Konya Mühendislik Bilimleri Dergisi, c. 10, s. 3, 762-773, 2022
Konya Journal of Engineering Sciences, v. 10, n. 3, 762-773, 2022
ISSN: 2667-8055 (Electronic)
DOI: 10.36306/konjes.1077177

A SYSTEMATIC AND EFFICIENT INPUT SELECTION METHOD FOR ARTIFICIAL NEURAL NETWORKS USING MIXED-INTEGER NONLINEAR PROGRAMMING

[1]Hasan SILDIR, [2]Erdal AYDIN

[1]Gebze Technical University, Chemical Engineering Department, Kocaeli, TURKIYE
[2]Koc University, Chemical and Biological Engineering Department, Istanbul, TURKIYE
[1]hasansildir@gtu.edu.tr, [2]eaydin@ku.edu.tr

(Geliş/Received: 22.02.2022; Kabul/Accepted in Revised Form: 04.08.2022)

ABSTRACT: Selection of input variables of the empirical models has vital effect on the prediction performance, reduced overfitting and reduced computational load. Various trials and error and sequential methods in the literature to deal with input selection for artificial neural networks (ANNs). However, these methods are not considered as automatic and systematic. This study proposes a novel and efficient mixed integer nonlinear programming-based approach to handle optimal input selection and the ANN training simultaneously for classification problems. Such selection uses binary (0-1) variables to represent the presence of the input variables and trains traditional continuous network weights simultaneously. Two classification case studies are given to demonstrate the advantages by using widely used data sets and statistical measures. The first data set is related to the characterization of the type of a tumor related to breast cancer, the second data set is about predicting the type of a biotechnological product using different features, the last one is related to heart failure prediction. Results show that better test performance can be achieved with optimally selected inputs, resulting in reduced overfitting. The proposed approach delivers a significant advantage during the design and training of the ANNs and is also applicable to other empirical models.

Keywords: *Input selection, Artificial neural networks, Mixed-integer programming*

Kesikli ve Sürekli Optimizasyon Kullanarak Yapay Sinir Ağları için Sistematik Girdi Seçimi Yöntemi

ÖZ: Ampirik modellerin girdi değişkenlerinin seçimi, tahmin performansı, azaltılmış fazla uydurma ve hesaplama yükünün azaltılması üzerinde önemli etkiye sahiptir. Literatürde yapay sinir ağları (YSA) için girdi seçimi ile ilgili çeşitli deneme yanılma yöntemleri mevcuttur ancak bu metodlar sistematik ve otomatik olarak kabul edilmemektedir. Bu çalışma, sınıflandırma problemleri için optimal girdi seçimi ve YSA eğitimini aynı anda ele almak için yeni ve verimli bir karma tamsayılı doğrusal olmayan programlama tabanlı bir yaklaşım önermektedir. Bu seçim, girdi değişkenlerinin varlığını temsil etmek için ikili (0-1) değişkenleri kullanır ve geleneksel sürekli ağ ağırlıklarını veya parametrelerini aynı anda eğitir. Yaygın olarak kullanılan veri setleri ve istatistiksel ölçümler kullanarak avantajları göstermek amacıyla üç sınıflandırma vaka çalışması sunulmuştur. Birinci veri seti meme kanseri ile ilgili tümörün tipin-in karakterizasyonu ile ilgili olup, ikinci veri seti ise farklı özellikler kullanılarak bir biyoteknolojik ürünün tipinin tahmin edilmesi ile ilgilidir, son veri seti ise kalp sağlığı ile ilgilidir. Sonuçlar, optimal olarak seçilen girdiler ile düşük fazla uydurma sayesinde daha iyi test performansının elde edilebileceğini göstermektedir. Önerilen yaklaşım, YSA'ların tasarımı ve eğitimi sırasında önemli bir avantaj sağlar ve diğer ampirik modellere de uygulanabilir.

Anahtar Kelimeler: *Girdi Seçimi, Yapay sinir ağları, Kesikli ve sürekli optimizasyon*

1. INTRODUCTION

Modeling of complex processes are primarily handled through mechanistic or empirical models. The mechanistic models contain mathematical expressions with the actual physical dynamics. Thus, such models have a more trustable nature to extrapolation. On the other hand, the derivation of such models is a challenging task due to experimental challenges and the theoretical issues related to knowledge on underlying mechanism.

Empirical models, with a high number of varieties, deliver an alternative to mechanistic models once the data are abundant and the processing conditions are similar or within an acceptable extrapolation region although they are constructed on tailored black-box mathematical formulations with no physical background.

ANNs are one type of empirical models and have obtained gradually obtained more attention over the past decades both due to increased computational power and data availability. Feedforward ANNs propagate the input information, which is represented by a vector for a particular sample, to succeeding layers through linear and nonlinear operations, which incorporate the elements of input vector with different weights. Traditionally, all ANN variables interact in a fully connected sense.

ANNs have found different application areas both due to their flexibility to represent complex interactions and theoretical modifications resulting in a different terminology despite underlying logic is based on regression. In (Mutlu and Yucel 2018), artificial intelligence based methods have been used for the prediction of biomass gasification reaction. (Akdag, Komur, and Ozguc 2009), used ANNs for the calculation of heat transfer related parameters from an experimental apparatus. Furthermore, ANNs have found significant applications in time-series data both from the manufacturing processes and economics. In (Kocak and Un 2014), gold price estimations were considered. An application on environmental studies can be found in (Yetilmezsoy, Ozkaya, and Cakmakci 2011). Electricity consumption using ANNs is applied by (Azadeh, Ghaderi, and Sohrabkhani 2008).

A major problem with the ANN is the scalability and the management in real time when high number of inputs are candidates for the model development since measurements and advanced sensor technology provide a significant amount of different data, some of which are redundant or correlated. There are various trial and error and sequential input selection methods for the ANNs (Castellano and Fanelli 2000; Leahy, Kiely, and Corcoran 2008; Verikas and Bacauskiene 2002) combine the tools from optimization and statistics.

In theory, higher number of inputs, especially when they are redundant or do not carry additional statistical insight, contribute to famous overfitting problem by introducing more weights to the architecture and bringing about significant computational load with parameter identifiability issues (Schittkowski 2007). A representative input subset selection has been a focus for the development of more efficient and accurate predicting ANN architectures with improved computational performance.

In (Verikas and Bacauskiene 2002), an input reduction method is implemented on publicly available benchmark problems and provides a similar performance with smaller input subspace. The input selection might in turn delivering a better or a similar performance in the test data despite reduced training performance. Such a performance drop in training is theoretically intuitive as less parameters are included in the optimization problem with fewer inputs. However, the impact on the test instances is beneficial for many cases (Ledesma et al. 2008; Sildir, Aydin, and Kavzoglu 2020).

There are various input selection algorithms in the literature (Castellano and Fanelli 2000; Van De Wal and De Jager n.d.). Those methods usually handle the input reduction in a computationally simplified domain by excluding the non-convex and non-smooth optimization problem, focusing on statistical significance among variables or straightforward interactions between inputs and outputs. Next, the resulting input subset is used for the development of ANNs. A major theoretical challenge with such approaches is the compatibility of the inputs for the ANN development although they are obtained from a simplified set of interactions rather than the actual ANN formulation during the selection. In addition, those inputs are selected using a sequential manner, which requires the successive and recursive process in which only one input is removed per iteration. Thus, any theoretical potential from the co-existence of

inputs can not be exploited due to the heuristic nature of the approach. On the other hand, such sequential input selection algorithms have found significant applications (Rückstieß, Osendorfer, and Smagt 2011) due to its practical usage and satisfactory performances once the data do not contain significant nonlinearity and complexity. With many variations (Aha and Bankert 1996), the cross-validation score is a popular approach to eliminate or add an input (Ferri et al. 1994). The method is also used in this study for the comparison purposes.

One more typical solution to cope with overfitting is to include a regularization term to penalize larger values of ANN parameters during the NLP (Non-linear programming) solution, which unfortunately cannot regularize the hyper parameters (Manngård, Kronqvist, and Böling 2018). Dropout regularization is an alternative and efficient method to include structure detection element into ANNs, using random sampling based techniques (Poernomo and Kang 2018). Another recent advancement related to feature selection includes Bayesian optimization-based methods. These methods are not non-parallelizable, but also often converge to suboptimal local solutions, which in turn brings about poor performance for feature selection. On the other hand, efficient decomposition algorithms are present for larger search spaces, integrating the derivate-based and blackbox optimization approaches for the solution of mixed integer nonlinear programming (MINLP), (Diaz et al. 2017; Feurer and Hutter 2019; Stamoulis et al. 2018).

This study focuses on the development of an MINLP formulation for the simultaneous design and the training of feedforward ANNs by representing the selection of the inputs by a binary decision variable in addition to traditional continuous ANN weights. This approach obtains the structure and the weights of the ANN automatically and simultaneously, unlike sequential feature algorithms. Thus, resulting best features are selected considering the ANN architecture, in addition to traditional concerns such as accuracy, and ensures a satisfactory performance with the subset. Furthermore, in theory, MINLP problems can be solved to global optimum in theory, guaranteeing to achieve the best possible feature selection (Sahinidis 1996). Finally, the decomposable structure of the resulting mathematical optimization formulations also increases the potential of the proposed methods for large-scale data sets, unlike the cutting-edge Bayesian optimization or sequential feature selection methods.

The rest of the paper is organized as follows: Section 2 includes the theoretical background of the approach. Section 3 provides the results and comparison to ANNs with all inputs. The last section concludes the study.

2. Methodology

2.1 Theoretical Background

Traditional feedforward ANN applications comprise of fully connected networks where all inputs, hidden layer neurons and outputs are connected completely, in a fully-connected manner. These ANNs are typically defined as follows:

$$y = f_{OL}(W_{OL} \cdot f_{HL}(W_{HL} \cdot u + B_{HL}) + B_{OL}) \tag{1}$$

where y is the vector of outputs; W_{HL} and B_{HL} are the matrices of hidden layer weights and bias vector respectively; W_{OL} and B_{OL} are the matrices of output layer weights and bias vector respectively; u is the vector of inputs or features; f_{HL} and f_{OL} are hidden layer and output layer activation functions respectively. The mathematical representation given by Eq. 1 represents a fully connected feedforward artificial neural network (FC-ANN) which transforms the information in input, u, to the succeeding layers, and eventually to the output vector, y. The dimensions of ANN weights in Eq. 1 depend on the number of inputs, outputs and number of neurons (a hyper parameter), which are determined manually before training. In general, as the dimensions get larger, higher number of connections and parameters are introduced, which in turn provides higher capability of fitting to the training data. More parameters can also be introduced by

adding more hidden layers, and thus connections. This task lies within the concept of deep learning, providing beneficial application pathways in the literature (Alom et al. 2019).

2.2 Proposed Method

The hyperparameter management of ANNs is a challenging task since it requires significant number of trials. The common approach to especially decide on which input to select through ANN training is trial and error, where sequential selection and training steps are followed. In practice, those hyperparameters include the number of neurons, selection of inputs, and selection of activation functions. For instance, f_{HL} has a wide range of functions to address the different dataset and training algorithm needs, unlike f_{OL}, which is restricted to some form of normalization function once the classification is under consideration. In this case, a *softmax* activation used to calculate the probabilities of the outputs. Using the *softmax* function, the probability of j^{th} output, y_j, based on a particular vector, v, is calculated from:

$$y_j = \frac{e^{v_j}}{\sum_{i=1}^{M} e^{v_i}} \tag{2}$$

where M is the number of outputs. In ANN applications, the vector v for the calculation of the output probabilities are delivered by the hidden layer, due to feedforward information flow throughout the network.

ANNs are traditionally trained using nonlinear optimization methods where the sum of squared errors, or similar metrics, is minimized. As a result, optimal network parameters are obtained. Such a nonlinear optimization problem assumes a fixed ANN architecture including all the inputs in the dataset, in general.

In this study, the main idea is to represent the existence of the features using a binary variable and embed the selection procedure into the training algorithm, which in turn results in an MINLP problem. Accordingly, the training problem can be utilized simultaneously with the feature selection in a rigorous way, unlike many other sequential feature selection methods. The MINLP problem formulated to address both the input selection and training is given by:

$$\min_{W_{OL},W_{HL},B_{OL},B_{HL},u_s} z = \sum_{i=1}^{N}\sum_{j=1}^{M} -y_{ij}^{measurement} \cdot ln(y_{ij})$$

$$s.t.$$

$$y_i = f_{OL}(W_{OL} \cdot f_{HL}(W_{HL} \cdot u_i + B_{HL}) + B_{OL}), \in \{1, \dots, N\}$$

$$W_{HL}^{ij} \le W_{HL}^{max} \cdot u_s^j, i \in \{1, \dots, neuron\}, j \in \{1, \dots, inputs\} \tag{3}$$

$$W_{HL}^{ij} \le W_{HL}^{max} \cdot u_s^j, i \in \{1, \dots, neuron\}, j \in \{1, \dots, inputs\}$$

$$\sum_{i=1}^{U} u_{s,i} = u_{desired}$$

$$u_s \in \{0,1\}$$

where z is the cross-entropy function, u_s is the binary vector includes the decision variable to select a particular input to account for in the ANN architecture; where u_i is the vector of i^{th} inputs; y_i is the vector of ith output. This way, the most statistically major inputs can be selected while minimizing the traditional cross-entropy formulation. Please note that the major difference of Eq.3 to Eq.1 is the inclusion of the

binary input selection vector, u_s, embedded into the ANN equation. This way, the optimization detects whether or not to use the information from the particular feature value while training the ANN. Computation of the output variable given by Eq. 3 for a particular input sample, u, requires both the value of continuous weights (or parameters) and bias values, and binary input selection decision variables to be computed during training. Thus, to detect the optimal inputs and train the ANN network systematically and in a simultaneous fashion, a mixed-integer nonlinear optimization problem is formulated. Additional linking constrains given by Eq. 3 ensure setting the weights for the columns, which correspond to the eliminated inputs, to zero. On the other hand, the same constraints ensure the weight limits of the columns, tightening the search space to favor computational efficiency. To the best of the author's knowledge, formulation of the presented method as a mixed-integer nonlinear programming for classification problems in a systematic, where the nonlinearity of the ANN networks are considered, is still a scarce subject. Eq.3, on the other hand, represents a highly non-convex mixed-integer nonlinear programming problem, whose solution to global optimality is a quite challenging task. Solving MINLP problems requires branching on binary variables and solving the corresponding NLP relaxations at the same time. Derivative-based and derivative-free methods are available for the solution of the problem. For this work, a derivative-based local MINLP solver (DICOPT) is used, but evolutionary algorithms (e.g. genetic algorithm) may turn out to be useful for identifying a heuristic or initial solution. The major contribution of this study includes the development of Eq. 3 to address the simultaneous input selection and ANN training using rigorous optimization formulation, rather than evolutionary algorithms. Suggested formulation is applied in PYOMO language and the related MINLP problem is solved using the DICOPT solver (Hart, Watson, and Woodruff 2011; Kocis and Grossmann 1989). PYOMO is an internationally recognized open-source algebraic optimization modeling language which provides a user-friendly environment. Several useful open-source MINLP solvers can also be integrated in PYOMO (Kronqvist et al. 2019). Optimal results were obtained in less than 30 seconds for all of the cases given in results section.

For detailed explanation of MINLP problems and related solution algorithms, the reader is referred to (Kronqvist et al. 2019). Finally, a flowchart of the proposed method is represented below:

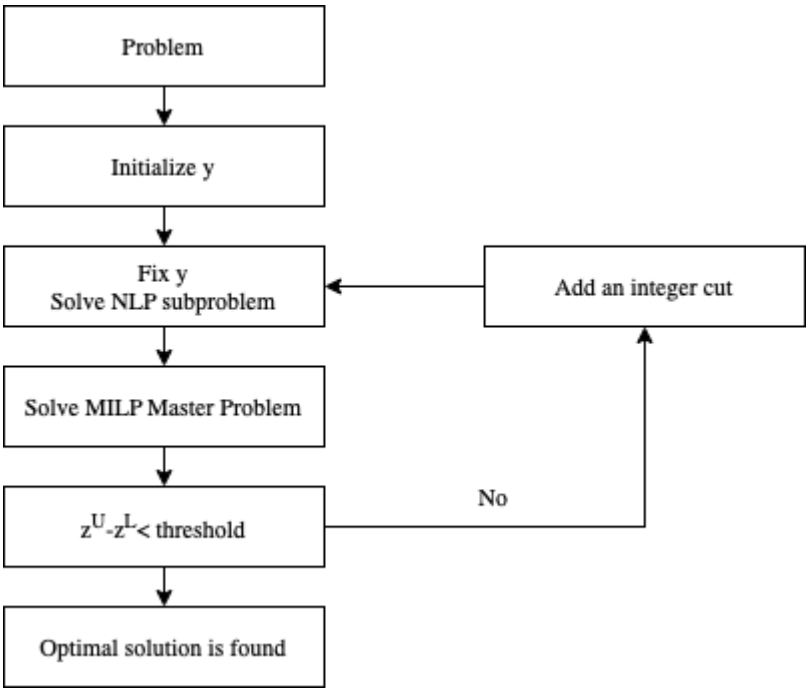

Figure 1: Flowchart of the solution of the MINLP problem using Outer Approximation (Duran and Grossmann 1986).

Here, y represents the set of binary variables comprised of the candidate features and the NLP subproblem represents the standard ANN training. z^U and z^L are the upper and the lower bounds of the objective function for the problem defined by Eq.3, respectively. MILP Master Problem stands for the linearized version of the original MINLP problem and in this solution setting, the training problem and the MILP Master problem are solved iteratively until the optimal values of both problems approach close to each other. Convergence is assured by automatically including integer cuts to the training problem, which stands for the optimal selection of the features or inputs, represented by the feature selection vector as mentioned before. As opposed to selecting the features randomly and sequentially, the addition of the integer cuts ensures the selection of the best features while training the ANN.

3. RESULTS

This section compares the training and the test performance of ANNs for all inputs and selected inputs obtained from the proposed optimization problem. The application includes three commonly used datasets. All datasets are publicly available classification benchmarks but in addition to that, the first case study also justifies the use of machine learning techniques for medical applications which is a relatively new area. The second application is also associated with chemical engineering and biotechnology, which is again a unique and interesting application area for engineering and the last dataset is related to heart failure detection.

The performance criterion is the confusion matrix for the training and the test data. 50% training ratio is used for two cases after shuffling the data randomly. Traditional fully connected ANN (FC-ANN) which utilizes all the available inputs, without any input selection algorithm, is the first architecture for performance evaluation. Secondly, SIS-ANN (Sequential Input Selection based ANN) results are presented in which ANN is trained using selected inputs from sequential input selection algorithm. Finally, MIP-ANN (Mixed Integer Programming based ANN) results, which demonstrate the performance obtained the mixed-integer formulation in Eq. 3, is presented. All architectures use hyperbolic tangent activation functions in two hidden layer neurons.

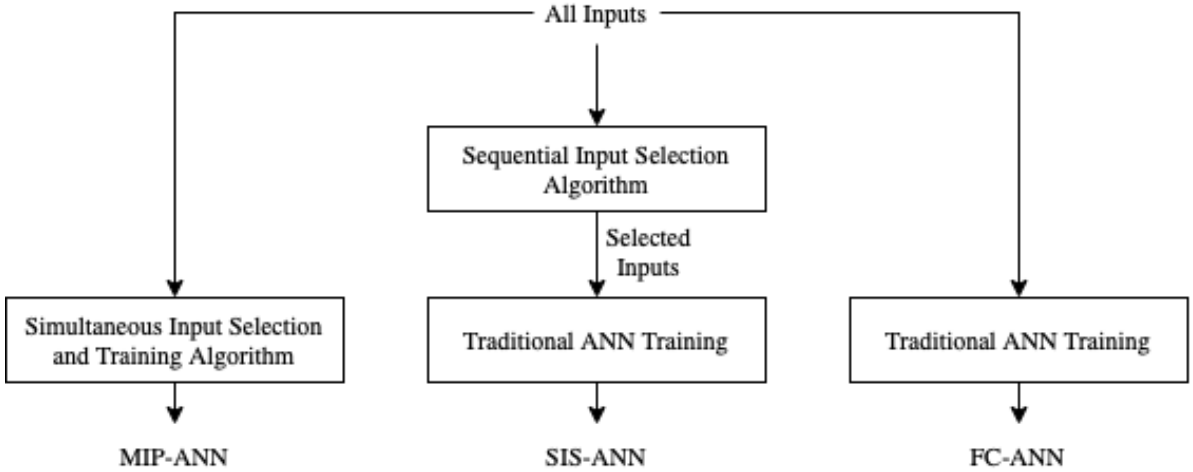

Figure 2: Different ANN methodologies which are used for comparison

3.1. Breast Cancer Dataset

The breast cancer dataset is a popular classification benchmark which found significant attention from the literature (Agarap 2018; Benbrahim, Hachimi, and Amine 2019; Lavanya and Rani 2011). The dataset contains mean, standard deviation and worst values for 10 different structural properties to diagnose a patient as cancer or not. The problem requires the measurement of high number of input variables once the input reduction is not applied. On the other hand, the determination of an optimal subset is a

challenging task with high number of combinations, hindering the use of trial and error procedures. Table 1 provides the available inputs from which dark market entries are selected based on the corresponding algorithm. Note that, FC-ANN inputs are not included in Table 1 since all of them are used for the method. In this case, a significant input reduction is required from the algorithms to observe the theoretical contributions more clearly. Both sequential input selection and mixed-integer based approach focuses on the worst values of particular inputs for diagnosis. Unlike sequential input selection algorithm, which uses perimeter and symmetry for the decision making, mixed-integer formulation uses radius and concavity to calculate the cancer status.

Table 1. Selected inputs for Case 1

		radius	texture	perimeter	area	smoothness	compactness	concavity	concave points	symmetry	fractal dimension
SIS-ANN	**Mean**										
	Std										
	Worst			■						■	
MIP-ANN	**Mean**										
	Std										
	Worst	■							■		

Table 2 includes the training and the test performances for FC-ANN, SIS-ANN and MIP-ANN. The former delivers the best training performance, thanks to high number of weights in the ANN architecture. However, a significant test performance drop is obtained for the FC-ANN architecture, which is not desired in many cases and considered a measure of overfitting. SIS-ANN delivers a reduced test performance compared to FC-ANN using two inputs shown in Table 1. However, overfitting is decreased, as well, since a perfect classification is not observed in the training.

Table 2. Classification comparison of different ANN approaches for Case 1

		FC-ANN		**SIS-ANN**		**MIP -ANN**	
		C_1	**C_2**	**C_1**	**C_2**	**C_1**	**C_2**
Train	**C_1**	109	0	103	6	97	12
	C_2	0	175	4	171	0	175
	Accuracy	1.000		0.965		0.957	
Test	**C_1**	94	9	89	14	84	19
	C_2	10	172	13	169	1	181
	Accuracy	0.933		0.905		0.930	

MIP-ANN delivers a similar test performance, compared to FC-ANN, with a convenient training performance. Such similarity is an indication of eliminated overfitting. In addition, despite poor performance compared to both FC-ANN and SIS-ANN in the training data, MIP-ANN outperforms in test. Such a performance is possible due to tailored formulation in Eq. 3, which accounts for the ANN architecture and the impact of the inputs simultaneously during the ANN development. As a major practical advantage, only two measurements for real-time applications would deliver a satisfactory performance besides its computational advantages, when a new prediction or model update is required.

3.2. Wine Dataset

Wine dataset contains 178 samples of 13 inputs which are used for the prediction of 3 classes. The dataset has found significant applications in the literature (Bredensteiner and Bennett 1999; Zhong and Fukushima 2007). The available and selected inputs are presented in Table 3.

Table 3. Selected inputs for Case 2

	alcohol	malic_acid	ash	lcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	hue	od280/od315_of_diluted_wines	proline
SIS-ANN	■	■									■		■
MIP -ANN				■			■					■	■

Four input samples out of 13 were selected using sequential input selection and mixed- integer based selection algorithm. The last input is selected by both algorithms; however, other selected inputs are different for the two approaches.

Table 4 includes the performances of FC-ANN, SIS-ANN and MIP-ANN based on the confusion matrices.

Table 4. Classification comparison of different ANN approaches for Case 2

		FC-ANN			**SIS-ANN**			**MIP -ANN**		
		C_1	**C_2**	**C_3**	**C_1**	**C_2**	**C_3**	**C_1**	**C_2**	**C_3**
Train	**C_1**	26	0	0	26	0	0	24	2	0
	C_2	0	37	0	0	37	0	2	35	0
	C_3	0	0	26	0	0	26	0	0	26
	Accuracy	1.000			1.000			0.955		
Test	**C_1**	32	1	0	32	1	0	32	1	0
	C_2	7	26	1	3	31	0	1	32	1
	C_3	0	1	21	3	3	16	0	0	22
	Accuracy	0.887			0.887			0.966		

FC-ANN and SIS-ANN have similar training and test performances, despite different samples are misclassified in the test. Both ANN architectures have a similar performance drop based on their training and test accuracies. The performance drop in the FC-ANN clearly stems from the overfitting problem since all the available information is already present in the training. However, SIS-ANN suffers from the selection of an inefficient input subset. MIP-ANN delivers a similar training and test performance. In addition, the latter is the superior among three ANN architectures.

3.3. Heart Failure Clinical Records Dataset

Heart failure clinical records dataset (Chicco and Jurman 2020) includes 299 measurements, which enable the prediction of death event under follow-up period based on 12 important and potential indicators of current health status of a human. Unlike other cases, in order to show the superiority of the proposed methodology under relatively small training ratio, only 10% randomly selected data of the data set is used for the training. Such a small training ratio is also useful to demonstrate the generalization capability of the method, which is an important and highly-encountered concern when data are limited or measurements are challenging. Table 5 shows the selected inputs based on the methods.

Table 5. Selected inputs for Case 3

	age	anemia	high blood pressure	creatinine phosphokinase	diabetes	ejection fraction	platelets	sex	serum creatinine	serum sodium	smoking	time
SIS-ANN					■			■				
MIP -ANN				■								■

The impact of the input selection is observed on the prediction performance both from the point selection of actual indicator, which drives or leads the complex and highly-nonlinear nature of the existing interactions in the human body, and the reduction in overfitting thanks to eliminated connections from the particular inputs to the hidden layer. The impact of the overfitting is observed on FC-ANN, which suffers from significant performance difference in training and test despite all inputs are available in the training set. On the other hand, SIS-ANN, although significant amount of connections and related weights are removed from the ANN structure due to feature selection, delivers a relatively-low prediction accuracy since a better input selection could not be performed with a simple input selection algorithm without considering the architecture of the ANN formulation explicitly.

Table 6. Classification comparison of different ANN approaches for Case 3

		FC-ANN		**SIS-ANN**		**MIP -ANN**	
		C_1	**C_2**	**C_1**	**C_2**	**C_1**	**C_2**
Train	**C_1**	17	0	14	3	17	0
	C_2	0	12	3	9	5	7
	Accuracy	1.000		0.793		0.827	
Test	**C_1**	133	53	132	54	174	12
	C_2	32	52	31	53	55	29
	Accuracy	0.685		0.685		0.751	

4. CONCLUSION

ANNs are empirical models with high number of tuning parameters. They have flexible structure which requires the pre-specification of inputs, neuron number, activation functions, training algorithm, training data selection and many other issues with high theoretical complexity. In addition, the approach to deal with such issues are usually data and case dependent; thus generalization of ANN design and training methods is a challenging task. A major and commonly encountered issue is overfitting, which is caused by the selection of high number of inputs and hidden neuron number in addition to lack of statistically meaningful dataset.

This study focuses on the selection of the optimal inputs through an MINLP formulation and does not explicitly address other aforementioned considerations during training of the ANN development. The impact of the proposed approach is implemented on three publicly available and widely used datasets, showing the contribution of the approach. The method delivers statistically sufficient and meaningful

input subset, resulting in a desired training performance and good prediction performance despite significant reduction in total number of inputs. This has several advantages over the traditional trial and error based and sequential approaches. Once the number of features (or inputs) is decreased, the ANN number of parameters for training also decreases together with the training computational load without causing underfitting. In addition, model update when a new measurement becomes available gets faster. Once those ANNs are implemented on actual implementations with real-time prediction, a smaller input space would enable a more robust functionality over ANNs considering all inputs. At the same time, please note that some inputs obtained from the sensors are exposed to failures in daily operation.

A particular input might be eliminated from input space both due to its correlation with other inputs and its irrelevance to the output. The former consideration may show the multiplicity in ANNs which is beyond the scope of this paper. In practice there are many less-dependent input subset combinations among a dependent input set. Here, the optimization algorithm has a major role over the subset selection among the correlated inputs. The solution of MINLPs is usually computationally expensive task for larger networks and datasets.

Many other hyper-parameter related performance influencing architectural decisions exist in the ANN formulation. Activation function selection, number of hidden neurons, optimization algorithm tuning parameters, training ratio, scaling of the data and initialization of the optimization are only some of those. Thus, a more comprehensive and ultimate comparison is out of the scope of the paper and limited to the input selection only.

Our future works include the development of reformulations, which decompose the problem into smaller and easier-to-solve problems for handling large datasets.

5. ACKNOWLEDGEMENT

This publication has been produced benefiting from the 2232 International Fellowship for Outstanding Researchers Program of TUBITAK (Project No: 118C245). However, the entire responsibility of the publication belongs to the owner of the publication.

REFERENCES

Agarap, Abien Fred M. 2018. "On Breast Cancer Detection: An Application of Machine Learning Algorithms on the Wisconsin Diagnostic Dataset." In *Proceedings of the 2nd International Conference on Machine Learning and Soft Computing*, 5–9.

Aha, David W, and Richard L Bankert. 1996. "A Comparative Evaluation of Sequential Feature Selection Algorithms." In *Learning from Data*, Springer, 199–206.

Akdag, Unal, M. Aydin Komur, and A. Feridun Ozguc. 2009. "Estimation of Heat Transfer in Oscillating Annular Flow Using Artifical Neural Networks." *Advances in Engineering Software* 40(9): 864–70.

Alom, Md Zahangir et al. 2019. "A State-of-the-Art Survey on Deep Learning Theory and Architectures." *Electronics (Switzerland)* 8(3): 292.

Azadeh, A., S. F. Ghaderi, and S. Sohrabkhani. 2008. "Annual Electricity Consumption Forecasting by Neural Network in High Energy Consuming Industrial Sectors." *Energy Conversion and Management* 49(8): 2272–78.

Benbrahim, Houssam, Hanaâ Hachimi, and Aouatif Amine. 2019. "Comparative Study of Machine Learning Algorithms Using the Breast Cancer Dataset." In *International Conference on Advanced Intelligent Systems for Sustainable Development*, Springer, 83–91.

Bredensteiner, Erin J., and Kristin P. Bennett. 1999. "Multicategory Classification by Support Vector Machines." *Computational Optimization and Applications* 12(1–3): 53–79.

Castellano, Giovanna, and Anna Maria Fanelli. 2000. "Variable Selection Using Neural-Network Models." *Neurocomputing* 31(1–4): 1–13.

Chicco, Davide, and Giuseppe Jurman. 2020. "Machine Learning Can Predict Survival of Patients with Heart Failure from Serum Creatinine and Ejection Fraction Alone." *BMC medical informatics and*

decision making 20(1): 1–16.

Diaz, G I, A Fokoue-Nkoutche, G Nannicini, and H Samulowitz. 2017. "An Effective Algorithm for Hyperparameter Optimization of Neural Networks." *IBM Journal of Research and Development* 61(4/5): 9:1-9:11.

Duran, Marco A., and Ignacio E. Grossmann. 1986. "An Outer-Approximation Algorithm for a Class of Mixed-Integer Nonlinear Programs." *Mathematical Programming* 36(3): 307–39.

Ferri, Francesc J, Pavel Pudil, Mohamad Hatef, and Josef Kittler. 1994. "Comparative Study of Techniques for Large-Scale Feature Selection." In *Machine Intelligence and Pattern Recognition*, Elsevier, 403–13.

Feurer, Matthias, and Frank Hutter. 2019. "Hyperparameter Optimization." In *Automated Machine Learning*, Springer, Cham, 3–33.

Hart, William E., Jean Paul Watson, and David L. Woodruff. 2011. "Pyomo: Modeling and Solving Mathematical Programs in Python." *Mathematical Programming Computation* 3(3): 219–60.

Kocak, Habip, and Turgut Un. 2014. "Forecasting the Gold Returns with Artifical Neural Network and Time Series." *International Business Research* 7(11).

Kocis, Gary R, and Ignacio E Grossmann. 1989. "Computational Experience with DICOPT Solving MINLP Problems in Process Systems Engineering." *Computers & Chemical Engineering* 13(3): 307–15.

Kronqvist, Jan, David E Bernal, Andreas Lundell, and Ignacio E Grossmann. 2019. "A Review and Comparison of Solvers for Convex MINLP." *Optimization and Engineering* 20(2): 397–455.

Lavanya, D, and Dr K Usha Rani. 2011. "Analysis of Feature Selection with Classification: Breast Cancer Datasets." *Indian Journal of Computer Science and Engineering (IJCSE)* 2(5): 756–63.

Leahy, Paul, Ger Kiely, and Gearóid Corcoran. 2008. "Structural Optimisation and Input Selection of an Artificial Neural Network for River Level Prediction." *Journal of Hydrology* 355(1–4): 192–201.

Ledesma, Sergio et al. 2008. "Feature Selection Using Artificial Neural Networks." In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 351–59.

Manngård, Mikael, Jan Kronqvist, and Jari M Böling. 2018. "Structural Learning in Artificial Neural Networks Using Sparse Optimization." *Neurocomputing* 272: 660–67.

Mutlu, Ali Yener, and Ozgun Yucel. 2018. "An Artificial Intelligence Based Approach to Predicting Syngas Composition for Downdraft Biomass Gasification." *Energy* 165: 895–901.

Poernomo, Alvin, and Dae-Ki Kang. 2018. "Biased Dropout and Crossmap Dropout: Learning towards Effective Dropout Regularization in Convolutional Neural Network." *Neural Networks* 104: 60–67. https://www.sciencedirect.com/science/article/pii/S0893608018301096.

Rückstieß, Thomas, Christian Osendorfer, and Patrick van der Smagt. 2011. "Sequential Feature Selection for Classification." In *Australasian Joint Conference on Artificial Intelligence*, Springer, 132–41.

Sahinidis, Nikolaos V. 1996. "BARON: A General Purpose Global Optimization Software Package." *Journal of Global Optimization* 8(2): 201–5.

Schittkowski, K. 2007. "Experimental Design Tools for Ordinary and Algebraic Differential Equations." In *Industrial and Engineering Chemistry Research*, 9137–47.

Sildir, Hasan, Erdal Aydin, and Taskin Kavzoglu. 2020. "Design of Feedforward Neural Networks in the Classification of Hyperspectral Imagery Using Superstructural Optimization." *Remote Sensing* 12(6). https://www.mdpi.com/2072-4292/12/6/956.

Stamoulis, Dimitrios, Ermao Cai, Da-Cheng Juan, and Diana Marculescu. 2018. "Hyperpower: Power- and Memory-Constrained Hyper-Parameter Optimization for Neural Networks." In *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, IEEE, 19–24.

Verikas, A., and M. Bacauskiene. 2002. "Feature Selection with Neural Networks." *Pattern Recognition Letters* 23(11): 1323–35.

Van De Wal, Marc, and Bram De Jager. *A Review of Methods for Input/Output Selection*.

Yetilmezsoy, Kaan, Bestamin Ozkaya, and Mehmet Cakmakci. 2011. "Artificial Intelligence-Based

Prediction Models for Environmental Engineering." *Neural Network World* 21(3): 193–218.
Zhong, Ping, and Masao Fukushima. 2007. "Regularized Nonsmooth Newton Method for Multi-Class Support Vector Machines." In *Optimization Methods and Software*, 225–36.