UDC 004.8
Original Scientific Paper

Boosting Ensembles of Heavy Two-Layer Perceptrons for Increasing Classification Accuracy in Recognizing Shifted-Turned-Scaled Flat Images with Binary Features

Vadim Romanuke *romanukevadimv@mail.ru*
Applied Mathematics and Social Informatics Department
Khmelnitskiy National University, Khmelnitskiy, Ukraine

Abstract

A method of constructing boosting ensembles of heavy two-layer perceptrons is stated. The benchmark classification problem is recognition of shifted-turned-scaled flat images of a medium format with binary features. The boosting gain is suggested in two aspects. The earliest one is the ratio of minimal recognition error percentage among the ensemble perceptrons to the recognition error percentage performed by the ensemble. The second gain type is the ratio of minimal variance of perceptrons' recognition error percentages over 26 classes to variance of the ensemble's recognition error percentages over 26 classes. Both ratios increase as the number of perceptron classifiers in the ensemble increase. The ensemble of 36 classifiers performs with increased accuracy, where recognition error percentage is decreased for 33 %, and the variance is decreased for more than 50 %. Further increment of classifiers into ensemble cannot increase accuracy much as there is the saturation effect of the boosting gain. And the gain itself depends on the range of noise modeling object's distortions. Thus, the heavier perceptron classifier the less gain is expected.
Keywords: boosting ensemble, two-layer perceptron, recognition, accuracy, boosting gain

1. Introduction

Classification is needed for automatization ensuring reliability of complex systems and controlling them. At particular classification, object recognition is an element of watching and registering objects of interest. The ideal situation is when recognition error rate is zero. Despite this is utopia, recognition error rate can be decreased via increasing accuracy of the classifier performing the recognition. For recognizing objects with a lot of independent features (whose number is of the order of hundreds and thousands), neural networks fit best [1, 16]. And for classifiers based on neural networks there are several known approaches to increasing classification accuracy.

2. Approaches to increasing classification accuracy

One of the most popular classifiers is projected with decision trees [4, 11]. Decision trees are boosted to random forests, what increases their classification accuracy [9, 12]. However, designing a decision tree for recognizing even medium format images is too ineffective. For fulfilling this task, there are neural networks of various types: perceptron, cognitron, neocognitron, convolutional and hierarchical neuronets. These networks' classification accuracy is increased either via architecture adjustment [1, 3, 2] or training process optimization [10, 7]. Boosting over neuronets is applied as well, but ensembles are aggregated from weak learners [8, 5], rather than heavy multilayer perceptrons or more complicated networks. Besides, most existing boosting methods were designed primarily for binary classification [8, 15, 17]. And in many cases, the extension to classification problem of $N \in \mathbb{N} \setminus \{1, 2\}$ classes isn't straightforward [8, 5].

3. Goal and tasks of the article

With statistically universal approximators based on two-layer perceptron with nonlinear transfer functions (TLPNLTF), being the lightest among the heavy neuronets, boosting TLPNLTF ensembles are to be tried. For trying their performance, the problem of recognizing shifted-turned-scaled (STS) flat images with binary features (monochrome images) is going to be considered. The goal is to register classification accuracy increment of TLPNLTF ensembles, and to score the gain as a ratio of the ensemble classification accuracy to classification accuracy of the smartest TLPNLTF in the ensemble.

Before boosting, the TLPNLTF classifier along with its training routine for recognizing STS images is formalized. Then the boosting training routine (BTR) is stated. After this, we take a few tens of TLPNLTF trained for recognizing STS images of a medium format. In this way, we are going to plot the dependence of recognition error percentage (REP) against the number of TLPNLTF classifiers within the ensemble. Thus the ratio showing the TLPNLTF boosting gain will be seen clearly.

In discussion, possibilities of propagation of the obtained result should be expounded. In conclusion, the scientific meaning and practical significance will be declared. And an outlook for further work on heavy boosting ensembles will be given.

4. Increasing classification accuracy with heavy boosting ensemble of TLPNLTF

For constructing a TLPNLTF classifier, its transfer functions can be set log-sigmoid. And for the problem of recognizing $N \in \mathbb{N} \setminus \{1\}$ classes, the single object input of TLPNLTF is $\mathbf{X} = [x_i]_{1\times Q} \in X \subset \mathbb{R}^Q$ by $Q \in \mathbb{N}$ features and the output of TLPNLTF is the number [14]

$$s_* \in \arg\max_{s=\overline{1,N}} v_s =$$

$$= \arg\max_{s=\overline{1,N}} \left\{ \left(1+\exp\left[-\left(\sum_{k=1}^{Q_{\text{HL}}} u_{ks} \cdot \left(1+\exp\left[-\left(\sum_{i=1}^{Q} x_i a_{ik} + h_k \right) \right] \right)^{-1} + b_s \right) \right] \right)^{-1} \right\} \tag{1}$$

of the concurrent class, where Q_{HL} is number of neurons in the hidden layer of TLPNLTF, and $Q_{\text{HL}} \cdot (Q+N+1)+N$ coefficients

$$\left\{ [a_{ik}]_{Q\times Q_{\text{HL}}}, [u_{ks}]_{Q_{\text{HL}}\times N}, [h_k]_{1\times Q_{\text{HL}}}, [b_s]_{1\times N} \right\} \tag{2}$$

are to be determined during the training routine. While being trained [6, 13], the input of TLPNLTF is fed with the training set

$$\left\{ \mathbf{Y} = [y_{is}]_{Q\times N} : y_{is} = x_i^{\langle s \rangle} \right\} \tag{3}$$

of pure representatives, where $\mathbf{X}_s = \left[x_i^{\langle s \rangle} \right]_{1\times Q} \in X \subset \mathbb{R}^Q$ is the s-th class pure representative. Then comes the second stage, when the input of TLPNLTF is fed with the training set

$$\left\{ \left\langle \{\mathbf{Y}\}_{r=1}^{R}, \{\tilde{\mathbf{Y}}_h\}_{h=1}^{H} \right\rangle : R \in \mathbb{N} \cup \{0\}, \tilde{\mathbf{Y}}_h = \Psi(\mathbf{Y}, \sigma_h) + \rho \cdot \sigma_h \cdot \mathbf{\Xi}, \rho \geqslant 0, \sigma_h = h\sigma_0 H^{-1} \ \forall h = \overline{1,H}, \right.$$

$$\left. H \in \mathbb{N}, \sigma_0 > 0, \mathbf{\Xi} = [\xi_{is}]_{Q\times N}, \xi_{is} \in \mathcal{N}(0,1) \right\} \tag{4}$$

by the infinite set $\mathcal{N}(0,1)$ of standard normal variate's values, where the matrix mapping $\Psi(\mathbf{Y}, \sigma_h)$ returns $Q \times N$ matrix with noised N classes representatives modeling probable

object's distortions at the level σ_h [13]. In the case of STS images, this mapping is applied successively for scaling $\Psi_{\text{scale}}\left(\mathbf{Z}, \sigma_h^{\langle\text{scale}\rangle}\right)$, turning $\Psi_{\text{turn}}\left(\mathbf{Z}, \sigma_h^{\langle\text{turn}\rangle}\right)$, and shifting $\Psi_{\text{shift}}\left(\mathbf{Z}, \sigma_h^{\langle\text{shift}\rangle}\right)$ representatives of N classes in $Q \times N$ matrix $\mathbf{Z}$:

$$\Psi\left(\mathbf{Y}, \sigma_h\right) = \Psi_{\text{shift}}\left(\Psi_{\text{turn}}\left(\Psi_{\text{scale}}\left(\mathbf{Y}, \sigma_h^{\langle\text{scale}\rangle}\right), \sigma_h^{\langle\text{turn}\rangle}\right), \sigma_h^{\langle\text{shift}\rangle}\right) \tag{5}$$

by some relationships among $\sigma_h = \sigma_h^{\langle\text{shift}\rangle}$ and $\sigma_h^{\langle\text{scale}\rangle} > 0$ and $\sigma_h^{\langle\text{turn}\rangle} > 0$. The set (4) feeds the input of TLPNLTF for $Q_{\text{pass}} \in \mathbb{N}$ times until validation error starts increasing. For making sure that the pure representatives (3) have not been disassociated from those N classes, the input of TLPNLTF is re-fed with the set (3) at the final third stage of the training routine.

For ensemble of $B \in \mathbb{N} \setminus \{1\}$ trained TLPNLTF (1), BTR starts with generating the training set (4) whose parameters $\{R, H, \rho, \sigma_0\}$ may differ from those ones when a TLPNLTF is trained. The set (4) is re-generated for $T \in \mathbb{N}$ times. Thus the training set for BTR includes $M = (R+H) \cdot N \cdot T$ training samples. At the q-th iteration of boosting, these samples have the weights in vector $\mathbf{D}(q) = \left[d_\tau(q)\right]_{1\times M}$ by $q = \overline{1, q_0}$ at some final iteration number q_0 [14].

Initially, $d_\tau(1) = M^{-1} \;\; \forall \tau = \overline{1, M}$. Matrix $\mathbf{A} = \left[\bar{a}_{\alpha\tau}\right]_{B\times M}$ is of flags of classifiers' correct responses, where $\bar{a}_{\alpha\tau} = 1$ is the correct classification of τ-th sampled object by the α-th TLPNLTF, otherwise $\bar{a}_{\alpha\tau} = 0$. The classifiers' weighted errors are in matrix $\mathbf{E}(q) = \left[\eta_\alpha(q)\right]_{B\times M}$, where the α-th classifier's weighted error

$$\eta_\alpha(q) = \sum_{\tau=1}^{M} d_\tau(q) \cdot \left(1 - \bar{a}_{\alpha\tau}\right), \; \alpha = \overline{1, B}. \tag{6}$$

BTR starts from $q = 1$. One after another, weighted errors (6) and the best TLPNLTF

$$\alpha_*(q) \in \arg\min_{\alpha=\overline{1, B}} \eta_\alpha(q) \tag{7}$$

along with the minimal weighted error (MWE)

$$\eta_*(q) = \min_{\alpha=\overline{1, B}} \eta_\alpha(q) \tag{8}$$

are found. MWE (8) lets learn the coefficient

$$\gamma(q) = 1 - \eta_*(q) \tag{9}$$

and calculate the new distribution $\mathbf{D}(q+1)$ of weights

$$d_\tau(q+1) = \frac{\tilde{d}_\tau}{\sum_{\upsilon=1}^{M} \tilde{d}_\upsilon} \;\; \text{by} \;\; \tilde{d}_\tau = d_\tau(q) \cdot \exp\left[-\gamma(q)\left(2 \cdot \bar{a}_{\alpha_*(q),\tau} - 1\right)\right] \tag{10}$$

over M training samples. BTR continues if

$$\eta_*(q) < 1 - N^{-1} - \varepsilon \tag{11}$$

for some $\varepsilon \geqslant 0$ tolerating MWE. While inequality (11) is true then $\tilde{q} = q$ and $q = \tilde{q} + 1$, and (6) — (10) are re-found. As soon as inequality (11) becomes false then $q_0 = q$ and there are calculated the coefficients

$$\tilde{\gamma}(q)=\frac{\gamma(q)}{\sum\limits_{p=1}^{q_0}\gamma(p)} \quad \text{by} \quad q=\overline{1,\, q_0} \tag{12}$$

for convex combination of classifiers

$$\tilde{v}_s=\sum_{\alpha=1}^{B}\beta(\alpha)v_s(\alpha) \tag{13}$$

with weights

$$\beta(\alpha)=\sum_{q\in\left\{\overline{1,\, q_0}\right\},\, \alpha=\alpha_*(q)}\tilde{\gamma}(q), \tag{14}$$

where $v_s(\alpha)$ is the s-th output neuron value of the α-th TLPNLTF. The boosted classifier output is

$$s_*\in\arg\max_{s=\overline{1,\, N}}\tilde{v}_s\,. \tag{15}$$

For recognizing STS 60×80 monochrome images [13], there have been prepared 36 trained TLPNLTF by

$$\left\{R=2,\, H=8,\, \rho\in\{0.01,\, 0.02\},\, \sigma_0=1,\, \sigma_h^{\langle\text{scale}\rangle}=\sigma_h^{\langle\text{turn}\rangle}=0.2\sigma_h^{\langle\text{shift}\rangle}\right\} \tag{16}$$

with $Q_{\text{HL}}=300$ and $N=26$, whose averaged REP $\left\{p_{\text{RE}}(\alpha)\right\}_{\alpha=1}^{36}$ are in Figure 1. Variances of REP over 26 existing classes $\left\{v_{\text{REP}}(\alpha)\right\}_{\alpha=1}^{36}$ are in Figure 2. Having initialized BTR for every $B=\overline{2,\, 36}$ with

$$\{R=1,\, H=8,\, \rho=0,\, \sigma_0=1,\, \varepsilon=0\} \tag{17}$$

in the training set (4), which is re-generated for $T=100$ times, the averaged REP $\tilde{p}_{\text{RE}}(B)$ is decreasing against the number of classifiers in the ensemble (there is an almost decreasing barred realization of this stochastic polyline in Figure 3). The inequality

$$\tilde{p}_{\text{RE}}(2)<\min_{\alpha=\overline{1,\, 36}}p_{\text{RE}}(\alpha) \tag{18}$$

is true for the realization in Figure 3, and it is expected that (18) is true for the decreasing expectance of the averaged REP (as this is a stochastic polyline). Figure 4 with variance of REP over 26 classes $\tilde{v}_{\text{REP}}(B)$ confirms that the greater number of classifiers the harder decrement is. The expected polyline of the variance seems to be decreasing also, and the inequality

$$\tilde{v}_{\text{REP}}(2)<\min_{\alpha=\overline{1,\, 36}}v_{\text{REP}}(\alpha) \tag{19}$$

is true for the realization in Figure 4. However, it is obvious that the rate of stochasticity (volatility) of polyline $\tilde{v}_{\text{REP}}(B)$ is higher than the rate of stochasticity of polyline $\tilde{p}_{\text{RE}}(B)$. Therefore, we see a protuberance in Figure 4 starting from $B=13$ right to $B=34$. Local protrusions by $B\in\left\{\overline{29,\, 33}\right\}$ and at $B=26$ along with crevasses at $B\in\{12,\, 28,\, 34\}$ sharpen the impression about highly stochastic polyline $\tilde{v}_{\text{REP}}(B)$.

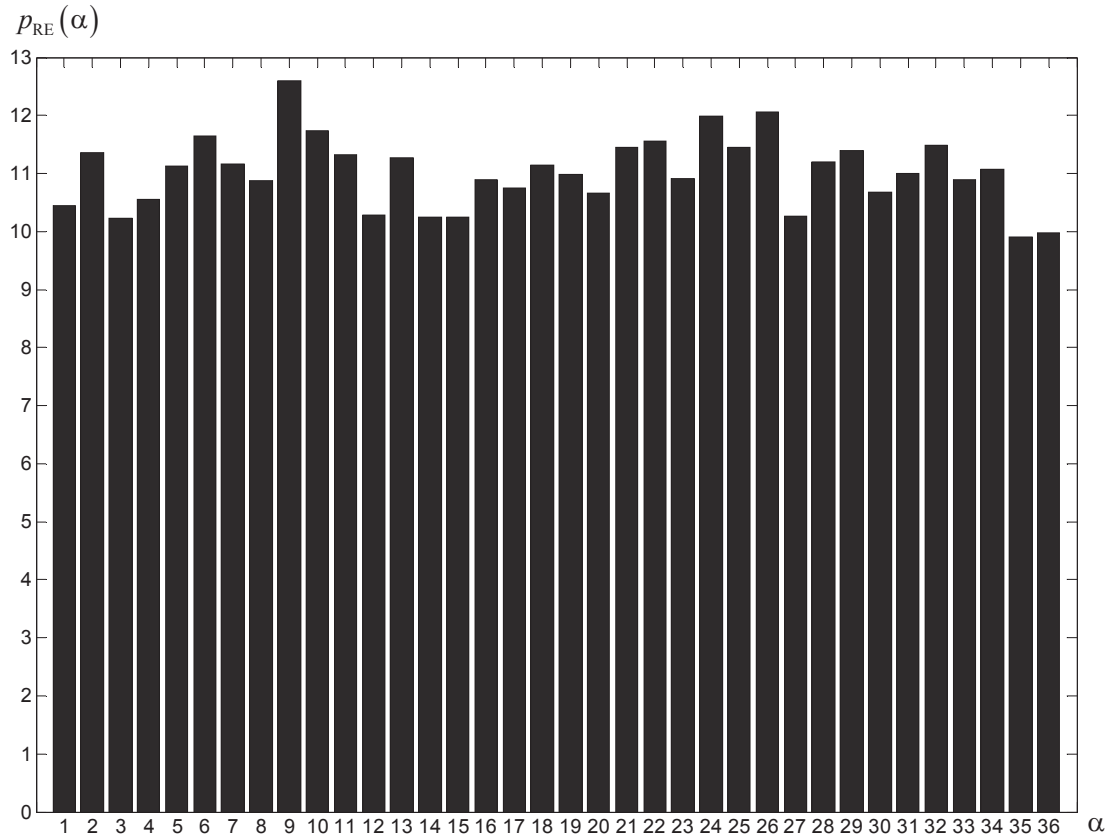

Figure 1. Averaged REP of preliminarily prepared 36 trained TLPNLTF to be boosted within ensembles of them

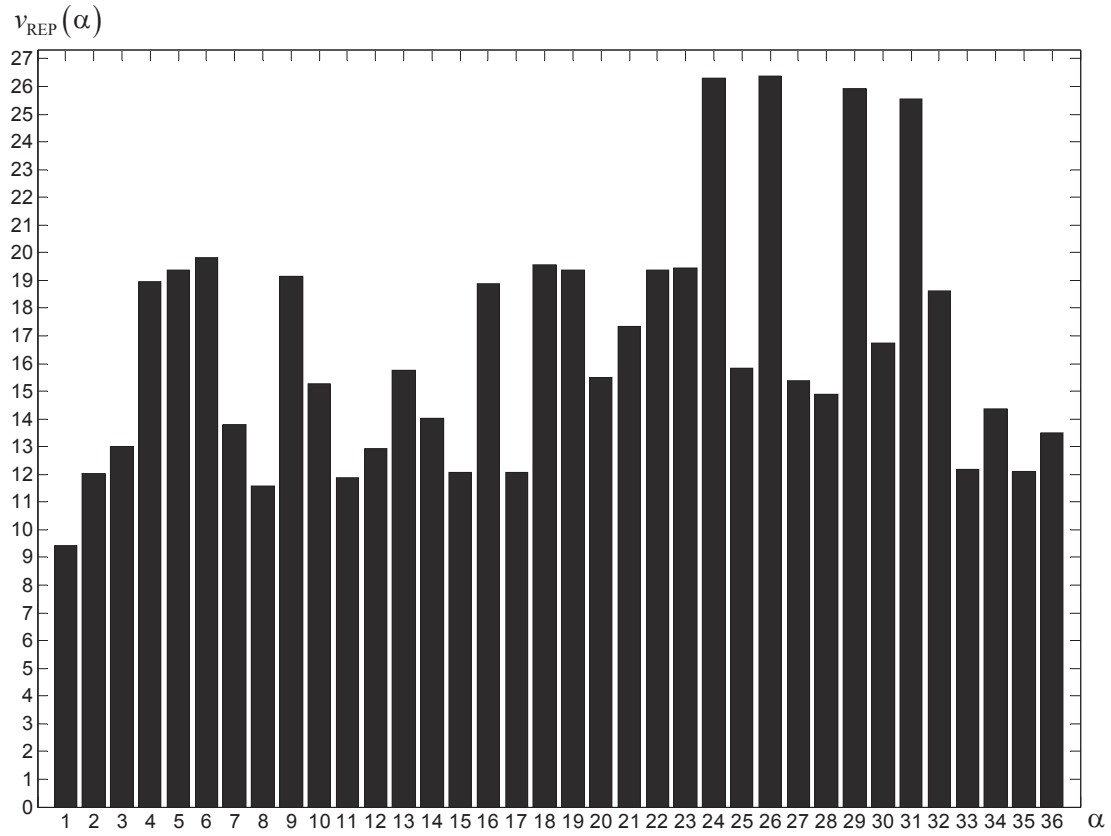

Figure 2. Variances of REP over 26 existing classes of preliminarily prepared 36 trained TLPNLTF before boosting

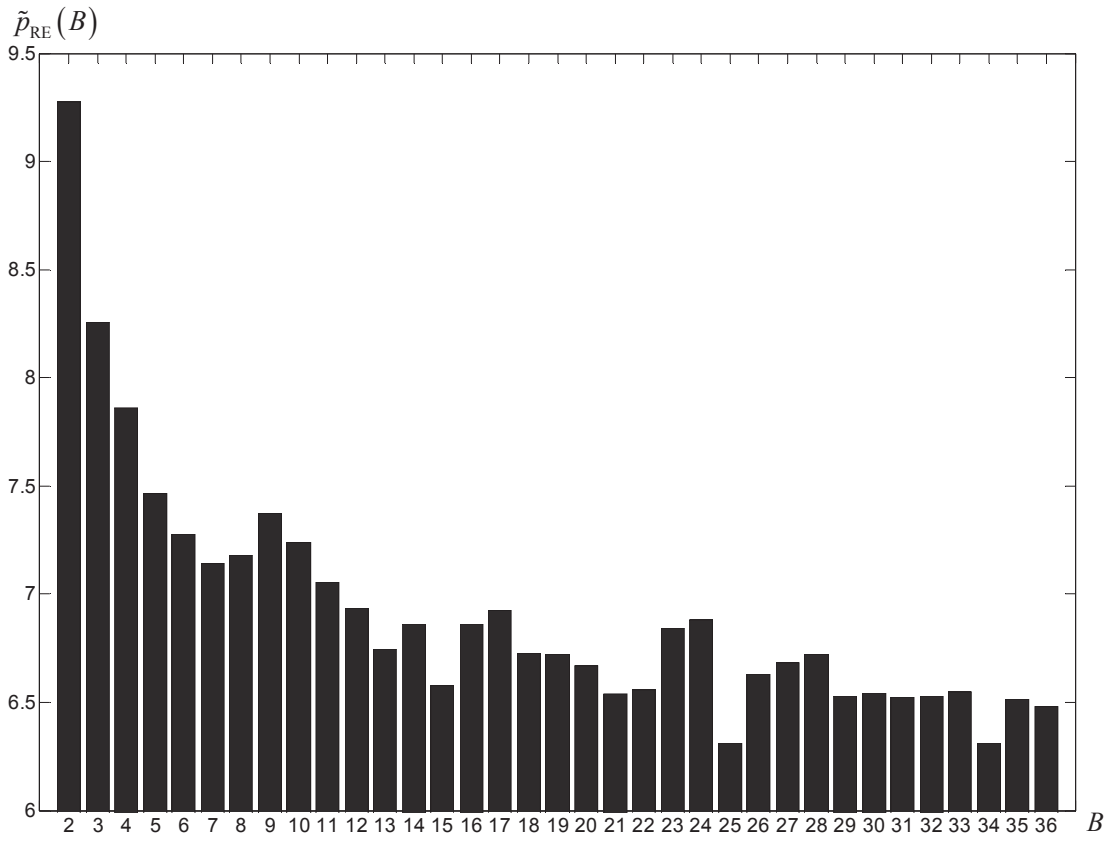

Figure 3. Averaged REP $\tilde{p}_{\mathrm{RE}}(B)$ against the number of TLPNLTF classifiers in the ensemble

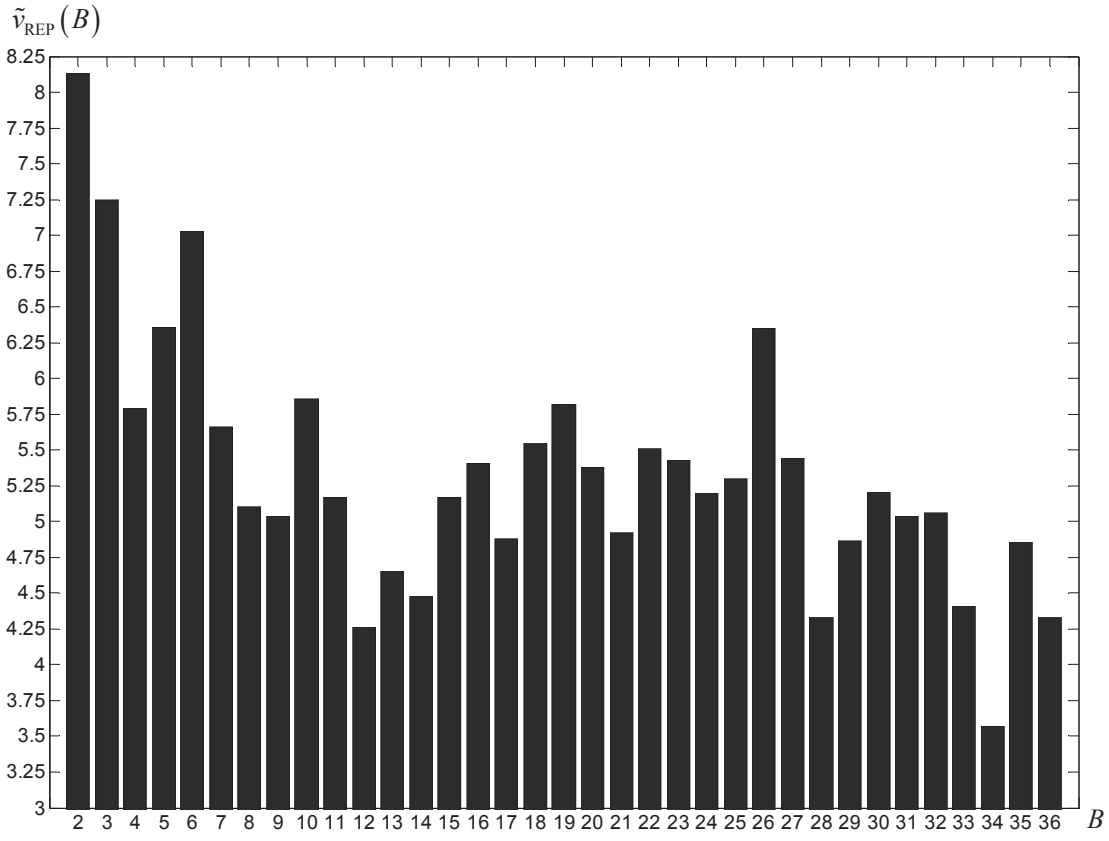

Figure 4. Variance of REP over 26 classes $\tilde{v}_{\mathrm{REP}}(B)$ against the number of TLPNLTF classifiers in the ensemble

The TLPNLTF boosting gain is displayed in Figure 5 with the ratios for the averaged REP

$$g_{\mathrm{REP}}(B)=\frac{\min\limits_{\alpha=1,\,B} p_{\mathrm{RE}}(\alpha)}{\tilde{p}_{\mathrm{RE}}(B)} \tag{20}$$

and variance of REP

$$g_{\tilde{v}_{\mathrm{REP}}}(B)=\frac{\min\limits_{\alpha=1,\,B} v_{\mathrm{REP}}(\alpha)}{\tilde{v}_{\mathrm{REP}}(B)} \tag{21}$$

over 26 classes. As it is seen, the gains are increasing with the increasing number of TLPNLTF classifiers in the ensemble. Clearly, increment for REP gain is more stable and predictable than increment for REP variance gain. Nonetheless, saturation of the increments must be existing. This is because speed of the increment is apparently slowing down when $B>12$.

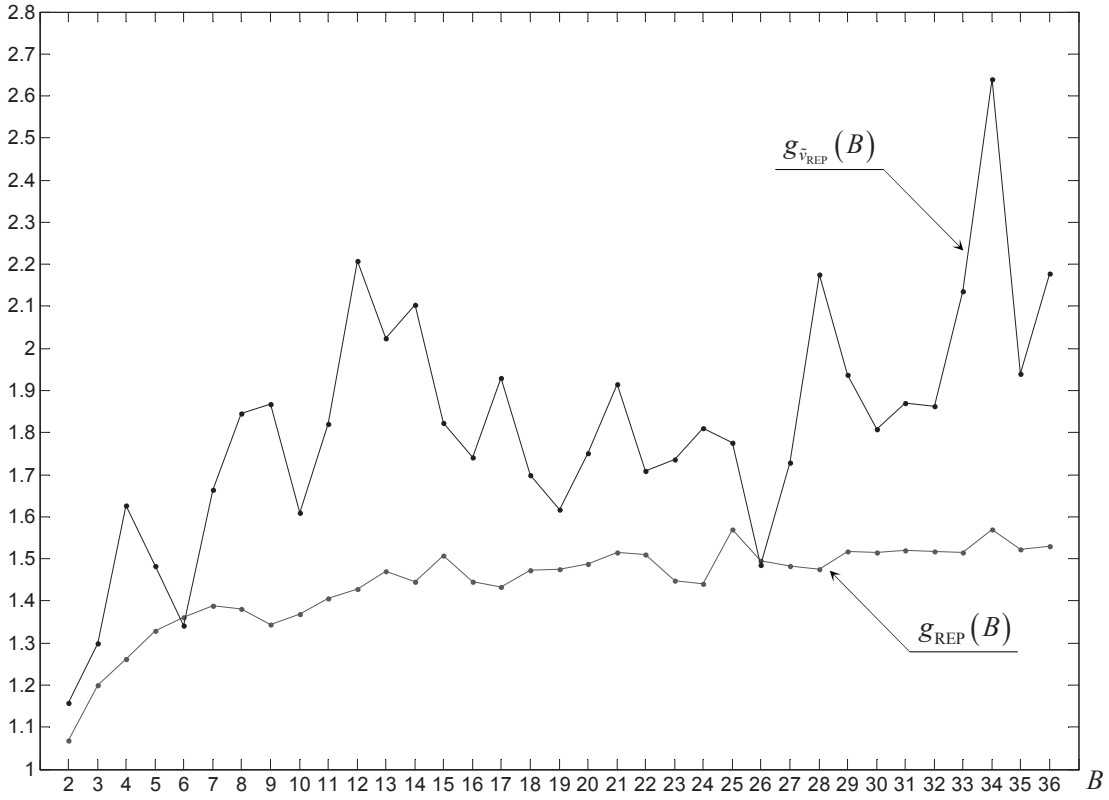

Figure 5. The TLPNLTF boosting gain against the number of TLPNLTF classifiers in the ensemble

The ratio showing the TLPNLTF boosting gain is expected to be the highest at the maximal number of TLPNLTF classifiers in the ensemble. In the tested case, it is $B=36$. Predictably, for $B>36$ those gains for REP and for variance of REP over 26 classes will slowly grow. However, increasing classification accuracy with heavy boosting TLPNLTF ensemble up to have

$$\lim_{B\to\infty} \tilde{p}_{\mathrm{RE}}(B)=0 \tag{22}$$

and

$$\lim_{B\to\infty} \tilde{v}_{\mathrm{REP}}\left(B\right) = 0 \tag{23}$$

is improbable. In machine learning, the registered saturation effect in Figures 3 — 5 is an ordinary phenomenon.

5. Results and discussion

With boosting ensembles of heavy TLPNLTF whose number $B > 20$, classification accuracy in recognizing STS images is increased for 33 %, what corresponds to the 50 % REP gain seen in Figure 5. Probably, there is an ultimate increment that cannot be exceeded for the considered problem. Figure 5 hints that the ultimate increment is about 38 %, what corresponds to the saturation 60 % REP gain. Variance of REP over 26 classes is decreased with the ensemble for more than 50 %, corresponding to the 100 % variance-of-REP gain. Nevertheless, volatility of the variance decrement gain is very high.

The obtained gain result shall be propagated over problems of classifying objects having a few thousands features within two or three tens of classes. This is so because there are $Q = 4800$ features and $N = 26$ classes in the problem considered and investigated above. And if integers Q and N differ from 4800 and 26 just for a few percent, then the gains must be close to realizations in Figure 5. Only one should remember that Figure 1 and Figure 3 show REP averaged over the whole range of noise modeling STS images. At maximal level $\sigma_H = \sigma_0$ of noise modeling ultimate object's distortions by dint of (5), the gain is less, though. While recognizing STS images distorted ultimately at $\sigma_8 = \sigma_0 = 1$, the gain is approximately one third, i. e. $g_{\mathrm{REP}}\left(36\right) \in \left(1.31; 1.34\right)$ if those REP in (20) are calculated at $\sigma_0 = 1$ only. But under the reduced distortions by half, $g_{\mathrm{REP}}\left(36\right) > 5$.

Another peculiarity is that the heavier classifier the less gain is expected. And vice versa, the gain is believed to be greater for less Q requiring lighter TLPNLTF classifier. Here the TLPNLTF lightness is in "inverse proportion" to number $Q_{\mathrm{HL}} \cdot \left(Q + N + 1\right) + N$ of coefficients within matrices (2).

It ought to be mentioned that recognition with the ensemble is lingered over calculating B outputs of TLPNLTF. The recognition operation part itself, including convex combination of classifiers (13) with weights (14) by coefficients (12) for calculating the boosted classifier output (15), takes insignificant time. Obviously, this demerit could be compensated with parallel calculations.

6. Conclusion

The stated method of a strong classifier construction is a kind of straight boosting, when the ensemble is formed outright, for the given number B of weak (initial) classifiers. Every initial classifier itself can perform as well. Nevertheless, the TLPNLTF classifiers' ensemble classification accuracy can be increased with convex combination of classifiers (13). In the investigated problem of recognizing medium format STS images, this increment comes with the inequalities (18) and (19).

Ratios (20) and (21) are principal to claim the gain of the boosting. If they are greater than 1, and they are increasing polylines, the boosting gain is effective enough. However, the TLPNLTF boosting gain with the ratios (20) and (21) depends on the range of noise modeling STS images. For a common problem of classification, the shape and monotonicity properties of polylines (20) and (21) depend on the range of the object's distortions.

There are two crucial distinctions of the stated TLPNLTF boosting method from other multiclass boosting approaches. The first one is the linear rule (9) for re-calculation of distribution (10) over M training samples while BTR runs. The second one is the condition (11) letting stop BTR if MWE becomes too great. Elements of distribution (10) are calculated as well for binary classification approach (AdaBoost), but here we have used heavy

TLPNLTF classifiers, rather than weak learners. This is an evidence of that the exponential loss function in (10) is universal in weighting difficultly classified objects with the set of M training samples. Generally, the straight boosting method is applicable for any classification problem, where more than one homogeneous TLPNLTF is available. The method earnest restriction is that convergences (22) and (23) are unlikely. Therefore, classification accuracy does not seem to be increased infinitely.

Being based on convex combination of TLPNLTF classifiers (13) with weights (14) by coefficients (12) for calculating the boosted classifier output (15) along with statements (6) — (11), strong classifier constructions are easily programmable. The registered classification accuracy increment of 36 TLPNLTF ensemble and the scored gain allow to apply the ensemble and the similarly constructed TLPNLTF ensembles for problems of classifying objects within two or three tens of classes by a few thousands features in every class. Investigation on heavy boosting ensembles could be advanced focusing on ultimate and diverse object's distortions. The boosting gain at these distortions (strictly including, for instance, the STS distortion type) may be insignificant needing much more (heavier) TLPNLTF classifiers.

7. Acknowledgements

The work is technically supported by the Parallel Computing Center at Khmelnitskiy National University (http://parallelcompute.sourceforge.net).

References

[1] Arulampalam, G.; Bouzerdoum, A. A generalized feedforward neural network architecture for classification and regression. *Neural Networks*, 16 (5 — 6): pp. 561 — 568, 2003.

[2] Benardos, P. G.; Vosniakos, G.-C. Optimizing feedforward artificial neural network architecture. *Engineering Applications of Artificial Intelligence*, 20 (3): pp. 365 — 382, 2007.

[3] Castillo, P. A.; Merelo, J. J.; Arenas, M. G.; Romero, G. Comparing evolutionary hybrid systems for design and optimization of multilayer perceptron structure along training parameters. *Information Sciences*, 177 (14): pp. 2884 — 2905, 2007.

[4] Farid, D. M.; Zhang, L.; Rahman, C. M.; Hossain, M. A.; Strachan, R. Hybrid decision tree and naïve Bayes classifiers for multi-class classification tasks. *Expert Systems with Applications*, 41 (4/2): pp. 1937 — 1946, 2014.

[5] Fernández-Baldera, A.; Baumela, L. Multi-class boosting with asymmetric binary weak-learners. *Pattern Recognition*, 47 (5): pp. 2080 — 2090, 2014.

[6] Hagan, M. T.; Menhaj, M. B. Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks*, 5 (6): pp. 989 — 993, 1994.

[7] Kathirvalavakumar, T.; Jeyaseeli Subavathi, S. Neighborhood based modified backpropagation algorithm using adaptive learning parameters for training feedforward neural networks. *Neurocomputing*, 72 (16 — 18): pp. 3915 — 3921, 2009.

[8] Nie, Q.; Jin, L.; Fei, S. Probability estimation for multi-class classification using AdaBoost. *Pattern Recognition*, 47 (12): pp. 3931 — 3940, 2014.

[9] Özçift, A. Random forests ensemble classifier trained with data resampling strategy to improve cardiac arrhythmia diagnosis. *Computers in Biology and Medicine*, 41 (5): pp. 265 — 271, 2011.

[10] Plaza, J.; Plaza, A.; Perez, R.; Martinez, P. On the use of small training sets for neural network-based characterization of mixed pixels in remotely sensed hyperspectral images. *Pattern Recognition*, 42 (11): pp. 3032 — 3045, 2009.

[11] Polat, K.; Güneş, S. A novel hybrid intelligent method based on C4.5 decision tree classifier and one-against-all approach for multi-class classification problems. *Expert Systems with Applications*, 36 (2/1): pp. 1587 — 1592, 2009.

[12] Puissant, A.; Rougier, S.; Stumpf, A. Object-oriented mapping of urban trees using Random Forest classifiers. *International Journal of Applied Earth Observation and Geoinformation*, 26: pp. 235 — 245, 2014.

[13] Romanuke, V. V. An attempt for 2-layer perceptron high performance in classifying shifted monochrome 60-by-80-images via training with pixel-distorted shifted images on the pattern of 26 alphabet letters. *Radioelectronics, informatics, control*, 2: pp. 112 — 118, 2013.

[14] Romanuke, V. V. Accuracy improvement in wear state discontinuous tracking model regarding statistical data inaccuracies and shifts with boosting mini-ensemble of two-layer perceptrons. *Problems of tribology*, 4: pp. 55 — 58, 2014.

[15] Shen, C.; Li, H.; van den Hengel, A. Fully corrective boosting with arbitrary loss and regularization. *Neural Networks*, 48: pp. 44 — 58, 2013.

[16] Siniscalchi, S. M.; Yu, D.; Deng, L.; Lee, C.-H. Exploiting deep neural networks for detection-based speech recognition. *Neurocomputing*, 106: pp. 148 — 157, 2013.

[17] Zheng, S. QBoost: Predicting quantiles with boosting for regression and binary classification. *Expert Systems with Applications*, 39 (2): pp. 1687 — 1697, 2012.