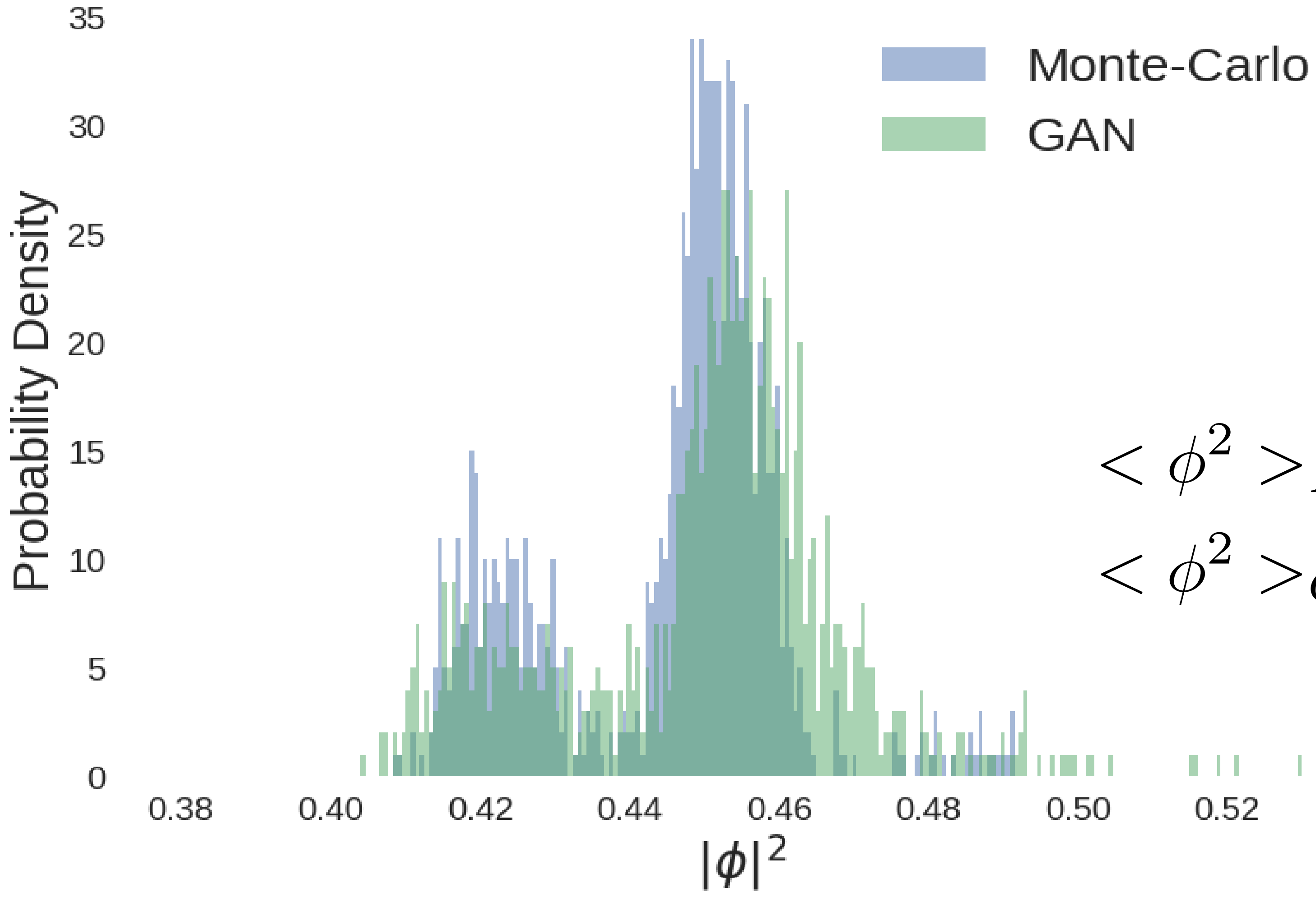



$$< n >_{MC} = 0.580$$

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

# Fine tune GAN

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




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

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

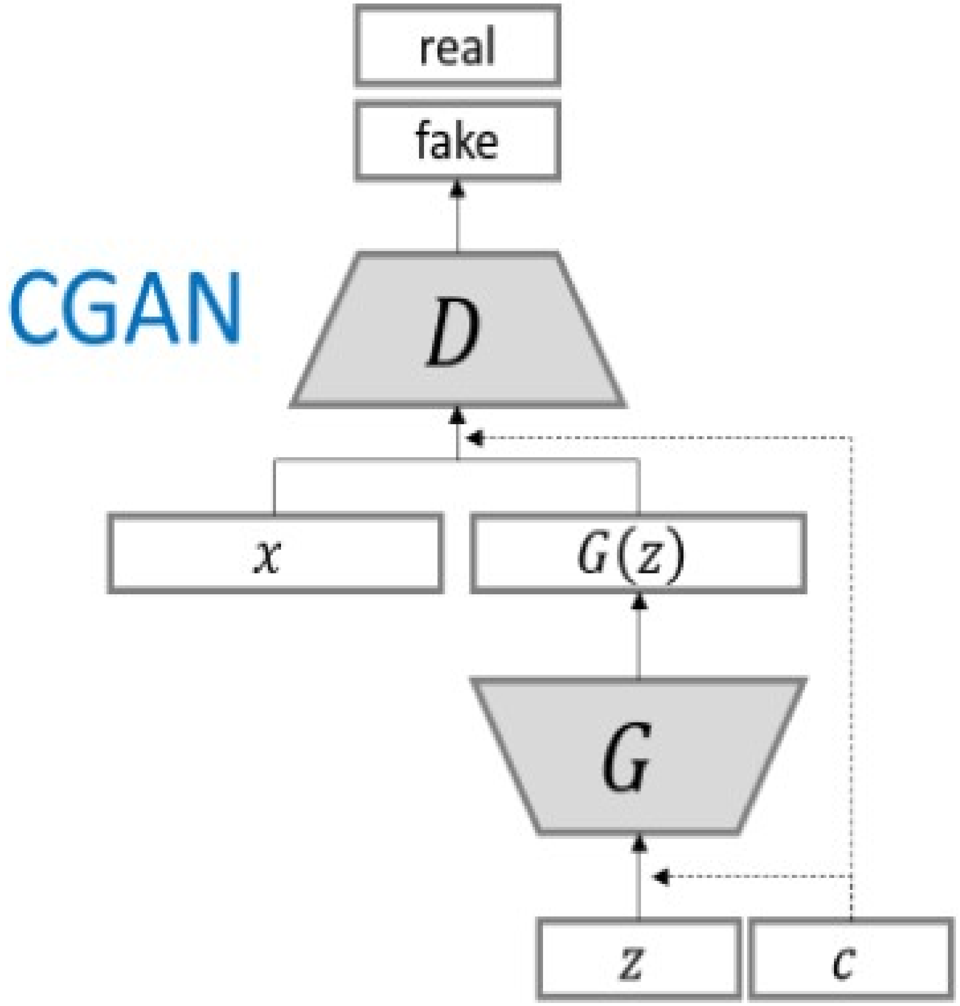
# add conditional information n

**make GAN conditional on particle density n :**

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

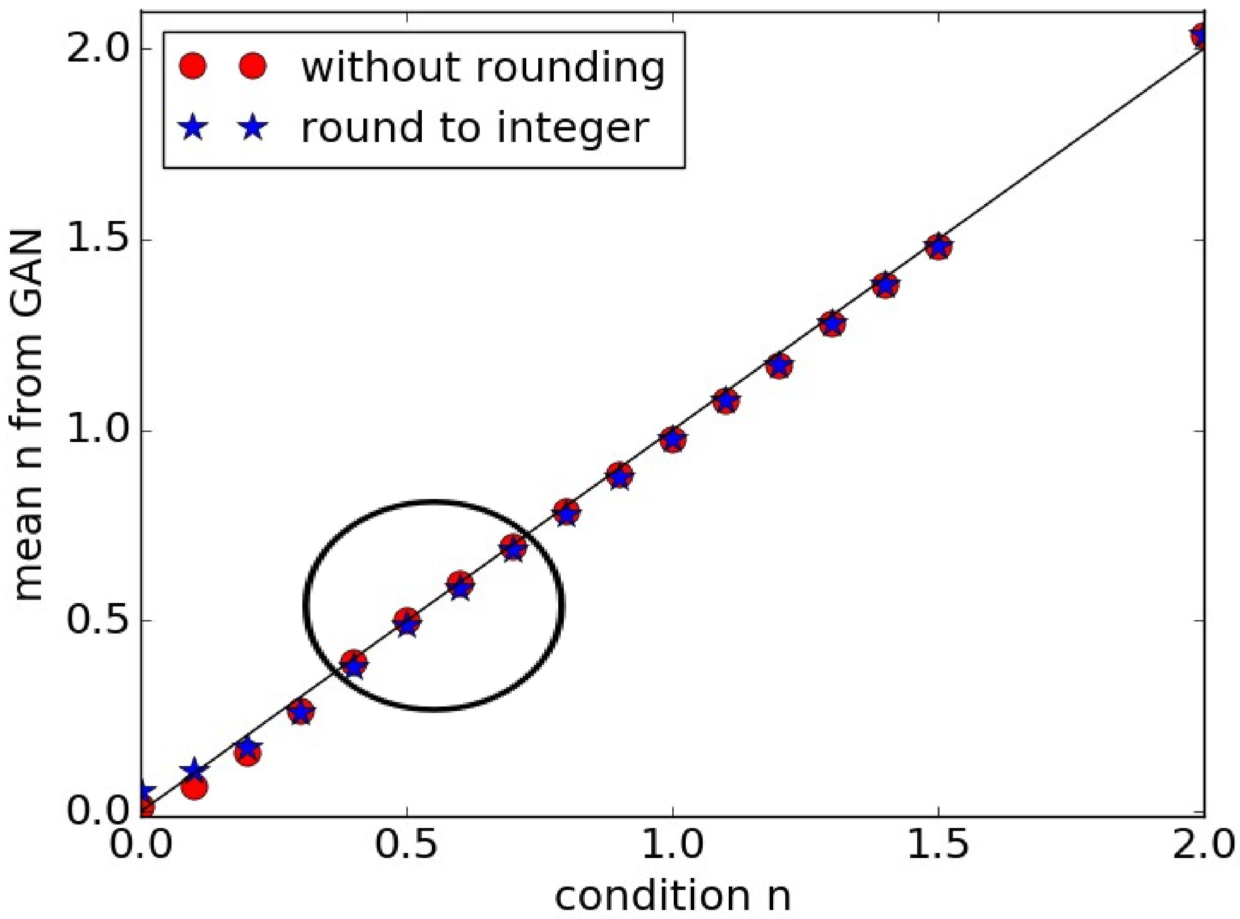
**Once trained, in generating stage,**

**We specify different n values.**



# Conditional GAN

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



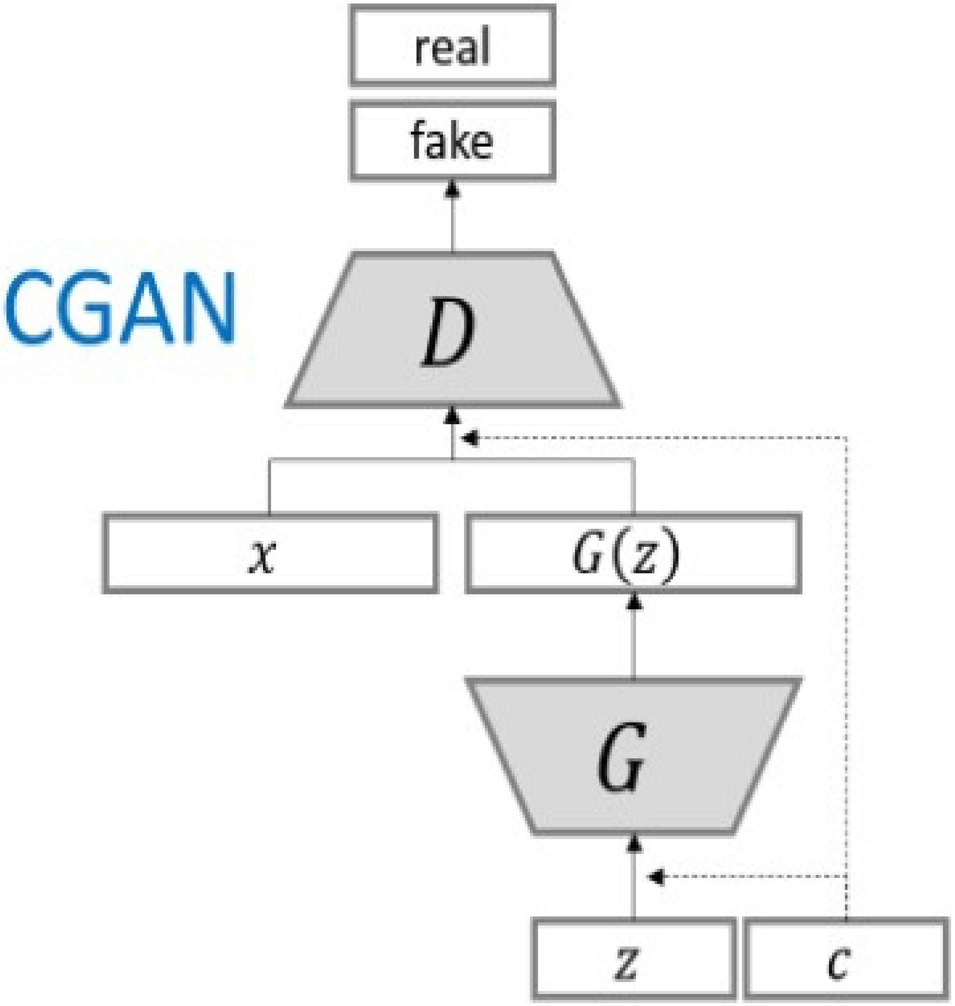
# add conditional information mu

**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**

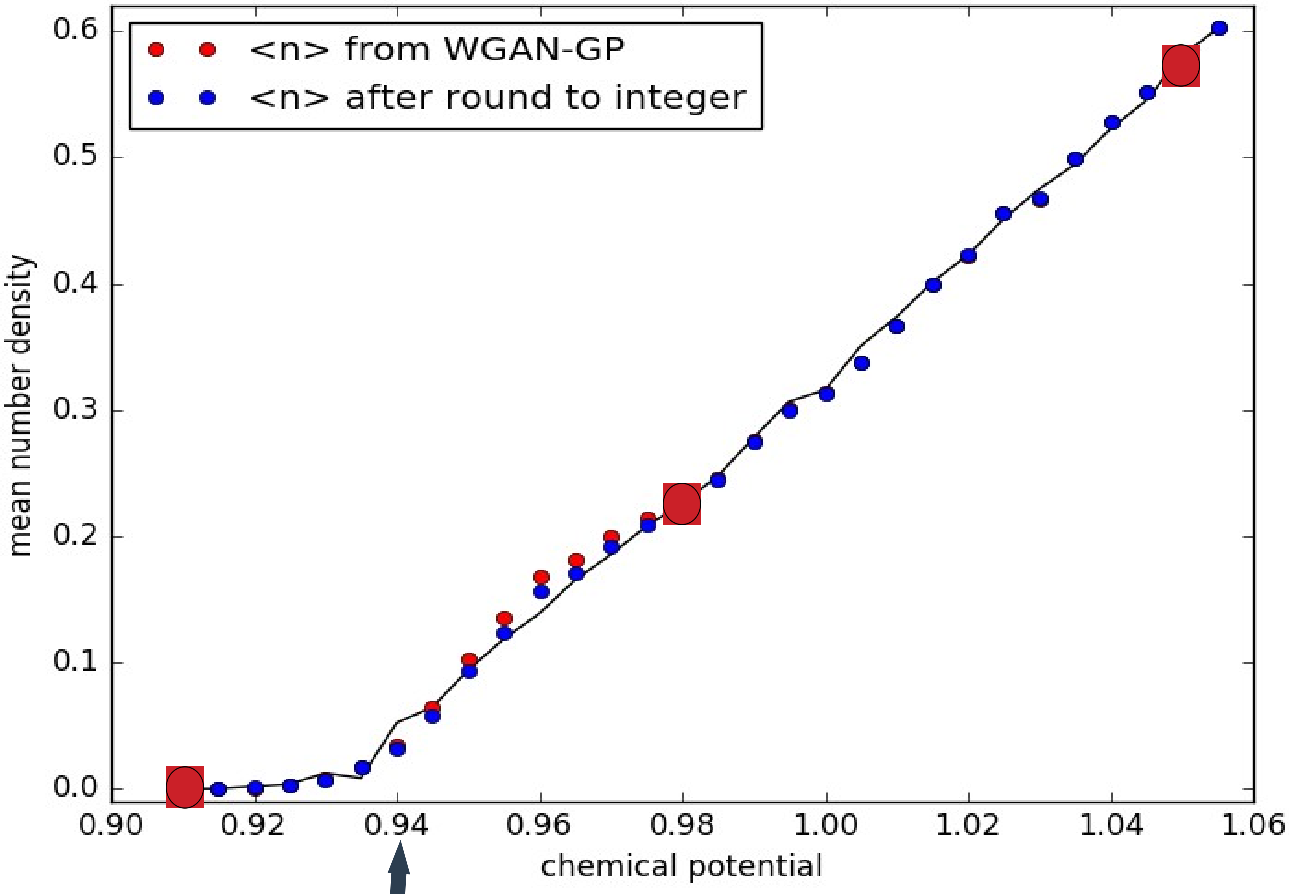
**Once trained, in generating stage,**

**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

(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

***Thanks!***

***热烈祝贺华南师范大学量子物质研究院成立！***

# Dualization approach for $\lambda\phi^4$

**Euclidean continuum action for complex 1+1d scalar field**

$$S^{\text{cont}} = \int_0^L \mathrm{d}x_1 \int_0^{1/T} \mathrm{d}x_2 \left[ (D_\nu \phi)^* (D_\nu \phi) + m^2 \phi^* \phi + \lambda (\phi^* \phi)^2 \right] \,, \qquad D_\nu = \partial_\nu + i\mu\delta_{\nu,2}$$

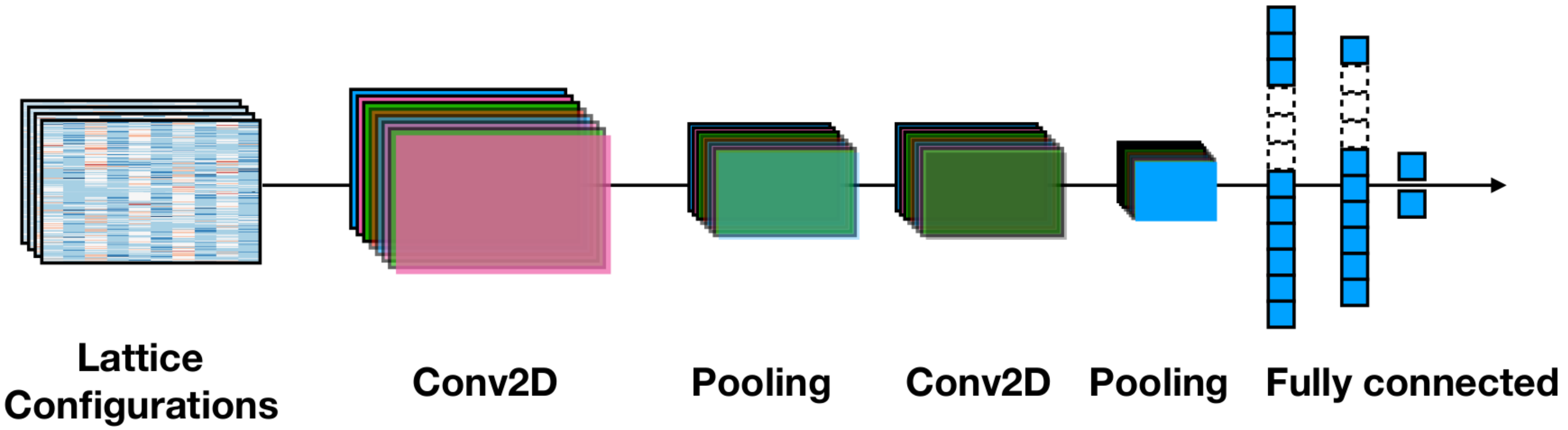
**On a lattice with n labels the lattice sites:**

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

**Partition function is defined from path integral:**

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

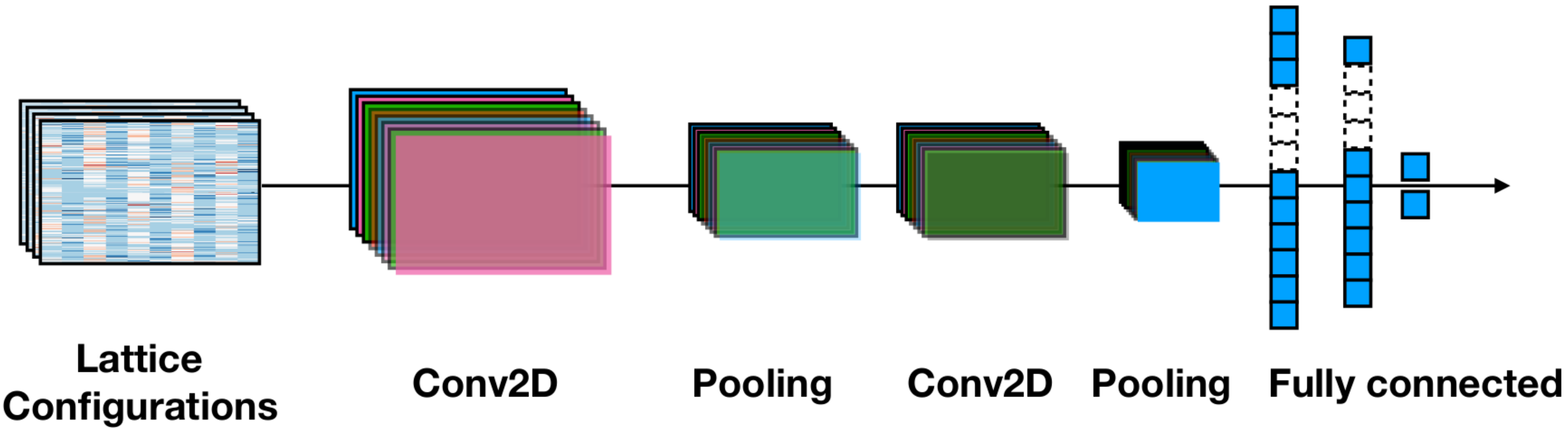
# DCNN Architechture – Classification




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

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

# DCNN Architechture – Regression




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

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

# Dualization approach for $\lambda\phi^4$

## Flux representation for partition function :

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

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

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

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

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

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

**Net particle density and squared field expectation**

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

**Flux representation for above :**

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

# GAN – distribution

## Zero-sum game – Nash equilibrium

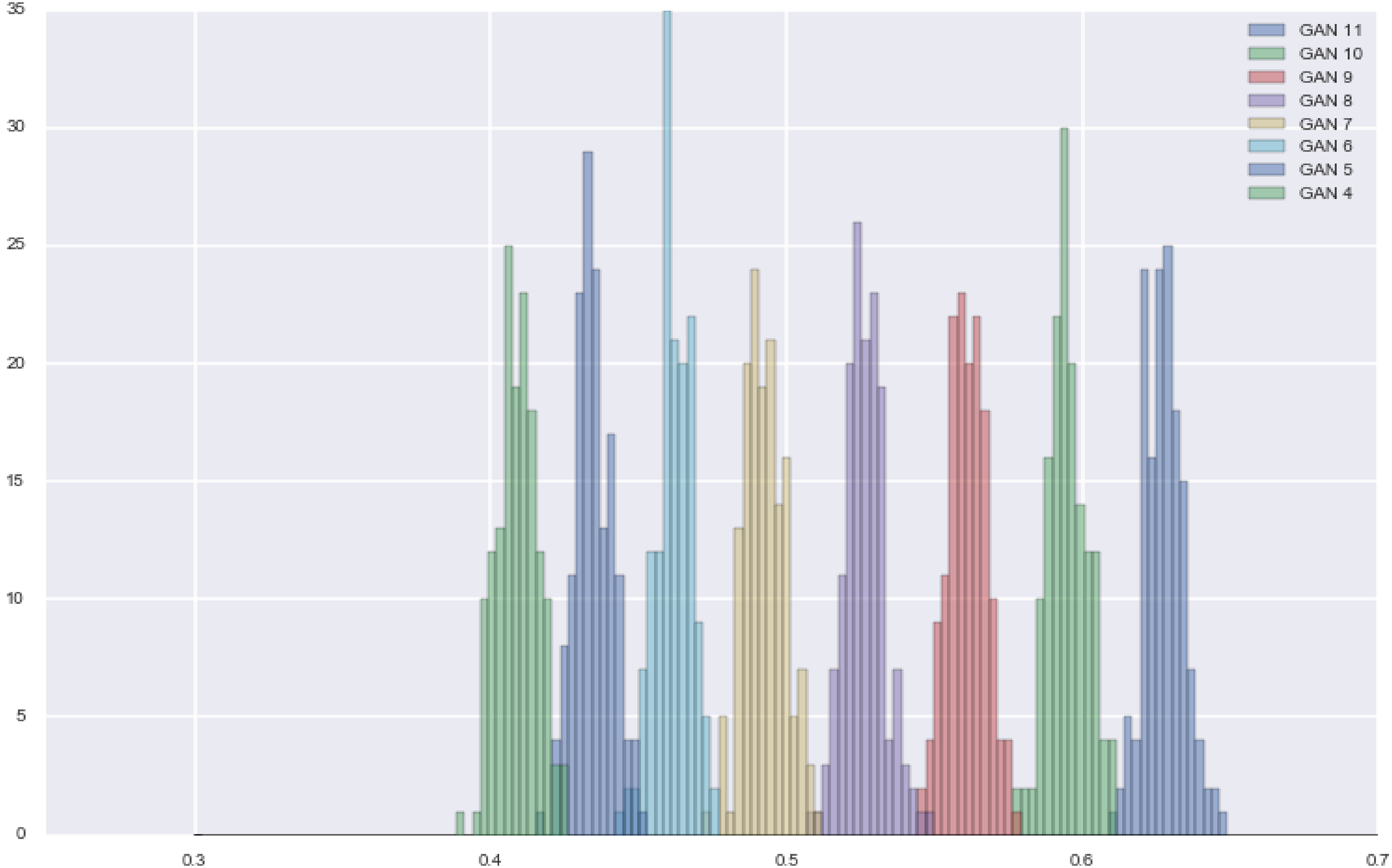
$$G^* = \arg\min_G \max_D (-\mathcal{L}_D(G, D))$$

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

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

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

# GAN – distribution



# GAN – distribution