International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 17 (2018) pp. 13292-13295

Binary Images Classification Algorithm Based on Moore-Penrose Inverse Matrix

Nicolás Ortiz[1], Robinson Jiménez[2], Mauricio Mauledoux M.[3]

Department of Mechatronics Engineering, Nueva Granada Military University, Bogotá, Colombia.

Abstract

This paper presents a novel binary images classification algorithm that uses Moore-Penrose inverse matrix to update a weight matrix based on the error obtained for an image sample in a training set. It can be seen as a perceptron in terms of matrices. Weight matrix update is made estimating a desire weight for a desire output using Moore-Penrose inverse. Evaluation was made over a 3 image classification problem example and using a sign traffic dataset. Obtained results shows good algorithm performance and accuracy if image size is appropriate. Results suggest training images must have rich features and discriminant information between classes.

Keywords: Binary Images, Image classification, Moore-Penrose inverse, Modified Perceptron, Discriminant features.

INTRODUCTION

Machine learning is a subject that studies how to use computers to simulate human learning activities [1], being visual object classification or recognition one of the most important tasks. Nevertheless, classification between the objects is easy task for humans but it has proved to be a complex problem for machines [2]. In this sense, as a field that is in a continuous development, machine learning has been made many advancements in pattern recognition by using several methodologies and techniques among which it can be find: supervised and unsupervised methodologies; parametric and nonparametric techniques; contextual and spectral-contextual classifiers; and hard and soft classifications [3]. Framed in these methodologies useful algorithms have been successfully tested an implemented with different image classification purposes including: identifying a terrorist from only two finger using Artificial Neural Network (ANN) [4], hand segmentation based using Artificial Neural Network (ANN) [5], face recognition with deep neural nets (DNN) [6]–[10], Face Recognition by using decision trees [11] among others.

In this document, it is presented a novel algorithm for binary images classification focused on treating images as whole feature by using a matrix representation of a linear classification function. First, classification function definition is made. Then, the classification function parameters update that is based on the definition and properties of Moore-Penrose inverse (see [12]) and some mathematical considerations for dealing with the activation function are presented. Finally, obtained experimental results are shown.

CLASSIFICATION ALGORITHM

In this section classification algorithm is described taking into account the mathematical assumptions made to update classification weight matrix by using Moore-Penrose inverse.

Classification Function

Let $\mathcal{X} := \{A_1, A_2, \cdots, A_n\} \subseteq M_{p\times q}$ and $\mathcal{Y} := \{y_1, y_2, \cdots, y_n\} \subseteq [-1,1] \times \cdots \times [-1,1]$ the input and output training sets, where $M_{p\times q}$ is the set of all $p \times q$ matrices, and $[-1,1] \times \cdots \times [-1,1]$ the output space depending on the number of classes, this means, $[-1,1] \times [-1,1]$ for two classes classification task, $[-1,1] \times [-1,1] \times [-1,1]$ for three classes classification task and so on such that $y_i \in [-1,1] \times \cdots \times [-1,1]$. A simple classification function, for a perceptron is denoted by $f(x) = \phi(\mathrm{w} \cdot \mathrm{x})$, where ϕ is an activation function. In this sense, a simple classification function in terms of matrixes can be defined as:

$$f(A) = \phi(\mathbf{W} \cdot A \cdot \Upsilon)$$

Where $\mathbf{W}$ is a weight matrix and Υ is called output resizing matrix that, basically, maintain dimensions consistency. In this way, for a l number of classes, and a desire $l \times 1$ output $f(A) = y$, matrices dimensions must be: $l \times p$ for $\mathbf{W}$ and $q \times 1$ for ⍰.

Weigh Update

It is clear that for a classification problem, it is desire that $\forall (A_i, y_i) \in \mathcal{X} \times \mathcal{Y}, f(A_i) = y_i$. However, for an initial weight matrix $\mathbf{W}^{(\mathrm{k})}$ and a constant random Υ matrix it is expected that $f(A_i) \neq y_i$. Thereby, the weight update problem is reduced to find $\Delta\mathbf{W}^{\mathrm{k}} = \mathbf{W}_{\mathrm{d}}^{\mathrm{k}} - \mathbf{W}^{\mathrm{k}}$ matrix, where $\mathbf{W}_{\mathrm{d}}^{\mathrm{k}}$ is the desire weight matrix, such that $f(A_i) = \phi\left(\left(\mathbf{W}^{\mathrm{k}} + \Delta\mathbf{W}^{\mathrm{k}}\right) \cdot A_i \cdot \Upsilon\right) = y_i$. Then, it can be inferred that, if ϕ is a linear function defined as $\phi(s) = s$:

$$\phi\left(\mathbf{W}_{\mathrm{d}}^{\mathrm{k}} \cdot A_i \cdot \Upsilon\right) - \phi\left(\mathbf{W}^{\mathrm{k}} \cdot A_i \cdot \Upsilon\right) = y_i - f(A_i) := \Delta y \quad (1)$$

$$\mathbf{W}_{\mathrm{d}}^{\mathrm{k}} \cdot A_i \cdot \Upsilon - \mathbf{W}^{\mathrm{k}} \cdot A_i \cdot \Upsilon = \Delta y \quad (2)$$

Then:

$$\Delta\mathbf{W}^{\mathrm{k}} \cdot A_i \cdot \Upsilon = \Delta y$$

By using the Moore-Penrose inverse denoted by $\mathcal{X}^{+}$, and its properties:

$$\Delta \mathbf{W}^{\mathrm{k}} = \Delta y (A_i \cdot \square)^{+} \quad (3)$$

However, ϕ is not always a linear function. In fact, for classification tasks, ϕ is commonly a nonlinear mapping from $\mathbb{R}^l$ to $[-1,1] \times \cdots \times [-1,1]$, this means that $\exists x, y$ such that $\phi(x) + \phi(y) \neq \phi(x+y)$. Nevertheless, it can be think on an approximation in such a way that:

$$\phi(x) + \phi(y) \approx \frac{1}{\Omega(\cdot)} \cdot (x+y) \quad (4)$$

Where $\Omega(\cdot)$ is a function. It may think $\Omega(\cdot)$ takes values on a $\mathbb{R}^l$ rectangle $(-1,1) \times \cdots \times (-1,1)$ for a mapping of $[-1,1] \times \cdots \times [-1,1]$. By using Equation 4 in Equation 1, it is obtained:

$$\frac{1}{\Omega(\cdot)} \Delta \mathbf{W}^{\mathrm{k}} \cdot A_i \cdot \Upsilon \approx \phi\left(\mathbf{W}_{\mathrm{d}}^{\mathrm{k}} \cdot A_i \cdot \Upsilon\right) - \phi\left(\mathbf{W}^{\mathrm{k}} \cdot A_i \cdot \Upsilon\right) = \Delta y \quad (5)$$

It can be noted in Equation 5 that depending on Δy, $\Omega(\cdot)$ increase or decrease its value. Thereby, may be a mapping from $[-1,1] \times \cdots \times [-1,1]$ to $(-1,1) \times \cdots \times (-1,1)$ that depends on Δy. Then:

$$\frac{1}{\Omega(\Delta y)} \Delta \mathbf{W}^{\mathrm{k}} \cdot A_i \cdot \Upsilon \approx \Delta y$$

$$\Delta \mathbf{W}^{\mathrm{k}} \approx \Omega(\Delta y) \cdot \Delta y \cdot (A_i \cdot \Upsilon)^{+} \quad (6)$$

Finally, Ω function selection must be made taking into account that, for an activation function that maps data into the $[-1,1] \times \cdots \times [-1,1]$ space:

(i) $\Omega(\Delta y_i) \in (0,1)$, where $\Omega(\Delta y)_i$ is the i-th component of $\Omega(\Delta y)$.

(ii) If $|\Delta y_i| > |\Delta y_j|$ then, in virtue of Equation 5 and taking into account that big Δy_i values implies that:

$$\left|\Delta \mathbf{W}^{\mathrm{k}} \cdot A_i \cdot \Upsilon\right| \ll \left|\phi\left(\mathbf{W}_{\mathrm{d}}^{\mathrm{k}} \cdot A_i \cdot \Upsilon\right) - \phi\left(\mathbf{W}^{\mathrm{k}} \cdot A_i \cdot \Upsilon\right)\right|$$

Then $\Omega(\Delta y_i) < \Omega(\Delta y_j)$.

(iii) $\Omega(\Delta y_i) \approx \Delta y_i$ when $\Delta y_i \approx 0$.

Ω-Function

Taking into account the Ω function requirements described above, the proposed function was:

$$\Omega(\Delta y) = \lambda^{-|\Delta y|} \quad (7)$$

Where λ is a constant variable which value depends on the activation function ϕ. In Figures 1 and 2 different approximations given some λ values for $\phi(x) = \tanh(x)$ and $\phi(x) = \arctan(x)$ are shown.

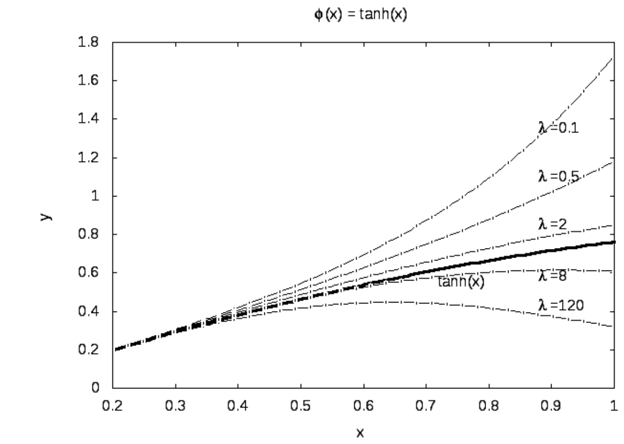

Figure 1. Different approximations given some λ for an activation function $\phi(x) = \tanh(x)$ to $\phi(x) \approx \Omega(x - \tanh(x)) \cdot x$

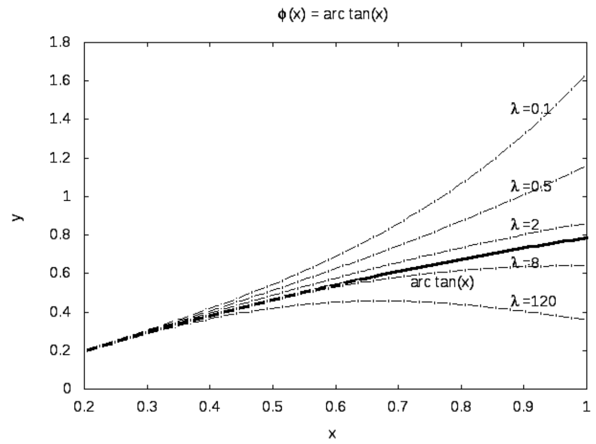

Figure 2. Different approximations given some λ for an activation function $\phi(x) = \arctan(x)$ to $\phi(x) \approx \Omega(x - \arctan(x)) \cdot x$

As it can be appreciated, a good approximation is obtained with $2 \leq \lambda \leq 8$ for both activation functions, $\phi(x) = \tanh(x)$ and $\phi(x) = \arctan(x)$

Classification Algorithm

In Table I, the classification algorithm is described. A learning rate α is used, in terms of making sure that no local minima where dismiss and to make sure algorithm does not diverge. In this way, by using Equation 6, weight update stays as:

$$\mathbf{W}^{\mathrm{k+1}} = \mathbf{W}^{\mathrm{k}} + \alpha \cdot \Omega(\Delta y) \cdot \Delta y \cdot (A_i \cdot \Upsilon)^{+} \quad (8)$$

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 17 (2018) pp. 13292-13295

Table 1. Classification Algorithm

1. Let $\mathcal{X} := \{A_1, A_2, \cdots, A_n\}$ and $\mathcal{Y} := \{y_1, y_2, \cdots, y_n\}$ the input and output training sets.
2. Initialize $\mathbf{W}^{(0)}$ and □, and pick up a good λ value.
3. Take a tuple (A_i, y_i) and update the weight matrix as follows:

   $$\mathbf{W}^{k+1} = \mathbf{W}^{k} + \alpha \cdot \Omega(\Delta y) \cdot \Delta y \cdot (A_i \cdot □)^{+}$$

   Until $f(A_i) \approx y_i,\ \forall (A_i, y_i)$
4. end

SIMULATION RESULTS

In this section obtained simulation results for an image classification example and a traffic signs recognition problem using the BelgiumTS traffic sign dataset [13] are presented. Results are presented in terms of the minimum square error. It was used $\phi(x) = \tanh(x)$ as activation function, $\alpha = 0.1$, $\lambda = e$ and □ $= [1, 1, \cdots, 1]^T$. Initial Weight matrix, $\mathbf{W}^{(0)}$, is initialized with random values in the range $[-1, 1]$.

Image Pattern Recognition

Consider the problem of classifying the next image patterns:

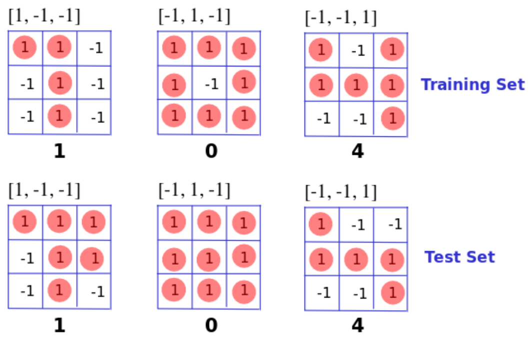

Figure 3. Classification Sets for Image Pattern Recognition

Obtained epoch error is shown below in Figure 4.

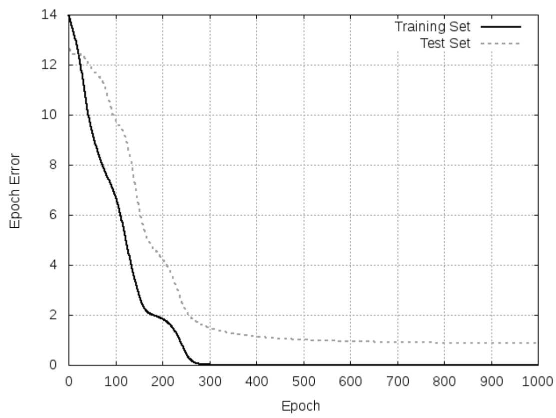

Figure 4. Minimum Square Error per Epoch

Traffic Sign Recognition

In this section obtained simulation results for a traffic sign recognition problem using the BelgiumTS traffic sign dataset [13] is presented. Classification was made using different image sizes and no image processing steps, except for an image binarization (see Figure 5) and a training set length of 100 samples.

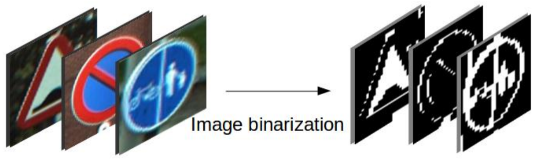

Figure 5. Binary Image Conversion (Images taken from BelgiumTS Traffic dataset)

Below, in Figures 6 and 7, obtained epoch errors for the training and test sets using different image sizes, are shown.

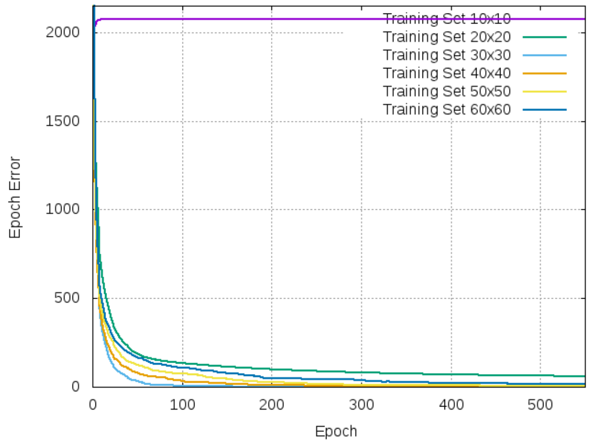

Figure 6. Minimum Square Error per Epoch for the Training set using different image sizes

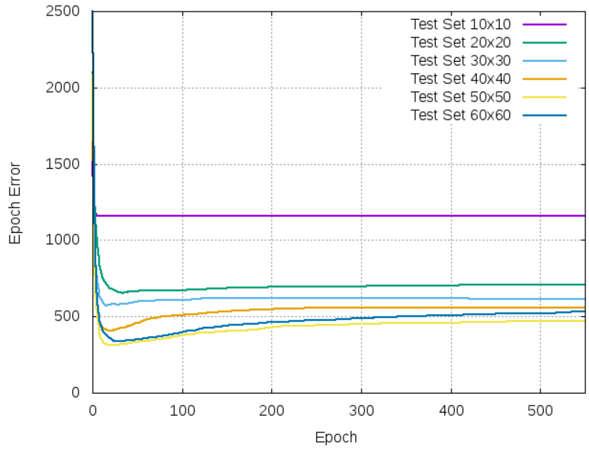

Figure 7. Minimum Square Error per Epoch for the Test set using different image sizes

CONCLUSIONS

A novel classification algorithm, which updating process is made using Moore-Penrose inverse matrix calculation, has been proposed which is useful for image recognition or classification tasks with several classes. Experimental results shows that for a good algorithm performance the image size is critical, this suggest that training images must contain rich features and discriminant information between classes. On the other hand, presented algorithm is simple in construction unlike other approaches as deep neural nets (DNNs) or even Conventional Artificial Neural Nets (ANNs) with a considerable number of neurons, suggesting computational cost is better.

ACKNOWLEDGEMENT

The research for this paper was supported by Davinci research Group of Nueva Granada Military University.

REFERENCES

[1] Wang, H., Ma, C., and Zhou, L., 2009, "A Brief Review of Machine Learning and Its Application," *2009 International Conference on Information Engineering and Computer Science*, pp. 1–4.

[2] Kamavisdar, P., Saluja, S., and Agrawal, S., 2013, "A survey on image classification approaches and techniques," *International Journal of Advanced Research in Computer and Communication Engineering*, 2(1), pp. 1005-1009.

[3] Bhuvaneswari, B. N., and Sivakumar, V. G., 2016, "Novel Image Classification technique using Particle Filter Framework optimised by Multikernel Sparse Representation," *Brazilian Archives of Biology and Technology*, 59(SPE2).

[4] Hassanat, A. B., *et al*, 2017, "Victory sign biometrie for terrorists identification: Preliminary results," in *2017 8th International Conference on Information and Communication Systems, ICICS 2017*, pp. 182–187.

[5] Hassanat, A., Alkasassbeh, M., Al-Awadi, M., and Alhasanat, E., 2015, "Colourbased lips segmentation method using artificial neural networks," in *2015 6th International Conference on Information and Communication Systems, ICICS 2015*, IEEE, pp. 188-193.

[6] Liu, J., 2015, "Targeting Ultimate Accuracy: Face Recognition via Deep Embedding," *CVPR*, pp. 4–7.

[7] Parkhi, O. M., Vedaldi, A., and Zisserman, A., 2015, "Deep Face Recognition," in *Proceedings of the British Machine Vision Conference 2015*, pp. 41.1–41.12.

[8] Schroff, F., Kalenichenko, D., and Philbin, J., 2015, "FaceNet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 815–823.

[9] Sun, Y., Liang, D., Wang, X., and Tang, X., 2015, "DeepID3: Face Recognition with Very Deep Neural Networks," *CVPR*, pp. 2–6.

[10] Taigman, Y., Yang, M., Ranzato, M., and Wolf, L., 2014, "DeepFace: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1701–1708.

[11] Maturana, D., Mery, D., and Soto, A., 2011, "Face recognition with decision tree-based local binary patterns," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6495 LNCS(PART 4), pp. 618–629.

[12] Rakha, M. A., 2004, "On the Moore-Penrose generalized inverse matrix," *Applied Mathematics and Computation*.

[13] Mathias, M., Timofte, R., Benenson, R., and Van Gool, L., 2013, "Traffic sign recognition–How far are we from the solution?," *Neural Networks (IJCNN), The 2013 International Joint Conference on*, IEEE, pp. 1-8.