Classification performance of Extreme Learning Machine Radial Basis Function with k-means, k-medoids and mean shift clustering algorithms

Aleksandra Konopka[1 [0000−0003−1730−5866]],
Karol Struniawski[1 [0000−0002−4574−2986]] and
Ryszard Kozera[1,2 [0000−0002−2907−8632]]

[1] Institute of Information Technology, Warsaw University of Life Sciences - SGGW,
ul. Nowoursynowska 159, 02-776 Warsaw, Poland
{aleksandra_konopka, karol_struniawski, ryszard_kozera}@sggw.edu.pl
[2] School of Physics, Mathematics and Computing,
The University of Western Australia,
35 Stirling Highway, Crawley, WA 6009, Perth, Australia
ryszard.kozera@uwa.edu.au

Abstract. Extreme Learning Machine (ELM) is a feed-forward neural network with one hidden layer. In its modification called ELM Radial Basis Function the input data is a priori clustered into a number of sets represented by their centroids. The matrix of distances between each sample and centroid is calculated and applied as input data to the neural network. This work conducts a comparison study of the ELM Radial Basis Function classification performance upon applying either k-means, k-medoids or mean shift clustering methods. Generated results are obtained from two datasets i.e. Wine Quality-White and Ionosphere. The computations are based on full datasets or on the same both sets reduced by a feature selection algorithm. The parameters of the classifiers such as number of neurons in hidden layer, the value of k in k-means and k-medoids, the value of radius in mean shift are optimized through an iterative procedure upon maximizing an accuracy or minimizing Mean Square Error and computation time. The different distance metrics for k-means and k-medoids, and mean shift with Gaussian or flat kernel function are also compared. The results obtained with Softplus and linear activation function (applied in most of the computations in this work) are juxtaposed with the results generated by other activation functions.

Keywords: Neural Networks · Machine Learning · Extreme Learning Machine · Radial Basis Function · Clustering Algorithms

1 Introduction

The Backpropagation Algorithm (BA) introduced in 1986 by Rumelhart et al. [21] represents an important component of machine learning. The main problem of BA stems from the fact that it usually yields local minima of associated network's residual error function. In addition, BA computational cost and training

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

time, especially for the large datasets, may preclude its practical application. The new concept of neural network called Extreme Learning Machine (ELM) was introduced by Huang et al. in 2004 [7]. ELM converges much faster than traditional learning schemes as relieved from the time-consuming iterations and is also more likely to reach a global optimum [10]. ELM is successfully adapted to various machine learning applications such as classification and regression in medicine, chemistry, transportation, economy, agriculture, robotics etc. It also outperforms other methods in training time and approximation ability [9, 18]. ELM is characterized by yielding extremely fast training time in comparison to other machine learning methods like e.g. Multilayer Perceptron trained with BA [11]. Currently, ELM still evolves to further improve generalization capacity in case of special applications. One of its variants called Extreme Learning Machine Radial Basis Function (ELM-RBF), weaving core principles of ELM with feature space mapping using RBF kernels, yields comparable results to BA with considerably faster computation time [2]. In the field of ELM-RBF, the most commonly used clustering method is k-means, although application of k-medoids is also found in the literature.

In this paper, the performance of ELM-RBF combined with k-means, k-medoids or mean shift clustering methods is thoroughly investigated. The comparison involves manipulating with multiple variables, such as parameter values of clustering methods, number of neurons or different activation functions in ELM-RBF. In order to obtain the most significant results a comparison is conducted on two datasets - Wine Quality-White and Ionosphere [3]. The characteristics of the selected benchmark sets allow to obtain significant results and compare different algorithms, such as those used for solving k-medoids problem, as their application relies on the size of the input dataset.

2 Extreme Learning Machine

Extreme Learning Machine consists of input, hidden and output layer aimed to solve classification and regression tasks by supervised learning [7]. Assume input data is described as pairs of values $\eta = \{(x_i, t_i)\}_{i=1}^{N}$, where $x_i = \{x_{ij}\}_{j=1}^{d}$ forms matrix $X_{d\times N}$ of d features. Here t_i is recognized as affiliation to the given class establishing T. For the classification need $t_i \in [0, M] \subseteq \mathbb{N}$, whereas for regression the target value $t_i \in \mathbb{R}^M$. Here M represents either the number of classes or dimension of target values. The number of neurons in input, output and hidden layer is assumed here to be equal to d, M and L, respectively. Here L is given a priori as there is no universal method for L optimization. Neurons in ELM are McCulloch-Pitts neurons [14] with identity function on input and output layers and any activation function $f : \mathbb{R} \to \mathbb{R}$ on hidden layer units. Matrix of weights $W_{d\times L}$ with coefficients $w_{ij} \in (-1, 1)$ connecting input neurons with L hidden layer neurons and bias values b_i of $b = \{b_i\}_{i=1}^{N}$ are randomly selected with the aid of uniform distribution function, where $i = 1, \ldots, d$ and $j = 1, \ldots, L$. Thus, $H = f(X^T W + b)$ is obtained as an output of the hidden layer. Wages β between hidden and output layer are computed once using algebraic

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

transformations. To calculate β the following matrix equation $H\beta = T$ should be solved. The matrix H is non-invertible and therefore solving $H\beta = T$ can be reformulated into the optimization task estimating $\hat{\beta}$ based on minimization of a Mean Square Error (MSE) between H and T [7]. The corresponding solution reads as $\hat{\beta} = H^{\dagger}T$, where $H^{\dagger}$ is the Moore-Penrose pseudo-inverse operation [20]. Various methods for $H^{\dagger}$ evaluation can be applied including e.g. Cholesky factorization of a singular matrix [12].

3 Extreme Learning Machine Radial Basis Function

Extreme Learning Machine Radial Basis Function is a method with training similar to its archetype which is based on random generation of W, b and calculation of β using generalized inverse of matrix H. The extra component involves input data transformation with the aid of Radial Basis Function [15]. Specifically, let $X_{d\times N}$ be a set of d features for N observations. First, a vector quantization technique is applied as a clustering algorithm. The aim of the latter is to partition N samples into a certain number (given a priori or designated automatically during algorithm's run) of k clusters using e.g. k-means clustering algorithm. Each sample x_i is assigned to exactly one cluster $\{c_j\}_{j=1}^{k}$, which in turn is recognized as the closest centroid to x_i in terms of a considered metric. Next, x_i is transformed to the new feature space based on a chosen kernel function. Note here that when $k > d$ then X is mapped to a higher dimension. The definition of the RBF kernel for ELM-RBF is given as $K(x_i, c_j) = exp\{\frac{-\|x_i - c_j\|^2}{2\sigma^2}\}$. The matrix $K_{N\times k}$ is computed as a measure of distance between each x_i and c_j. The σ_j value is determined as a $\sigma_j = \frac{max\{d_j\}}{\sqrt{2k}}$. Finally, K is treated as an input matrix to the typical ELM network.

4 Clustering methods

Clustering methods divide samples into disjoint groups. In k-means, k-medoids and mean shift each cluster is represented by a centroid which is calculated through an iterative procedure.

4.1 Mean Shift

Mean shift is an unsupervised learning algorithm [5] commonly applied in clustering, tracking and smoothing. This algorithm locates maxima of a density function with the aid of an iterative procedure upon updating candidates for centroids. Mean shift requires to specify the bandwidth of a window, which is shifted until the algorithm converges. The number of clusters is a priori unknown and depends on the density of input data samples. The points are assigned to the corresponding local maxima computed by the algorithm.

The input data of the algorithm is a set of n data points $\mathcal{Q} = \{q_i\}_{i=1}^{n}$, where $q_i = (q_{i_1}, q_{i_2}, \ldots, q_{i_d}) \in \mathbb{R}^d$. The value of bandwidth b is specified arbitrarily. Let $C = \{\{c_{ij}\}_{i=1}^{n}\}_{j=1}^{s_i}$ be the set of all locations of the window

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

shifted in the algorithm, where $c_{ij} = (c_{ij_1}, c_{ij_2}, \ldots, c_{ij_d})$, $i = 1, \ldots, n$ is a number of point that is currently considered and $j = 1, \ldots, s_i$ is a number of the shift for i'th point. Then first point $q_1 \in \mathcal{Q}$ is selected and it is set as the first location of window's center ($c_{11} = q_1$). Then Euclidean distances $d(c_{ij}, q_i) = \sqrt{(c_{ij_1} - q_{ij_1})^2 + \cdots + (c_{ij_d} - q_{ij_d})^2}$ between current c_{ij} and each q_i are computed. All q_i with $d(c_{ij}, q_i) \leq b$ (i.e. inside the window centered at c_{ij}) are selected to the set $\hat{\mathcal{Q}}_{ij} = \{\hat{q}_m\}_{m=1}^{w_{ij}}$. Subsequently, the new location of the window is calculated as a mean value of all $\hat{\mathcal{Q}}_{ij}$ for each of their d dimensions $c_{i(j+1)} = (\frac{1}{w_{ij}} \sum_{m=1}^{w_{ij}} \hat{q}_{m_1}, \ldots, \frac{1}{w_{ij}} \sum_{m=1}^{w_{ij}} \hat{q}_{m_d})$. Such procedure is repeated for all $\mathcal{Q}$ until they all converge to their corresponding local maxima. An output of this method is the set of points assigned to each of the generated disjoint clusters.

The above procedure iterates over each of the points from the dataset resulting in high computation time for large input data. Despite the fact that shifting a selected point does not influence other iterations and the process can be parallelized it is still very slow. The elapsing computation time is highly correlated with the size of input matrix of data making this approach inefficient. The mean shift algorithm can be optimized to render the results with less computation. The Bart Finkston's implementation (available on Mathworks [4]) presents an approach which highly reduces the computational complexity of the algorithm. The idea is to mark the points which were inside a shifting window (in any of s_i iterations) as visited to disregard them in upcoming iterations when new starting points c_{i1} are to be selected. We note how many times each of the points is positioned inside a bandwidth considering all s_i iterations for i'th starting point until q_i converges to c_{is_i}. The procedure is repeated selecting random points from $\mathcal{Q}$ not so-far visited. Subsequently, the matrix with votes for all generated centroids is used to attach $\mathcal{Q}$ to the appropriate clusters applying majority voting.

4.2 K-means

K-means is a common clustering method for which each of the samples is assigned to one of k disjoint clusters [19]. The number of k is selected arbitrarily. Each of the clusters is represented by a centroid which is equal to mean value of all observations within a group. The samples in a given iteration are assigned to the closest centroid in accordance with a distance metric i.e Euclidean distance.

Let a set of n data points $\mathcal{Q} = \{q_i\}_{i=1}^{n}$, where $q_i = (q_{i_1}, q_{i_2}, \ldots, q_{i_d}) \in \mathbb{R}^d$ be given. Assume $C = \{\{c_{lj}\}_{l=1}^{k}\}_{j=1}^{m}$ is the set of all locations of k centroids-to-be in the algorithm, where $c_{lj} = (c_{lj_1}, c_{lj_2}, \ldots, c_{lj_d})$, $l = 1, \ldots, k$ and $j = 1, \ldots, m$ is a number of the algorithm's iteration. Let $C_m = \{c_{lm}\}_{l=1}^{k}$ be the set of centroids yielded by the algorithm after reaching termination condition (after m iterations). The value of m is set a posteriori once the stopping conditions are met. We start by selecting starting locations of all the k centroids $C_1 = \{c_{l1}\}_{l=1}^{k}$, they can be set randomly or upon applying an optimization algorithm (e.g k-means++ [1]). Then, the samples are assigned to the closest centroid by means of selected distance metric. In case of Euclidean distance ρ_E, the respective values are equal to $d_{i,l,j} = \rho_E(q_i, c_{lj})$. Subsequently, new locations of the centroids

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

$C_{j+1} = \{c_{l(j+1)}\}_{l=1}^{k}$ are computed, where the new coordinates of each $c_{l(j+1)}$ are mean values of all the respective coordinates for data points q_i assigned to a given l'th centroid in j'th iteration. Here $c_{l(j+1)} = (\bar{c}_{lj_1}, \bar{c}_{lj_2}, \ldots, \bar{c}_{lj_d})$ with $\bar{c}_{lj_{dm}} = (1/a_{lj}) \sum_{i=1}^{a_{lj}} \hat{q}_{lj_i}$, where $dm = 1, \ldots, d$, a_{lj} is a number of points $\hat{q}_{lj_i}$ linked to l'th centroid in j'th iteration. The algorithm iterates until one of the stopping conditions is met i.e. either when computed centroids no longer switch their location or the distance between c_{lj} and $c_{l(j+1)}$ is smaller than prescribed ε or lastly, if the preselected maximal number of iterations is exceeded.

4.3 K-medoids

K-medoids is a method applied for clustering [8]. The data is assigned to one of k medoids, where each medoid is a specific point from the dataset and represents its cluster. The value of k and the dissimilarity measure are arbitrally selected. In the first step of the algorithm, k points are chosen as starting medoids. The latter is achieved either in a random manner or upon applying a specific method (e.g k-means++ [1]) to solve an optimization problem leading to a reduction of computational time of the algorithm. The k-medoids problem can be solved with numerous algorithms such as: Partitioning Around Medoids (PAM), Voronoi-iteration k-medoids, Reynolds' improvements, FastPAM, FasterPAM algorithms, CLARA, CLARANS, FastCLARA and FastCLARANS, Lloyd's iterations [8, 17].

In Partitioning Around Medoids algorithm, once a starting set of medoids is selected the values of a cost function are evaluated for all possible swaps of a medoid in a given cluster with another point belonging to the same cluster. When all the combinations are computed we finally apply only this one which has the minimal value of evaluated function. Subsequently, the affiliation of all points to points being set as current medoids is recalculated. This procedure is repeated for all the medoids as long as the value of the cost function decreases, otherwise the algorithm terminates. PAM algorithm has a high computation complexity as it calculates all possible swaps of all the medoids. In such setting, this algorithm is in practice predominantly applicable to a small amount of input data. For more complex computations one can apply variants of PAM (such as FastPAM or FasterPAM).

5 Experiments and Results

The computations were conducted on two sample datasets, namely Wine Quality-White and Ionosphere [3]. Wine Quality-White comprises of 4898 wine samples described by eleven features. Each of the wine samples was categorized by experts to the quality measure (ranging within $C = \{3, 4, \ldots, 9\}$). In order to perform computations on this set of features, a Mean Square Error is calculated to verify how a prediction deviates from an actual class as wine quality is represented by scale measure from C. We remark here that a misclassification of a wine from the class 3 to the predicted class 4 is a minor concern as opposite to assigning it to class 9. The Ionosphere is a dataset describing signals that pass through the

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

ionosphere. These signals are classified into two disjoint groups: "good" signals having evidence of some type of structure and "bad" signals deprived of such a feature. This dataset consists of 351 samples described by 34 features. In this binary classification task, the rendered results are compared calculating accuracy (ACC) representing the percentage of correctly classified samples in the whole classification process. These datasets are used here for classification with the aid of ELM-RBF applying a selected clustering method: k-means, k-medoids or mean shift. All computations described in this work are performed in Matlab.

In preliminary computations, which were carried on Wine Quality-White dataset, some of the parameters were selected and they are fixed in this work. The estimation of generalized inverse for ELM-RBF is based on Cholesky factorization of a singular matrix [12] yielding fast computation time and low MSE. To compare k-means, k-medoids and mean shift, different activation functions are chosen to obtain the best clustering results. In doing so, a linear activation function $f_L(x) = ax$ is used as it yields prominent classification results while applying k-means and k-medoids. On the other hand, Softplus activation function $f_{SP}(x) = log(1 + \exp(x))$ is also applied as it renders the best categorization results for mean shift.

Our computations exploit Matlab implementations of k-means, k-medoids and mean shift. Most of the conducted experiments admit in these methods default parameters unless specified otherwise. More specifically, the k-means and k-medoids clustering algorithms in Matlab have default distance metric set to Squared Euclidean distance. In addition, a default method for choosing initial cluster centroid positions is k-means++ algorithm [1]. In case of k-medoids, an algorithm to find medoids is available in three variants which are applied by default depending on number of rows (samples) in the input data. More specifically, for the number of rows less than 3000 PAM algorithm is applied. For the number of rows between 3000 and 10000 a variant of the Lloyd's iterations is selected. In case of larger datasets, a default algorithm is CLARA [8]. Thus, the algorithm applied for Wine Quality-White and Ionosphere datasets are Lloyd's iterations and PAM, respectively. In this work, the applied mean shift algorithm [6] allows the implementation of Gaussian or flat kernel for distance calculations. The classification results were generated for both datasets applying 10 times a 20% cross-validation. Note that the computation results obtained for same values of parameters can still vary as all three clustering methods rely on randomness. Such difference is still noticeable even upon applying multiple cross-validations. The computations are conducted on three computers: K1 - Ryzen 5600G CPU, 16 GB DDR4 3600MHz RAM, K2 - Ryzen 3900X CPU, 64GB DDR4 3600MHz and K3 - Dell 7750 Xeon W10885M CPU, 128GB DDR4 2933 MHz.

A classification task is performed on Wine Quality-White dataset for ELM-RBF with k-means and k-medoids for k ranging from 10 to 100 (with step-size 10) (see Fig. 1). These computations are applied on computer K1. The number of hidden-layer neurons n varies from 100 to 1000 (with step-size 100). Lastly, a linear or Softplus activation function is applied. For both k-means and k-medoids MSE is the highest for $k \in \{10, 20\}$. The lower value of k gets,

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

the worse classification result is rendered. Indeed, in the extreme case MSE attains the value 4.780 for k-medoids with $k = 10$ and $n = 500$ combined with Softplus activation function. For $k > 20$ the value of MSE for k-means and k-medoids stabilizes within the interval [0.619,0.793]. The best MSE result equal to 0.619 is achieved by k-medoids with linear activation function, $k = 100$ and $n = 700$ rendering the computation time equal to 409 seconds. Slightly worse result MSE = 0.628 is achieved for k-medoids with $k = 40$, $n = 100$ and linear activation function. Nevertheless, selecting the last set of parameters reduces computation time to 78 seconds. The best result obtained applying k-means yields MSE = 0.620 for $k = 100$, $n = 700$ and linear activation function, for which the computation time is equal to 402 seconds.

The mean shift clustering method is combined with ELM-RBF and the results obtained for Wine Quality-White dataset are based on the following choice of parameters: the values of the boundary width (radius) r are attuned from 0.3 to 1.3 (with step-size 0.1 - as it generated amounts of clusters from around 2 to 614) with the number of neurons n varying from 100 to 1000 (with step-size 100) combined with a linear or Softplus activation functions and Gaussian or flat kernel (see Fig. 2). The best result achieved for ELM-RBF with mean shift for the tested parameters is MSE = 0.686 for Softplus activation function, $n = 700$, $r = 0.6$ (which renders an averege of 22 clusters in 10 cross-validations) with flat kernel function applied. This result is worse then the best result for k-medoids (and k-means) for this dataset for same tested number of neurons. The computation time is equal to 323 seconds. The best MSE (the lowest) are achieved for $r \in \{0.6, 0.7\}$ rendering between 10 and 22 clusters. ELM-RBF with mean shift obtains the worst MSE results for $r \leq 0.5$ (when $r = 0.5$ around 52 clusteres are rendered) applying linear kernel function and reaches even 2.02.

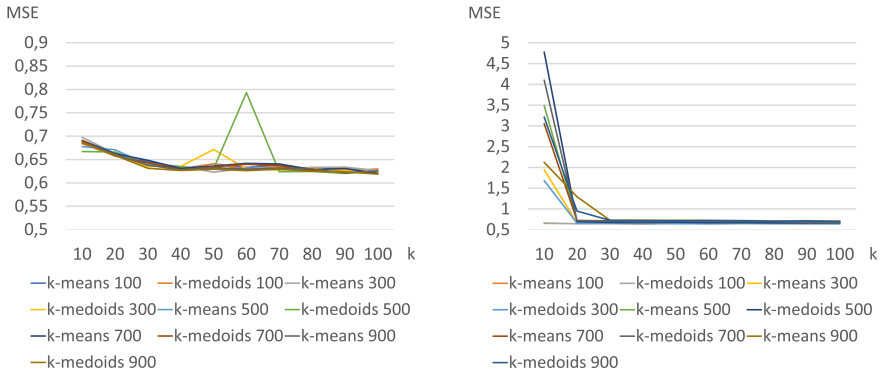

Fig. 1. MSE calculated on classification result for Wine Quality-White dataset for ELM-RBF with k-means or k-medoids , k varying between 10 and 100, 100-900 neurons applying linear (left) or Softplus activation function (right).

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

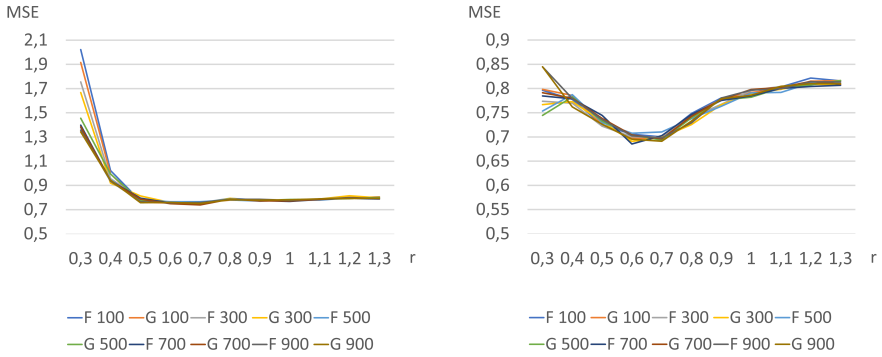

Fig. 2. MSE calculated on classification result for Wine Quality-White dataset for ELM-RBF with mean shift for r varying between 0.3 and 1.3, 100-900 neurons applying linear (left) or Softplus activation function (right), and Gaussian (G) or flat kernel (F).

The results generated by ELM-RBF on Wine Quality-White dataset are also analyzed on higher number of neurons n ranging from 1500 to 5000 (with step-size 500), for k varying from 10 to 100 (with step-size 10) in k-means and k-medoids applying linear or Softplus activation function (see Fig. 3). These computations are performed on the computer K2. Top three results are attained for k-medoids with linear activation function for $n = 2000$, $k = 100$, MSE $= 0.618$, for $n = 2000$, $k = 70$, MSE $= 0.618$ and for $n = 2500$, $k = 70$, MSE $= 0.619$. These results are equal to the best result for n ranging between 100 and 1000, but their computation time is much longer amounting to 1241, 1239 and 1676 seconds, respectively. Even if these computations were performed on faster computer, a huge increase in time is noticeable due to the enlarged number of neurons. The difference in results generated by Softplus and linear activation function is significant. Indeed, the values of MSE for linear function are in the interval $[0.618, 0.788]$, whereas for Softplus in $[0.732, 3.338]$. ELM-RBF with mean shift is also tested on higher values of n admitted to vary between 1500 and 5000 (with step-size 500) (see Fig. 4). The selected values of r range from 0.6 to 1 (with step-size 0.1) yielding the number of clusters between 4 and 22. The best MSE is equal to 0.688 for $n = 2000$, Gaussian kernel, Softplus activation function and $r = 0.7$ (11 clusters) computed in 1303 seconds.

The classification is also conducted on Ionosphere dataset applying ELM-RBF combined with k-means and k-medoids as clustering methods. The calculations were performed on computer K1. The computations were executed for k ranging from 10 to 100 for a number of neurons n varying from 100 up to 1000 (with step-size 100) (see Fig. 5). The results generated with linear activation function outperform those rendered with Softplus. The highest accuracy equal to 0.946 is reached for k-medoids with linear function, $n = 100$, $k = 96$ taking 13 seconds of execution time. The best result for k-means ACC $= 0.941$ is obtained for linear function, $n = 200$, $k = 80$ in 9 seconds. The accuracy for lin-

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

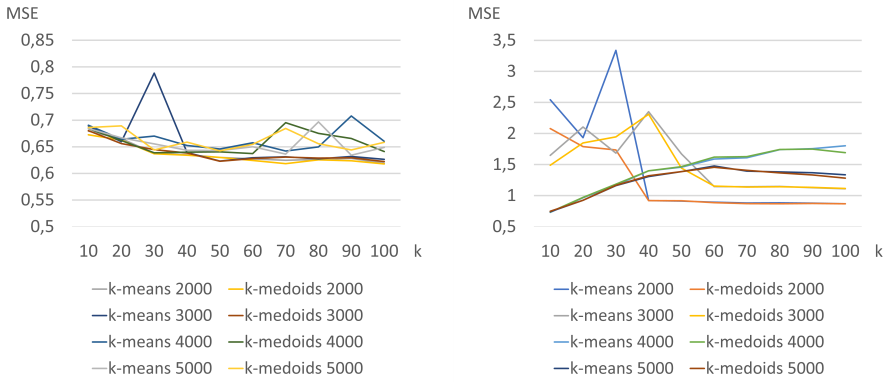

Fig. 3. MSE calculated on classification result for Wine Quality-White dataset for ELM-RBF with k-means or k-medoids for k varying between 10 and 100, 2000-5000 neurons applying linear (left) or Softplus activation function (right).

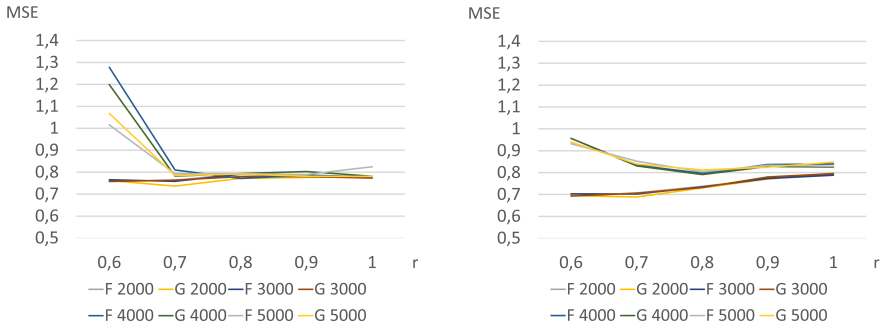

Fig. 4. MSE calculated on classification result for Wine Quality-White dataset for ELM-RBF with mean shift for r varying between 0.6 and 1, 2000-5000 neurons applying linear (left) or Softplus activation function (right), and Gaussian (G) or flat kernel (F).

ear activation function rapidly increases with k running over $k = \{1, 2, \ldots, 10\}$. Once $k > 10$ the rate of improvement in classification results decelerates. The best 39 results are generated for $n \leq 200$. The lowest ACC for linear function equal to 0.599 is attained for k-medoids $k = 3$, $n = 900$ and the worst overall result for the considered methods and parameters reads as ACC = 0.587, and is achieved for k-medoids with Softplus activation function combined with $n = 200$ and $k = 11$. ELM-RBF with mean shift is also used as a classification method on Ionosphere dataset. The considered parameters' values are: n ranging from 100 to 1000 (with step-size 100), r from 0.5 to 7 (step-size 0.5) rendering from 1 to 216 clusters (see Fig. 6). The highest ACC = 0.808 is registered for Softplus activation function, $n = 100$, $r = 4.5$ rendering around 14 clusters. The 40 best

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

results calculated with Softplus function are generated for $r = 4.5$ or $r = 5$ yielding around 14 and 4 clusters, respectively.

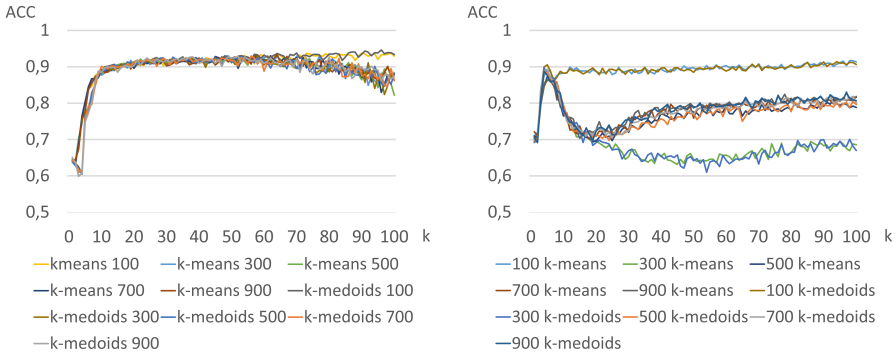

Fig. 5. ACC calculated on classification result for Ionosphere dataset for ELM-RBF with k-means or k-medoids for k varying between 1 and 100, 100-900 neurons applying linear (left) or Softplus activation function (right).

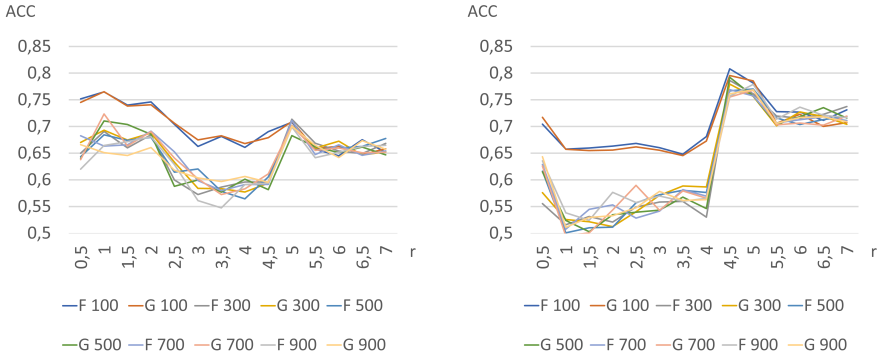

Fig. 6. ACC calculated on classification result for Ionosphere dataset for ELM-RBF with mean shift for r varying between 0.5 and 7, 100-900 neurons applying linear (left) or Softplus activation function (right), and Gaussian (G) or flat kernel (F).

K-means and k-medoids applied to ELM-RBF for clustering are tested on Ionosphere for larger number of neurons in the hidden layer. Tested parameters are n ranging from 1500 to 5000 (with step-size 500) and k from 1 to 100 (see Fig. 7). These computations are performed on the computer K3. The best ACC = 0.937 is attained for k-means with linear function, $n = 1500$ and $k = 49$. In

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

contrast, the worst ACC equal to 0.602 is obtained for k-medoids with linear function, $n = 3000$ and $k = 4$. The best result for $1500 \leq n \leq 5000$ equal to 0.937 is lower then the best result for $100 \leq n \leq 1000$ reading as ACC = 0.946. The classification process is performed on Ionosphere with ELM-RBF combined with mean shift testing the values of n ranging from 1500 to 5000 (with step-size 500) (see Fig. 8). The admitted values of r are taken from 0.5 to 7 (with step-size 0.5). The top 32 ACC, which are higher than 0.74, are obtained for $r = 4.5$ or $r = 5$ (around 13 or 4 centroids, respectively). The best ACC = 0.792 is achieved with Softplus function, $n = 3500$, $r = 4.5$ and Gaussian kernel. This result is worse than the best one calculated for $100 \leq n \leq 1000$ - ACC = 0.808.

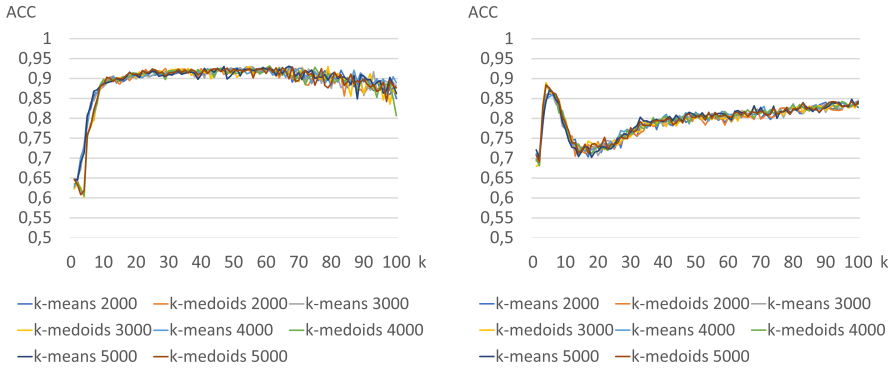

Fig. 7. ACC calculated on classification result for Ionosphere dataset for ELM-RBF with k-means or k-medoids for k varying between 1 and 100, 2000-5000 neurons applying linear (left) or Softplus activation function (right).

Feature selection method - Fast Correlation Based Filter (FCBF) [22] - is applied on Ionosphere to reduce the set of features leaving those that are highly correlated with affiliation to the class and their correlation between other features is low. The initial set of features is reduced from 34 to 4 represented by numbers: 5, 6, 28 and 33. The real aim of applying feature selection filtering is the hope to improve classification result and to reduce computation time. In the next step, ELM-RBF combined with k-means and k-medoids is tested on the set of features selected from Ionosphere. These computations are performed on the computer K2. The parameters involved are n varying between 100 and 1000 (with step-size 100) and k ranging from 1 to 100 (with step-size 1). The best ACC equal to 0.910 is obtained for k-means, linear activation function, for $n = 200$ and $k = 34$ and is computed in 11 seconds. This result is worse then the best result for the whole set of features which is equal to 0.946. The classification is also conducted on the selected Ionosphere features applying ELM-RBF with mean shift clustering method for n running from 100 to 1000 (with step-size 100), $r \in [0.01, 1.5]$ (with step-size 0.01) rendering from 1 up to 255 clusters. The best

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

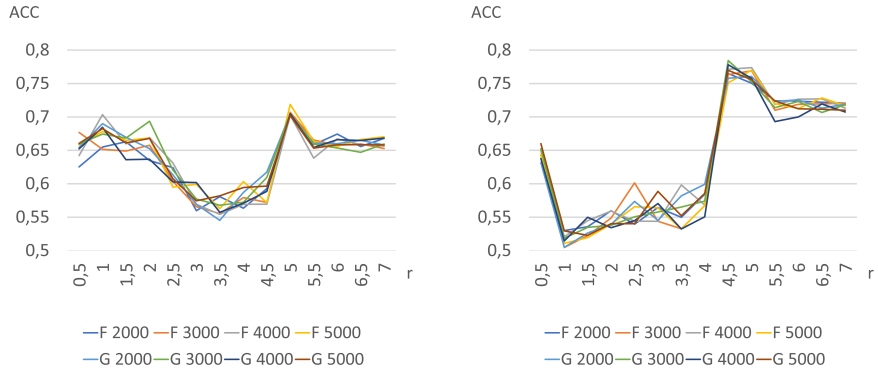

Fig. 8. ACC calculated on classification result for Ionosphere dataset for ELM-RBF with mean shift for r varying between 0.5 and 7, 2000-5000 neurons applying linear (left) or Softplus activation function (right), and Gaussian (G) or flat kernel (F).

achieved result ACC = 0.892 is obained for linear activation function (on the whole set of features Softplus activation function gave the best results for mean shift), flat kernel, $n = 100$, $r = 0.51$ rendering 40 clusters and is calculated in 6 seconds outperforming the best result obtained for the whole set of features applying mean shift clustering method which is equal to 0.808 for Softplus, flat kernel, $n = 100$ and $r = 4.5$ (14 clusters).

A juxtaposition of the best ACC or MSE generated for all the selected ranges of parameters for ELM-RBF with k-means, k-medoids and mean shift for Wine Quality-White and Ionosphere is presented in Tab. 1 and Tab. 2.

In previous computations, the default distances for k-means and k-medoids were applied and the activation functions (Softplus and linear) were selected a priori as they gave the best classification results for Wine Quality-White dataset. Subsequently, the parameters that yielded the best ACC for the whole set of features in Ionosphere for the three considered clustering methods were selected and other distance functions for k-means and k-medoids were considered and com-

n rng	**k rng**	**r rng**	**Method**	**act fun**	**n**	**r**	**k**	**Kernel**	**MSE**	**t**
100-1000	10-100	-	k-medoids	linear	700	-	100.0	-	0.6188	409
100-1000	10-100	-	k-means	linear	700	-	100.0	-	0.6202	402
100-1000	-	0.3-1.3	mean shift	Softplus	700	0.6	21.5	flat	0.6856	323
1500-5000	10-100	-	k-means	Softplus	4000	-	10.0	-	0.7315	2357
1500-5000	10-100	-	k-medoids	Softplus	4500	-	10.0	-	0.7368	2566
1500-5000	-	0.6-1	mean shift	Softplus	2000	0.7	10.8	Gaussian	0.6882	1303

Table 1. The best classification results (measured with MSE) on Wine Quality-White dataset applying ELM-RBF for each of the considered clusterization methods (k-means, k-medoids and mean shift) on analyzed ranges of parameters. *Act fun* column stands here for the activation function, t for computation time in seconds and *rng* for range.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

bined with various activation functions tested for all three clustering methods. In case of k-means, the tested distances are: *cityblock*, *correlation*, *cosine* and *sqeuclidean*. In case of k-medoids: *chebychev*, *cityblock*, *correlation*, *cosine*, *euclidean*, *hamming*, *jaccard*, *minkowski*, *spearman* and *sqeuclidean* were analyzed [13]. For k-means, k-medoids and mean shift the tested activation functions for ELM-RBF are: sigmoid, tanh, relu, rbf, linear, swish, ELiSH, HardTanH, TanhRe, ELUs, Softplus, LReLU and BinaryStep [16]. The best five results obtained for k-means and k-medoids are presented in Tab. 3. For both k-means and k-medoids the best classification results are rendered for linear activation function. The highest ACC is obtained for *cityblock* (ACC = 0.94) and *sqeuclidean* distance (ACC = 0.93). The best five results for mean shift with $n = 100$, $r = 4.5$ (rendering 14 or 13 clusters) and flat kernel, which are computed in $t \in [4.1, 5.1]$ seconds are rendered for: Softplus, tanh, swish, TanhRe and sigmoid activation functions, and their ACC are equal to: 0.794, 0.783, 0.780, 0.769 and 0.763, respectively.

6 Conclusions

The best classification results for Wine Quality-White with ELM-RBF are obtained for k-medoids and k-means. The latter is attained for $n \in N_{1000} = \{100, 200, \ldots, 1000\}$ with MSE = 0.62 which is 0.07 better than the best result applying mean shift. As it turns out, further enlargement of n (from 1500 to 5000) does not improve the MSE results. The best classification result is achieved for $n = 700$. Similarly, the best classification results for Ionosphere dataset are derived for k-means and k-medoids with $n \in N_{1000}$ for which ACC $\in [0.94, 0.95]$. Again, admitting a higher number of neurons on this dataset does not ameliorate the classification results - a fenomenon also manifested on Wine Quality-White data. The best classification result is achieved for $n = 100$, which is the lower

n rng	**k rng**	**r rng**	**features**	**Method**	**act fun**	**n**	**r**	**k**	**krn**	**ACC**	**t**
100-1000	1-100	-	all	k-medoids	linear	100	-	96	-	0.946	13
100-1000	1-100	-	all	k-means	linear	200	-	80	-	0.941	9
100-1000	-	0.5-7	all	mean shift	Softplus	100	4.50	14	f	0.808	5
1500-5000	1-100	-	all	k-means	linear	1500	-	49	-	0.938	60
1500-5000	1-100	-	all	k-medoids	linear	4500	-	48	-	0.933	169
1500-5000	-	0.5-7	all	mean shift	Softplus	3500	4.50	13	g	0.792	155
100-1000	1-100	-	selected	k-means	linear	200	-	34	-	0.910	11
100-1000	1-100	-	selected	k-medoids	linear	200	-	41	-	0.907	14
100-1000	-	0.01-1.5	selected	mean shift	linear	100	0.51	40	f	0.892	6

Table 2. The best classification results (measured with ACC) on Ionosphere dataset applying ELM-RBF for each of the considered clustering methods (k-means, k-medoids and mean shift) on analyzed ranges of parameters. *Act fun* stands here for the activation function, *rng* for range, *krn* for kernel (f - flat or g - Gaussian) and t for computation time in seconds. In features column there is an information about whether the computations were performed on the whole set of features or only on ones selected by FCBF.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

Distance	Method	activation function	n	k	ACC	time
cityblock	k-medoids	linear	100	96	0.940	11
sqeuclidean	k-medoids	linear	100	96	0.933	11
cityblock	k-medoids	TanhRe	100	96	0.929	13
cityblock	k-medoids	ELUs	100	96	0.927	13
cityblock	k-medoids	Softplus	100	96	0.927	12
cityblock	k-means	linear	200	80	0.942	9
sqeuclidean	k-means	linear	200	80	0.927	9
sqeuclidean	k-means	relu	200	80	0.850	10
sqeuclidean	k-means	HardTanH	200	80	0.833	11
cityblock	k-means	TanhRe	200	80	0.828	11

Table 3. The top 5 ACC on the Ionosphere dataset for k-means, $n = 100$, $k = 96$ and k-medoids, $n = 200$, $k = 80$ analyzed for different distances and activation functions.

bound of the analyzed numbers of neurons. For both datasets and lower values of n the best results for k-means and k-medoids are attained for high values of $k \in \{80, 81, \ldots, 100\}$. Again, admitting a higher number of neurons n the best results are obtained for lower values of k, i.e. $k = 10$ for Wine Quality-White and $k \in \{48, 49\}$ for Ionosphere. Furthermore, classification conducted on features selected by FCBF from Ionosphere dataset does not improve the overall best classification result. Nevertheless, the ACC obtained by ELM-RBF with mean shift on reduced set of data increases accuracy rate from 0.81 to 0.89 ACC. For k-means and k-medoids the best results are obtained with the aid of linear activation function (and *cityblock* or *sqeuclidean* distance metrics). Mean shift rendered in most computations the best results on Softplus activation function; however, the best outcome achieved with this clustering method on reduced set of Ionosphere features is attained with linear activation function. In further research, one should verify results rendered applying other parameters especially when the best classification in this work is observed on their boundary values as it is expected to obtain in those cases better results. The *cityblock* metric should be further analyzed for k-means and k-medoids in ELM-RBF.

References

1. Arthur, D., Vassilvitskii, S.: K-means++: The advantages of careful seeding. In: Proc. Annu. ACM-SIAM Symp. Discrete Algorithms. pp. 1027–1035 (2007)
2. Dhini, A., Surjandari, I., Kusumoputro, B., Kusiak, A.: Extreme learning machine – radial basis function (ELM-RBF) networks for diagnosing faults in a steam turbine. J. Ind. Prod. Eng. **39**(7), 572–580 (2022). https://doi.org/10.1080/21681015.2021.1887948
3. Dua, D., Graff, C.: UCI machine learning repository (2017), http://archive.ics.uci.edu/ml
4. Finkston, B.: Mean shift clustering (2023), http://bit.ly/3wVVngu
5. Fukunaga, K., Hostetler, L.: The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Trans. Inf. Theory **21**(1), 32–40 (1975). https://doi.org/10.1109/TIT.1975.1055330

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13

6. Gong, H.: An open-source implementation of meanshift clustering for matlab/octave. (2015), https://github.com/hangong/meanshift_matlab
7. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: a new learning scheme of feedforward neural networks. In: IEEE Proc. Int. Jt. Conf. Neural Netw. vol. 2, pp. 985–990 (2004). https://doi.org/10.1109/IJCNN.2004.1380068
8. Kaufman, L., Rousseeuw, P.: Finding Groups in Data: An Introduction to Cluster Analysis. John Wiley (1990). https://doi.org/10.2307/2532178
9. Konopka, A., Struniawski, K., Kozera, R., Trzciński, P., Sas-Paszt, L., Lisek, A., Górnik, K., Derkowska, E., Głuszek, S., Sumorok, B., Frac, M.: Classification of soil bacteria based on machine learning and image processing. In: Computational Science – ICCS 2022. pp. 263–277. Springer International Publishing (2022). https://doi.org/10.1007/978-3-031-08757-8_23
10. Leung, H.C., Leung, C.S., Wong, E.W.M.: Fault and noise tolerance in the incremental extreme learning machine. IEEE Access **7**, 155171–155183 (2019). https://doi.org/10.1109/ACCESS.2019.2948059
11. Li, H.T., Chou, C.Y., Chen, Y.T., Wang, S.H., Wu, A.Y.: Robust and lightweight ensemble extreme learning machine engine based on eigenspace domain for compressed learning. IEEE TCAS-I **66**(12), 4699–4712 (2019). https://doi.org/10.1109/TCSI.2019.2940642
12. Lu, S., Wang, X., Zhang, G., Zhou, X.: Effective algorithms of the Moore-Penrose inverse matrices for extreme learning machine. Intell. Data Anal. **19**(4), 743–760 (2015). https://doi.org/10.3233/IDA-150743
13. MathWorks: k-medoids clustering - Matlab k-medoids (2023), http://bit.ly/3RwXlNR
14. McCulloch, W.S., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. Bull. Math. Biophys. **5**(4), 115–133 (1943). https://doi.org/10.1007/BF02478259
15. Mojrian, S., Pinter, G., Joloudari, J.H., Felde, I., Szabo-Gali, A., Nadai, L., Mosavi, A.: Hybrid machine learning model of extreme learning machine radial basis function for breast cancer detection and diagnosis; a multilayer fuzzy expert system. In: RIVF. pp. 1–7 (2020). https://doi.org/10.1109/RIVF48685.2020.9140744
16. Nader, A., Azar, D.: Evolution of activation functions: an empirical investigation. ACM TELO **1**(2), 1–36 (2021). https://doi.org/10.1145/3464384
17. Park, H.S., Jun, C.H.: A simple and fast algorithm for k-medoids clustering. Expert Syst. Appl. **36**(2), 3336–3341 (2009). https://doi.org/10.1016/j.eswa.2008.01.039
18. Peng, X., Lin, P., Zhang, T., Wang, J.: Extreme learning machine-based classification of ADHD using brain structural MRI data. PLOS ONE **8**(11), 1–12 (2013). https://doi.org/10.1371/journal.pone.0079476
19. Pérez-Ortega, J., Almanza-Ortega, N.N., Vega-Villalobos, A., Pazos-Rangel, R., Zavala-Díaz, C., Martínez-Rebollar, A.: The k-means algorithm evolution. In: Introduction to Data Science and Machine Learning, chap. 5. IntechOpen, Rijeka (2019). https://doi.org/10.5772/intechopen.85447
20. Rao, C.R., Mitra, S.K.: Generalized Inverse of Matrices and its Applications. John Wiley & Sons (1971)
21. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**(6088), 533–536 (1986). https://doi.org/10.1038/323533a0
22. Yu, L., Liu, H.: Feature selection for high-dimensional data: a fast correlation-based filter solution. In: ICML. pp. 856–863 (2003)

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-36027-5_13