HYPERPARAMETER TUNING IN THE LEARNING OF MULTITHRESHOLD NEURONS

VLADYSLAV KOTSOVSKY, LISOVSKA TETIANA

ICST
Information Control Systems and Technologies

RELEVANCE OF THE PROBLEM

- Multithreshold-based neural systems are more powerful compared to the single-threshold ones and is successfully employed in the machine learning, e.g. in pattern classification.
- The design of fast learning algorithms as well as the proper choice of their hyperparameters allow to use the capacity of multithreshold paradigm more effectively.

GOALS OF THE RESEARCH

- The development of the learning algorithm for a multi-valued multithreshold neuron that should effectively use the advantages of multiple thresholds.
- The study of its finiteness for intended applications in classification.
- Investigation what values of algorithm hyperparameters would be used in order to speed-up the training process and improve the capacity of resulting neuron.

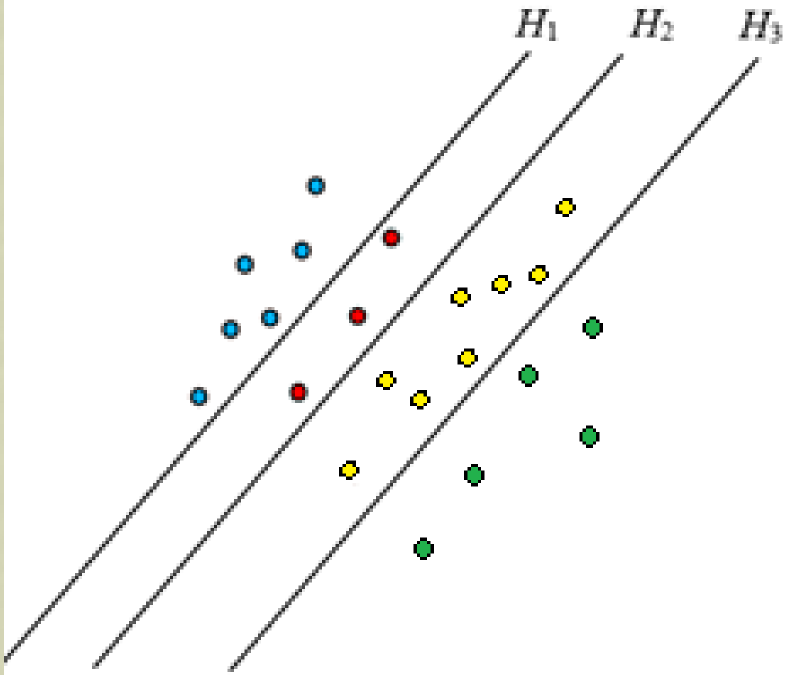
MODEL OF MULTI-VALUED K-THRESHOLD NEURON

The k-threshold multi-valued neuron with the *weight vector* $\mathbf{w} = (w_1, \ldots, w_n) \in \mathbf{R}^n$ and (ordered) *threshold vector* $\mathbf{t} = (t_1, \ldots, t_k) \in \mathbf{R}^k$ is the computation unit with n inputs $x_1, \ldots, x_n$ whose single output $y \in \mathbb{Z}_k$ is calculated by the following rule:

$$y = f_{\mathbf{t}}(\mathbf{w} \cdot \mathbf{x}), \qquad (1)$$

where

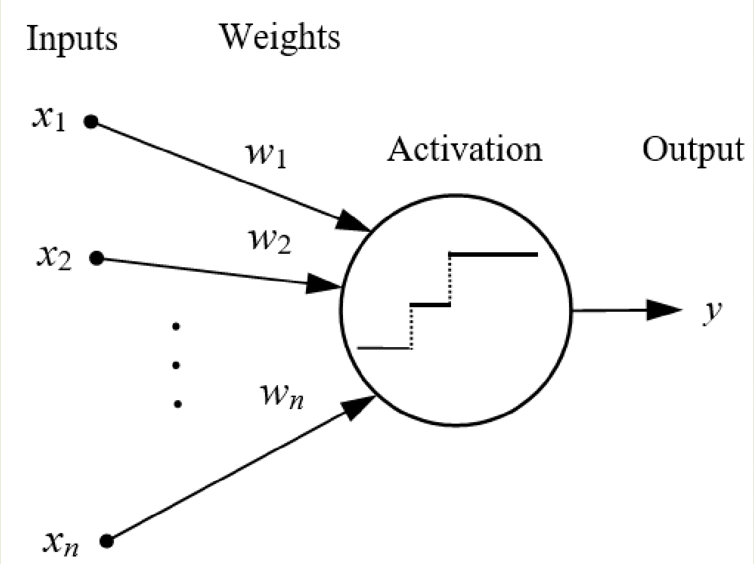
$$f_{\mathbf{t}}(x) = \begin{cases} 0, & \text{if } x < t_1, \\ 1, & \text{if } t_1 \le x < t_2, \\ \dots\dots\dots\dots\dots\dots & \\ k-1, & \text{if } t_{k-1} \le x < t_k, \\ k, & \text{if } t_k \le x. \end{cases}$$



ONLINE LEARNING ALGORITHM

ShiftedMultithreshold $(A_0, A_1, \ldots, A_k, r, \sigma, \mathbf{v}^0, \eta, d)$

$B \leftarrow \text{NormalizedSet}(A_0, A_1, \ldots, A_k)$
$\mathbf{v} \leftarrow \mathbf{v}^0$
$(i, j, err) \leftarrow (0, 0, 1)$
`while` $i < r$ `and` $err > 0$:
    $err \leftarrow 0$
    shuffle B
    `for` $\mathbf{b}$ `in` B:
        $s \leftarrow \mathbf{b} \cdot \mathbf{v}$
        `if` $s > 0$:
            `continue`
        $j \leftarrow j + 1$
        $err \leftarrow err + 1$
        $\mathbf{v} \leftarrow \mathbf{v} + \eta(j)(d - \sigma s)\mathbf{b}$
    $i \leftarrow i + 1$
$\mathbf{w} \leftarrow (v_1, \ldots, v_n)$
$\mathbf{t} \leftarrow (-v_{n+1}, \ldots, -v_{n+k})$
`return` $\mathbf{w}, \mathbf{t}$



CONVERGENCE OF LEARNING

Proposition. If finite sets $A_0, A_1, \ldots, A_k$ are strongly *k*-separable,

$$\eta(j) = \eta_1(j) + \frac{\eta_2(j)}{d + \left|\mathbf{b}^j \cdot \mathbf{v}^{j-1}\right|},$$

where $\mathbf{b}^j$ is a train vector used in *j*th correction, $\mathbf{v}^{j-1}$ is the value of sought vector $\mathbf{v}$ after previous correction and

$$0 \le \eta_1(j) \le 2,\ 0 \le \eta_2(j) \le \eta_{\max},\ \eta_1(j) + \eta_2(j) \ge \eta_{\min},$$

where $\eta_{\min}$ and $\eta_{\max}$ are arbitrary positive constants, then there exists r such that after at most r corrections ShiftedMultithreshold yields a multi-valued *k*-threshold neuron $(\mathbf{w}, \mathbf{t})$, which produces the partition $(A_0, A_1, \ldots, A_k)$.

HYPERPARAMETER TUNING

Performance comparison for different σ

Dataset size	Average number of corrections	
	$\sigma = 0$	$\sigma = 1$
258	1413.51	157.14
512	3701.84	218.03
1024	8989.07	274.27
2048	15791.62	338.86
4096	20159.93	415.48

Performance comparison for initial approximations

Dataset size	Average number of corrections	
	Random	Optimized
258	157.14	46.52
512	218.03	90.99
1024	274.27	133.37
2048	338.86	167.3
4096	415.48	216.71

Performance results for constant learning rates

Dataset size	Best learning rate	Average number of corrections
258	1.949	44.01
512	1.986	88.24
1024	2.003	128.81
2048	1.928	162.08
4096	2.051	201.33

RESULTS ON REAL-WORLD DATASETS

Classifier	Accuracy on training set (in %)		Accuracy on test set (in %)	
	balance-scale	dry-bean	balance-scale	dry-bean
Perceptron	84.25	20.91	82.23	16.59
5-Nearest Neighbor	88.07	81.07	81.85	72.34
Random Forest	100	99.98	82.93	90.03
MLP Classifier	95.28	53.51	92.74	49.60
Multithreshold	88.84	57.23	83.89	51.22

- Multithreshold algorithm performed well on 3-class classification task on balance-scale dataset and the online modification had the second-best accuracy on the test set.
- Classification on the dry-bean datasets was more difficult. Learning for both linear perceptron and MLP failed completely. Multi-valued MTN yielded had the best accuracy among all considered neural-like models. But its accuracy was considerably worse than in the case of the use of random forest classifier.

CONCLUSIONS

- The proposed online learning algorithm is able to learn a multi-valued *k*-threshold neural unit, whose performance is concurrent compared to popular classifiers.
- The learning mode defined by σ is extremely significant to the performance of online learning and its proper value 1 decreases the number of corrections in 10 times.
- The initial approximation also matters. The use of the improved approximation requires additional calculations but this can reduce the number of corrections in 2–4 times compared to random initial approximation.
- Constant learning rate in the case $1.93 < \eta < 2.05$ is good choice for the relaxation learning.
- The generalization ability of k-threshold neuron decreases with the growth of k.
- The shift hyperparameter d has mainly the theoretical importance as a guarantee of the finite learning. Its practical application is limited by small values, whereas larger d can significantly decrease the learning process.

THANK YOU FOR YOUR ATTENTION