Reti neurali:

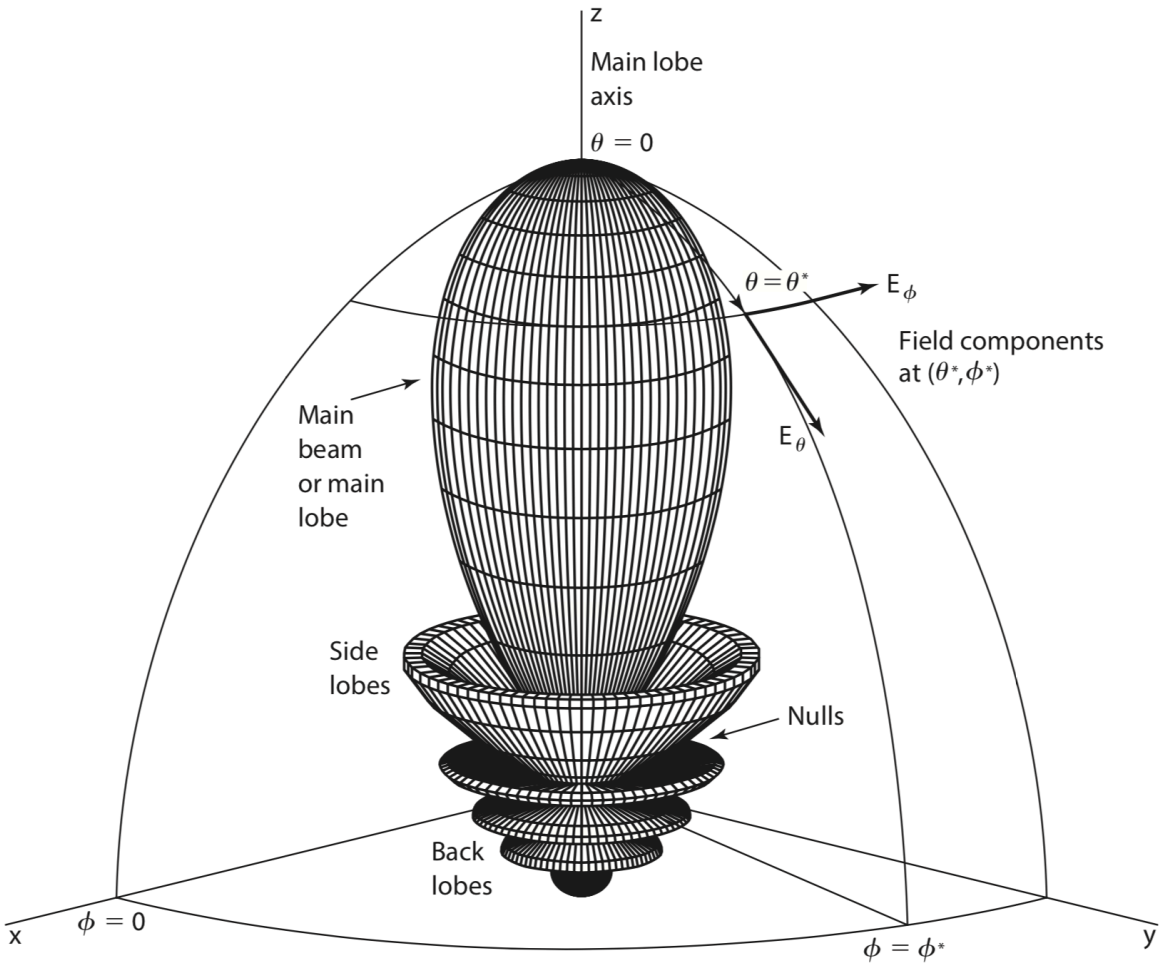
Stima delle proprietà di sistemi ottici nelle microonde

UNIVERSITÀ DEGLI STUDI DI MILANO

Eleonora Gatti

23 giugno 2020

Diagramma di radiazione



The quest for CMB B-modes – M. Tomasi, C. Franceschet, S. Realini

Parametri di interesse

- Fwhm (x)
- Fwhm (y)
- Ellitticità
- Co max
- Cx max

Idea

	id	x	y	z	fwhm_x	fwhm_y	e	tilt	co_max	cx_max
0	21	-0.316965	0.326756	50	0.398546	0.365183	1.091361	131.516741	53.84	13,1
1	42	-0.318192	0.272044	50	0.364137	0.398634	1.094738	36.860917	53.86	12,79
2	63	-0.319138	0.217441	50	0.364608	0.396464	1.087371	31.285228	53.88	12,11
3	83	-0.320197	0.163140	45	0.361428	0.384678	1.064326	22.716133	54.07	12,21
4	103	-0.321053	0.108822	40	0.358714	0.376443	1.049424	17.159739	54.20	11,6
...	...	...	...	...	...	...	...	...	...	...
164	3464	0.329983	-0.108774	45	0.351460	0.385294	1.096266	12.457245	54.17	12,91
165	3486	0.330462	-0.163029	50	0.351985	0.395278	1.122996	17.597785	54.04	13,01
166	3507	0.331717	-0.217468	50	0.351115	0.398046	1.133661	22.414060	54.02	12,87
167	3528	0.333326	-0.271998	50	0.350770	0.401112	1.143520	26.751476	53.99	13,38
168	3549	0.335293	-0.326679	50	0.350950	0.404582	1.152821	30.414636	53.95	13,95

Dataset

Grasp

Metodo di simulazione classico
Tempo di calcolo: 2 giorni

↓

Eccessivamente dispendioso in termini di tempo

Idea

Necessità: cercare una via alternativa

Una rete neurale è in grado di prevedere i parametri di interesse più efficacemente di una classica interpolazione?

Interpolazione vs Rete neurale

Interpolazione

Interp2d

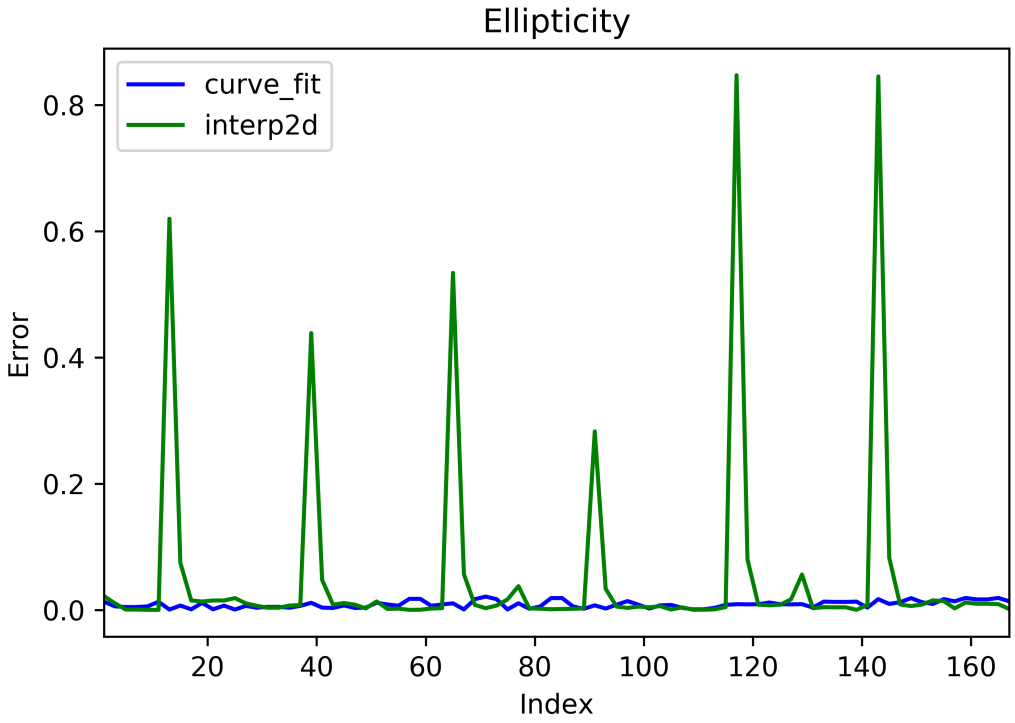
Modulo scipy.interpolate

Interpolazione lineare

Curve Fit

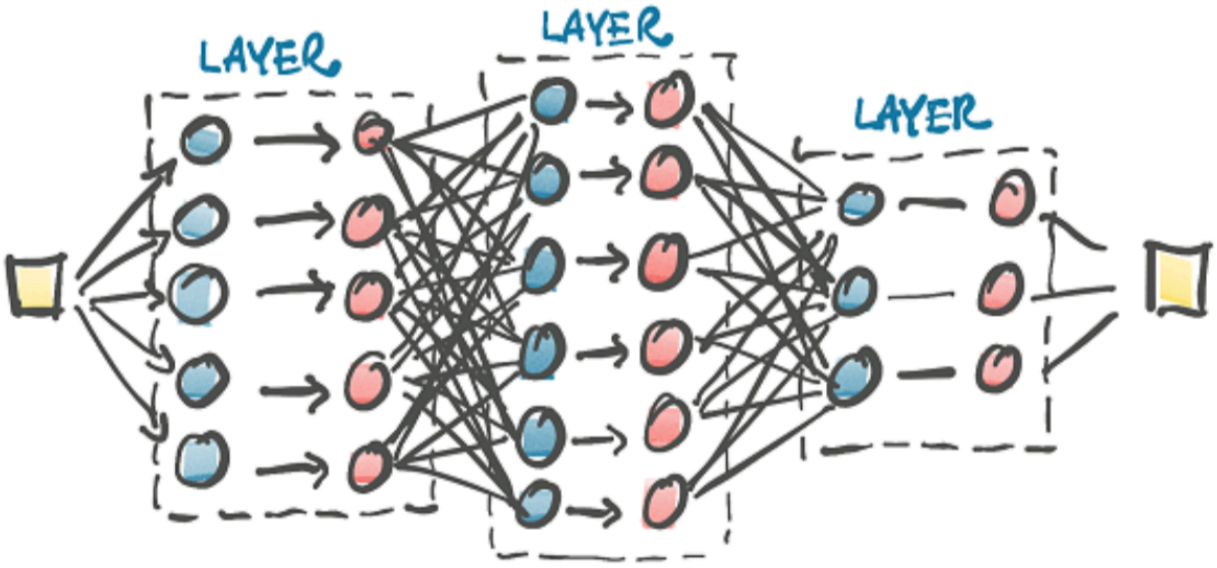
Modulo scipy.optimize

Fit su paraboloide



Rete Neurale: cos'è?

- Modello computazionale costituito dall'interconnessione tra neuroni
- Corrispondenza Input-Output

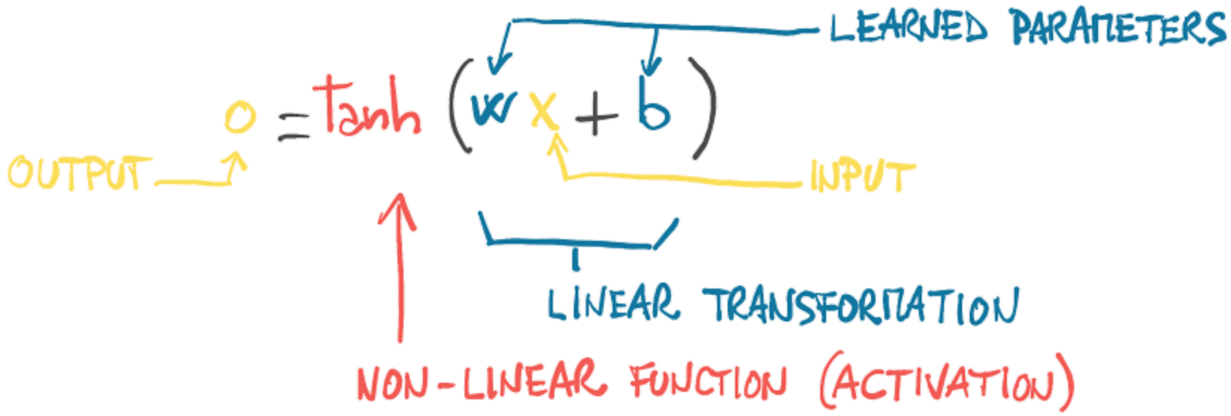


Deep Learning with PyTorch – E. Stevens

- Conoscenza acquisita tramite processo di apprendimento
- Conoscenza immagazzinata nei parametri della rete

Neurone

Funzione non lineare (**funzione di attivazione**) applicata ad una trasformazione lineare

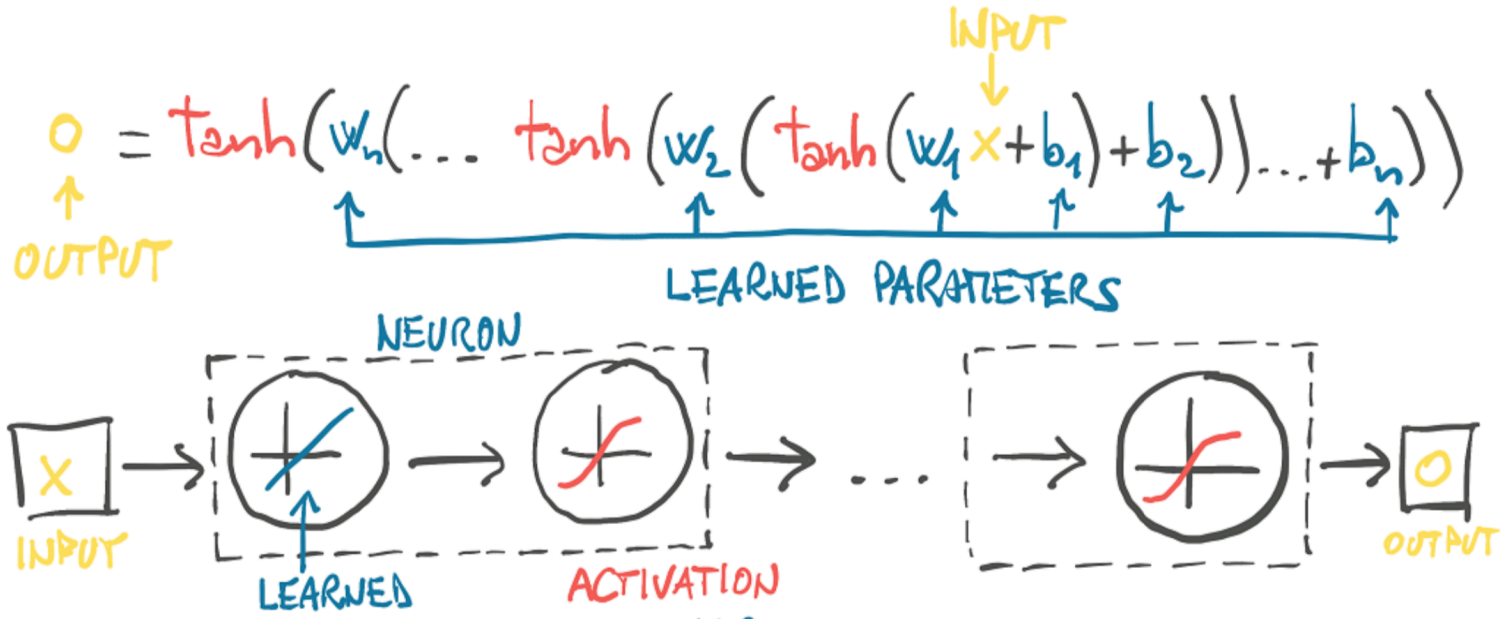


Deep Learning with PyTorch – E. Stevens

Uno o più neuroni ⟶ Layer

Input layer + hidden layer(s) + output layer ⟶ Rete

Rete Neurale



Deep Learning with PyTorch – E. Stevens

I parametri w_i (weights) e b_i (biases) sono i responsabili dell'apprendimento

My Net

Libreria utilizzata: PyTorch

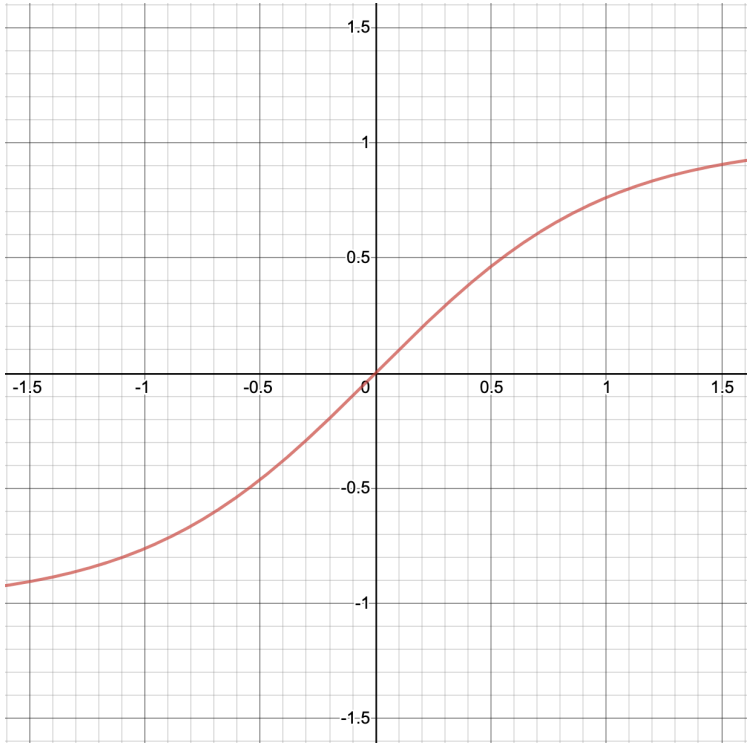
```
class Net(nn.Module):
    def __init__(self):
        super(Net, self).__init__()

        self.hidden1 = nn.Linear(2, 4)
        self.activation = nn.Tanh()
        self.hidden2 = nn.Linear(4, 4)
        self.output = nn.Linear(4, 1)
```

Hidden Layers
2

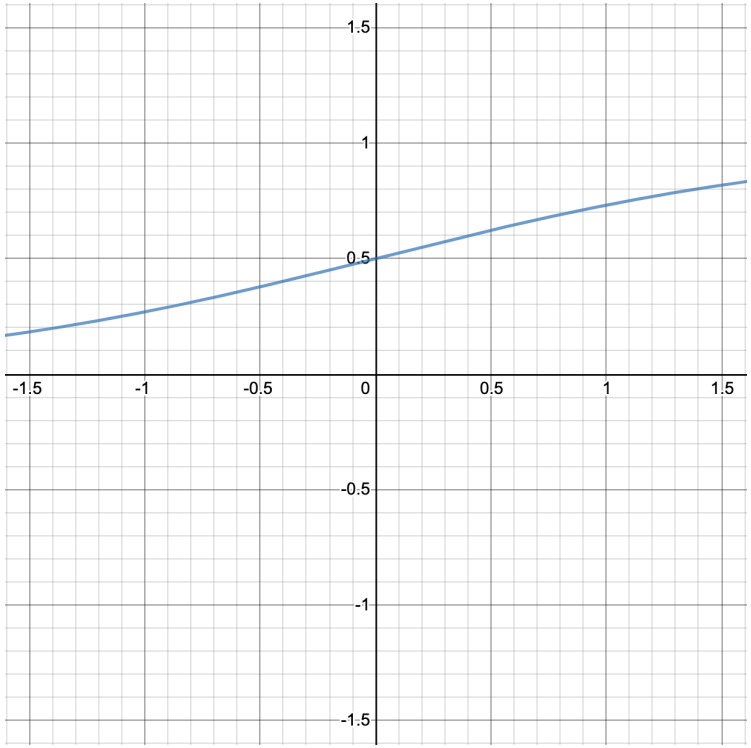
Neuroni/layer
4

Funzione di attivazione
Tanh

Funzioni d'attivazione



Tanh



Sigmoid

Motivo della scelta: codominio limitato

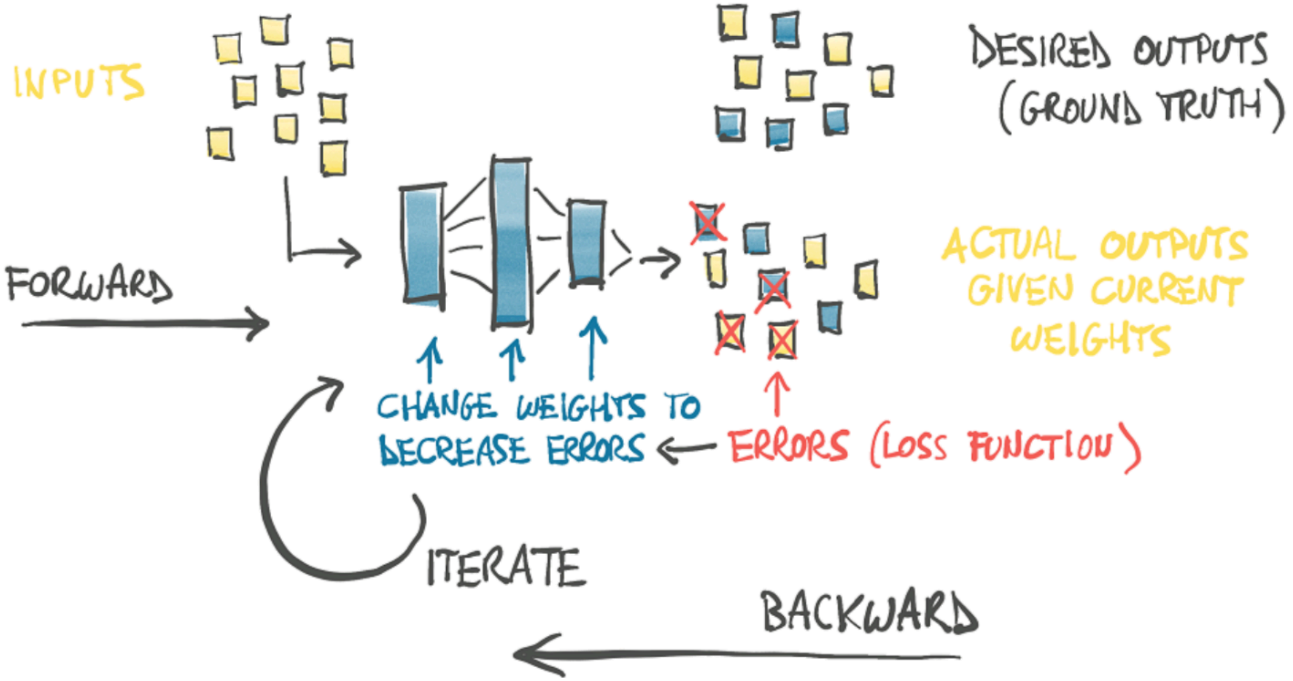
Pre training

- Normalizzazione: [-1,1] Tanh
  [0,1] Sigmoid
- Split dataset (scelta casuale): Training set (75%)
  Validation set (25%)

```
loader = torch.utils.data.DataLoader(dataset,
                                     batch_size = round(len(dataset)*0.75),
                                     shuffle=True)
train, validation = loader
```

Training

Apprendimento ⟶ Stima dei parametri di rete w_i e b_i



Deep Learning with PyTorch – E. Stevens

Training

Modello con parametri non noti

↓

Stima w_i e b_i

↓

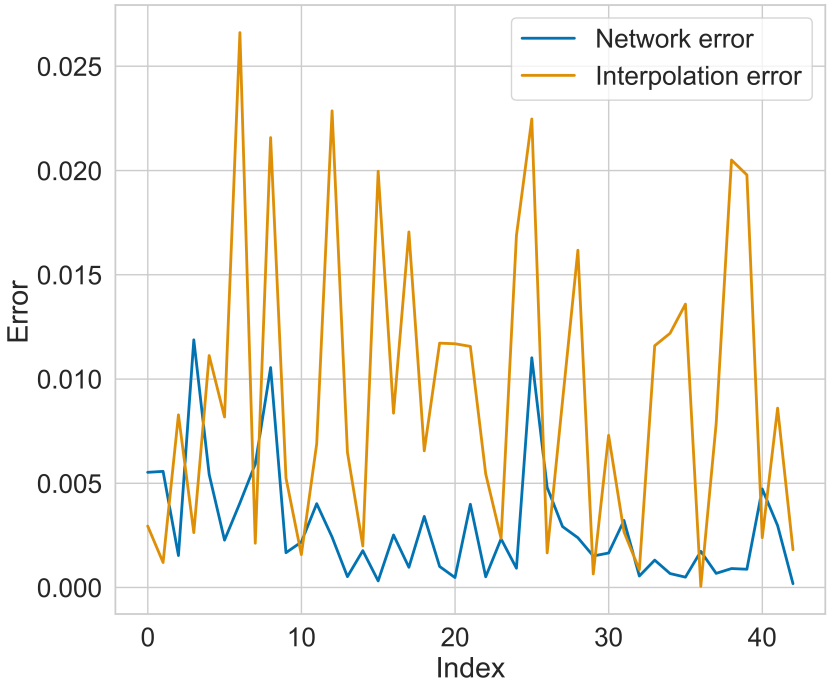
Minimizzare l'**errore** tra dati veri e "misurati"

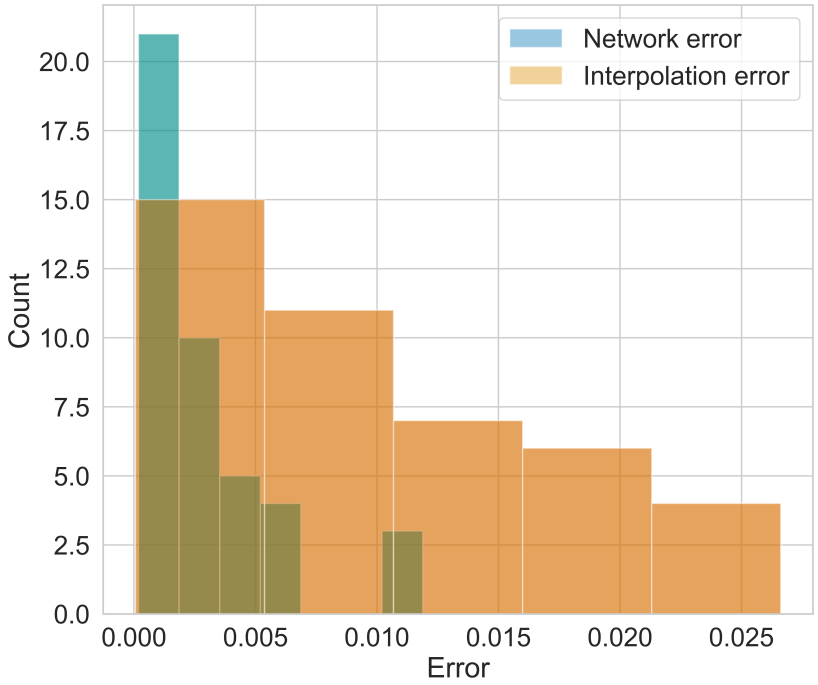
Misura dell'errore: **loss function**

```
loss_fun = nn.MSELoss()
```

Risultati

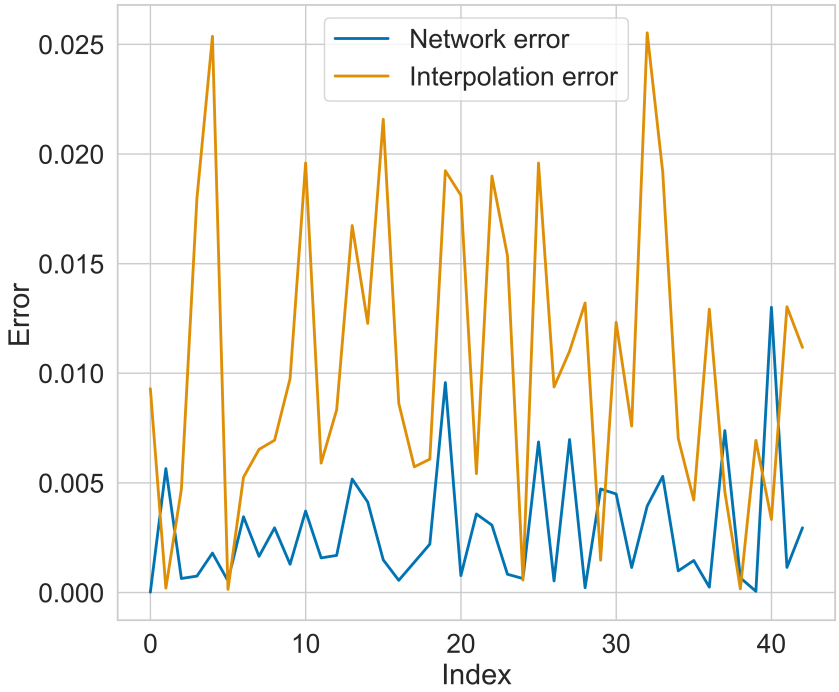
Activation Function: Sigmoid
Layers: 1
Parameter: Ellipticity

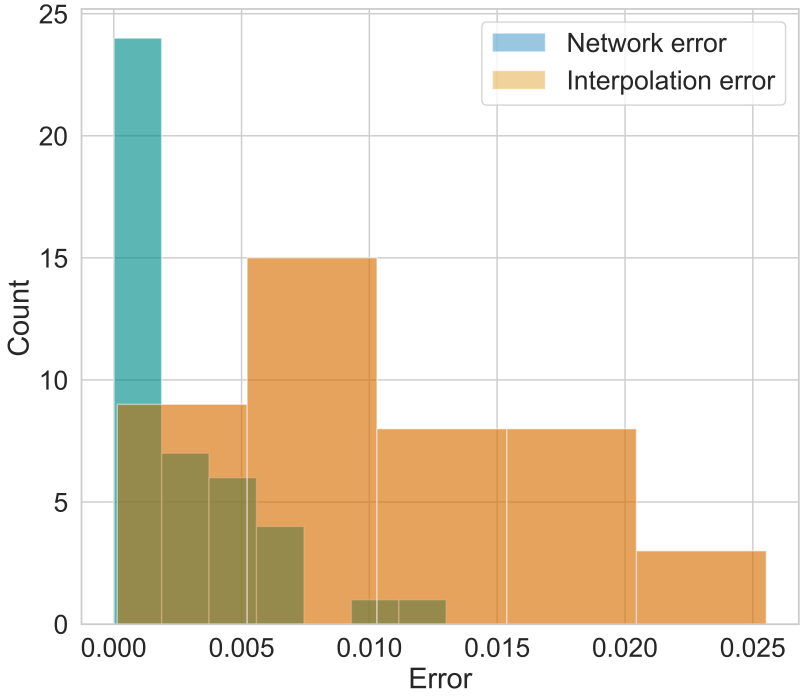




Risultati

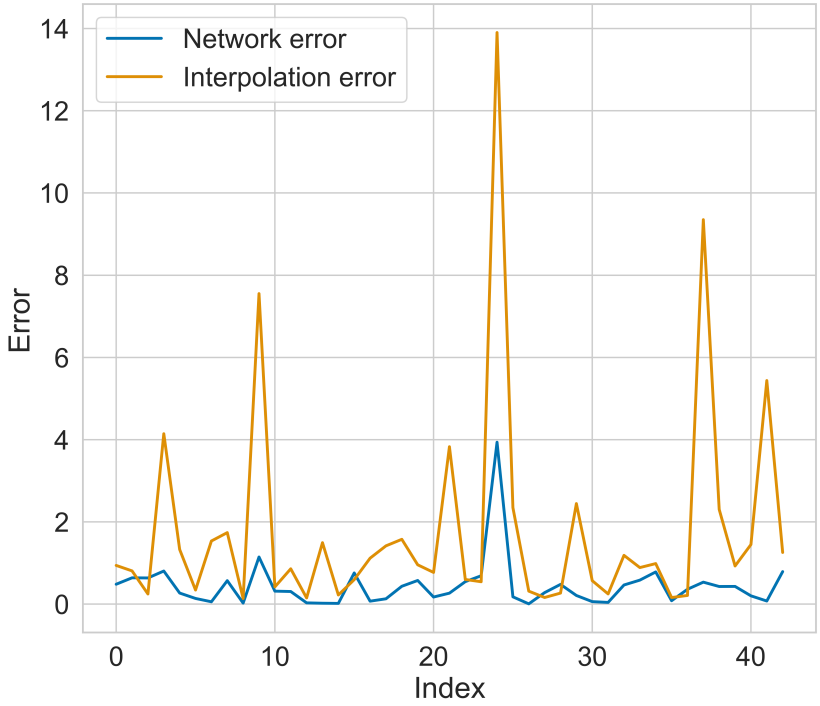
Activation Function: Tanh
Layers: 2
Parameter: Ellipticity

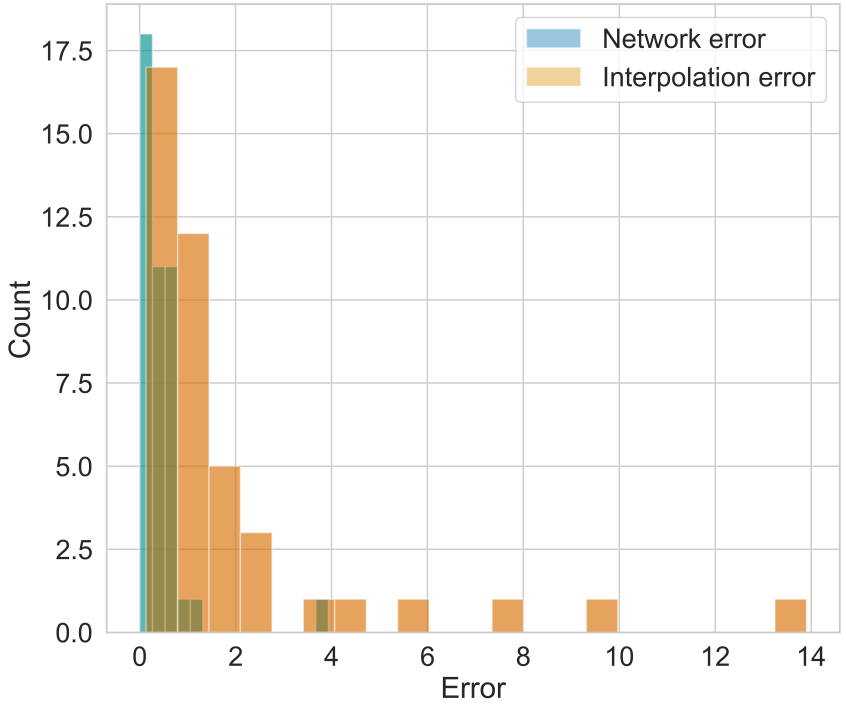




Risultati

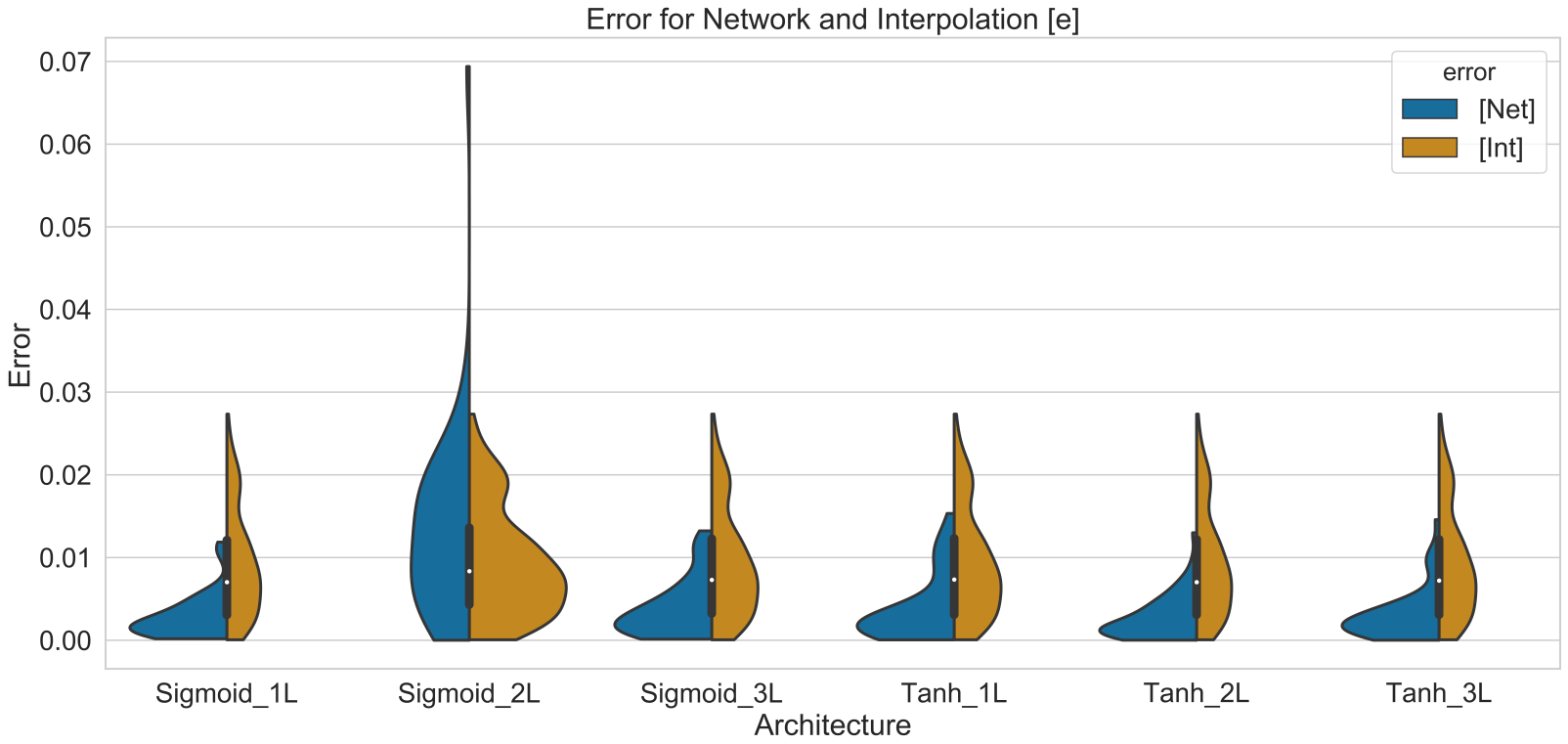
Activation Function: Sigmoid
Layers: 3
Parameter: Cx Max





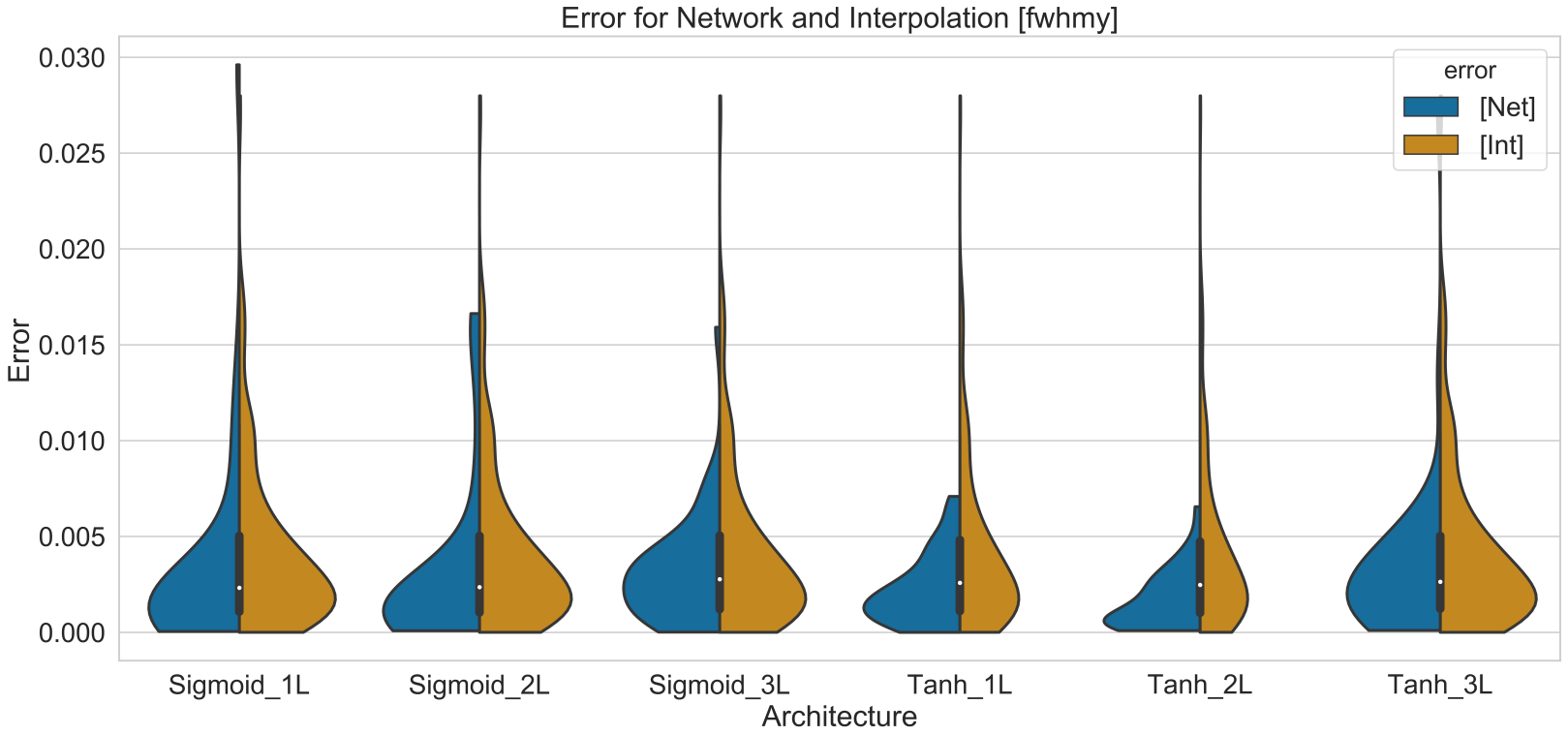
Risultati

Con la giusta architettura…



… vincono le reti neurali

Risultati


Error for Network and Interpolation [fwhmy]
error
[Net]
[Int]
Error
0.030
0.025
0.020
0.015
0.010
0.005
0.000
Sigmoid_1L
Sigmoid_2L
Sigmoid_3L
Tanh_1L
Tanh_2L
Tanh_3L
Architecture

Risultati

GRASP: fornisce il dato esatto

La giusta **rete neurale**: fornisce una stima del parametro affetta da un errore minore rispetto a quello che si otterrebbe tramite interpolazione

Vantaggio: *secondi* vs *giorni*


Grazie per l’attenzione