EFFICIENT INFORMATION PROPAGATION IN DEEP FEED FORWARD NEURAL NETWORK.

*Mirco Milletarì, **Bambu***
CQT, 27/07/‘18

OR. . . WHAT STAT.MECH. CAN DO FOR YOUR NN

OVERVIEW

- *From the perceptron to Deep Learning*
- *A typical stat. Mech. Analysis*
- *From P to MLP : some confusing definitions*

***Expected Signal Propagation**: providing faster optimisation using Stat. Mech.*

- *Getting the dimensions right*
- *A variational model for the MLP: Hamiltonian, RG and a new "activation"*
- *Understanding **ESP**: A numerical study of the index of critical points.*

COLLABORATORS

Thiparat Chotibut

SUTD (Singapore)

Paolo E. Trevisanutto

CA2DM/NUS (Singapore)

Acknowledgements

Bill Phillips
Stirling (UK)

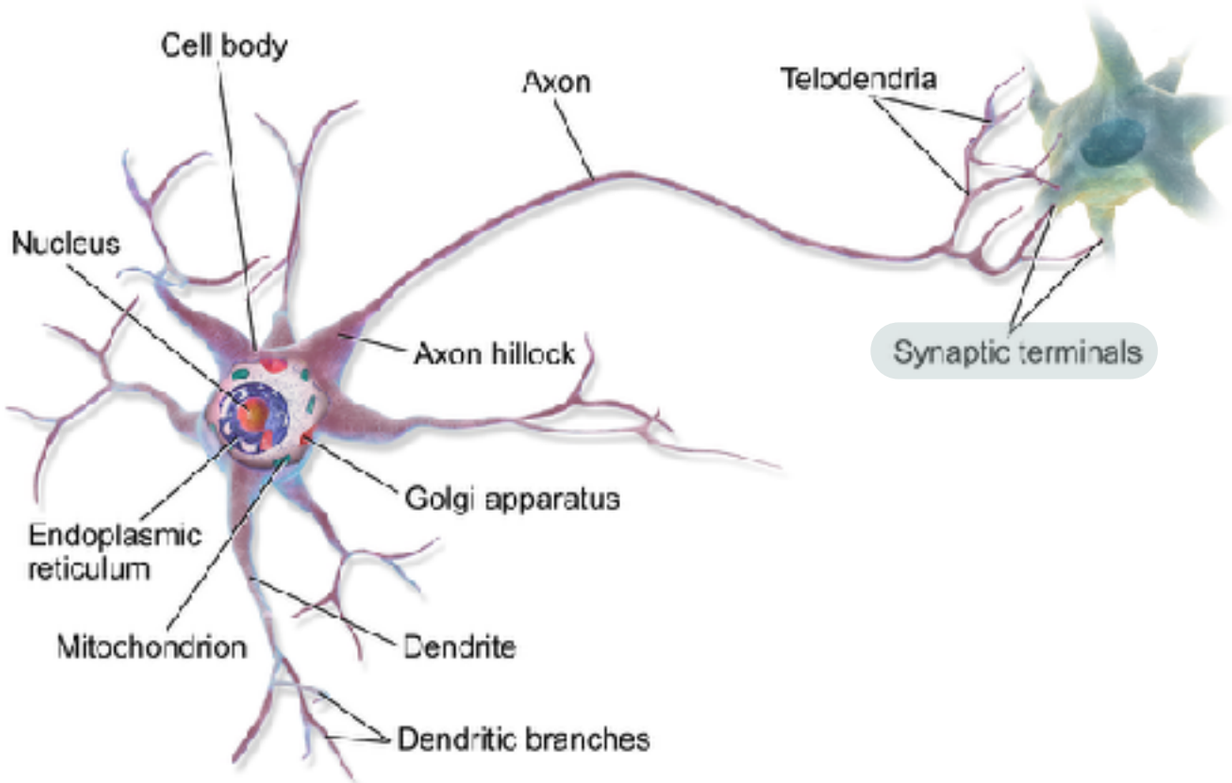
Fabio Hipolito
Aalborg (DK)

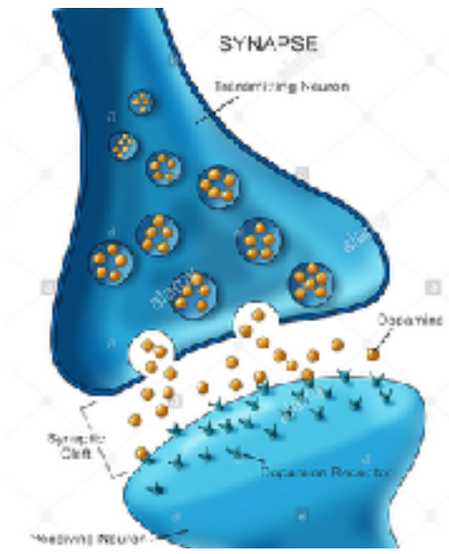
Miguel Costa
Singapore

Everyone at

bambu

FROM BIOLOGY TO PHYSICS





noise ⟶ *asynchronous dynamics*

. Variation in the number of discharged vesicles

. Variation in the number of neurotransmitter

Post synaptic potential

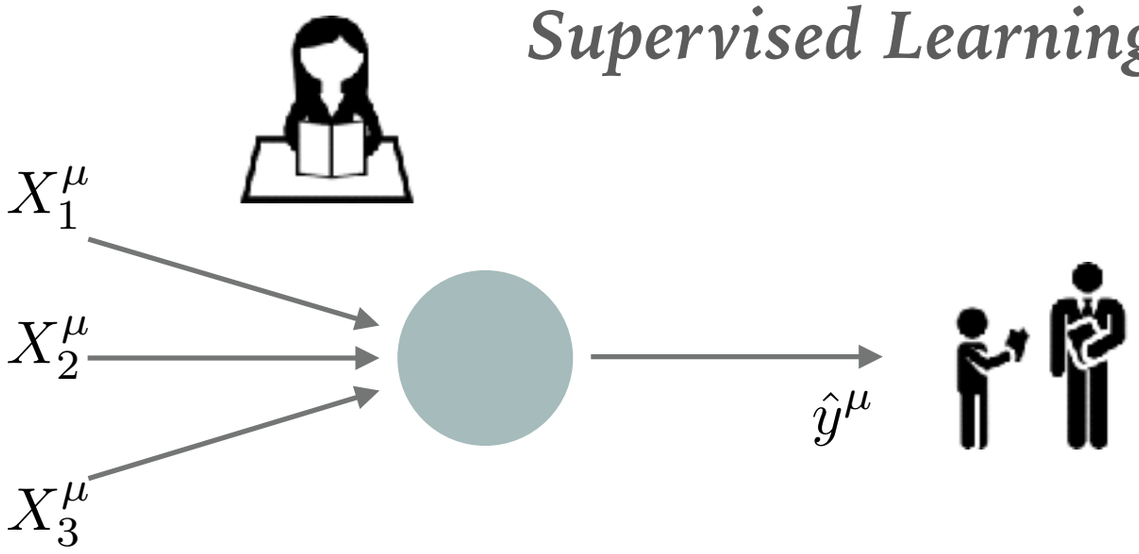
$$P(U_i) = \frac{e^{-\frac{(U_i - \bar{U}_i)^2}{2\delta^2}}}{\sqrt{2\pi\delta^2}}$$

$$P(s_i = 1) = \int_{b_i}^{\infty} dU_i P(U_i) = \frac{1}{2}\left[1 + erf\left(\frac{\bar{U}_i - b_i}{\delta\sqrt{2}}\right)\right] \simeq \sigma[\beta(\bar{U}_i - b_i)]$$

Threshold potential

See e.g. D.J. Amit, Modeling Brain Functions (1988)

EXAMPLE: LOGISTIC REGRESSION



$$\textit{input:}\quad X_i^\mu \begin{cases} i \in [1,n] & \textit{Features} \\ \mu \in [1,m] & \textit{Examples} \end{cases}$$

$$\textit{Output:} \begin{cases} y^\mu & \textit{“teacher”} \\ \hat{y}^\mu & \textit{“student”} \end{cases}$$

Learning rule: $\{\mathbf{X}^\mu, y^\mu\} \to \mathbf{J}$

Training error/Loss: $\mathscr{L} = -\frac{1}{m}\sum_{\mu=1}^{m}\{y^\mu \log(\hat{y}^\mu[\boldsymbol{\theta}, \mathbf{X}^\mu]) + (1-y^\mu)\log(1-\hat{y}^\mu[\boldsymbol{\theta}, \mathbf{X}^\mu])\},$

Generalization error

$$\mathscr{E} = \sum_{\mathbf{X},y} P(\mathbf{X}, y)\, e(\mathbf{J}; \mathbf{X}, y)$$

$\mathscr{L} \to \mathscr{E}$ *(Frequencies to probabilities)*

$$m, N \to \infty \qquad \frac{N}{m} = \alpha$$

Engel, Van den Broeck: The statistical mechanics of learning

(Z) *Perceptron Learning and the storage problem*

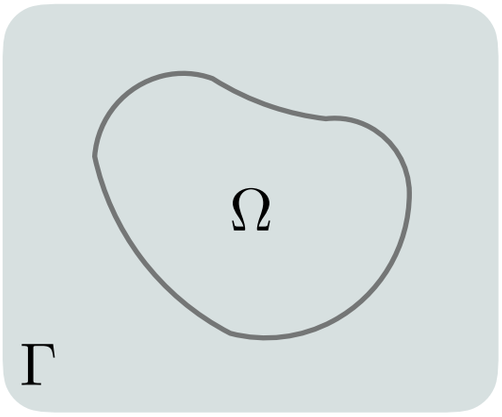
Task: reproduce a mapping we no correlations between in/out

It gives a measure of flexibility under the implementation of different in/out relations

Determine the maximal training set size m_c such that the training error is zero.

$$\frac{1}{\sqrt{n}}\, y^\mu \sum_{j=1}^{n} J_{ij}\, X_j^\mu \geq \kappa$$

Stability



Phase space volume of J satisfying the training error condition

$$\Omega(\mathbf{X}^\mu, y^\mu) = \int [\prod_{i\neq j} dJ_{ij}] \delta\left(\sum_{i\neq j} J_{ij}^2 - n\right) \prod_{\mu,i} \theta\left(\frac{y^\mu}{\sqrt{n}} \sum_{j=1}^{n} J_{ij}\, X_j^\mu - \kappa\right)$$

Typical value is given by $S = \frac{1}{n}\langle\langle \log \Omega(\mathbf{X}^\mu, y^\mu)\rangle\rangle = \frac{1}{n}\langle\langle \lim_{r\to 0} \frac{\Omega^r(\mathbf{X}^\mu, y^\mu) - 1}{r}\rangle\rangle$

E. Gardner: The space of interactions in neural network models (1988)

REPLICA SYMMETRIC SOLUTION

1. Average over $P(\mathbf{Z}^\mu) = \prod_j \left[\frac{1}{2}\delta(Z_j^\mu + 1) + \frac{1}{2}\delta(Z_j^\mu - 1)\right]$

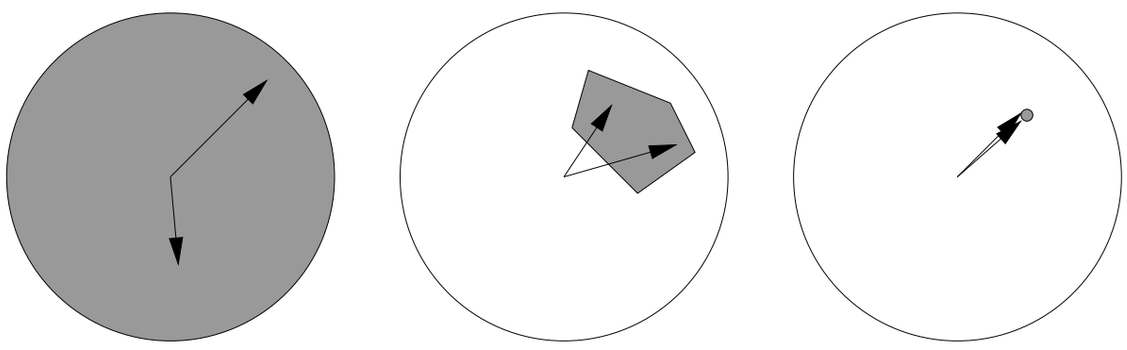
2. Introduce the order parameter $q^{ab} = \frac{1}{n}\sum_{i\neq j} J_{ij}^a J_{ij}^b$

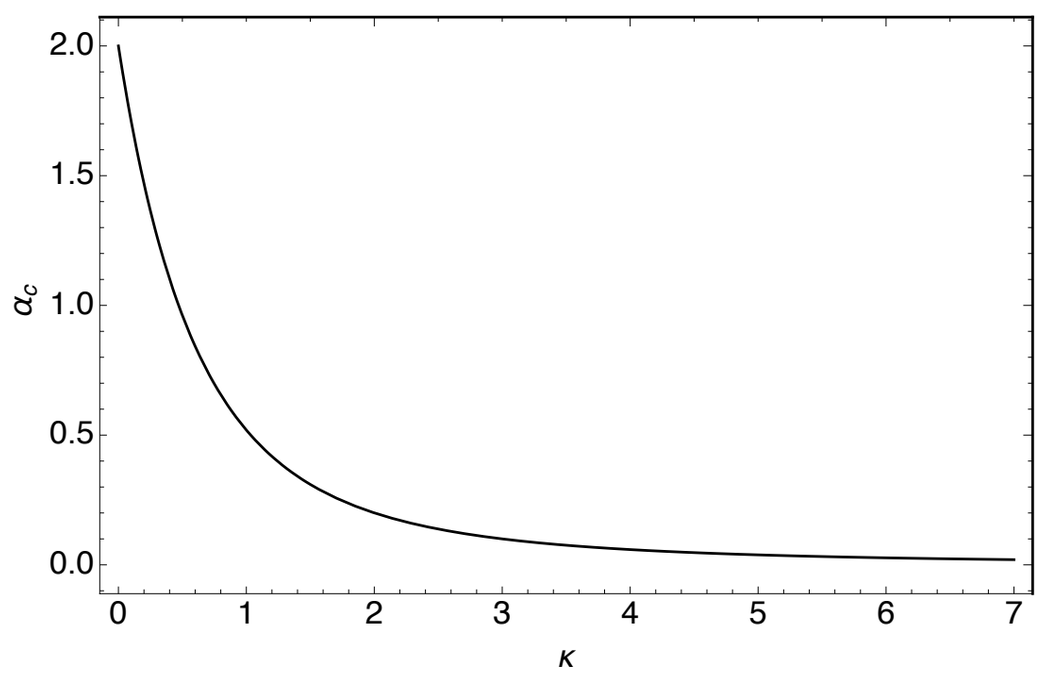
3. Find the mean field, replica symmetric solution in the limit $n \to \infty$

$$\frac{q}{1-q} = \frac{\alpha}{\pi}\int \frac{dz}{\sqrt{2\pi}} e^{-z^2/2 - \frac{(\kappa - z\sqrt{q})^2}{1-q}} erfc\left(\frac{\kappa - z\sqrt{q}}{\sqrt{1-q}}\right)^{-2}$$

a) $q \to 0, \alpha \to 0$ *every J solves the problem*

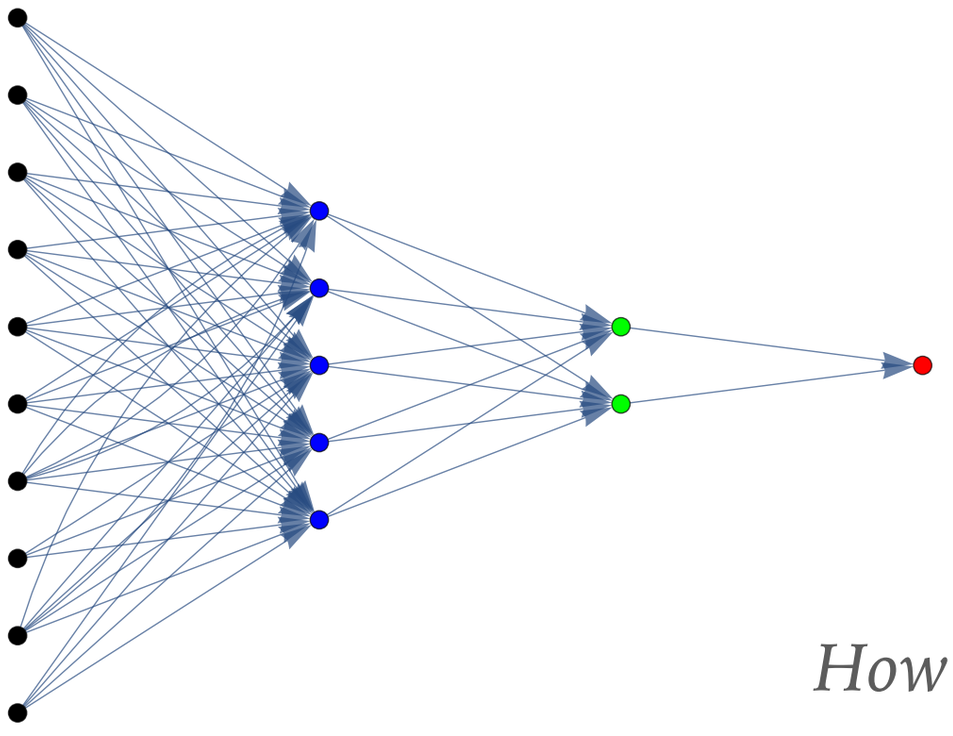
b) $q \to 1, \frac{1}{\alpha_c} = \int_{-\infty}^{\kappa} e^{-z^2/2}(\kappa - z)^2$



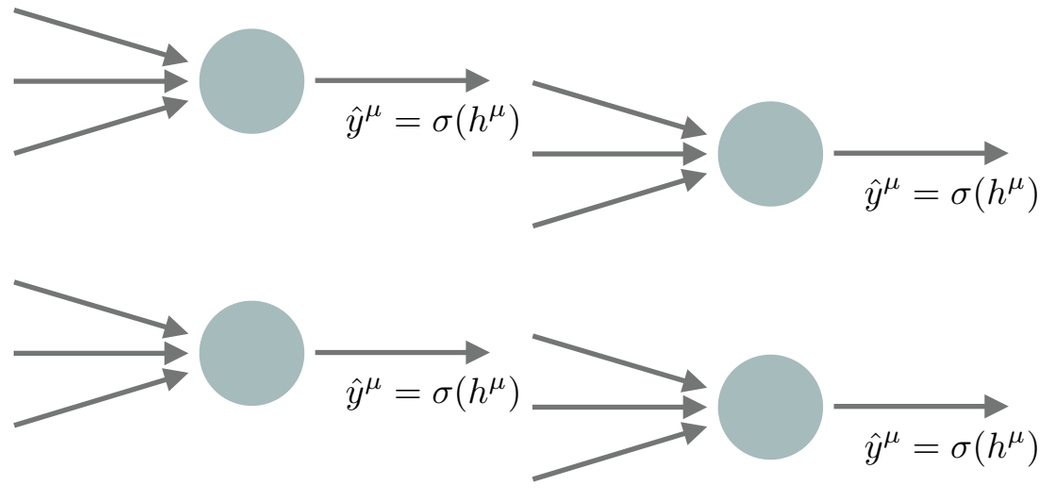


E. Gardner: The space of interactions in neural network models (1988)

FROM P TO MLP: MODERN DEEP LEARNING



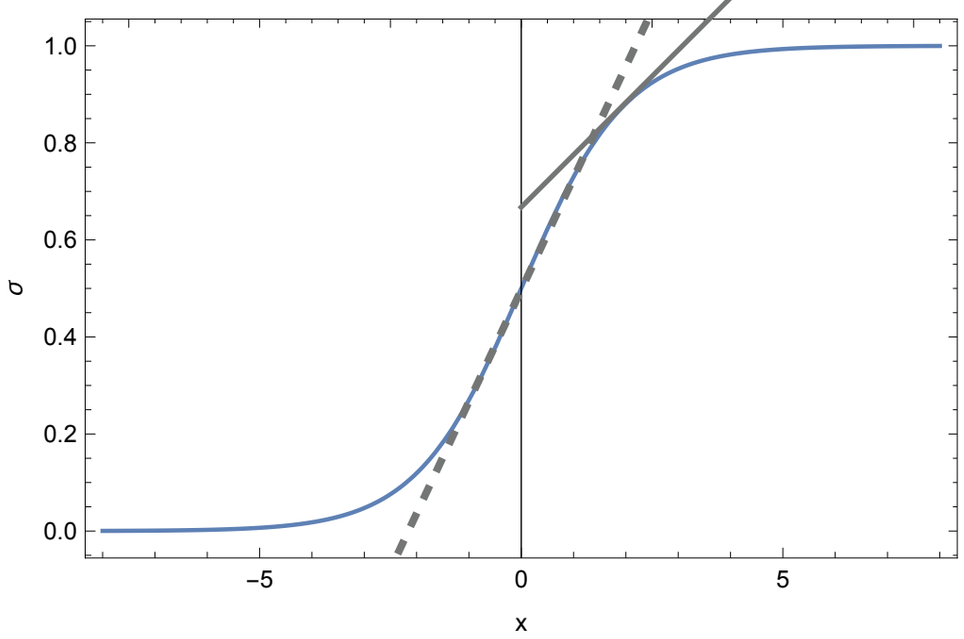
Glue together several units



How many units in each layer? How many layers?

More layers is better than more units! ⟶ *Why? Mostly an empirical rule*

*More layers comes with a cost: the **vanishing gradient problem**.*



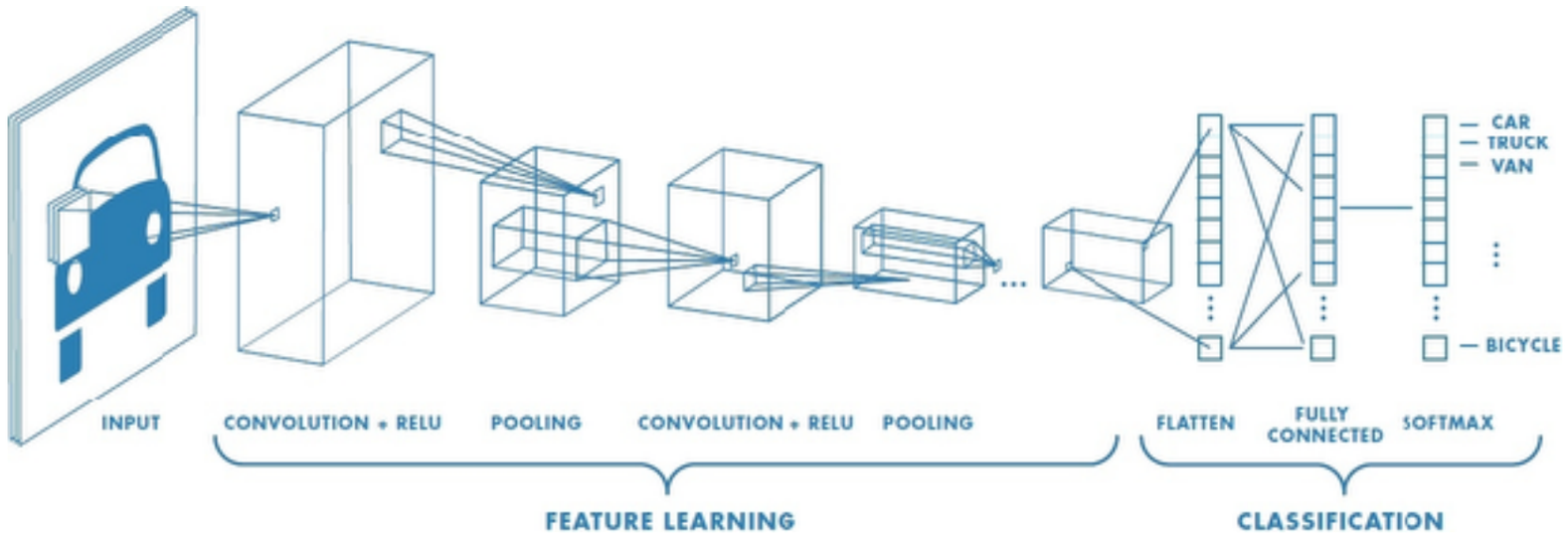
Considered industry standard in DL

$$ReLu = \max\{x, 0\}$$

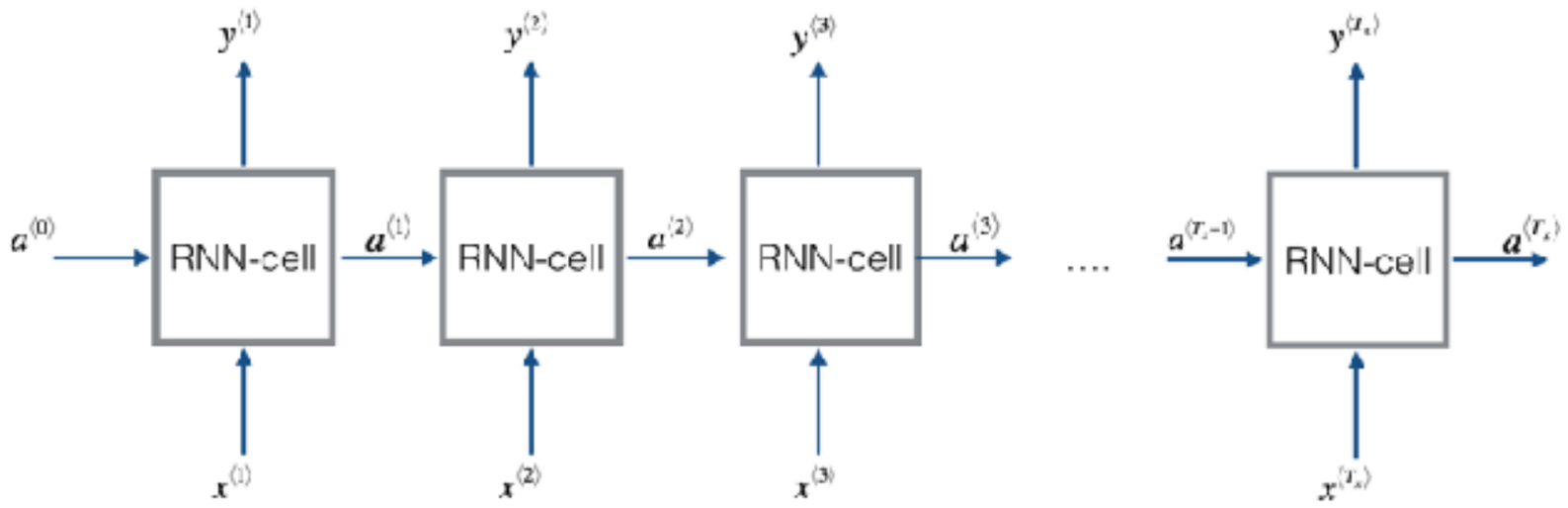
A zoo of other activations exists, but ReLu is the most flexible.

SUCCESSFUL ARCHITECTURES BASED ON THE FORWARD PASS

Convolutional



Recurrent



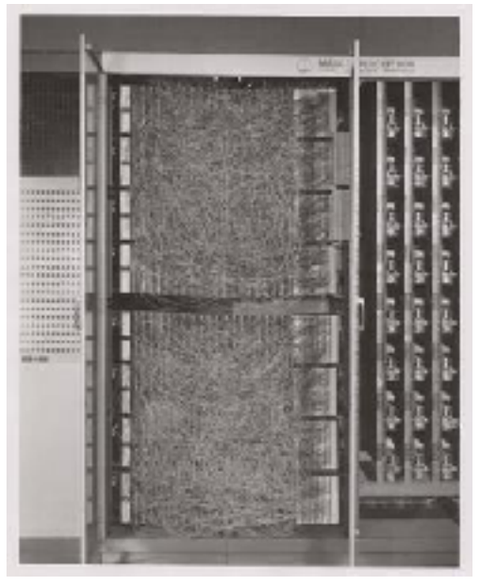
Next I will reconsider the Forward Pass in a general setting

EXPECTED SIGNAL PROPAGATION

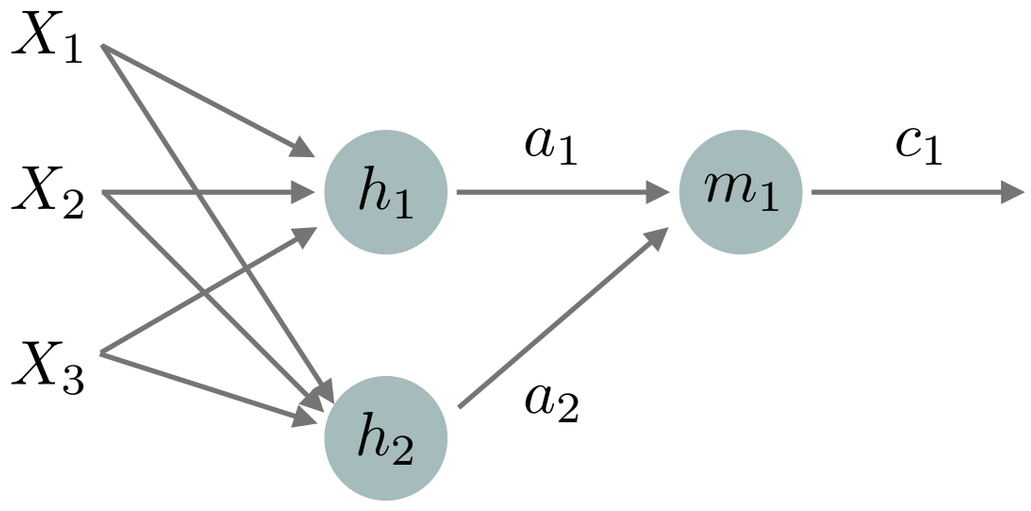
M.M., T. Chotibut and P.E. Trevisanutto

arXiv:1805.08786

FROM P TO MLP: GETTING THE DIMENSIONS RIGHT



In the Mark 1 perceptron the hardware is a NN. Each unit takes a physical signal as an input



Follow the standard recipe of DL:

1. *The information is encoded in an electric signal, therefore* $[X_i] = C$

2. *The unit adds up the signals* $h_i = \sum_j w_{ij} X_j$ → $[h_i] = C$

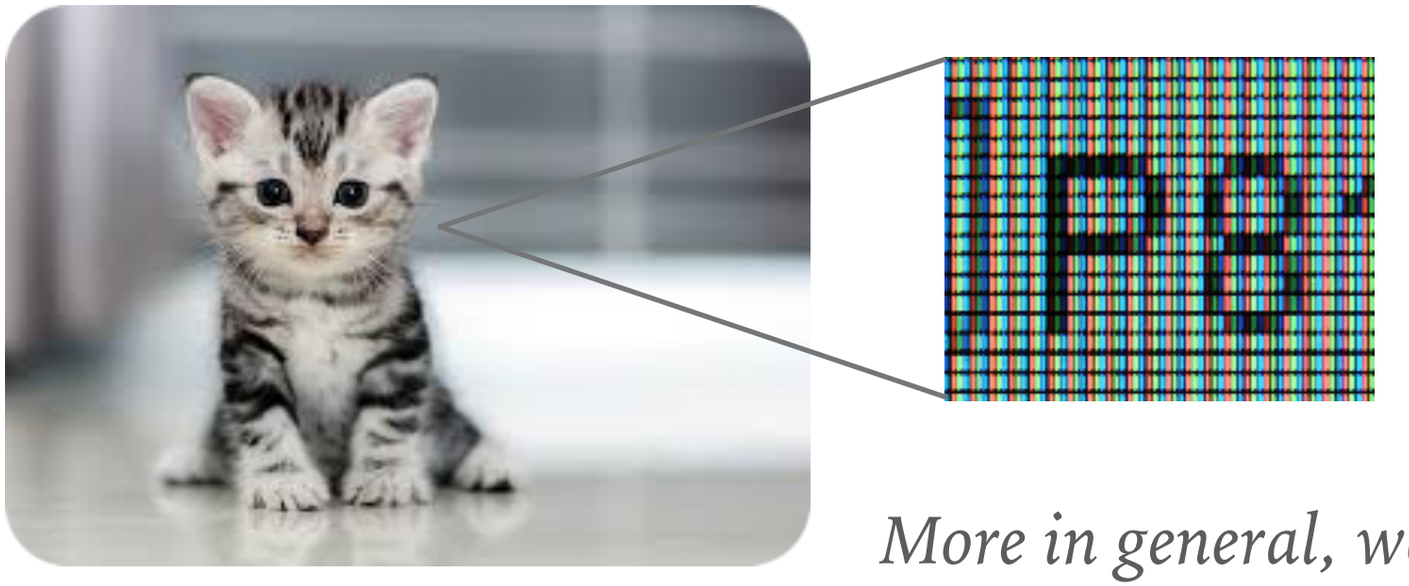
3. *Output:* $a_i = \sigma(\beta h_i)$ *Dimensionless* → $[\beta] = 1/h$

4. *Repeat 1-3:* $m_1 = \sum_j \omega_{1j} a_j$ → *Dimensionless* ← $c_1 = \sigma(\beta m_1)$

→ $[\beta]$ *Dimensionless*

Dimensional mismatch!

OK, BUT MY NN IS SOFTWARE BASED...



Values of each pixel correspond to voltage values.

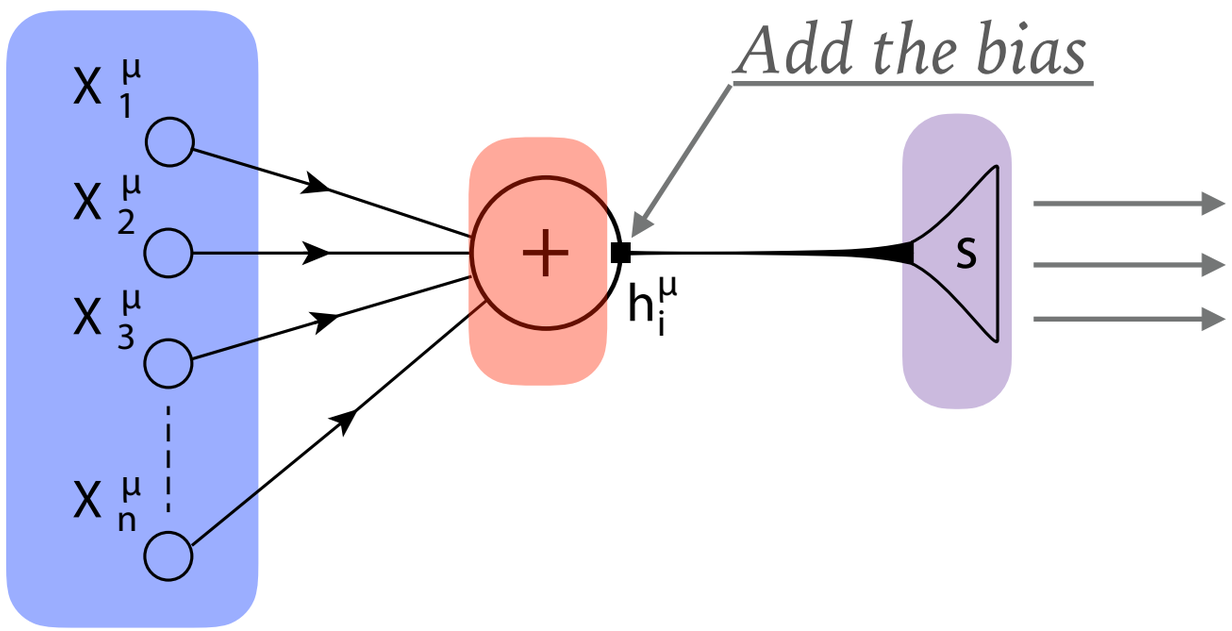
*More in general, we can measure information in **bits***

TAKE HOME POINTS

	$\sigma(\beta x) = \frac{1}{1+e^{-\beta x}}$	$\tanh(\beta x/2) = [1 - 2\,\sigma(\beta x)]$	$ReLu = \max(x, 0)$
Domain:	$[0, 1]$	$[-1, 1]$	$[0, \infty)$
Info:	*Distribution of x*	*Distribution of x*	*Value of x*

COMMUNICATION CHANNEL AND COARSE GRAINING

$\mu \in [1, m]$ *Example index*

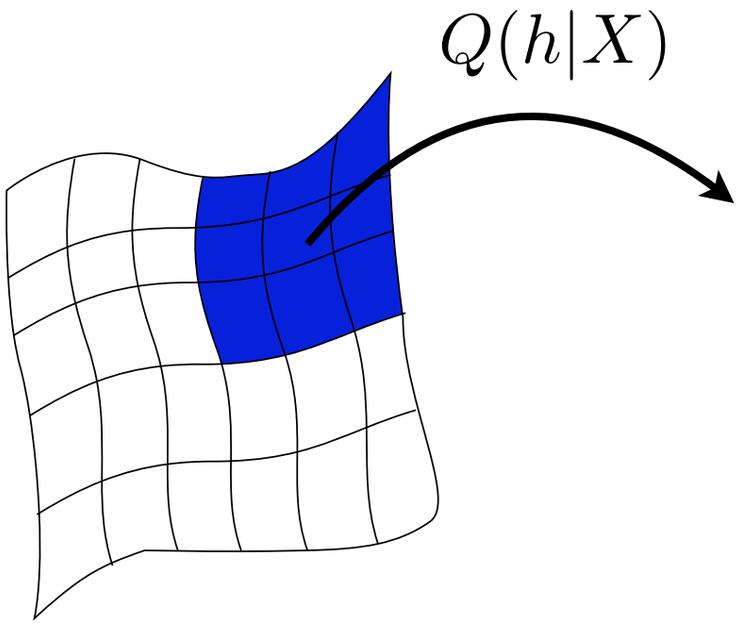


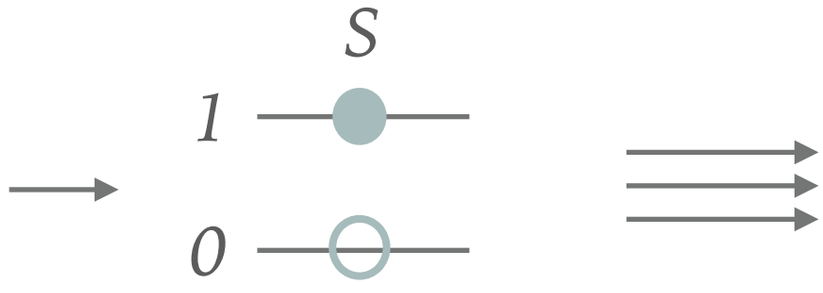
In a neuron, the result of the computation is encoded in the neurotransmitters.

$$P(\mathbf{h}) = \int d\mathbf{X}\, Q(\mathbf{h}|\mathbf{X})\, P(\mathbf{X})$$

$$P(\mathbf{s}) = \int d\mathbf{h}\, P(\mathbf{s}|\mathbf{h})\, P(\mathbf{h})$$

Associated phase space





MAX ENTROPY AND DATA DISTRIBUTION

Entropy functional in the absence of lateral connections

$$F = -\sum_{\mathbf{s}} P(\mathbf{s}) \log P(\mathbf{s}) + \eta \left(\sum_{\mathbf{s}} P(\mathbf{s}) - 1 \right) + \sum_i \lambda_i \left(m_i - \sum_{\mathbf{s}} s_i P(\mathbf{s}) \right)$$

See e.g. H. C. Nguyen, R. Zecchina, and J. Berg. Advances in Physics, 66:3, pp. 197-261, 2017.

$$\frac{\delta F}{\delta P(\mathbf{s})} = 0$$

$$\frac{\delta F}{\delta \eta} = 0$$

$$\frac{\delta F}{\delta \lambda_i} = 0$$

The simplest possible "Ising" model

$$P(\mathbf{s}|\mathbf{h}) = \frac{1}{Z} e^{\sum_i \beta_i \, s_i \, h_i}$$

$$Z[\mathbf{h}] = \prod_i \sum_{s_i = \{0,1\}} e^{\beta_i \, s_i \, h_i} = \prod_i \left(1 + e^{\beta_i \, h_i}\right)$$

What about the input distribution?

$$P(\mathbf{X}) = \frac{1}{m} \sum_{\mu=1}^{m} \delta_{\mathbf{x},\mathbf{x}^\mu}$$

Empirical distribution

$$P(\mathbf{X}) = \frac{1}{N} e^{-\frac{1}{2} X_i^\mu \Lambda_{\mu\nu}^{ij} X_j^\nu}$$

A priori distribution

EXPECTED PROPAGATION AND ACTIVATIONS

Using the linear transition function

$$Q(\mathbf{h}|\mathbf{X}) = \prod_i \delta\left(h_i - \sum_j w_{ij}\, X_j - b_i\right)$$

First step of the Renormalization group

$$P(\mathbf{h}) = \int d\mathbf{X}\, Q(\mathbf{h}|\mathbf{X})\, P(\mathbf{X})$$

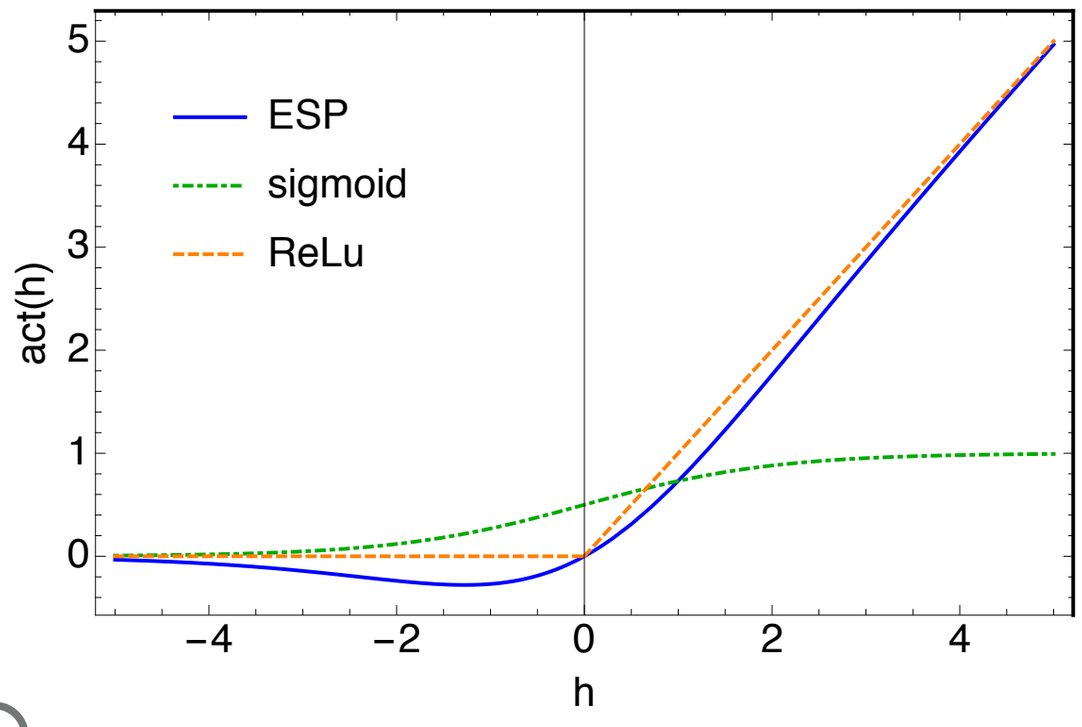
S.K. Ma, Modern Theory of critical Phenomena
M. Cassandro and G.Jona-Lasinio, Critical Point Behaviour and Probability theory, Advances in Physics, 27:6, 913-941 (1978)

Expected value of the gate (synaptic state)

$$\langle s_i \rangle = \frac{1}{\beta_i}\frac{\partial}{\partial b_i}\log Z_i = \prod_\mu \frac{1}{1+e^{-\beta_i\, h_i^\mu}} \equiv \prod_\mu \sigma(\beta_i\, h_i^\mu),$$

Expected value of the processed information

$$a_i^\mu \equiv \langle h_i^\mu \rangle = \frac{\partial}{\partial \beta_i}\log Z_{i,\mu} \equiv h_i^\mu \langle s_i \rangle = h_i^\mu\, \sigma(\beta_i\, h_i^\mu)$$

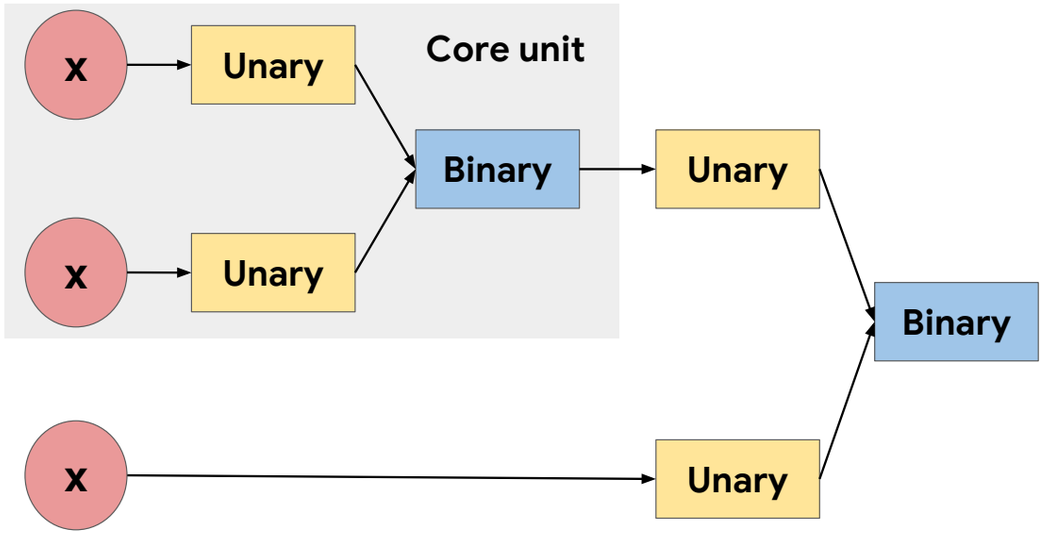


Expected Signal Propagation

THE "BRUTE FORCE" APPROACH

A Google Brain group obtained the same result using a search algo trained with reinforcement learning: Ramachandran et al. ***arXiv:1710.0594*** *(2017). See also Elfwing et al.* ***arXiv:1702.03118***

Search Space



Examples

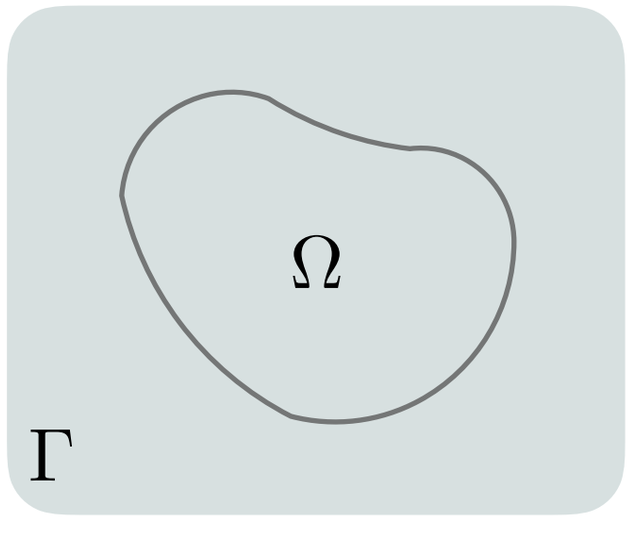
$$u(x) = x^2$$

$$u(x) = \sigma(x)$$

$$b(x_1, x_2) = x_1 \times x_2$$

$$b(x_1, x_2) = e^{-(x_1 - x_2)^2}$$

Sampling



The configurational space quickly increases in dimension.
The computational complexity increases rapidly.

Best solution is the one taking more space!

IMPLEMENTATION

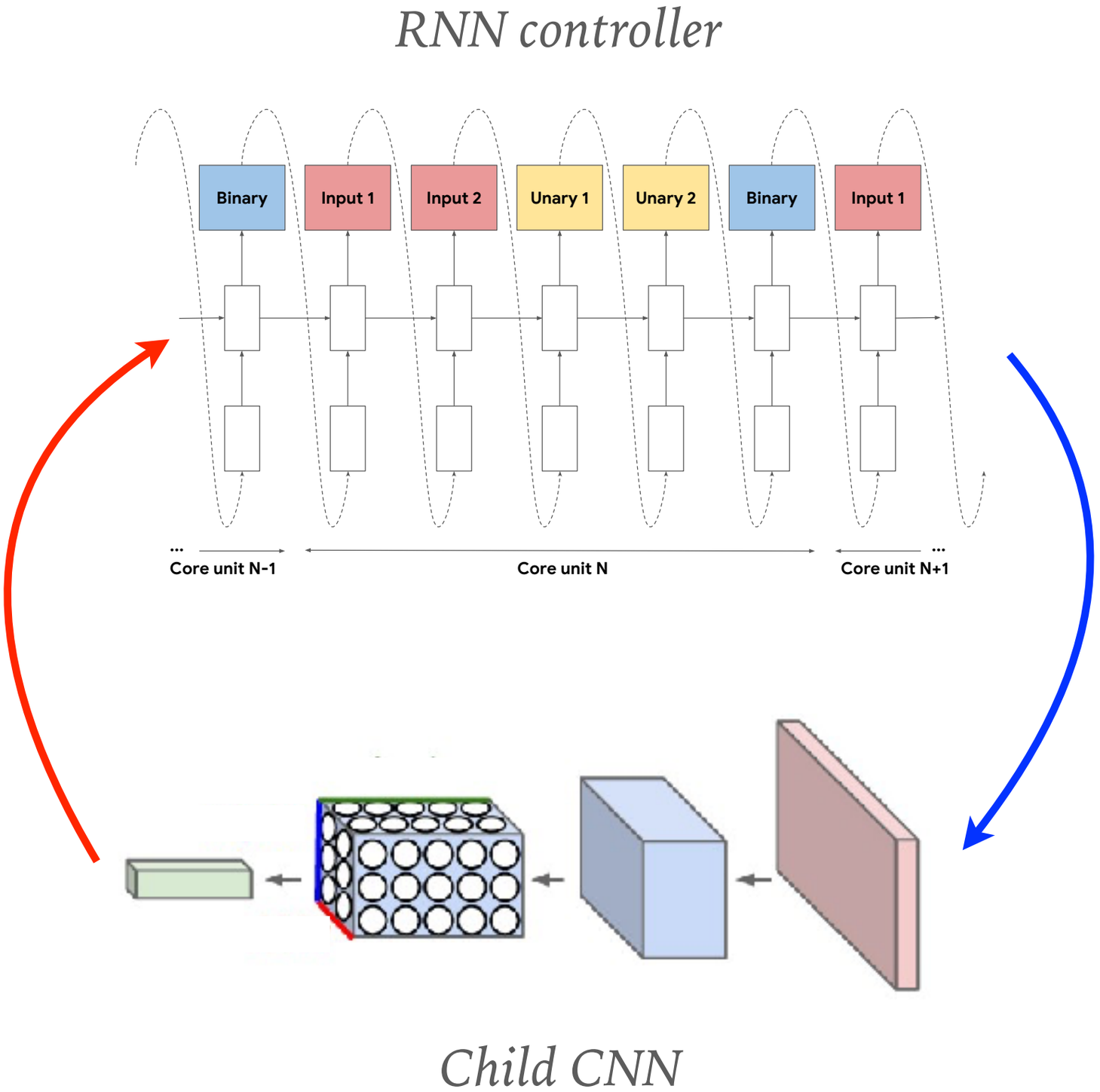
1) ***Predict*** *one component of activation. Feed the prediction back in at the following time step.*

Out: *Candidate activation*

2) ***Train*** child network.

Out: *A*ccuracy score. Use as A reward in a **RL** algo to train the controller

~800 Titan X GPUs !



B. Zoph, Q. V. Le, arXiv:1611.01578 (2016).

https://github.com/Neoanarika/Searching-for-activation-functions

MEANING

Ramachandran et al. found all functions of the form $a(x) = x\,f(x)$ *to outperform Heuristic ones in the literature. Among them,* ***ESP*** *was the best performant.*

Function	**RN**	**WRN**	**DN**
ReLU $[\max(x, 0)]$	74.2	77.8	83.7
$x \cdot \sigma(\beta x)$	75.1	78.0	83.9
$\max(x, \sigma(x))$	74.8	78.6	84.2
$\cos(x) - x$	75.2	76.6	81.8
$\min(x, \sin(x))$	73.4	77.1	74.3
$(\tan^{-1}(x))^2 - x$	75.2	76.7	83.1
$\max(x, \tanh(x))$	74.8	76.0	78.6
$\mathrm{sinc}(x) + x$	66.1	68.3	67.9
$x \cdot (\sinh^{-1}(x))^2$	52.8	70.6	68.1

Table 2: CIFAR-100 accuracy.

Model	**ResNet**	**WRN**	**DenseNet**
LReLU	74.2	78.0	83.3
PReLU	74.5	77.3	81.5
Softplus	76.0	78.4	83.7
ELU	75.0	76.0	80.6
SELU	73.2	74.3	80.8
GELU	74.7	78.0	83.8
ReLU	74.2	77.8	83.7
Swish-1	75.1	78.5	83.8
Swish	75.1	78.0	83.9

Table 5: CIFAR-100 accuracy.

In the noiseless limit

$$\lim_{\beta_i \to \infty} \langle h_i^\mu \rangle = h_i^\mu \, \theta(h_i^\mu) \equiv \max\{\mathrm{h}_\mathrm{i}^\mu, 0\} \equiv \mathrm{ReLu}$$

Physically, the FFN corresponds to a system in local equilibrium

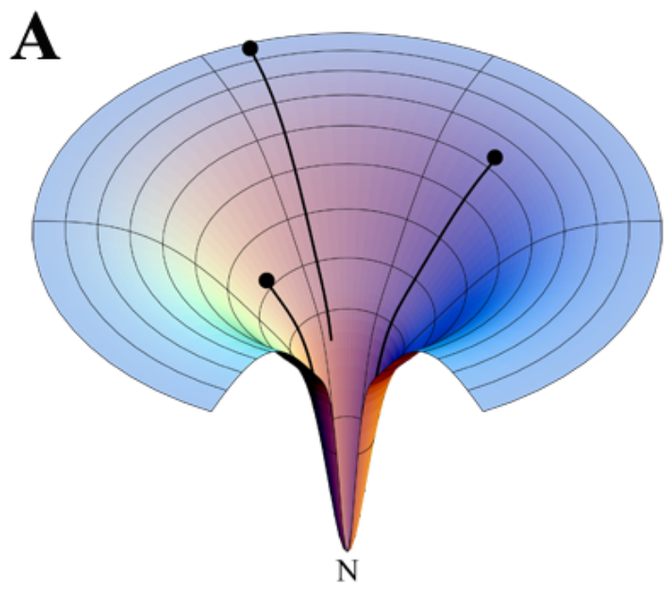
BACK-PROPAGATION AND VANISHING GRADIENTS I

Consider the output layer "L":

$$\frac{\partial \mathscr{L}}{\partial \mathbf{w}_L} = \frac{1}{m} \sum_{\mu=1}^{m} \left[\mathbf{e}^{\mu} \, \mathbf{g}(\mathbf{h}_L^{\mu}) \right] \mathbf{a}_{L-1}$$

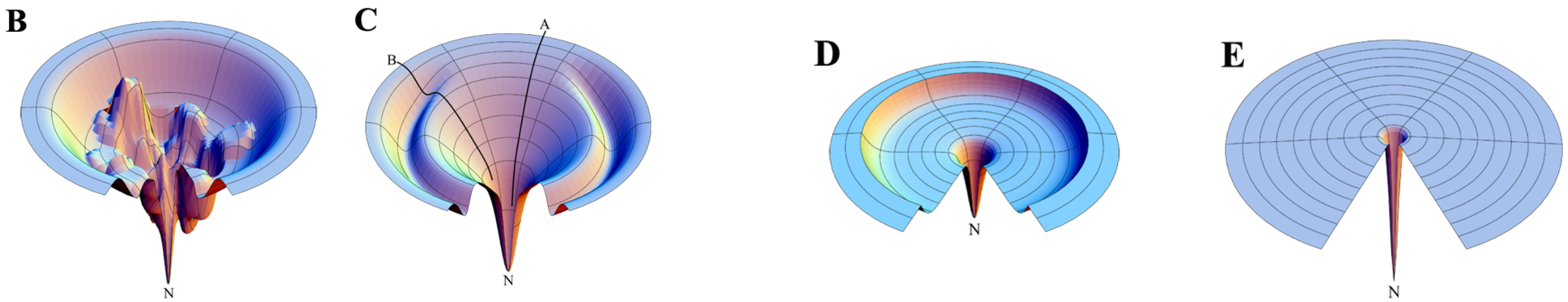
For a Linear Network g=1 and the optima are located at $\mathbf{e}^{\mu} = 0 \longrightarrow \hat{y}^{\mu} = y^{\mu}$

Usually ends up with a smooth energy manifold



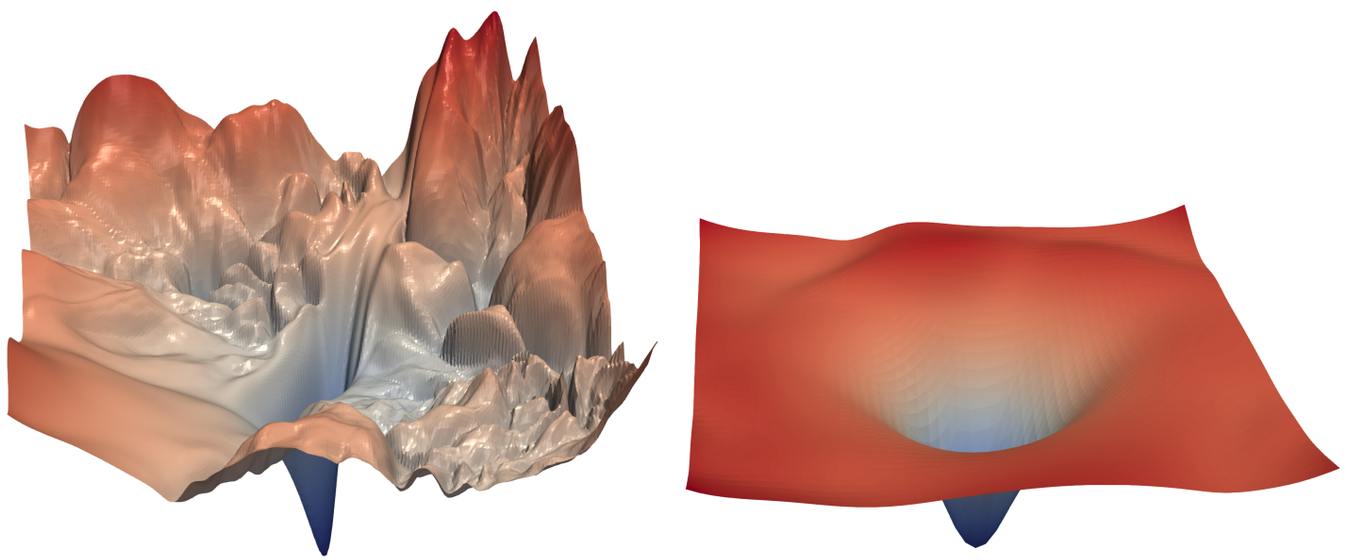
BACK-PROPAGATION AND VANISHING GRADIENTS II

In a non-linear network, one can also have $g(\mathbf{h}^{\mu}) = 0$*, we are adding optima.*



Rugged *Moat* *Champagne* *Golf course*

Dill & Chan, Nat Struct. Bill 4, 10-19,1997



(a) without skip connections (b) with skip connections

Loss surface of ResNet-56

From Li et al. ***arXiv:1712.09913***

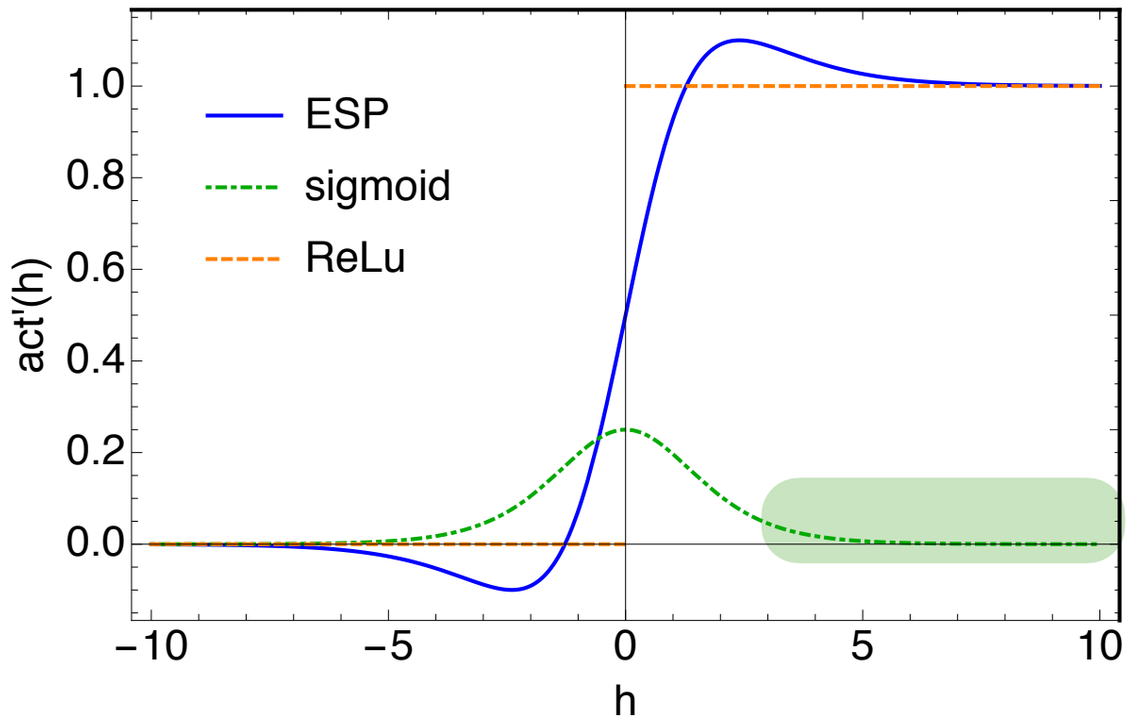
BACK-PROPAGATION AND VANISHING GRADIENTS III

A paradox

$$\sigma(\beta_i\, h_i) \equiv P(s_i = 1|\bar{x}_i) \simeq 1$$

$$\Rightarrow \quad g(h) \simeq 0$$

The back propagated signal is small!



On the other hand, both ESP and ReLu give $g(h) \simeq 1$

ReLu corresponds to a purely excitatory network, i.e. no unlearning

$$\frac{\partial \mathscr{L}}{\partial \mathbf{w}_L} \propto [\mathbf{e}^\mu\, \mathbf{g}(\mathbf{h}_L^\mu)]$$

Tishby et al. show two phases of GD: representation learning And empirical error minimization. *arXiv:1503.02406v1 , arXiv:1703.00810*

OPTIMIZATION WITH ESP: GEOMETRY OF THE LOSS FUNCTION

Given a set of parameters $\boldsymbol{\theta}_{\alpha}^{[l]} = (\mathbf{w}^{[l]}, \mathbf{b}^{[l]}, \boldsymbol{\beta}^{[l]})$*,one needs to study the Hessian*

$$H_{\alpha,\beta}^{l,l'} = \frac{\partial^2 \mathscr{L}}{\partial \theta_{\alpha}^{[l]} \, \partial \theta_{\beta}^{[l']}}$$

The eigenvalues of the Hessian determine the nature of the optima

Indices at critical points

$$\alpha = \frac{1}{N} \sum_j I(\lambda_j < 0)$$

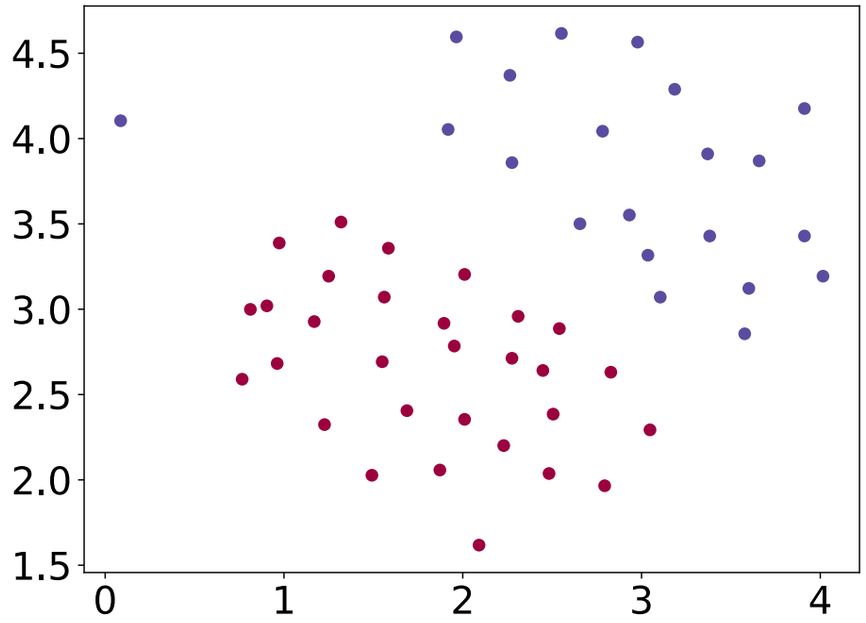
Relative fraction of negative eigenvalues. Measures Descent/Ascent directions.

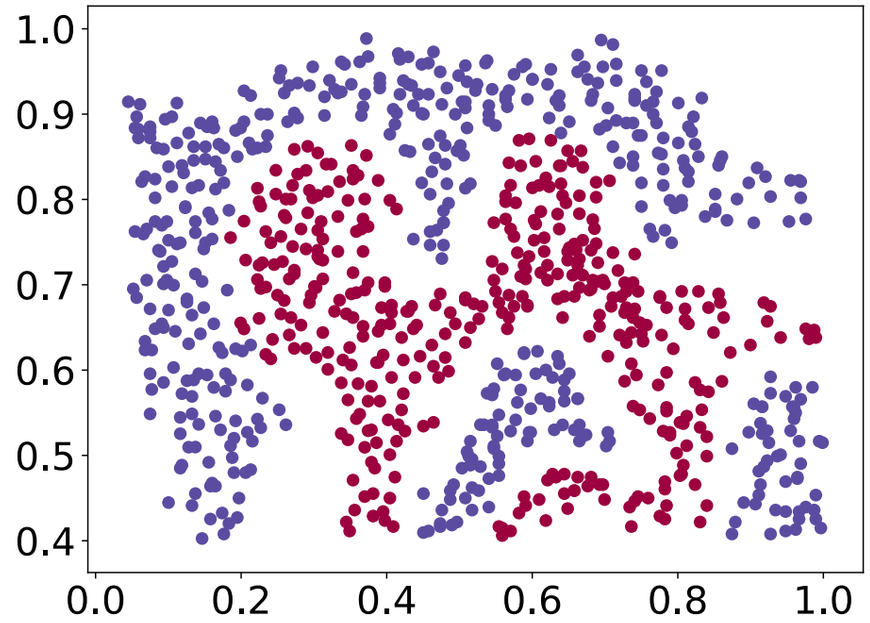
$$\gamma = \frac{1}{N} \sum_j I(\lambda_j = 0)$$

Relative fraction of zero eigenvalues. Measures null Directions (plateaus of the energy manifold).

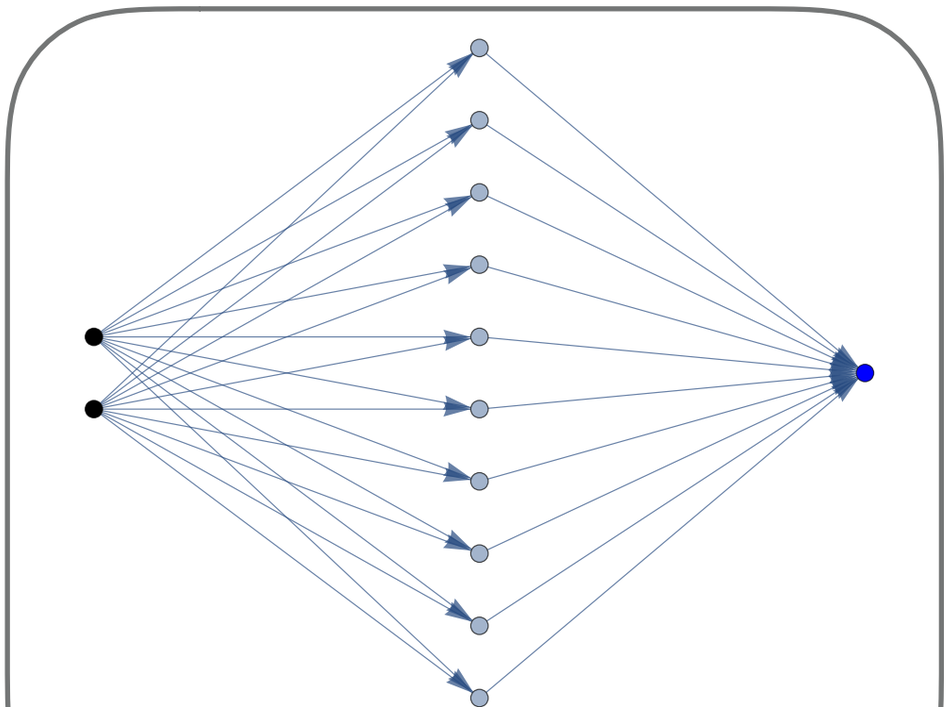
See also J. Pennington et al. Geometry of Neural Network Loss Surfaces via Random Matrix Theory
Y. N. Dauphin et al. Identifying and attacking the saddle point problem in high-dimensional non- convex optimization

RESULTS I: BINARY CLASSIFICATION

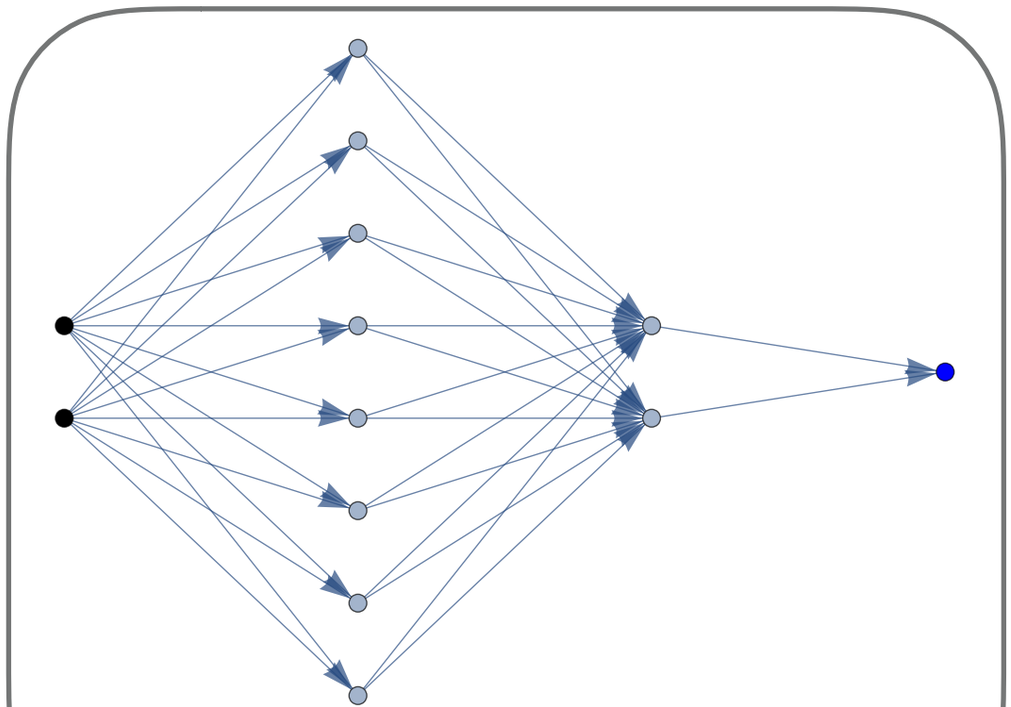




"cross entropy" loss: $$\mathscr{L} = -\frac{1}{m}\sum_{\mu=1}^{m}\{y^{\mu}\ \log(\hat{y}^{\mu}[\boldsymbol{\theta}, \mathbf{X}^{\mu}]) + (1-y^{\mu})\ \log(1-\hat{y}^{\mu}[\boldsymbol{\theta}, \mathbf{X}^{\mu}])\}\,,$$



2-10-1



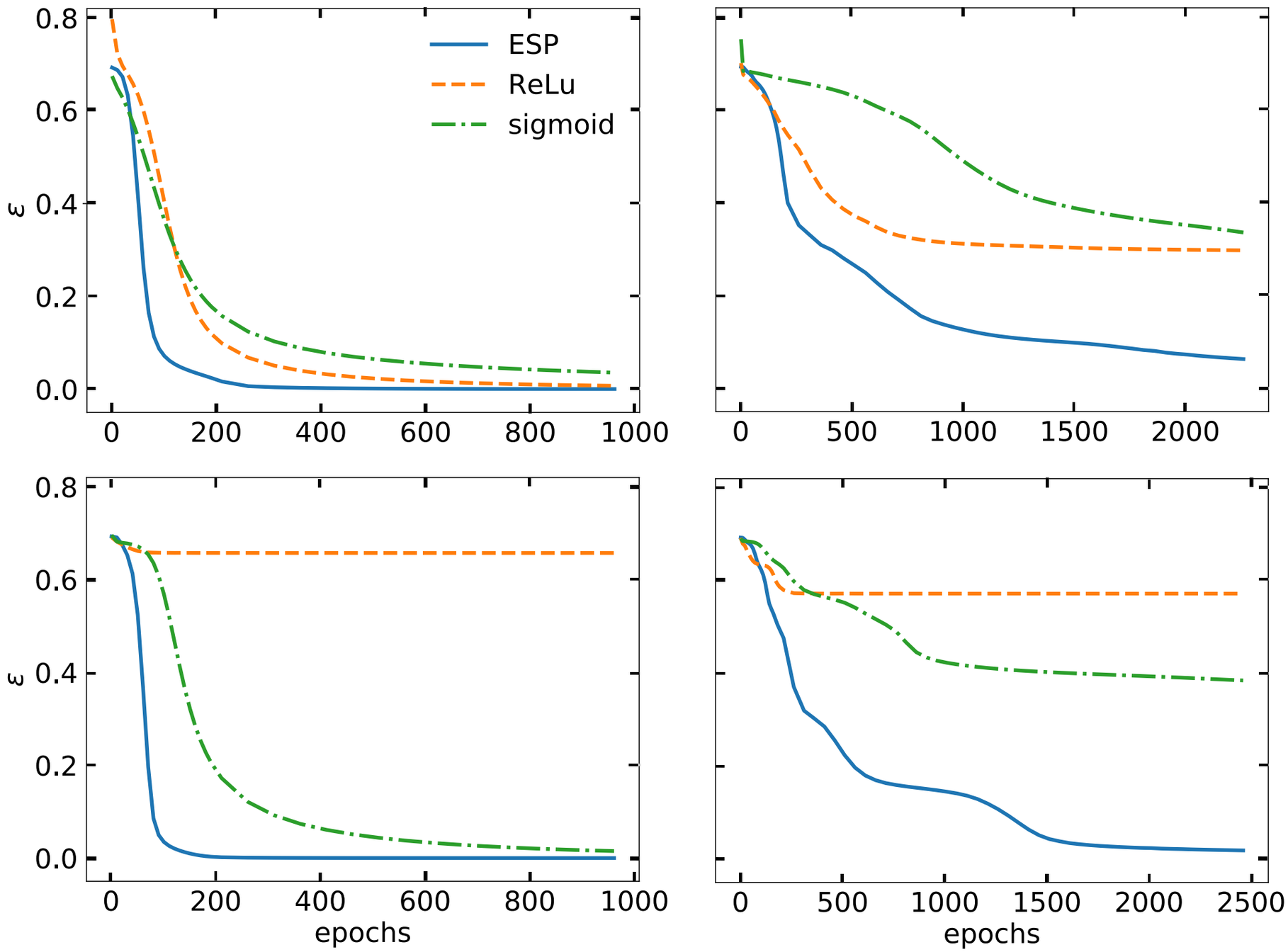
2-8-2-1

LOSS (ENERGY) AS A FUNCTION OF TRAINING EPOCHS

2-10-1 *2-8-2–1*

Linear

Non-linear



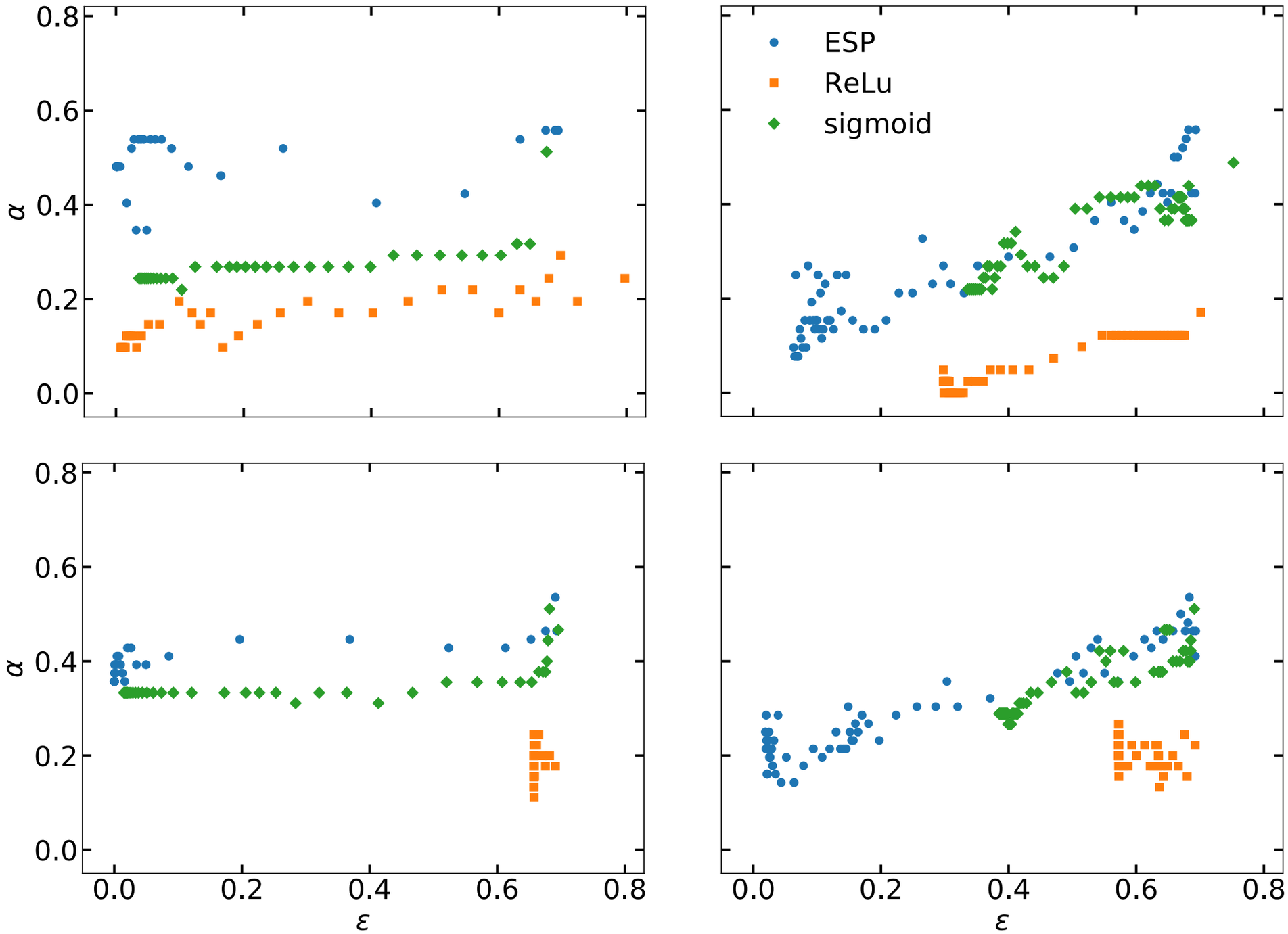
ALPHA INDEX AS A FUNCTION OF ENERGY

2-10-1

2-8-2–1

Linear

Non-linear



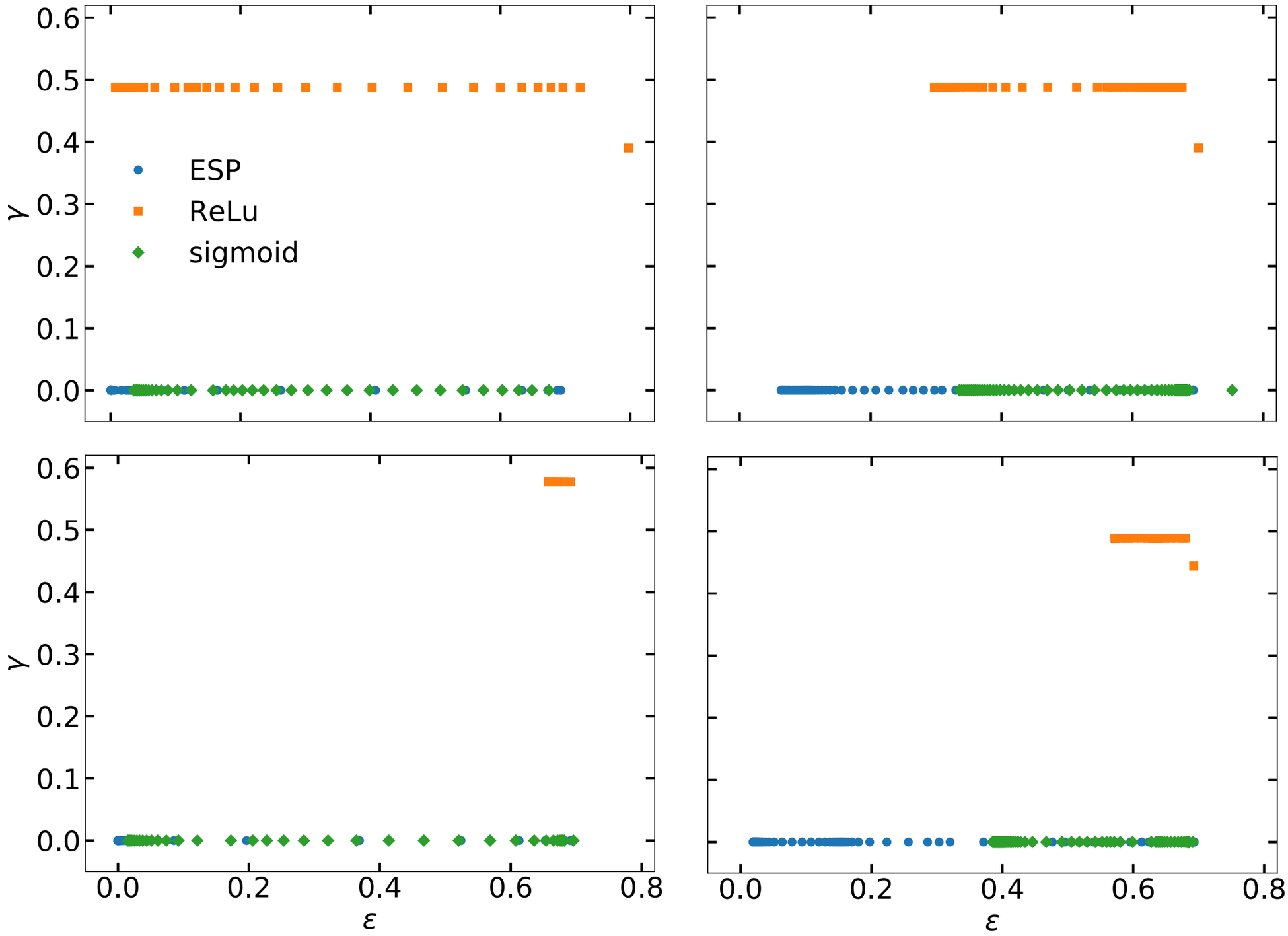
GAMMA INDEX AS A FUNCTION OF ENERGY

2-10-1

2-8-2–1

Linear

Non-linear



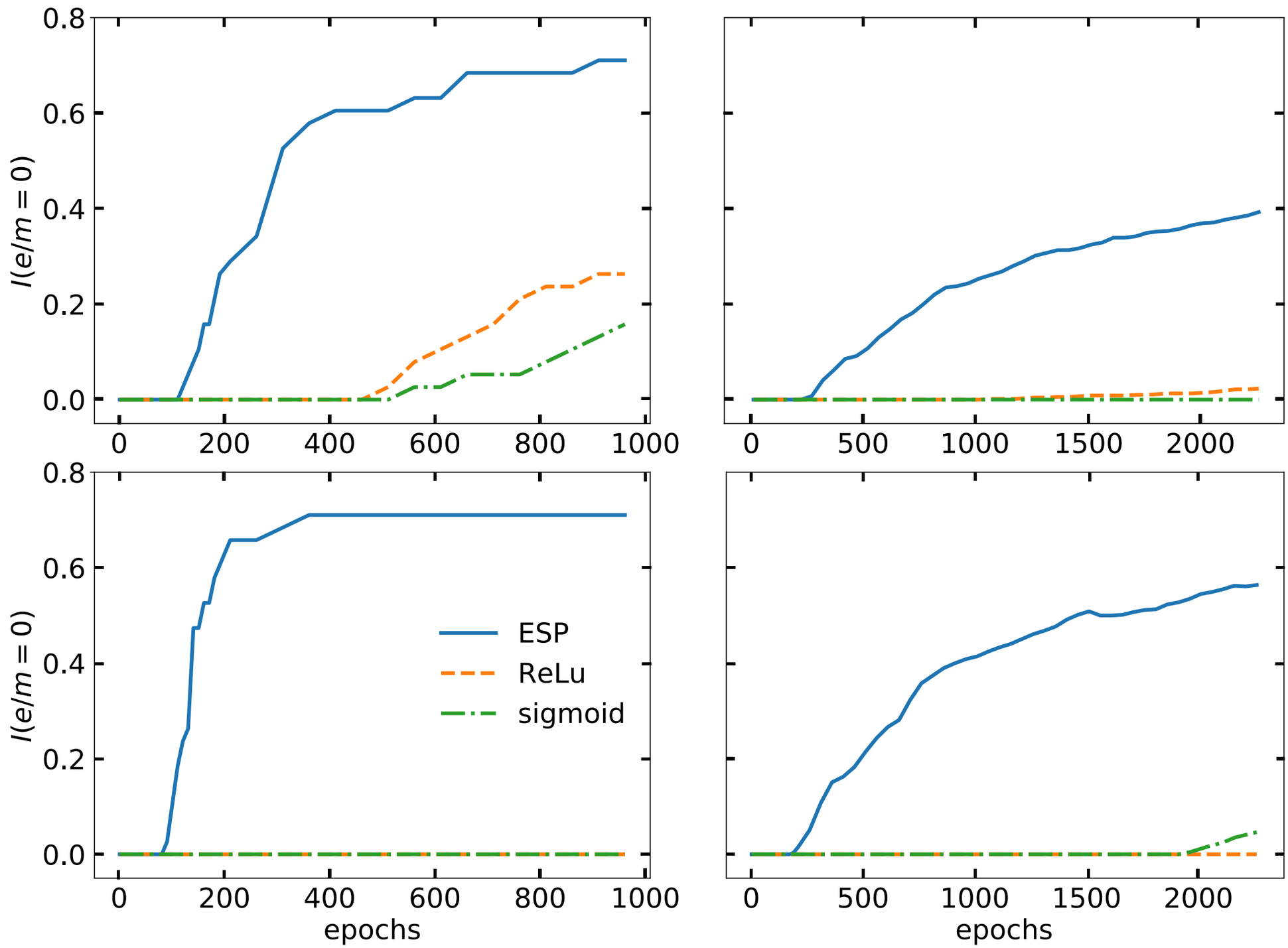
RESIDUALS AS A FUNCTION OF THE TRAINING EPOCH

2-10-1

2-8-2–1

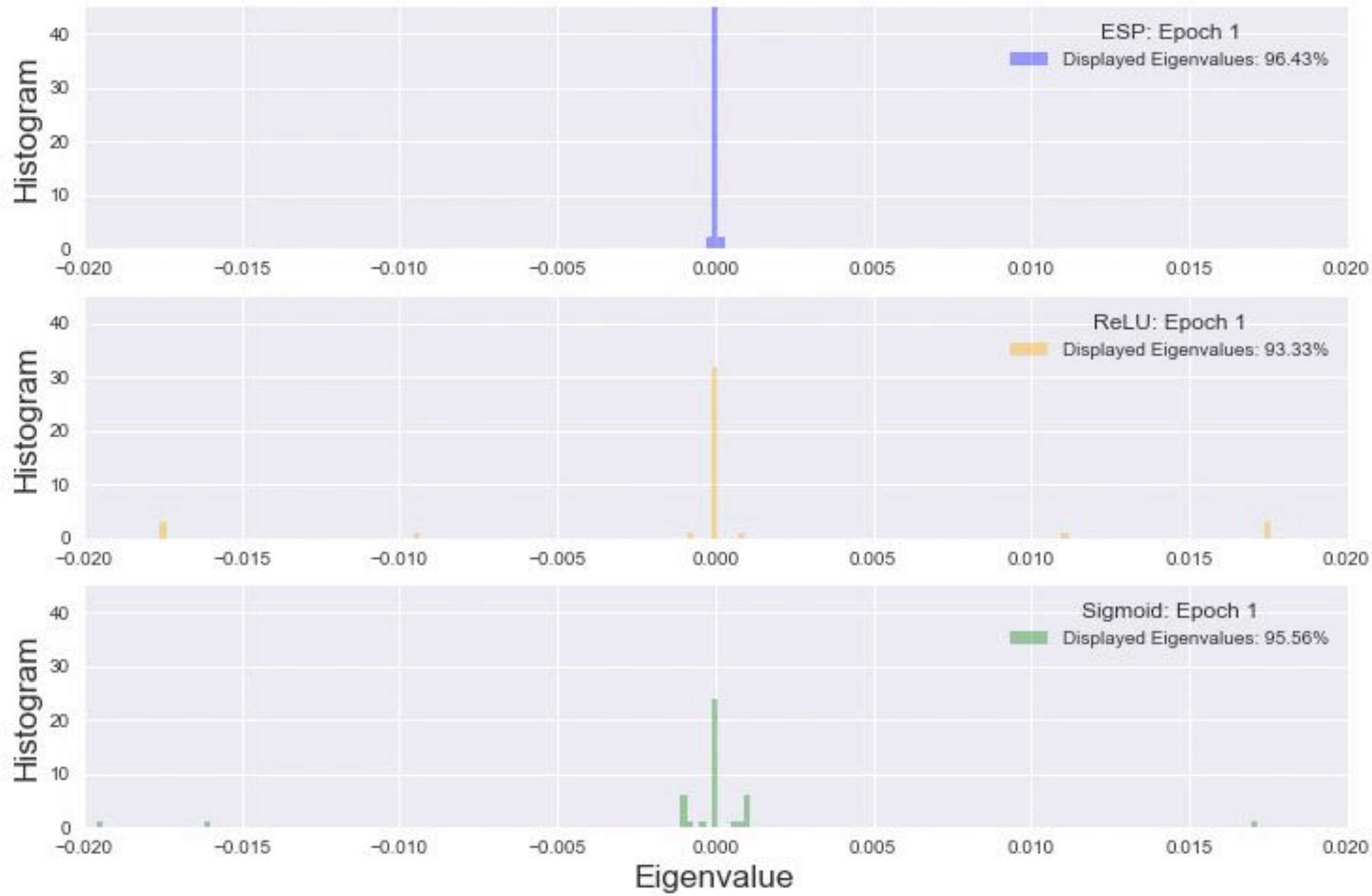
Linear

Non-linear



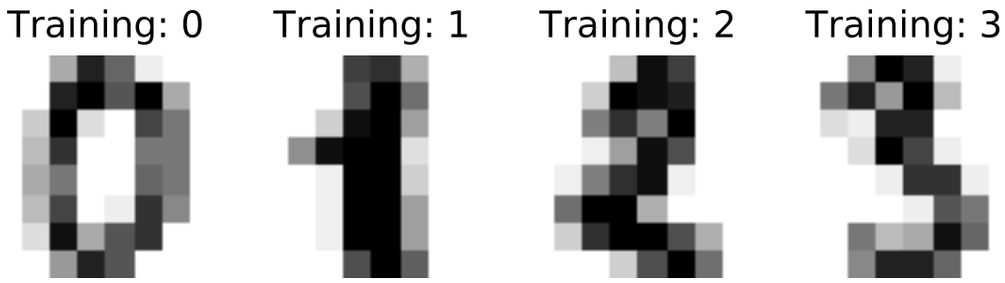
EIGENVALUES DISTRIBUTION: ESP IS THE LESS SINGULAR

L. Sagun, L, Bottou and Y. LeCun. Eigenvalues of the Hessian in Deep Learning: Singularity and Beyond. arXiv:1611.07476v2

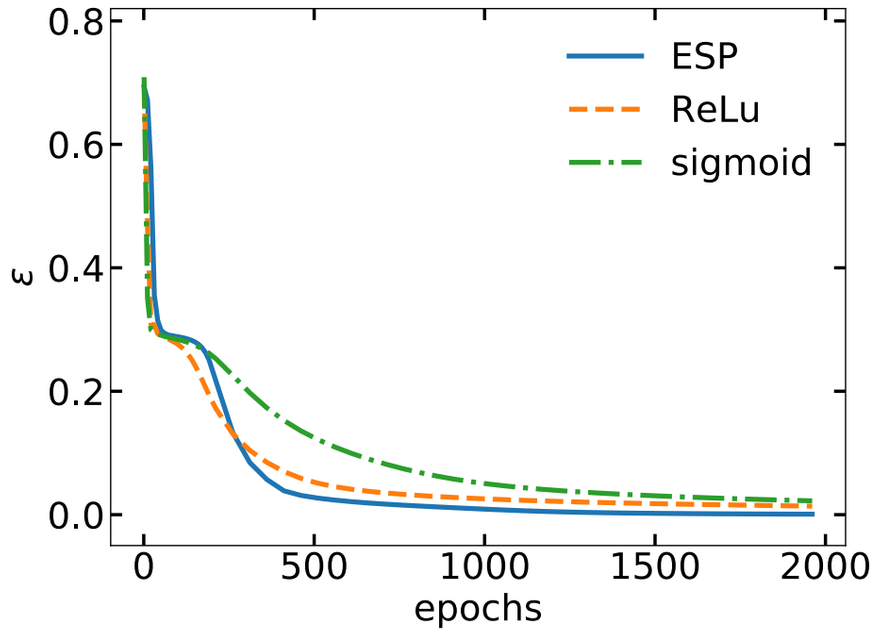


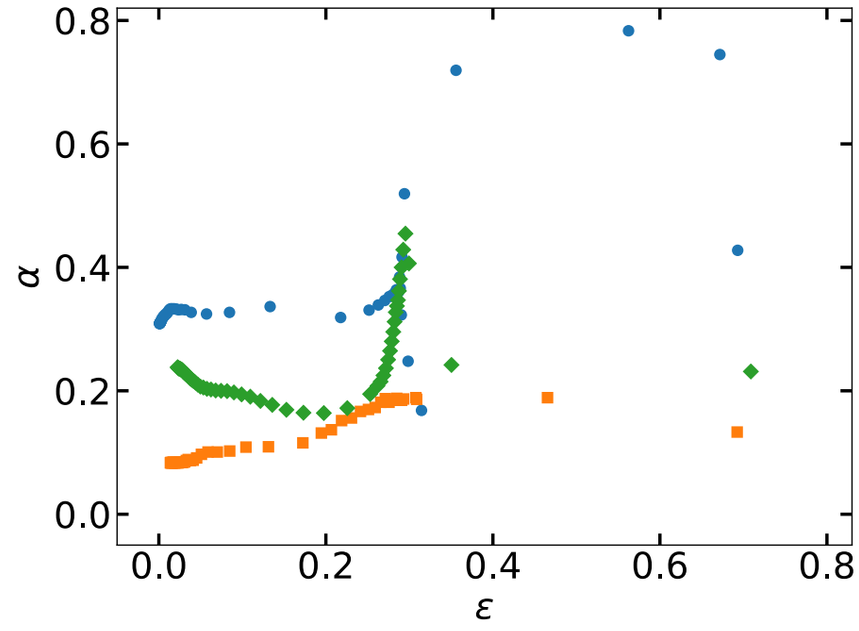
Non-linear 2-8-2-1 (Large eigenvalues are not shown)

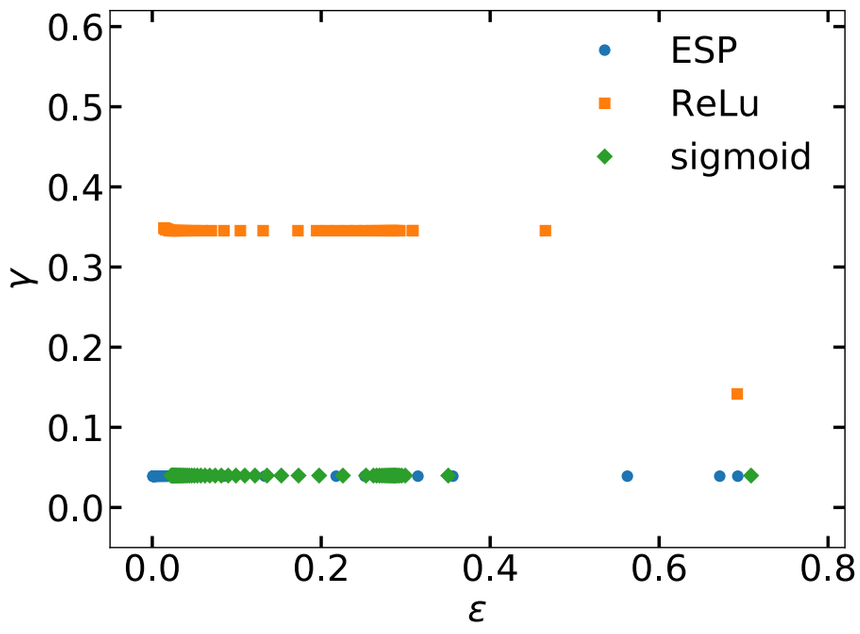
RESULTS II: MULTICLASS CLASSIFICATION ON MNIST

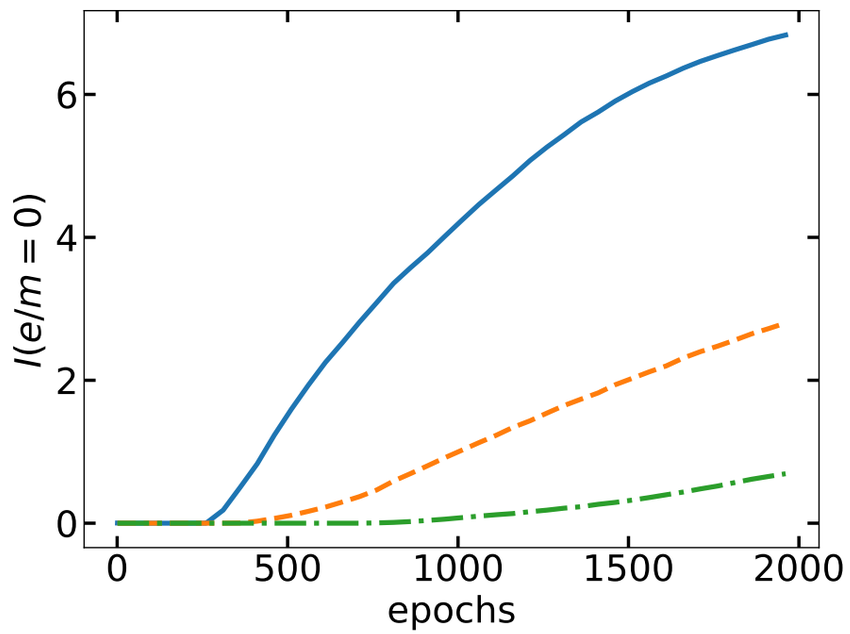


10 digits, 8x8 greyscale images
10-25-10 FF Network









CONCLUSIONS AND PERSPECTIVES

- *Introduced an "energy based" model to describe how current FFN operate*
- *The model returns an optimal, natural activation (ESP) previously found via brute force methods*
- *An analysis of ESP shows when and why it leads to more consistent performances*

Open problems

- *Scaling law of the index of critical points*
- *Larger Networks*
- *How are correlations built?*
- *Connections to the RG*

Relevant Literature

M.M., T. Chotibut and P.E. Trevisanutto ***arXiv:1805.08786,*** *2018.*

P. Ramachandran, B. Zoph and Q. V. Le. **arXiv:1710.05941** , 2017.

S. Hayou, A. Doucet and J. Rosseau, **arXiv:1805.08266v1**, 2018.