Software Impacts 15 (2023) 100485

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

XyGen: Synthetic data generator for feature selection

Firuz Kamalov [a,*], Said Elnaffar [a], Hana Sulieman [b], Aswani Kumar Cherukuri [c]

[a] Canadian University Dubai, Dubai, United Arab Emirates
[b] American University of Sharjah, Sharjah, United Arab Emirates
[c] Vellore Institute of Technology, Vellore, India

ARTICLE INFO

Keywords:
Feature selection
Synthetic data
Machine learning
Data mining

ABSTRACT

Given the large number of feature selection algorithms, it has become imperative to have a uniform procedure for evaluating the performance of the algorithms. We propose a library of synthetic datasets designed specifically to test the effectiveness of feature selection algorithms. The datasets are inspired by applications in the field of electronics and have a range of characteristics to provide a variety of test scenarios. The software comes in the form of a Python library with standard interface for loading and generating datasets. Each dataset is implemented as a function that allows control of various parameters of the data.

Code metadata

Current code version	*2.0*
Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2023-41
Permanent link to Reproducible Capsule	https://codeocean.com/capsule/8977698/tree/v1
Legal Code License	*MIT License*
Code versioning system used	*Git*
Software code languages, tools, and services used	*Python, Pandas, Numpy*
Compilation requirements, operating environments & dependencies	*c ≥ 3.9, Numpy, Pandas*
If available Link to developer documentation/manual	https://github.com/SaidElnaffar/Synthetic-Datasets-for-Features-Selection-Algorithms/blob/5439faad6bba70fc0e22910ae56cd0372ea7c0bc/README.md
Support email for questions	firuz@cud.ac.ae, said.elnaffar@cud.ac.ae

1. Synthetic data generator for feature selection

Feature selection has been an active area of research with dozens of new algorithms being proposed every year. In this software package, we provide a Python library for generating synthetic datasets that are designed specifically to test the effectiveness of feature selection algorithms. The library consists of functions that allow to load and generate 5 different datasets. Each dataset consists of a number of relevant, redundant, correlated, and irrelevant variables. The target variable is calculated based on a predetermined rule/formula using the relevant features as described in [1]. The redundant variables are linear transformations of the relevant features, while the correlated features are obtained by randomly flipping 30% of the target variable labels. The library functions allow to specify dataset parameters such as the number of irrelevant variables, the number of instances, and the random seed. Since the relevant variables are known a priori, synthetic data enables direct evaluation of feature selection algorithms. To mimic real life scenarios, the datasets are inspired by applications in the field of electronics.

Feature selection has become an important part of the process in many data science and machine learning applications. As a result, a large number of feature selection algorithms have been proposed in the literature [2–6]. However, there does not exist a universal benchmark for evaluating these algorithms. To fill this gap, we propose a software package called XyGen that allows to generate synthetic data tailored explicitly to assess feature selection algorithms. We aim that the proposed software package and the corresponding datasets would be used in evaluating the existing and future feature selection algorithms and

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author.
E-mail addresses: firuz@cud.ac.ae (F. Kamalov), said.elnaffar@cud.ac.ae (S. Elnaffar), hsulieman@aus.edu (H. Sulieman), cherukuri@acm.org (A.K. Cherukuri).

https://doi.org/10.1016/j.simpa.2023.100485
Received 24 January 2023; Received in revised form 20 February 2023; Accepted 23 February 2023

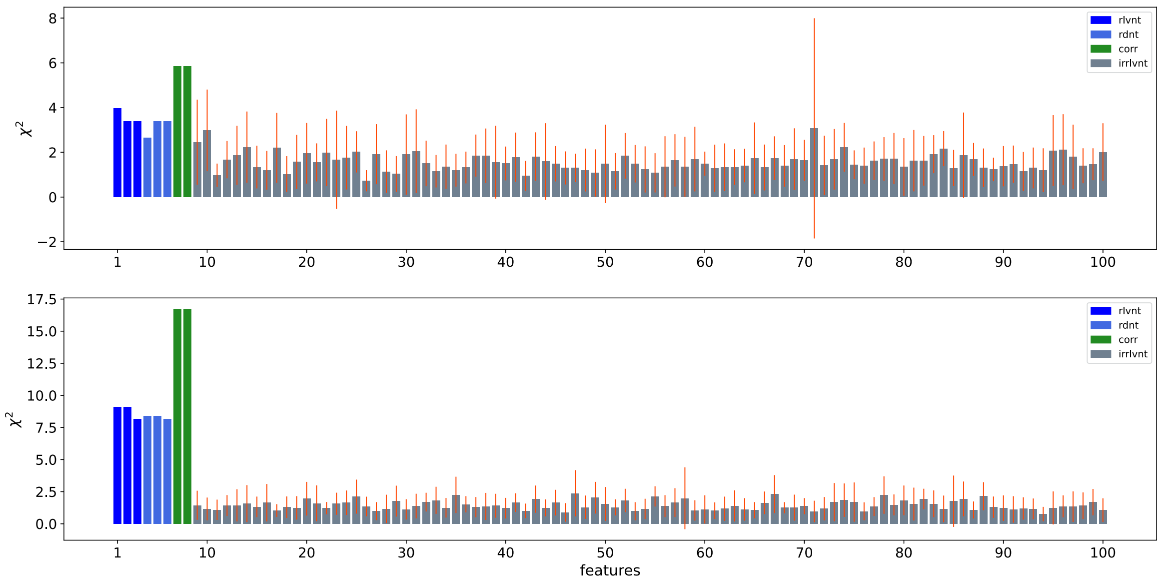

Fig. 1. Results of χ^2-univariate feature selection based on the ADDER dataset generated using XyGen with sample size 20 (top) and 50 (bottom).

Table 1
Summary the XyGen datasets.

Name	Relevant	Redundant	Correlated	Irrelevant	Samples	Target
ORAND	3	3	2	92	50	Binary
ANDOR	4	4	2	90	50	Binary
ADDER	3	3	2	92	50	4-class
LED-16	16	16	2	66	180	36-class
PRC	5	5	2	88	500	Continuous

Table 2
Rankings of the first 10 features in the ADDER dataset according to various feature selection algorithms.

	1	2	3	4	5	6	7	8	9	10
Boruta [7,8]	1	1	1	1	1	1	1	1	61	63
reliefF [9,10]	6	6	0	3	1	3	2	5	51	52
Fisher [11,12]	6	6	3	4	0	1	5	2	61	48
mRMR [13,14]	6	0	1	2	7	51	3	4	5	8
CIFE [15,16]	6	0	1	2	41	4	5	13	33	39
RFS [17,18]	61	34	63	44	35	66	35	35	25	59
F-score [3]	7	6	0	1	3	4	2	5	61	48
RFE [19,20]	6	5	4	3	1	2	14	8	56	57
GA_1 [21]	1	1	0	0	1	1	1	0	0	0
GA_2 [22]	0	0	0	1	0	0	0	1	0	0

provide a standard approach to measure and analyze the effectiveness of algorithms.

A summary of the datasets generated via XyGen is presented in Table 1. The table shows the default parameter values of the datasets. The datasets include different types of the target variable including binary, multi-class, and continuous values. While the number of relevant, redundant, and correlated features is fixed, the number of irrelevant features and the sample size can be specified through the corresponding data generating functions. In addition, the random seed can be specified to generate different irrelevant features for algorithm stability analysis. The details of the datasets used in the XyGen library can be found in [1].

2. Impact and use cases

The majority of the existing synthetic datasets used to evaluate feature selection algorithms were originally designed for classification tasks [23–29]. On the other hand, the XyGen data is designed specifically for use in feature selection. The XyGen data includes redundant as well as correlated features to provide a rich setting to test the algorithms. There are two primary advantages of using synthetic data over real life data: (i) the knowledge of the relevant features, and (ii) control of the data characteristics. In the traditional approach using real life data, feature selection algorithms are evaluated based on the accuracy of the classifier trained on the selected features. On the other hand, the nature of all the variables in synthetic data is known so the selected features can be evaluated directly.

As an example, the ANDOR dataset generated via XyGen was used to compare the performance of several feature selection algorithms in [1]. The study showed that while most of the algorithms are able to distinguish between the relevant and irrelevant variables, they fail to separate the relevant variables from the redundant and correlated variables. Several XyGen generated datasets were used to evaluate the performance of a new feature selection algorithm called Nested Ensemble Selection.

Synthetic data enables an in-depth analysis of feature selection algorithms by controlling the parameters of the dataset. In particular, XyGen allows to specify the number of irrelevant features and the size of the dataset. By varying the number of irrelevant variables the corresponding performance of the selection algorithm can be observed and analyzed [30]. Similarly, the sensitivity of an algorithm to the size of the dataset can be investigated by varying the number of instances in XyGen.

As an illustration of the use of XyGen, consider the results of applying the χ^2 univariate feature selection algorithm on the ADDER dataset. The results are shown in Fig. 1, where the top and bottom subfigures are based on sample size 20 and 50, respectively. It shows that increase in the sample size increases the difference between the relevant and irrelevant features. On the other hand, the algorithm fails to distinguish between the relevant and redundant variables. It also shows that the algorithm assigns high scores to the (randomly) correlated variables. Thus, we obtain a better understanding of the performance of the algorithm and its characteristics.

To illustrate the application of XyGen for comparative studies, consider the results of feature selection on the ADDER dataset using ten popular algorithms. The rankings of the first 10 features are presented in Table 2, where the features 1–3 are relevant, features 4–6 are redundant, features 7–8 are correlated, and features 9–10 are irrelevant. Note that the genetic algorithms provide only the feature support. It can be seen in Table 2 that the genetic algorithm GA_1 produces the most robust results with only the redundant feature 5 included incorrectly in the selected subset. On the other hand, the RFS algorithm completely

Table 3
Rankings of the first 15 features in the PRC dataset according to various feature selection algorithms.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Boruta	1	1	1	1	1	1	1	1	1	1	1	1	52	4	44
R_regression	40	46	37	51	35	3	4	3	5	3	3	2	76	71	75
F_regression	32	27	40	28	29	3	3	4	3	4	2	4	85	80	62
mRMR	10	48	8	1	0	11	3	9	6	5	2	30	4	99	7
RFS	78	65	43	60	62	52	79	64	59	68	50	73	31	46	51
RFE	100	99	98	97	96	95	94	93	92	91	90	89	88	87	86
GA_2	0	0	0	0	1	0	0	0	0	0	1	1	0	0	1

fails to identify the relevant features. Another important observation is that most of the algorithms fail to properly differentiate the relevant features from the redundant and correlated features. The above analysis is made possible by the knowledge of the variables and the different types of features incorporated in the ADDER dataset. While XyGen generates synthetic datasets, it is derived from real life applications in electronics which enhances their plausibility.

The XyGen package can also be used to generate data with continuous valued target variable. In particular, the target variable in the PRC dataset is based on the cumulative resistance in a parallel connected circuit which takes a continuous value [1]. In Table 3, we present the results of applying several feature selection algorithms on the PRC dataset. It can be seen that while F_regression, R_regression, and mRMR assign importance to the relevant variables, they fail to discard the correlated features. The RFS, RFE, and genetic algorithms perform poorly.

3. Conclusion and future development

In this report, we presented a Python package called XyGen which allows to generate synthetic data designed for feature selection. While XyGen is aimed at evaluating feature selection algorithms, it can also be used for classification and regression tasks. For example, researchers can analyze the performance of a classification model with different sample sizes or number of irrelevant variables.

As part of future development, we aim to expand the collection of the datasets in XyGen. In addition, we hope to establish a forum where researchers can share the results of feature selection algorithms based on XyGen datasets.

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.

References

[1] F. Kamalov, H. Sulieman, A.K. Cherukuri, Synthetic data for feature selection, 2022, arXiv preprint arXiv:2211.03035.

[2] A. Alsahaf, N. Petkov, V. Shenoy, G. Azzopardi, A framework for feature selection through boosting, Expert Syst. Appl. 187 (2022) 115895.

[3] A. Bommert, X. Sun, B. Bischl, J. Rahnenführer, M. Lang, Benchmark for filter methods for feature selection in high-dimensional classification data, Comput. Statist. Data Anal. 143 (2020) 106839.

[4] F. Kamalov, Orthogonal variance decomposition based feature selection, Expert Syst. Appl. 182 (2021) 115191.

[5] F. Kamalov, F. Thabtah, H.H. Leung, Feature selection in imbalanced data, Ann. Data Sci. (2022) 1–15.

[6] B. Remeseiro, V. Bolon-Canedo, A review of feature selection methods in medical applications, Comput. Biol. Med. 112 (2019) 103375, Chicago.

[7] M.B. Kursa, W.R. Rudnicki, Feature selection with the Boruta package, J. Stat. Softw. 36 (2010) 1–13.

[8] R. Tang, X. Zhang, Cart decision tree combined with boruta feature selection for medical data classification, in: 2020 5th IEEE International Conference on Big Data Analytics (ICBDA), IEEE, 2020, pp. 80–84.

[9] I. Kononenko, E. Šimec, M. Robnik-Šikonja, Overcoming the myopia of inductive learning algorithms with RELIEFF, Appl. Intell. 7 (1) (1997) 39–55.

[10] L. Sun, T. Yin, W. Ding, Y. Qian, J. Xu, Multilabel feature selection using ML-relieff and neighborhood mutual information for multilabel neighborhood decision systems, Inform. Sci. 537 (2020) 401–424.

[11] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection, Adv. Neural Inf. Process. Syst. 18 (2005).

[12] M. Li, H. Wang, L. Yang, Y. Liang, Z. Shang, H. Wan, Fast hybrid dimensionality reduction method for classification based on feature selection and grouped feature extraction, Expert Syst. Appl. 150 (2020) 113277.

[13] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (8) (2005) 1226–1238.

[14] X. Yan, M. Jia, Intelligent fault diagnosis of rotating machinery using improved multiscale dispersion entropy and mRMR feature selection, Knowl.-Based Syst. 163 (2019) 450–471.

[15] D. Lin, X. Tang, Conditional infomax learning: an integrated framework for feature extraction and fusion, in: European Conference on Computer Vision, Springer, Berlin, Heidelberg, 2006, pp. 68–82.

[16] G. Wei, J. Zhao, Y. Feng, A. He, J. Yu, A novel hybrid feature selection method based on dynamic feature importance, Appl. Soft Comput. 93 (2020) 106337.

[17] F. Nie, H. Huang, X. Cai, C. Ding, Efficient and robust feature selection via joint l2, 1-norms minimization, Adv. Neural Inf. Process. Syst. 23 (2010).

[18] Z. Zhang, Y. Xu, J. Yang, X. Li, D. Zhang, A survey of sparse representation: algorithms and applications, IEEE Access 3 (2015) 490–530.

[19] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines, Mach. Learn. 46 (1) (2002) 389–422.

[20] M. Lee, J.H. Lee, D.H. Kim, Gender recognition using optimal gait feature based on recursive feature elimination in normal walking, Expert Syst. Appl. 189 (2022) 116040.

[21] S.S. Shreem, H. Turabieh, S. Al Azwari, F. Baothman, Enhanced binary genetic algorithm as a feature selection to predict student performance, Soft Comput. 26 (4) (2022) 1811–1823.

[22] F.A. Fortin, F.M. De Rainville, M.A.G. Gardner, M. Parizeau, C. Gagné, DEAP: Evolutionary algorithms made easy, J. Mach. Learn. Res. 13 (1) (2012) 2171–2175.

[23] L.A. Belanche, F.F. González, Review and evaluation of feature selection algorithms in synthetic problems, 2011, arXiv preprint arXiv:1101.2320.

[24] V. Bolon-Canedo, N. Sánchez-Marono, A. Alonso-Betanzos, A review of feature selection methods on synthetic data, Knowl. Inf. Syst. 34 (3) (2013) 483–519.

[25] G.H. John, R. Kohavi, K. Pfleger, Irrelevant features and the subset selection problem, Mach. Learn. Proc. 1994 (1994) 121–129.

[26] G. Kim, Y. Kim, H. Lim, H. Kim, An MLP-based feature subset selection for HIV-1 protease cleavage site analysis, Artif. Intell. Med. 48 (2–3) (2010) 83–89.

[27] A. Mamalakis, I. Ebert-Uphoff, E.A. Barnes, Neural network attribution methods for problems in geoscience: A novel synthetic benchmark dataset, Environ. Data Sci. 1 (2022) e8.

[28] A. Torfi, E.A. Fox, C.K. Reddy, Differentially private synthetic medical data generation using convolutional gans, Inform. Sci. 586 (2022) 485–500.

[29] X. Wang, L. Xie, C. Dong, Y. Shan, Real-esrgan: Training real-world blind super-resolution with pure synthetic data, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 1905–1914.

[30] U.M. Khaire, R. Dhanalakshmi, Stability of feature selection algorithm: A review, J. King Saud Univ.-Comput. Inform. Sci. 34 (4) (2022) 1060–1073.