www.editada.org

International Journal of Combinatorial Optimization Problems and Informatics, 17(5), 2026, 37-43. ISSN: 2007-1558.
https://doi.org/10.61467/2007.1558.2026.v17i5.1273

Does size matter? The influence of reduct size on the performance of supervised classifiers

Manuel S. Lazo-Cortés[1], Rafael Rodríguez-Puente[2], Adolfo Diaz-Sardinas[2]

[1]Tecnológico Nacional de México/IT Tlalnepantla,

[2]University of Namibia

Emails: manuel.lc@tlalnepantla.tecnm.mx; rpuente@unam.na; adiaz@unam.na

Abstract. This study examines how the size of rough-set reducts influences the performance of supervised classifiers. Using twenty-one benchmark datasets, two reduct representations were compared: minimum-length reducts, which contain the smallest attribute subsets that preserve consistency, and maximum-length reducts, which correspond to the largest irredundant subsets. Eleven classifiers were evaluated using five performance measures, and statistical differences were analyzed through the Friedman–Nemenyi procedure. For most classifiers, both reduct representations produced lower predictive performance than using the full set of attributes. However, JRIP, Logistic Regression, and multilayer perceptron showed broadly comparable performance across representations, except in the PRC area, where maximum-length reducts tended to perform worse. For SMO, minimum-length reducts were generally preferable to maximum-length reducts. These findings indicate that the practical value of reduct-based dimensionality reduction depends largely on the classifier being used. In some cases, substantial attribute reduction can be achieved without a meaningful loss of predictive accuracy.

Keywords: rough sets; reducts; dimensionality reduction; supervised classification; feature selection.

Article Info
Received Dec 11, 2025
Accepted Aug 23, 2026

1 Introduction

The dimensionality of the feature space plays an important role in the performance and efficiency of supervised classification methods. High dimensionality is commonly associated with the curse of dimensionality, which may increase computational cost and reduce predictive performance when irrelevant, redundant, or weakly informative attributes are present. To address this issue, feature selection methods aim to reduce dimensionality while preserving, or in some cases improving, classification accuracy (Guyon & Elisseeff, 2003). In addition to computational benefits, reduced representations may improve several characteristics, such as interpretability and generalization to new data.

Rough set theory, originally proposed by Pawlak (Pawlak, 1982; Pawlak, 1991), offers a well-established approach to attribute reduction without requiring extra information about data, such as probability distributions or membership functions. In this context, a reduct is a group of attributes that is itself irreducible and allows objects to be classified just as well as all the original attributes would. There is usually more than one way to reduce attributes, so there is no single correct answer. The same dataset may contain several alternative reducts, often differing in both size and composition. This leads to a practical question: to what extent does the selected reduct, and especially its size, affect classification performance?

Most previous studies have concentrated on the computational side of reduct generation and on the development of algorithms for obtaining reducts (Chen et al., 2015; Choromanski et al., 2020; Hu et al., 2000; Jensen &

Shen, 2003; Lazo-Cortés et al., 2016; Rodríguez-Diez et al., 2021; Rodríguez-Diez et al., 2020). Other works have examined the use of reduct-based representations in classification settings (Niu et al., 2021; Stączyk, 2023). By contrast, the specific influence of reduct size on predictive performance has received much less attention.

Minimum-length reducts are appealing because they offer compact and often interpretable representations. Even so, a smaller subset does not necessarily lead to the best predictive results. Larger reducts, on the other hand, may retain additional complementary information, although this usually comes with increased dimensionality.

This paper presents an empirical study of how reduct size affects the performance of supervised learning algorithms. In particular, minimum-length and maximum-length reducts are compared as alternative feature representations across a diverse set of classifiers. The objective is to clarify how reduct size interacts with classifier characteristics and to provide further insight into dimensionality reduction strategies derived from rough set theory.

2 Methodology

In this work, reduct size is examined not only as the cardinality of the selected subset, but also as a structural property associated with competing effects such as compactness, information retention and redundancy. The smaller reducts can provide simpler representations with lower variance, whereas the larger reducts can retain more complementary information at the cost of more redundancy or noise. We expect that the net effect will depend on the inductive properties of the classifier.

Formally, let $R \subseteq A$ be a reduct. Although $|R|$ denotes its cardinality, its predictive effect may depend on

$$Performance(R) = f(|R|, I(R), Red(R), Int(R), C),$$

where $I(R)$ represents preserved information, $Red(R)$ residual redundancy, $Int(R)$ attribute interactions, and C the classifier.

2.1 Datasets and Preprocessing

We started from 63 datasets from the UCI Machine Learning Repository (Kelly et al., n.d.), chosen to span a wide range of sizes, dimensionalities, and numbers of classes. Each dataset was randomly split into training and test partitions with a ratio of 70/30. Then, reducts were created from the training partition using the CC-GEAC algorithm (Gonzalez-Diaz et al., 2024). For each data set, one minimum and one maximum cardinality reduct were randomly selected from the obtained family.

A single train/test split was adopted deliberately. Because reducts are generated from the training data, repeated resampling would require repeating the reduct search itself, and it would introduce an additional source of variation beyond reduct size.

To better isolate the role of reduct size, datasets whose minimum and maximum reducts differed by two or fewer attributes were excluded. This left a final sample of 31 datasets.

The main characteristics of these datasets are presented in Table 1, including the number of instances (inst), attributes (attr, where "+1" denotes the class label), classes (classes), and the sizes of the minimum (min-card) and maximum (max-card) reducts.

Table 1. Main characteristics of the datasets used in the study, including the number of instances, attributes, classes, and reduct cardinalities.

N.	Name	inst	attr	classes	min-card	max-card
1	Audiology	226	69+1	24	10	32
2	Australian	690	14+1	2	3	7
3	Autos	205	25+1	2	2	6
4	Colic	368	22+1	2	6	10
5	Credit-a	690	15+1	2	3	6
6	Credit-g	1000	20+1	2	5	12
7	Cylinder-bands	540	39+1	2	2	11
8	Dermatology	366	34+1	2	2	12
9	Flags	194	28+1	2	4	14
10	German	1000	20+1	2	2	13
11	Heart-c	303	13+1	2	2	6
12	Heart-statlog	270	13+1	2	2	7
13	Hepatitis	155	19+1	2	2	11
14	Hypothyroid	3772	29+1	4	3	7
15	Ionosphere	351	34+1	2	2	6
16	Labor	57	16+1	2	2	6
17	Led24	3200	24+1	10	18	21
18	Letter	20000	16+1	26	10	13
19	Lung-cancer	32	56+1	3	3	11
20	Molecular-biology-promoters	106	57+1	2	1	7
21	Mushroom	8124	22+1	2	4	8
22	Ozone-level-8hr	2534	72+1	2	3	6
23	Pendigits	10992	16+1	10	4	7
24	Qsar-biodeg	1055	41+1	2	2	9
25	Sick	3772	29+1	2	3	6
26	Soybean	683	35+1	19	9	15
27	Spambase	4601	57+1	2	10	14
28	Sponge	76	44+1	2	2	10
29	Thyroid	7200	21+1	3	3	6
30	Vehicle	846	18+1	4	3	6
31	Vote	435	16+1	2	7	12

2.2 Classifiers and Evaluation

Eleven classifiers from the main algorithmic families available in Weka 3.9.6 (Frank et al., 2016) were used to make a broad and balanced comparison. The goal was to encompass substantially different learning mechanisms and inductive biases, to diminish dependence on any single classification paradigm.

The selected classifiers were:

Naïve Bayes, representing probabilistic learning.

Logistic Regression, SMO, and Multilayer Perceptron, representing linear, margin-based, and neural function learners.

IBk, representing instance-based learning.

Bagging, AdaBoostM1, and Random Forest, representing ensemble methods.

JRip, representing rule-based learning.

J48, representing decision-tree induction.

HyperPipes, included as a simple low-complexity baseline.

This selection includes interpretable models, probabilistic approaches, distance-based methods, nonlinear learning algorithms, ensemble strategies, and rule-based systems. Such a selection allows us to examine the effect of feature reduction across a variety of classifier families. All experiments were run on Weka 3.9.6 using the default parameter settings.

Predictive performance was evaluated using five commonly used measures: true positive rate (TPR), false positive rate (FPR), recall, ROC area, and PRC area. To compare the three data representations (all attributes, minimum-length reducts, and maximum-length reducts) across multiple datasets, the Friedman test followed by the Nemenyi post-hoc procedure was employed.

3 Results

Each classifier was evaluated under three feature representations for every dataset: the full set of attributes (All), a minimum-length reduct (Min), and a maximum-length reduct (Max). Predictive performance was measured using TPR, FPR, Recall, ROC area, and PRC area.

Differences among the three representations were assessed using the Friedman test followed by the Nemenyi post-hoc procedure. Table 2 reports the corresponding p-values, whereas Table 3 summarizes statistically significant pairwise differences at the 0.05 significance level.

Table 2. Friedman–Nemenyi post-hoc p-values for pairwise comparisons among full representation, minimum-length reducts, and maximum-length reducts across five evaluation measures.

	TPR			FPR			Recall			ROC area			PRC area		
Classifier	Full vs. Max	Full vs. Min	Max vs. Min	Full vs. Max	Full vs. Min	Max vs. Min	Full vs. Max	Full vs. Min	Max vs. Min	Full vs. Max	Full vs. Min	Max vs. Min	Full vs. Max	Full vs. Min	Max vs. Min
AdaBoost	**0.003**	**0.001**	0.810	**0.002**	**0.002**	0.782	**0.003**	**0.001**	0.810	**0.001**	**0.000**	0.577	**0.001**	**0.000**	0.198
Bagging	**0.001**	**0.000**	0.375	**0.001**	**0.000**	0.545	**0.001**	**0.000**	0.375	**0.001**	**0.000**	0.144	**0.000**	**0.000**	**0.010**
IBk	**0.000**	**0.000**	0.179	**0.000**	**0.017**	0.681	**0.000**	**0.000**	0.179	**0.001**	**0.026**	0.779	**0.005**	**0.020**	0.166
J48	**0.002**	**0.001**	0.524	**0.001**	**0.000**	0.641	**0.002**	**0.001**	0.524	**0.022**	**0.000**	0.475	**0.026**	**0.000**	0.096
Naïve Bayes	**0.011**	**0.001**	0.051	**0.001**	**0.000**	**0.043**	**0.011**	**0.001**	0.051	**0.000**	**0.000**	**0.027**	**0.000**	**0.000**	**0.002**
Random Forest	**0.000**	**0.000**	**0.012**	**0.000**	**0.000**	0.098	**0.000**	**0.000**	**0.012**	**0.000**	**0.000**	0.066	**0.000**	**0.000**	**0.017**
HyperPipes	**0.000**	**0.000**	0.709	**0.001**	**0.001**	0.862	**0.000**	**0.000**	0.709	**0.000**	**0.000**	0.516	**0.000**	**0.000**	0.758
JRip	0.056	0.272	**0.025**	0.241	0.139	0.244	0.056	0.272	**0.025**	0.191	0.155	0.119	**0.035**	0.236	**0.022**
Logistic Regression	0.113	0.155	**0.026**	0.299	0.213	**0.030**	0.113	0.155	**0.026**	0.060	0.217	**0.032**	**0.005**	0.196	**0.007**
MLP	0.260	0.173	**0.033**	0.123	0.260	0.144	0.260	0.173	**0.033**	0.186	**0.035**	**0.006**	**0.028**	0.071	**0.005**
SMO	**0.001**	0.128	**0.012**	**0.002**	0.252	**0.025**	**0.001**	0.128	**0.012**	**0.003**	0.206	**0.013**	**0.002**	0.064	**0.003**

Table 3. Summary of statistically significant pairwise differences among feature representations at the 0.05 level. Arrows indicate the preferred representation.

Classifier	TPR			FPR			Recall			ROC area			PRC area		
	Full vs. Max	Full vs. Min	Max vs. Min	Full vs. Max	Full vs. Min	Max vs. Min	Full vs. Max	Full vs. Min	Max vs. Min	Full vs. Max	Full vs. Min	Max vs. Min	Full vs. Max	Full vs. Min	Max vs. Min
AdaBoost	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔
Bagging	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇐
IBk	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔
J48	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔
Naïve Bayes	⇐	⇐	⇔	⇐	⇐	⇐	⇐	⇐	⇔	⇐	⇐	⇐	⇐	⇐	⇐
Random Forest	⇐	⇐	⇐	⇐	⇐	⇔	⇐	⇐	⇐	⇐	⇐	⇔	⇐	⇐	⇐
HyperPipes	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔	⇐	⇐	⇔
JRip	⇔	⇔	⇒	⇔	⇔	⇔	⇔	⇔	⇒	⇔	⇔	⇔	⇐	⇔	⇒
Logistic Regression	⇔	⇔	⇒	⇔	⇔	⇒	⇔	⇔	⇒	⇔	⇔	⇒	⇐	⇔	⇒
MLP	⇔	⇔	⇒	⇔	⇔	⇔	⇔	⇔	⇒	⇔	⇒	⇒	⇐	⇔	⇒
SMO	⇐	⇔	⇒	⇐	⇔	⇒	⇐	⇔	⇒	⇐	⇔	⇒	⇐	⇔	⇒

Viewed as a whole, the results indicate that classifier families do not respond uniformly to reduct-based dimensionality reduction. For a number of methods, both reduct representations performed worse than the complete attribute set. Others were less sensitive, and in some cases the minimum-length reducts gave better results than the maximum-length alternatives.

For AdaBoost, Bagging, IBk, J48, Naïve Bayes, Random Forest, and HyperPipes, the full representation was generally the strongest option across most metrics. JRip, Logistic Regression, and MLP were more stable, often showing similar performance regardless of the representation used.

SMO behaved differently from the remaining classifiers. Maximum-length reducts often reduced performance, whereas minimum-length reducts were usually comparable to the full representation. When the two reduct types were compared directly, the minimum-length reducts were more frequently favored.

4 Discussion

The results show that the impact of reduct-based dimensionality reduction is not uniform across classifiers. For many methods, e.g., AdaBoost, Bagging, IBk, J48, Naïve Bayes, Random Forest, and HyperPipes, in a large number of cases, the minimum-length and maximum-length reducts performed worse than the full feature representation.

These methods differ considerably in their learning mechanisms, but many of them seem to benefit from information distributed over multiple attributes, combinations of weak signals, or geometric relationships among variables. When that wider structure is pruned, predictive performance may be lost, but the rough-set consistency condition is still preserved.

A different pattern was observed for JRip, Logistic Regression, and Multilayer Perceptron, which frequently showed performance statistically comparable to that obtained with the full representation. This suggests that some classifiers are more tolerant of dimensionality reduction; this can occur when the primary discriminative structure (or information) remains present in a smaller subset of attributes.

In several cases, minimum-length reducts were also preferable to maximum-length reducts. A possible explanation is that compact reducts reduce redundancy and noise, leading to simpler and better-conditioned

learning problems. By contrast, larger reducts may retain additional attributes that do not contribute useful information and may even hinder generalization.

SMO behaved differently from most of the other classifiers. Maximum-length reducts often underperformed relative to the full representation, whereas minimum-length reducts were usually comparable to it. In direct comparisons, the minimum-length reducts were also favored more often than the maximum-length ones.

This may be related to the sensitivity of support vector methods to feature-space geometry. Eliminating redundant dimensions can sometimes be preferable to preserving a larger representation that also carries additional noise.

More generally, these results can be viewed through the lens of inductive bias. Reducts preserve discernibility, but they do not preserve every type of structure used by learning algorithms. Distance-based techniques rely on the neighborhood relation. Ensembles can benefit from multiple weak predictors. Probabilistic classifiers often use more extensive distributional information. Linear/rule-based/margin-based models, on the other hand, work well when most of the predictive signal is concentrated on a relatively small set of variables.

From the bias-variance point of view, smaller reducts may reduce variance and overfitting, but they may increase bias if relevant complementary information is eliminated. This may be the reason why the same dimensionality reduction strategy may harm some families of classifiers and help others.

While the present study concentrates on predictive performance evaluation, the reduct-based dimensionality reduction can also offer computational advantages. Smaller reducts can reduce training or prediction times (especially for distance-based methods such as k-nearest neighbors) and reduce memory requirements. The benefits of such reductions, however, must be weighed against the cost of computing reducts, which can be considerable depending on the dataset and the reduction algorithm used.

5 Conclusions

In this work, we asked whether reduct size makes a meaningful difference to supervised classification. To study this, we compared the full feature sets with the minimum- and maximum-length reducts across twenty-one benchmark datasets, eleven classifiers, and five evaluation criteria.

The evidence suggests that there is no single answer for all classifiers. The full representation was often the best choice for AdaBoost, Bagging, IBk, J48, Naïve Bayes, Random Forest, and HyperPipes. Here, we see that reducing the feature space seems to eliminate information used for prediction.

Some methods did not act the same way. For example, the performance of JRip, Logistic Regression, and the Multilayer Perceptron remained stable after reduction. It was surprising that, sometimes, the shorter reducts were equal to or better than the longer ones. This point makes clear that performance is not solely a function of reduct length.

SMO showed a distinct behavior. It should be noted that the shorter reducts generally performed better than longer ones overall.

These results suggest an important practical issue: the feature selection for the dimensionality reduction with a reduct-based method must be aware of the particular classifier to be used. What works for one classifier may not work for another, and thus they need to be evaluated together for best results. That is, a good reduct for one of the classifiers may not be a good choice for the other, and vice versa.

Shorter reducts are especially useful if the goal is to choose a simpler model, as shorter reducts improve the interpretability and decrease the required computational resources. However, the main feature to consider is still the predictability of the model.

Some questions remain to be resolved. For example, it would be interesting to study some repeated resampling schemes, to compare the performance of several reducts of the same size, or to study other properties of reducts such as stability, granularity, or robustness.

References

Chen, Y., Zhu, Q., & Xu, H. (2015). Finding rough set reducts with fish swarm algorithm. *Knowledge-Based Systems, 81*, 22–29. https://doi.org/10.1016/j.knosys.2015.02.002

Choromański, M., Grześ, T., & Hońko, P. (2020). Breadth search strategies for finding minimal reducts: Towards hardware implementation. *Neural Computing and Applications, 32*, 14801–14816. https://doi.org/10.1007/s00521-020-04833-7

Frank, E., Hall, M. A., & Witten, I. H. (2016). *The WEKA workbench: Online appendix for "Data mining: Practical machine learning tools and techniques"* (4th ed.). Morgan Kaufmann.

González-Díaz, Y., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A., & Lazo-Cortés, M. S. (2024). An algorithm for computing all rough set constructs for dimensionality reduction. *Mathematics, 12*(1), Article 90. https://doi.org/10.3390/math12010090

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research, 3*, 1157–1182.

Hu, K., Diao, L., Lu, Y., & Shi, C. (2000). A heuristic optimal reduct algorithm. In K. S. Leung, L.-W. Chan, & H. Meng (Eds.), *Intelligent data engineering and automated learning—IDEAL 2000* (Lecture Notes in Computer Science, Vol. 1983, pp. 139–144). Springer. https://doi.org/10.1007/3-540-44491-2_21

Jensen, R., & Shen, Q. (2003). Finding rough set reducts with ant colony optimization. In *Proceedings of the 2003 UK Workshop on Computational Intelligence* (pp. 15–22).

Kelly, M., Longjohn, R., & Nottingham, K. (n.d.). *The UCI Machine Learning Repository*. Retrieved September 1, 2026, from https://archive.ics.uci.edu

Lazo-Cortés, M. S., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A., & Sanchez Diaz, G. (2016). A new algorithm for computing reducts based on the binary discernibility matrix. *Intelligent Data Analysis, 20*(2), 317–337. https://doi.org/10.3233/IDA-160807

Niu, J., Chen, D., Li, J., & Wang, H. (2022). Fuzzy rule-based classification method for incremental rule learning. *IEEE Transactions on Fuzzy Systems, 30*(9), 3748–3761. https://doi.org/10.1109/TFUZZ.2021.3128061

Pawlak, Z. (1982). Rough sets. *International Journal of Computer & Information Sciences, 11*(5), 341–356. https://doi.org/10.1007/BF01001956

Pawlak, Z. (1991). *Rough sets: Theoretical aspects of reasoning about data*. Kluwer Academic Publishers. https://doi.org/10.1007/978-94-011-3534-4

Rodríguez-Diez, V., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A., Lazo-Cortés, M. S., & Olvera-López, J. A. (2020). MinReduct: A new algorithm for computing the shortest reducts. *Pattern Recognition Letters, 138*, 177–184. https://doi.org/10.1016/j.patrec.2020.07.004

Rodríguez-Diez, V., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A., Lazo-Cortés, M. S., & Olvera-López, J. A. (2021). A comparative study of two algorithms for computing the shortest reducts: MiLIT and MinReduct. In E. Roman-Rangel, Á. F. Kuri-Morales, J. F. Martínez-Trinidad, J. A. Carrasco-Ochoa, & J. A. Olvera-López (Eds.), *Pattern recognition: 13th Mexican Conference, MCPR 2021* (Lecture Notes in Computer Science, Vol. 12725, pp. 57–67). Springer. https://doi.org/10.1007/978-3-030-77004-4_6

Stańczyk, U. (2023). How transformations of representation for input data can affect the properties of induced decision reducts and rules. *Procedia Computer Science, 225*, 3603–3612. https://doi.org/10.1016/j.procs.2023.10.355