FsMo - FsAe: A Comparative Study of Multi-Objective Using Binary Differential Evolution and Autoencoder-Based Feature Selection on the Lung Gene Expression Data

Abstract

High-dimensional datasets, such as those related to lung gene expression, present major challenges due to the presence of irrelevant or redundant features. In this study, we conduct a comparative analysis of multi-objective feature selection and an autoencoder-based approach to address this issue. We employ the Binary Differential Evolution (BDE) algorithm and evaluate three configurations: the original dataset (without feature selection), an autoencoder-based feature selection method(FsAe), and a multi-objective feature selection method (FsMo), which simultaneously optimizes the Mean Squared Residue (MSR) and the number of selected features. Our experimental results show that the FsMo method outperforms both the autoencoder-based method and the unfiltered dataset in terms of classification accuracy.

Keywords: Microarray data, feature selection, multi-objective , Differential evolution, Mutation, Selection, Optimization.

1 Introduction

The high dimensionality of microarray gene expression data characterized by thousands of genes and only a limited number of samples leads to a severe imbalance known as the "curse of dimensionality." This issue hampers accurate inference and significantly increases computational complexity.

FS tends to pick up a small significant subset of features from the original dataset by removing irrelevant, redundant, or noisy features based on a predefined evaluation measure. FS methods mainly include three categories that are the filter, wrapper, and embedded approaches [1].

In this study, we focus on a high -dimensional lung dataset expressions to evaluate the efficiency of different feature selection techniques. We present a comparative analysis between a multi-objective feature selection approach, the Mean Squared Residual (MSR) [2, 3] is optimized simultaneously with limiting the number of selected gene and an autoancoder -based method, both combined with Binary Differential Evolution

(BDE). The purpose is to show how multi-objective optimization can help to cope with challenges generated by high -dimensional data, and increase both data quality and interpretation of models. Experimental results clearly show that the multi-objective approach (FsMo) improves both autocoders and original data sets.

2 Review on feature selection methods

Feature selection [4] is a widely used machine learning technique to address the challenges of high-dimensional data. Its main goal is to reduce the number of features by selecting only those that are most relevant to the classification task. Depending on how they interact with the learning algorithm, feature selection methods are generally grouped into three categories: filter, wrapper, and hybrid approaches [5].

3 FsMo and autocoders approaches

3.1 Differential evolution (DE)

The well-known evolutionary algorithm DE was first put forth by Storn and Price in 1997 [6]. DE employs three strategies crossover, mutation, and selection in each generation to arrive at the global optimum solution. As illustrated in Figure 2.

FsMo based on a binary variant of the Differential Evolution algorithm (BDE). Since feature selection (FS) is inherently binary, population members in BDE are initialized as binary vectors. For a dataset with N features (number of genes), each solution in the DE algorithm consists of N components, where each component represents a feature. As shown in Figure 1, a value of 1 indicates that a feature is selected, while 0 means it is not.

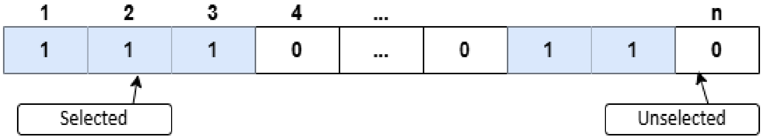

Fig. 1: Representation of binary string encoding for FS solution.

FsMo utilizes multi-objective differential evolution to simultaneously optimize the Mean Squared Residue (MSR) and the number of selected features, therefore it applies a non-dominated sorting genetic algorithm (NSGA-II) [7].

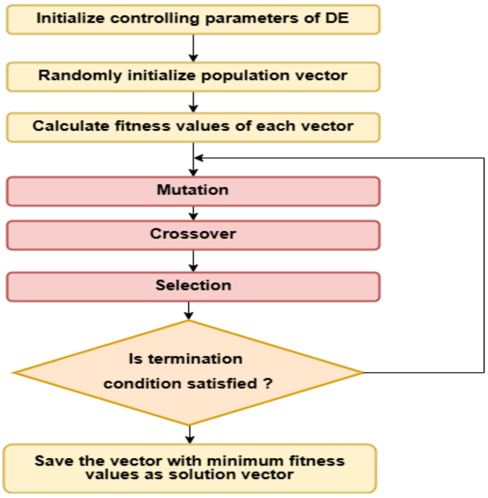
Initialize controlling parameters of DE
Randomly initialize population vector
Calculate fitness values of each vector
Mutation
Crossover
Selection
Is termination condition satisfied ?
Save the vector with minimum fitness values as solution vector

Fig. 2: Phases of differential evolution

3.2 autocoders

An autoencoder is a specific type of a neural network, which is mainly designed to encode the input into a compressed and meaningful representation, and then decode it back such that the reconstructed input is similar as possible to the original one.

Autoencoders have been first introduced in [8] as a neural network that is trained to reconstruct its input. Their main purpose is learning in an unsupervised manner an "informative" representation of the data that can be used for various implications such as clustering. As shown in Figure 3

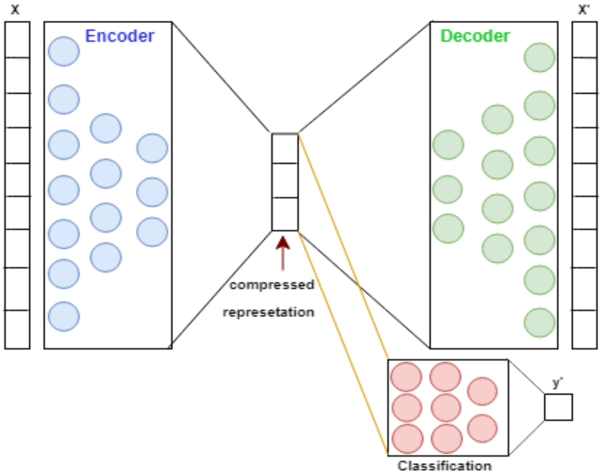

Fig. 3: An autoencoder

4 Illustration of FsMo and autocoders algorithms

for the FsMo We split the dataset into two subsets, training 80% and testing 20%, and then apply the FsMo method to the training set. We take the identical genes that the algorithm returned on the test set and classify them without running the algorithm again. The autoencoder must compress the dataset into a latent space of the same size as the one returned by FsMo which allows a fair comparison during the classification phase. As illustrated in Figure 5 . The FsMo and autocoders algorithms was evaluated using real DNA microarray data Lung illustrated in Table 1. by comparing accuracy of the new dataset, generated through feature selection (FS) methods, namely FsMo and FsAe, and the original dataset. As illustrated in Table 3.

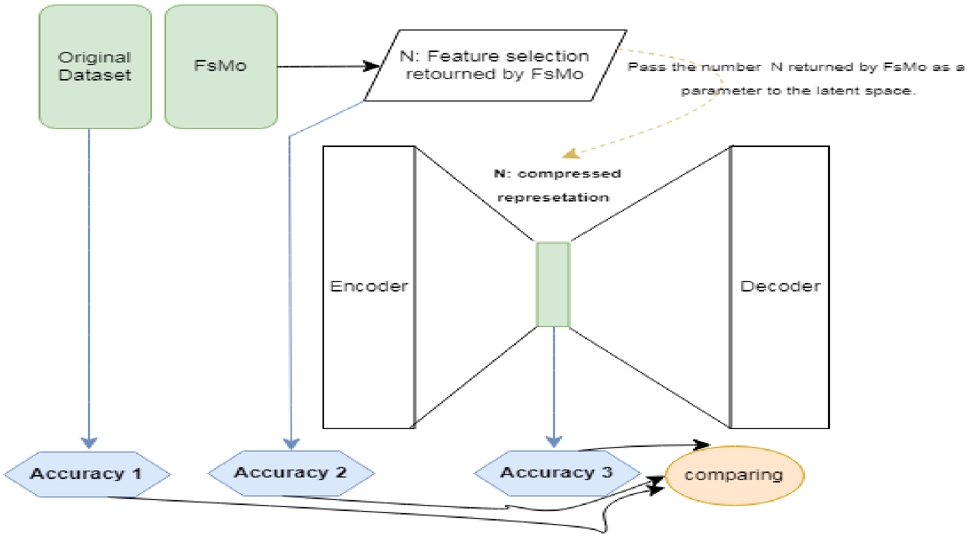
Original Dataset
FsMo
N: Feature selection retourned by FsMo
Pass the number N returned by FsMo as a parameter to the latent space.
N: compressed represetation
Encoder
Decoder
Accuracy 1
Accuracy 2
Accuracy 3
comparing

Fig. 4: llustration of FsMo and autocoders FsAe

Table 1: Description of the high-dimensional Lung microarray dataset

Dataset	**#Features**	**#Samples**	**#Classes**
Lung	12,600	203	5

The Support Vector Machine(SVM), Naïve Bayes(NB), and K-Nearest Neighbors(KNN) classifiers were used to evaluate the quality of the selected features. The population was fixed at 20 and the number of iterations at 50. After applying our algorithms, we observed that most of the solutions reduced the dataset by approximately 50%.

5 Experimental results

The feature selection method (FsMo) returned 6,287 genes, representing an approximate reduction of 50.1%. This value will subsequently be provided to the autoencoder in the latent space, where it is expected to return the same number of genes.

Since tuning the autoencoder to achieve good results is empirical, we tested several architectures and selected the one shown in in Table 2, which achieved the best performance.

Table 2: Summary of the Autoencoder architecture with input size 12,600 and latent dimension 6,287

Layer (type)	**Output Shape**	**Param #**
InputLayer (input_layer)	(None, 12600)	0
Dense (dense)	(None, 8192)	103,227,392
Dropout (dropout)	(None, 8192)	0
Dense (dense_1)	(None, 6400)	52,435,200
Dropout (dropout_1)	(None, 6400)	0
Dense (dense_2)	(None, 6287)	40,243,087
Dense (dense_3)	(None, 6400)	40,243,200
Dense (dense_4)	(None, 8192)	52,436,992
Dense (dense_5)	(None, 12600)	103,231,800
Total parameters	391,817,671 (1.46 GB)	
Trainable parameters	391,817,671 (1.46 GB)	
Non-trainable params	0 (0.00 B)	

A classification will then be performed on this reduced dataset, allowing us to compare the accuracy of the models before(All features) and after the feature selection and dimensionality reduction process(FsMo and FsAe). as presented in Figure 5 and detailed in Table 3.

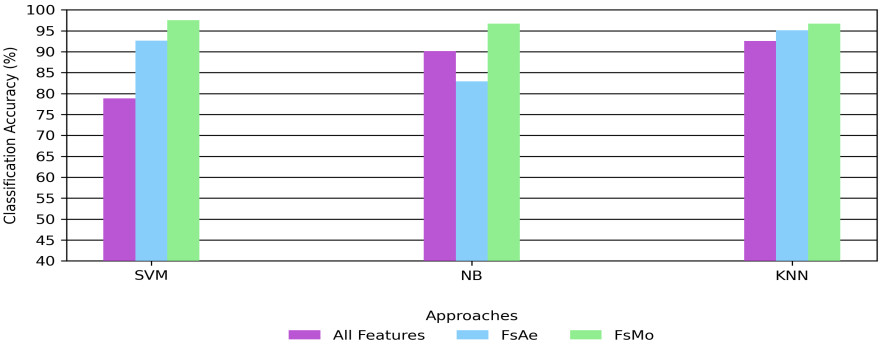

Fig. 5: Accuracy Plot for Experimental Results

Table 3: The comparison of classifiers using all features FsMo and FsAe in accuracy

Classifier	All features	FsAe	FsMo
SVM	78.82	92.68	97.56
NB	90.15	82.92	96.72
KNN	92.61	95.12	96.72

Table 4: Comparison between FsAe and FsMo

Criterion	FsAE	FsMo
Method Type	Deep learning (neural network-based)	Evolutionary algorithm (stochastic search)
Execution Complexity	Moderate: $O(n \times d \times \text{epochs})$ n: samples, d: gènes	High: $O(\text{generations}\times\text{population}\times\text{evaluation time})$
Memory Usage	Medium to high (due to neural network)	Low to medium
Interpretability	Low (black-box model)	Medium to high (explicit feature sets)
Hyperparameters	Network architecture, learning rate, dropout, etc.	Population size, mutation/crossover rates, generations

6 Conclusion

In this study, we investigated the effectiveness of feature selection techniques on a high-dimensional Lung dataset . We compared the original dataset without feature selection, an autoencoder-based feature selection method(FsAe), and a multi-objective feature selection approach (FsMo)using Binary Differential Evolution (BDE), which aims to balance Mean Squared Residue (MSR) with the number of selected features. Our results demonstrated that multi-objective approaches not only reduce dimensionality efficiently but also maintain a strong balance between model accuracy and interpretability.

Moving forward, more studies could try applying these techniques to other complex datasets, like those related to breast cancer, cns cancer, or brain cancer. It might also be interesting to assess them using different evaluation metrics, such as F1 score, recall, and precision, to see how well they adapt to different cases.

References

[1] Pham, T.H., Raahemi, B.: Bio-inspired feature selection algorithms with their applications: A systematic literature review. IEEE Access **11**, 43733–43758 (2023)

[2] Cheng, Y., Church, G.M.: Biclustering of expression data. In: Ismb, vol. 8, pp. 93–103 (2000)

[3] Noronha, M.D., Henriques, R., Madeira, S.C., Zárate, L.E.: Impact of metrics on biclustering solution and quality: a review. Pattern Recognition **127**, 108612

(2022)

[4] Bolón-Canedo, V., Alonso-Betanzos, A.: Ensembles for feature selection: A review and future trends. Information fusion **52**, 1–12 (2019)

[5] Kannan, S.S., Ramaraj, N.: A novel hybrid feature selection via symmetrical uncertainty ranking based local memetic search algorithm. Knowledge-Based Systems **23**(6), 580–585 (2010)

[6] Storn, R., Price, K.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. Journal of global optimization **11**, 341–359 (1997)

[7] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE transactions on evolutionary computation **6**(2), 182–197 (2002)

[8] Rumelhart, D.E., Hinton, G.E., Williams, R.J., et al.: Learning internal representations by error propagation. Institute for Cognitive Science, University of California, San Diego La . . . (1985)