Breast Cancer Microarray Dataset with the Decision Tree Classifier and Efficient Scaling Techniques

Maha A. Hana
Information Systems Dept.
Faculty of Computers & Information,
Helwan University, Egypt

Elsayed Badr
Scientific Computing
Faculty of Computers & AI,
Benha University, Egypt

Sally Gamal
Information Systems Dept.
Faculty of Computers & Information,
Helwan University, Egypt

Naglaa Shehata
Instructor at Helwan University, Helwan Egypt

ABSTRACT
Badr et al. [1] proposed efficient scaling techniques EST with support vector machine on the data set Wisconsin from UCI machine learning with a total 569 rows and 33 columns. In this work, we try to evaluate the validity of the results reached by Badr et al. [1] in the case of using different datasets, different classifiers and dimensionality reduction tools? So, the decision tree algorithm is applied on the used breast cancer microarray dataset (BCMD) contains 289 patients and 35981 attributes**.** We use principal components analysis (PCA) to reduce the number of attributes. We also propose new scaling techniques to improve the accuracy of the decision tree algorithm. Experimental results show that the decision tree algorithm with new scaling techniques (equilibration, geometric mean and arithmetic mean) achieves 84.98 %, 80.65 % and 79.96 % accuracy against to the traditional normalization (normalization [0, 1], normalization [-1, 1] and standard normalization) by 75.44 %, 76.85% and 78.93%.

General Terms
Data Mining, Classification

Keywords
Machine Learning, Breast Cancer, Decision Tree, scaling techniques

1. INTRODUCTION
Improving the accuracy of identifying the breast cancer disease is very important task. Breast cancer disease is the second most common type of cancer after lung cancer. Breast cancer is the most widespread by 12.3% of all cancer for males and females of all ages. It is the most spreading in women worldwide, accounting 25.4% of the whole cases diagnosed in 2018 [2]. Defects in breast cancer diagnosis by experts can be avoided by expert systems and artificial intelligent techniques. These expert systems can examine the medical data in shorter time and help junior physicians.

Tomlin [3] performed a computational study comparing arithmetic mean, geometric mean, equilibration, Curtis and Reid scaling technique [4], Fulkerson and Wolfe scaling technique [5], and various combinations on six test problems. The conclusion of Tomlin's comparative study was that geometric mean scaling method, optionally followed by equilibration or Curtis and Reid scaling technique are the best combined scaling techniques.

The scaling techniques can improve the accuracy of classifiers. Elsayed Badr et al. [1] proposed ten efficicent scaling techniques for optimizing SVM. These scalling techniques are efficient for linear programming approach [12-20]. The scalling techniques that they applied with SVM on WDBC dataset are arithmetic mean, de Buchet for three cases (p=1, 2), equilibration, geometric mean, IBM MPSX, *Lp*-norm for three cases (p=1 or 2). They were the first to use EST for metaheuristic approach. There are many inquiries about using EST with other classifiers, other datasets and other dimensionality reduction tools such as principal components analysis (PCA).

In this paper, we try to answer the following question: What if we use a different dataset, different classifier and dimensionality reduction tool with EST? Practically, the decision tree algorithm is applied on the used breast cancer microarray dataset (BCMD) [6] contains 289 patients and 35981 attributes**.** We use principal components analysis (PCA) to reduce the number of attributes. We also propose new scaling techniques to improve the accuracy of the decision tree algorithm. Experimental results show that the decision tree algorithm with new scaling techniques (equilibration, geometric mean and arithmetic mean) achieves 84.98 %, 80.65 % and 79.96 % accuracy against to the traditional normalization (normalization [0, 1], normalization [-1, 1] and standard normalization) by 75.44 %, 76.85% and 78.93%.

For more details about scaling techniques, the reader can refer to [8-14]. On the other hand, for more details about the linear programming, the reader is referred to [15-21].

The rest of this paper is organized as follows. The algorithm that is used in the study: Decision Tree is described in Section 2. The proposed model that uses te grid search technique is introduced in section 3. In Section 4, detailed descriptions of new scaling techniques, arithmetic mean, equilibration, geometric mean are proposed. Experimental design which has data description, experimental setup, measure for performance evaluation and a comparative study are introduced in section 5. In Section 6 the main results and discussion are proposed. Finally, conclusions and future works are introduced in section 7.

2. PRELIMINARIES: Decision Tree
Decision tree [7] is a classifier that is expressed as a recursive partition of the instance space. It creates a predictive model, which maps observations about a node to conclusions about the nodes' target value. In a tree structure leaves represent the class labels and branches represent conjunctions of feature leading to the class labels. Figure 1 shows the illustrated example of binary decision tree.

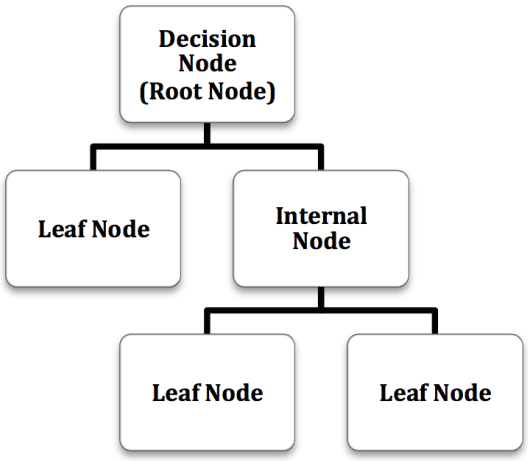

Figure 1. Illustrated example of binary decision tree

Decision tree provides a powerful technique for classification and prediction in Breast Cancer diagnosis problem. Various decision tree algorithms are available to classify the data, including ID3, C4.5, C5, J48, CART and CHAID. In this paper we have chosen CART decision tree algorithm [7] to establish the model.

3. THE PROPOSED CLASSIFICATION MODEL

Many times while working on a dataset and using a Machine Learning model we don't know which set of hyper parameters will give us the best result. Passing all sets of hyper parameters manually through the model and checking the result might be a hectic work and may not be possible to do. To get the best set of hyper parameters we can use Grid Search. Grid Search passes all combinations of hyper parameters one by one into the model and checks the result. Finally it gives us the set of hyper parameters which gives the best result after passing in the model. A grid search method must be guided by some performance metric, typically measured by cross-validation on the training set [21] or evaluation on a held-out validation set [22]. We use the grid search to determine the entropy or giniIndex metrics for the decision tree CART.

4. SCALING TECHNIQUES

Here, we introduce the mathematical notations of ten scaling techniques in addition to the normalization scaling techniques with ranges [0, 1] and [-1, 1]. First of all, we introduce the following mathematical preliminaries as shown in Table 1.

The scaled matrix is expressed as *RAS*, such that $R = diag\ (r_1, \ldots, r_m)$ and $S = diag\ (s_1, \ldots, s_n)$. All scaling techniques proposed in this section apply first rows scaling and after that columns scaling. Then, the matrix after full scaling (row and column) is given by:

$$A^R = RA;\ A^{RS} = A^R S \tag{1}$$

Table 1. Mathematical preliminaries for scaling techniques

Symbol	Description
$A\ (a_{ij})$:	m x n matrix (with m (observations) and n (attributes)).
r_i:	The scaling agent of row i
s_j:	The scaling agent of column j
R:	Diagonal matrix such that $R = diag\ (r_1, \ldots, r_m)$
S:	Diagonal matrix such that $S = diag\ (s_1, \ldots, sn)$
N_i:	$N_i = \{j : A_{ij} \neq 0\}$, such that $1 \leq i \leq m$
M_j:	$M_j = \{i : A_{ij} \neq 0\}$ such that $1 \leq j \leq n$
n_i:	The number of elements for the set N_i
m_j:	The number of elements for the set M_j
$A^R\ (a_{ij}^R)$	The scaled matrix by row R scaling agent.
$A^{RS}(a_{ij}^{RS})$	The final scaled matrix.

1) Arithmetic scaling technique [11]: First, Equation (2) represents the rows scaling such that each row (instance) is divided by the arithmetic mean of the absolute value of the non-zero elements in that row (instance).

$$r_i = \frac{n_i}{\sum_{j \in N_i} |a_{ij}|};\ a_{ij} \neq 0 \tag{2}$$

Second, Equation (3) represents the columns scaling such that each column (attribute) is divided by the arithmetic mean of the absolute value of the non-zero elements in that column (attribute).

$$s_j = \frac{m_j}{\sum_{i \in M_j} |a_{ij}^R|\,|a_{ij}^R|};\ a_{ij}^R \neq 0 \tag{3}$$

2) Equilibration scaling technique [11]: The largest element in absolute value is the corner stone for this scaling method. Each row of the matrix A is divided by the largest element in absolute value in that row. Then, each column of the scaled matrix A by the row factor divided by the largest element in absolute value in that column. The range of the final scaled matrix A is [-1, 1].

3) Geometric mean scaling technique [11]: First, Equation (4) represents the rows scaling such that each row (instance) is divided by the geometric mean of the absolute value of the non-zero elements in that row (instance).

$$r_i = (\max_{j \in N_i} |a_{ij}|\ \min_{j \in N_i} |a_{ij}|)^{-1/2} \tag{4}$$

Second, Equation (5) represents the columns scaling such that each column (attribute) is divided by the geometric mean of the absolute value of the non-zero elements in that column (attribute).

$$s_j = (\max_{j \in M_j} |a_{ij}^R|\ \min_{j \in M_j} |a_{ij}^R|)^{-1/2} \tag{5}$$

4) Normalization scaling technique [-1, 1] [21]: Equation (6) is used for normalization scaling method with range [-1, 1] such that a, a', max_k and min_k are the original value, the scaled value, the maximum value and the minimum value of feature k respectively.

$$a' = 2\left(\frac{a - min_k}{max_k - min_k}\right) - 1 \tag{6}$$

Normalization scaling method avoids the numerical difficulties during the calculation.

5) Normalization scaling technique [0, 1] [21]: Another normalization scaling technique is formulated from the updated equation (6) as follows:

$$a' = \frac{a - min_k}{max_k - min_k} \quad (7)$$

6) **Standardization scaling technique**: Standardization is another scaling technique where the values are centered around the mean with a unit standard deviation. This means that the mean of the attribute becomes zero and the resultant distribution has a unit standard deviation. Here's the formula for standardization:

$$a' = \frac{a - \mu}{\sigma} \quad (8)$$

μ is the mean of the feature values and σ is the standard deviation of the feature values.

5. EXPERIMENTAL DESIGN

In this section, we introduce data description, measure for performance evaluation and the comparative study.

5.1 Data description

In this work, we have run the proposed model on the breast cancer microarray dataset (BCMD) contains 289 patients and 35981 attributes [6]. This dataset is taken from the structural bioinformatics and computational biology lab (SBCBLab). SBCBLab has a solid history of research in Bioinformatics, with several publications in the area. The group has vast knowledge in Artificial Intelligence, Machine Learning, Metaheuristic, and Massively Parallel Processing.

5.2. Experimental setup

The proposed model was developed by Python. CART, implementation was enhanced, which is originally developed by Chang and Lin [24]. Table 3 describes the experiments computing environment.

Table 2. Description of the computing environment

CPU	Intel(R) Xeon(R) CPU @ 2.30GHz No. CPU Cores: 2
RAM Size	13 GB RAM
Python version	Python 3.7.10

Salzberg [25] introduced the k-fold CV which is used to guarantee the valid results. In this paper, $k = 10$.

5.3. Measure for performance evaluation

In order to test the performance of the proposed model, we use accuracy. According to the confusion matrix, accuracy is defined as follows:

$$Acc = (TruPos + TruNeg) / [TruPos + FlsPos + TruNeg + FlsNeg] \times 100\% \quad (9)$$

Where: Acc: Accuracy; TruPos: true positive; TruNeg: true negative; FlsPos: false positive qnd FlsNeg,: false negative.

6. EXPERIMENTAL RESULTS AND DISCUSSIONS

Table 3 shows a comparison among classification accuracies of decision tree with normalization scaling [0, 1], normalization scaling [-1, 1] and without scaling. It is apparent from these tables that the average accuracy rates achieved by decision tree CART with normalization scaling [0, 1] (75.44%), normalization scaling [-1, 1] (76.85%) are better than that obtained by CART with without-scaling technique (76.86%).

On the other hand, the average CPU Time rates achieved by decision tree CART with normalization scaling [0, 1] (0.1829) which is less than CPU Time obtained by CART with without-scaling technique.

Table 3: Accuracy for WBCD database using SVM with C and γ which were calculated by grid search technique

(Without scaling and Normalization scaling [0,1]

	Without (S0)	Normalization [0, 1] (S1)	Normalization [-1, 1] (S2)
Fold	**Accuracy %**	**Accuracy %**	**Accuracy %**
1	79.31	62.07	65.52
2	93.10	79.31	86.21
3	72.41	75.86	72.41
4	72.41	72.41	75.86
5	62.07	72.41	75.86
6	68.97	68.97	68.97
7	79.31	82.76	75.86
8	72.41	75.86	75.86
9	79.31	86.21	86.21
10	89.29	78.57	85.71
Average	76.86	75.44	76.85
Criterion	Gini	Gini	Gini
CPU Time (s)	0.2252	0.1829	0.2095

Table 4 and Table 5 show a comparison among classification accuracies of decision tree with standardization, equilibration, arithmetic mean and geometric mean scaling techniques. It is apparent from these tables that the average accuracy rates achieved by decision tree CART with standardization (78.93%), equilibration (84.79%), arithmetic mean (79.96%) and geometric mean (80.65%) are better than that obtained by CART with traditional scaling technique.

Table 4: Accuracy and CPU Time for WBCD database using SVM with C and γ which were calculated by grid search technique

	Standard (S3)	Equilibration (S4)
Fold	**Accuracy %**	**Accuracy %**
1	79.31	89.66
2	86.21	82.76
3	82.76	96.55
4	65.52	79.31
5	82.76	86.21
6	72.41	79.31
7	79.31	82.76
8	68.97	75.86
9	82.76	86.21
10	89.29	89.29
Average	78.93	84.79
Criterion	Entropy	Entropy
CPU Time (s)	0.2373	0.2119

Table 5: Accuracy and CPU Time for WBCD database using SVM with C and which were calculated by grid search technique

	Arithmetic (S5)	Geometric (S6)
Fold	**Accuracy %**	**Accuracy %**
1	72.41	75.86
2	96.55	82.76
3	79.31	89.66
4	75.86	75.86
5	72.41	75.86
6	65.52	65.52
7	82.76	82.76
8	79.31	82.76
9	86.21	86.21
10	89.29	89.29
Average	79.96	80.65
Criterion	Entropy	Gini
CPU Time (s)	0.1786	0.1867

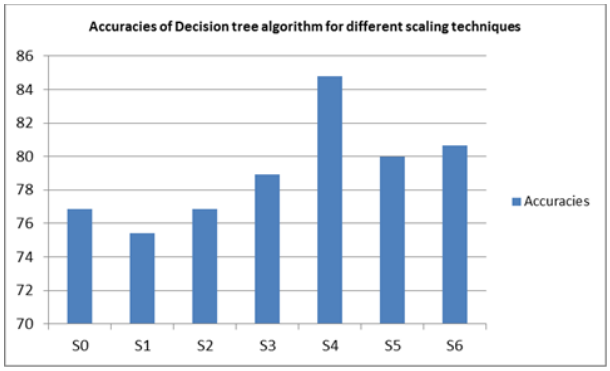

Figure 2: Accuracies of decision tree algorithm for efficient scaling techniques (normalization [0,1], standardization, normalization [-1,1], equilibration, geometric mean, arithmetic mean and without scaling)

Table 6: Accuracies and CPU Time of decision tree algorithm for efficient scaling techniques

Symbols	Scaling Technique	Accuracy (%)	CPU Time (s)
S0	Without scaling	76.86	0.2252
S1	Normalization [0, 1]	75.44	0.1829
S2	Normalization [-1, 1]	76.85	0.2095
S3	Standard scaling	78.93	0.2373
S4	Equilibration scaling	84.79	0.2119
S5	Arithmetic mean	79.96	0.1786
S6	Geometric mean	80.65	0.1867

From Table 6 and Figure 2, it is clear that equilibration scaling technique overcomes other scaling technique. This results match with Badr et al.'s results [1].

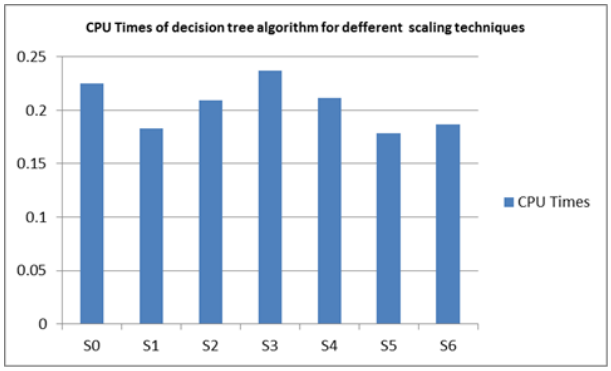

Figure 3: CPU Time of decision tree algorithm for efficient scaling techniques (normalization [0,1], standardization, normalization [-1,1], equilibration, geometric mean, arithmetic mean and without scaling)

By comparing the results presented in this work with the results of Badr et al.[1], we find that they are correspondent. Hence, we can say that the equilibration scaling technique is better for different classifiers with different data sets. On the other hand, we cannot be certain that the equilibration scaling technique is better at all, because that requires more practical experiments on more than one data set and also a different set of classifiers.

7. CONCLUSION AND FUTURE WORK

In this work, the decision tree algorithm is applied on the used breast cancer microarray dataset (BCMD) contains 289 patients and 35981 attributes**.** We use principal components analysis (PCA) to reduce the number of attributes. We also propose new scaling techniques to improve the accuracy of the decision tree algorithm. Experimental results show that the decision tree algorithm with new scaling techniques (equilibration, geometric mean and arithmetic mean) achieves 84.98 %, 80.65 % and 79.96 % accuracy against to the traditional normalization (normalization [0, 1], normalization [-1, 1] and standard normalization) by 75.44 %, 76.85% and 78.93%. In future work, the varying models and different datasets are applied with the efficient scaling techniques.

8. REFERENCES

[1] Elsayed Badr, Mustafa Abdul Salam, Sultan Almotairi, Hagar Ahmed, "From Linear Programming Approach to Metaheuristic Approach: Scaling Techniques", Complexity, vol. 2021, Article ID 9384318, 10 pages, 2021. https://doi.org/10.1155/2021/9384318

[2] Bray F, Ferlay J, Soerjomataram I, Siegel RL, Torre LA, Jemal A, "Global Cancer Statistics 2018," GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA Cancer J Clin, in press.

[3] Tomlin, J. A. 1975. On scaling linear programming problems. *Mathematical Programming Studies* 4, 146-166. DOI= http://dx.doi.org/10.1007/BFb0120718.

[4] Curtis, A. R. and Reid, J. K. 1972. On the automatic scaling of matrices for Gaussian elimination. *IMA Journal of Applied Mathematics* 10, 1, 118-124. DOI= http://dx.doi.org/10.1093/imamat/10.1.118

[5] Fulkerson, D. R. and Wolfe, P. 1962. An algorithm for scaling matrices. *SIAM Review* 4, 2, 142-146. DOI= http://dx.doi.org/10.1137/1004032.

[6] http://sbcb.inf.ufrgs.br/cumida (accessed May 01, 2021).

[7] J. Han and M. Kamber, "Data Mining Concepts and Techniques", Morgan Kauffman Publishers, 2000.

[8] Larsson, T. 1993. On scaling linear programs-Some experimental results. *Optimization* 27, 4, 335-373. DOI= http://dx.doi.org/10.1080/02331939308843895

[9] de Buchet, J. 1966. Experiments and statistical data on the solving of large-scale linear programs. In *Proceedings of the Fourth International Conference on Operational Research*, Hertz, D. A. and Melese, J., Eds. Wiley-Interscience, New York, 3-13.

[10] Elble, J. M. and Sahinidis, N. V. 2012. Scaling linear optimization problems prior to application of the simplex method. *Computational Optimization and Applications* 52, 2, 345-371. DOI= http://dx.doi.org/10.1007/s10589-011-9420-4

[11] Benichou, M., Gauthier, J. M., Hentges, G., and Ribiere, G. 1977. The efficient solution of large-scale linear programming problems-Some algorithmic techniques and computational results. *Mathematical Programming* 13, 1, 280-322. DOI= http://dx.doi.org/10.1007/BF01584344

[12] Ploskas, N. and Samaras N. 2013. A Computational Comparison of Scaling Techniques for Linear Optimization Problems on a GPU. *Optimization Methods and Software*. Paper under review.

[13] Triantafyllidis, C. and Samaras, N. "Three nearly scaling-invariant versions of an exterior point algorithm for linear programming", *Optimization.* **2014**, *64*(10), 2163–2181.

[14] Ploskas, N. and Samaras, N. "A computational comparison of scaling techniques for linear optimization problems on a graphical processing unit", *International Journal of Computer Mathematics.* **2015**, *92*(2), 319–336.

[15] E. M. Badr and H. elgendy (2020) "A Hybrid water cycle - particle swarm optimization for solving the fuzzy underground water confined steady flow" Indonesian Journal of Electrical Engineering and Computer Science Vol 19, No1: 2020

[16] Elsayed M. Badr, Mahmoud I. Moussa in *Wireless Networks (*2019*)*, An upper bound of radio *k*-coloring problem and its integer linear programming model, First Online: 18 March 2019.

[17] Badr, E.;Aloufi,K.A Robot's Response Acceleration Using the Metric Dimension Problem. *Preprints* 2019, 2019110194 (doi:10.20944/preprints201911.0194.v1).

[18] E.S. Badr, K. Paparrizos, Baloukas Thanasis and G. Varkas (2006), Some computational results on the efficiency of an exterior point algorithm, in Proc. of the 18th National Conference of Hellenic Operational Research Society (HELORS), 15-17 June, Rio, Greece, pp. 1103-1115

[19] E. S. Badr, K. Paparrizos, N. Samaras, and A. Sifaleras (2005), On the Basis Inverse of the Exterior Point Simplex Algorithm, in Proc. of the 17th National Conference of Hellenic Operational Research Society (HELORS), 16-18 June, Rio, Greece, pp. 677-687.

[20] E.S. Badr, M. Moussa, K. Paparrizos, N. Samaras, and A. Sifaleras, Some computational results on MPI parallel implementation of dense simplex method, World Academy of Science, Engineering and Technology (WASET), 23, 2008,778–781.

[21] E. M. Badr and Sultan Almotiari (2019) " On a Dual Direct Cosine Simplex Type Algorithm and Its Computational Behavior" Mathematical Problems in Engineering Volume 2020, Article ID 7361092, 8 pages. https://doi.org/10.1155/2020/7361092

[22] Chin-Wei Hsu, Chih-Chung Chang and Chih-Jen Lin (2010). A practical guide to support vector classification. Technical Report, National Taiwan University.

[23] Chicco D (December 2017). "Ten quick tips for machine learning in computational biology". BioData Mining. 10 (35): 35. doi:10.1186/s13040-017-0155-3. PMC 5721660. PMID 29234465.

[24] Vapnik, V.N. "The nature of statistical learning theory", Springer: New York, 1995.

[25] Chang, C.C. and C.J. Lin, LIBSVM: a library for support vector machines. 2001, Software available at http://www.csie.ntu.edu.tw/cjlin/libsvm

[26] Salzberg, S. L., On comparing classifiers: Pitfalls to avoid and a recommended approach. Data Min Knowl Discov 1(3):317–328, 1997.

[27] EM Badr, MA Salam, M Ali, H Ahmed, Social Media Sentiment Analysis using Machine Learning and Optimization Techniques, International Journal of Computer Applications (0975 – 8887) Volume 178 – No. 41, August 2019.

[28] Elsayed Badr, Mustafa Abdulsalam and Hagar Ahmed. "The impact of scaling on Support Vector Machine in Breast Cancer Diagnosis". International Journal of Computer Applications 175(19):15-19, September 2020