Error analysis in artificial neural networks: the imbalanced distribution case

R. ALEJO
UAEM
CU UAEM Atlacomulco
Toluca-Atlacomulco Km. 60, 50450
MEXICO
ralejoe@uaemex.mx

J.M. SOTOCA
Universitat Jaume I
Dept. Lenguajes y Sistemas Informáticos
Campus Riu Sec s/n 12071 Castelló
SPAIN
sotoca@uji.es

M. G. DE LA ROSA
UAEM
CU UAEM Atlacomulco
Toluca-Atlacomulco Km. 60, 50450
MEXICO
mgonzalezr@uaemex.mx

Abstract: A comparative empirical study is presented using different cost functions designed to reduce the imbalanced class influence in the training data. This work is focused in the learning and classification process by using perceptron multilayer and radial basis functions neural networks. This artificial neural networks were trained by means of the back-propagation algorithm in batch mode using two class databases.

Key–Words: back-propagation algorithm, class imbalance problem and cost functions.

1 Introduction

An artificial neural network (ANN) with supervised learning, uses a data training sample (TS) previously analyzed by a human expert [1]. This TS is characterized according to the problem to be solved.

Mathematically it can be defined as follows [1]

$$\mathrm{TS} = \mathrm{TS}_1 \cup \mathrm{TS}_2 \cup ... \cup \mathrm{TS}_m \,, \tag{1}$$

where

$$\mathrm{TS}_i = (\mathbf{x}_j, \varphi(\mathbf{x}_j)),\ j = 1, ..., n_i \,, \tag{2}$$

$\mathbf{x}_j = [x_1, x_2, ..., x_d]^T$ is the vector which characterize a specific problem, $\varphi(\mathbf{x}_j)$ the class which belongs to and n_i the number of class samples i.

Generally, the supervised learning methods, like ANN, are designed to work with balanced TS, that is TS where the differences in the number of class samples is not considerable. However, there are several applications where the imbalance problem between classes is significant[2].

Some examples are: fraudulent phone calls identification [3]; defective products detection [4] in the automobile industry context; and also fraudulent credit card transactions [5] where the number of legal transactions is greater than the number of illegal transactions. The imbalanced problem also appears in medical issues, where rare sickness are harlypresented. An imbalanced TS is characterized with a considerable difference in the number of samples for the different classes, in other words, if for some TS_i we have that:

$$\|\mathrm{TS}_i\| << \|\mathrm{TS}_j\| \ i \neq j;\ i, j = 1, ..., m \tag{3}$$

where m is the total number of classes in the TS, the TS is imbalanced.

Recently, the TS imbalanced problem has been considered a critical issue in data mining and in automatic learning [6]. In the context of ANN multilayer perceptron (MLP) trained with the back-propagation algorithm and two classes domain, the class imbalance problem has been formulated as follows: the majority class dominates the training process while the elements of the minority class can be ignored, in consequence the minority class converges very slowly [7].

Several strategies have been proposed in order to face this problem. In [8], the back-propagation algorithm is analyzed and a modification is proposed to speed up the network convergence process. The idea is centered in the gradient vector computation and the gradient vector direction in order to allow error reduction in both classes and in consequence to avoid that the minority class is ignored.

In [7] a very similar idea is exposed to modify the back-propagation algorithm and speed up convergence. This strategy basically consists in including a cost function in the training algorithm and reducing its value using a heuristic strategy in order to decrease the probabilistic impact in the data distribution.

The most popular strategies to face the class imbalance problem are the techniques of under-sampling (which eliminates samples in the majority class) and

[1] Valid definition for classification tasks.

over-sampling (replicates samples in the minority class) [1].

In recent works [6], the class imbalance problem is considered as a cost-sensitive problem where the classification error cost should be different for each class. The main disadvantage of this strategy is the need of a priori information of the problem, therefore the error cost must be quantified before the process.

In this research three strategies are designed to face the imbalanced TS problem during the training phase using the following neural network models: MLP, RBF-NN and RVFL-NN.

2 Error analysis: imbalanced class case

Empirical analysis of the back-propagation algorithm [8] show that the class imbalanced TS problem during the training phase generates unequal contributions to the mean square error (MSE) for the ANN, in other words, the most important contributions to the MSE are given by the majority class. In consequence, the network training process is dominated by the majority class.

Consider a TS with two classes ($m = 2$) and N training samples, such that $N = \sum_i^m n_i$ and n_i is the class sample number i. Also consider that the MSE in each class can be expressed like

$$E_i(U) = \frac{1}{N} \sum_{n=1}^{n_i} \sum_{p=1}^{L} (y_p^n - F_p^n)^2 \,, \qquad (4)$$

in such way that the MSE can be expressed as follows:

$$E(U) = \sum_{i=1}^{m} E_i(U) = E_1(U) + E_2(U) \,. \qquad (5)$$

If $n_1 << n_2$ then $E_1(U) << E_2(U)$ y $\|\nabla E_1(U)\| << \|\nabla E_2(U)\|$. Hence $\nabla E(U) \approx \nabla E_2(U)$. So, $-\nabla E(U)$ is not always the best direction to minimize the MSE in both classes [8].

Considering that the class imbalance problem affects negatively the back-propagation algorithm due to the disproportioned contribution to the MSE classes (Ec. 5), it is possible to include an additional term in the cost function (γ) , in such a way that modifies the imbalanced class in the TS:

$$\begin{aligned} E(U) &= \textstyle\sum_{i=1}^{m} \gamma(i) E_i(U) \\ &= \gamma(1) E_1(U) + \gamma(2) E_2(U) \\ &= \tfrac{1}{N} \textstyle\sum_{i=1}^{m} \gamma(i) \sum_{n=1}^{n_i} \sum_{p=1}^{L} (y_p^n - F_p^n)^2 \,. \end{aligned} \qquad (6)$$

In this way $\gamma(1)\|\nabla E_1(U)\| \approx \gamma(2)\|\nabla E_2(U)\|$ and can be avoided that the minority class will be ignored during the training process.

In this research the following cost functions are studied:

- **Option 0**: $\gamma(i) = 1$, in other words, this is the back-propagation algorithm with not modification.
- **Option 1**: $\gamma(i) = n_{max}/n_i$; where $i = 1, ..., m$; m is the total number of classes, and n_{max} is the number of samples of the majority class..
- **Option 2**: $\gamma(i) = N/n_i$; where $i = 1, ..., m$ and N is the total number of samples.
- **Option 3**: $\gamma(i) = \frac{\|\nabla E_{max}(U)\|}{\|\nabla E_i(U)\|}$, where $\|\nabla E_{max}(U)\|$ is related to the majority class. This function is a simplified version to the one proposed by [8].

3 MSE analysis during the trainning step

To evaluate the strategies presented in section 2, an empirical comparative analysis was done using the MSE measure for each class in three different situations. The databases used were identified as: V2Cls, Phoneme and B2Cls. V2Cls and B2Cls databases are related to the Vowel and Balance databases respectively. Each data set was transformed into a two classes datasets.

In each of them, class 1 is considered the minority class whereas the other classes belongs the majority class. In this way imbalanced data bases with two classes are obtained. The objective is to generate prototype imbalanced problems in order to represent different usual scenarios. The use of databases with two classes allows us to simplify the MSE in each class.

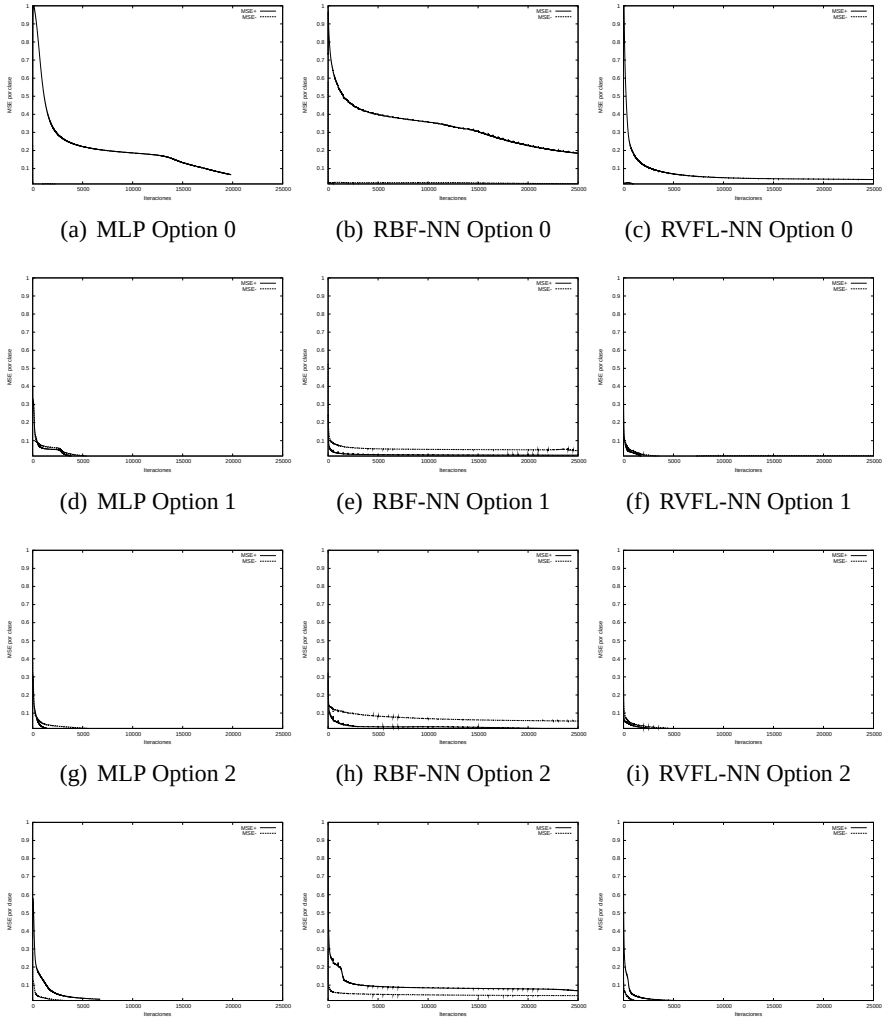
In table 1, the most relevant databases characteristics from the datasets given above are presented. In the last columns, the obtained values for the complexity measures F2, N2, D2 y D3 can be identified [9]. Notice that the datasets are ordered from minimum complexity index database to greater one. It can be also observed that the three databases show different class imbalanced levels between classes.

The evaluation for strategies Option 0, 1, 2 and 3 were done using the MLP, RBF-NN and RVFL-NN models, all trained with the back-propagation algorithm. The free network parameters were defined using a trial an error strategy.

In Fig. 1, the MSE from both classes is reported (MSE+, for the positive or minority class and MSE-,

Table 1: Databases characteristics

Dataset	Size	Atributes	Distribution	Ratio	F2	N2	D2	D3
V2Cls	528	10	48/480	0.100	0.048	0.139	0.172	0.002
Phoneme	5404	5	1586/3818	0.415	0.27	0.258	0.011	0.104
B2Cls	625	4	49/576	0.085	1.0	0.652	0.952	0.142



(a) MLP Option 0 (b) RBF-NN Option 0 (c) RVFL-NN Option 0

(d) MLP Option 1 (e) RBF-NN Option 1 (f) RVFL-NN Option 1

(g) MLP Option 2 (h) RBF-NN Option 2 (i) RVFL-NN Option 2

(j) MLP Option 3 (k) RBF-NN Option 3 (l) RVFL-NN Option 3

Figure 1: MSE by class for V2Cls dataset

for the negative or majority class one) regarding the three RNA models and cost functions (Options 0-3).

In Fig. 1a, 1b, 1c, it can be observed the MSE performance for each class when the TS is imbalanced and the training algorithm is not modified. Initially, the majority class MSE value decreases very fast, meanwhile for the minority class increases. Later the minority class MSE value decreases very slowly.

It is interesting to note that the MLP model converges faster than the RBF-NN and RVFL-NN, whereas for the last two, convergence condition is not observed and both are affected by the class imbalance as can be see in 1b, and 1c. When Option 1 is used during the training phase (see fig. 1d, 1e, 1f) the MSE value decreases uniformly avoiding that the process will be dominated by the majority class. Option 1 convergence effect is important. In Fig. 1d, 1e and 1f can be observed an incremental effect with respect to the

Table 2: Classification performance of V2Cls

MLP	Option 0	Option 1	Option 2	Option 3
Acc	99.43(0.52)	99.62(0.51)	99.62(0.52)	99.43(0.52)
$gmean$	99.69(0.28)	99.79(0.28)	99.79(0.28)	99.69(0.28)
$Kappa$	0.97(0.03)	0.98(0.03)	0.98(0.03)	0.97(0.03)
RBF-NN	Option 0	Option 1	Option 2	Option 3
Acc	98.67(0.85)	99.24(1.04)	98.86(1.57)	98.3(2.05)
$gmean$	95.41(4.25)	99.58(0.58)	99.37(0.87)	99.05(1.14)
$Kappa$	0.92(0.05)	0.96(0.06)	0.94(0.08)	0.91(0.1)
RVFL-NN	Option 0	Option 1	Option 2	Option 3
Acc	99.81(0.42)	99.62(0.52)	99.43(0.85)	99.62(0.52)
$gmean$	99.9(0.23)	99.79(0.28)	99.69(0.47)	99.79(0.28)
$Kappa$	0.99(0.03)	0.98(0.03)	0.97(0.05)	0.98(0.03)

convergence speed especially in the RVFL model.

Option 2 (Fig. 1g, 1h and 1i), shows an equivalent effect with respect to option 1 when MLP models are used, meanwhile in RBF-NN and RVFL-NN networks a speed up convergence effect for the minority class can be observed, however, this condition was not finally and completely observed in the RBF-NN .

Regarding option 3 (Fig. 1j, 1k and 1l), convergence speed improves when MLP and RVFL-NN were used. For the RBF-NN, expected better results were not observed, however, reported values were better than the ones observed when Option 0 is applied. In Fig. 1, it can be seen for the MLP and RVFL-NN the options 1, 2 and 3 significant convergence results, meanwhile in RBF-NN, only in options 1 and 2 satisfactory values were obtained.

In general terms, options 1, 2 and 3 help to obtain faster convergence speeds in ANN when these were trained using imbalanced databases with the backpropagation algorithm.

On the other hand, and considering classification purposes, $g - mean$ and $Kappa$ coefficient values near to 100% can be obtained in all options (see Table 2), in other words, this values were not increased nor decreased when using options 1, 2 and 3 [2].

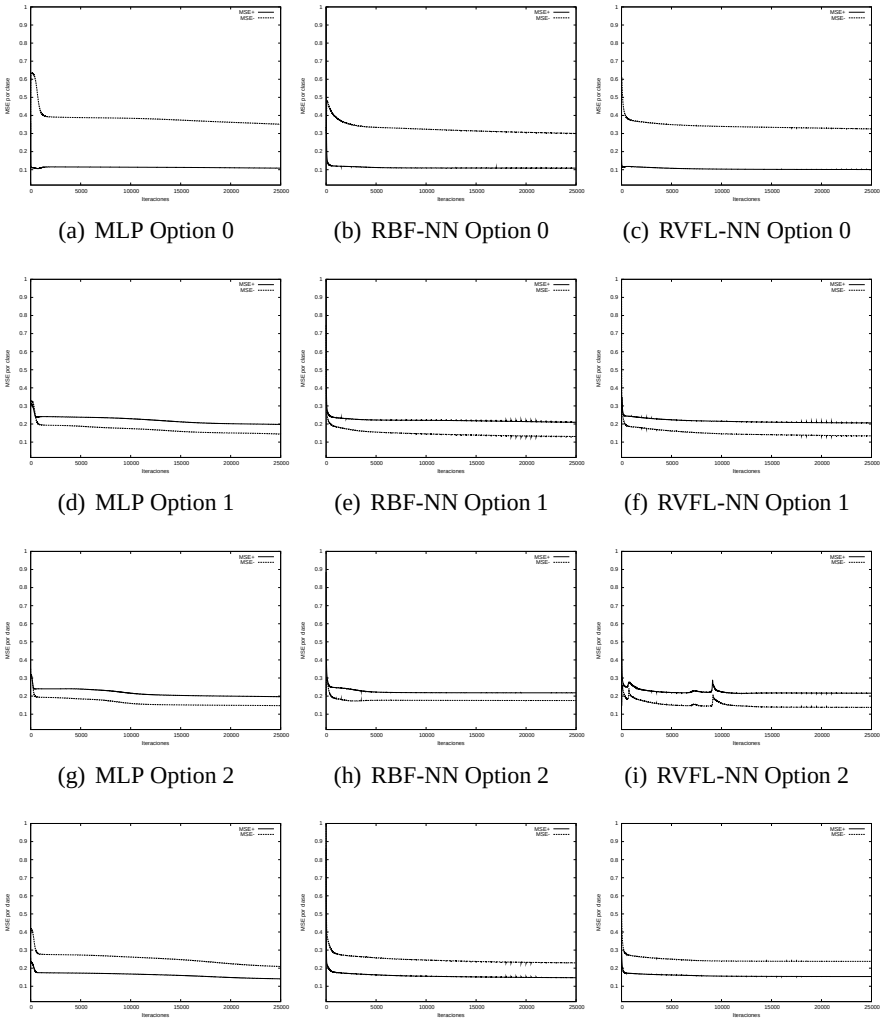
In Fig. 2, the MSE performance class value is presented for the Phoneme database. A different performance can be observed with respect to the V2Cls database. It can be seen in Fig. 2a, 2b and 2c, that the MSE value for the minority class is greater than the MSE value for the majority class during the training phase and also convergence is not obtained for the ANN in neither of the three models. When options 1, 2 and 3 were used the percentage error for the minority class was considerably decreases. However, convergence for the ANN was not observed. It is interesting to note that there is no significant difference between the values obtained when option 1, 2 and 3 were applied. From other point view, when the classifier performance is evaluated using the following criteria values: accuracy, $g-mean$ value and $kappa$ coefficient, important results were obtained.

In table 3, for the MLP models it can be appreciated in terms of accuracy that there was no statistic difference (except option 3), this means that accuracy was not affected for the ANN due to the applied options. As regard to the $g - mean$, high $g - mean$ values they denote a good classifier performance in both classes (minority class and majority class), in other words, the successful classified sample percentage value were increased in the minority class when options 1, 2 and 3 were applied.

The RBF-NN models reported a similar performance when there are not significant statistical accuracy differences. The $g-mean$ values were increased (with statistical differences) and the classification confidence value was increased for options 2 and 3, meanwhile in option 1 the $kappa$ coefficient value stays the same.

In the RVFL-NN model, there were no statistical significant differences when precision values were compared (except for option 1), however, $g - mean$ values and confidence values in the classification phase were significantly increased. In general terms, when options 1, 2 and 3 are used, an increment of the

[2] In order to validate these results a K-Fold Cross-Validation a Paired t statistic was used.



(a) MLP Option 0 (b) RBF-NN Option 0 (c) RVFL-NN Option 0

(d) MLP Option 1 (e) RBF-NN Option 1 (f) RVFL-NN Option 1

(g) MLP Option 2 (h) RBF-NN Option 2 (i) RVFL-NN Option 2

(j) MLP Option 3 (k) RBF-NN Option 3 (l) RVFL-NN Option 3

Figure 2: MSE by class for Phoneme dataset

classifier performance and convergence speed for the minority class were observed as in early results.

In Fig. 3, for database B2Cls different results can be observed. In Fig. 3a, 3b and 3c, the MSE value for the minority class is increased but decreased slowly when compared with the databases reported earlier. This situation is a consequence of the fact that the minority class was ignored during the training phase and the successfully sample classification number for this class is zero (See Table 4). This result is expected because of the considerable data complexity value of this database (see Table 1).

In the MLP models, better results were obtained when options 1, 2 and 3 were applied during the training phase. In Fig. 3d, 3g and 3j the MSE values for the minority class were reduced, which provides, as a consequence, a better classifier performance.

With respect to accuracy values there is no significant statistical difference between the reported values when options 0, 1, 2, and 3 were used, meanwhile important incremented values were obtained for $g - mean$ values. Notice that when option 0 was applied $g - mean$ reported values were zero, in neither case there was no correct identification.

However, when using options 1, 2 and 3 accuracy percentage values for this class were increased in at least 90%. Classification confidence value was also increased in at least 55%. The RBF-NN and RVFL-NN models reported similar results. Similar values occurred in other cases with options 1, 2 and 3. The convergence speed was increased for the less representative class. However, results were not as good as the ones reported for the MLP model.

In table 4, can be observed that for the RBF-NN and RVFL-NN models the accuracy percentage for the minority class was increased when option 1 and

Table 3: Classification performance of Phoneme dataset

MLP	Option 0	Option 1	Option 2	Option 3
Acc	80.01(1.41)	80.33(1.68)	80.87(1.87)	81.35(1.84)
$gmean$	74.58(2.2)	82.16(1.6)	82.59(1.87)	80.62(2.17)
$Kappa$	0.51(0.04)	0.58(0.03)	0.59(0.04)	0.58(0.04)
RBF-NN	Option 0	Option 1	Option 2	Option 3
Acc	79.5(1.26)	77.07(2.27)	78.81(1.17)	79.55(1.46)
$gmean$	75.21(1.88)	80.19(2.01)	81.62(1.5)	79.81(1.45)
$Kappa$	0.51(0.03)	0.53(0.04)	0.56(0.03)	0.55(0.03)
RVFL-NN	Option 0	Option 1	Option 2	Option 3
Acc	80.27(1.53)	78.94(2.24)	79.22(1.97)	80.18(1.46)
$gmean$	75.59(2.07)	81.45(1.7)	81.82(1.89)	80.02(1.22)
$Kappa$	0.52(0.04)	0.56(0.04)	0.56(0.04)	0.56(0.03)

2 were applied and important accuracy reduction was reported. From another point view, for option 3, it seems that it does not help to increase the classifier performance.

4 Conclusion

Reported results help to understand the different possibilities that may appear when ANN are trained with the back-propagation algorithm with batch processing and two imbalanced classes databases. Evidences show that using RBF-NN and RVFL-NN models, sensitivity is increased with respect to the induced noise in the TS. Besides, benefits are reported when cost functions are applied in order to avoid that the less representative class will be ignored during the training phase (specially when the MLP is used).

Acknowledgements: This work has been partially supported by grants DPI2006-15542-C04-03 from the Spanish CICYT , SEP-2003-C02-44225 from the Mexican CONACyT, and Generalitat Valenciana under the project GV/2007/105.

References:

[1] R. Barandela, R.M. Valdovinos, J.S. Sánchez, and F.J. Ferri. The imbalanced training sample problem: Under or over sampling? In *SSPR/SPR*, page 806, 2004.

[2] N. Japkowicz and S. Stephen. The class imbalance problem: a systematic study. *Intelligent Data Analysis*, 6:429–449, 2002.

[3] T. Fawcett and F. Provost. Adaptive fraud detection. *Data Min. Knowl. Discov.*, 1(3):291–316, 1997.

[4] Y. Murphey, H. Guo, and L. Feldkamp. Neural learning from unbalanced data. *Applied Intelligence.*, 21:117–128, 2004.

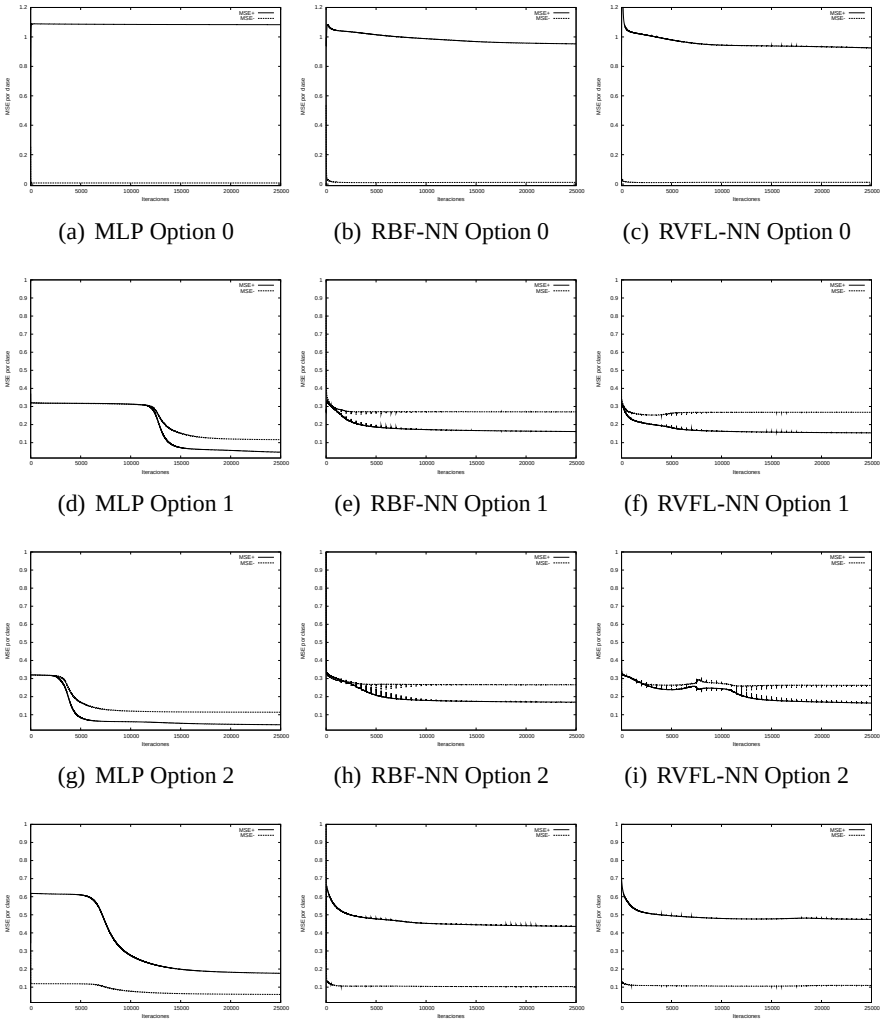
[5] P.K. Chan, W. Fan, A.L. Prodromidis, and S.J. Stolfo. Distributed data mining in credit card fraud detection. *IEEE Intelligent Systems*, 14(6):67–74, 1999.

[6] Z.-H. Zhou and X.-Y. Liu. Training cost-sensitive neural networks with methods addressing the class imbalance problem. *IEEE Transactions on Knowledge and Data Engineering.*, 18:63–77, 2006.

[7] L. Bruzzone and S.B. Serpico. Classification of imbalanced remote-sensing data by neural networks. *Pattern Recognition Letters*, 18:1323–1328, 1997.

[8] R. Anand, K.G. Mehrotra, C.K. Mohan, and S. Ranka. An improved algorithm for neural network classification of imbalanced training sets. *IEEE Transactions on Neural Networks*, 4:962–969, 1993.

[9] J.S. Sánchez, R. Mollineda, and J.M. Sotoca. An analysis of how training data complexity affects the nearest neighbor classifiers. *Pattern Analysis and Applications*, 10(3):189–201, 2007.



(a) MLP Option 0 (b) RBF-NN Option 0 (c) RVFL-NN Option 0

(d) MLP Option 1 (e) RBF-NN Option 1 (f) RVFL-NN Option 1

(g) MLP Option 2 (h) RBF-NN Option 2 (i) RVFL-NN Option 2

(j) MLP Option 3 (k) RBF-NN Option 3 (l) RVFL-NN Option 3

Figure 3: MSE by class for B2Cls dataset

Table 4: Classification performance of B2Cls dataset

MLP	Option 0	Option 1	Option 2	Option 3
Acc	92.16(0.36)	90.88(1.66)	91.36(1.99)	92.8(2.33)
$gmean$	0.0(0.0)	91.27(5.71)	91.54(5.79)	92.33(6.13)
$Kappa$	0.0(0.0)	0.57(0.07)	0.58(0.08)	0.63(0.1)

RBF-NN	Option 0	Option 1	Option 2	Option 3
Acc	91.36(1.54)	69.28(4.58)	65.28(4.26)	86.4(6.22)
$gmean$	0.0(0.0)	72.62(9.74)	71.01(4.2)	18.47(27.3)
$Kappa$	0.0(0.0)	0.18(0.05)	0.15(0.02)	0.14(0.1)

RVFL-NN	Option 0	Option 1	Option 2	Option 3
Acc	92.0(0.57)	64.64(9.97)	66.56(4.61)	86.08(6.46)
$gmean$	0.0(0.0)	65.98(10.04)	74.2(5.03)	18.91(17.26)
$Kappa$	0.0(0.0)	0.13(0.05)	0.18(0.03)	0.1(0.06)