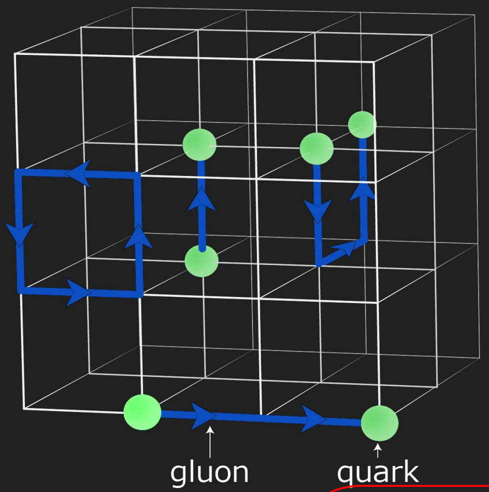
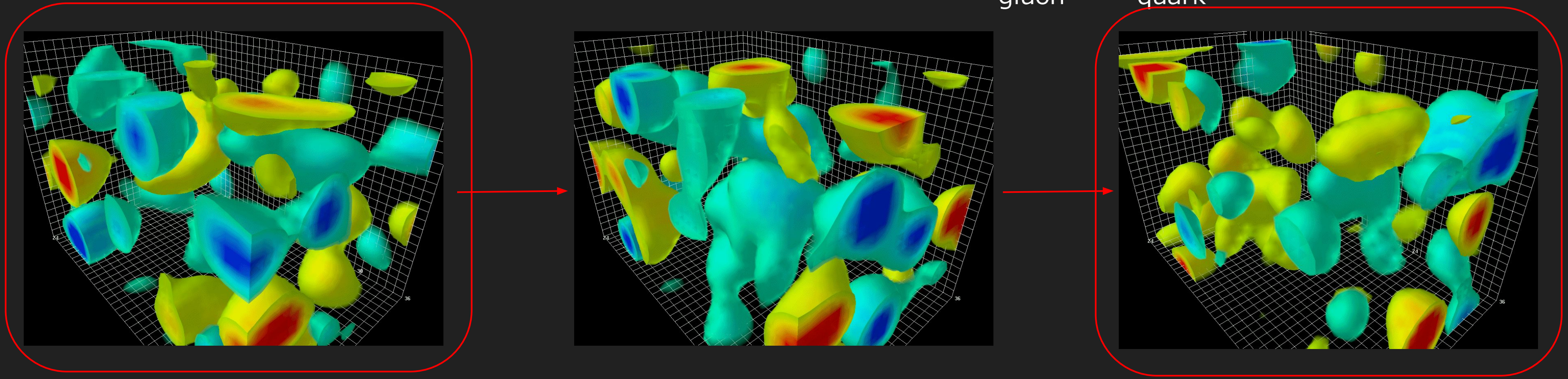
Machine learning action parameters for lattice QCD

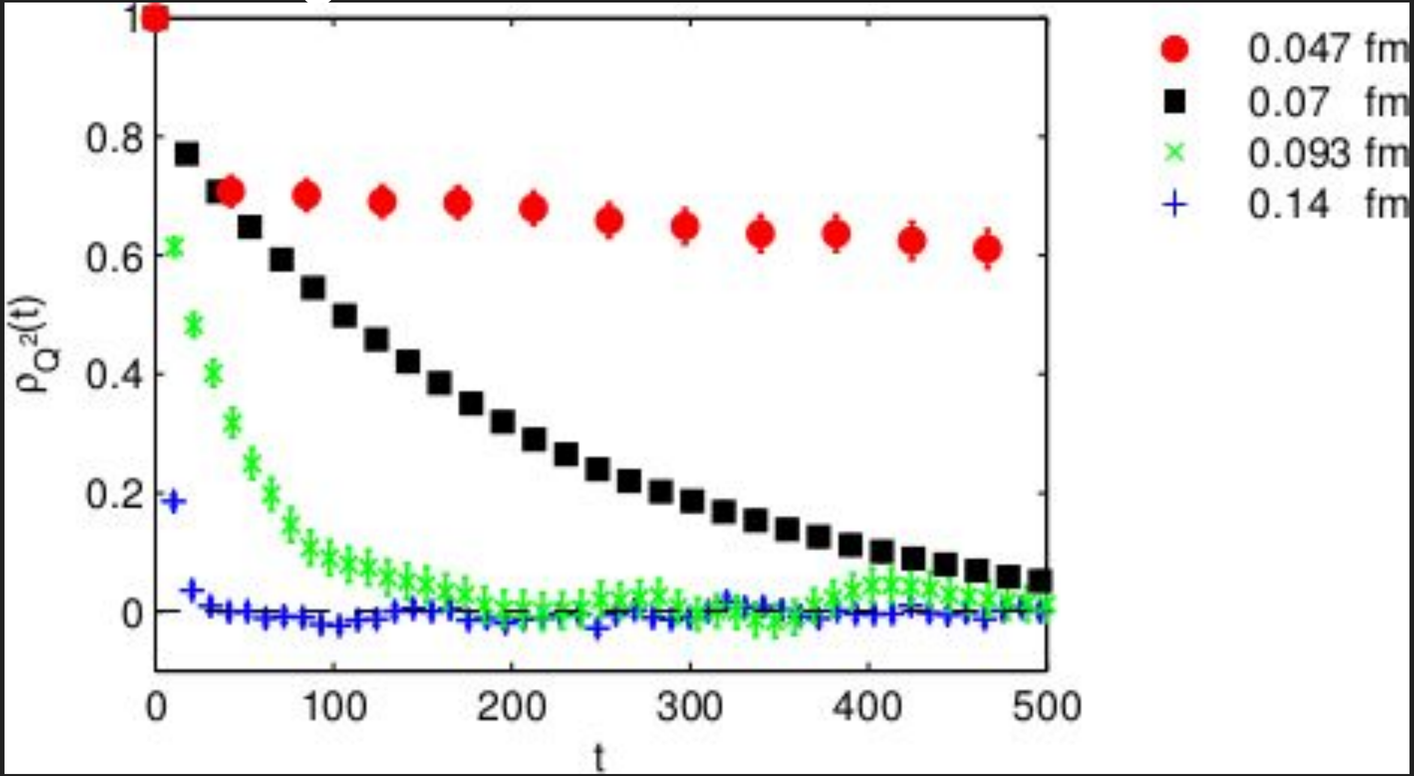
Emmy Trewartha

Hybrid Monte-Carlo



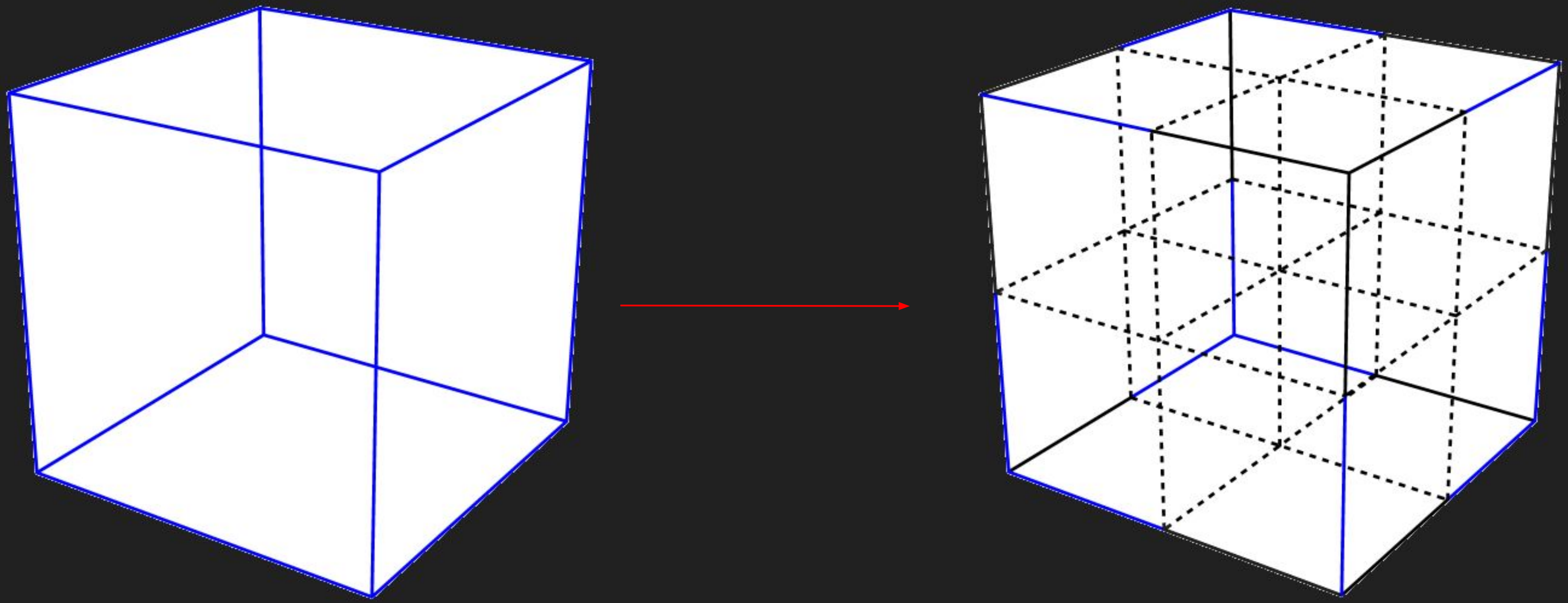


Critical Slowing down



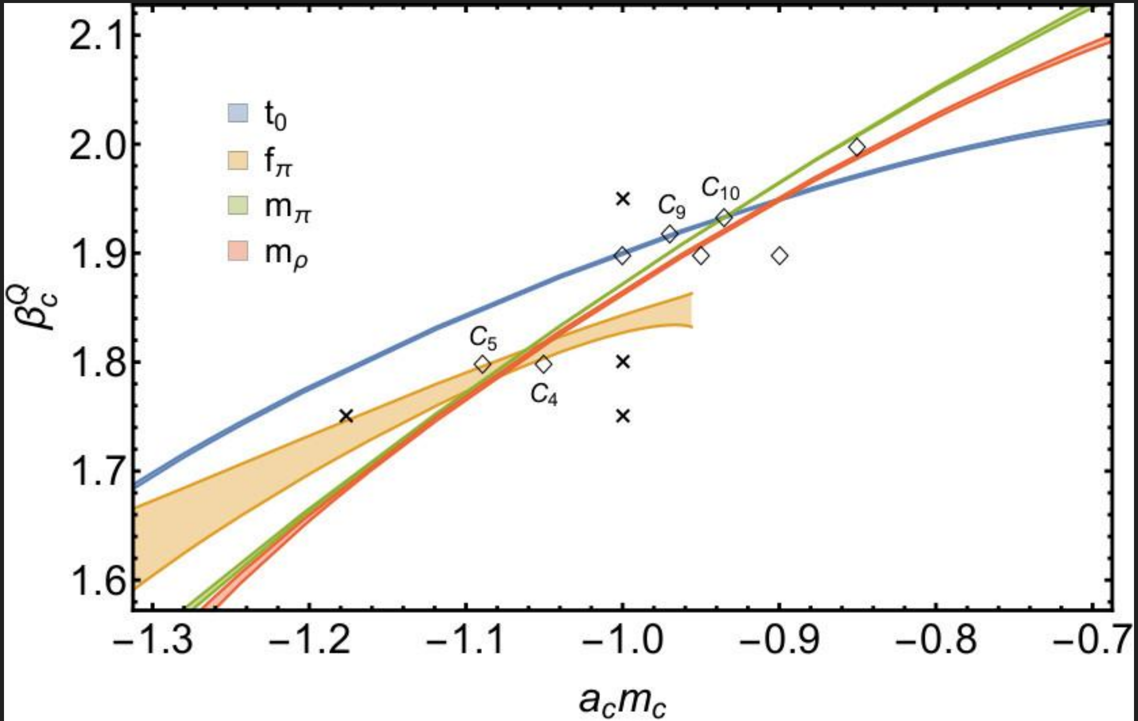
S. Schaefer, R. Sommer, and F. Virotta (ALPHA), Nucl. Phys. B845, 93 (2011), 1009.5228

Multiscale Lattice Generation



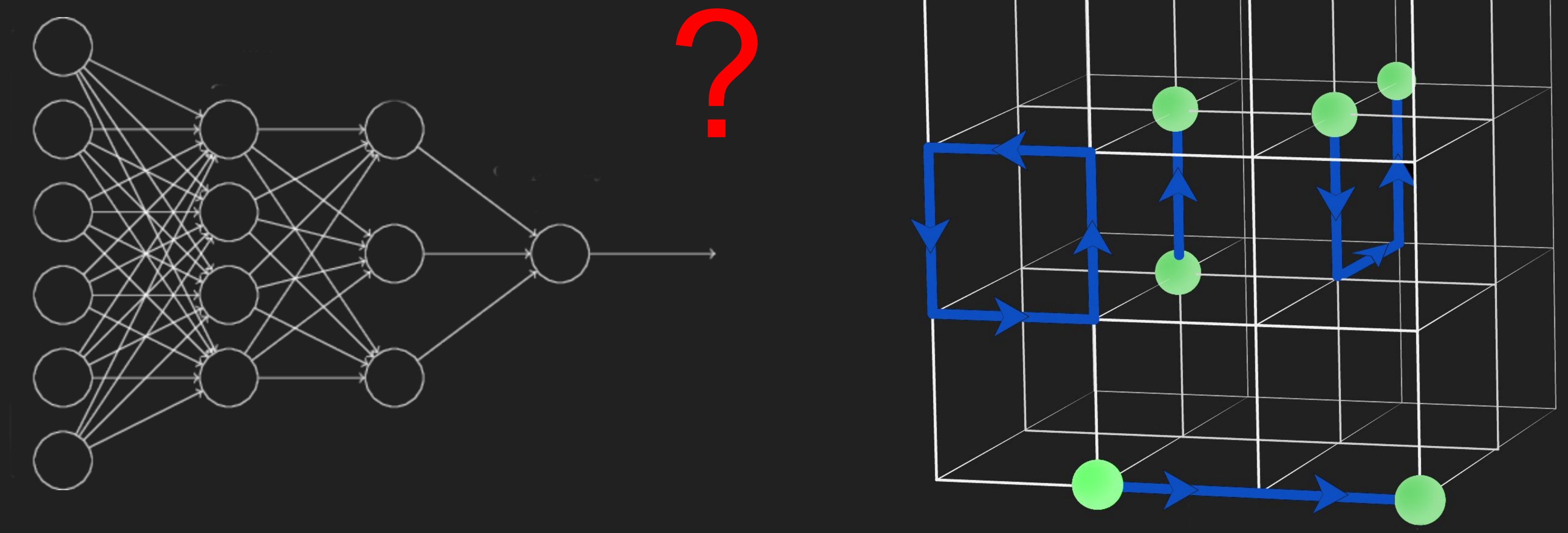
From Endres et al, PRD 92 no. 11 114516 (2015)

Multiscale Lattice Generation - Parameter Matching



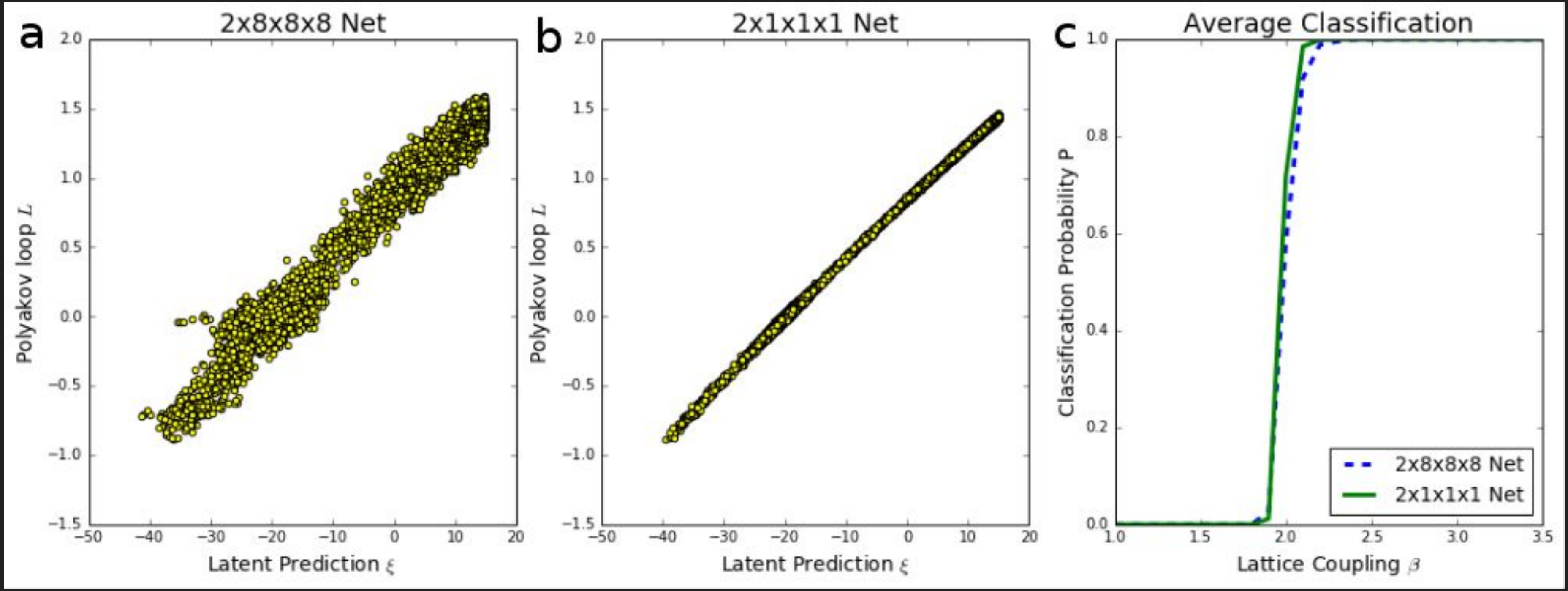
From Endres et al, PRD 94 no. 11 114502 (2016)

Can neural networks help?

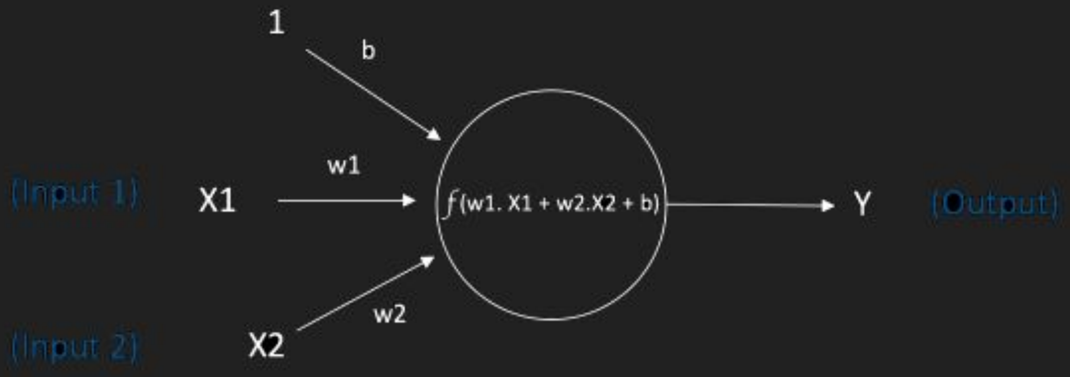


Previous work: NNs able to learn SU(2) deconfinement transition using Polyakov loop

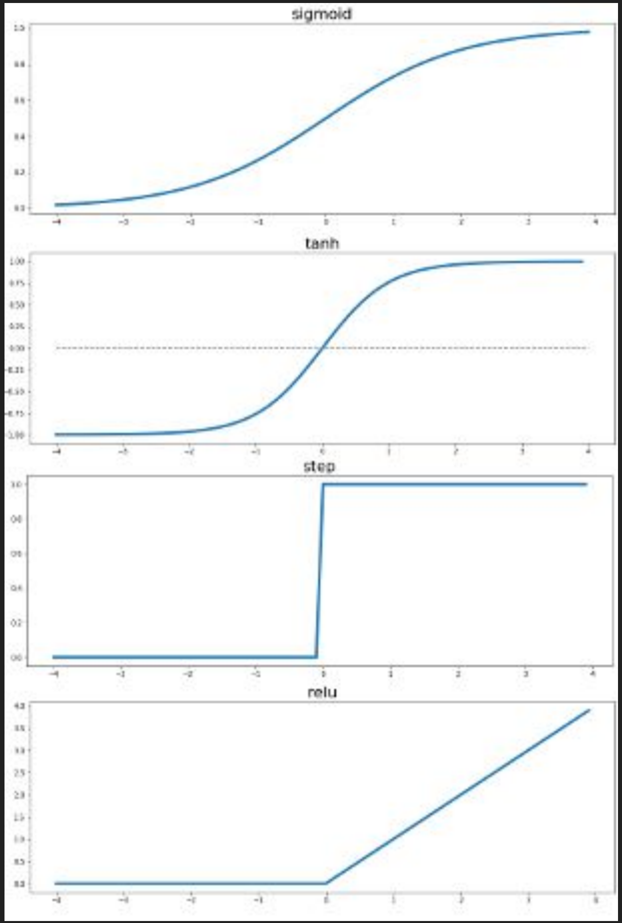
SJ Wetzel and M Scherzer, PRB96 no. 18 184410 (2017)

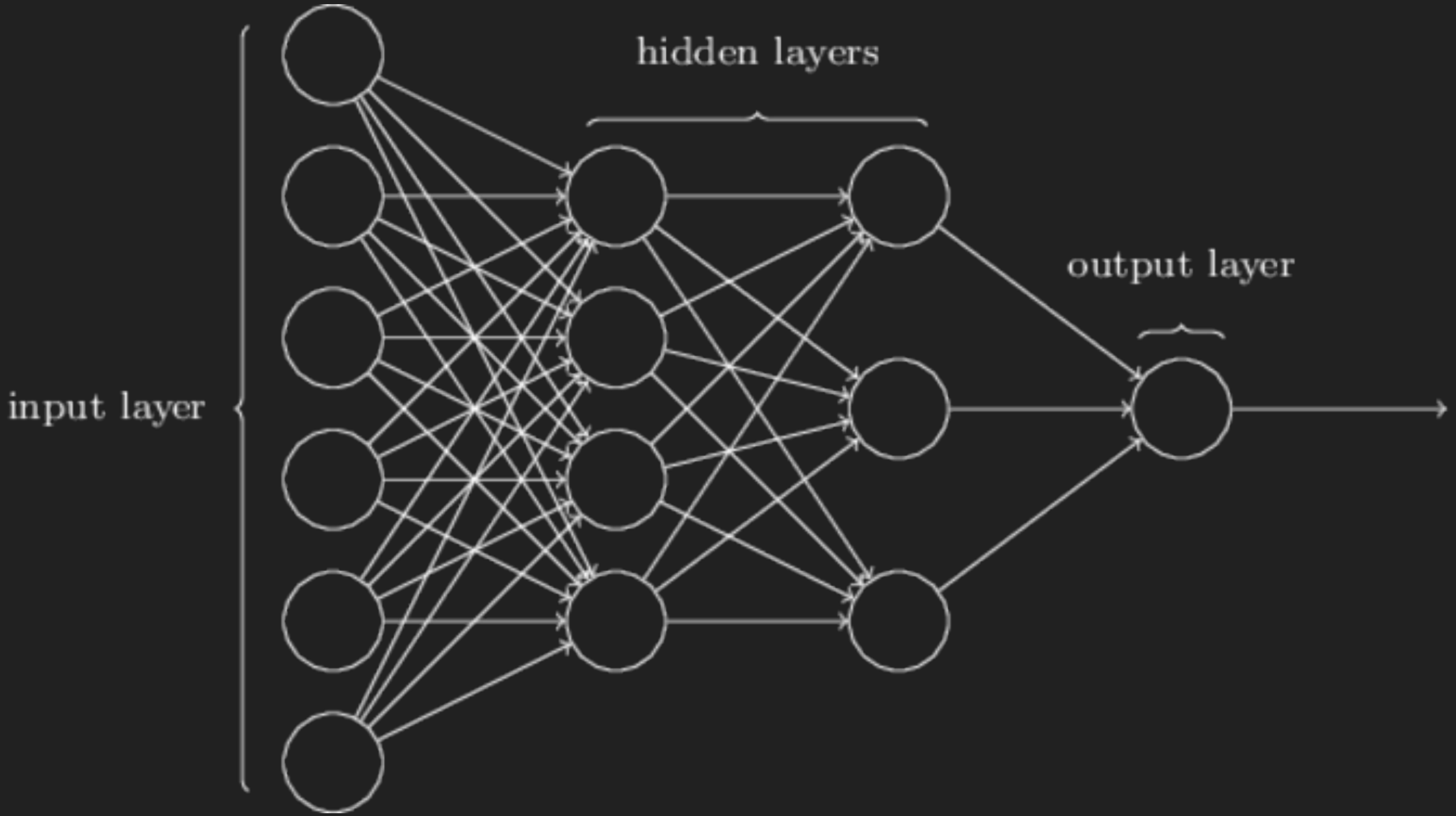


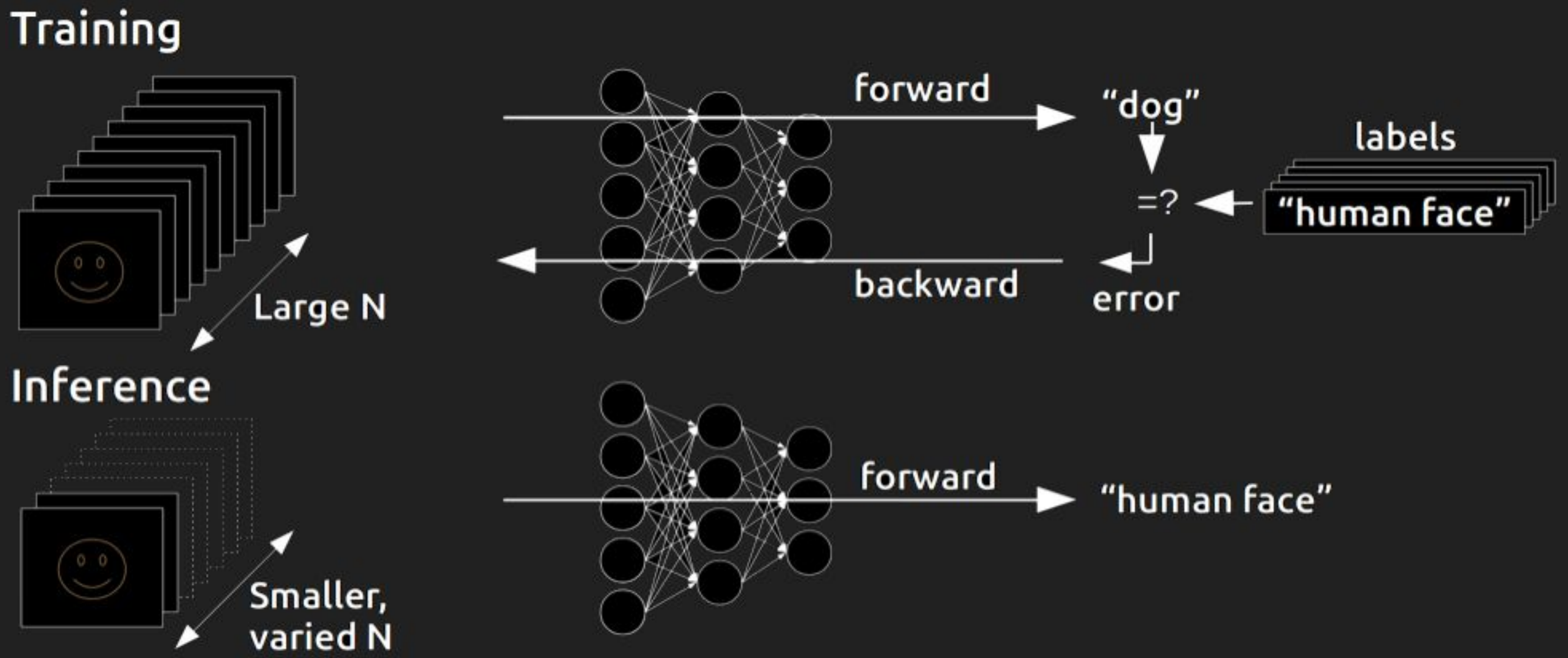
Neural Networks



Output of neuron = Y= f(w1. X1 + w2.X2 + b)


sigmoid
tanh
step
relu


hidden layers
input layer
output layer


Training
forward
"dog"
labels
=?
"human face"
backward
error
Large N
Inference
forward
"human face"
Smaller,
varied N

Training set:

$12^3 \times 36$ SU(2) ensembles of 1000 configurations each

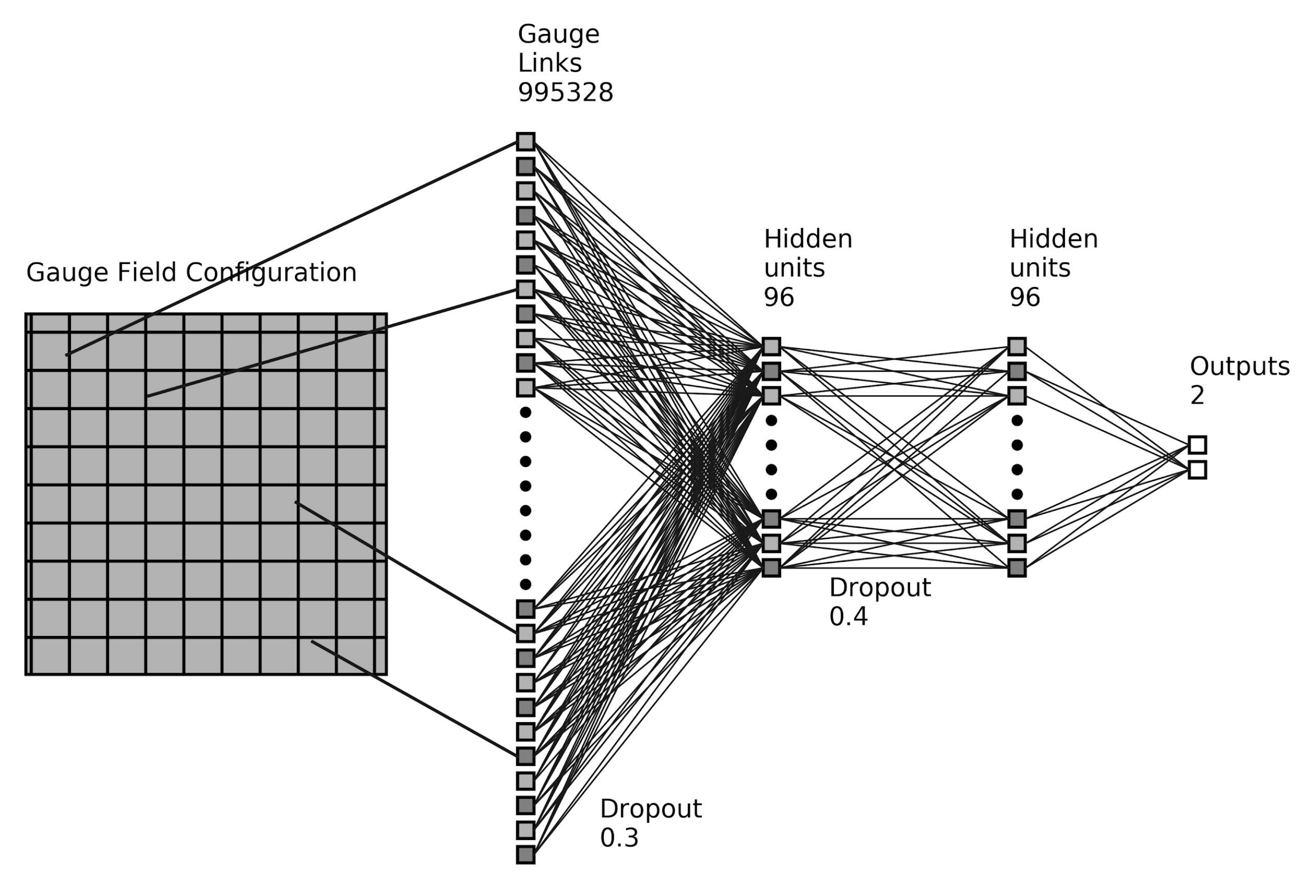
Two grids in β, m space:

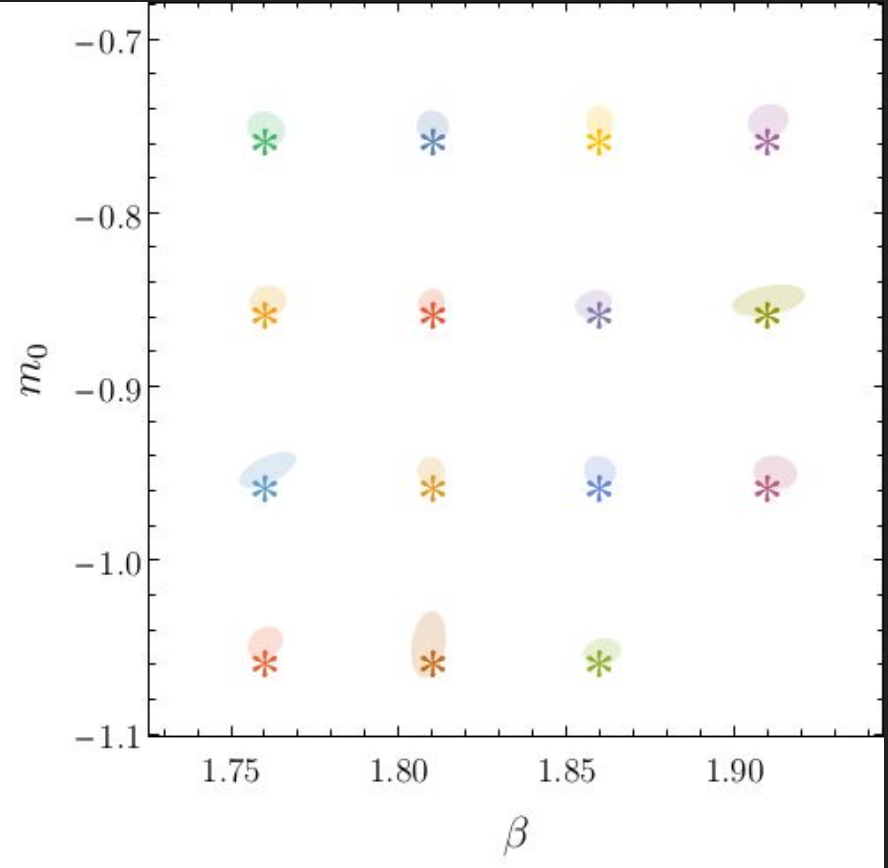
$\beta \in \{1.785, 1.835, 1.885, 1.935, 1.985\}$ and $m \in \{-0.7, -0.8, -0.9, -1.0\}$, excluding the pair $\{\beta, m\} = \{1.985, -1.0\}$

$\beta \in \{1.76, 1.81, 1.86, 1.91\}$ and $m \in \{-0.75, -0.85, -0.95, -1.05\}$, excluding the pair $\{\beta, m\} = \{1.91, -1.05\}$

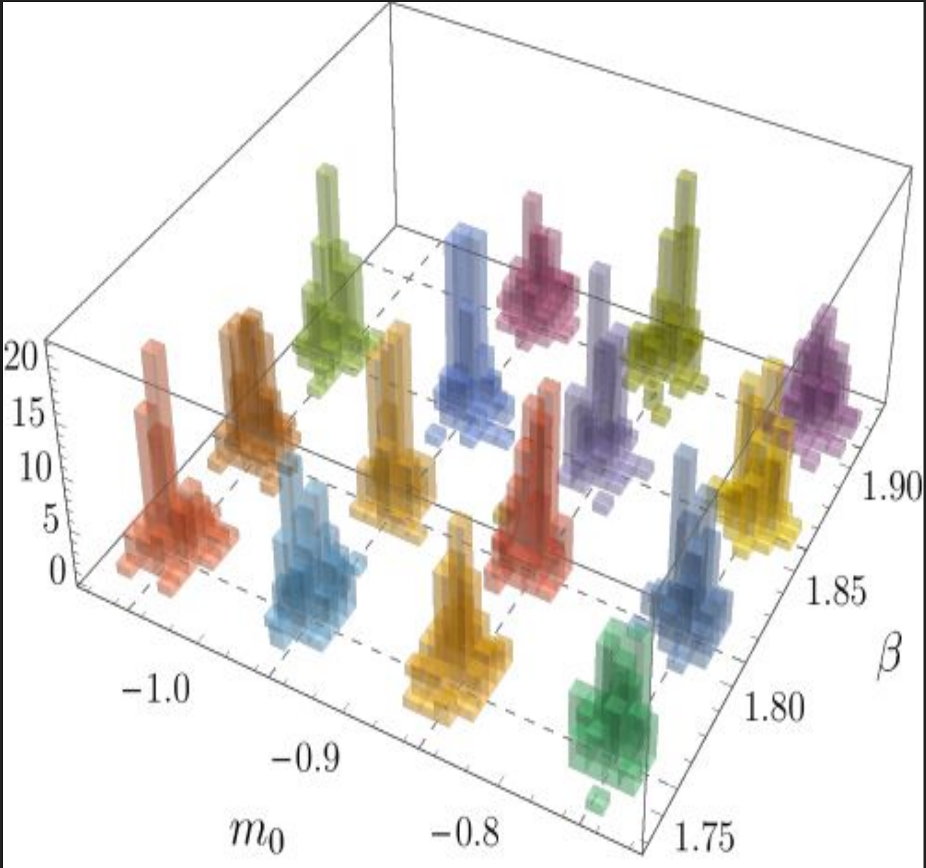
850 randomly selected configurations used for training, 150 for validation

First Attempt: Configurations as input directly





−0.7
−0.8
−0.9
−1.0
−1.1
m_0
1.75
1.80
1.85
1.90
β



20
15
10
5
0
−1.0
−0.9
−0.8
m_0
1.90
1.85
1.80
1.75
β

Measure independence of configs using autocorrelation;

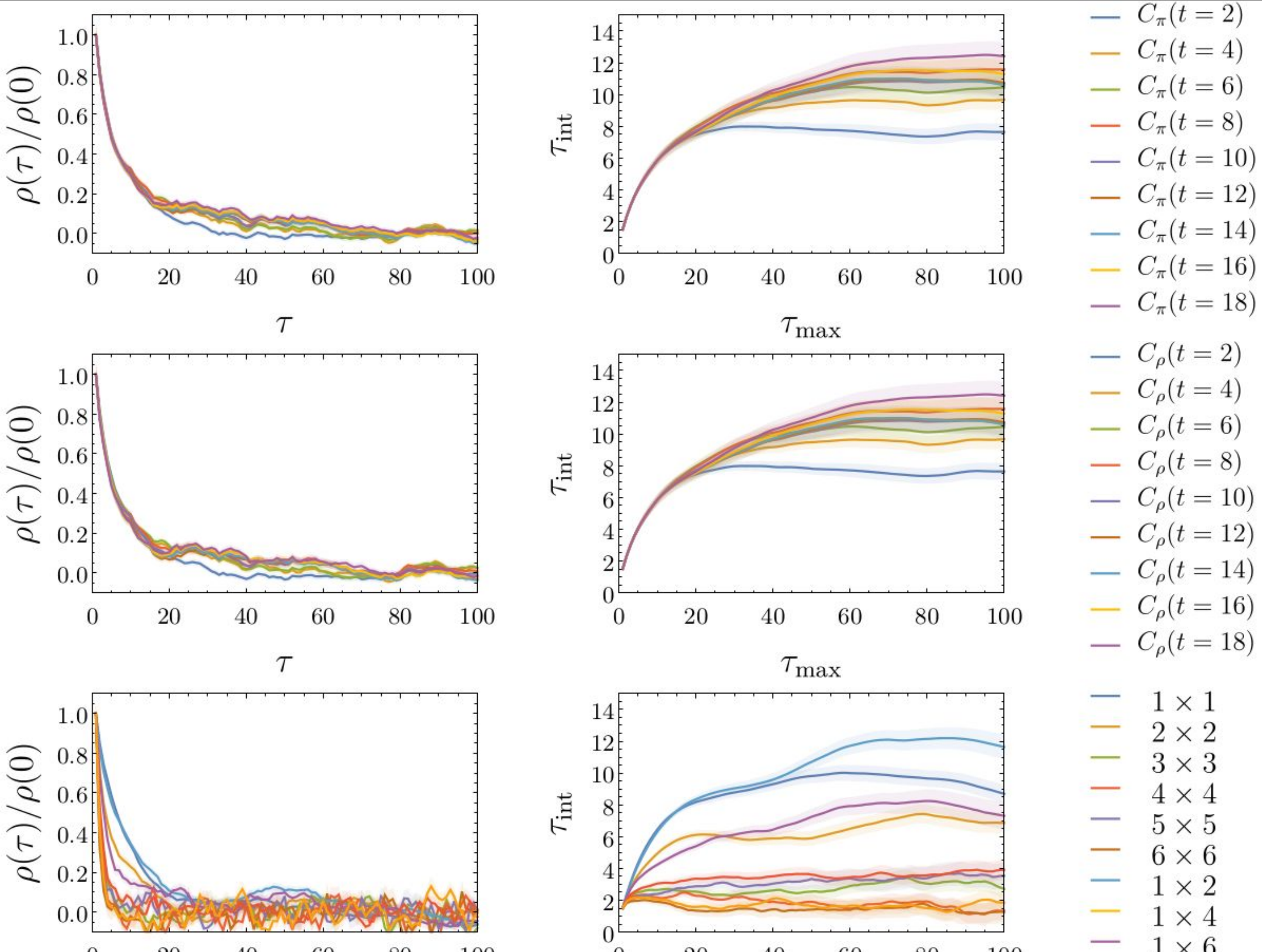
$$\rho(\tau) = \sum_{\tau'} \langle (\mathcal{O}(\tau') - \langle \mathcal{O}(\tau') \rangle)(\mathcal{O}(\tau' + \tau) - \langle \mathcal{O}(\tau' + \tau) \rangle) \rangle$$

At large τ behaves as;

$$\frac{\rho(\tau)}{\rho(0)} \approx \exp[-\frac{\tau}{\tau_{exp}}]$$

Then define

$$\tau_{int} = \frac{1}{2} + \lim_{\tau_{max} \to \infty} \frac{1}{\rho_0} \sum_{\tau=0}^{\tau_{max}} \rho(\tau)$$



$\rho(\tau)/\rho(0)$
τ
$\tau_{\rm int}$
$\tau_{\rm max}$
$C_\pi(t=2)$
$C_\pi(t=4)$
$C_\pi(t=6)$
$C_\pi(t=8)$
$C_\pi(t=10)$
$C_\pi(t=12)$
$C_\pi(t=14)$
$C_\pi(t=16)$
$C_\pi(t=18)$
$C_\rho(t=2)$
$C_\rho(t=4)$
$C_\rho(t=6)$
$C_\rho(t=8)$
$C_\rho(t=10)$
$C_\rho(t=12)$
$C_\rho(t=14)$
$C_\rho(t=16)$
$C_\rho(t=18)$
1×1
2×2
3×3
4×4
5×5
6×6
1×2
1×4
1×6

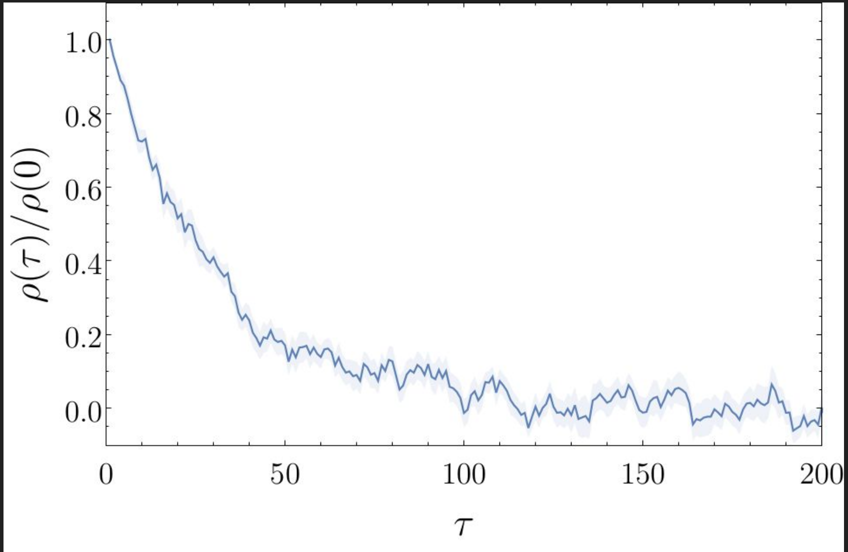
Define;

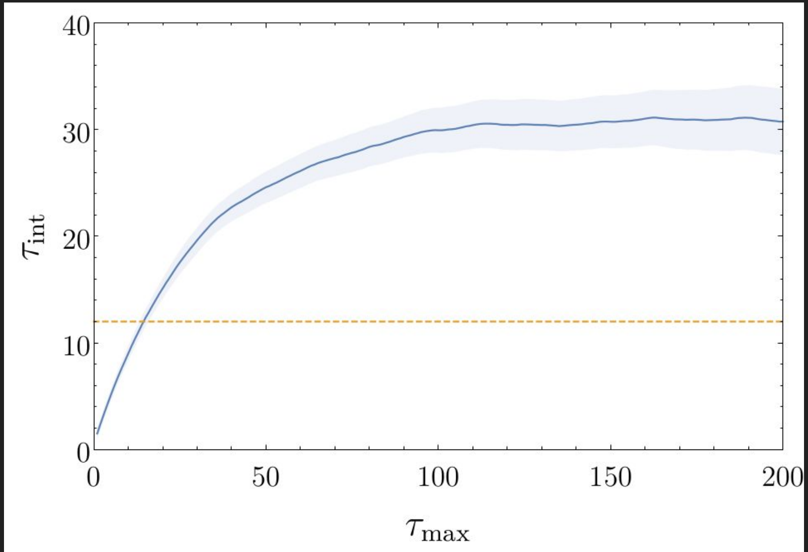
$$\rho(\tau) = \left[P_\alpha\left(c^\alpha(\tau)\right) + P_\beta\left(c^\beta(\tau)\right)\right] - 1$$

$$\tau_{int} = \frac{1}{2} + \lim_{\tau_{max}\to\infty} \frac{1}{\rho_0} \sum_{\tau=0}^{\tau_{max}} \rho(\tau)$$

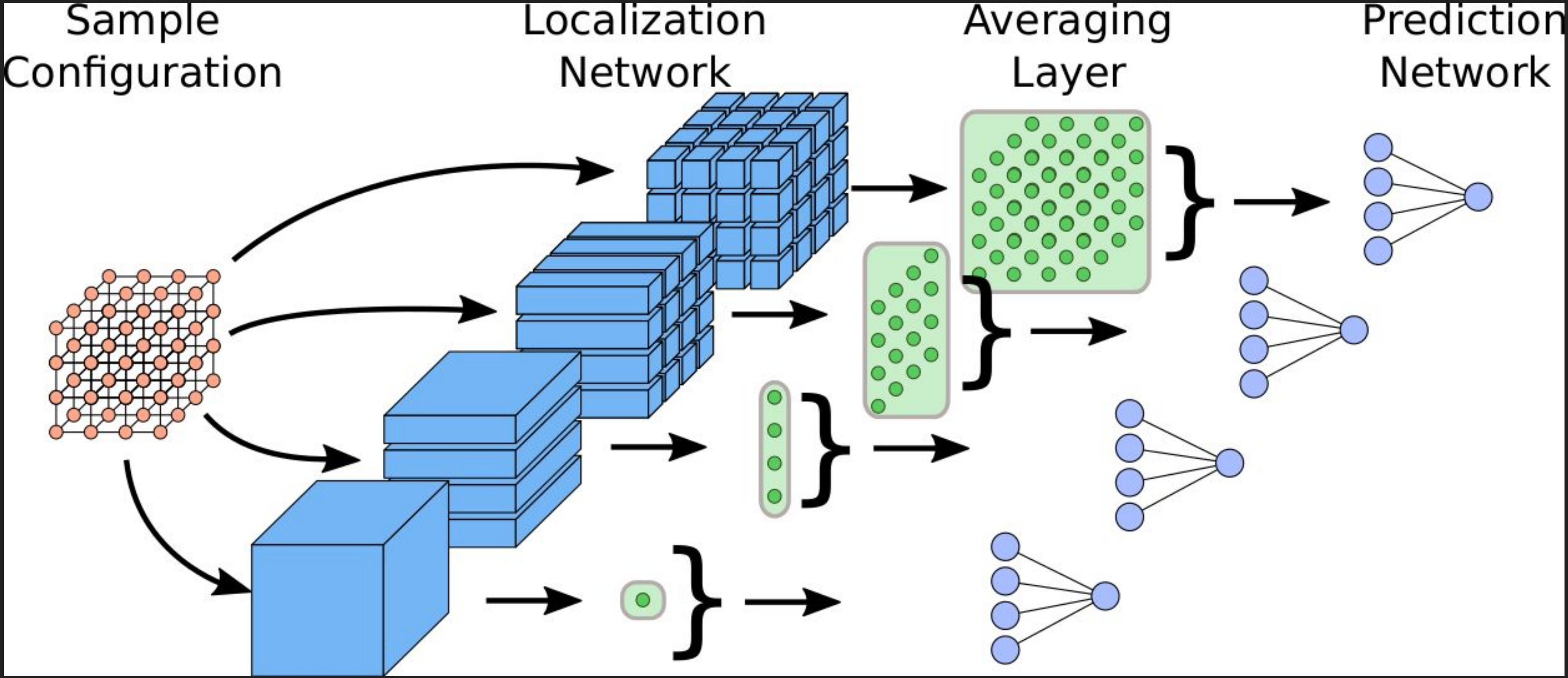
Generate ten independent streams of 10,000 trajectories denoted F1, . . . , F10, saved every trajectory, generated with the same values of β = 1.76 and m0 = −0.75.

Autocorrelation from classifier network



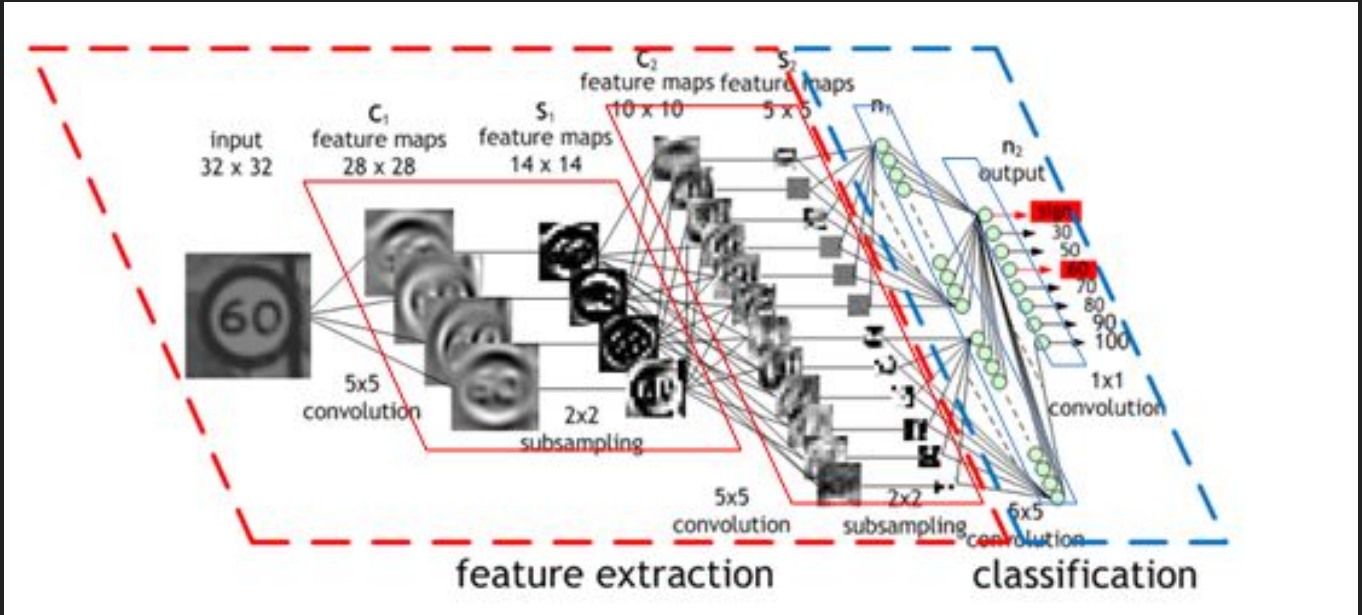


Interpretability - custom network design?

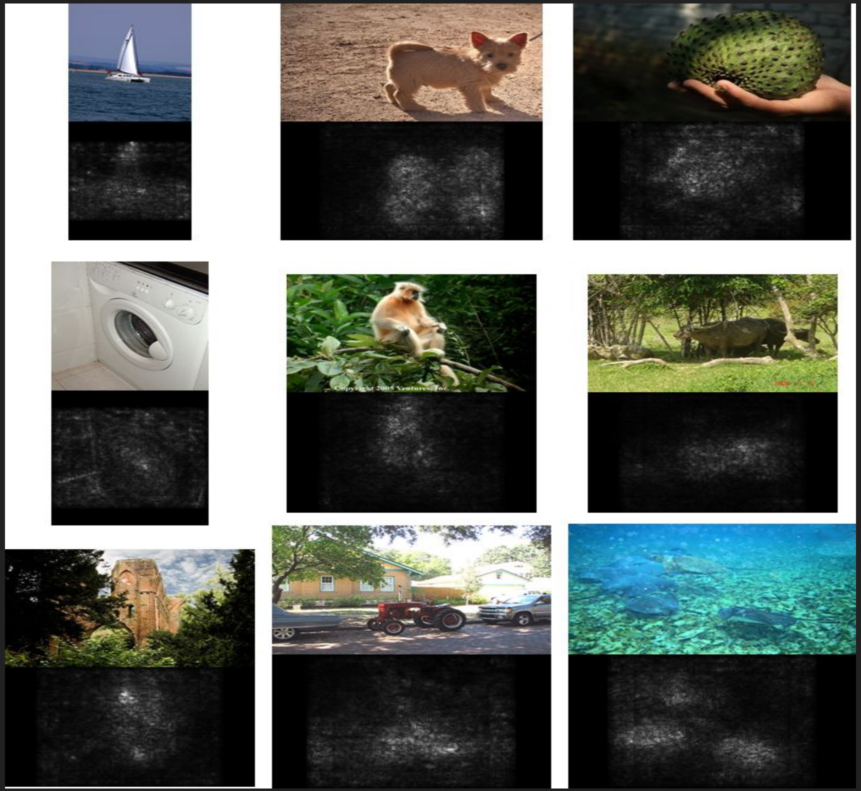


SJ Wetzel and M Scherzer, PRB96 no. 18 184410 (2017)

Interpretability - interpretable mid-layers?

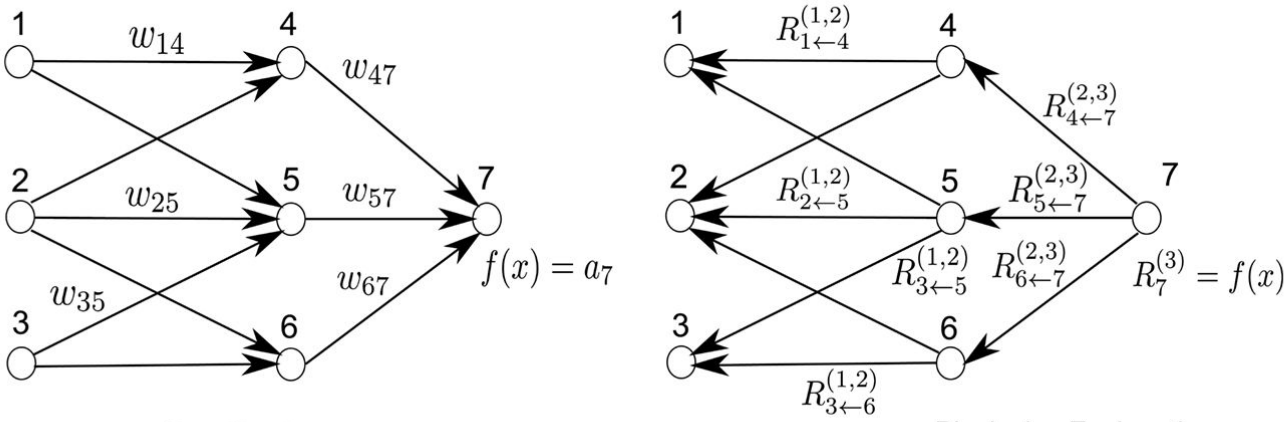


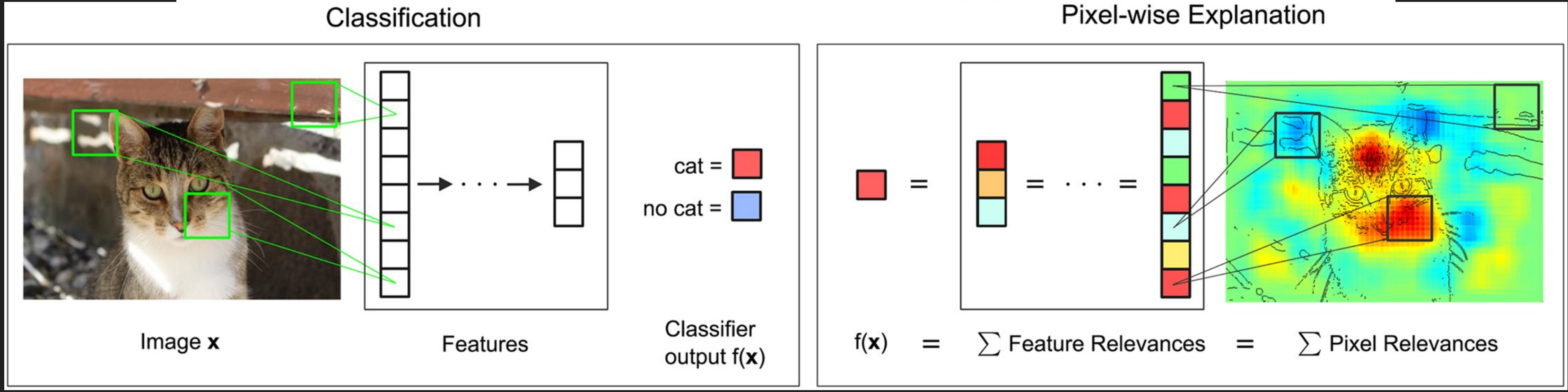
Interpretability - Partial Derivatives?



K. Simonyan, A. Vedaldi, and A. Zisserman. ICLR Workshop, 2014

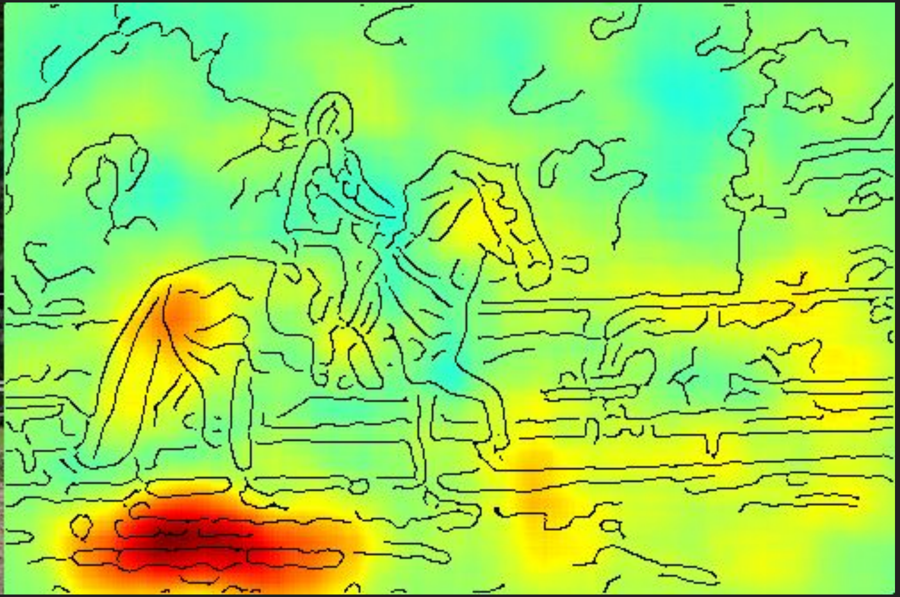
Interpretability - Pixelwise Relevance?



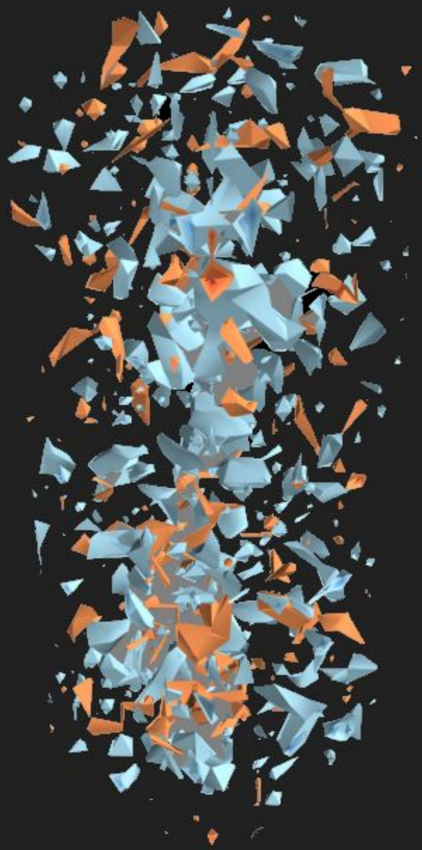


Bach S, Binder A, Montavon G, Klauschen F, Müller KR, et al. (2015) PLOS ONE 10(7): e0130140

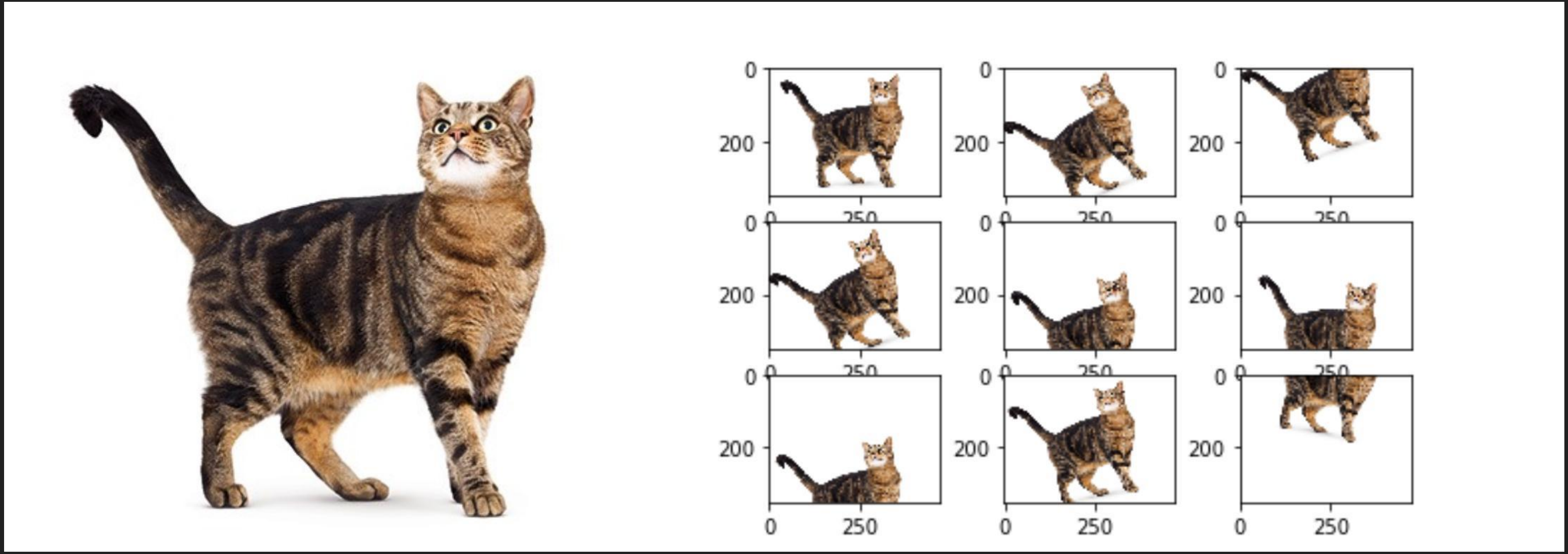
Interpretability -Pixelwise Relevance?



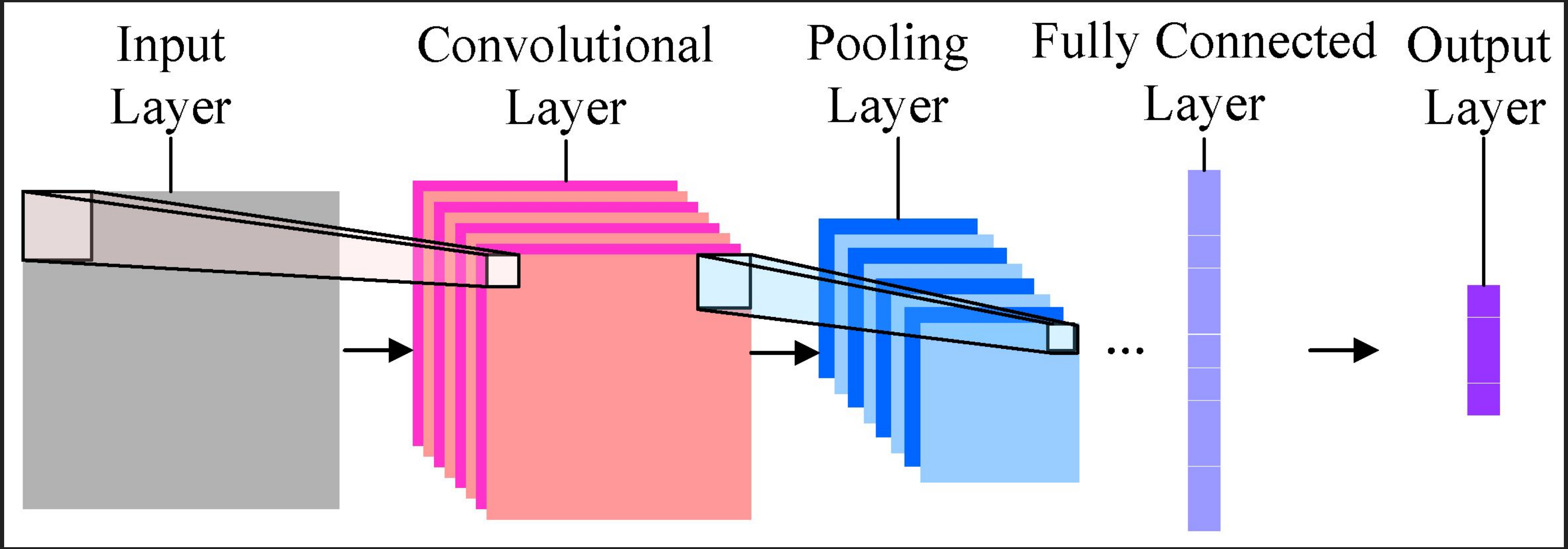
Sitewise Classifier Derivatives



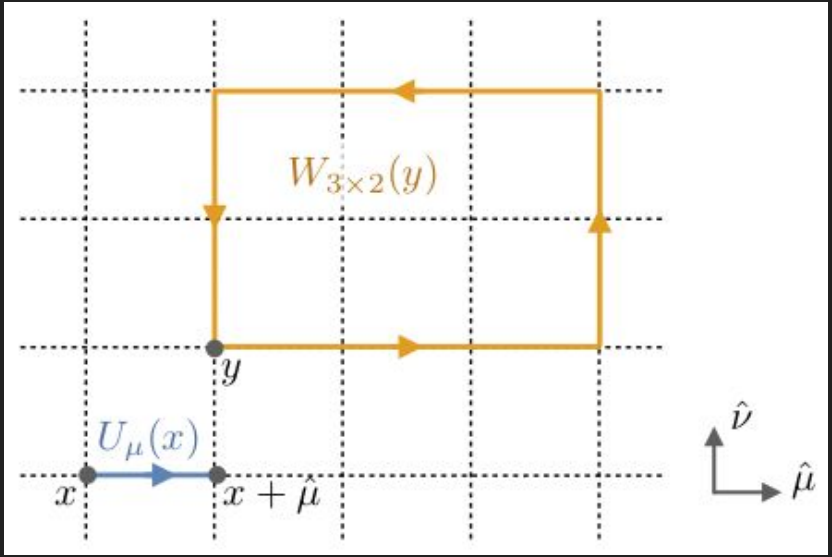
Incorporating Symmetries in Network Design



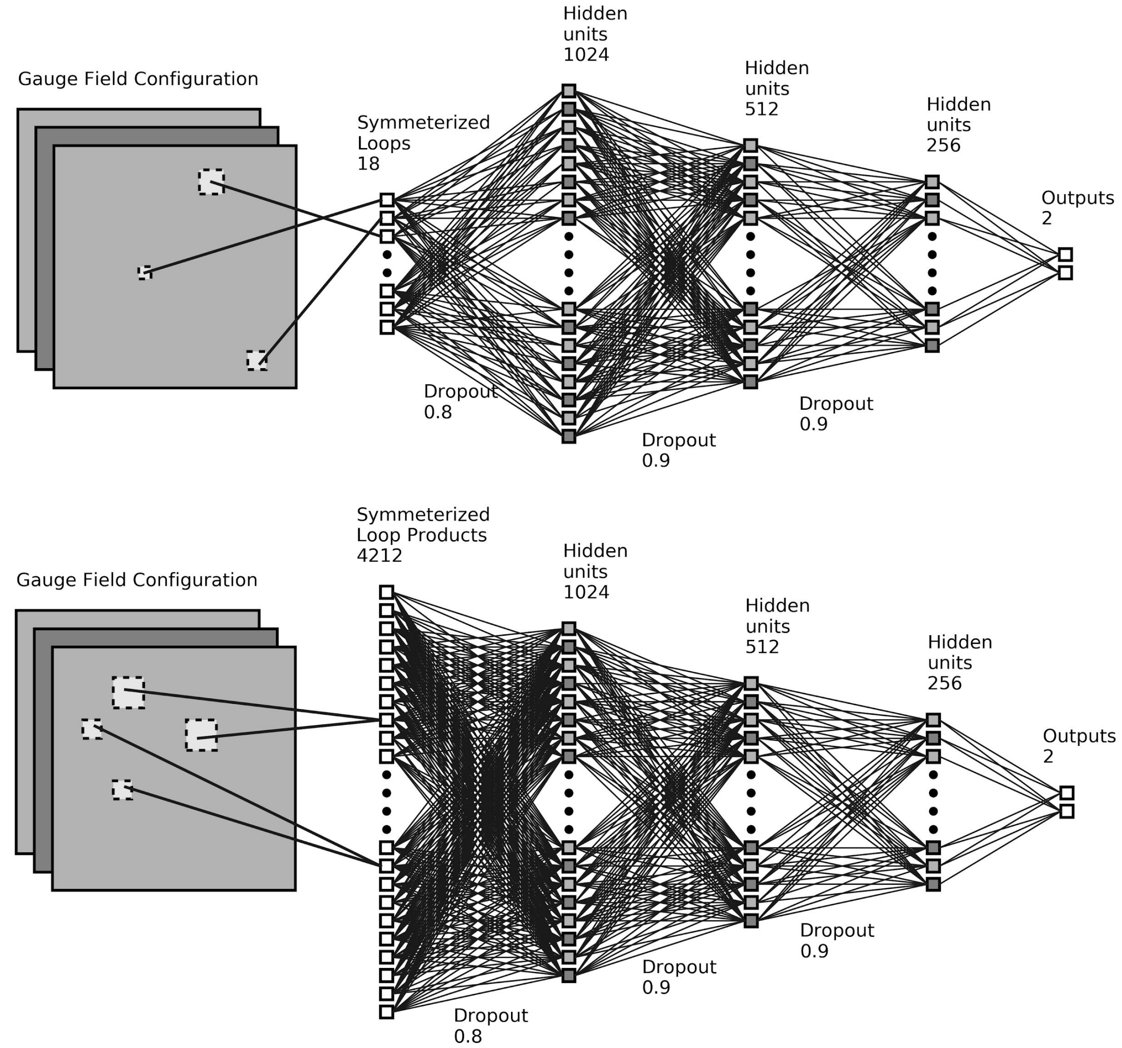
Incorporating Symmetries in Network Design



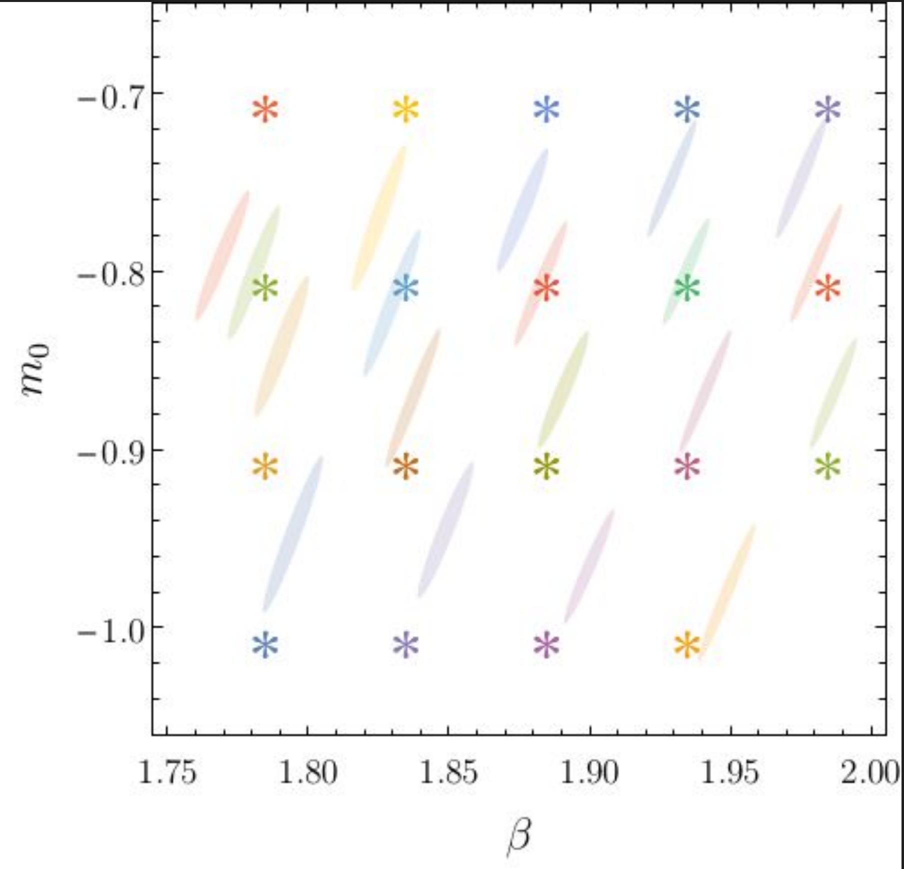
Second attempt: Incorporate symmetries directly



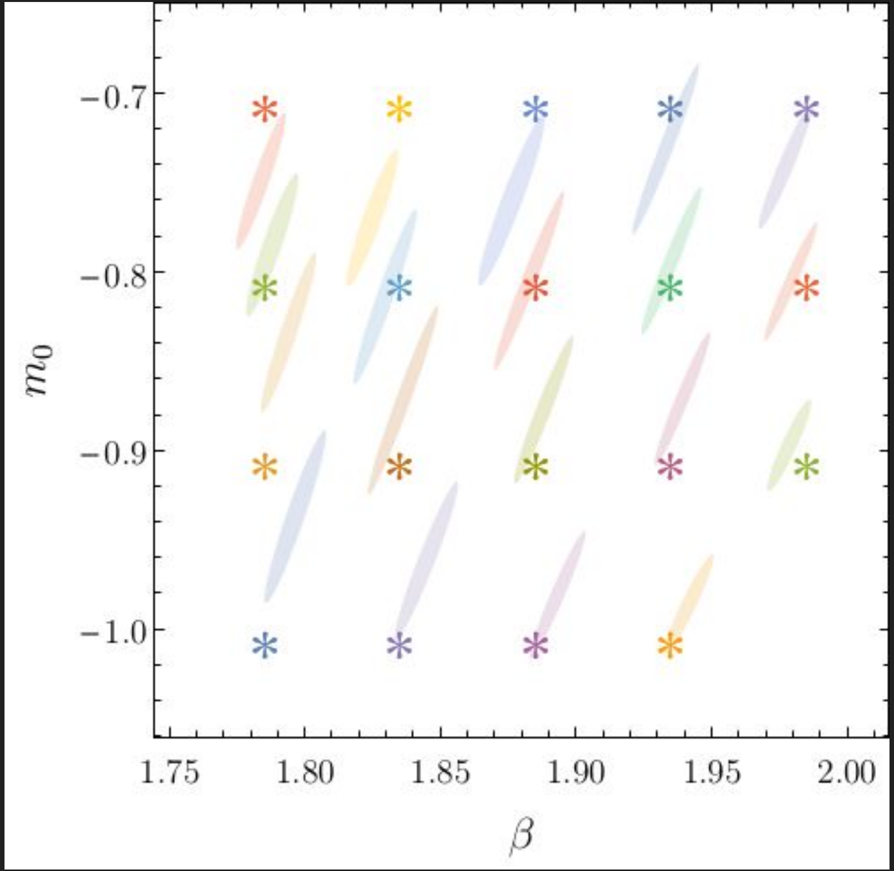
$$\mathcal{W}_{j\times k,l\times m}(R) = \sum_{|r|=R} \sum_{\ell\in\mathcal{O}(j\times k)} \sum_{\ell'\in\mathcal{O}(l\times m)} \sum_x \mathcal{W}_\ell(x)\mathcal{W}_{\ell'}(x+r)$$



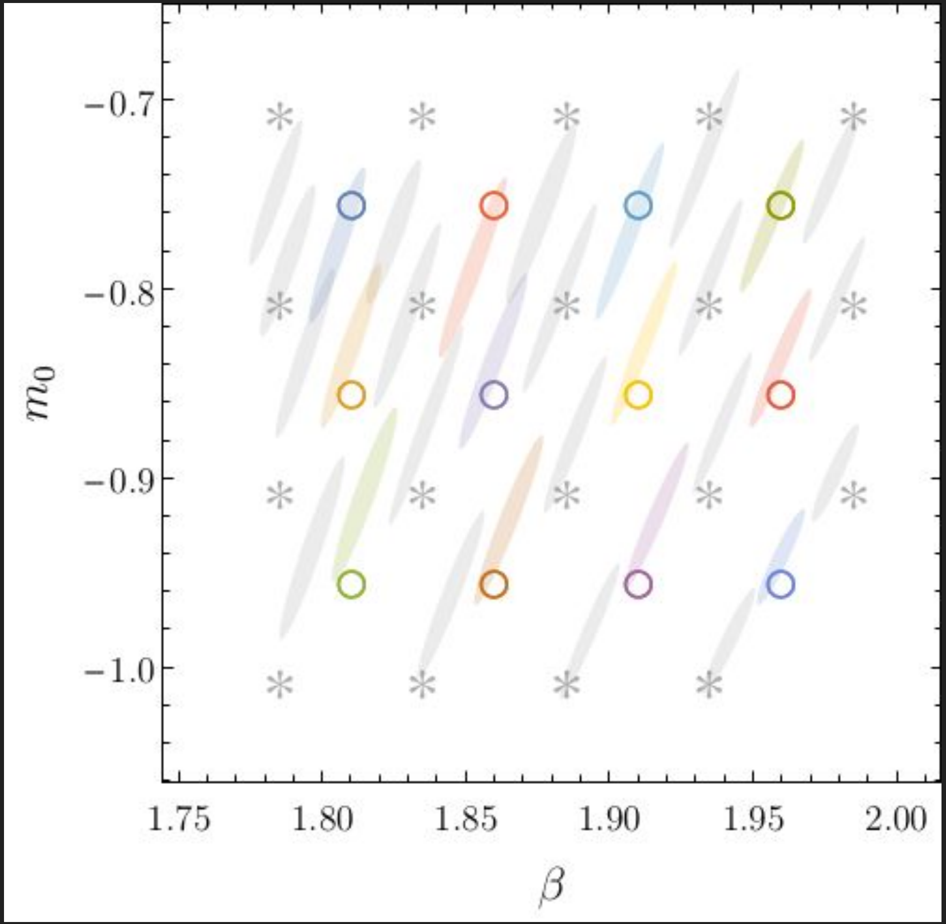
Gauge Field Configuration
Symmeterized
Loops
18
Hidden
units
1024
Hidden
units
512
Hidden
units
256
Outputs
2
Dropout
0.8
Dropout
0.9
Dropout
0.9
Gauge Field Configuration
Symmeterized
Loop Products
4212
Hidden
units
1024
Hidden
units
512
Hidden
units
256
Outputs
2
Dropout
0.8
Dropout
0.9
Dropout
0.9



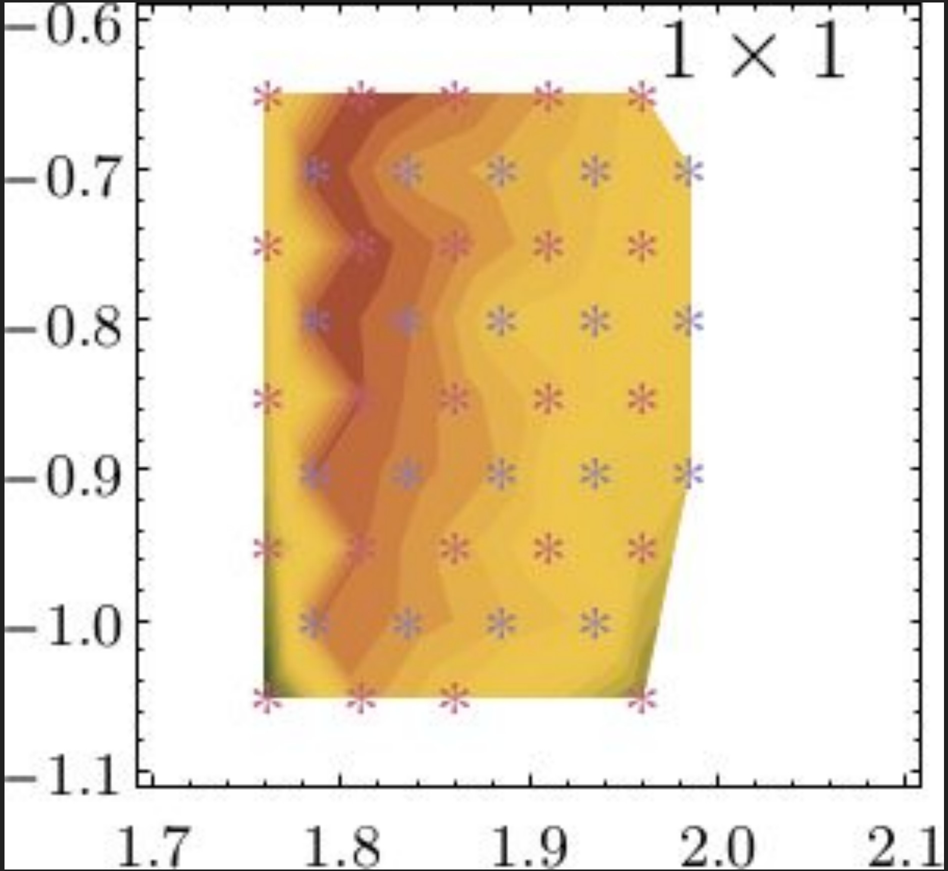
Symmetrized Loops



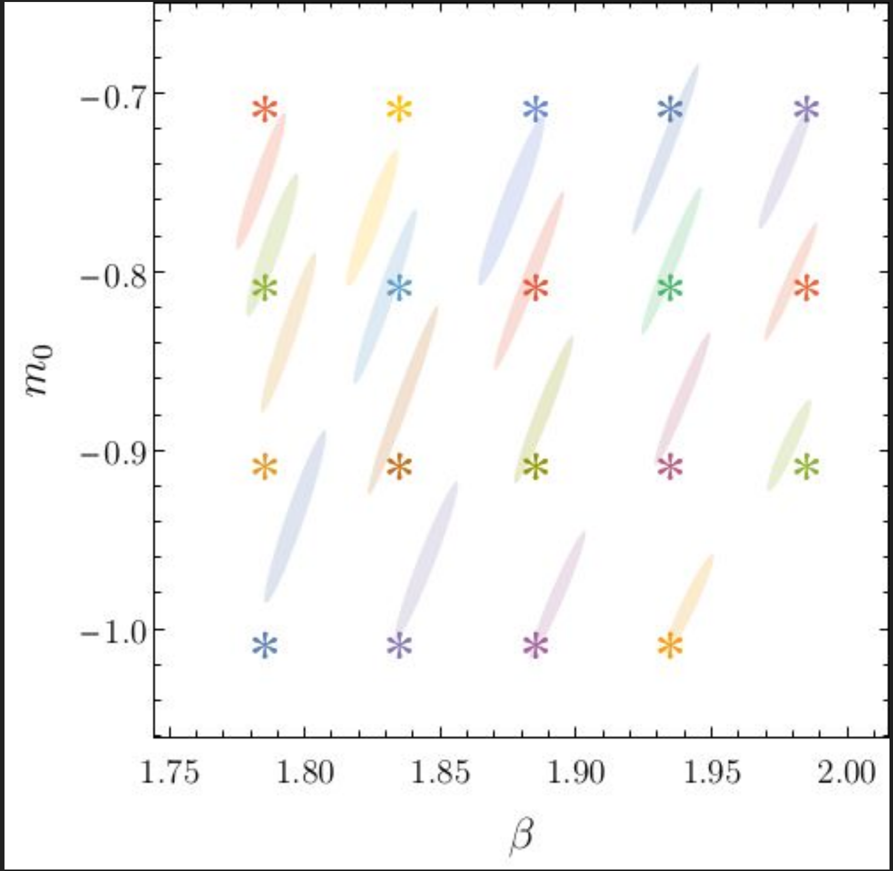
Correlated Products



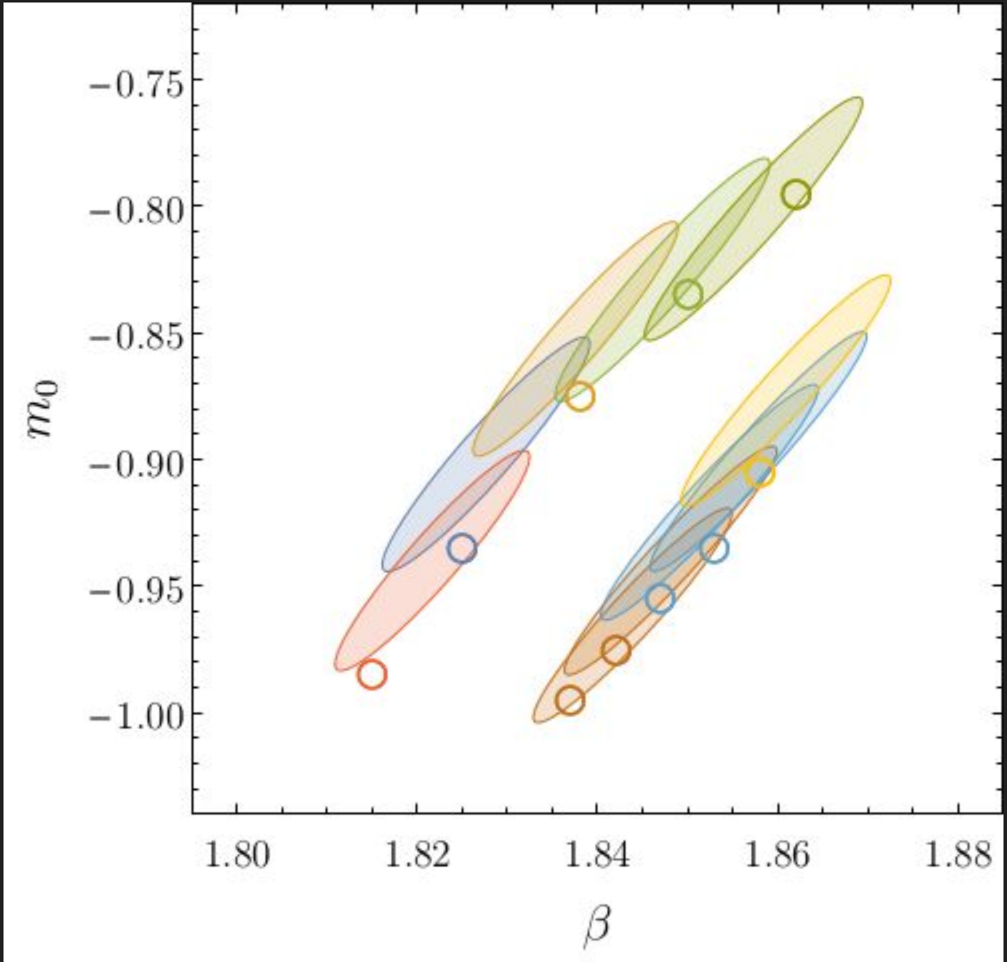
Correlated Products on Test Ensembles



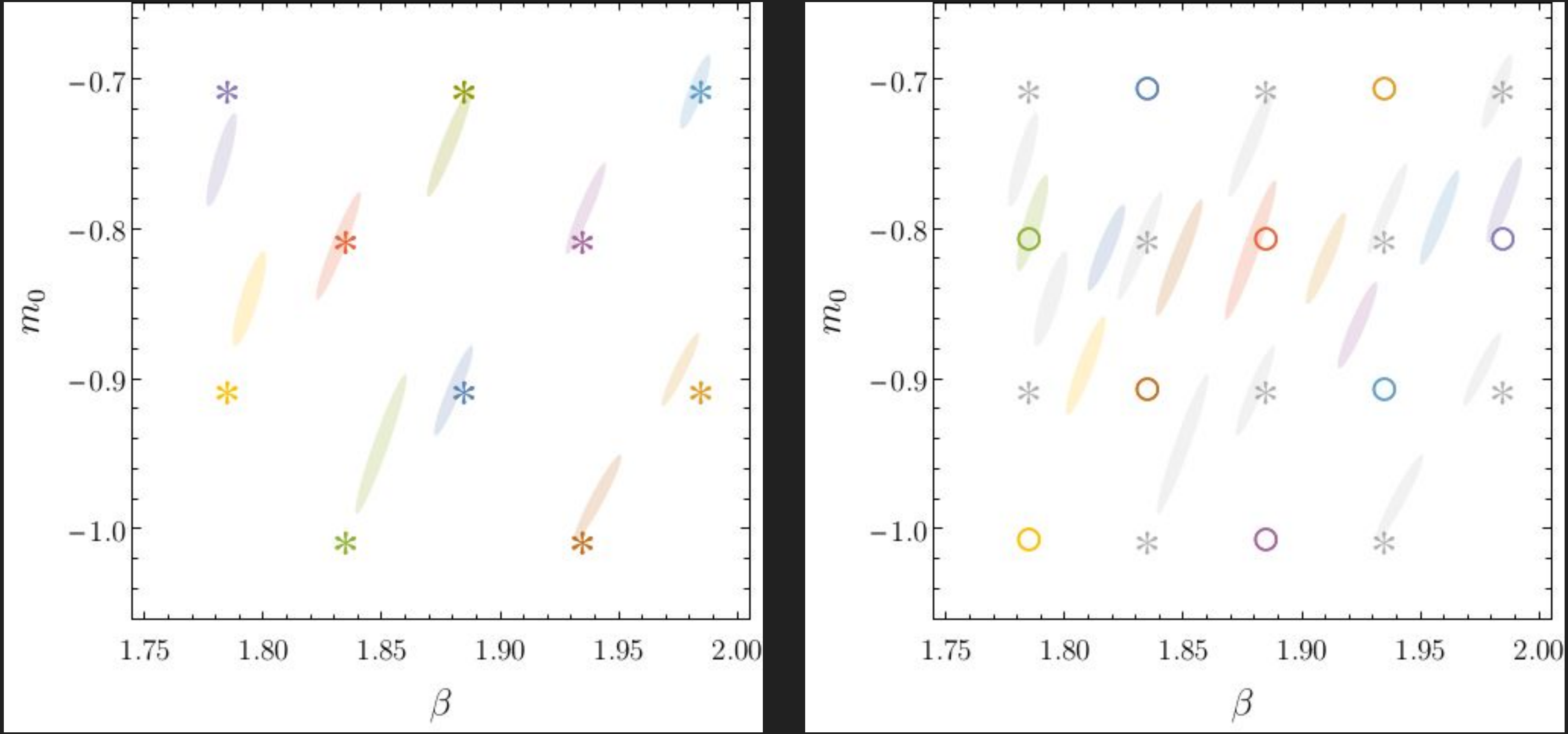
1 × 1
−0.6
−0.7
−0.8
−0.9
−1.0
−1.1
1.7
1.8
1.9
2.0
2.1



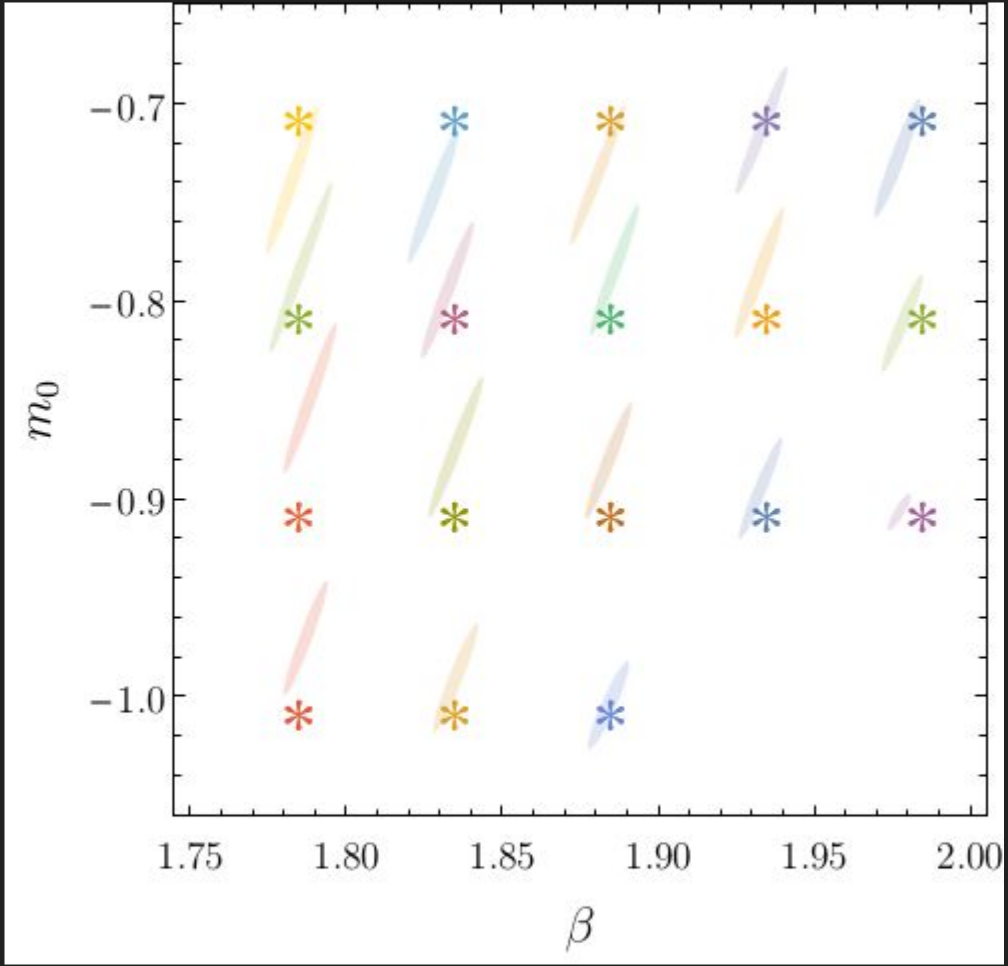
m_0
−0.7
−0.8
−0.9
−1.0
1.75
1.80
1.85
1.90
1.95
2.00
β



−0.75
−0.80
−0.85
−0.90
−0.95
−1.00
m_0
1.80
1.82
1.84
1.86
1.88
β



-0.7
-0.8
-0.9
-1.0
m_0
1.75
1.80
1.85
1.90
1.95
2.00
β
-0.7
-0.8
-0.9
-1.0
m_0
1.75
1.80
1.85
1.90
1.95
2.00
β



−0.7
−0.8
−0.9
−1.0
m_0
1.75
1.80
1.85
1.90
1.95
2.00
β

Summary

Symmetry respecting neural networks are able to solve the lattice parameter regression problem well

Fully connected networks reveal an unknown feature of longer correlation length than any observable studied

Neural networks are able to learn non-trivial features of lattice gauge field theories