Knowledge-Based Systems 299 (2024) 112025

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Optimal evolutionary framework-based activation function for image classification

Luca Parisi [a,b,c,d,*], Ciprian Daniel Neagu [a], Narrendar RaviChandran [c], Renfei Ma [e], Felician Campean [a]

[a] Faculty of Engineering and Digital Technologies, University of Bradford, Richmond Road, Bradford, BD7 1DP, United Kingdom
[b] Tutorantis, Department of Computer Science, 5 South Charlotte Street, Edinburgh, EH2 4AN, United Kingdom
[c] University of Auckland Rehabilitative Technologies Association (UARTA), University of Auckland, 11 Symonds Street, Auckland, 1010, New Zealand
[d] Parkinson's UK, 215 Vauxhall Bridge Rd, Pimlico, London SW1V 1EJ, United Kingdom
[e] Faculty of Biology, Shenzhen MSU-BIT University, Shenzhen, Guangdong, China

ARTICLE INFO

Keywords:
Machine learning
Decision support
Image classification
Genetic algorithm
Activation functions
Neural networks

ABSTRACT

Typically, supervised Machine Learning (ML)-based image classifiers leverage algorithms derived from either Artificial Neural Networks (ANNs) or optimal separating hyperplane (OSH)-based algorithms. However, despite recent progress has been made to enhance ANNs' classification performance via the Rectified Linear Unit (ReLU)-based activation functions (AFs), there is currently no AF that scales across and benefit both ANNs and OSH-based classifiers. Moreover, the lack of globally optimal AFs leads to a high variance in image classification-related results. Thus, this study seeks to overcome this limitation by implementing a next-generation evolutionary framework ('ActiGen') to generate a novel and more reliable AF, which can scale to two families of AFs for two classifiers. The proposed evolutionary knowledge-based framework leverages a Multi-Objective (MO) optimisation method based on Genetic Algorithms (GA), or 'MOGA', to improve the generalisation of such classifiers. This evolutionary framework and its generated AF are validated using nine open-access datasets: seven image-based datasets, consisting of 22,136 images in total, and two large (561 features for 10,929 instances, 124 features for 1,700 instances) tabular datasets. These diverse datasets include both binary and multi-class classification, such as images of breast masses, those acquired via cardiac computed tomography, photos of famous people from the Internet, images of handwritten digits and those drawn on a graphics tablet, human faces with different lighting, details, and expressions, smartphone-related data captured during various activities and postural transitions, and clinical data on complications of myocardial infarction. Findings demonstrate that the proposed evolutionary optimisation framework ('ActiGen-MOGA') was able to generate a novel scalable AF, which led to achieve the highest classification performance and the fastest convergence across six out of nine datasets. In the best classification task, the ActiGen-MOGA-based AF led to a classification performance of 80 % and 78 % higher than the polynomial and Rectified Linear Unit (ReLU) AFs respectively.

1. Introduction

1.1. The need for an evolutionary framework to derive a reliable activation function for image classification

Recent advancements in supervised learning have significantly contributed to the fields of kernel and activation functions (AFs), particularly for optimal separating hyperplane (OSH)-based classifiers like the Support Vector Machine (SVM) [8] and Artificial Neural Networks (ANNs), such as the Multi-Layer Perceptron (MLP) [49]. These developments have been well-documented [40]. However, a critical gap remains in providing usable, reproducible, replicable, and reliable functions that can apply to both categories of classifiers, specifically SVMs and MLPs [40]. Furthermore, the absence of automated frameworks for deriving optimal AFs tailored to chosen classifier families and candidate AF families for specific tasks highlights the need for a scalable AF across various classifiers. Currently, addressing these challenges largely relies on manual interventions by subject matter experts,

* Corresponding author.
E-mail addresses: luca.parisi@tutorantis.com, lparisi1@bradford.ac.uk, luca.parisi@ieee.org (L. Parisi), d.neagu@bradford.ac.uk (C.D. Neagu), nrav195@aucklanduni.ac.nz (N. RaviChandran), marenfei@smbu.edu.cn (R. Ma), f.campean@bradford.ac.uk (F. Campean).

https://doi.org/10.1016/j.knosys.2024.112025
Received 29 May 2023; Received in revised form 27 February 2024; Accepted 29 May 2024
Available online 31 May 2024

primarily Machine Learning (ML) engineers or data scientists. Thus, to fill this research gap, there is the need for deriving an optimal and scalable AF in an automated manner. Optimal is defined as reaching the global minimum for the chosen algorithms.

OSH- and ANN-based learning processes involve defining optimal decision boundaries [8,23,29,35,51] and constructing neural maps to capture patterns for class discrimination [10,32–34,49], respectively. Existing wisdom suggests distinct sets of gold-standard functions for these algorithms, as evidenced in the Python 'scikit-learn' library for ML [3,47]. The quest for a single function capable of scaling and benefiting both supervised learning-based algorithm categories represents a complex optimisation problem due to inherent differences in their learning processes. Open-source software, notably Python and related communities, offer accessible resources such as the 'scikit-learn' library [47], which provides classes like 'MLPClassifier' and 'SVC' (Support Vector Classifier) implementing MLPs and SVMs, respectively

Despite state-of-the-art AFs, real-world classification tasks often encounter reliability-related issues, including slow convergence [21] or a lack thereof [53]. These issues arise from challenges like local minima during optimisation [6,43–45]. While partial solutions exist, such as weighted averaging of outputs [16,24,55], these approaches provide limited ad-hoc improvements. Furthermore, existing solutions primarily focus on optimizing classifier hyperparameters or employ band-aid approaches. Notably, scikit-learn offers separate sets of AFs for OSH-based classifiers and ANNs, making it challenging to find a function suitable for both SVMs and MLPs simultaneously [40]. The only commonality lies in the 'sigmoid' kernel function for SVMs and its variant 'hyperbolic tangent sigmoid' ('tanh') [27], which better suits ANNs due to its extended range and steeper derivatives.

This study aims to address these challenges by proposing a hybrid ML-driven approach that blends the fundamental properties of individual classifiers to define an AF that scales across multiple classifier types. This approach targets the unique learning processes of each classifier to optimise their predictive potential. The rationale behind optimizing activation functions (AFs) for Support Vector Machines (SVMs) and Multilayer Perceptrons (MLPs) is rooted in the requirement to ensure consistent and harmonised learning processes across these two distinct model families, especially in the context of ensemble classifiers and two-headed models [22].

Ensemble classifiers often use a combination of SVMs and MLPs, each with its unique AFs, to harness the strengths of both models. However, the challenge arises in achieving cohesion and synergy between these diverse components [22]. This is where the need for a unified AF becomes evident. By employing a unified AF that is adaptable and optimised for both SVMs and MLPs, we establish a consistent foundation for the learning processes within ensemble classifiers [40]. This consistency ensures that data is processed and classified in a uniform manner, irrespective of whether it is processed by an SVM or an MLP.

This cohesion in learning processes not only simplifies the overall modeling approach but also promotes the convergence of learning towards globally optimal solutions. It mitigates the discrepancies that may arise from using different AFs and facilitates the harmonization of the ensemble's decision-making process. Ultimately, the utilisation of a unified activation function across SVMs and MLPs contributes to achieving more reliable and consistent image classification-related results, leading to enhanced performance in ensemble classifiers and two-headed models [24,40,55].

The research focuses on establishing an innovative approach based on open-source kernels, AFs, and frameworks that can semi-automate their selection and optimisation. This approach aims to save time, enhance consistency, and systematically discover AFs suitable for a variety of classifiers, datasets, applications, and levels of heterogeneity and noise. This solution seeks to scale the AF across both SVMs and MLPs, achieving faster convergence and ultimately resulting in more reliable ML-driven models. Recent enhancements in AFs for image classification, including 'Quantum ReLU' (QReLU) and its modified version ('m-QReLU') addressing the 'dying ReLU' problem, as well as 'hyper-sinh' for deep neural networks (DNNs), have been achieved through expert-driven and heuristic methods, focusing on specific algorithms like DNNs [43,46]. However, these methods do not guarantee that the AF can scale to multiple classifier types, datasets, applications, and noise levels, particularly in complex image classification tasks.

Deploying image classifiers with globally non-optimal AFs may result in issues like exploding gradients in ANNs [32-34] or convergence problems in OSH-based SVM classifiers [8]. These drawbacks can hinder generalization, affecting classifier performance across diverse datasets and applications [36,37]. To ensure consistent classification performance, it is imperative to establish an optimisation framework for generating novel, reliable AFs that can scale across multiple classifier categories. This framework would enable the semi-automated discovery of optimal AFs suitable for various learning-based algorithms, such as ANN- and OSH-based models, thereby enhancing image classification for decision support.

1.2. The need for a novel multi-objective evolutionary approach to improve image classification

Evolutionary algorithms, such as Genetic Algorithms (GAs), offer a valuable approach for identifying the global minimum within solution spaces. They operate by simulating bio-inspired processes, including mutation and crossover, and these processes are mathematically represented through fitness functions [18,26]. GAs have demonstrated success in optimizing activation functions (AFs) to create hybrid intelligent systems tailored for data classification in constrained scenarios [41]. However, their applicability falters when faced with large and noisy image datasets, as previously acknowledged in the literature [18,26,30,38,39,50].

The limitation of GAs in handling these challenging datasets lies in their tendency to converge to local minima. Consequently, improvements in existing AFs and related classification performance remain modest within this constrained optimisation framework [41]. Traditional AF optimisation for image classifiers typically relies on a manual and heuristic selection of optimal hyperparameter ranges [16,24]. This approach yields only marginal enhancements and carries a low probability of discovering transformative solutions capable of significantly advancing image classification performance.

As a consequence, the use of single-objective GA-based optimisation, although effective in denoising data [42], cannot ensure AFs' generalizability across multiple classifier types, datasets, and applications. The prevailing method for mathematically deriving AFs is expert-driven and entails a laborious, multi-step process of defining baseline candidate AFs. This approach is time-consuming, not entirely reproducible, and falls short of optimizing the learning process to attain the necessary robustness for image classification, particularly in the presence of noisy images. To address these limitations, this study introduces a novel multi-objective GA-based optimisation framework. This framework aims to semi-automate the generation of novel and dependable AFs capable of scaling across various types of classifiers. It achieves this goal by optimising existing families of AFs, ultimately enhancing image classification results.

1.3. Related studies

In the existing literature, we have identified two key research works, each with a somewhat similar objective. The first study [16] sought to optimise the weights and an asymmetric Activation Function (AF) within a specific family of AFs in Artificial Neural Networks (ANNs) for time-series forecasting. This was achieved through computationally intensive techniques like simulated annealing and Tabu search. However, it is important to note that this approach was constrained by its reliance on pre-setting the weights and the order of a single family of asymmetric AFs. Furthermore, this method was not extended to benefit classification processes. This aligns with the broader literature [11,14,15,52], which primarily focused on optimizing the initialization of weights in ANNs. Unfortunately, these efforts did not significantly improve the existing state of practice since weights are typically initialised with small random numbers to prevent dead neurons before training and are subsequently updated through learning algorithms. Therefore, the initial optimisation task undertaken by Gomes & Ludermir [16] appears somewhat redundant and potentially counterproductive, as it may introduce bias into the weight initialization process by relying on a globally suboptimal evolutionary approach [25], rather than allowing networks to initialise weights from inherently unbiased small random numbers.

The second, more recent study conducted by Kunc & Kléma [24] centred on the development of an adaptive AF based on the performance of ANNs in reconstructing gene expression profiles. While this study fine-tuned the AF's parameters, it did not extend its optimisation to encompass the entire family of AFs. Consequently, this effort resulted in a narrowly-focused, task-specific optimisation of parameters for a selected AF. This approach may not generalise effectively to different types of classifiers, datasets, or applications.

In the broader literature, the practice has been to tailor the parameters of standard AFs slightly to suit the requirements of specific algorithms and their corresponding applications, whether in the context of time-series forecasting [16] or classification tasks [24]. This common approach has led to the manual, time-consuming, and non-scalable process of discovering novel AFs. It lacks generalizability across various classifier types and datasets, relying solely on the expertise of Subject Matter Experts (SMEs) and often resulting in only incremental improvements.

To derive an optimal AF and develop a versatile function that can effectively serve both Support Vector Machines (SVM) and Multilayer Perceptron (MLP), it is essential to recognise the substantial benefits of a unified activation function (AF). Drawing upon our previous research [38,39,41,42], this study introduces a multi-objective Genetic Algorithm (GA)-based framework. The significance of a unified AF for SVM and MLP becomes evident in its potential to substantially enhance the performance of both classifier types. Traditionally, SVM and MLP employ different AFs, leading to the need for distinct modeling processes and hyperparameter tuning. However, by optimizing a comprehensive set of hyperparameters, including polynomial degrees or orders, regularization parameters, and coefficients, this framework aims to create a novel, globally optimal AF.

This unified AF is designed to seamlessly integrate with both SVM and MLP, thereby simplifying and unifying the modeling process. This integration leads to improved image classification outcomes across a wide range of datasets and scenarios. Using the same AF for both classifier families, the framework ensures consistency and harmonisation in the way data are processed and classified. Furthermore, the proposed framework offers the added advantage of accelerating convergence in both MLP and SVM for image classification applications. This acceleration is especially valuable in real-world contexts, where computational efficiency and faster decision-making processes are crucial for timely and accurate results. Consequently, the development of a unified AF represents a substantial advancement in machine learning, as it simplifies the model selection process, promotes consistency, and ultimately enhances classification performance across diverse domains.

1.4. Rationale and aim of the proposed contribution

In the context of image classification, the manual and heuristic approaches discussed in sub-sections 1.1 and 1.3 reveal a significant problem. These methods lack an objective framework to guide the discovery of a novel, globally optimal, unified Activation Function (AF) in a semi-automated manner. Consequently, they cannot effectively optimise learning across various types of classifiers when dealing with noisy images. This issue arises because the current non-optimal AFs being used are neither generalizable [24] nor computationally efficient [16].

To address this challenge, our study introduces a novel expert-based multi-objective evolutionary framework. This framework optimises families of AFs and their associated hyperparameters. It then generates a blended novel AF tailored to the chosen types of classifiers. For the first time, we propose an objective optimisation framework for deriving a new AF capable of scaling across multiple types of image classifiers, such as Artificial Neural Networks (ANNs) and optimal separating hyperplane (OSH)-based algorithms like Multi-Layer Perceptron (MLP) and Support Vector Machine (SVM).

As detailed in sub-section 1.2, our multi-objective evolutionary approach aims to achieve higher image classification performance, irrespective of the dataset or application in question. Furthermore, this method semi-automates the process of discovering novel AFs once families of AFs have been selected for optimizing model learning processes. Consequently, it reduces the need for extensive human intervention by expert developers and machine learning engineers, as elaborated upon in sub-sections 1.1 and 1.3.

The novel AF produced through our evolutionary framework is rigorously validated and proven competitive against gold-standard alternatives, accommodating both kernel and AF requirements [40]. Additionally, we have made this novel AF freely accessible within the 'scikit-learn' Python library [47], making it readily available for use with both the 'MLPClassifier' and the 'SVC' classes.

In summary, this study offers two main contributions:

1) From a knowledge-based systems perspective: We have developed and validated a multi-objective evolutionary framework capable of generating novel AFs, thereby enhancing real-world image classification. This framework employs Genetic Algorithms (GA) to optimise chosen families of AFs, enabling scalability across two types of classifiers—ANN- and OSH-based models.
2) From a theoretical and scientific research standpoint: Through the multi-objective evolutionary approach described in point 1, we have discovered a new and more reliable AF for image classification. This novel AF is adaptable to various types of supervised learning-based algorithms. Consequently, it bridges the gap between ANN- and OSH-based classifiers in two ways: (1) it offers a reliable, common approach across both algorithm types, potentially facilitating explainability [58] and adoption, and (2) it consistently demonstrates robustness when consumed by various categories of classifiers.

1.5. Paper structure

The remaining sections cover the following topics: Section 2 introduces the proposed multi-objective evolutionary framework to generate the m-arcsinh AF for reliable image classification. Section 3 describes the methodology and data sources used for validating this novel contribution. Section 4 discusses the results and the potential of the proposed framework for human-in-the-loop-based ML-driven applications. Section 5 provides conclusions and highlights areas for future work.

2. Methods

2.1. Requirements for an optimal kernel for SVM and activation function for MLP

The optimal function for both a kernel in Support Vector Machines (SVM) and an activation function (AF) in Multilayer Perceptrons (MLP) [21,27,53] is chosen based on specific criteria detailed in the sub-section 4.3 on evaluation metrics. This selected function must satisfy two crucial requirements: (1) maximising the margin width in SVM and (2) improving the MLP's ability to categorise input data into target classes by appropriately extending the range of its underlying activation function. Two distinct families of functions have been identified as fulfilling these dual criteria, distinguishing them from other alternatives.

In the realm of kernels, the linear kernel, a widely applied representative of the kernel family, stands out as particularly well-suited for SVM due to its intrinsic capability to maximise margin width, a pivotal factor contributing to SVM's predictive performance. The linear kernel efficiently separates data by establishing linear decision boundaries, a fundamental principle underlying the strength of SVM. It is important to note, however, that the linear kernel is not suitable for MLPs, especially when dealing with non-linearly separable data. This limitation arises from its inability to facilitate effective gradient descent under such circumstances, rendering it less versatile for meeting the complex learning needs of MLPs.

Conversely, the s-shaped AF family, with hyperbolic tangent (tanh) as a representative member (as shown in Eq. (1)), exhibits a unique characteristic – an extended range coupled with sigmoidal behavior.

$$c_i = tanh\left(net_{(i,h)} + d_h\right), \{c_i \varepsilon R| - 1 \leq c_i \leq +1\} \tag{1}$$

- where c_i represents the output of the tanh activation function for a particular neuron or unit, often denoted as "i."
- *tanh* is the hyperbolic tangent function, which is used as the activation function in neural networks.
- $net_{(i,h)}$ represents the weighted sum of inputs to neuron "i" in a specific layer, denoted by "h." This is the result of multiplying each input by its corresponding weight, summing these products, and potentially adding a bias term.
- d_h is an optional bias term, which is added to the weighted sum before applying the tanh function.

2.2. The tanh function for MLP as basis of the optimal kernel for SVM

The tanh function possesses a distinctive feature in its output range, spanning from -1 to +1. This sets it apart from the sigmoid function, whose range extends from 0 to +1. The tanh function exhibits sigmoidal behavior, characterised by an S-shaped curve, with output values bounded between -1 and +1.

In neural networks, the tanh function is often favoured due to its capability to handle both positive and negative inputs, resulting in a centred activation around zero. This centeredness aids in faster convergence during training, particularly when dealing with data having a mean near zero. Eq. (1) for the tanh function outlines how inputs undergo transformation within a neural network's neuron, yielding an output within the range of -1 to +1, making it well-suited for various ML-driven tasks.

These unique characteristics position the *tanh* AF as an ideal choice for the MLP. Leveraging the *tanh* as a kernel function in SVM allows for reliable maximisation of the margin width, even when faced with data requiring non-linear separation. *Tanh*'s sigmoidal nature enables it to capture non-linear relationships within data, contributing to the adaptability of SVM. In contrast to the linear and *tanh* functions that can be hybridised to achieve non-linear and high-range learning, combining other functions, such as the Radial Basis Function (RBF) and sigmoid, may result in skewed learning tendencies that are more locally optimal for either SVM or MLP individually, but not globally optimal for both algorithms.

2.3. MLP's fundamental architecture and learning process informing the optimisation framework

To achieve globally optimal learning, it is essential to optimise the hyperparameters of the functions separately. This distinct optimisation approach is imperative, considering the diverse requirements and characteristics of the SVM and the MLP. Fine-tuning each function independently ensures optimal performance in their respective roles, with the linear kernel enhancing SVM's capabilities and tanh enabling MLP to effectively handle complex, non-linear data. This strategic dichotomy in optimisation enables the harnessing of the full potential of these functions within their specialised contexts, ultimately leading to superior classification outcomes.

The training of the MLP involves the application of the backpropagation algorithm, where initially randomised weighted inputs (denoted as 'w' in Eq. (2)) are propagated forward. These weights, initially assigned randomly, play a crucial role in determining the strength of connections between neurons in different layers of the MLP. The errors are iteratively propagated backward through training iterations or epochs, denoted as 'n' in Eq. (2), until the adjustment of the generated weights reaches the lowest mean squared error (MSE) between the predicted outputs and the actual target values [38,43,44]. Notably, an MLP with one hidden layer ('h') can mathematically describe any Boolean-bounded functions for both binary and multi-class classification [9,38,41,42]. The hidden layer(s) in an MLP assume a crucial role in capturing complex patterns and relationships within the data.

The proposed optimisation framework was expected to return one as the optimal number of hidden layers. To account for a larger range of inputs, the *tanh* AF (Eq. (1)), including an appropriate summation of weighted inputs (Eq. (2)), was expected to be the output from the proposed framework as a transfer function in the hidden layer. Furthermore, in Eq. (2), i for the 'number of hidden layers' to be optimised; x represents the input matrix, and b is the bias:

$$net_{(i,h)} = \sum_{1}^{i} \left(w_{(n,i)} \cdot xi + b\right)_h \tag{2}$$

The weights ('w') are specific to the connections between input neurons and hidden layer neurons. The weighted summation in Eq. (2) determines the input to the activation function (tanh in this case) for each neuron in the hidden layer. The equation also includes a bias term ('b') for each neuron in the hidden layer. The bias term allows for an additional degree of freedom in adjusting the neuron's activation threshold.

In summary, Eq. (2) encapsulates the essential components of an MLP's hidden layer operation during training. It showcases the role of weights, the application of the tanh activation function, and the inclusion of bias terms in determining the output of each neuron within the hidden layer. This combination of weighted summation and activation function application enables the MLP to learn and represent complex patterns in the data, ultimately contributing to its ability to perform tasks such as classification and function approximation.

2.4. ActiGen-MOGA: A multi-objective evolutionary framework for an optimal activation for reliable image classification

Conventional SVMs often face challenges in convergence, particularly when dealing with non-linearly separable inputs, leading to compromised generalisation and misclassifications. To address this, preventing trapping at local *minima* is crucial in SVMs, and guiding the definition of an appropriate Optimal Separating Hyperplane (OSH) to maximise the margin width can enhance the linear separability of inputs into the expected target classes.

In this study's innovative evolutionary framework, the process commences with the selection of an initial SVM-based learning function leveraging the linear kernel, parameterised by a real number denoted as 'p.' Concurrently, for the MLP, we initiate with a family of S-shaped activation functions, such as the sigmoid or logistic functions. These choices serve as the starting point for our evolutionary optimisation.

The Genetic Algorithm (GA) assumes a pivotal role in guiding this optimisation process, mitigating issues related to premature convergence and working towards identifying the global *minimum*. The overarching objective is to generate a novel activation function, labelled as 'm-arcsinh,' depicted in Eq. (7) in Fig. 2. This unique AF is meticulously designed and optimised to enhance the classification performance of both the MLP and the SVM.

To achieve this, we employ non-uniform mutation (as specified in Eq. (3)) and the process of crossover for each chromosome within our genetic population, with the aim of optimizing these functions. These steps are carried out iteratively, driven by a fitness function that guides the evolution of our functions:

$$x'_k = \begin{cases} x_k + \Delta(t, x_{ub} - x_k), & \textit{if } \alpha \textit{ random } \beta \textit{ is } 0 \\ x_k - \Delta(t, x_k - x_{lb}), & \textit{if } \alpha \textit{ random } \beta \textit{ is } 1 \end{cases} \tag{3}$$

where x_k is a chosen element, $x_i^t = \{x_1, x_2, \ldots, x_m\}$ is a chromosome with t generations, and lb and ub represent the lower and the upper bounds of x_k.

Eq. (3) delineates the mutation process undergone by each gene (hyperparameter) in the genetic population. This mutation is not uniform; its variation over time depends on the current gene's value, upper and lower bounds, and random factors (α random β). This dynamic variability ensures the genetic population explores a diverse range of hyperparameter values, facilitating the GA in a more effective search for optimal activation functions and hyperparameters.

This study's innovative expert-based multi-objective evolutionary framework stems from the imperative to optimise not only the activation functions but also their associated hyperparameters concurrently. This optimisation, depicted in Fig. 1 within the context of the chosen types of classifiers, employs a multi-objective evolutionary approach. The aim is to systematically explore and identify optimal hyperparameters enhancing the classification performance of both SVM and MLP simultaneously.

Guided by the fitness function through non-uniform mutation and crossover operations, the evolutionary process progressively refines and selects the most favourable genes. This iterative refinement culminates in the derivation of the 'm-arcsinh' as the optimal activation function, marking a significant milestone in this study's research endeavours.

2.5. The equation of the ActiGen-MOGA-derived activation and its decomposition for MLP and SVM

In the proposed multi-objective GA-based framework (ActiGen-MOGA), the fitness functions capturing the optimal underlying ML-based learning for the ANN- (MLP) and OSH-based (SVM) algorithms respectively are defined by the following parameters to be optimised, to generate a novel blended AF that could scale to both the MLP and the SVM for image classification:

- *inputs* is the matrix of the input image data to be classified.
- h denotes the number of hidden layers required in the MLP to transform *inputs* to facilitate learning [31].
- p is the power (any real value), degree or order of the polynomial in either families of s-shaped AF (in short, *AF*) or SVM-based kernel (in short, *kernel*) to improve their efficiency in discriminating amongst *inputs* into the expected or target classes [28].
- c is a factor of the selected family of AF (any real value) to refine the output from it and minimise its relative error [13].
- ν is one of the four mathematical operators to ensure appropriate approximation [4] and sensitivity of the outputs [5] from the generated AF based on the *inputs* considered.
- n_{max} is the maximum number of training iterations or epochs.

The optimised set of solutions ($\nu_{1/2}, p_{1/2}, c_{1/2}, AF, kernel$) was sought to maximise the *generalisation* in both the MLP and the SVM models, which is defined in this study by the weighted F1-score, thus leading to a novel AF with parameters from their blended learning process (as per requirements no. 1 and 2 listed at the start of the sub-section 2.1), quantified by the metrics mentioned in the sub-section 2.4. Such optimisation problems are solved separately but concurrently by a collaborative optimisation process whereby coordination is achieved via adaptive Lagrangian penalties. The training is stopped when the highest number of epochs before convergence (n_{max}) was achieved, i.e., when the target error (ε) was lower than a threshold, e.g., 10^{-5}.

Eq. 4 represents the ActiGen-MOGA-based optimisation methodology that concurrently maximises the generalisation of both the MLP and the SVM ($argmax\left(generalisation\left(\nu_{\frac{1}{2}}, h, p_{\frac{1}{2}}, c_{\frac{1}{2}}, n_{max}, AF\right)\right)$), whilst minimising the target error ($argmin\left(\varepsilon\left(\nu_{\frac{1}{2}}, h, p_{\frac{1}{2}}, c_{\frac{1}{2}}, n_{max}, AF\right)\right)$), thus yielding a novel optimal AF. Eq. 4 can be decomposed into Eq. 5 for the MLP alone and Eq. 6 for the SVM alone:

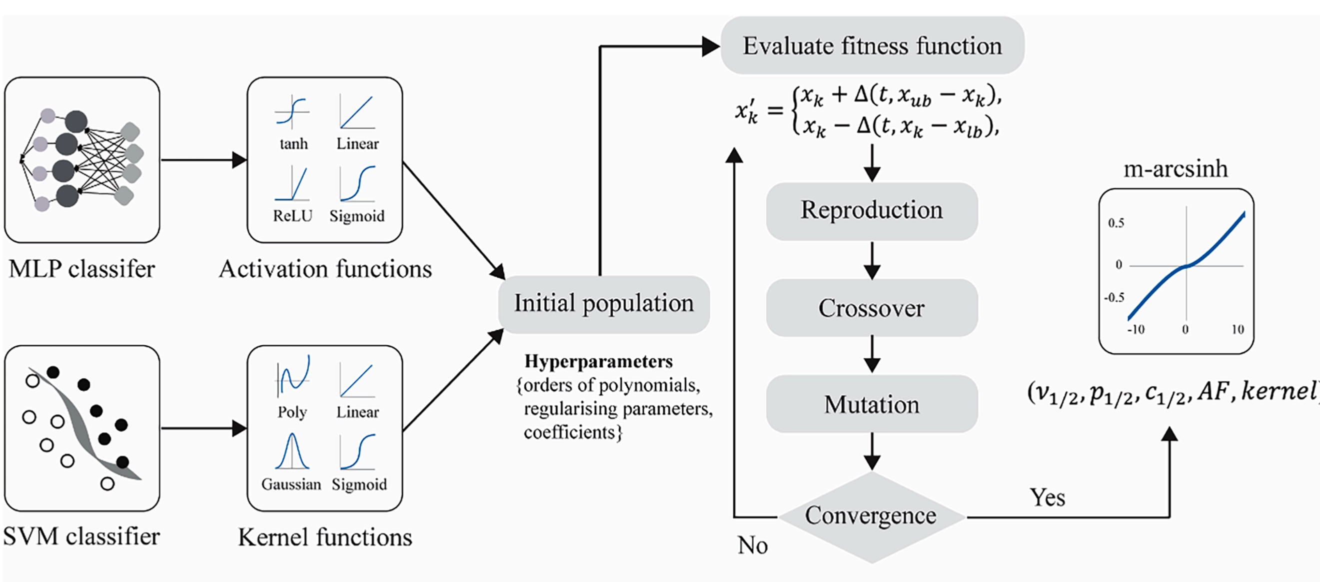

Fig. 1. The proposed multi-objective optimisation framework for a genetic algorithm-based activation function generation.

$$f = argmax\left(generalisation\left(\nu_{\frac{1}{2}}, h,\ p_{\frac{1}{2}}, c_{\frac{1}{2}}, n_{max}, AF\right)\right) \mid argmin\left(\varepsilon\left(\nu_{\frac{1}{2}}, h,\ p_{\frac{1}{2}}, c_{\frac{1}{2}}, n_{max}, AF\right)\right) \quad (4)$$

$$f_{MLP}(\nu_1, h,\ p_1, c_1,\ n_{max},\ AF) = \nu_1(h,\ p_1, c_1, inputs,\ AF) \quad (5)$$

$$f_{SVM}(\nu_2,\ p_2,\ c_2, n_{max}, kernel) = \nu_2(p_2,\ c_2, inputs, kernel), \quad (6)$$

$$\{p \in Z\}$$

$$\{c \in Z\}$$

$$\{n \in Z^+ : n_{max} | \varepsilon < 10^{-5}\}$$

| in Eq. 4 is the bitwise operator, which we used to indicate that Eq. 4 can be decomposed into Eq. 5 (for the MLP alone) and 6 (for the SVM alone) using a bitwise OR operation.

The m-arcsinh is derived from the evolution process by solving both fitness functions in Eqs. (5) and (6) and reaching a global *optimum* between them concurrently, thus being an optimal function for both SVM and MLP. The genes are derived, and the evolution is performed via non-uniform mutation and crossover as per Eq. (3). The fitness values are computed by solving Eq. (4), which derives a unique globally optimal solution whilst ensuring that generalisation is achieved.

The ActiGen-MOGA generates a novel AF (Eq. (7)) that concurrently accounts for a weighted interaction effect between the hyperbolic nature of the inverse hyperbolic sine function (as the optimal *AF* was the 'arcsinh' function, with $p_1 = 1$), suitable for the MLP, and the slightly non-linear characteristic of the square root function (the optimal *kernel*, with $p_2 = \frac{1}{2}$), appropriate for the SVM. As expected from the literature mentioned above [9,38,43,44], 1 was found the optimal value for *h*.

$$y = \text{arcsinh}(x) \times \frac{1}{3} \times \frac{1}{4} \times \sqrt{|x|} = \text{arcsinh}(x) \times \frac{1}{12} \times \sqrt{|x|} \quad (7)$$

With a higher weight ($c_1 = \frac{1}{3}$) attributed to the 'arcsinh' and a slightly lower one ($c_2 = \frac{1}{4}$) to the square root function, thus satisfying both requirements (**1**) and (2), the modified (m-) arcsinh (m-arcsinh) (Fig. 2) was generated via the ActiGen-MOGA framework as per Eq. (7):

The derivative of m-arcsinh (Fig. 3) is expressed as:

$$\frac{d_y}{d_x} = \sqrt{|x|} \times \frac{1}{12 \times \sqrt{x^2+1}} + \frac{x \times \text{arcsinh}(x)}{24 \times x^{3/2}} \quad (8)$$

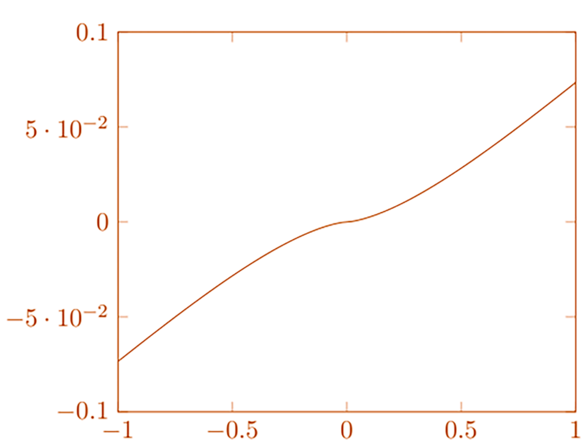

Fig. 2. The m-arcsinh activation function, generated by the proposed ActiGen-MOGA optimisation framework as the optimal function for both the Multi-Layer Perceptron (MLP) and the Support Vector Machine (SVM) for image classification.

3. Data and Modelling pipelines

Given the original motivation to target a generalised approach by the proposed evolutionary optimisation framework ('ActiGen-MOGA'), the datasets and related case scenarios reported in this section for the framework evaluation show diversity in terms of topic, data structures, and features, as well as the hyperparameters used for models' comparison.

3.1. Datasets and pre-processing

This study leveraged nine open-access datasets: five of them from the University of California at Irvine (UCI) ML Repository and four from the Python library 'scikit-learn', accounting for 22,136 images in total. Such datasets involved both binary (datasets no. 1 and 6) and multi-class classification (datasets no. 2-5, and 7-9) tasks. A comprehensive description of these datasets used is provided below.

The datasets from the UCI ML repository used in this study are as follows:

1. The 'Optical Recognition of Handwritten Digits' (OptDigits) datasets [22], to recognise handwritten digits (from 0 to 9), given 5,620 images in total and 64 features per each image from 43 people, 30 of which in the training data and the remaining 13 for testing.
2. The 'SPECTF' dataset [7], which has 267 images (80 images for training, 187 for testing) collected via a cardiac Single Proton Emission Computed Tomography (SPECT), describing whether each patient has a physiological or pathophysiological heart based on 44 features.
3. The 'Pen-based handwritten digits recognition' dataset [1], to recognise handwritten digits (from 0 to 9), drawn on a WACOM PL-100V pressure-sensitive tablet with an integrated LCD display and a cordless *stylus*, based on 250 images from 44 writers, 30 writers' images for training, 14 for testing.
4. The 'Smartphone-Based Recognition of Human Activities and Postural Transitions' dataset [56], which has recordings of 30 subjects carrying out activities and postural transitions whilst having a smartphone on their waist, along with embedded inertial sensors. This dataset has 561 features and 10,929 instances.

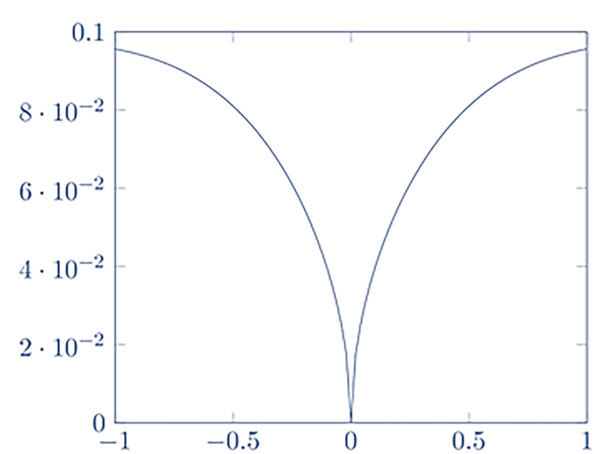

Fig. 3. The derivative of the m-arcsinh activation function.

5. The 'Myocardial infarction complications' dataset [57], which has clinical data including demographics, patients' history, indicators of cardiovascular and respiratory health, biomarkers from electrocardiogram measurements, and haematological and pharmacological markers. This dataset has 124 features and 1,700 instances.

The open-access datasets from scikit-learn leveraged in this study are the following:

1. The 'Breast cancer Wisconsin (diagnostic)' dataset [54], having 30 characteristics of cell *nuclei* from 569 digitised images of a fine needle aspirate of breast masses, to detect whether they correspond to either malignant or benign breast cancer.
2. The 'LFW people' dataset [19], having 13,233 JPEG photos of 5,749 famous people collected from the Internet, each of which is composed of 5,828 features [20], to identify the individual appearing on each photo.
3. The 'Handwritten Digits' dataset [2], to recognise handwritten digits (from 0 to 9), given about 180 images per class (1,797 images in total) and 64 features per each image.
4. The 'Olivetti faces' dataset [48] with 10 different 64×64 images regarding the faces of 40 different subjects to be classified, which were taken between April 1992 and April 1994 at the 'AT and T' Laboratories Cambridge. Such photos were captured against a dark homogeneous background at various times, with differing lighting conditions, facial expressions (open/closed eyes, smiling/not smiling) and details (glasses/no glasses). Subjects were in an upright, frontal position, with little side movement at a time.

The input data were randomised prior to performing classification. Then, the interquartile range-based method was used to remove outliers from the input data. Thereafter, cleaned data were standardised, i.e., transformed to z-scores to have an average of 0 and a standard deviation of 1, which characterise a standard normal distribution. Finally, min-max normalisation was applied on the transformed input data. For the case studies where data were not already provided in two separate partitions for training and testing (since the number of folds (k) or partitions in the k-fold cross-validation method leveraged should equal the number of input features [12,36]) k varied based on the image dataset considered. The experiments were conducted on an AMD E2-9000 Radeon R2 1.8 GHz processor, and 4 GB DDR4 RAM.

3.2. Modelling strategy

The 'MLP' and the 'SVM', implemented in the Python library 'scikit-learn' [47], are the two selected supervised learning algorithms representing ANN- and OSH-based classifiers respectively, with the following initial hyperparameters:

- For the MLP:
- Learning rate = 0.6 [32,33].
- Momentum = 0.8 [32,33].
- Random state = 1, which is a fixed random number to control random processes in the MLP and ensure the reproducibility of its training.
- 'Max iter' = 300, which is the maximum number of training iterations or epochs.
- For the SVM:
- Gamma = 0.001, which is the kernel coefficient for the AFs evaluated.
- Random state = 13.
- Class weight = 'balanced', thus setting the hyperparameter C by adjusting the weights to be inversely proportional to the class frequencies in the input data.

Along with guiding the generation of a novel AF to suit both the MLP and the SVM, their hyperparameters were optimised via the proposed multi-objective evolutionary framework leveraging GA, as described in sub-section 2.1. All hyperparameters were consistent when comparisons are performed using different activation functions to train and test the model in a specific dataset.

3.3. Metrics for evaluating classification performance

The same data pre-processing and encoding methodology in the sub-section 2.2 was adopted for all nine open-access datasets used to enable a fair comparison of the classification performance of both the SVM and the MLP regardless of the AF leveraged. This performance was assessed by the following two gold-standard criteria, including classification accuracy and reliability [17]:

1. Accuracy, measuring the predictive power of discriminating input data into the expected or target classes, quantified based on the testing set, which represents previously unseen or unknown data to the classification algorithm.
2. Reliability, which is the predictive capability of assigning a suitable degree of certainty on the classification outcomes.

For a binary classification task (datasets no. 1 and 6, described in the sub-section 2.2), the outcome from each image classified is either 'True' or 'False', thus, yielding four results [17], i.e., 'True Positive' (TP), 'False Positive' (FP), 'True Negative' (TN), and 'False Negative' (FN). For multi-class classification (datasets no. 2-5, and 7-9, described in the sub-section 2.2), the overall outcome can be obtained from each binary classification problem into which the multi-class classification task considered can be decomposed, as illustrated in Table 1 (e.g., a four-class classification problem).

To assess the performance brought by the optimal AF (m-arcsinh in Eq. (7) and Fig. 2 of the sub-section 2.1) for each image classification task as per the sub-section 2.2, generated via the proposed multi-objective evolutionary framework (ActiGen-MOGA in the sub-section 2.1), the test classification accuracy of the SVM and the MLP was evaluated. The following metrics assessed their reliability [17]: the precision, the sensitivity/recall, and the F-measure or F1-score, which is the harmonic mean between the precision and the recall.

In this study, the best-performing classifier was determined by its highest accuracy and reliability, whilst retaining a low computational training time in seconds on the same hardware in the sub-section 2.2 when considering different AFs, including the novel m-arcsinh function generated via the proposed ActiGen-MOGA evolutionary framework.

4. Results and Analysis

The ActiGen-MOGA evolutionary optimisation framework was employed to derive the novel blended activation function 'm-arcsinh' (as detailed in sub-section 2.1). This function was designed to scale effectively for both the MLP and the SVM models, which were subsequently evaluated on datasets that had been pre-processed and encoded (as explained in sub-section 2.2) via the classifiers described in sub-section 2.3. The performance of the SVM and the MLP, utilizing the 'm-arcsinh' function generated through the evolutionary framework ActiGen-MOGA as a kernel and activation function, respectively, was assessed based on classification accuracy, reliability, and computational cost (outlined in sub-section 2.4).

The main findings from these evaluations are summarised in Tables 2-4 and Figs. 4-5, while additional findings are available in the Appendix (Tables 5A-11A). Following an extensive analysis of classification and computational performance, 'm-arcsinh' demonstrated superiority over state-of-the-art activation functions and kernel functions when used with both ANN- and OSH-based classifiers, as outlined below

For the MLP:

Table 1
Confusion matrix for a four-class classification problem, wherein TP, TN, FP, and FN cases are the images involved in this study for each of the four binary classification tasks (no. 1-4) derived from the multi-class classification problem considered.

		Expected label$_1$		**Expected label$_2$**		**Expected label$_3$**		**Expected label$_4$**	
		True (T_1)	False (F_1)	True (T_2)	False (F_2)	True (T_3)	False (F_3)	True (T_4)	False (F_4)
Predicted class$_1$	Positive (P_1)	T_1P_1	F_1P_1						
	Negative (N_1)	T_1N_1	F_1N_1						
Predicted class$_2$	Positive (P_2)			T_2P_2	F_2P_2				
	Negative (N_2)			T_2N_2	F_2N_2				
Predicted class$_3$	Positive (P_3)					T_3P_3	F_3P_3		
	Negative (N_3)					T_3N_3	F_3N_3		
Predicted class$_4$	Positive (P_4)							T_4P_4	F_4P_4
	Negative (N_4)							T_4N_4	F_4N_4

Table 2
Image classification performance of a Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) with different kernel and activation functions respectively, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated on the 'Breast cancer Wisconsin (diagnostic)' dataset [54] in scikit-learn.

Classifier	Function	Training time (s)	Number of epochs N	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	**m-arcsinh (This study)**	0.007	2	0.97	0.97	0.97	0.97
SVM	Linear	1.312	112	0.98	0.98	0.98	0.98
SVM	Poly	311.706	29,876	0.98	0.98	0.98	0.98
SVM	Sigmoid	0.012	4	0.39	0.15	0.39	0.21
MLP	**m-arcsinh (This study)**	9.830	53	0.91	0.92	0.91	0.91
MLP	Identity	3.124	17	0.92	0.92	0.92	0.92
MLP	Logistic	3.638	21	0.92	0.92	0.92	0.92
MLP	tanh	3.568	20	0.90	0.90	0.90	0.90
MLP	ReLU	3.132	17	0.92	0.92	0.92	0.92

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; RBF: Radial Basis Function; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

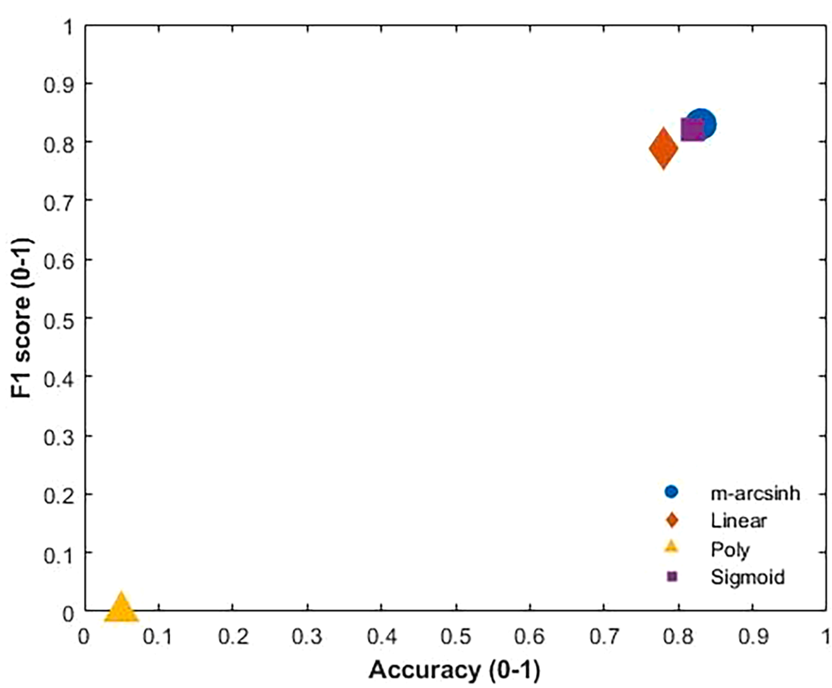

Fig. 4. Image classification performance of a Support Vector Machine (SVM) with different kernel functions, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated via test accuracy and F1-score (0-1, meaning 0-100%) on the 'LFW people' dataset [19] in scikit-learn.

- Achieved the best classification performance on 7 out of 9 evaluated datasets (Tables 3, 5A in the Appendix, 6A-10A in the Appendix, Figs. 4-5).
- Ranked second-highest in classification performance on 2 out of 9 datasets (Tables 2 and 11A).
- Attained the second-fastest training time and the second-lowest number of epochs on 1 out of 9 datasets (Table 7A in the Appendix).

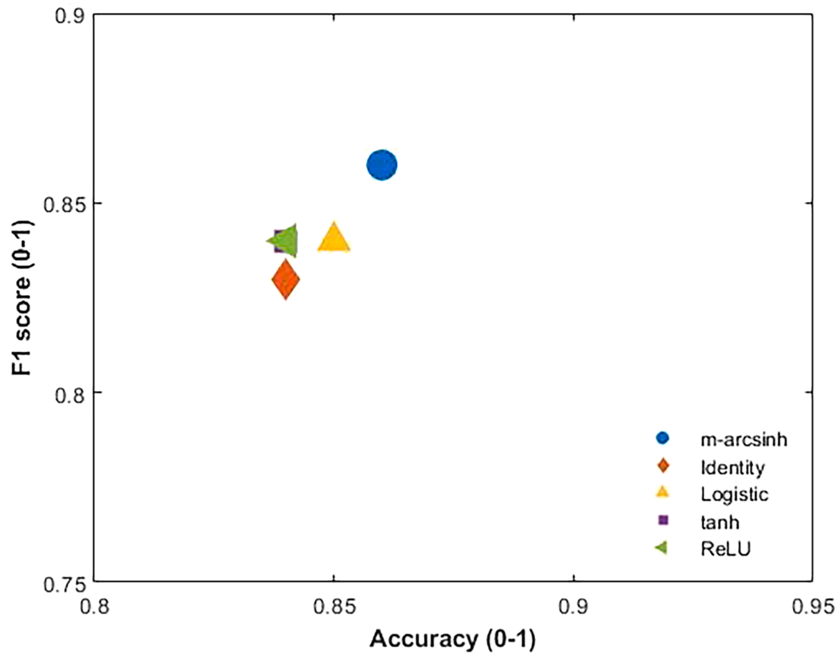

Fig. 5. Image classification performance of a Multi-Layer Perceptron (MLP) with different activation functions, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated via test accuracy and F1-score (0-1, meaning 0–100 %) on the 'LFW people' dataset [19] in scikit-learn.

Table 3
Image classification performance of a Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) with different kernel and activation functions respectively, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated on the 'LFW people' dataset [19] in scikit-learn.

Classifier	Function	Training time (s)	Number of epochs N	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	**m-arcsinh (This study)**	0.083	15	0.83	0.84	0.83	0.83
SVM	Linear	0.230	28	0.78	0.80	0.78	0.79
SVM	Poly	0.483	41	0.05	0.00	0.05	0.00
SVM	Sigmoid	0.570	47	0.82	0.83	0.82	0.82
MLP	**m-arcsinh (This study)**	7.101	33	0.86	0.86	0.86	0.86
MLP	Identity	6.225	28	0.84	0.84	0.84	0.83
MLP	Logistic	7.892	42	0.85	0.85	0.85	0.84
MLP	tanh	5.562	23	0.84	0.84	0.84	0.84
MLP	ReLU	4.755	21	0.84	0.84	0.84	0.83

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; RBF: Radial Basis Function; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

Table 4
Image classification performance of a Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) with different kernel and activation functions respectively, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated on the 'Handwritten Digits' dataset [2] in scikit-learn.

Classifier	Function	Training time (s)	Number of epochs N	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	**m-arcsinh (This study)**	0.037	6	0.95	0.95	0.95	0.95
SVM	Linear	0.033	5	0.93	0.93	0.93	0.93
SVM	Poly	0.043	8	0.95	0.95	0.95	0.95
SVM	Sigmoid	0.332	34	0.68	0.69	0.68	0.66
MLP	**m-arcsinh (This study)**	28.650	124	0.92	0.92	0.92	0.92
MLP	Identity	5.452	22	0.91	0.91	0.91	0.91
MLP	Logistic	14.182	61	0.94	0.94	0.94	0.93
MLP	tanh	7.258	37	0.93	0.93	0.93	0.93
MLP	ReLU	7.834	41	0.92	0.92	0.92	0.92

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; RBF: Radial Basis Function; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

- For the SVM:
- Achieved the best classification performance on 2 out of 9 datasets (Tables 3 and 4, Figs. 4-5).
- Ranked second-highest in classification performance on 6 out of 9 datasets (Tables 2, 5A-9A in the Appendix).
- Recorded the fastest training time and the lowest number of epochs on 5 out of 9 datasets (Tables 2, 3, 5A, 10A, and 11A in the Appendix).
- Secured both the highest classification performance and the fastest training time on 1 out of 9 datasets (Table 3, Figs. 4-5).
- Ranked second-fastest in training time and the second-lowest number of epochs on 2 out of 9 datasets (Tables 4 and 9A in the Appendix).

To provide an overall comparison of performance between 'm-arcsinh' and other activation functions, we primarily used the weighted F1-score as a key metric across the mentioned datasets (as displayed in Table 12A). In general, the weighted F1-score resulting from the use of 'm-arcsinh' in both the SVM and the MLP models demonstrates a highly competitive performance. These findings underscore the substantial advantages of 'm-arcsinh' in improving classification accuracy, reliability, and computational efficiency, positioning it as a valuable tool in the realm of machine learning-based image classification.

5. Discussion

5.1. Synthesis of salient findings and implications

In this study, the ActiGen-MOGA framework introduced a novel blended activation function (AF) named 'm-arcsinh' for image classification, marking a significant advancement. This framework not only showcased the potential of accelerating and semi-automating AF generation tailored to specific datasets and applications but also highlighted the exceptional reliability and computational efficiency of the 'm-arcsinh' function when compared to existing gold-standard AFs.

By reducing the dependence on subject matter experts and ensuring high algorithmic consistency, the ActiGen-MOGA framework demonstrated its practical utility. For instance, the Support Vector Machine (SVM) employing the 'm-arcsinh' AF outperformed the SVM with a polynomial kernel by an impressive margin of over 80 % in classification performance when tested on the 'LFW people' dataset, all while achieving faster training times. Similarly, the Multi-Layer Perceptron (MLP) exhibited superior performance when utilizing the AF generated by the ActiGen-MOGA framework, surpassing the performance of the conventional Rectified Linear Unit (ReLU) AF.

To solidify the framework's reliability and versatility, extensive evaluations were conducted across multiple datasets. The 'm-arcsinh' function, produced by the ActiGen-MOGA framework, emerged as a dependable and computationally efficient AF. It can be employed as a gold standard kernel and activation function for SVM and MLP algorithms, respectively.

In conclusion, the ActiGen-MOGA framework presents a transformative opportunity to enhance knowledge discovery and elevate the reliability and explainability of classification systems in the ever-evolving landscape of future computing systems.

5.2. Detailed discussion

The presented findings underscore the substantial success of the ActiGen-MOGA framework in the realm of image classification. The competitive classification results, especially those depicted in Figs. 4-5 and Tables 3, 5A-11A (found in the Appendix), reveal the framework's effectiveness. It introduced the 'm-arcsinh' blended activation function (AF), as illustrated in Eq. (7) and Figs. 2-3, showcasing its adaptability and scalability across both Optimal Separating Hyperplane (OSH)-based algorithms, represented by the Support Vector Machine (SVM), and Artificial Neural Networks (ANNs), represented by the Multi-Layer Perceptron (MLP).

These findings have several crucial implications. First, they highlight the immense potential of the ActiGen-MOGA framework in automating the generation of AFs for image classification. This process, which was once heavily reliant on time-consuming and potentially biased human expertise, can now benefit from a more objective and efficient approach. The framework's utilization of Genetic Algorithms (GAs) ensures the generation of optimal AFs, alleviating the challenge of identifying globally optimal AFs.

The reliability of 'm-arcsinh' is further substantiated by multiple metrics detailed in sub-section 2.4, surpassing many established AFs. For instance, when compared to the sigmoid AF, 'm-arcsinh' achieved a significantly higher F1-score of 0.97 for the SVM, while sigmoid scored only 0.21. Moreover, 'm-arcsinh' exhibits remarkable computational efficiency, evident in its substantially shorter training times. For example, with 'm-arcsinh,' SVM training took only 0.007 seconds compared to 1.312 seconds for SVM with a linear kernel and 311.706 seconds for SVM with a polynomial kernel.

These attributes collectively highlight the ActiGen-MOGA framework's consistent capacity to enhance image classification tasks for real-world applications. It notably reduces the dependence on human expertise in AF selection, relying on expert input solely for the initial AF family selection. Subsequently, the framework autonomously generates AFs tailored to multiple classifier categories. In the context of the largest image dataset (the 'LFW people' dataset), the SVM employing 'm-arcsinh' outperformed the SVM with a polynomial kernel by over 80 % in classification performance while maintaining the fastest training time. This demonstrates 'm-arcsinh's' efficacy in maximizing SVM margin width and enhancing linear separability more effectively than the polynomial kernel. Similarly, the MLP achieved a 3% performance improvement when using 'm-arcsinh' compared to the Rectified Linear Unit (ReLU) AF on the same dataset. These results provide further validation of the ActiGen-MOGA framework's accuracy, reliability, and computational efficiency.

The straightforward mathematical formulation of 'm-arcsinh' promotes faster knowledge discovery in the quest for more reliable AFs in image classification. Moreover, the ActiGen-MOGA framework enhances data pipeline explainability and classification reliability by generating AFs adaptable to multiple image classification algorithms. This development is poised to accelerate the adoption of machine learning-based decision support systems in translational applications within the computing systems community.

In conclusion, 'm-arcsinh,' derived through the ActiGen-MOGA framework, emerges as a new gold-standard kernel and AF for both SVM and MLP, readily available in scikit-learn. Its adoption promises significant impact and increased efficiency in the computing systems community, marking a substantial advancement in the field of image classification. However, it is essential to acknowledge potential limitations and sources of bias in the experimental design. This study has designed, developed, optimised, and validated the proposed AF on two families of classifiers, SVM and MLP. Thus, by design, the proposed unified AF (m-arcsinh) is not scalable to deep learning algorithms, e.g., Convolutional Neural Networks (CNNs), that are used to perform image classification, e.g., MNIST-digit, MNIST-fashion, CIFAR 10. In this case, other activation functions, such as hyper-sinh [44,45] and Quantum ReLU [46] are recommended, as intrinsically designed to scale with CNNs instead. To mitigate these, future research could explore larger and more diverse datasets and conduct cross-validation to ensure the robustness of the results. Additionally, considering the complexity of real-world applications, further investigation into the generalizability of 'm-arcsinh' across various domains and datasets would be beneficial.

6. Conclusions

In conclusion, our study presents a significant advancement in the field of machine learning-based image classification through the development and evaluation of the 'm-arcsinh' activation function, achieved via the ActiGen-MOGA evolutionary framework. The key findings of our research can be summarised as follows:

1. Improved accuracy and efficiency: Our 'm-arcsinh' activation function outperformed state-of-the-art alternatives in terms of accuracy, reliability, and computational efficiency. This enhancement was consistent across six out of nine diverse datasets, encompassing four distinct image classification tasks and utilizing two different classifiers: SVM and MLP.
2. Semi-automatic optimisation: The ActiGen-MOGA framework demonstrated its ability to semi-automatically optimise activation function families for selected classifiers. This reduced the need for extensive human intervention, making the framework highly accessible and adaptable within the computing systems community.
3. Translational impact: The 'm-arcsinh' activation function holds substantial potential for enhancing the generalisation and computational efficiency of image classification-aided decision-making processes. Its superior performance can contribute to more accurate and efficient real-world applications.
4. Transformative role of evolutionary algorithms: Our study underscores the transformative potential of evolutionary algorithms in tackling novel optimisation problems. The creation of 'm-arcsinh' serves as a prime example of how evolutionary approaches can lead to the development of more reliable activation functions, capable of scaling and generalising across various classifier types.

Looking ahead, several promising directions for future research and applications emerge:

1. Further optimisation: Continued research could focus on fine-tuning the 'm-arcsinh' activation function and exploring its potential for optimisation in specialised domains or specific classifiers. This could lead to even more tailored and efficient solutions.
2. Multi-modal data: Given the prevalence of multi-modal data in modern computing systems, extending the ActiGen-MOGA framework to handle diverse data types and hybrid classification scenarios is an intriguing avenue.
3. Additional real-world applications: The 'm-arcsinh' activation function and ActiGen-MOGA framework offer practical utility in fields such as healthcare, security, and autonomous systems. Future applications might involve medical image analysis, biometric security, and robotics, among others.
4. Interdisciplinary collaboration: Collaborative efforts with experts from various domains, including computer science, medicine, and engineering, can further explore the potential impact of the proposed framework in solving real-world problems and advancing scientific knowledge.

In summary, our research not only provides a novel AF but also showcases the broader possibilities of evolutionary algorithms in addressing complex optimisation challenges associated with deriving an optimal AF that can scale across different types of classifiers. By bridging the gap between diverse classifier categories, our framework holds the promise of supporting decision-making processes in the presence of noisy and multi-modal data, opening up new horizons for machine learning applications.

Funding

This research did not receive any funding.

CRediT authorship contribution statement

Luca Parisi: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Ciprian Daniel Neagu:** Writing – review & editing, Validation, Methodology, Investigation, Conceptualization. **Narrendar RaviChandran:** Writing – review & editing, Methodology, Conceptualization. **Renfei Ma:** Writing – review & editing, Visualization, Validation, Methodology, Investigation, Formal analysis, Conceptualization. **Felician Campean:** Writing – review & editing, Validation, Methodology, Investigation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The references to the datasets used in this study have been provided in the list of references for reproducibility.

Appendix

Table 5A
Image classification performance of a Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) with different kernel and activation functions respectively, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated on the 'Olivetti faces dataset [48] in scikit-learn.

Classifier	Function	Training time (s)	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	**m-arcsinh (This study)**	0.143	0.91	0.91	0.91	0.90
SVM	Linear	1.124	0.99	0.99	0.99	0.99
SVM	Poly	1.071	0.85	0.85	0.85	0.83
SVM	Sigmoid	1.364	0.00	0.00	0.00	0.00
MLP	**m-arcsinh (This study)**	105.341	0.75	0.78	0.75	0.75
MLP	Identity	109.109	0.75	0.78	0.75	0.75
MLP	Logistic	94.982	0.75	0.78	0.75	0.75
MLP	tanh	103.759	0.75	0.78	0.75	0.75
MLP	ReLU	104.581	0.75	0.78	0.75	0.75

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

Table 6A
Image classification performance of a Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) with different kernel and activation functions respectively, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated on the 'OptDigits' dataset [22] from the University California Irvine (UCI) Machine Learning repository.

Classifier	Function	Training time (s)	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	**m-arcsinh (This study)**	0.232	0.97	0.97	0.97	0.97
SVM	Linear	0.175	0.96	0.96	0.96	0.96
SVM	Poly	0.180	0.97	0.98	0.97	0.97
SVM	Sigmoid	2.384	0.71	0.75	0.71	0.72
MLP	**m-arcsinh (This study)**	53.586	0.98	0.98	0.98	0.98
MLP	Identity	9.572	0.98	0.98	0.98	0.98
MLP	Logistic	27.457	0.98	0.98	0.98	0.98
MLP	tanh	15.750	0.98	0.98	0.98	0.98
MLP	ReLU	14.254	0.98	0.98	0.98	0.98

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

Table 7A
Image classification performance of a Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) with different kernel and activation functions respectively, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated on the 'SPECTF' dataset [7] from the University California Irvine (UCI) Machine Learning repository.

Classifier	Function	Training time (s)	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	**m-arcsinh (This study)**	0.004	0.91	0.91	0.91	0.91
SVM	Linear	0.004	1.00	1.00	1.00	1.00
SVM	Poly	0.003	1.00	1.00	1.00	1.00
SVM	Sigmoid	0.003	0.50	0.25	0.50	0.33
MLP	**m-arcsinh (This study)**	0.047	0.54	0.76	0.54	0.41
MLP	Identity	0.080	0.54	0.76	0.54	0.41
MLP	Logistic	0.043	0.54	0.76	0.54	0.41
MLP	tanh	0.078	0.54	0.76	0.54	0.41
MLP	ReLU	0.096	0.54	0.76	0.54	0.41

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

Table 8A
Image classification performance of a Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) with different kernel and activation functions respectively, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated on the 'Pen-based handwritten digits recognition' dataset [1] from the University California Irvine (UCI) Machine Learning repository.

Classifier	Function	Training time (s)	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	**m-arcsinh (This study)**	0.728	0.99	0.99	0.99	0.99
SVM	Linear	4.651	0.99	0.99	0.99	0.99
SVM	Poly	0.196	1.00	1.00	1.00	1.00
SVM	Sigmoid	3.024	0.13	0.06	0.13	0.06
MLP	**m-arcsinh (This study)**	17.736	1.00	1.00	1.00	1.00
MLP	Identity	18.692	1.00	1.00	1.00	1.00
MLP	Logistic	18.268	1.00	1.00	1.00	1.00
MLP	tanh	20.490	1.00	1.00	1.00	1.00
MLP	ReLU	19.362	1.00	1.00	1.00	1.00

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

Table 9A
Image classification performance of a Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) with different kernel and activation functions respectively, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated on the 'Smartphone-Based Recognition of Human Activities and Postural Transitions' dataset [56] from the University California Irvine (UCI) Machine Learning repository.

Classifier	Function	Training time (s)	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	**m-arcsinh (This study)**	0.920	0.92	0.92	0.92	0.92
SVM	Linear	0.727	0.95	0.95	0.95	0.95

(*continued on next page*)

Table 9A (*continued*)

Classifier	Function	Training time (s)	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	Poly	8.408	0.76	0.78	0.76	0.76
SVM	Sigmoid	4.943	0.87	0.88	0.87	0.87
MLP	**m-arcsinh (This study)**	4.816	0.88	0.90	0.88	0.88
MLP	Identity	3.126	0.86	0.89	0.86	0.85
MLP	Logistic	3.129	0.29	0.09	0.29	0.14
MLP	tanh	1.400	0.29	0.09	0.29	0.13
MLP	ReLU	4.645	0.17	0.03	0.17	0.05

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

Table 10A
Image classification performance of a Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) with different kernel and activation functions respectively, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated to predict Atrial Fibrillation on the 'Myocardial infarction complications' dataset [57] from the University California Irvine (UCI) Machine Learning repository.

Classifier	Function	Training time (s)	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	**m-arcsinh (This study)**	0.028	0.63	0.87	0.63	0.70
SVM	Linear	3.050	0.72	0.86	0.72	0.77
SVM	Poly	9.171	0.77	0.85	0.77	0.80
SVM	Sigmoid	0.054	0.10	0.01	0.10	0.02
MLP	**m-arcsinh (This study)**	1.029	0.89	0.85	0.89	0.85
MLP	Identity	0.048	0.89	0.79	0.89	0.84
MLP	Logistic	0.167	0.89	0.79	0.89	0.84
MLP	tanh	0.083	0.89	0.79	0.89	0.84
MLP	ReLU	1.337	0.89	0.79	0.89	0.84

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

Table 11A
Image classification performance of a Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) with different kernel and activation functions respectively, including the m-arcsinh generated via the proposed ActiGen-MOGA framework. This performance was evaluated to predict Pulmonary Oedema on the 'Myocardial infarction complications' dataset [57] from the University California Irvine (UCI) Machine Learning repository.

Classifier	Function	Training time (s)	Accuracy (0-1)	Weighted precision (0-1)	Weighted recall (0-1)	Weighted F1-score (0-1)
SVM	**m-arcsinh (This study)**	0.035	0.73	0.90	0.73	0.79
SVM	Linear	5.542	0.76	0.90	0.76	0.81
SVM	Poly	14.911	0.83	0.88	0.83	0.85
SVM	Sigmoid	0.070	0.08	0.01	0.08	0.01
MLP	**m-arcsinh (This study)**	1.485	0.92	0.92	0.92	0.92
MLP	Identity	0.139	0.84	0.90	0.84	0.87
MLP	Logistic	0.148	0.93	0.87	0.93	0.90
MLP	tanh	0.091	0.07	0.00	0.07	0.01
MLP	ReLU	0.679	0.93	0.87	0.93	0.90

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

Table 12A
The comparison of the weighted F1-score resulting from the m-arcsinh and other activation functions on the 9 datasets evaluated. The last two experiments come from the same dataset but using a different target or myocardial infarction complication to predict.

Dataset	SVM				MLP				
	m-arcsinh	Linear	Poly	Sigmoid	m-arcsinh	Identity	Logistic	tanh	ReLU
'Breast cancer Wisconsin' [54]	**0.97**	0.98	0.98	0.21	**0.91**	0.92	0.92	0.90	0.92
'LFW people' [19]	**0.83**	0.79	0.00	0.82	**0.86**	0.83	0.84	0.84	0.83
'Handwritten Digits' [2]	**0.95**	0.93	0.95	0.66	**0.92**	0.91	0.93	0.93	0.92
'Olivetti faces' [48]	**0.90**	0.99	0.83	0.00	**0.75**	0.75	0.75	0.75	0.75
'OptDigits' [22]	**0.97**	0.96	0.97	0.72	**0.98**	0.98	0.98	0.98	0.98
'SPECTF' [7]	**0.91**	1.00	1.00	0.33	**0.41**	0.41	0.41	0.41	0.41
'Pen-based handwritten digits recognition' [1]	**0.99**	0.99	1.00	0.06	**1.00**	1.00	1.00	1.00	1.00
'Smartphone-Based Recognition of Human Activities and Postural Transitions' [56]	**0.92**	0.95	0.76	0.87	**0.88**	0.85	0.14	0.13	0.05
Atrial Fibrillation on the 'Myocardial infarction complications' [57]	**0.70**	0.77	0.80	0.02	**0.85**	0.84	0.84	0.84	0.84
Pulmonary Oedema on the 'Myocardial infarction complications' dataset [57]	**0.79**	0.81	0.85	0.01	**0.92**	0.87	0.90	0.01	0.90

SVM: Support Vector Machine; MLP: Multi-Layer Perceptron; Linear: Linear kernel; tanh: hyperbolic tangent sigmoid; ReLU: Rectified Linear Unit.

References

[1] E. Alpaydin, F. Alimoglu, Pen-based recognition of handwritten digits data set. University of California, Irvine, Mach. Learn. Reposit. Irvine Univ. California 4 (2) (1998).
[2] E. Alpaydin, C. Kaynak, Optical recognition of handwritten digits data set. UCI Mach. Learn. Reposit., 1998.
[3] Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., … & Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. *arXiv preprint arXiv:1309.0238*.
[4] T. Chen, H. Chen, Universal approximation to non-linear operators by neural networks with arbitrary activation functions and its application to dynamical systems, IEEe Trans. Neural Netw. 6 (4) (1995) 911–917.
[5] D. Chu, N.R. Zabet, B. Mitavskiy, Models of transcription factor binding: sensitivity of activation functions to model assumptions, J. Theor. Biol. 257 (3) (2009) 419–429.
[6] I. Revin, V.A. Potemkin, N.R. Balabanov, N.O. Nikitin, Automated machine learning approach for time series classification pipelines using evolutionary optimisation, Knowl. Based. Syst. 268 (2023) 110483.
[7] Cios, K. J., Kurgan, L. A., and Goodenday, L. S. (2001). Spectf heart data set - UCI machine learning repository, 2001. URL https://archive.ics.uci.edu/ml/datasets/SPECTF+Heart.
[8] Corinna Cortes, Vladimir Vapnik, Support-vector networks, Mach. Learn. 20 (3) (1995) 273–297.
[9] G. Cybenko, Approximation by superpositions of a sigmoidal function, Math. Control Signals Syst. 2 (4) (1989) 303–314.
[10] K. Davoudi, P. Thulasiraman, Evolving convolutional neural network parameters through the genetic algorithm for the breast cancer classification problem, Simulation (2021) 0037549721996031.
[11] S. Ding, C. Su, J. Yu, An optimizing BP neural network algorithm based on genetic algorithm, Artif. Intell. Rev. 36 (2) (2011) 153–162.
[12] K. Duan, S.S. Keerthi, A.N. Poo, Evaluation of simple performance measures for tuning SVM hyperparameters, Neurocomputing 51 (2003) 41–59.
[13] J. Faraone, M. Kumm, M. Hardieck, P. Zipf, X. Liu, D. Boland, P.H. Leong, Addnet: Deep neural networks using fpga-optimised multipliers, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 28 (1) (2019) 115–128.
[14] H. Faris, I. Aljarah, S. Mirjalili, Training feedforward neural networks using multi-verse optimiser for binary classification problems, Appl. Intell. 45 (2) (2016) 322–332.
[15] T. Goel, R. Murugan, S. Mirjalili, D.K. Chakrabartty, OptCoNet: an optimised convolutional neural network for an automatic diagnosis of COVID-19, Appl. Intell. 51 (3) (2021) 1351–1366.
[16] G.S.D.S. Gomes, T.B. Ludermir, Optimisation of the weights and asymmetric activation function family of neural network for time series forecasting, Expert. Syst. Appl. 40 (16) (2013) 6438–6446.
[17] G.L. Grunkemeier, R. Jin, Receiver operating characteristic curve analysis of clinical risk models, Ann. Thorac. Surg. 72 (2) (2001) 323–326.
[18] J.V. Hansen, J.B. McDonald, Some experimental evidence on the performance of GA-designed neural networks, J. Exp. Theor. Artif. Intell. 13 (3) (2001) 307–321.
[19] G.B. Huang, M. Mattar, T. Berg, E. Learned-Miller, Labeled faces in the wild: a database for studying face recognition in unconstrained environments. Workshop on faces in 'Real-Life' Images: detection, alignment, and recognition, 2008.
[20] G.B. Huang, E. Learned-Miller, Labeled faces in the wild: Updates and new reporting procedures, Dept. Comput. Sci., Univ. Massachusetts Amherst Amherst MA, USA Tech. Rep. 14 (003) (2014).
[21] Jacot, A., Gabriel, F., & Hongler, C. (2018). Neural tangent kernel: Convergence and generalization in neural networks. *arXiv preprint arXiv:1806.07572*.
[22] C. Kaynak, Methods of combining multiple classifiers and their applications to handwritten digit recognition, Bogazici University, 1995. Master's thesis.
[23] S.S. Keerthi, S. Sundararajan, K.W. Chang, C.J. Hsieh, C.J. Lin, A sequential dual method for large scale multi-class linear SVMs, in: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2008, pp. 408–416.
[24] V. Kunc, J. Kléma, On transformative adaptive activation functions in neural networks for gene expression inference, PLoS. ONE 16 (1) (2021) e0243915.
[25] Z. Xu, Z. Wang, J. Li, T. Jin, X. Meng, S. Gao, Dendritic neuron model trained by information feedback-enhanced differential evolution algorithm for classification, Knowl. Based. Syst. 233 (2021) 107536.
[26] S. Li, H. Wu, D. Wan, J. Zhu, An effective feature selection method for hyperspectral image classification based on genetic algorithm and support vector machine, Knowl. Based. Syst. 24 (1) (2011) 40–48.
[27] H.T. Lin, C.J. Lin, A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods, Neural Comput. 3 (1-32) (2003) 16.
[28] E. López-Rubio, F. Ortega-Zamorano, E. Domínguez, J. Muñoz-Pérez, Piecewise polynomial activation functions for feedforward neural networks, Neural Process. Lett. 50 (1) (2019) 121–147.
[29] O.L. Mangasarian, D.R. Musicant, Lagrangian support vector machines, J. Mach. Learn. Res. 1 (Mar) (2001) 161–177.
[30] S.M. Mousavi, S. Abdullah, S.T.A. Niaki, S. Banihashemi, An intelligent hybrid classification algorithm integrating fuzzy rule-based extraction and harmony search optimisation: medical diagnosis applications, Knowl. Based. Syst. 220 (2021) 106943.
[31] G. Panchal, A. Ganatra, Y.P. Kosta, D. Panchal, Behaviour analysis of multilayer perceptrons with multiple hidden neurons and hidden layers, Int. J. Comput. Theor. Eng. 3 (2) (2011) 332–337.
[32] L. Parisi, Exploiting kinetic and kinematic data to plot cyclograms for managing the rehabilitation process of BKAs by applying neural networks, Int. J. Biomed. Biol. Eng. 8 (10) (2014) 664–668.
[33] L. Parisi, Neural networks for distinguishing the performance of two hip joint implants on the basis of hip implant side and ground reaction force, Int. J. Med. Heal. Pharm. Biomed. Eng. 8 (10) (2014) 659–663.
[34] L. Parisi, P.R. Biggs, G.M. Whatling, C.A. Holt, A novel comparison of artificial intelligence methods for diagnosing knee osteoarthritis, in: XXV Congress of the International Society of Biomechanics, 2015, pp. 1227–1229.
[35] L. Parisi, M.L. Manaog, Preliminary validation of the lagrangian support vector machine learning classifier as clinical decision-making support tool to aid prediction of prognosis in patients with hepatitis, in: The 16th International Conference on Biomedical Engineering, National University of Singapore (NUS, 2016.
[36] L. Parisi, M.L. Manaog, A minimum viable machine learning-based speech processing solution for facilitating early diagnosis of parkinson's disease, in: MATLAB Conference 2017, 2017.
[37] L. Parisi, M.L. Manaog, The importance of selecting appropriate k-fold cross-validation and training algorithms in improving postoperative discharge decision-making via artificial intelligence, 2017 AUT Math. Sci. Sympos. 1 (1) (2017) 16.
[38] L. Parisi, N. RaviChandran, M.L. Manaog, Decision support system to improve postoperative discharge: a novel multi-class classification approach, Knowl. Based. Syst. 152 (2018) 1–10.
[39] L. Parisi, N. RaviChandran, Genetic algorithms and unsupervised machine learning for predicting robotic manipulation failures for force-sensitive tasks, in: 2018 4th International Conference on Control, Automation and Robotics (ICCAR), IEEE, 2018, pp. 22–25.
[40] L. Parisi, arXiv preprint, 2020.
[41] Parisi, L., & RaviChandran, N. (2020a). Evolutionary feature transformation to improve prognostic prediction of hepatitis. *Knowl. Based. Syst.*, 200, 106012.
[42] L. Parisi, N. RaviChandran, Evolutionary denoising-based machine learning for detecting knee disorders, Neural Process. Lett. 52 (3) (2020) 2565–2581.
[43] Parisi, L., Neagu, D., Ma, R., & Campean, F. (2020a). QReLU and m-QReLU: two novel quantum activation functions to aid medical diagnostics. *arXiv preprint arXiv:2010.08031*.
[44] Parisi, L., Ma, R., RaviChandran, N., & Lanzillotta, M. (2020b). hyper-sinh: an accurate and reliable function from shallow to deep learning in tensorflow and keras. *arXiv preprint arXiv:2011.07661*.
[45] L. Parisi, R. Ma, N. RaviChandran, M. Lanzillotta, hyper-sinh: An accurate and reliable function from shallow to deep learning in TensorFlow and Keras, Mach. Learn. Appl. 6 (2021) 100112.
[46] L. Parisi, D. Neagu, R. Ma, F. Campean, Quantum ReLU activation for convolutional neural networks to improve diagnosis of Parkinson's disease and COVID-19, Expert. Syst. Appl. 187 (2022) 115892.
[47] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, J. Vanderplas, Scikit-learn: machine learning in python, J. Mach. Learn. Res. 12 (Oct) (2011) 2825–2830. *URL*, http://jmlr.org/papers/v12/pedregosa11a.html.
[48] Roweis, S. Olivetti faces data set (2017). URL https://cs.nyu.edu/∼roweis/data.html.
[49] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors, Nature 323 (6088) (1986) 533–536.
[50] W.M. Shaban, A.H. Rabie, A.I. Saleh, M.A. Abo-Elsoud, A new COVID-19 patients detection strategy (CPDS) based on hybrid feature selection and enhanced KNN classifier, Knowl. Based. Syst. 205 (2020) 106270.
[51] G. van den Burg, P. Groenen, GenSVM: a generalised multi-class support vector machine, J. Mach. Learn. Res. 17 (2016) 1–42.
[52] D. Venkatesan, K. Kannan, R. Saravanan, A genetic algorithm-based artificial neural network model for the optimisation of machining processes, Neural Comput. Applic. 18 (2) (2009) 135–140.
[53] R. Vert, J.P. Vert, B. Schölkopf, Consistency and convergence rates of one-class SVMs and related algorithms, J. Mach. Learn. Res. 7 (5) (2006).
[54] Wolberg, W. H., Street, W. N. and Mangasarian, O. L. (1995). Breast cancer Wisconsin (diagnostic) data set - UCI machine learning repository. URL https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic).
[55] D.A. Otchere, T.O.A. Ganat, R. Gholami, S. Ridha, Application of supervised machine learning paradigms in the prediction of petroleum reservoir properties: Comparative analysis of ANN and SVM models, J. Petrol. Sci. Eng. 200 (2021) 108182.
[56] J.L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, D. Anguita, Transition-aware human activity recognition using smartphones, Neurocomputing 171 (2016) 754–767.
[57] S.E. Golovenkin, J. Bac, A. Chervov, E.M. Mirkes, Y.V. Orlova, E. Barillot, A. Zinovyev, Trajectories, bifurcations, and pseudo-time in large clinical datasets: applications to myocardial infarction and diabetes data, Gigascience 9 (11) (2020) giaa128.
[58] Y. Jia, J. McDermid, T. Lawton, I. Habli, The role of explainability in assuring safety of machine learning in healthcare, IEEE Trans. Emerg. Top. Comput. 10 (4) (2022) 1746–1760.