Advances in Morphological Neural Networks: Training, Pruning and Enforcing Shape Constraints

Nikolaos Dimitriadis and Petros Maragos

[1] École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
[2] School of ECE, National Technical University of Athens (NTUA), Athens, Greece
[1]nikolaos.dimitriadis@epfl.ch , [2]maragos@cs.ntua.gr

2021 IEEE International Conference on Acoustics, Speech and Signal Processing
6-11 June 2021 – Toronto, Ontario, Canada

Contributions

Binary Morphological Classifiers trained via Difference-of-Convex optimization	$\longrightarrow$	Extended to multiclass problems
Sparsity of Morphological Neural Nets	$\longrightarrow$	Showed quantitatively and qualitatively superior compression ability compared to ReLU FeedForward nets
Monotonic function approximation	$\longrightarrow$	Improved with softened morphological operators via Maslov Dequantization

Table of Contents

Morphological Operators for Vectors

$$\text{Dilation:} \qquad \delta_{\mathrm{w}}(\mathbf{x}) = w_0 \vee \left(\bigvee w_i + x_i \right)$$

$$\text{Erosion:} \qquad \varepsilon_{\mathrm{m}}(\mathbf{x}) = m_0 \wedge \left(\bigwedge m_i + x_i \right)$$

Softmax and Softmin scalar operations via Maslov Dequantization[1]

($h > 0$: temperature parameter)

$$\text{max}: \quad x \vee y \quad \longrightarrow \quad x \vee_h y = h \log(e^{x/h} + e^{y/h}) \quad : \text{ softmax}$$

$$\text{min}: \quad x \wedge y \quad \longrightarrow \quad x \wedge_h y = -h \log(e^{-x/h} + e^{-y/h}) \quad : \text{ softmin}$$

Morphological Operators for Vectors ↘

Softened Morphological operators

Softmax and Softmin scalar operations ↗

[1] Litvinov, G. L. "Maslov dequantization, idempotent and tropical mathematics: A brief introduction". In: *Journal of Mathematical Sciences* 140.3 (2007), pp. 426–444.

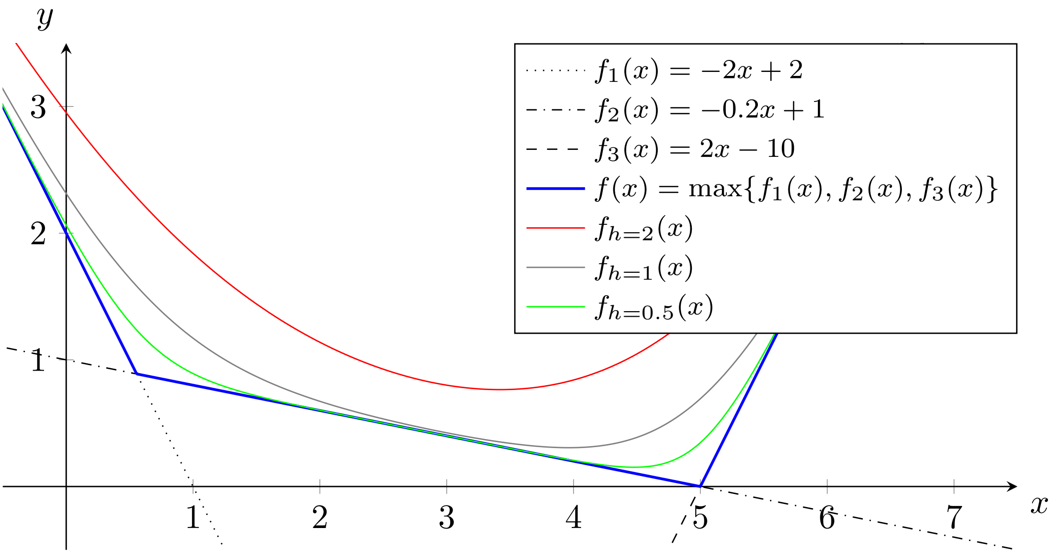
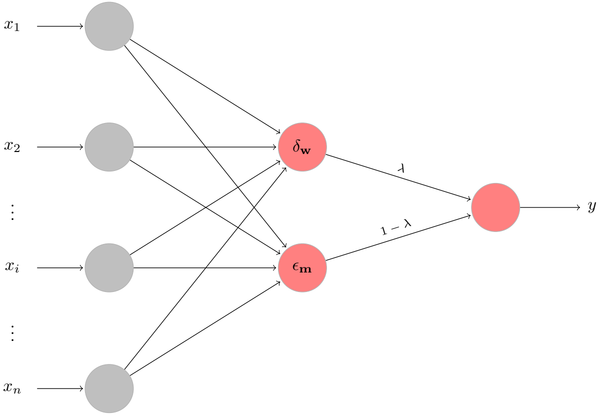


Figure: Effect of temperature parameter h

Table of Contents

Dilation-Erosion Perceptron (DEP)



Convex combination of one dilation and one erosion neuron:

$$y = f(\mathbf{x}) = \lambda \delta_{\mathrm{w}}(\mathbf{x}) + (1 - \lambda)\varepsilon_{\mathrm{m}}(\mathbf{x})$$

$$
\begin{aligned}
y = f(\mathbf{x}) = \lambda\delta_{\mathrm{w}}(\mathbf{x}) + (1-\lambda)\epsilon_{\mathrm{m}}(\mathbf{x}) &= \lambda\delta_{\mathrm{w}}(\mathbf{x}) - (1-\lambda)[-\epsilon_{\mathrm{m}}(\mathbf{x})] \\
&= \text{convex} - (-\text{concave}) \\
&= \text{convex} - (\text{convex})
\end{aligned}
$$

Training as Difference-of-Convex Program[1,2,3]

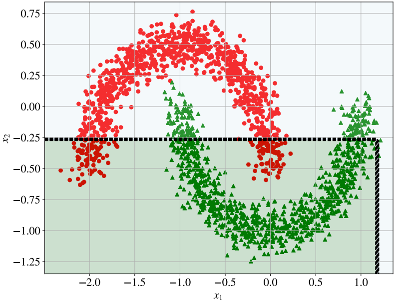
$$
\begin{aligned}
\text{minimize} \quad & \sum_{i=1}^{N} v_i \max\{0, \xi_i\} \\
\text{subject to} \quad & \lambda\delta_{\mathrm{w}}(\mathbf{x}_i) + (1-\lambda)\varepsilon_{\mathrm{m}}(\mathbf{x}_i) \geq -\xi_i \quad \forall \mathbf{x}_i \in \mathcal{P}, \\
& \lambda\delta_{\mathrm{w}}(\mathbf{x}_i) + (1-\lambda)\varepsilon_{\mathrm{m}}(\mathbf{x}_i) \leq +\xi_i \quad \forall \mathbf{x}_i \in \mathcal{N}
\end{aligned}
$$

[1] Charisopoulos, V. and Maragos, P. “Morphological Perceptrons: Geometry and Training Algorithms”. In: *Mathematical Morphology and Its Applications to Signal and Image Processing (Proc. ISMM 2017)*. Vol. 10225. LNCS. Springer, 2017, pp. 3–15. ISBN: 978-3-319-57240-6.

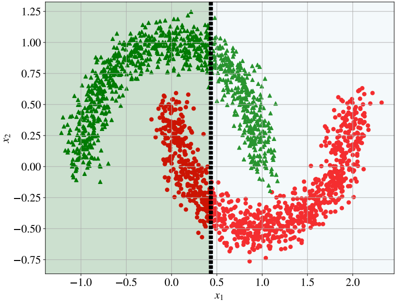
[2] Yuille, A. L. and Rangarajan, A. “The Concave-Convex Procedure”. In: *Neural computation* 15.4 (2003), pp. 915–936.

[3] Lipp, T. and Boyd, S. “Variations and extension of the convex–concave procedure”. In: *Optimization and Engineering* 17.2 (2016), pp. 263–287.

What is the effect of $\mathcal{N} \rightleftarrows \mathcal{P}$?



(a) correct labels



(b) reversed labels

Figure: `Double Moons` example[1].

Reduced Ordering:

Let R be a nonempty set, $\mathcal{L}$ be a complete lattice and $\rho : R \to \mathcal{L}$ be a surjective mapping. A reduced ordering is defined as: $\mathbf{x} \leq_\rho \mathbf{y} \Leftrightarrow \rho(\mathbf{x}) \leq \rho(\mathbf{y}), \forall \mathbf{x}, \mathbf{y} \in R$.

[1] Valle, M. E. "Reduced Dilation-Erosion Perceptron for Binary Classification". In: *Mathematics* 8.4 (2020). ISSN: 2227-7390.

Extending to multiclass problems

one-versus-the-rest

- positive class $\mathcal{C}_k$, negative class $\mathcal{C}_{-k}$
- imbalance: $|\mathcal{C}_k| \simeq \frac{N}{K} \ll |\mathcal{C}_{-k}| \simeq \frac{(K-1)N}{K}$

one-versus-one

- $\frac{K(K-1)}{2}$ distinct classifiers must be trained
- majority (hard) vote of all classifiers

Method

Use a bagging classifier for n Radial Basis Function (RBF) kernel estimators.

Results

	MNIST	FashionMNIST
$n = 5$	$\mathbf{97.72 \pm 0.01}$	$\mathbf{88.21 \pm 0.01}$
$n = 10$	$\mathbf{97.72 \pm 0.01}$	88.07 ± 0.01
$n = 15$	97.67 ± 0.01	88.11 ± 0.01
$n = 20$	97.64 ± 0.01	88.12 ± 0.01

Table: Results of Bagging *multiclass* r-DEP with n RBF kernels.

Performance similar achitectures trained via Gradient Descent

CCP training is very robust

Table of Contents

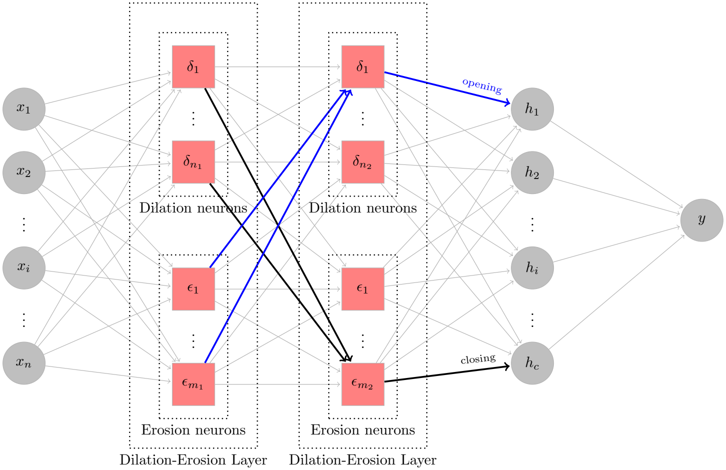


Figure: Dense Morphological Network with 2 hidden layers[1]

Focus is on sparsity. Apply ℓ_1 pruning.

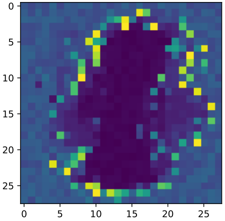
[1] Mondal, R., Santra, S., and Chanda, B. "Dense Morphological Network: An Universal Function Approximator". In: *arXiv* (2019). URL: http://arxiv.org/abs/1901.00109.

	p	Adaptive Momentum Estimation δ	ε	(δ, ε)	FF-ReLU	Stochastic Gradient Descent δ	ε	(δ, ε)	FF-ReLU
MNIST	100%	97.62	96.17	97.95	98.13	94.86	93.36	96.07	98.16
	75%	97.62	96.18	97.93	98.15	94.86	93.36	96.07	98.12
	50%	97.62	96.22	97.90	98.17	94.86	93.37	96.07	98.08
	25%	97.62	96.09	97.87	97.51	94.86	93.40	96.06	98.01
	10%	97.62	95.78	97.74	93.38	94.86	93.38	96.09	96.67
	7.5%	97.62	95.42	97.76	90.17	94.86	93.38	96.10	95.56
	5%	97.62	94.51	97.66	83.39	94.86	93.40	96.10	92.96
	2.5%	97.62	93.43	97.37	68.93	94.86	93.39	96.09	80.48
	1%	97.62	91.17	97.08	44.22	94.86	93.38	96.08	58.07
FashionMNIST	100%	86.31	86.82	88.32	88.82	82.06	85.23	86.21	87.79
	75%	86.30	86.81	88.30	88.88	82.00	85.23	86.21	87.75
	50%	86.22	86.80	88.33	88.18	82.05	85.25	86.20	87.19
	25%	85.95	86.85	88.31	82.15	81.90	85.26	86.28	84.35
	10%	85.58	86.27	88.05	65.89	81.67	85.27	86.23	73.22
	7.5%	85.47	86.15	87.99	57.93	81.63	85.27	86.21	63.95
	5%	85.37	85.81	87.76	49.12	81.52	85.24	86.22	47.73
	2.5%	84.91	85.47	87.56	42.48	81.14	85.26	86.22	38.84
	1%	81.14	84.86	86.85	28.13	80.68	85.27	86.18	35.46

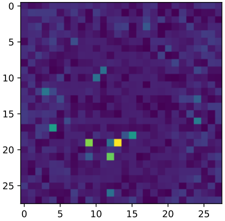
Table: Accuracy of pruned networks on the MNIST and FashionMNIST datasets.

Models: $\delta \to$ only dilation neurons, $\varepsilon \to$ only erosion, $(\delta, \varepsilon) \to$ split equally, FF-ReLU $\to$ FeedForward NN with ReLU.

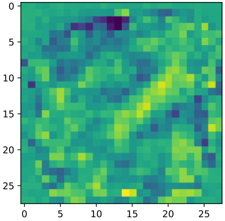
shades of red showcase the degree of (severe) deterioration in accuracy green indicates the absence of performance loss



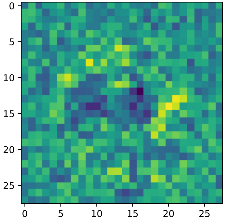
(a) (δ, ε) - Adam



(b) (δ, ε) - SGD



(c) FF-ReLU - Adam



(d) FF-ReLU - SGD

Figure: Examples of hidden layer activations for various models (MNIST dataset).

Table of Contents

Monotonic Network architecture

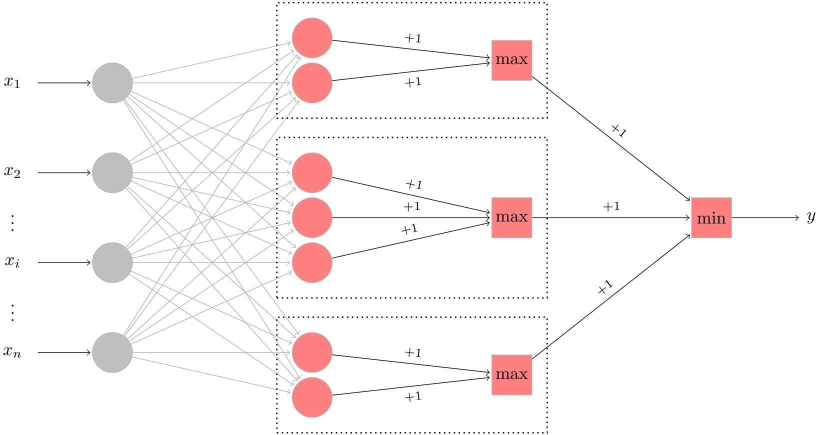


Figure: Monotonic network by Sill[1]. The gray edges correspond to nonnegative weights.

[1] Sill, J. "Monotonic Networks". In: *Adv. in NeurIPS*. 1998.

Characteristics

$$y = f(\mathbf{x}) = \bigwedge_{k \in [K]} \bigvee_{j \in [J]} \{\mathbf{w}_{k,j}^\top \mathbf{x} + b_{k,j}\}$$

- neither convex nor concave
- Monotonicity constraints $\longrightarrow \mathbf{w} \in \mathbb{R}^n_{\geq 0}$
- corresponds to morphological closing
- invertible for positive weights $\longrightarrow$ morphological opening

$$x = f^{-1}(y) = \bigvee_{k \in [K]} \bigwedge_{j \in [J]} \{w_{k,j}^{-1}(y - b_{k,j})\}$$

Details

- Used softened morphological operators
- Active group: affine term that determines the output for pattern $\mathbf{x} \in \mathbb{R}^n$
- "Hard" operators $\rightarrow$ $1-1$ correspondence between active group and output
  $\rightarrow$ only active hyperplane gets updated
  $\rightarrow$ a small fraction of hyperplanes dominate the training
- "Soft" operators alleviate undifferentiability $\rightarrow$ better approximation

Experiment Description

- strictly increasing function $f(x) = x^3 + x + \sin x, x \in [-4, 4]$
- scale both the domain and the image of f to $[-1, 1]$
- 100 observations uniformly and corrupt them with additive i.i.d zero-mean Gaussian noise $\varepsilon \sim \mathcal{N}(0, \sigma^2)$
- Glorot uniform initialization[1] for all network parameters

[1] Glorot, X. and Bengio, Y. "Understanding the difficulty of training deep feedforward neural networks". In: *Proc. 13th Int'l Conf. Artificial Intelligence & Statistics.* 2010.

Results

σ	0.05	0.1	0.15	0.2
Linear Reg.	0.0236	0.03077	0.04827	0.0505
Isotonic Reg.[1]	0.0042	0.01112	0.02557	0.0417
Sill Net[2]	0.00305	0.01107	0.02401	0.0390
Smooth Sill Net [ours]	**0.00294**	**0.00938**	**0.02302**	**0.0386**

Table: RMS error of monotonic regression methods with noise $\varepsilon \sim \mathcal{N}(0, \sigma^2)$

[1] Barlow, R. E. and Brunk, H. D. "The Isotonic Regression Problem and its Dual". In: *J. Amer. Stat. Assoc.* 67.337 (1972), pp. 140–147.
[2] Sill, J. "Monotonic Networks". In: *Adv. in NeurIPS*. 1998.

Results

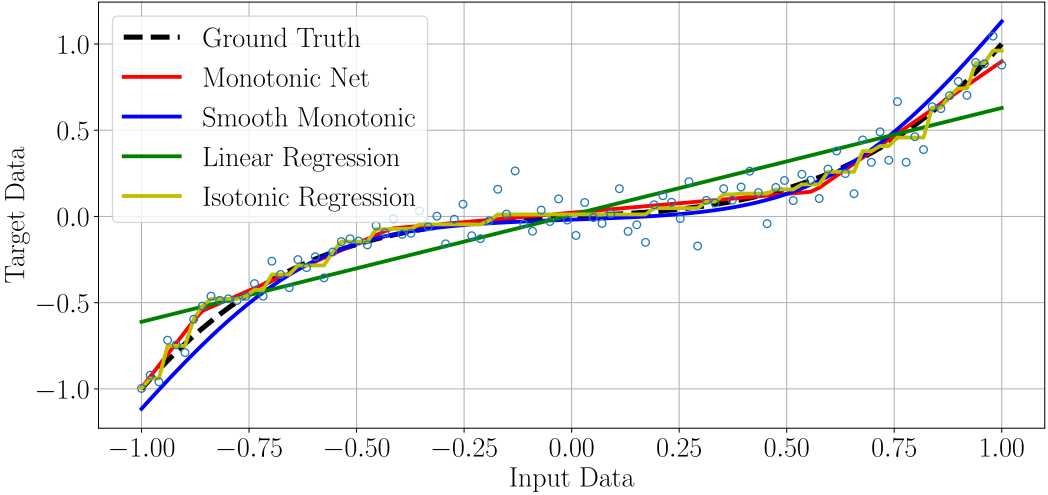


Figure: Comparison of monotonic regression methods
Smooth Monotonic is ours.

Conclusion

- Extended Binary Morphological Classifiers trained via CCP to multiclass problems
- Studied the sparsity of Morphological Neural Nets and showed their superior compression ability compared to their linear counterparts
- Improved convergence and accuracy of monotonic regression with softened morphological operators based on Maslov Dequantization

For a complete list of references please see the paper.

Thank you for your attention!