Using Machine Learning to Interpret Arrival Directions of Ultra High Energy Cosmic Rays

Oleg Kalashev[1], *Maxim Pshirkov*[1,2], *Mikhail Zotov*[3]

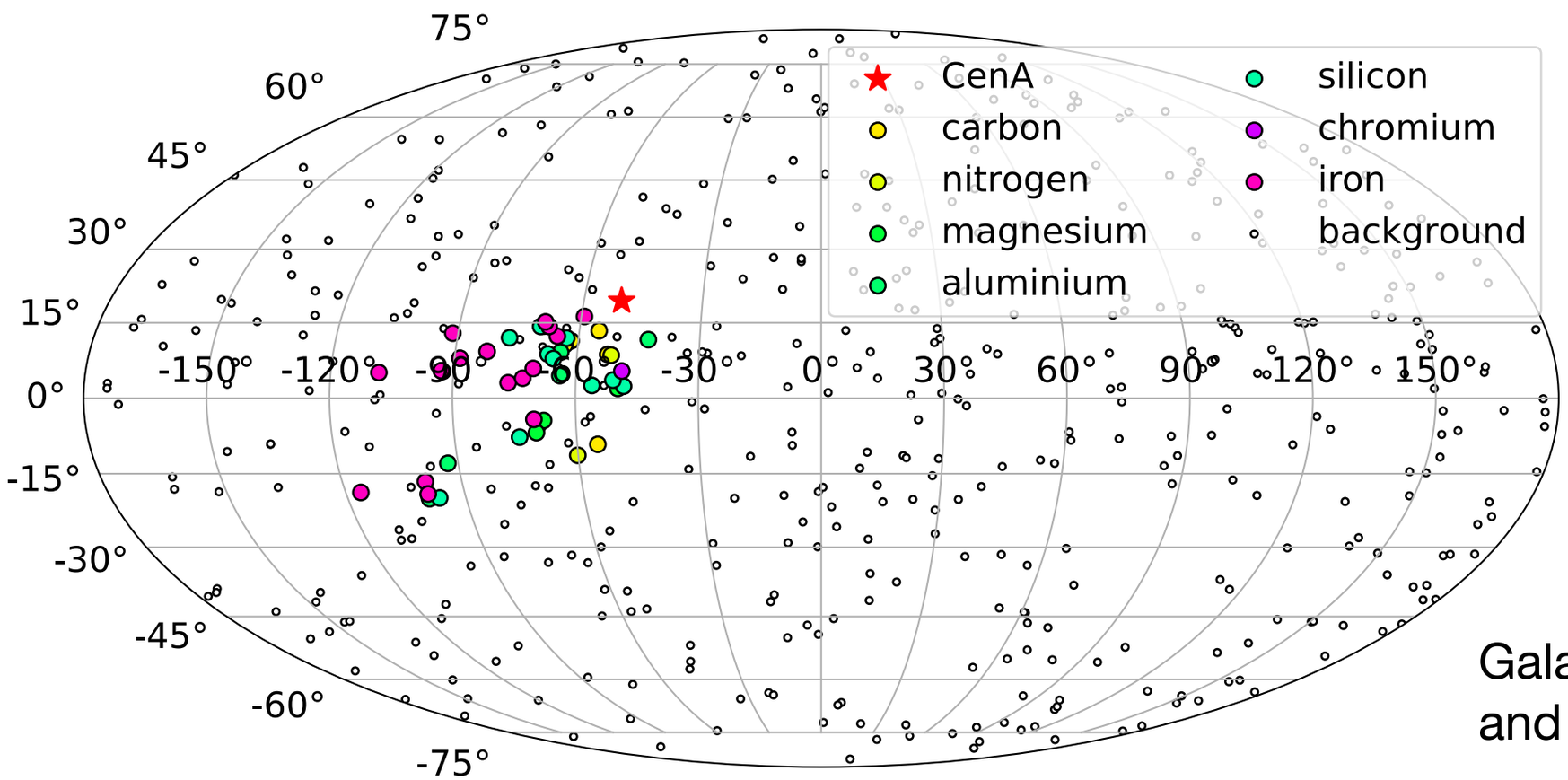
1. INR RAS, Moscow
2. Sternberg Astron. Inst., Moscow
3. SINP, Moscow

Cosmic Ray Anisotropy Workshop CRA2019

8 October 2019, GSSI, L'Aquila

Problem

- null-hypothesis: e.g. isotropic flux
- alternative hypothesis: e.g. (subclass of) AGN as UHECR sources (see talk by M. Zotov) η - fraction of the events from the nearest source; few nearest candidates, galactic magnetic field model
- observables: $\theta_i, \phi_i, i = 1..N$ for cosmic rays with energy above 57 EeV



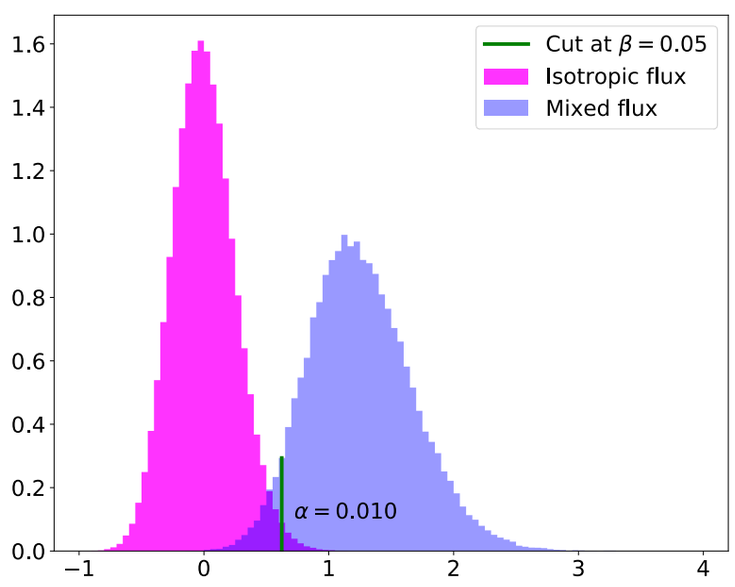
The case of $N_{\mathrm{UHECR}} = 500$ and 9% events coming from Cen A.

Galactic MF model by R. Jansson and G. R. Farrar 2012

Injected spectrum:
M. Kachelrieß et al 2017

Problem

- null-hypothesis: e.g. isotropic flux
- alternative hypothesis: e.g. (subclass of) AGN as UHECR sources (see talk by M. Zotov) η - fraction of the events from the nearest source; few nearest candidates, galactic magnetic field model
- observables: $\theta_i, \phi_i, i = 1..N$ for cosmic rays with energy above 57 EeV
- aim: find optimal test statistic $\xi = f(\theta_i, \phi_i)$ to reject null-hypothesis if alternative hypothesis is true



Type I and II errors:

probability

I : false positive $$\alpha = \int_{\xi_{th}}^{\infty} p_{iso}(\xi) d\xi \leq 0.01$$

II : false negative $$\beta = \int_{-\infty}^{\xi_{th}} p_{alt}(\xi) d\xi \leq 0.05$$

- Find minimal nearest event fraction η from the source to reject isotropy hypothesis

How to choose $\xi = f(\theta_i, \phi_i)$

- **Use angular power spectrum** $\xi = f(C_l)$
  - **e.g.**
    - IceCube, Auger [Hülss, Wiebusch, ICRC2007; Aab et al. JCAP 06 (2017) 026]:

$$\xi \propto \sum_{\ell=1}^{\ell_{max}} \left(\frac{C_\ell - \langle C_{\ell,\mathrm{iso}} \rangle}{\sigma_{l,iso}} \right)^2$$

    - O.K. et al (see talk by M. Zotov)

$$\xi \propto \sum_{\ell=1}^{\ell_{max}} \frac{C_\ell - \langle C_{\ell,\mathrm{iso}} \rangle}{\sigma_{l,iso}}$$

- **Use other function of direct observables**
  - **e.g.**
    - function using expected arrival direction density maps (TA correlation with LSS analysis, see talk by Peter Tinyakov)
    - functions obtained using machine learning ML (this talk)
      - use ML to build classifier which discriminates samples derived assuming null or alternative hypothesis
      - use the classifier output as test statistic ξ

Quick introduction to Supervised Machine Learning

$x \in \mathbb{X}$ **- sample** $x = (x^1, \ldots, x^d)$ **- sample features known**

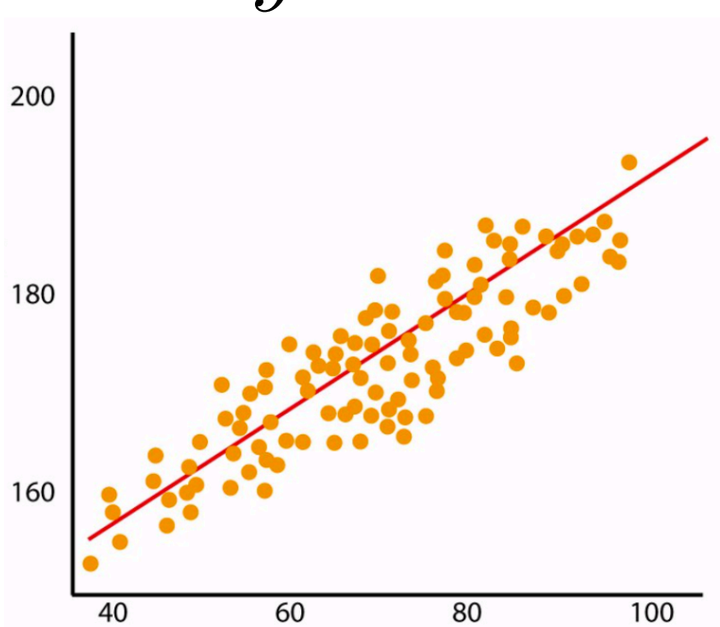
$y \in \mathbb{Y}$ **- answer (sample property which we want to predict)**

$X = (x_i, y_i)_{i=1}^{l}$ **training set is used to find a model** $y = a(x)\,, a \in \mathbb{A}$

by minimising loss (cost) function $Q(a, X)$ **:** $a(x) = argmin_{a \in \mathbb{A}} Q(a, X)$

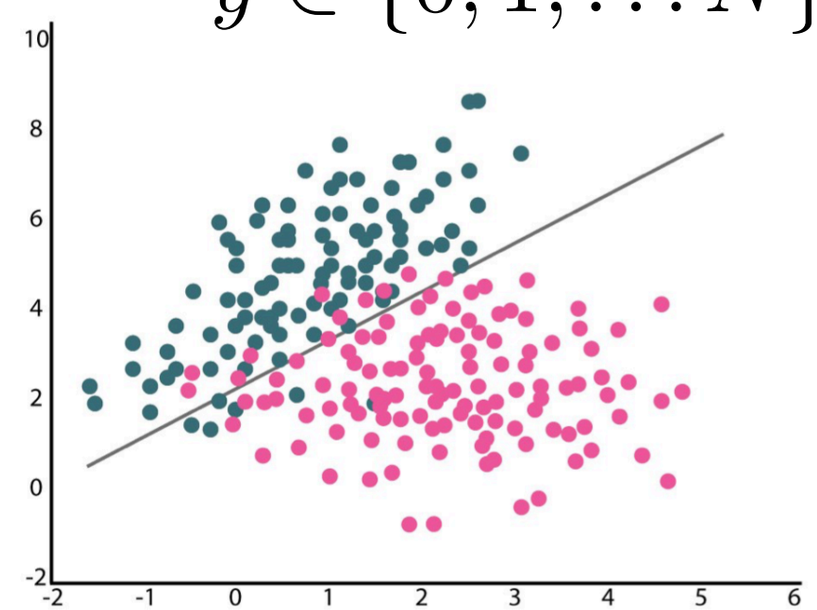
Regression: predict real value

$y \in \mathbb{R}$



Classification: predict class

$y \in \{0, 1, \ldots N\}$



Linear Regression

$$a(x) = w_0 + \sum_{j=1}^{d} w_j x^j$$ **or** $$a(x) = \sum_{j=0}^{d} w_i x^j = \langle w, x \rangle$$ **where** $x_0 \equiv 1$

Choice for Q: $|a(x) - y|$ **- not smooth,** $(a(x) - y)^2$ **is ok**

$$Q(a, x) = \frac{1}{\ell} \sum_{i=1}^{\ell} (a(x_i) - y_i)^2$$

$$Q(w, X) = \frac{1}{\ell} \|Xw - y\|^2 \to \min_w, \qquad X = \begin{pmatrix} x_{11} & \dots & x_{1d} \\ \dots & \dots & \dots \\ x_{\ell 1} & \dots & x_{\ell d} \end{pmatrix}, \quad y = \begin{pmatrix} y_1 \\ \dots \\ y_\ell \end{pmatrix}$$

Analytic solution $w_* = (X^T X)^{-1} X^T y.$ **requires** $O(d^3)$ **operations**

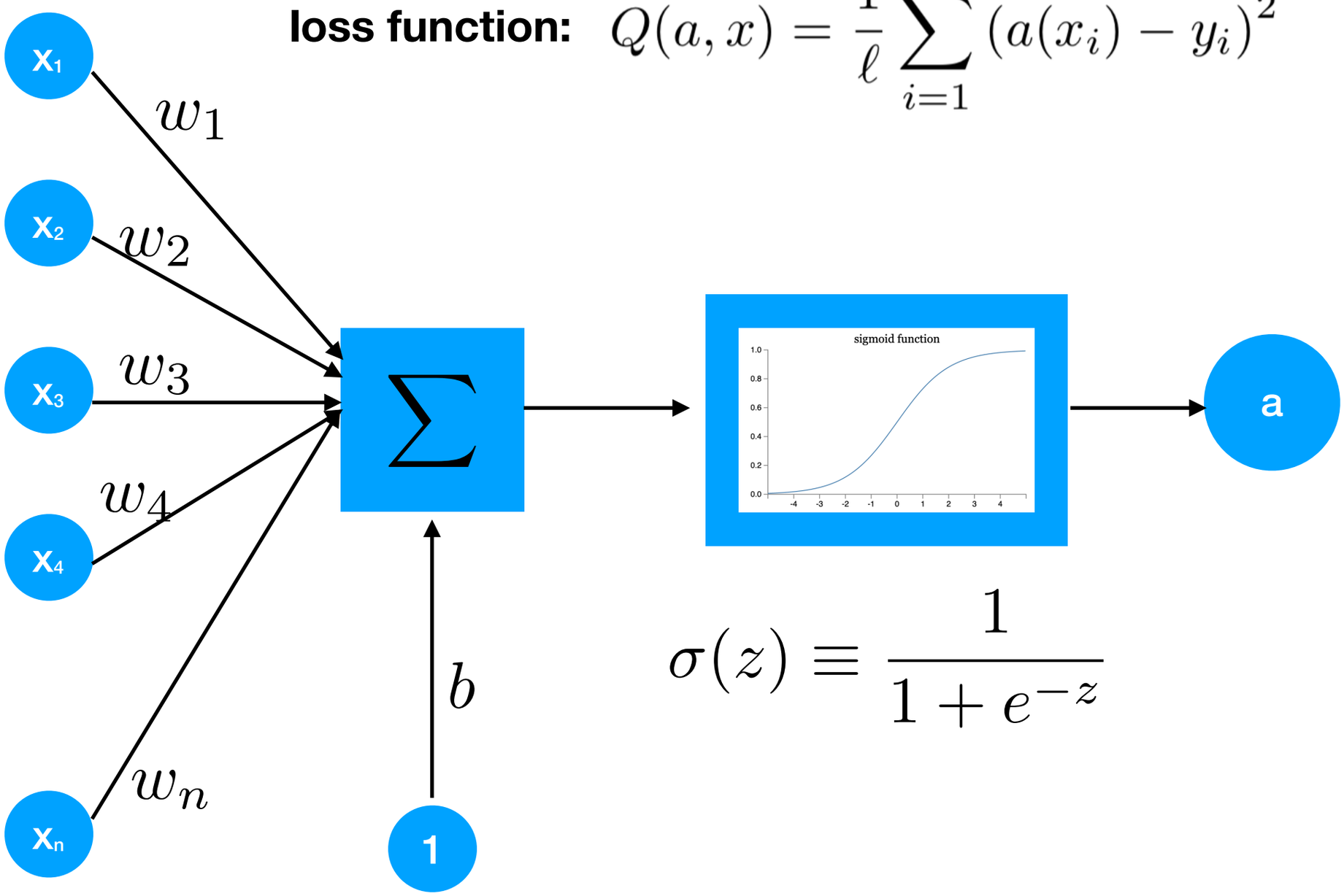
Alternative: numerical optimisation using gradient decent

Artificial Neural Networks

- Sigmoid neuron: $a(x) = \sigma(\sum w_i x_i + b)$

(single layer perceptron)

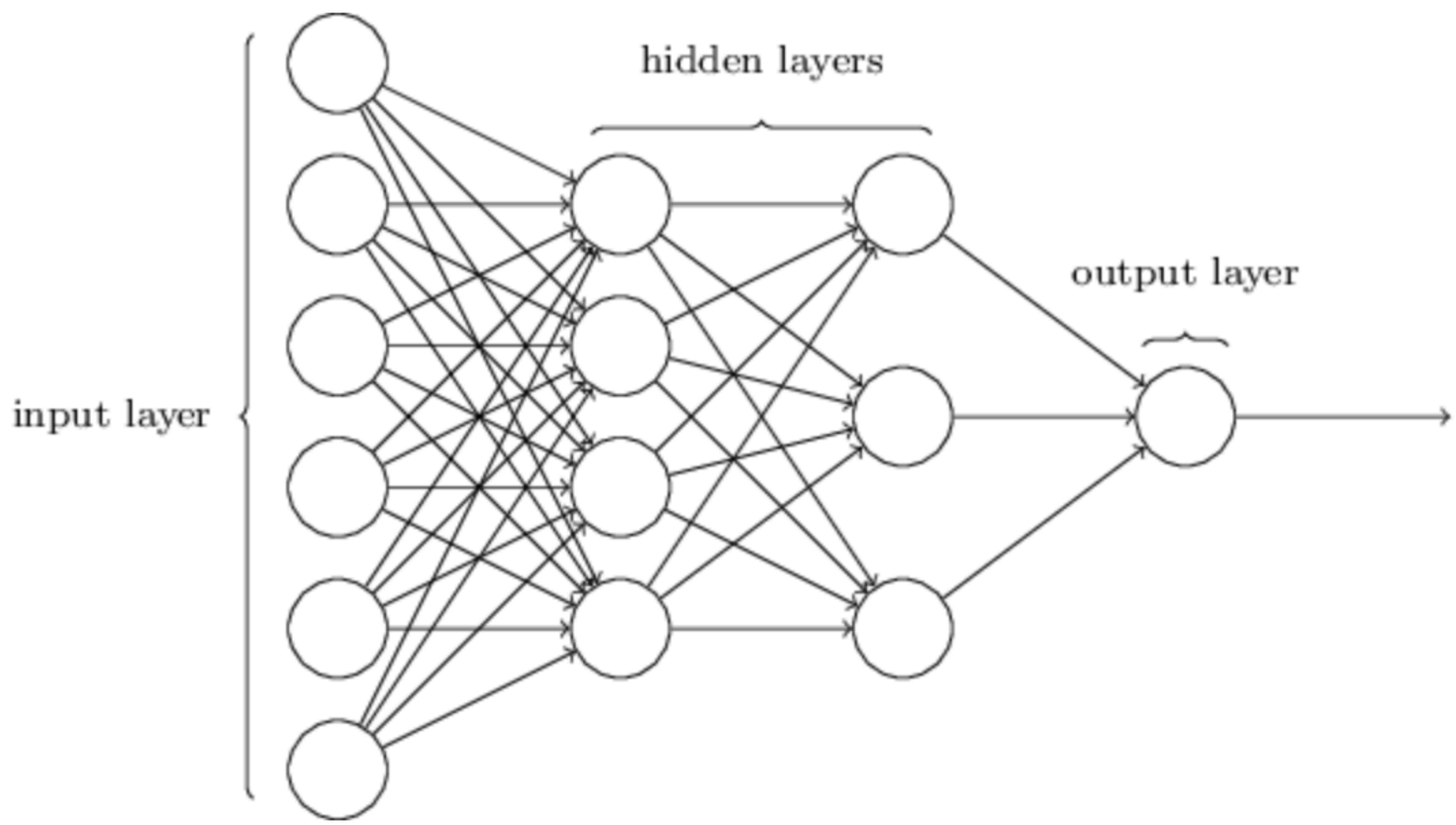
loss function: $Q(a, x) = \frac{1}{\ell} \sum_{i=1}^{\ell} (a(x_i) - y_i)^2$



$$\sigma(z) \equiv \frac{1}{1 + e^{-z}}$$

Artificial Neural Networks

- Multilayer Perceptron



Signal propagation: $$a_j^l = \sigma(z_j^l), \;\; z_j^l = \sum_k w_{jk}^l a_k^{l-1} + b_j^l$$

Vector form: $$a^l = \sigma(w^l a^{l-1} + b^l)$$

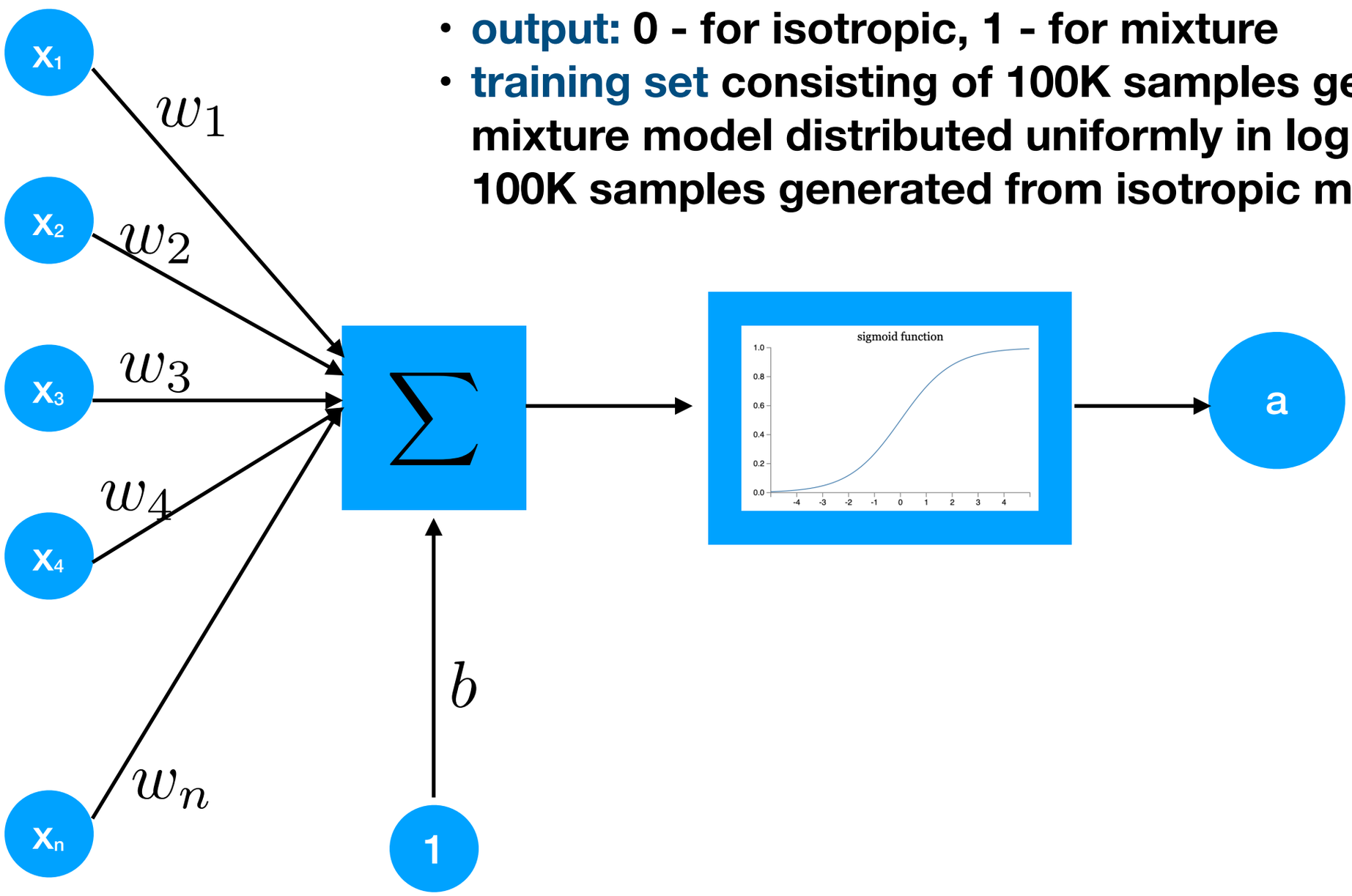
Theorem (K. Hornik, 1991): ANY continuous function can be approximated with ANY precision by MLP

Using toy ANN to build $\xi(C_l)$

- **input features:**

$$x_i = \frac{C_l - \langle C_{l,iso} \rangle}{\sigma_{l,iso}}, l \leq 32$$

- **output: 0 - for isotropic, 1 - for mixture**
- **training set consisting of 100K samples generated in mixture model distributed uniformly in log η and 100K samples generated from isotropic model**

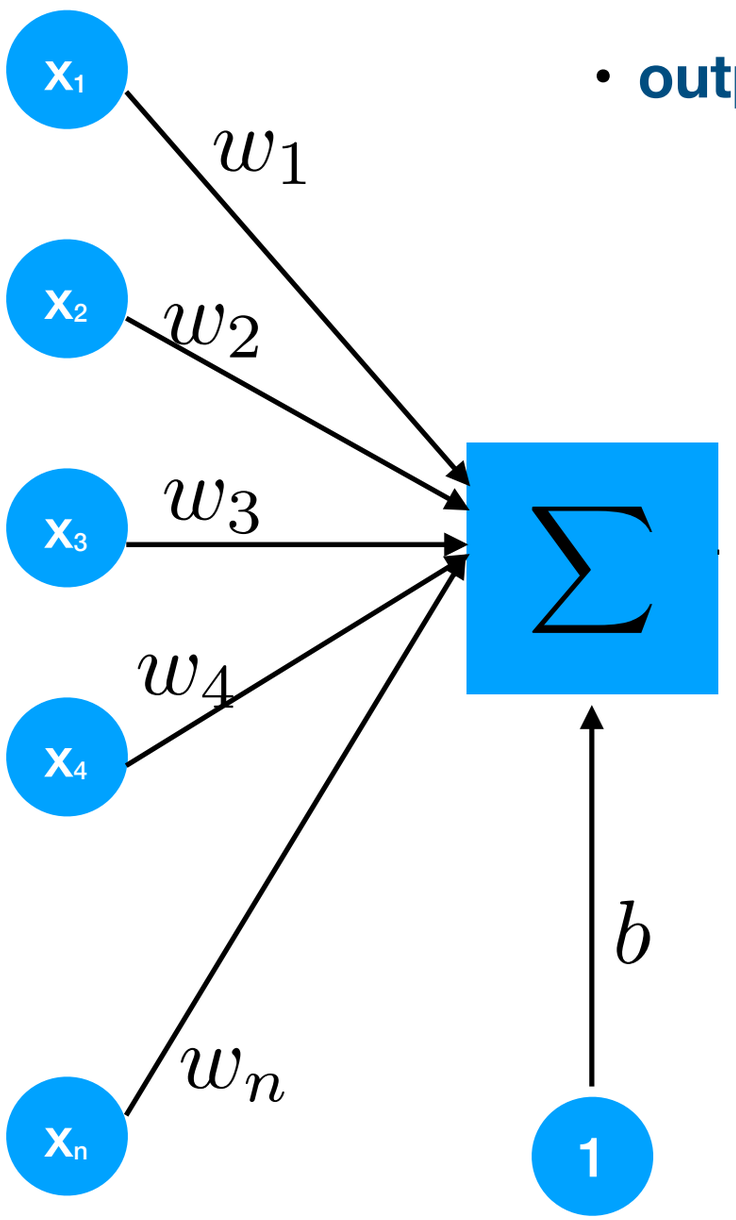


Using toy ANN to build $\xi(C_l)$

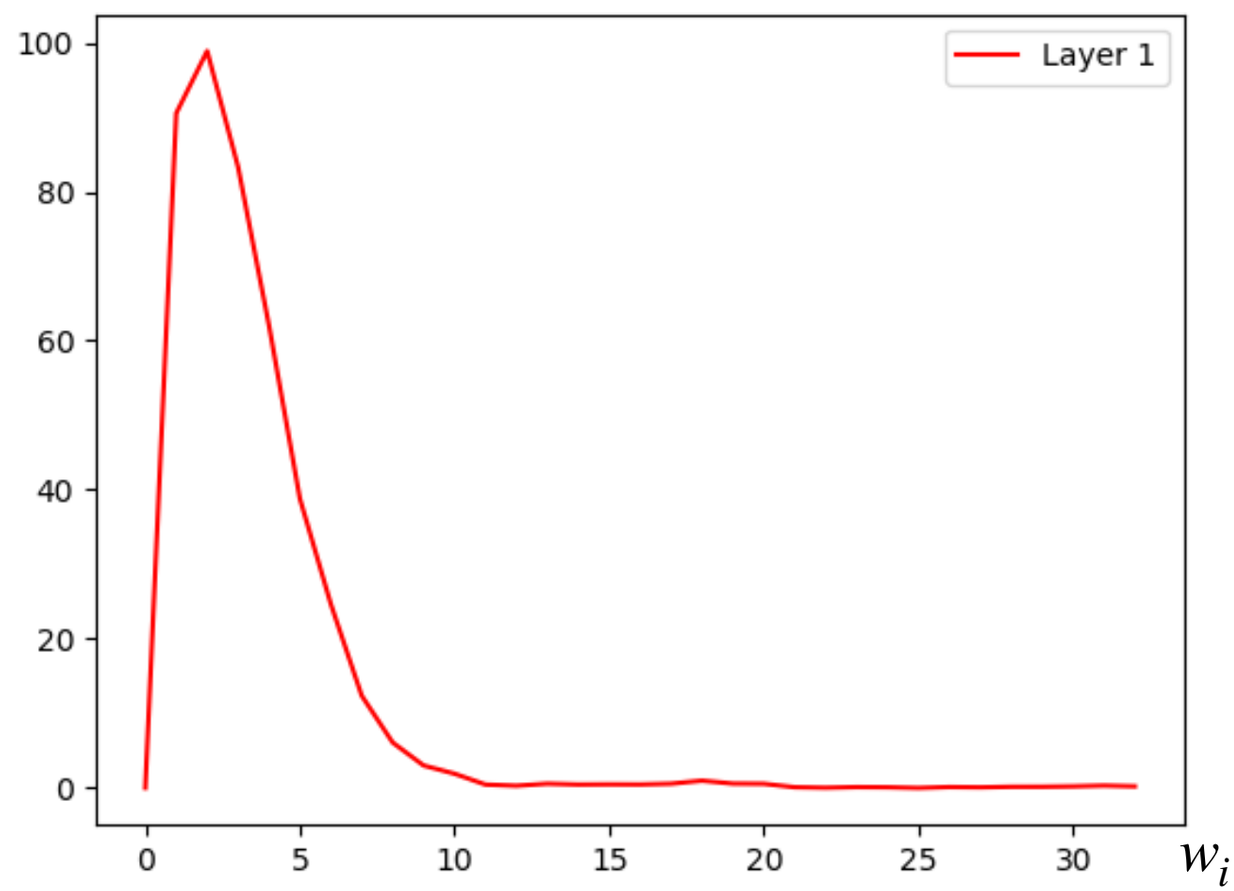
- **input features:**

$$x_i = \frac{C_l - \langle C_{l,iso} \rangle}{\sigma_{l,iso}}, l \leq 32$$

- **output:** **0 - for isotropic, 1 - for mixture**



Learned weights



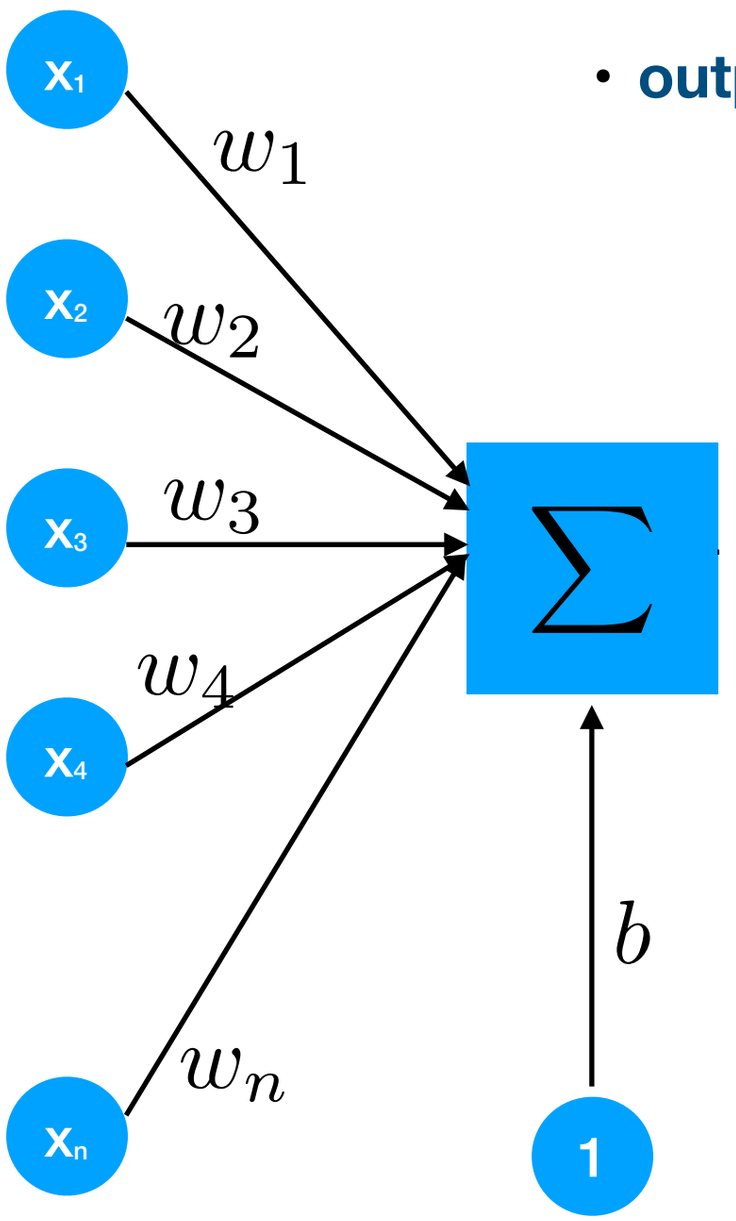
Using toy ANN to build $\xi(C_l)$

Test statistics used in original analysis:

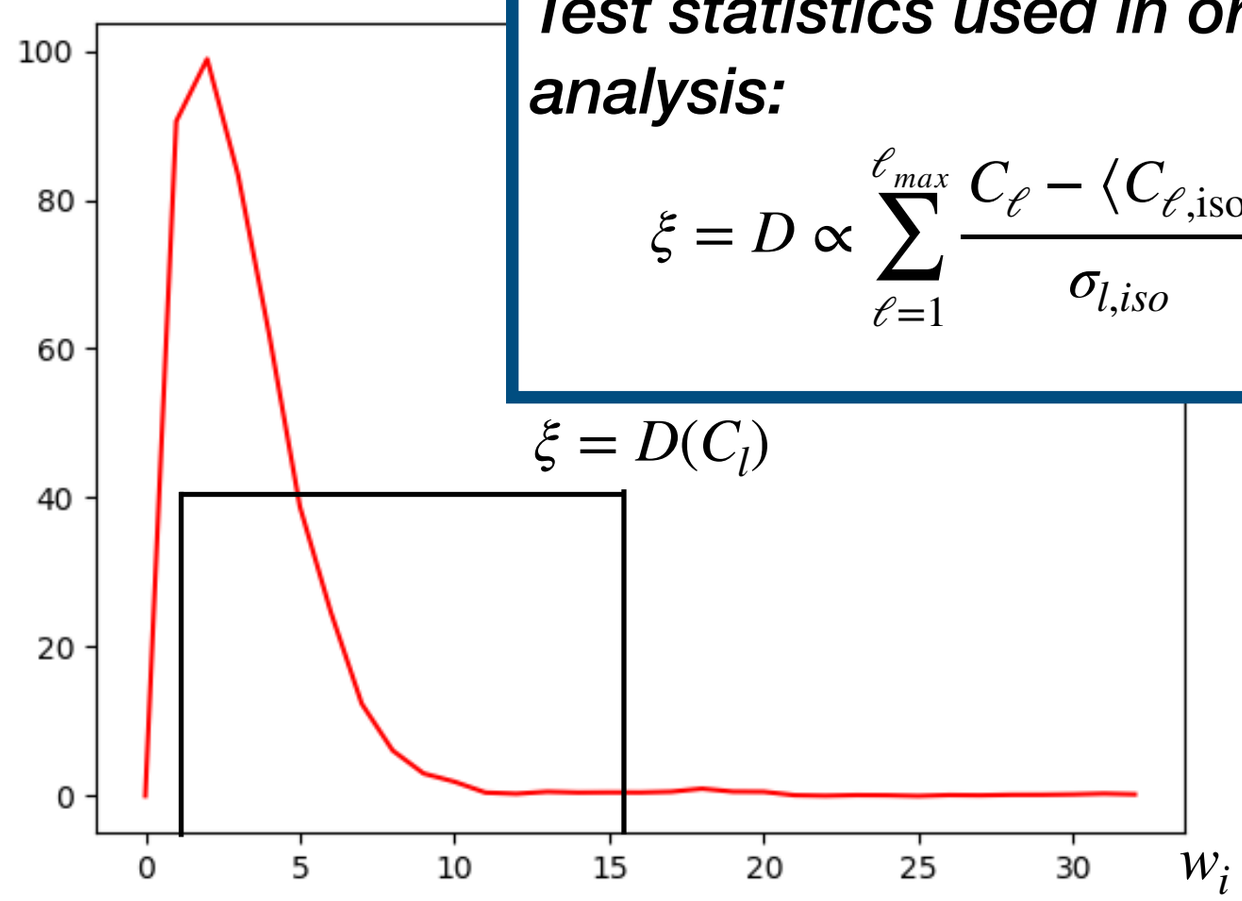
- **input features:**

$$x_i = \frac{C_l - \langle C_{l,iso} \rangle}{\sigma_{l,iso}}, l \leq 32$$

- **output:** **0 - for isotropic, 1 - for mixture**



Learned weights



Test statistics used in original analysis:

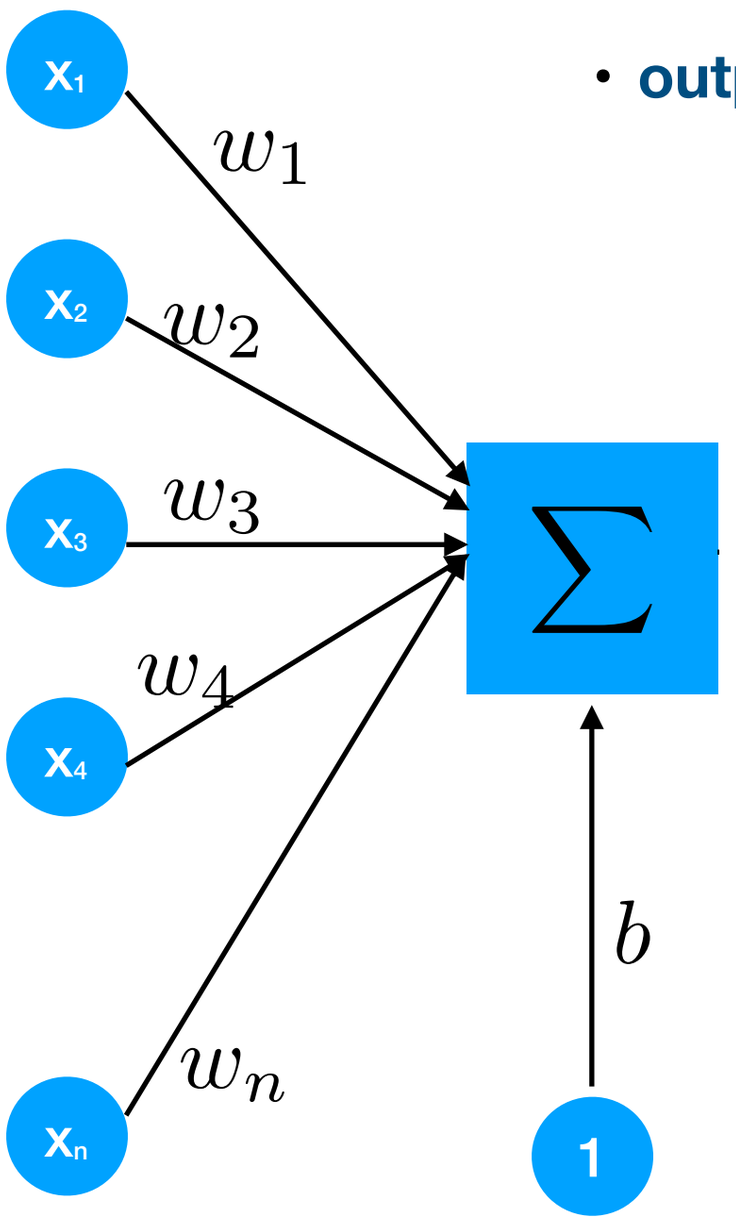
$$\xi = D \propto \sum_{\ell=1}^{\ell_{max}} \frac{C_\ell - \langle C_{\ell,\mathrm{iso}} \rangle}{\sigma_{l,iso}}, l_{max} = 16$$

Using toy ANN to build $\xi(C_l)$

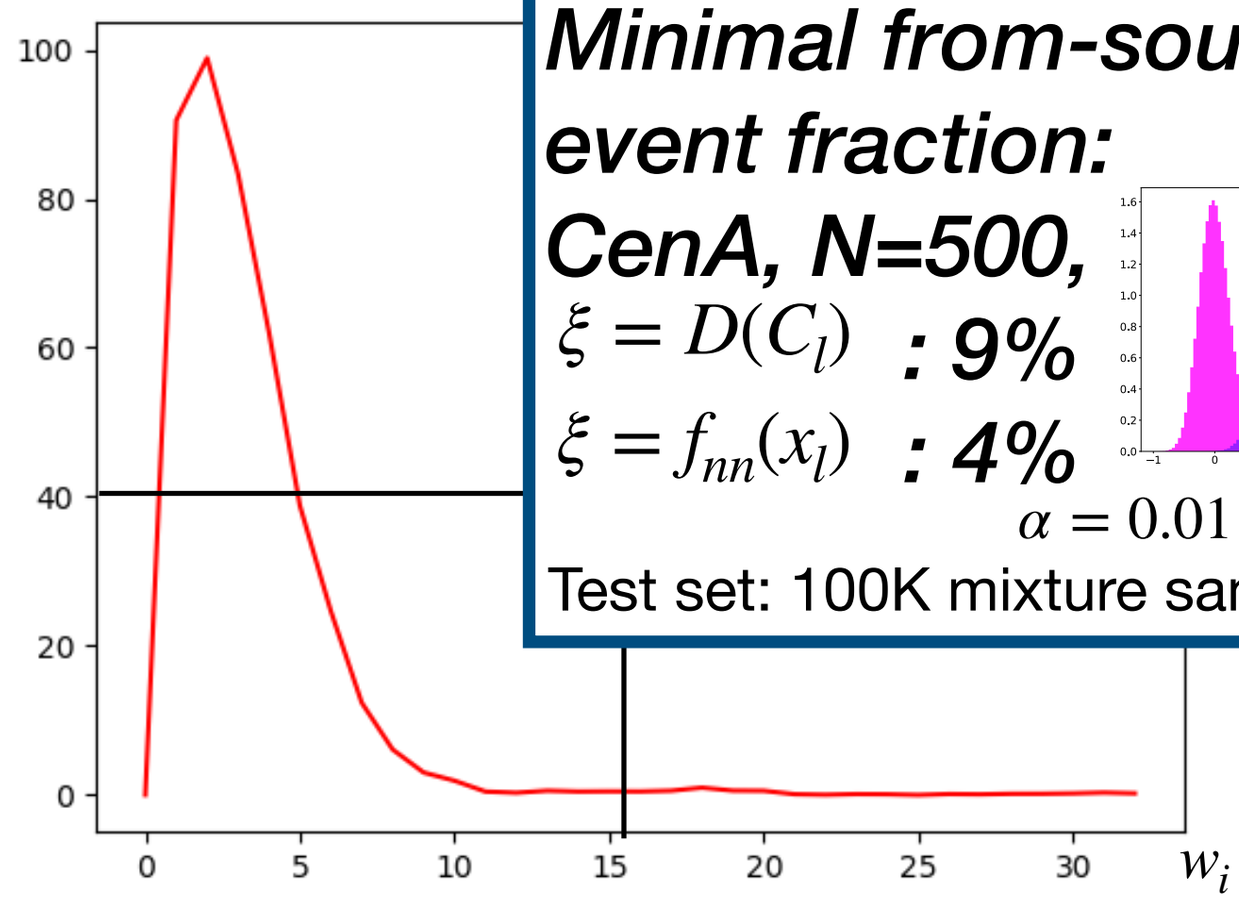
- **input features:**

$$x_i = \frac{C_l - \langle C_{l,iso} \rangle}{\sigma_{l,iso}}, l \leq 32$$

- **output:** **0 - for isotropic, 1 - for mixture**



Learned weights

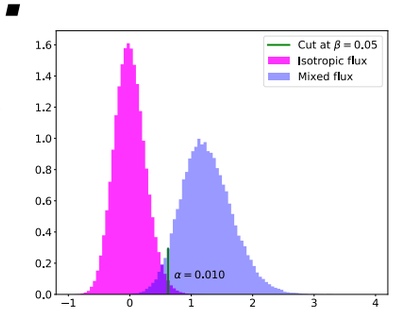


Minimal from-source event fraction:

CenA, N=500,

$\xi = D(C_l)$ ***: 9%***

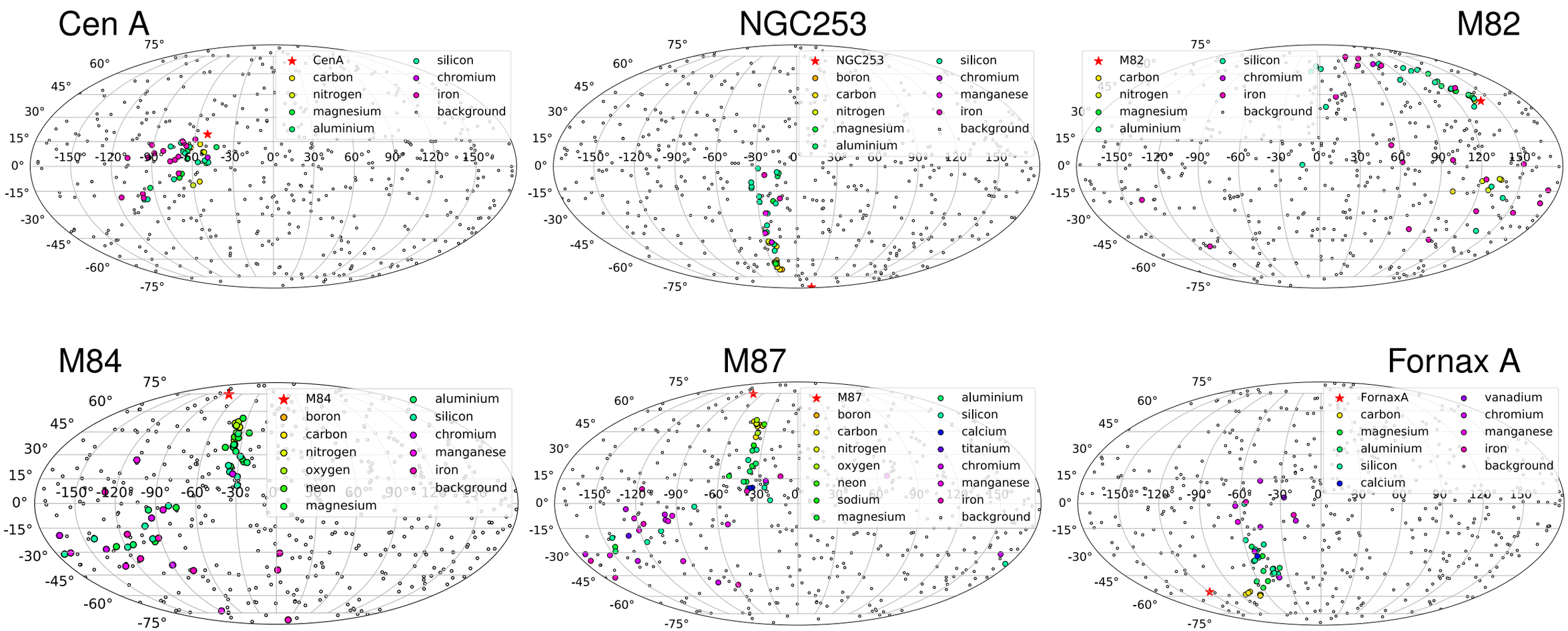
$\xi = f_{nn}(x_l)$ ***: 4%***



$\alpha = 0.01, \beta = 0.05$

Test set: 100K mixture samples

Can we do better?



Implement pattern recognition using HEALPix map of CR arrival directions as input

ANN using HEALPix map as input

Convolutional ANN - universal tool for image recognition

- Handwritten digits recognition

MNIST database
(“Modified National Institute of Standards and Technology”)

- Classification of galaxies using images from Sloan Digital Sky Survey

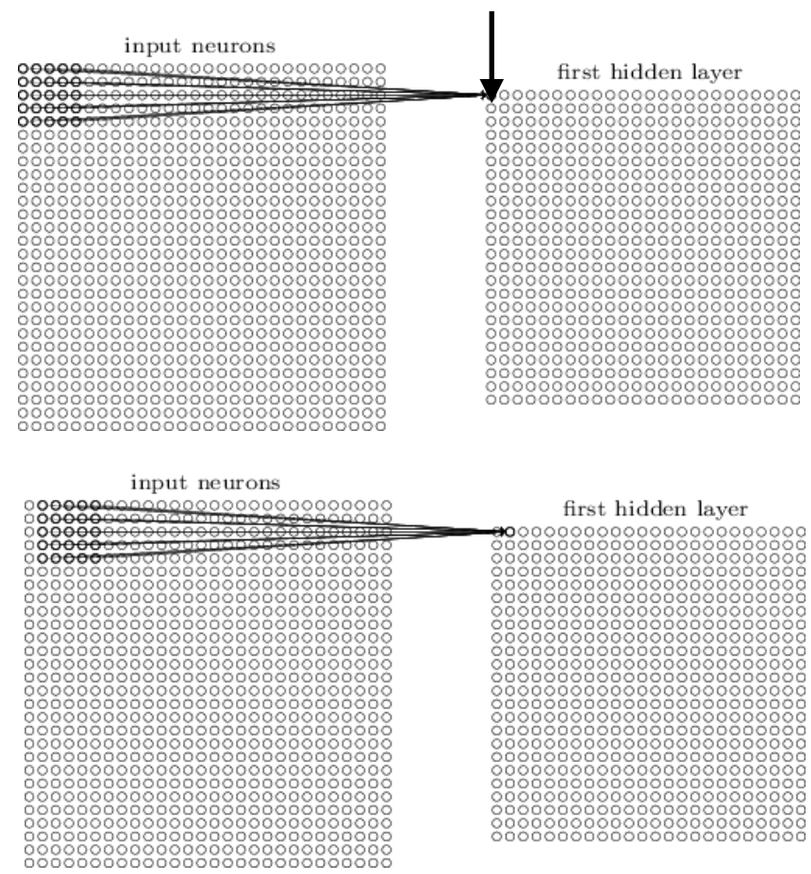
Galaxy Zoo Challenge kaggle competition

Convolutional ANN

The main idea: extract local features and build their maps

- Convolutional kernel usually has small size (compared to image)

$$\sigma\left(b + \sum_{l=0}^{4}\sum_{m=0}^{4} w_{l,m} a_{j+l,k+m}\right)$$



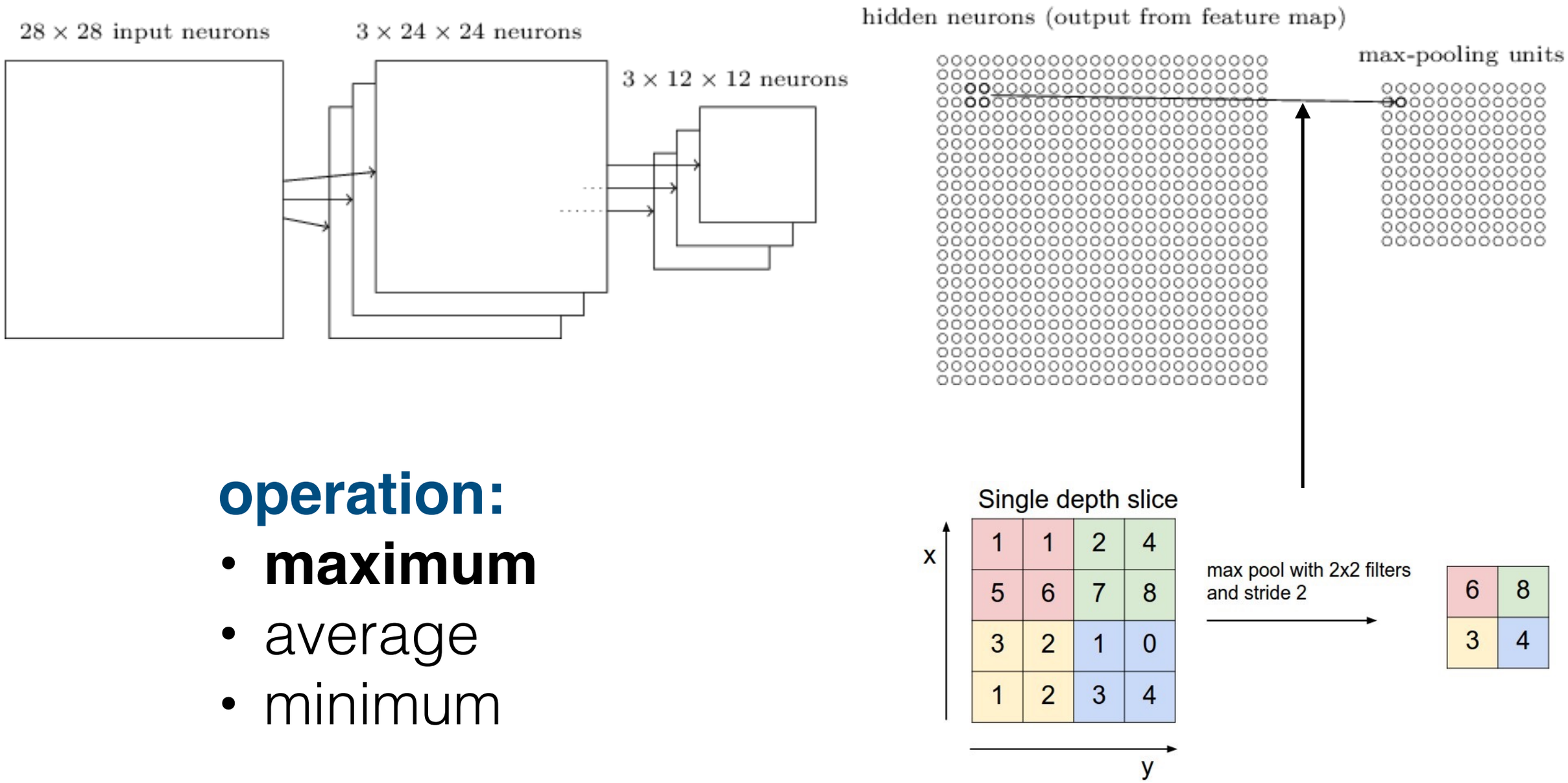
$1_{\times 1}$	$1_{\times 0}$	$1_{\times 1}$	0	0
$0_{\times 0}$	$1_{\times 1}$	$1_{\times 0}$	1	0
$0_{\times 1}$	$0_{\times 0}$	$1_{\times 1}$	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved Feature

Pooling (downsampling) layer



operation:

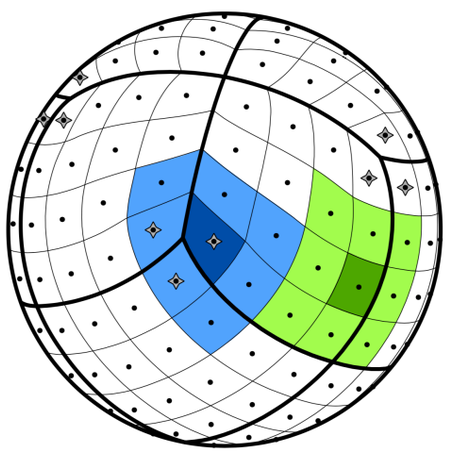
- **maximum**
- average
- minimum

Local translational invariance is achieved via convolution-pool combination

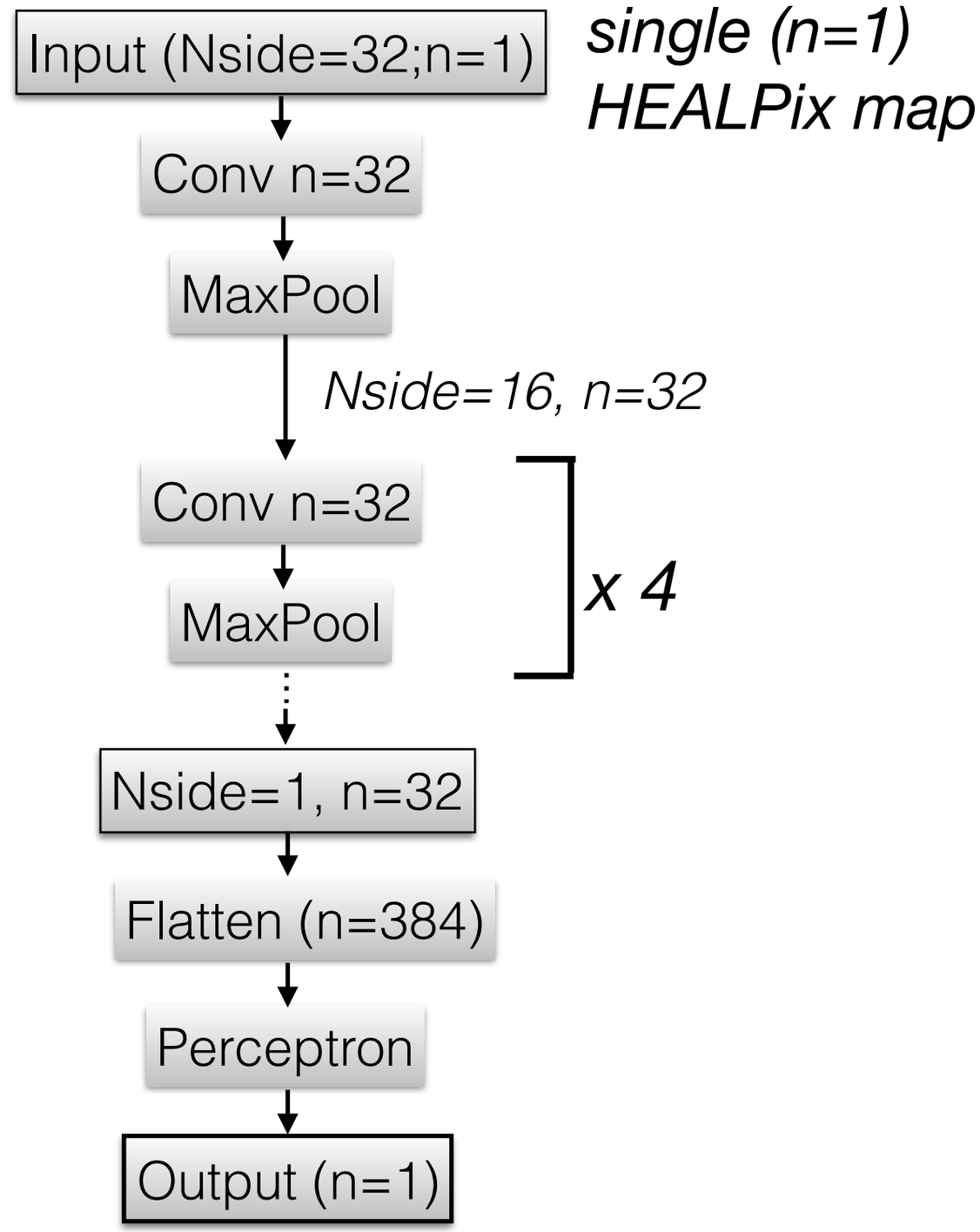
Convolutional NN on sphere

Several implementations exist:

- *Taco S. Cohen et al 2018*
- *Nathanaël Perraudin et al 2018*
- *N. Krachmalnicoff et al 2019 (used in this work)*



CNN on HEALPix map



Minimal from-source event fraction in % to reject isotropy hypothesis

using

$$\xi = D \propto \sum_{\ell=1}^{\ell_{max}} \frac{C_\ell - \langle C_{\ell,\mathrm{iso}} \rangle}{\sigma_{l,iso}}$$

N_{UHECR}	100	200	300	400	500
NGC 253	17	12	10	8	7
Cen A	21	14	12	10	9
M82	26	18	14	12	11
M87	29	20	16	14	12
Fornax A	19	13	11	9	8

using $\xi_{cnn} = f_{CNN}(\{\theta_i, \phi_i\})$ **given by the output of convolutional NN on HEALPix grid**

$\alpha = 0.01\,, \beta = 0.05$

N_{UHECR}	100	200	300	400	500
NGC 253	4	2.5	2	1.75	1.2
Cen A	6	4	2.7	2.5	2
M82	8	4.5	3.6	3	2.4
M87	9	5	3.3	2.5	2.4
Fornax A	5	3	2.3	2	1.6

How robust is ξ_{cnn} with respect to galactic magnetic field model

- **Train Conv NN with samples generated assuming MF model by *R. Jansson and G. R. Farrar 2012***
  - **obtain** $\xi_{cnn} = f^{JF}_{CNN}(\{\theta_i, \phi_i\}_{JF})$ *(minimal fraction shown on previous slide)*
- **Apply** f^{JF}_{CNN} **to samples generated assuming MF model by *M. Pshirkov et al 2011***
  - **obtain** $\xi_{cnn} = f^{JF}_{CNN}(\{\theta_i, \phi_i\}_{P})$

$\alpha = 0.01\,, \beta = 0.05$

$$\xi = D \propto \sum_{\ell=1}^{\ell_{max}} \frac{C_\ell - \langle C_{\ell,\mathrm{iso}} \rangle}{\sigma_{l,iso}}$$

N_{UHECR}	100	200	300	400	500
Cen A	21	14	12	10	9
Fornax A	19	13	11	9	8

$$\xi_{cnn} = f^{JF}_{CNN}(\{\theta_i, \phi_i\}_{JF})$$

N_{UHECR}	100	200	300	400	500
Cen A	6	4	2.7	2.5	2
Fornax A	5	3	2.3	2	1.6

The effect strongly depends on direction to the source

$$\xi_{cnn} = f^{JF}_{CNN}(\{\theta_i, \phi_i\}_{P})$$

N_{UHECR}	100	200	300	400	500
Cen A	9	6	4	3.25	2.6
Fornax A	11	7.5	6	3.5	3.2

Conclusions

- **ANNs provide efficient universal way to define test statistic as function of direct observables. In case of arrival directions interpretation, convolutional NN on HEALPix grid can be used**

- **pros:**

  - **optimal (trained to discriminate two hypothesis)**

  - **easy to apply to nonuniform exposure data (e.g. TA and Auger)**

- **cons:**

  - **not easy to interpret (see however perceptron sample)**

Thank you

The work is supported by the Russian Science Foundation grant 17-72-20291

Backup slides

Alternative Hypothesis

For details, see talk by M. Zotov

M. Kachelrieß, O. Kalashev, S. Ostapchenko, D.V. Semikoz "A minimal model for extragalactic cosmic rays and neutrinos," PRD 96, 083006 (2017); arXiv:1704.06893.

Basic assumptions:

- UHECRs are accelerated by (a subclass of) AGN
- the energy spectra of nuclei after the acceleration phase follow a power-law with a rigidity-dependent cutoff: $j_{\rm inj}(E) \propto E^{-\alpha} \exp[-E/(ZE_{\rm max})]$
- the CR nuclei diffuse first through a zone dominated by photo-hadronic interactions, and then they escape into a second zone dominated by hadronic interactions with gas.

The model matches:

- experimental data on the total CR flux, the mean EAS maximum depth $X_{\rm max}$ and its width RMS($X_{\rm max}$) above $\sim 10^{17}$ eV
- HE neutrino flux measured by IceCube

One of the consequences: there should be a source of UHECRs within $\sim$ 20 Mpc

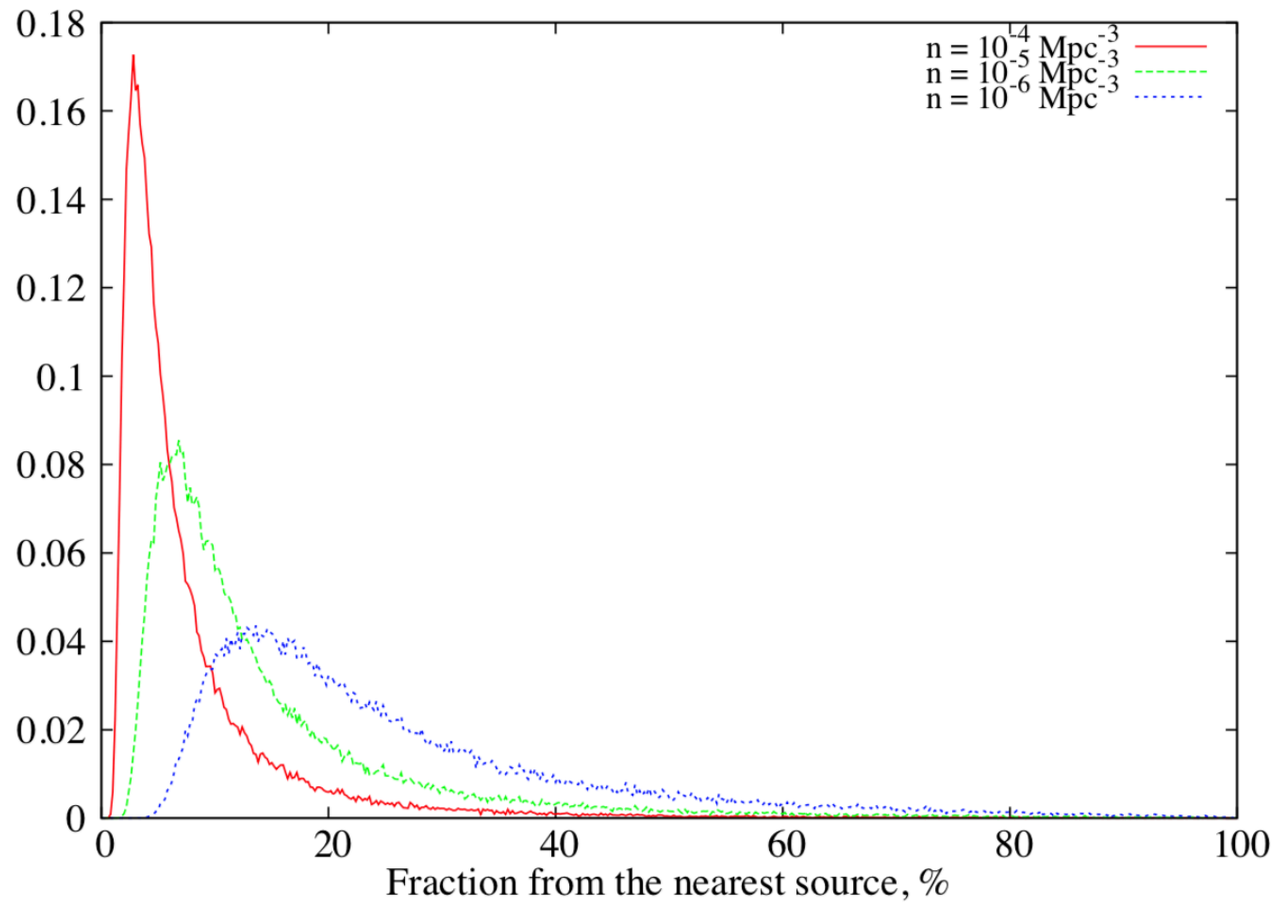
Alternative Hypothesis

The main idea of the study **From the talk by M. Zotov**

0. K-EUSO is expected to register from 120 to 500 CRs at $E \gtrsim 57$ EeV in 2 years
   [M. Casolino *et al.* PoS (ICRC2017) 368]
1. Assume EECRs propagate from a source to the Galaxy in the ballistic regime, so that deflections from the direction to the source are $\leq 2^\circ$.
2. Take a spectrum at the source and obtain a spectrum after propagating to the Galaxy: `TransportCR` code
   [O. Kalashev, E. Kido, J. Exp. Theor. Phys. 120 (2015) 790; arXiv:1406.0735]
3. Propagate CRs from the boundary of the Galaxy to the Earth with `CRPropa3`
   [R. Alves Batista *et al.* JCAP 05 (2016) 038; arXiv:1603.07142]
   assuming the Jansson–Farrar (2012) GMF model (actually backtrack)
4. Study large-scale anisotropy of CRs with the angular power spectrum.

Five AGN were chosen for the analysis: Cen A, NGC253, M82, M87, Fornax A.

Expected fraction of events from the nearest source



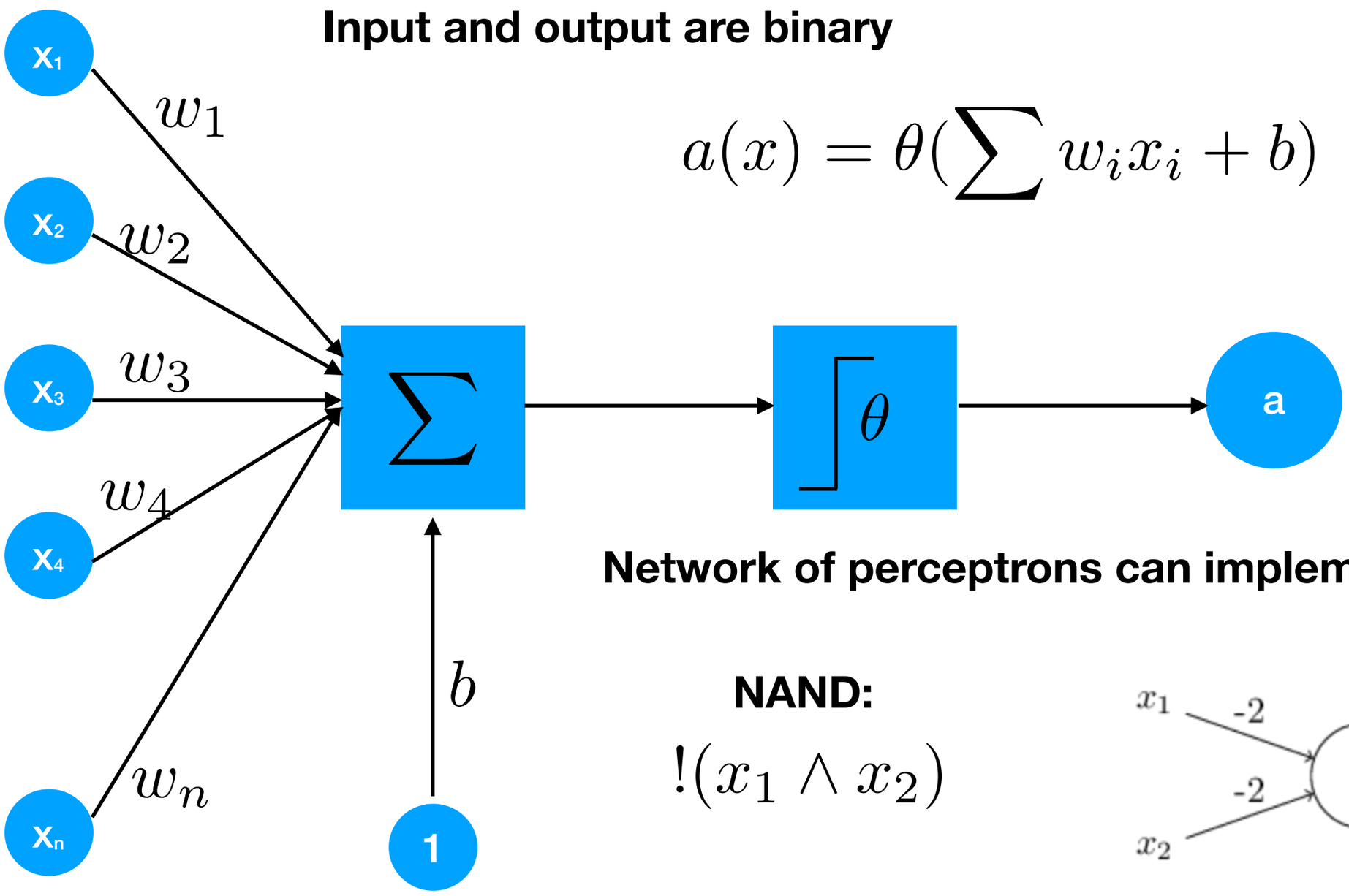
n, Mpc^{-3}	Closest	Second closest
10^{-4}	5.2	1.8
3×10^{-5}	7.5	2.7
10^{-5}	10.6	3.9
3×10^{-6}	15.0	5.0
10^{-6}	20.9	6.3

Average from-source flux fraction

Artificial Neural Networks

- Perceptron model.

Walter Pitts, Warren McCulloch (1943)



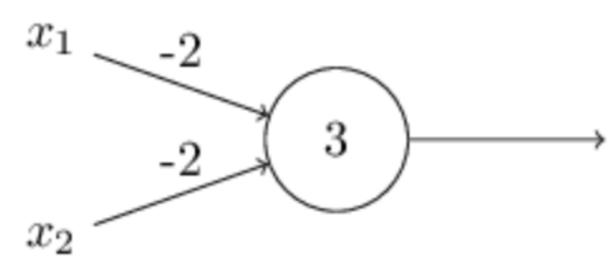
Input and output are binary

$$a(x) = \theta(\sum w_i x_i + b)$$

Network of perceptrons can implement binary logic

NAND:

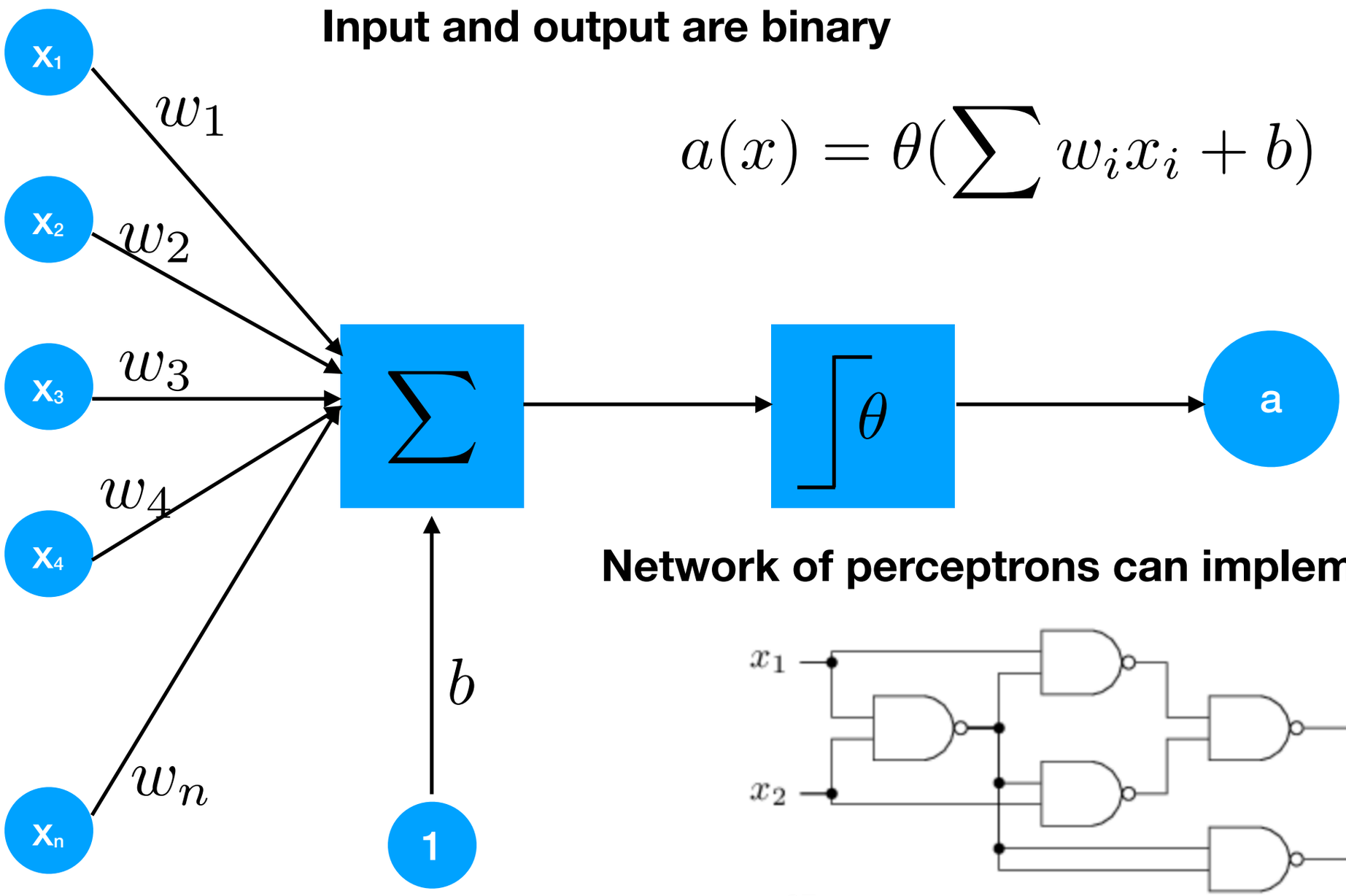
$$!(x_1 \wedge x_2)$$



Artificial Neural Networks

- Perceptron model.

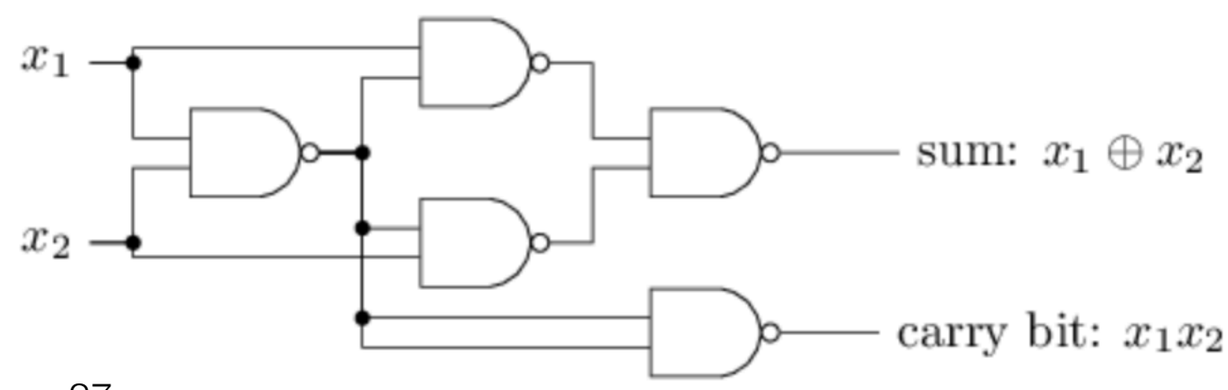
Walter Pitts, Warren McCulloch (1943)



Input and output are binary

$$a(x) = \theta(\sum w_i x_i + b)$$

Network of perceptrons can implement binary logic



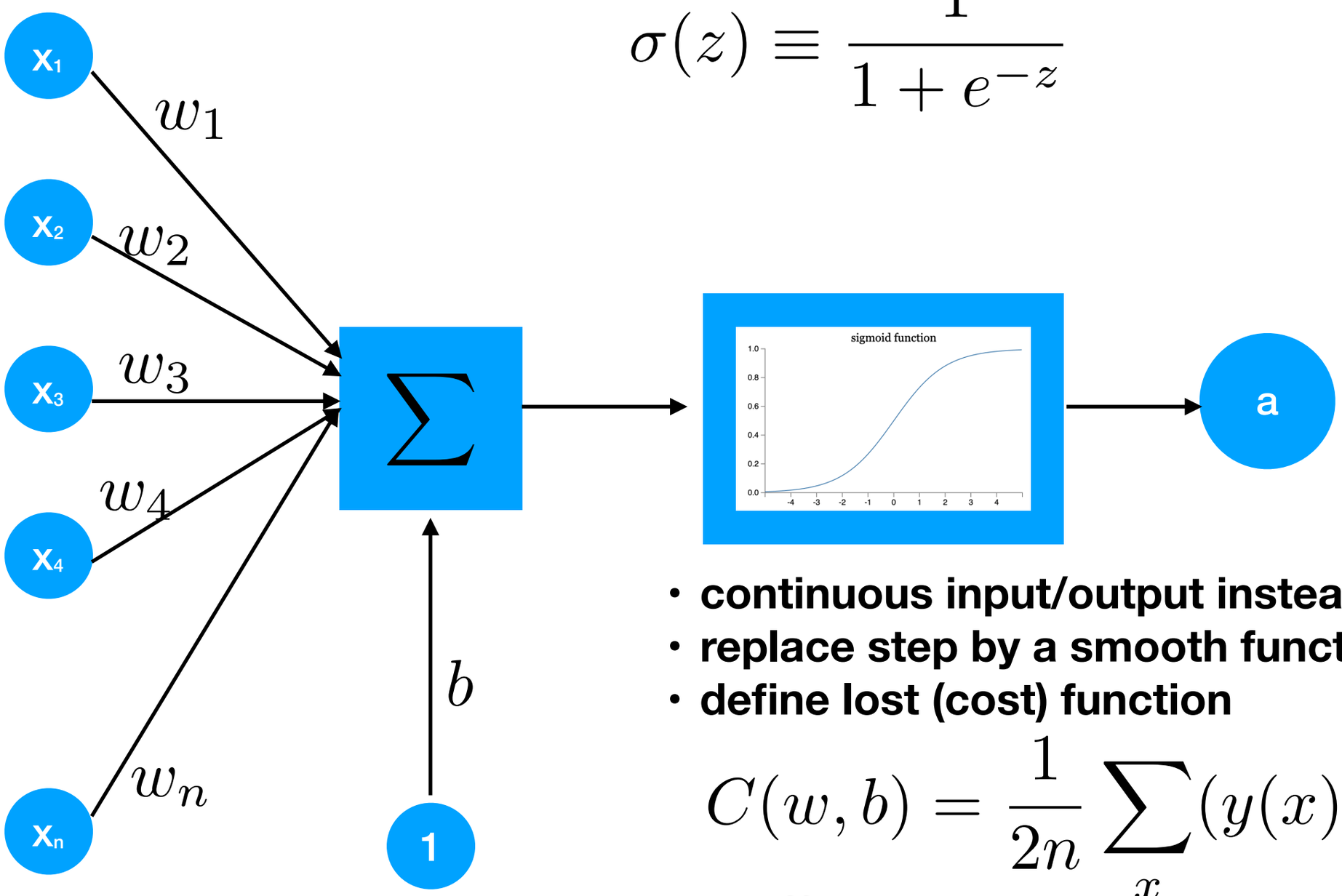
Artificial Neural Networks

- Sigmoid neuron

$$a(x) = \sigma(\sum w_i x_i + b)$$

We want to train the weights.

$$\sigma(z) \equiv \frac{1}{1+e^{-z}}$$



- **continuous input/output instead of binary**
- **replace step by a smooth function**
- **define lost (cost) function**

$$C(w,b) = \frac{1}{2n}\sum_x (y(x) - a)^2$$

Convolutional layer

building feature maps

0	0	0	0	0	0	...
0	156	155	156	158	158	...
0	153	154	157	159	159	...
0	149	151	155	158	159	...
0	146	146	149	153	158	...
0	145	143	143	148	158	...
...	...	...	...	...	...	...

Input Channel #1 (Red)

0	0	0	0	0	0	...
0	167	166	167	169	169	...
0	164	165	168	170	170	...
0	160	162	166	169	170	...
0	156	156	159	163	168	...
0	155	153	153	158	168	...
...	...	...	...	...	...	...

Input Channel #2 (Green)

0	0	0	0	0	0	...
0	163	162	163	165	165	...
0	160	161	164	166	166	...
0	156	158	162	165	166	...
0	155	155	158	162	167	...
0	154	152	152	157	167	...
...	...	...	...	...	...	...

Input Channel #3 (Blue)

padding: ‘same’

output image size is equal to input image size

-1	-1	1
0	1	-1
0	1	1

Kernel Channel #1

1	0	0
1	-1	-1
1	0	-1

Kernel Channel #2

0	1	1
0	1	0
1	-1	1

Kernel Channel #3

$$308 + -498 + 164 + 1 = -25$$

Bias = 1

Output

-25				...
				...
				...
				...
...	...	...	...	...

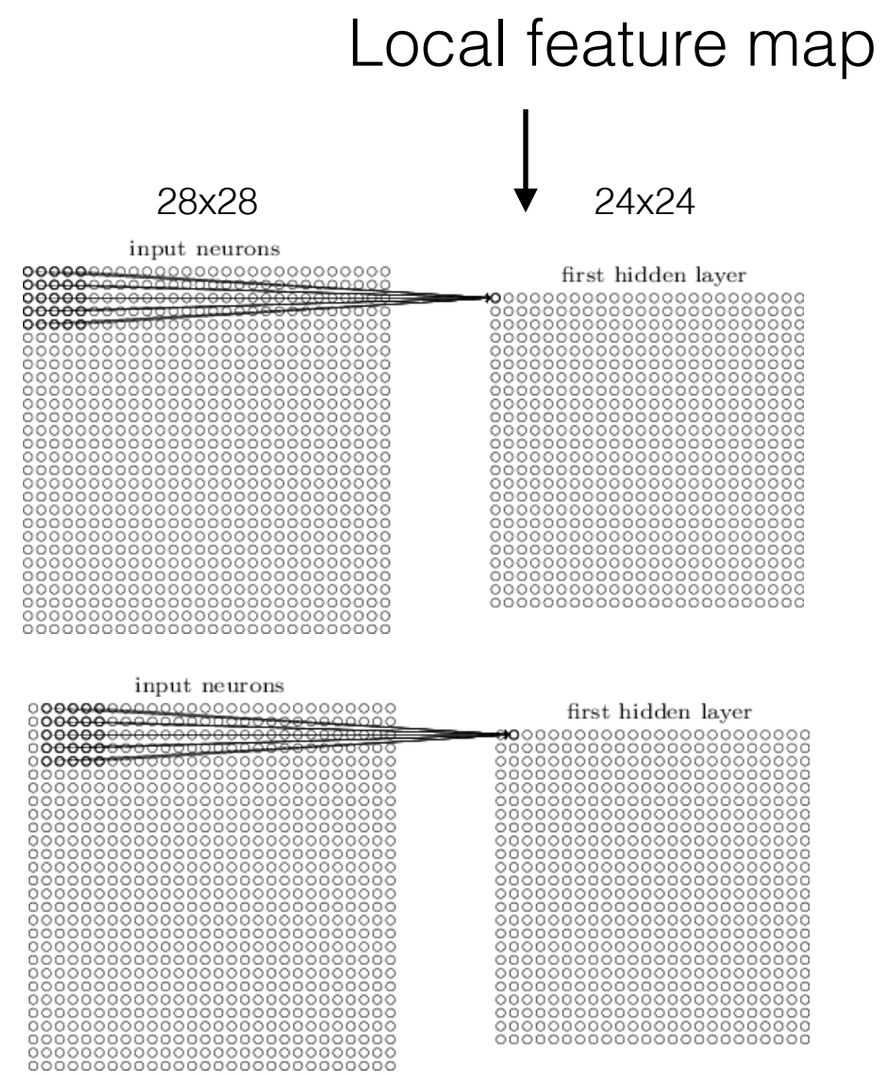
$$a^{l+1}_{d,m,n} = \sigma(b_d + \sum_{c,\alpha,\beta} W_{dc\alpha\beta} a^{l}_{c,m\delta_1+\alpha,n\delta_2+\beta})$$

Convolutional layer

building feature maps

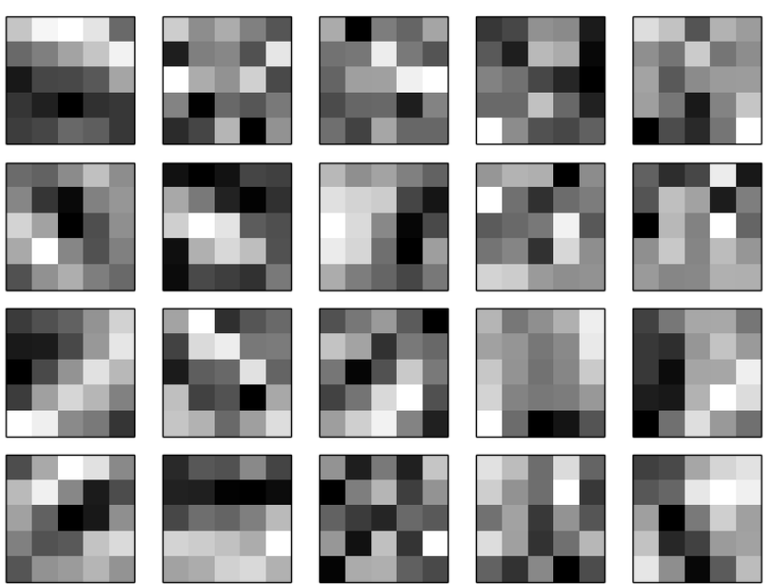
- Can be viewed as MLP with most of weights equal to zero and the rest of them are shared
- Number of independent weights:

5x5x*N* vs 28x28x24x24 , where
N - number of maps we want to build



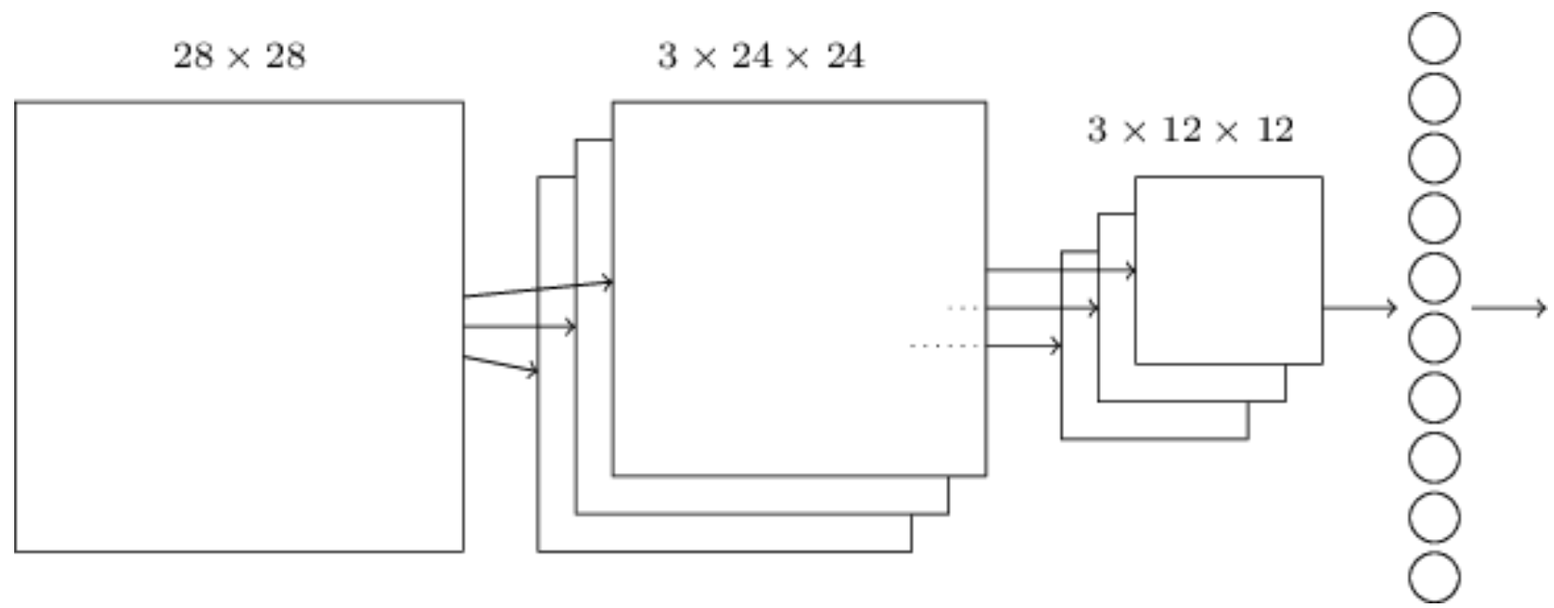
How filters look like (MNIST):

(weights are color-coded)

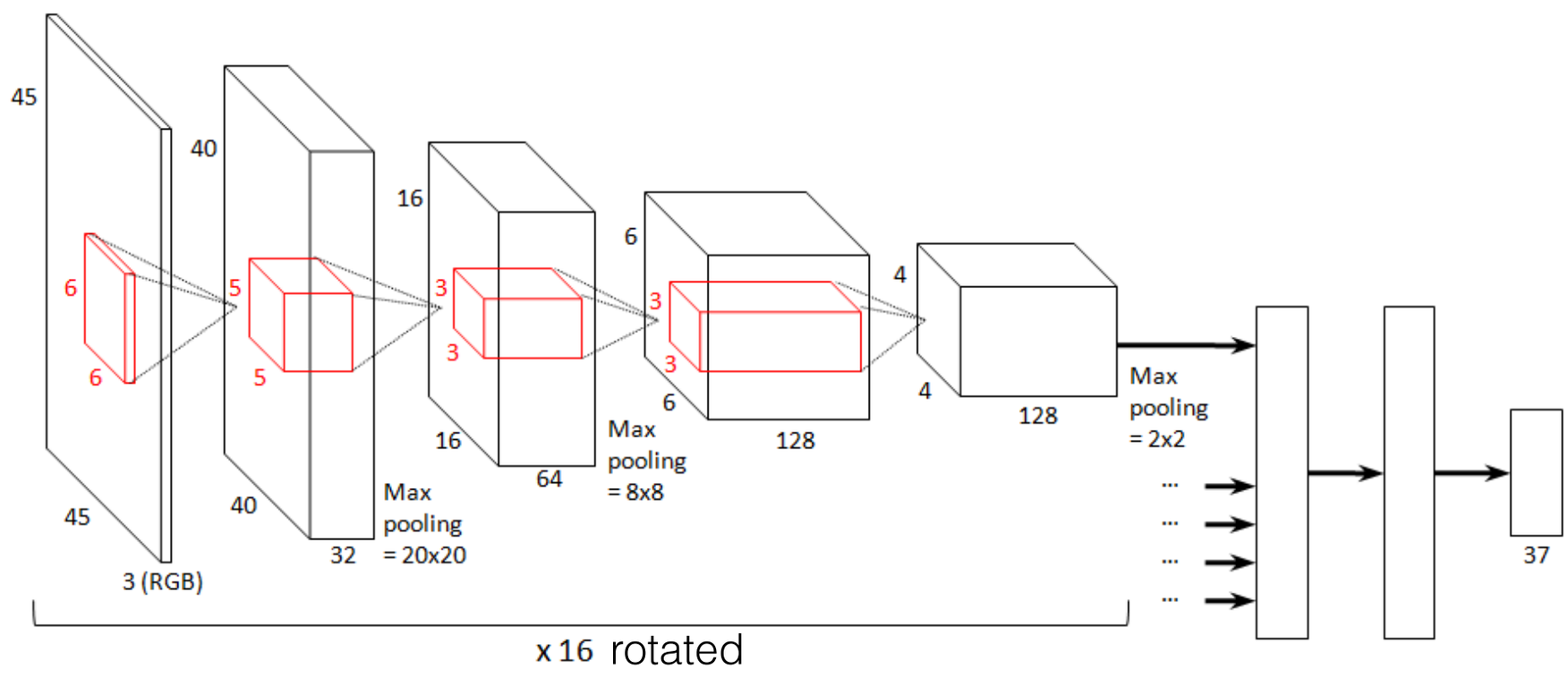


Convolutional NN architecture

minimal



Galaxy zoo <u>challenge</u> <u>winner</u>



Galaxy Zoo Challenge

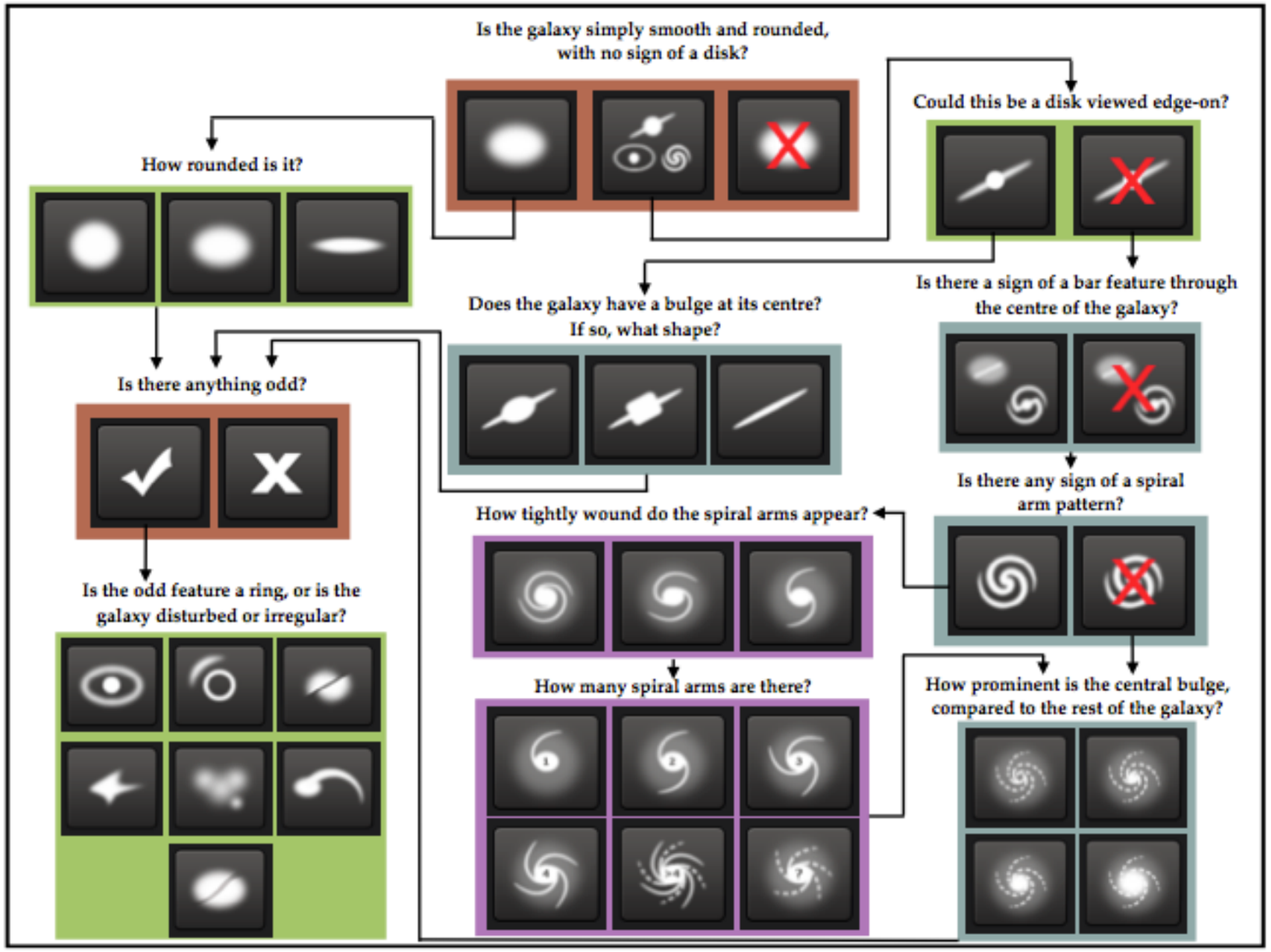
purpose: using image as input, we want to predict how Galaxy Zoo users (zooites) classify the image, i.e. predict vector of probabilities for all classes

training set: color images 424x424 along with vectors of probabilities for more than 60000 galaxies

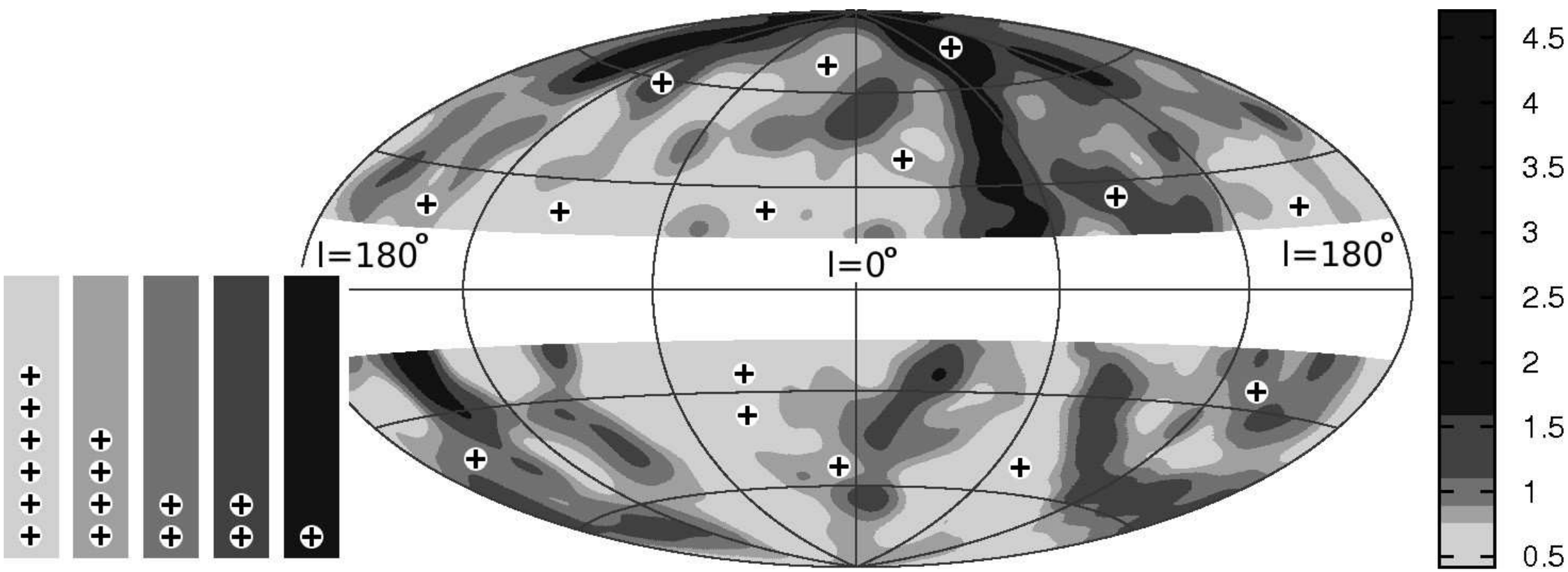
https://www.kaggle.com/c/galaxy-zoo-the-galaxy-challenge

Galaxy Zoo

Manual classification of galaxies using images from Sloan Digital Sky Survey



Test statistics using expected arrival direction density maps



Hylke B.J. Koers, Peter Tinyakov, JCAP 0904 (2009) 003