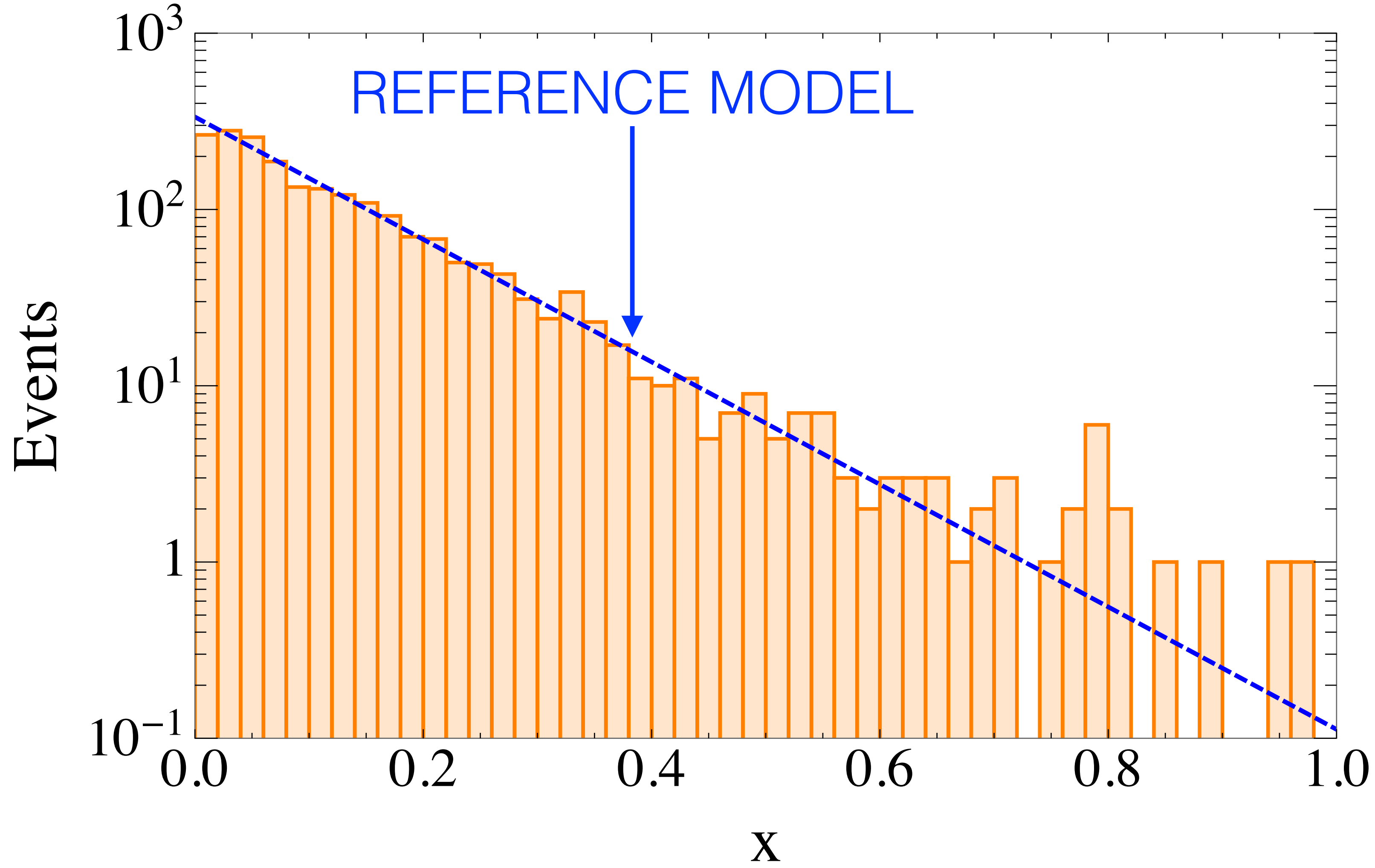
LEARNING NEW PHYSICS FROM A MACHINE

Raffaele Tito D’Agnolo - SLAC

Raffaele Tito D’Agnolo and Andrea Wulzer

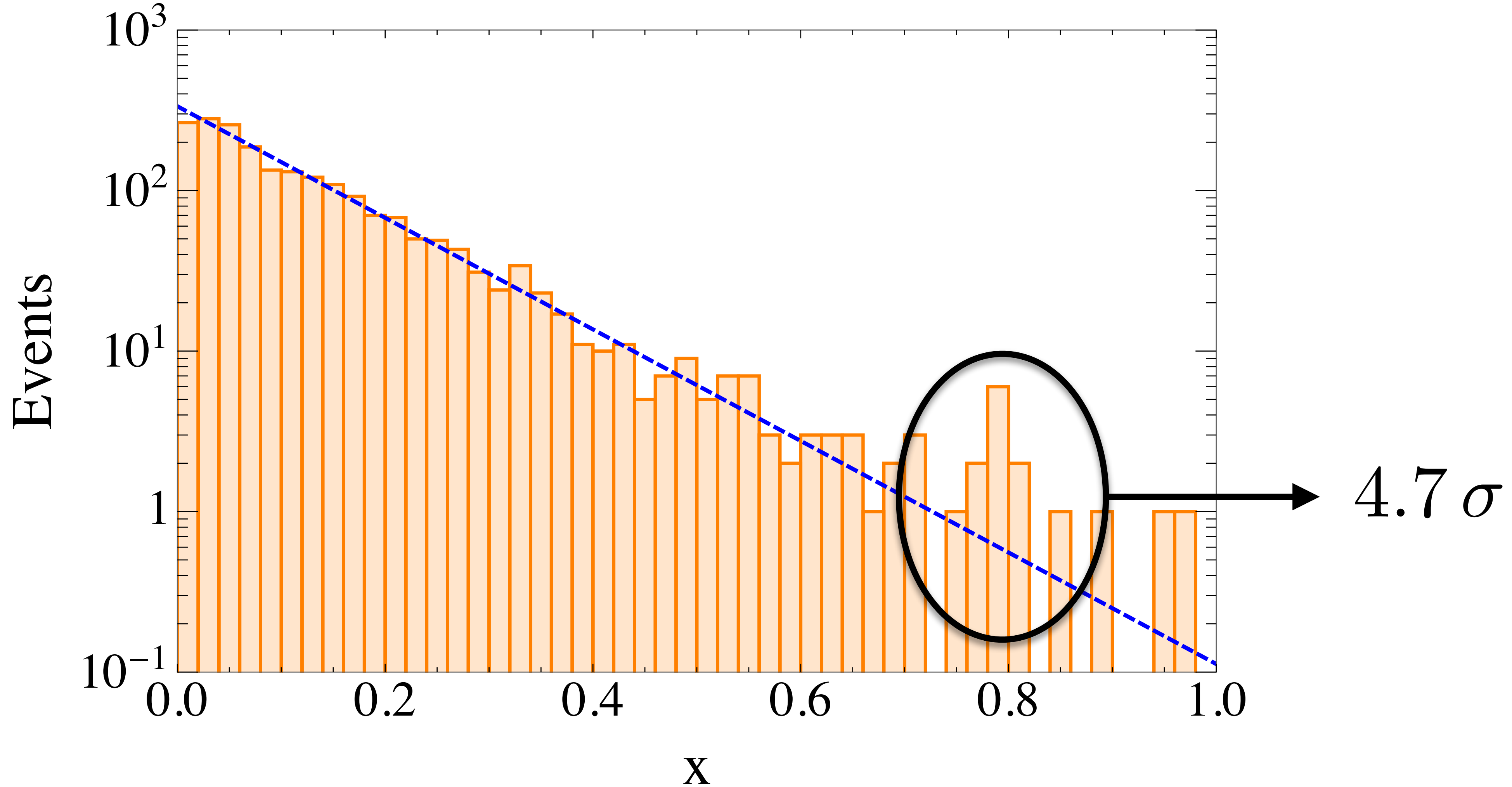
1806.02350, Phys.Rev. D99 (2019) no.1, 015014

THE PROBLEM



$$\chi^2 = 47 \quad N_{\rm bins} = 50 \quad p - \text{value} < 1\sigma$$

THE PROBLEM



$$t_{\rm id}(\mathcal{D}) = 2\,\log\left[\frac{e^{-N(\rm NP)}}{e^{-N(\rm R)}}\prod_{x\in\mathcal{D}}\frac{n(x|\rm NP)}{n(x|\rm R)}\right]$$

WHAT IS A NEURAL NETWORK?

SET OF FUNCTIONS

+

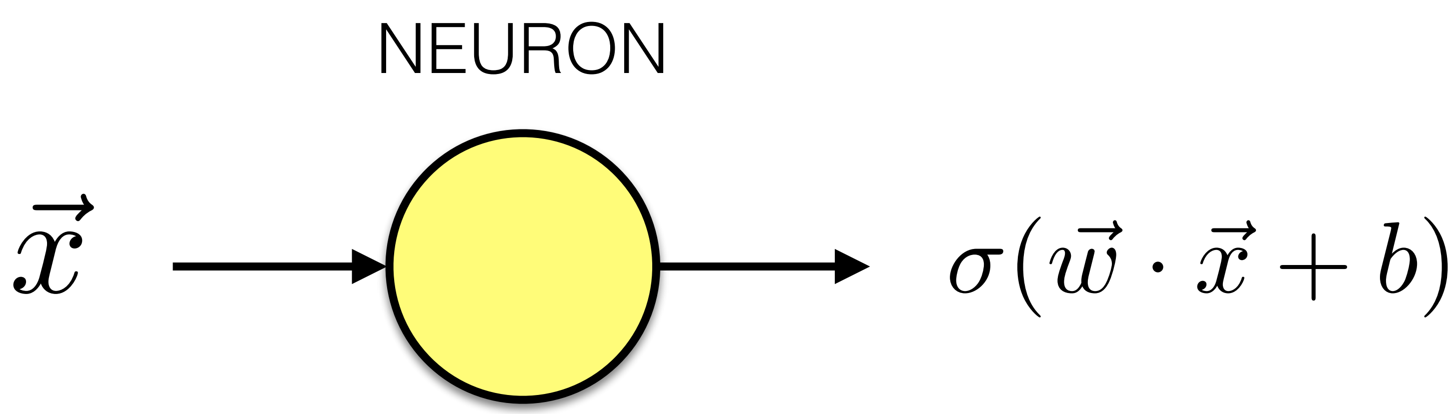
FITTING ALGORITHM

WHAT IS A NEURAL NETWORK?

SET OF FUNCTIONS

$$f_{w_1}^{(1)}\left(f_{w_2}^{(2)}\left(f_{w_3}^{(3)}(\ldots)\right)\right)$$

BUILDING BLOCKS



1. LINEAR TRANSFORMATION $z = \vec{w} \cdot \vec{x} + b$

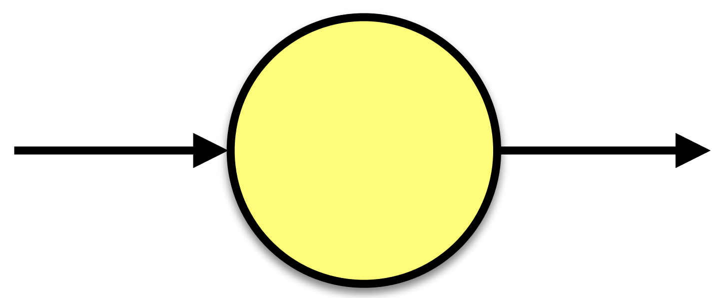
FREE PARAMETERS

2. NON-LINEAR TRANSFORMATION $\sigma(z)$ FIXED

BUILDING BLOCKS

NEURON

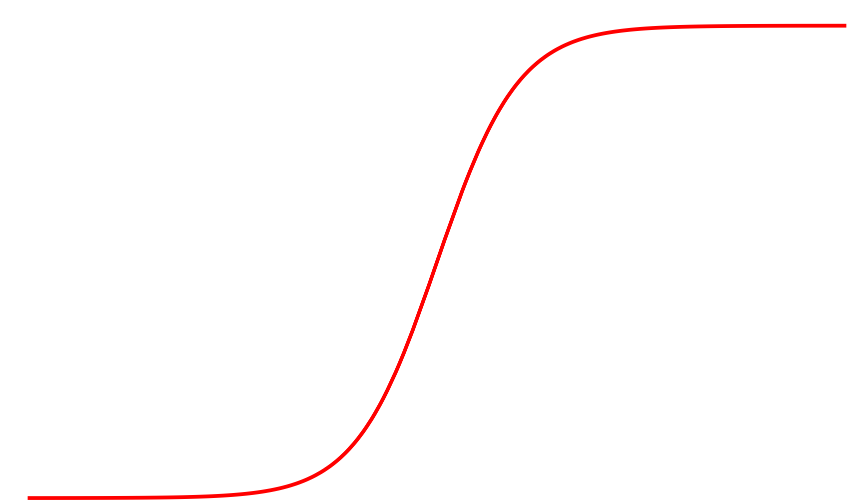
$$\vec{x}$$



$$\sigma(\vec{w} \cdot \vec{x} + b)$$

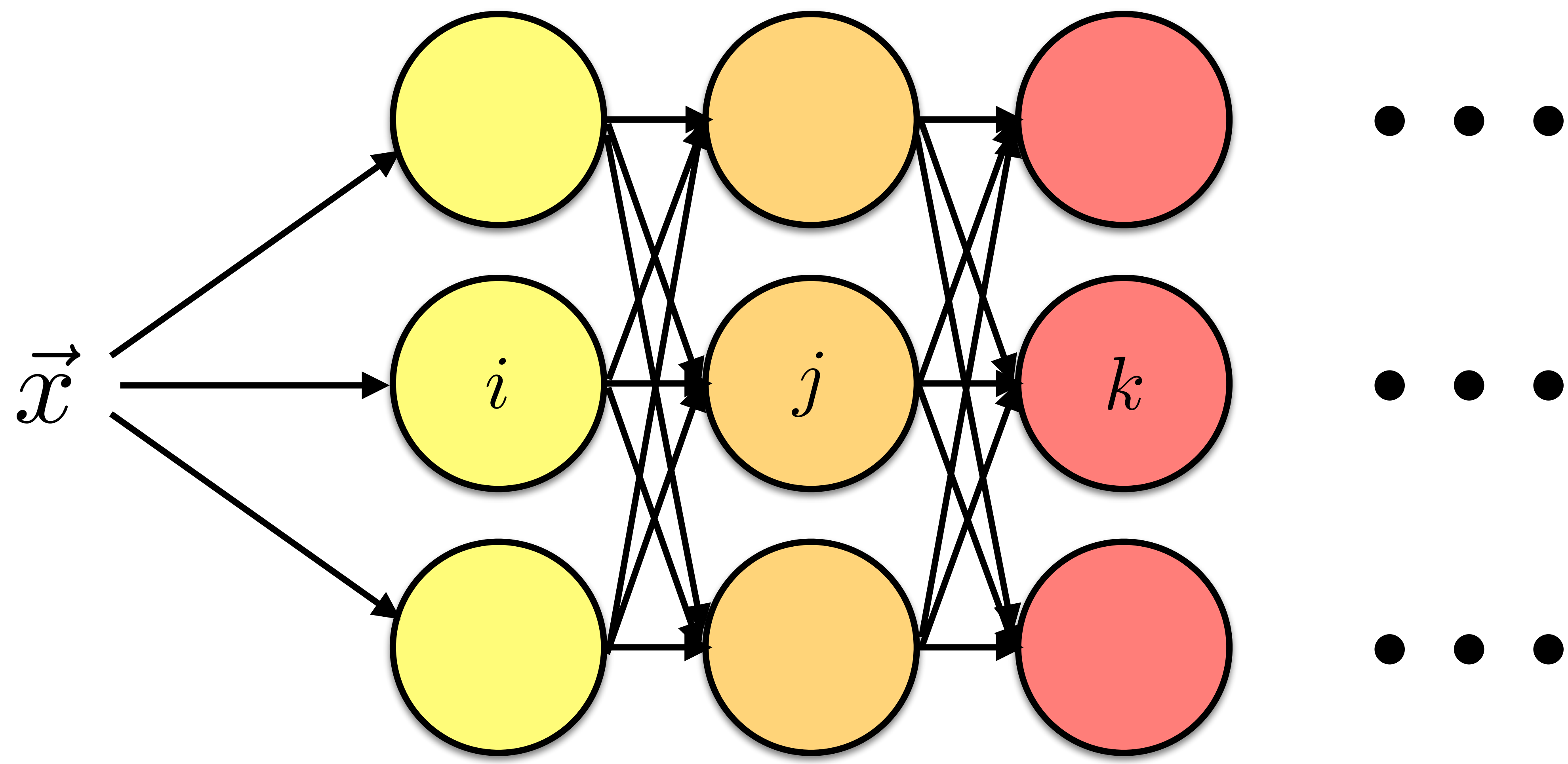
2. NON-LINEAR TRANSFORMATION

$$\sigma(z) = \begin{cases} \tanh(z) \\ \text{ReLU} \\ \frac{1}{1+e^{-z}} \\ \ldots \end{cases}$$



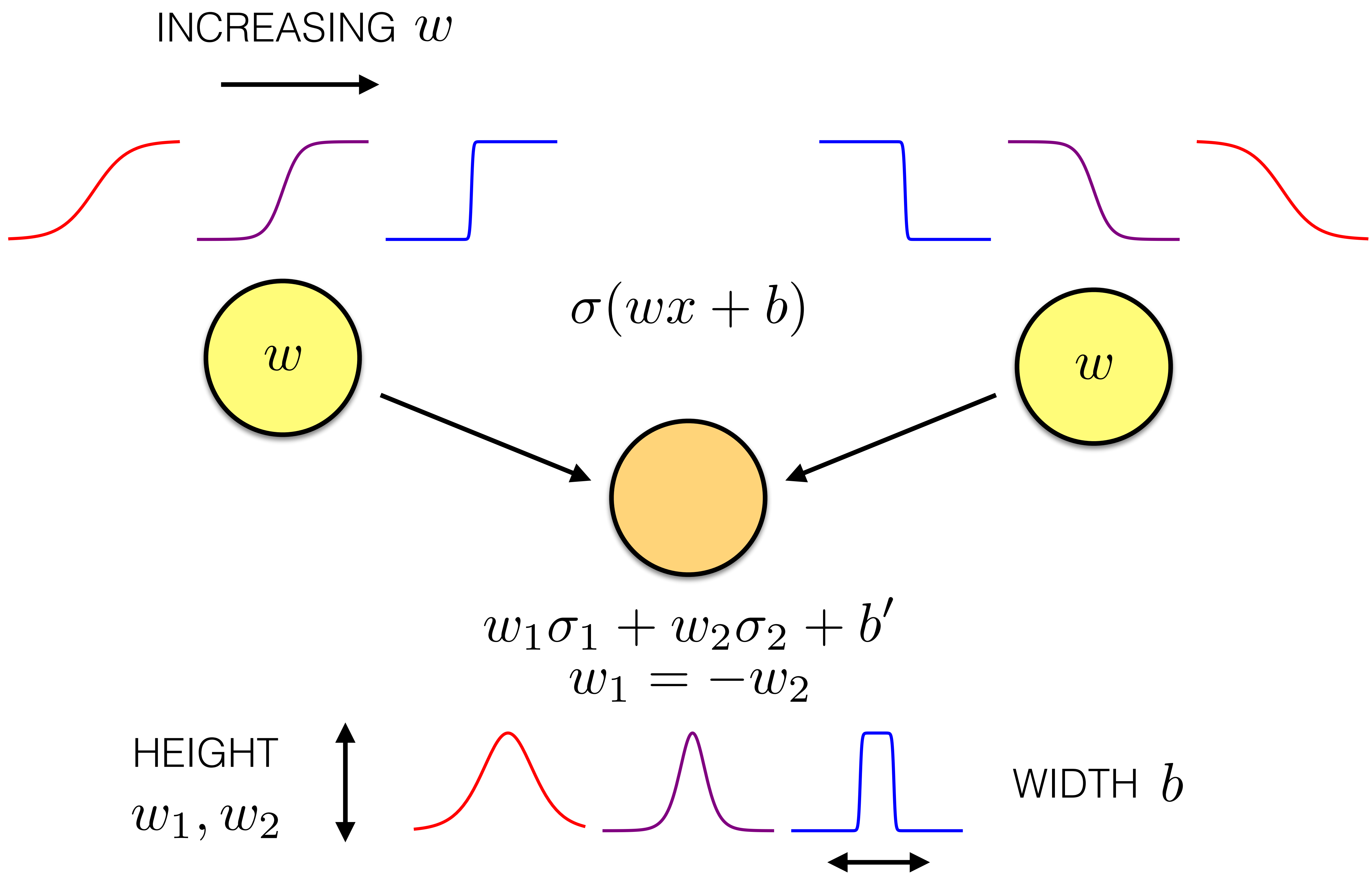
THE NETWORK

FEEDFORWARD, FULLY CONNECTED



$$\sigma\left(\sum_{k=1}^{3} w_{jk}\sigma_k\left(\sum_{i=1}^{3} w_{ki}\sigma_i\left(\sum_{l=1}^{d} w_{il}x_l + b_i\right) + b_k\right) + b_j\right)$$

UNIVERSAL APPROXIMANTS



MAXIMUM LIKELIHOOD DICTIONARY

ELEMENTARY STATISTICS	NEURAL NETWORK
PARAMETERS θ	WEIGHTS AND BIASES w, b
LIKELIHOOD $\mathcal{L}(\theta; x)$	LOSS FUNCTION $-L(w, b; x)$
DATA	TRAINING SAMPLE

FITTING ALGORITHM

LOSS FUNCTION

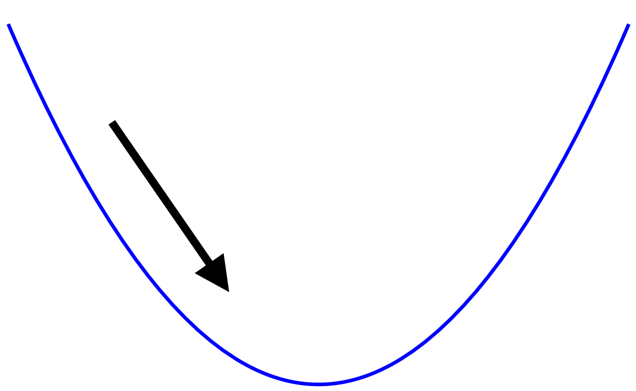
$$L = \frac{1}{N_c}\sum_{i=1}^{N_c}\left[1 - f_{NN}(\vec{x}_i, \mathbf{w}, \mathbf{b})\right]^2 + \frac{1}{N_d}\sum_{j=1}^{N_d}\left[f_{NN}(\vec{x}_j, \mathbf{w}, \mathbf{b})\right]^2$$

TRAINING

$$w_{t+1} \to w_t - \epsilon \partial_w \hat{L}$$

$\hat{L}$ SUBSET OF THE SAMPLE

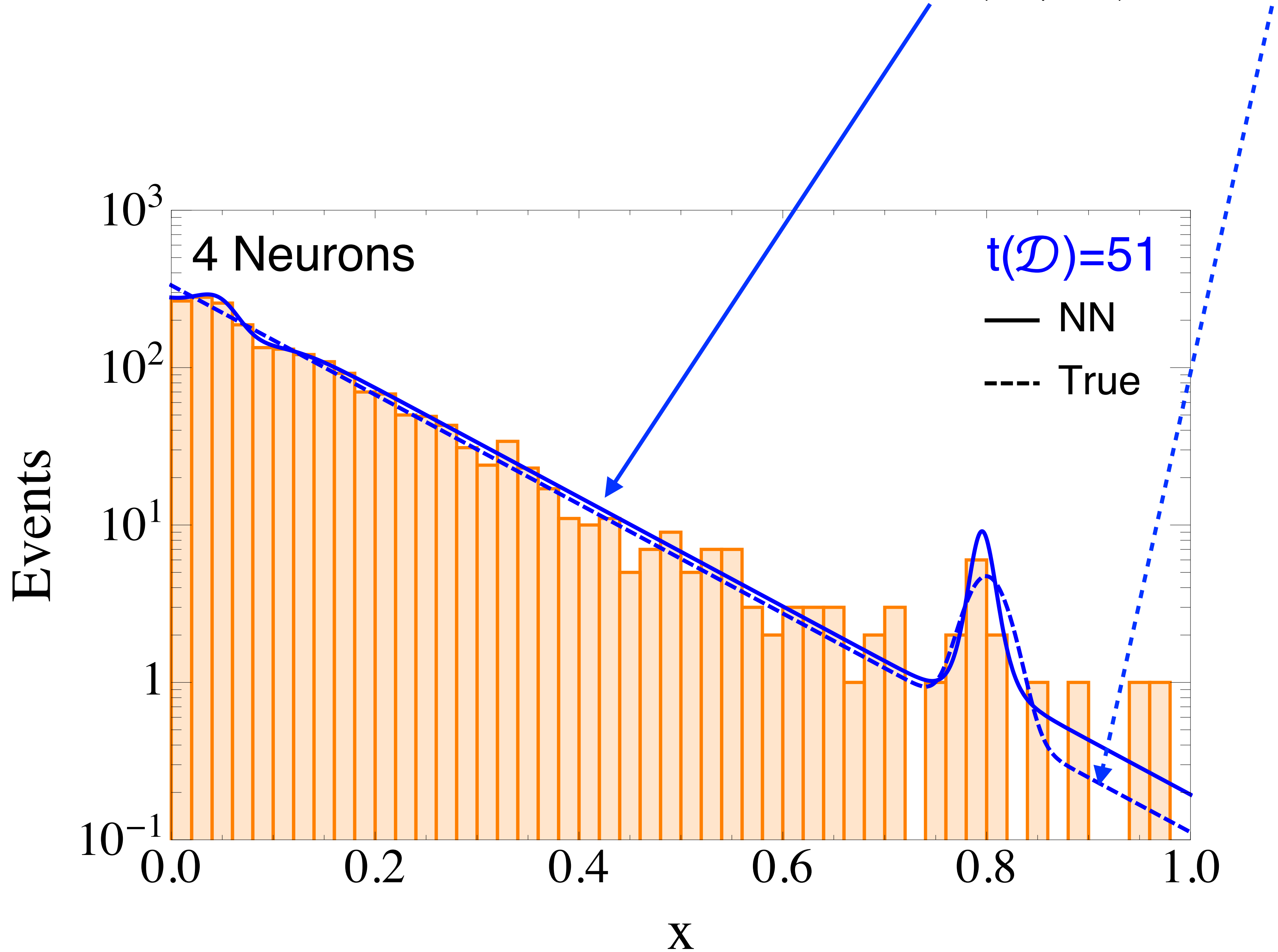
ϵ LEARNING RATE



LEARNING NEW PHYSICS

A SIMPLE STRATEGY

1. LEARN THE DATA DISTRIBUTION $n(x|\widehat{\mathbf{w}}) \approx n(x|\mathrm{T})$



A SIMPLE STRATEGY

1. LEARN THE DATA DISTRIBUTION $n(x|\widehat{\mathbf{w}}) \approx n(x|\mathrm{T})$

2. CHECK IF IT IS DIFFERENT FROM THE REFERENCE ONE

$$t(\mathcal{D}) = 2\,\log\left[\frac{e^{-N(\widehat{\mathbf{w}})}}{e^{-N(\mathrm{R})}}\prod_{x\in\mathcal{D}}\frac{n(x|\widehat{\mathbf{w}})}{n(x|\mathrm{R})}\right] \qquad p_{\mathrm{obs}} = \int_{t_{\mathrm{obs}}}^{\infty} dt\, P(t|\mathrm{R})$$

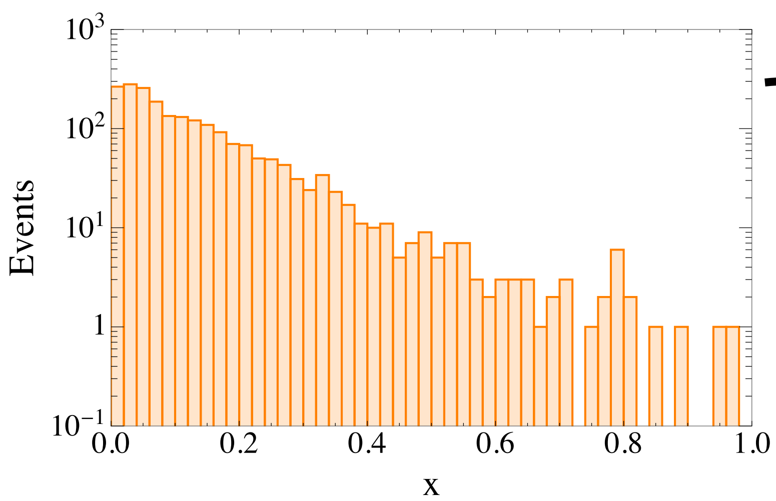
STANDARD LIKELIHOOD RATIO
NEYMAN-PERSON TEST STATISTIC

REFERENCE
DISTRIBUTED
TOYS

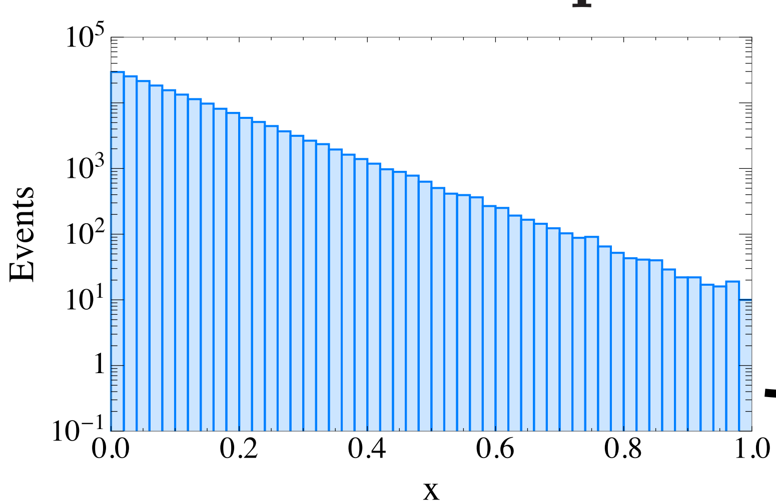
LEARNING PDFs

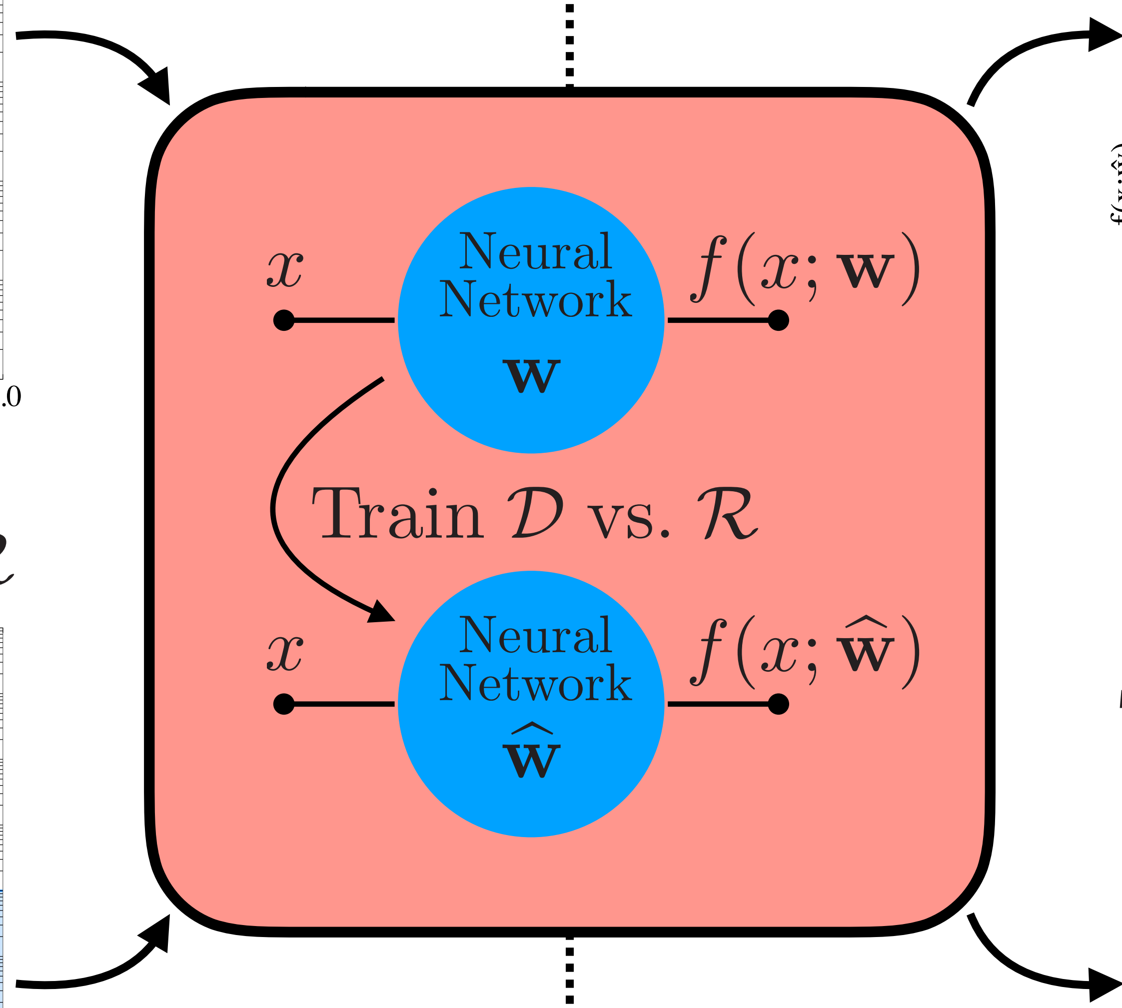
INPUT

Data sample $\mathcal{D}$



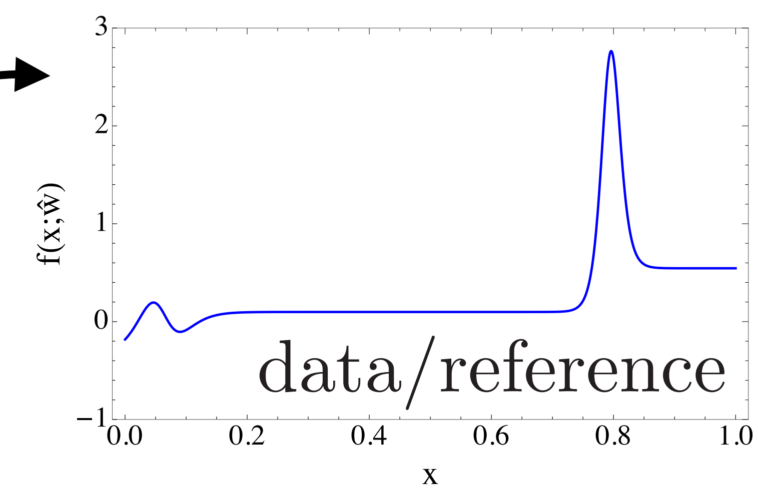
Reference sample $\mathcal{R}$





OUTPUT

Dist. log ratio



$$f(x;\widehat{\mathbf{w}}) \simeq \log\left[\frac{n(x|\mathrm{T})}{n(x|\mathrm{R})}\right]$$

Test statistic t computed on the data sample $\mathcal{D}$

$$t(\mathcal{D}) = -2 \underset{\{\mathbf{w}\}}{\mathrm{Min}}\, L[f]$$

THE LOSS FUNCTION

$$n(x|\mathbf{w}) = n(x|\mathrm{R})\, e^{f(x;\mathbf{w})}$$ → NEURAL NETWORK

$$t(\mathcal{D}) = 2\,\log\left[\frac{e^{-N(\widehat{\mathbf{w}})}}{e^{-N(\mathrm{R})}}\prod_{x\in\mathcal{D}}\frac{n(x|\widehat{\mathbf{w}})}{n(x|\mathrm{R})}\right]$$

$$= -2\,\underset{\{\mathbf{w}\}}{\mathrm{Min}}\left[\frac{N(\mathrm{R})}{\mathcal{N}_{\mathcal{R}}}\sum_{x\in\mathcal{R}}(e^{f(x;\mathbf{w})}-1)-\sum_{x\in\mathcal{D}}f(x;\mathbf{w})\right]$$

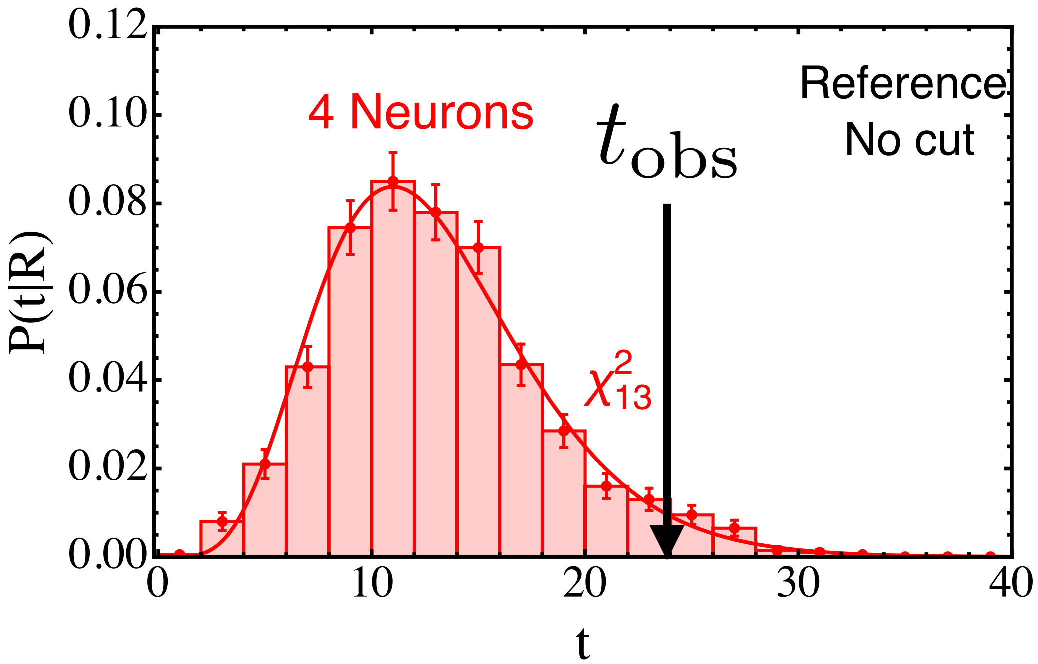
$$\equiv -2\,\underset{\{\mathbf{w}\}}{\mathrm{Min}}\,L[f(\,\cdot\,,\mathbf{w})]$$ → LOSS FUNCTION

SUMMARY

1. TRAIN THE NETWORK ON THE DATA
   - **INPUT**: ONE DATA SAMPLE AND ONE REFERENCE SAMPLE
   - **OUTPUT**: TEST STATISTIC ON THE DATA SAMPLE AND DISTRIBUTION LOG-RATIO
2. GENERATE TOY DATA SAMPLES THAT FOLLOW THE REFERENCE DISTRIBUTION AND TRAIN THE NETWORK AGAIN USING THEM AS DATA
   - **INPUT**: TOY DATA AND SAME REFERENCE SAMPLE AS ABOVE
   - **OUTPUT**: DISTRIBUTION OF THE TEST STATISTIC IN THE REFERENCE HYPOTHESIS

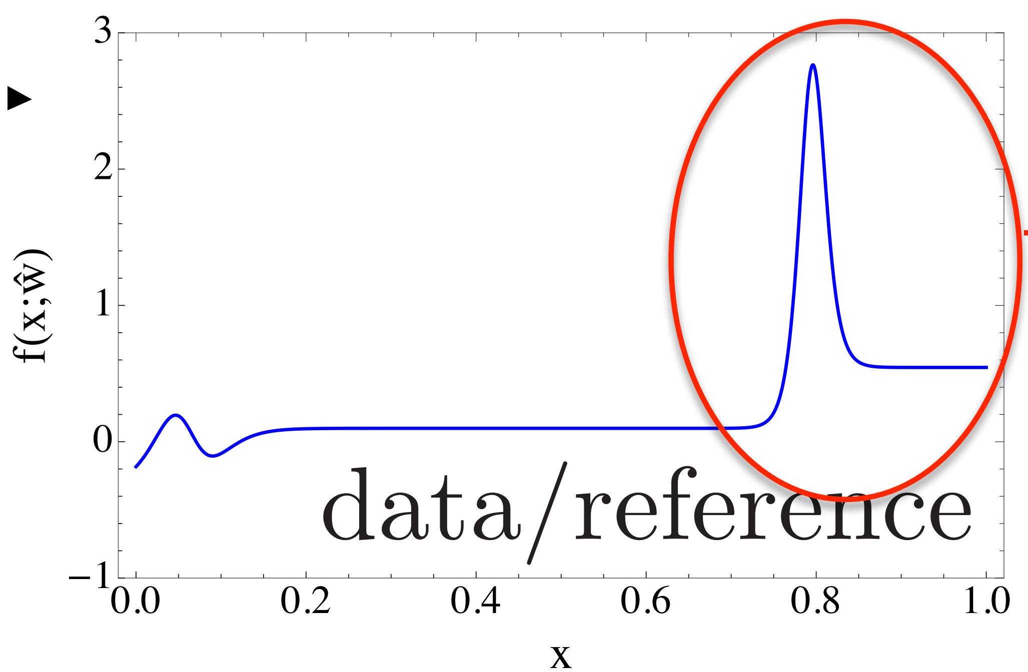
SUMMARY

3.



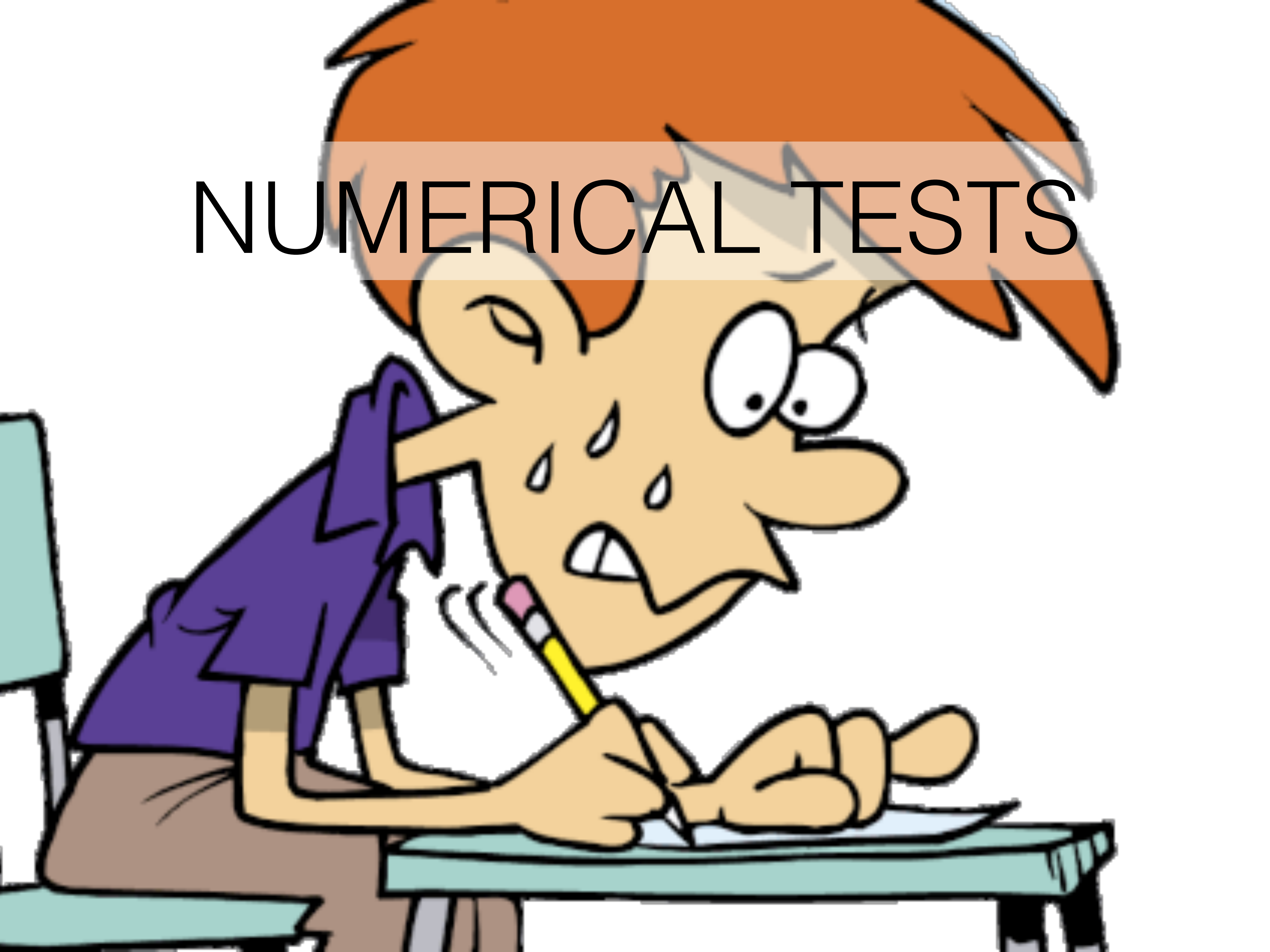
$$p_{obs} = \int_{t_{obs}}^{\infty} dt\, P(t|\mathrm{R})$$

4.

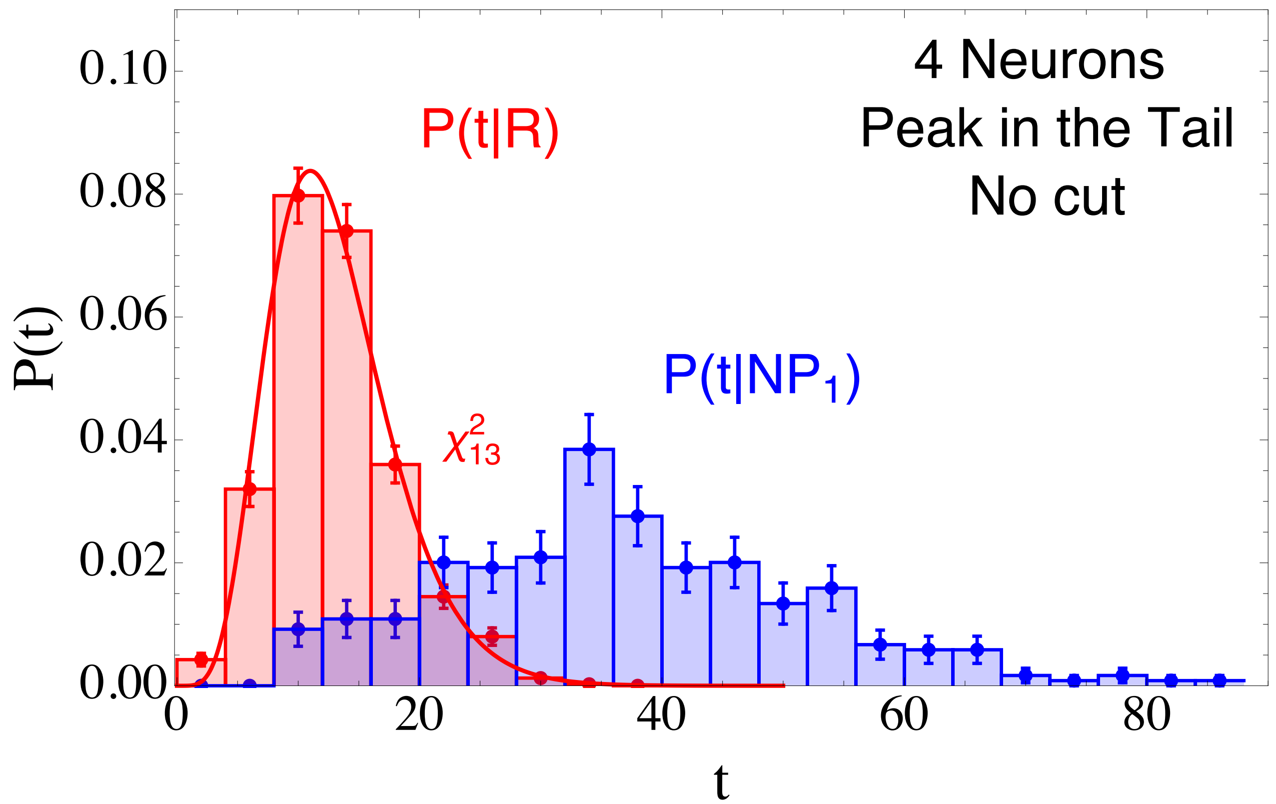


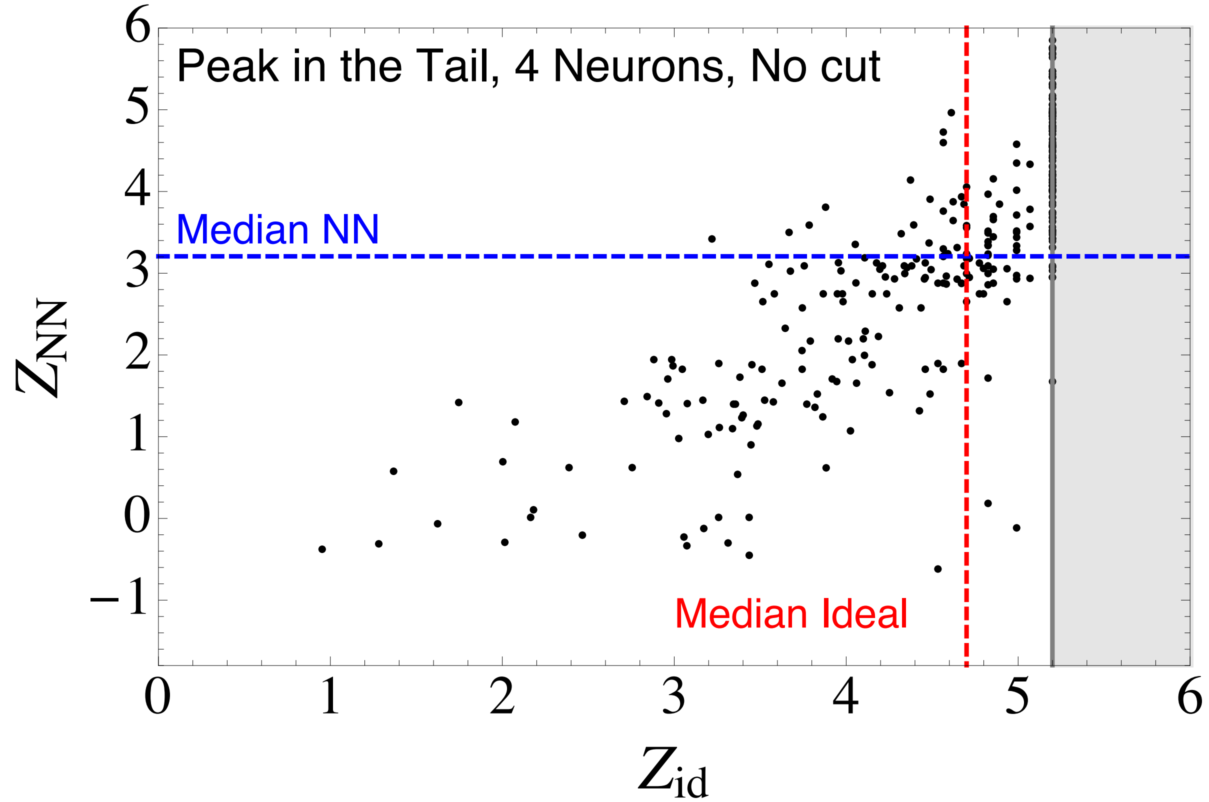
IDENTIFY AND CHARACTERIZE NEW PHYSICS

NUMERICAL TESTS

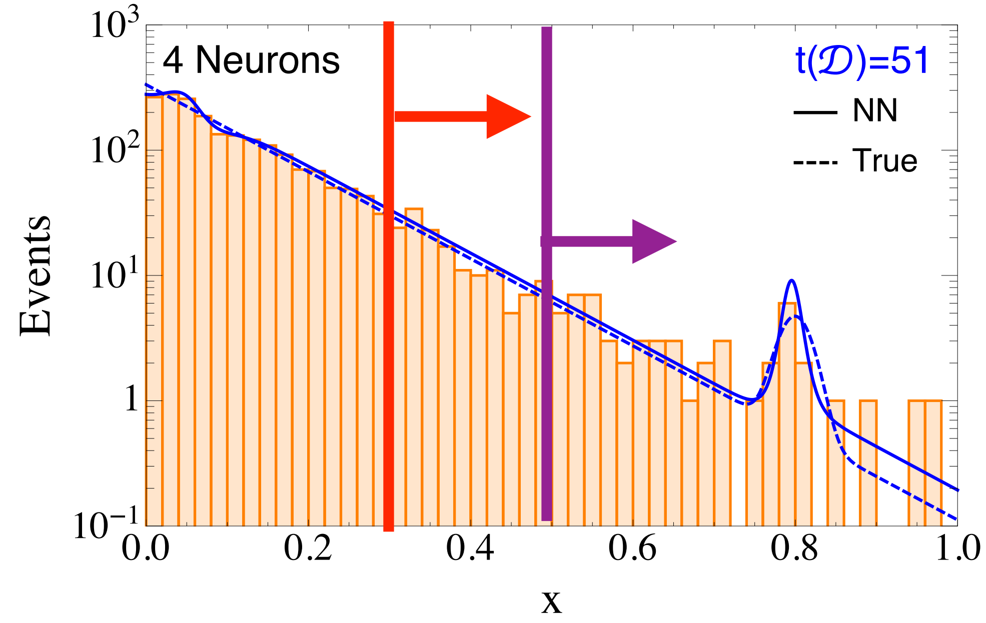


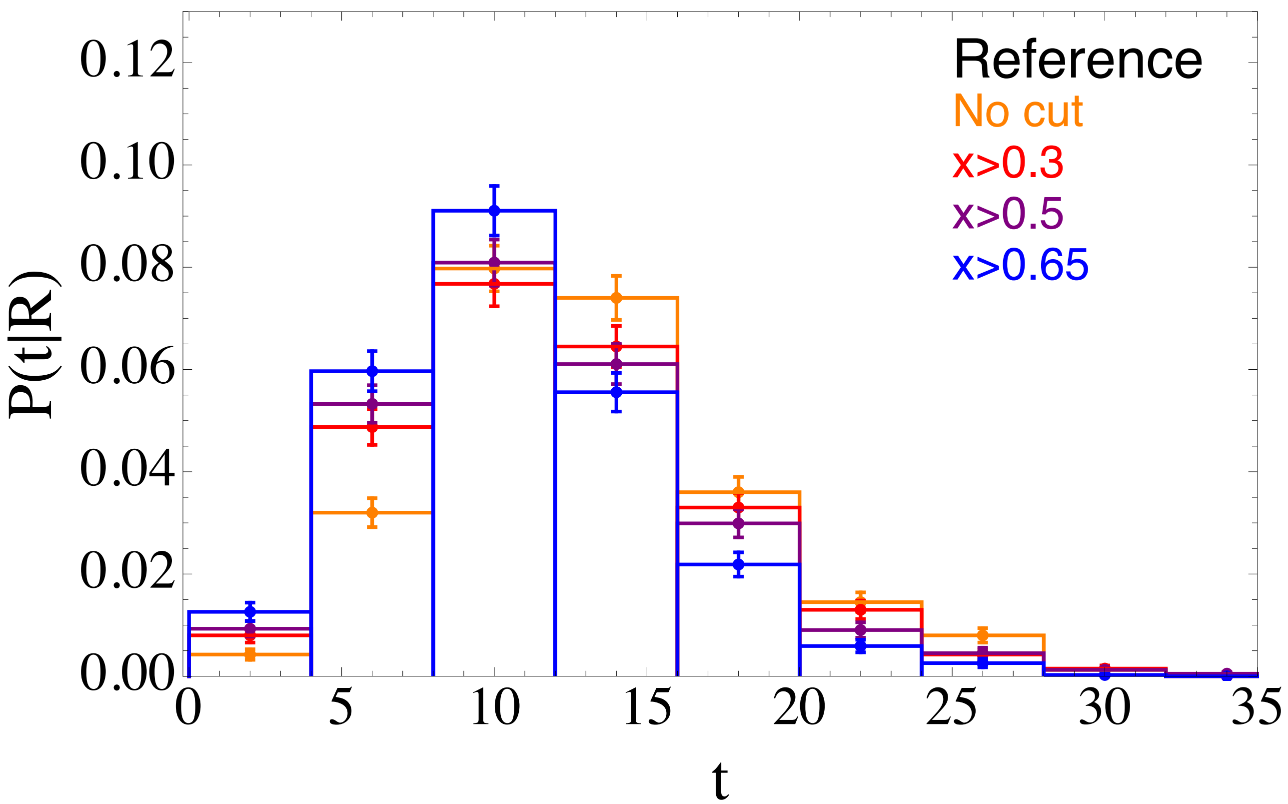
SENSITIVE TO NEW PHYSICS

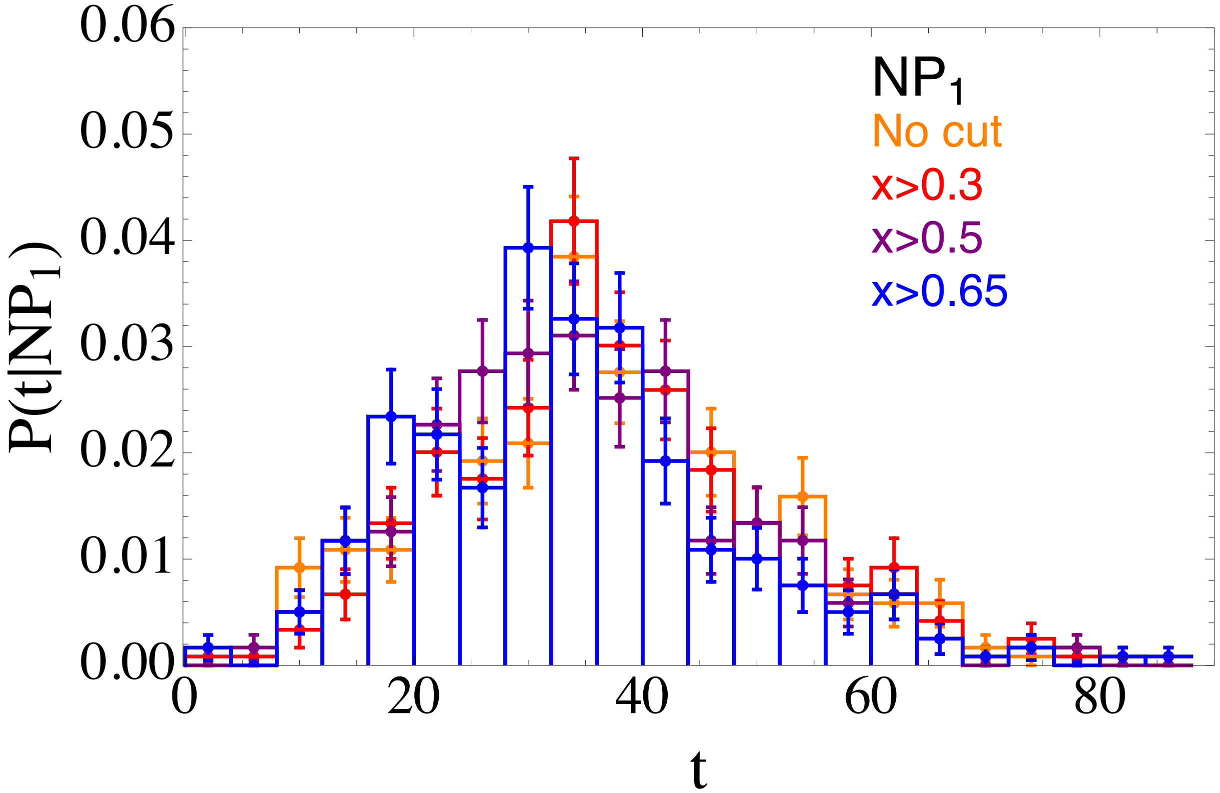




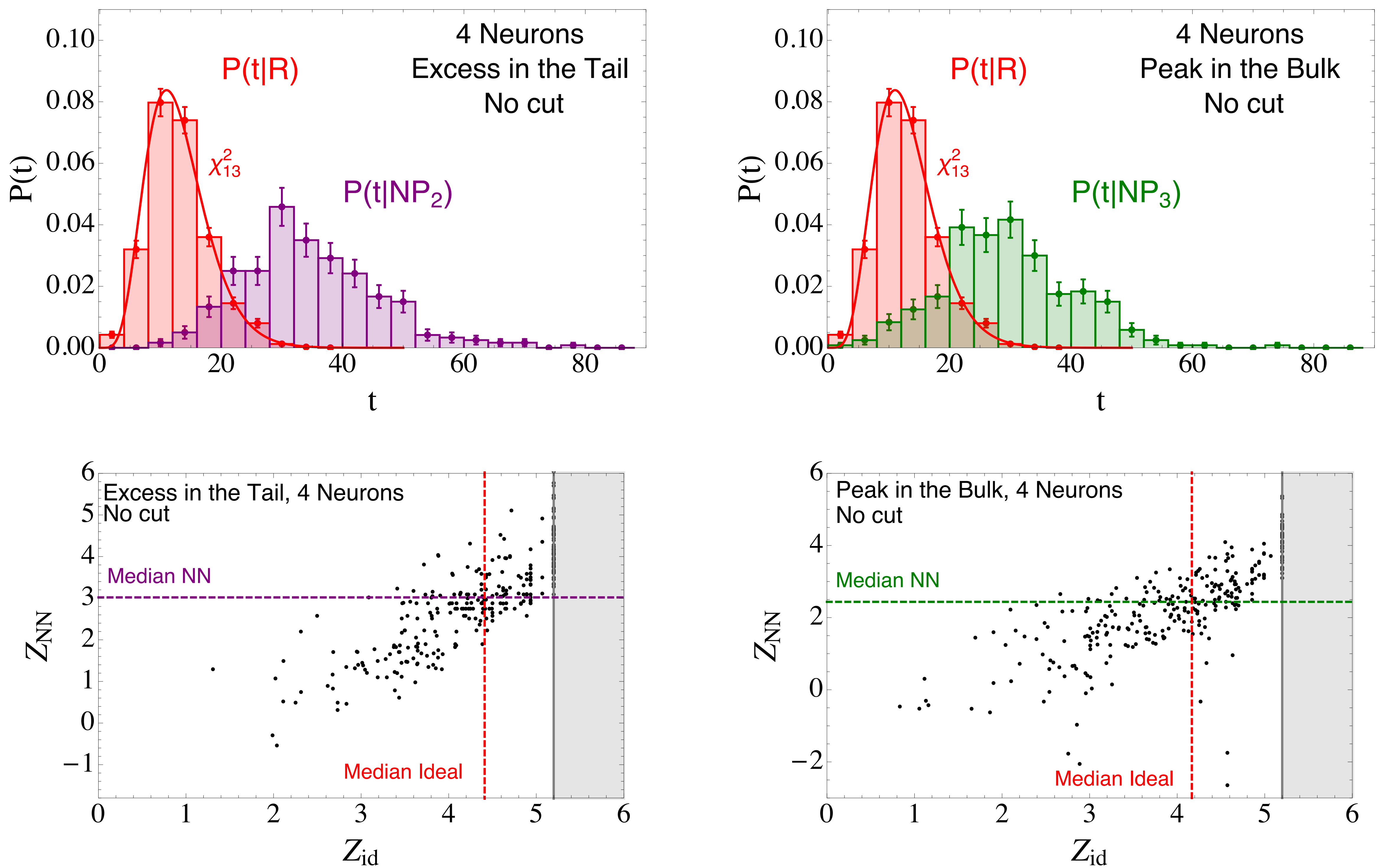
INSENSITIVE TO CUTS







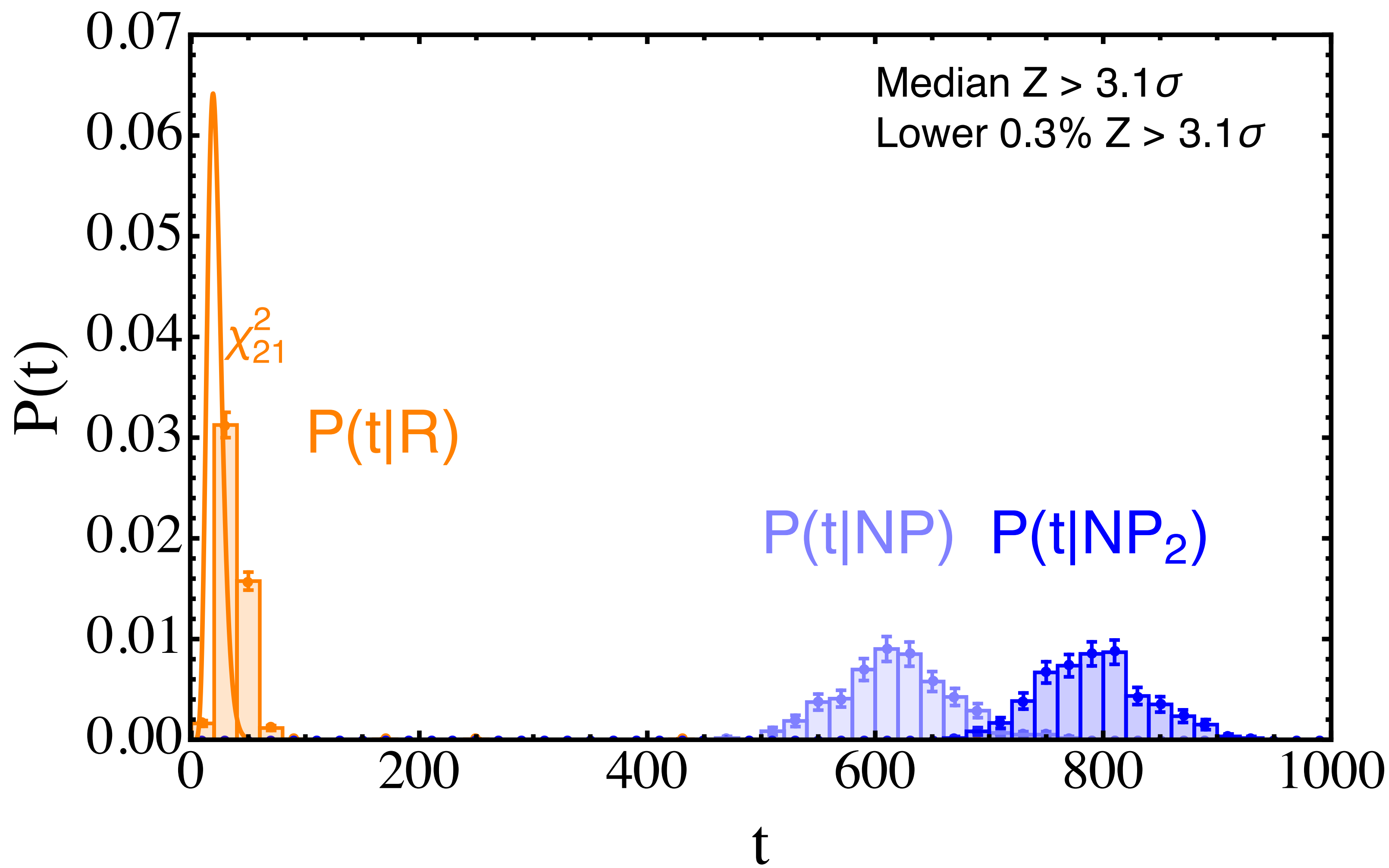
MODEL-INDEPENDENT



COMPARISONS

arXiv:1807.06038

Trial Dataset	$\boldsymbol{\mu}_T$	Σ_T	p-value	Z
$\mathcal{T}_{G0}$	$(1.0, 1.0)^T$	((1.0,0.0),(0.0,1.0))	8.2×10^{-1}	$0.2\ \sigma$
$\mathcal{T}_{G1}$	$(1.12, 1.12)^T$	((1.0,0.0),(0.0,1.0))	2.8×10^{-2}	$2.2\ \sigma$
$\mathcal{T}_{G2}$	$(1.0, 1.0)^T$	((0.95,0.1),(0.1,0.8))	4.0×10^{-4}	$3.5\ \sigma$
$\mathcal{T}_{G3}$	$(1.15, 1.15)^T$	((1.0,0.0),(0.0,1.0))	1.2×10^{-6}	$4.9\ \sigma$

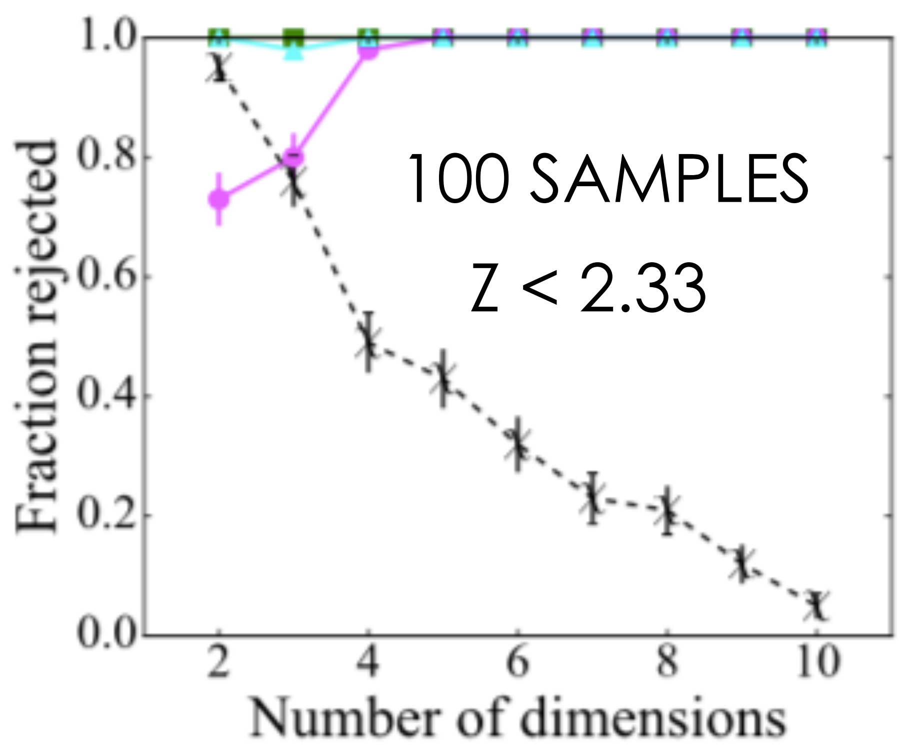


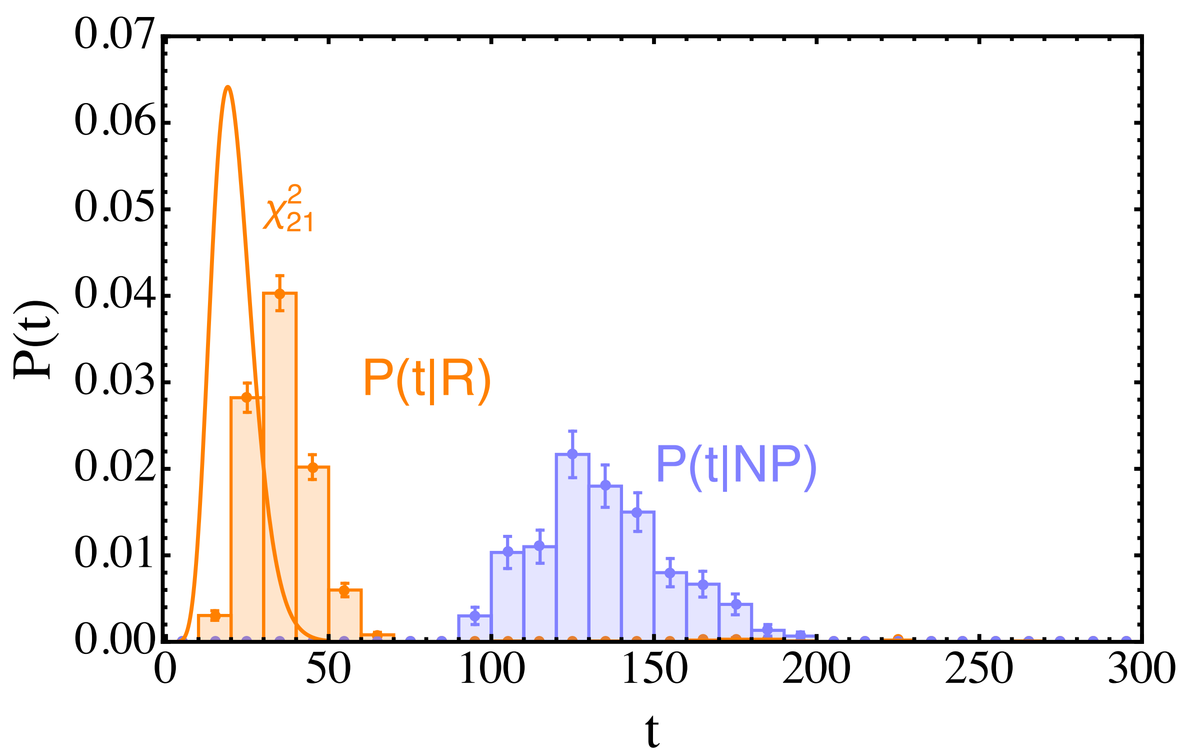
COMPARISONS

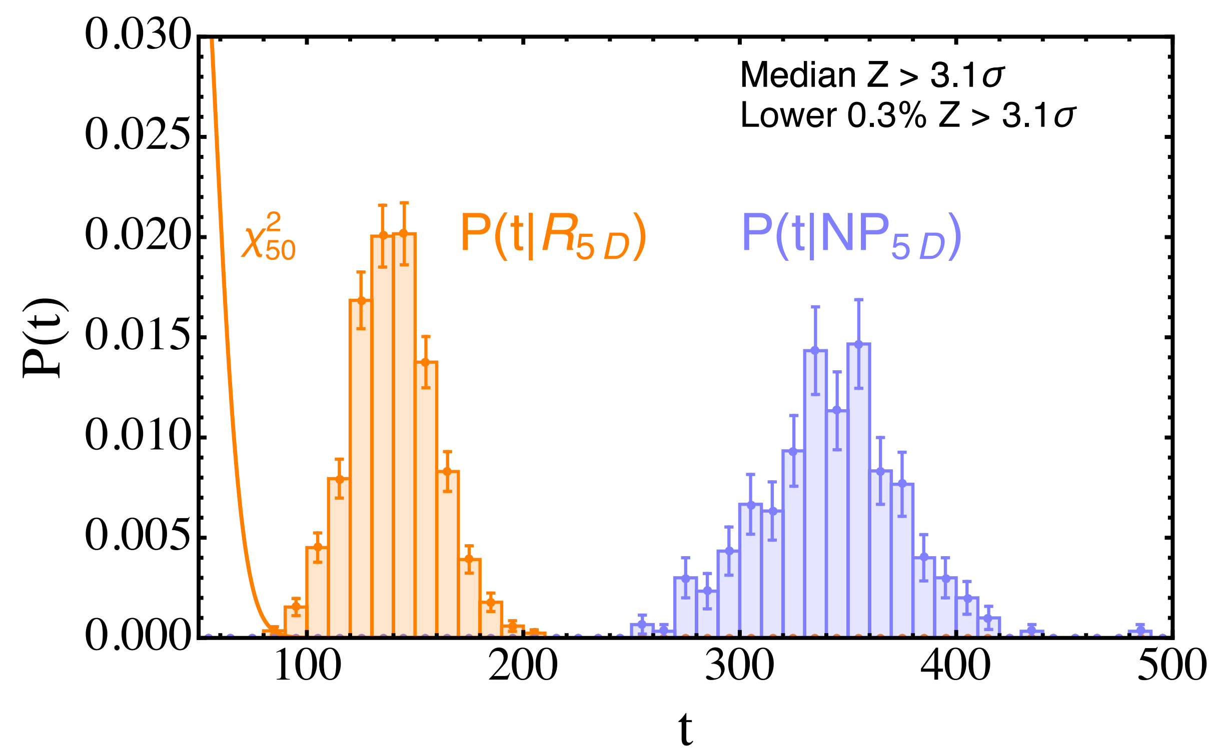
arXiv:1612.07186

$$n(x|R) = \prod_{k=1}^{D} \mathcal{N}(0,1)$$

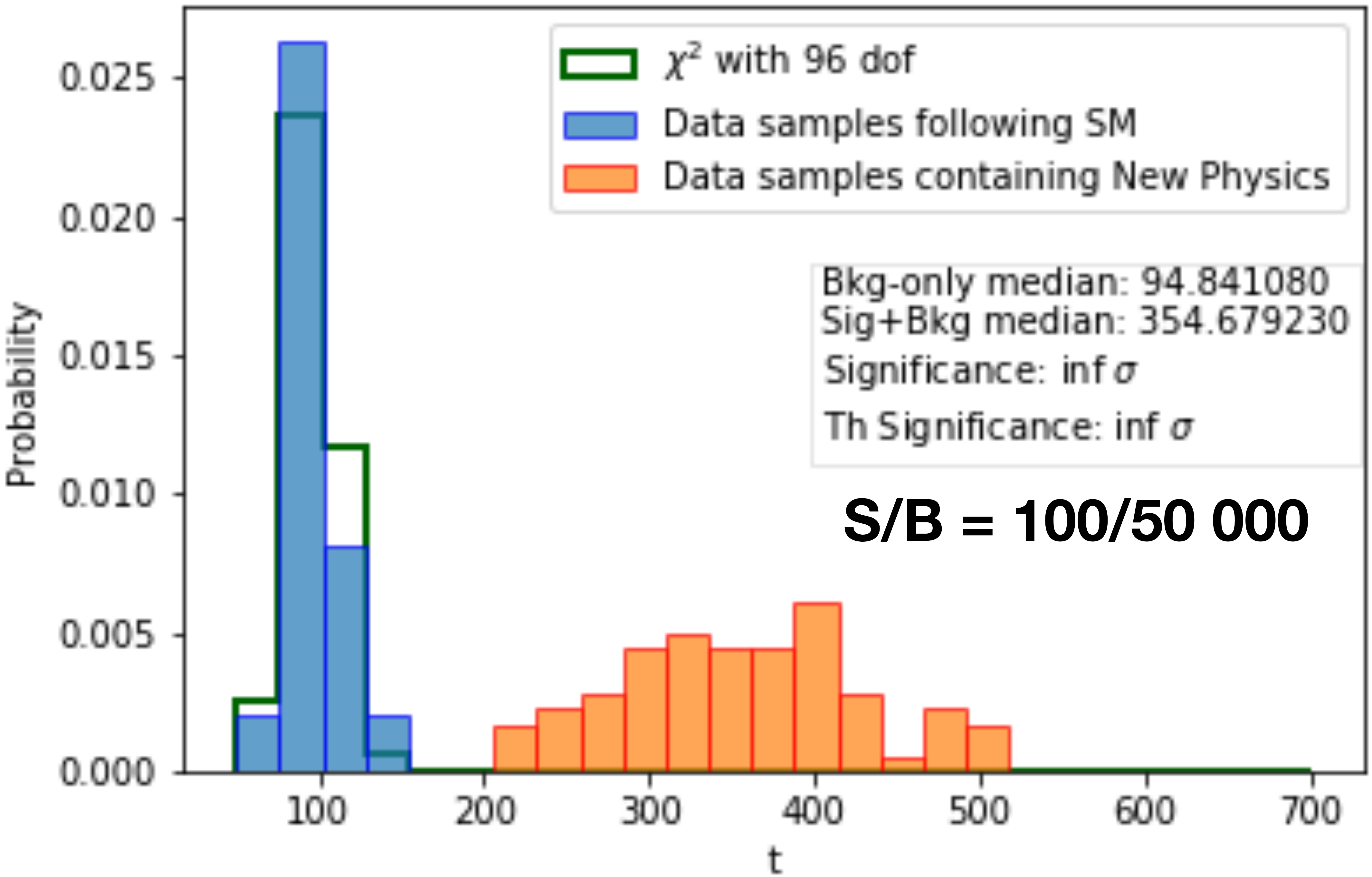
$$n(x|NP) = \prod_{k=1}^{D} \mathcal{N}(0,0.95)$$







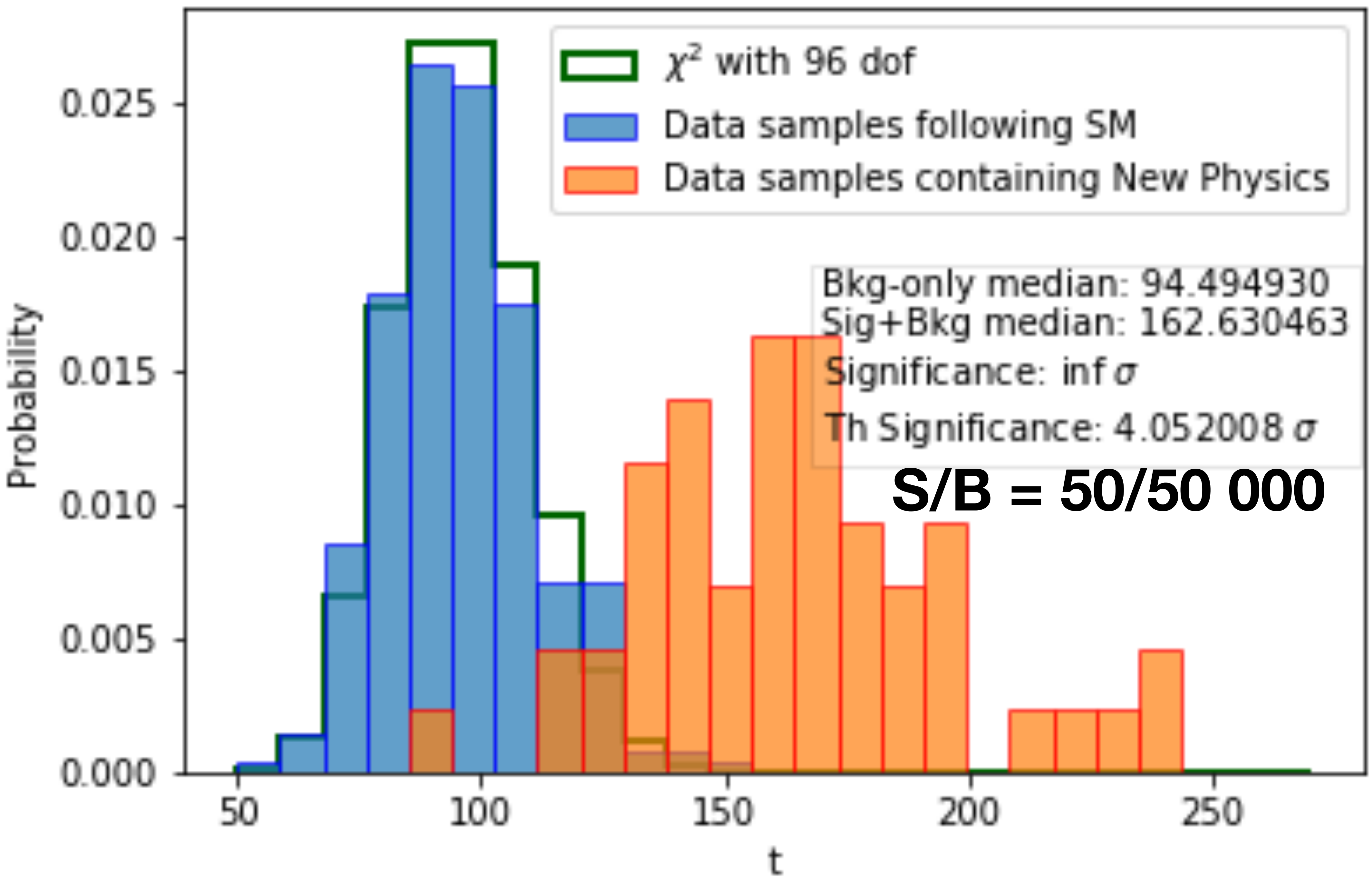
Z' TO DILEPTONS (5D)



S/B = 100/50 000

S/√B ≈ 26.0 ± 0.6

Z' TO DILEPTONS (5D)



S/B = 50/50 000

S/√B ≈ 13 ± 0.3

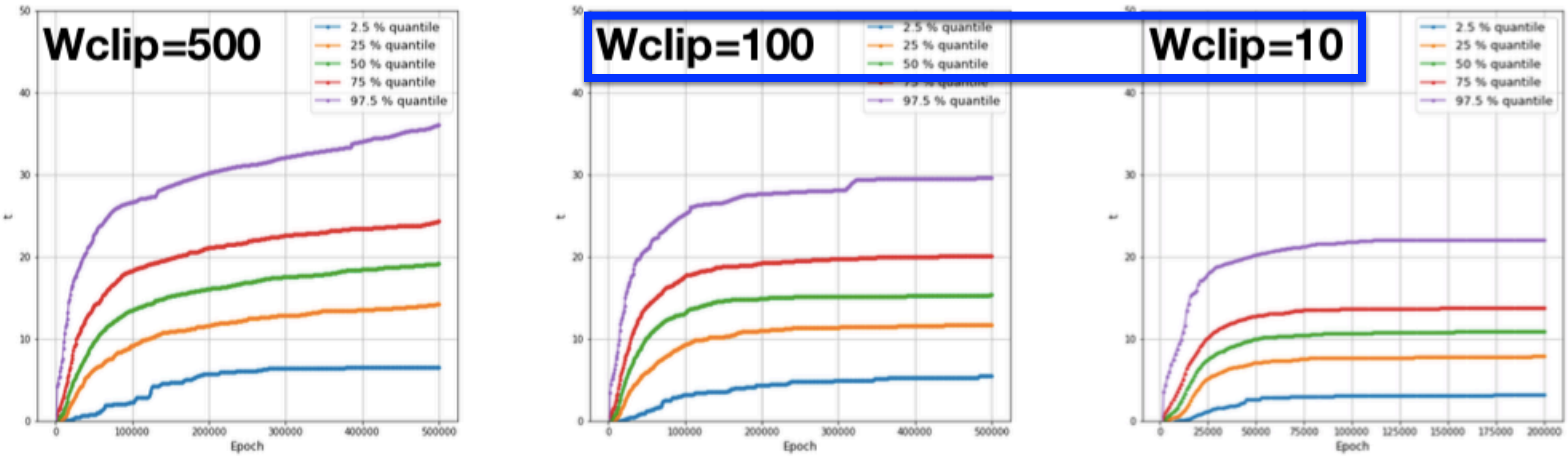
NETWORK ARCHITECTURE

Thanks to machine-learning algorithms,
the robot apocalypse was short-lived.

NETWORK ARCHITECTURE

O. Cerri, **Gaia Grosso**, RTD, M. Pierini, A. Wulzer, M. Zanetti in progress

1. GIVEN A NETWORK ARCHITECTURE FIND A RANGE OF THE WEIGHT CLIPPING THAT DOES NOT GENERATE OUTLIERS IN THE REFERENCE MODEL (=NO OVERFITTING)



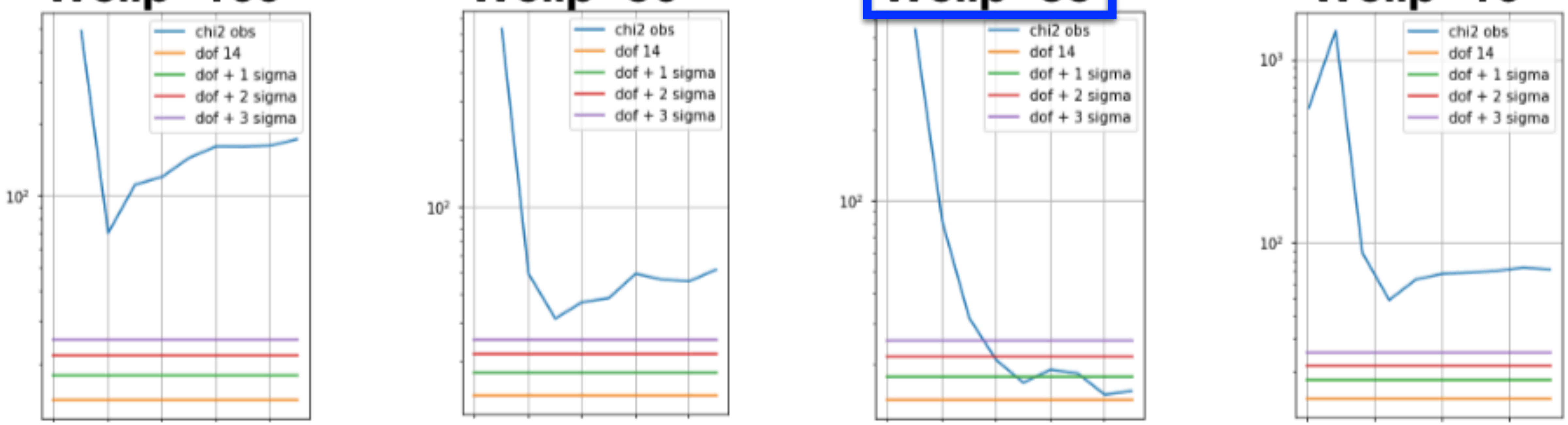
NETWORK ARCHITECTURE

O. Cerri, **Gaia Grosso**, RTD, M. Pierini, A. Wulzer, M. Zanetti in progress

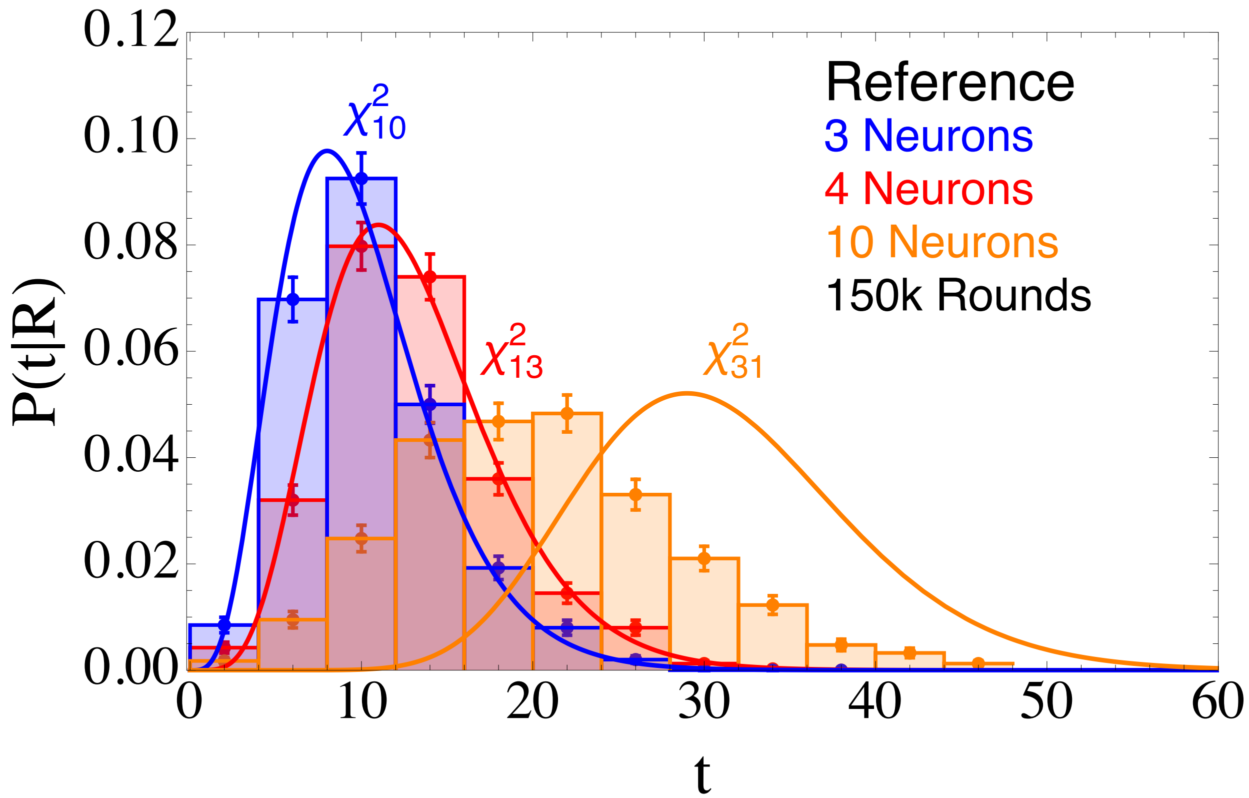
2. WITHIN THE RANGE CHOOSE THE WEIGHT CLIPPING THAT MAXIMIZES THE COMPATIBILITY WITH THE CHI2

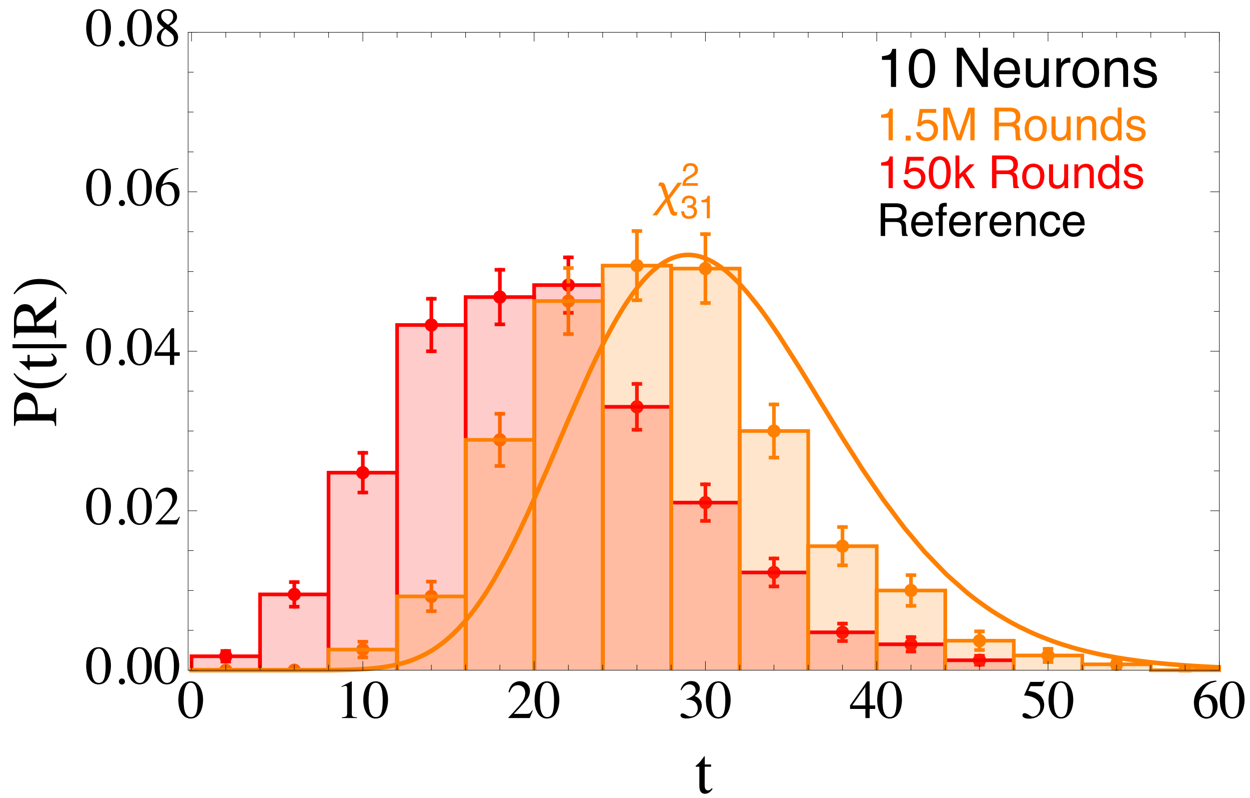
3. PICK THE MINIMUM VALUE OF TRAINING ROUNDS THAT DOES NOT SPOIL 1. AND 2.

Wclip=100 **Wclip=50** **Wclip=35** **Wclip=10**



NETWORK ARCHITECTURE





CONCLUSION AND OUTLOOK

- TODAY IN FUNDAMENTAL PHYSICS WE HAVE LARGE, MULTIVARIATE, SM-LIKE DATASETS AND STRONG REASONS TO BELIEVE THAT THEY SHOULD NOT BE SM-LIKE

- OUR BEST GUESSES FOR NEW PHYSICS ARE NOT BEING DETECTED AND ANYTHING THAT HELPS US TO SEARCH WITHOUT ANY BIAS CAN BE USEFUL

- NEURAL NETWORKS ARE WIDELY USED TO APPROXIMATE PROBABILITY DISTRIBUTIONS AND ARE IDEAL CANDIDATES FOR THIS TYPE OF PROBLEM

- TODAY I HAVE DESCRIBED AN APPLICATION OF NEURAL NETWORKS, FOUNDED ON SOLID STATISTICAL PRINCIPLES, WHICH GOES IN THIS DIRECTION

  - ITS VIRTUES (SENSITIVITY TO NP, MODEL-INDEPENDENCE, INSENSITIVITY TO CUTS) HAVE BEEN TESTED ON SIMPLE 1D AND 2D EXAMPLES

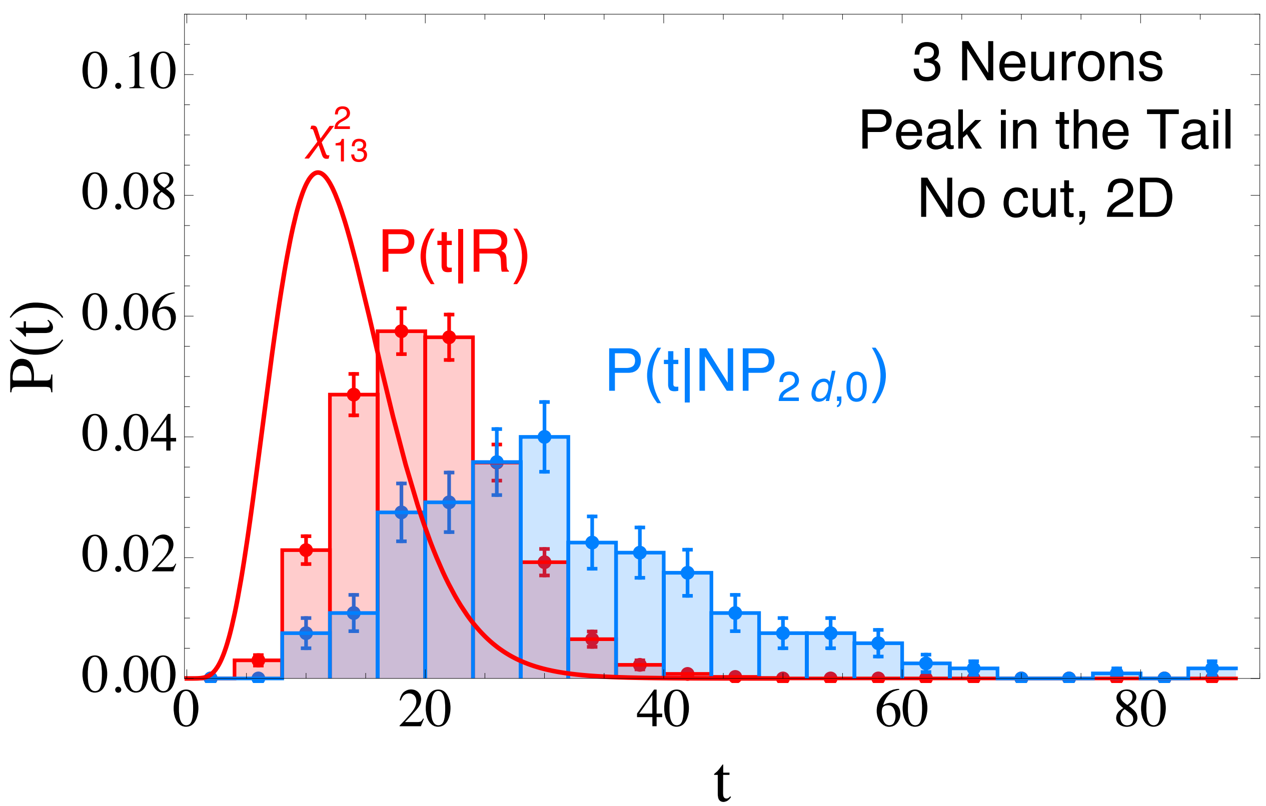
  - MORE WORK IS NEEDED IN THE 2D AND HIGHER-DIMENSIONAL CASE

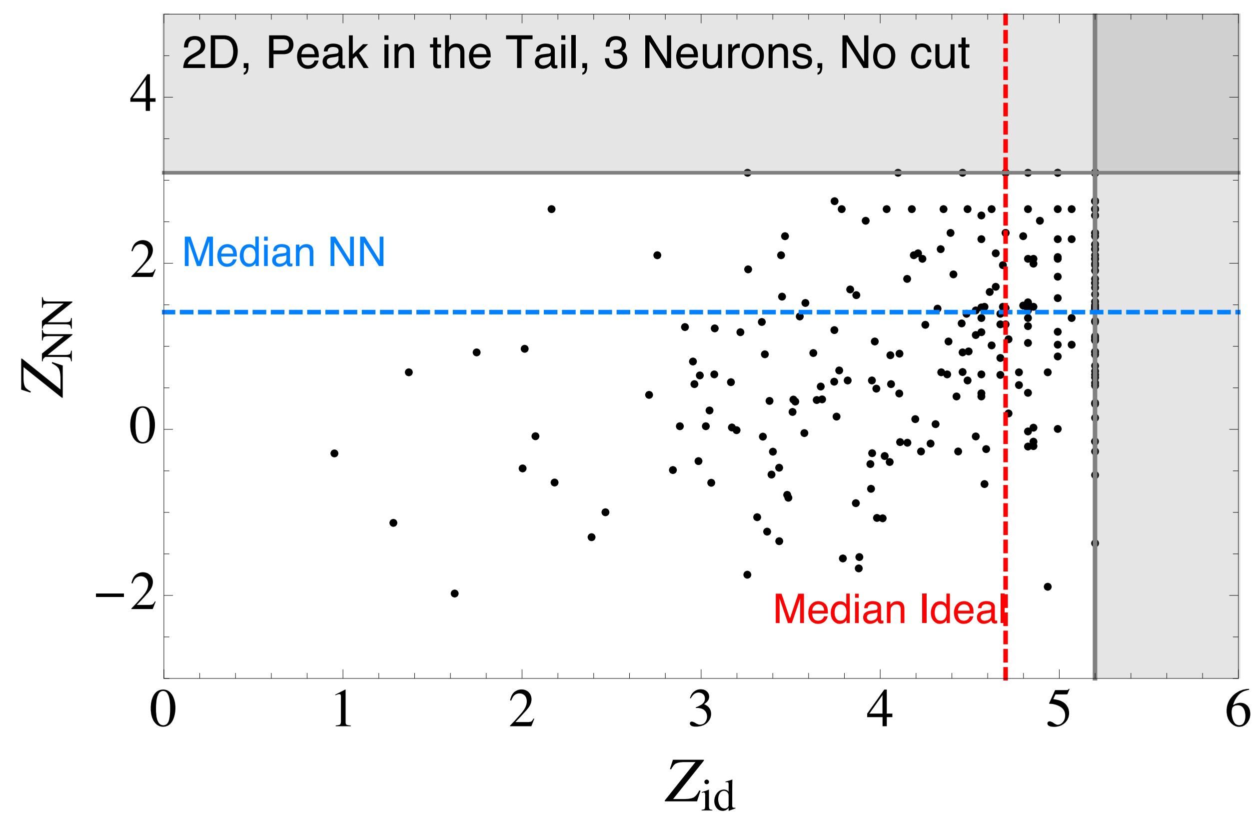
BACKUP

TWO DIMENSIONS

NP: x~EXPONENTIAL+PEAK y~UNIFORM

R: x~EXPONENTIAL y~UNIFORM





RECOVERS COMPARABLE SENSITIVITY TO 1D FOR x>0.3 OR DOUBLING THE EVENTS