ESANN 2018 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2018, i6doc.com publ., ISBN 978-287587047-6.
Available from http://www.i6doc.com/en/.

A Novel Filter Algorithm for Unsupervised Feature Selection Based on a Space Filling Measure

Mohamed Laib and Mikhail Kanevski *

University of Lausanne - Institute of Earth Surface Dynamics, Switzerland
Email: Mohamed.Laib@unil.ch

Abstract. The research proposes a novel filter algorithm for the unsupervised feature selection problems based on a space filling measure. A well-known criterion of space filling design, called the coverage measure, is adapted to dimensionality reduction problems. Originally, this measure was developed to judge the quality of a space filling design. In this work it is used to reduce the redundancy in data. The proposed algorithm is evaluated on simulated data with several scenarios of noise injection. Furthermore, a comparison with some benchmark methods of feature selection is performed on real UCI datasets.

1 Introduction

The ability of collecting and storing high dimensional data is improving very fast. However, the collected data could contain redundancy that makes hard the understanding, the visualisation, and the interpretability of the phenomenon under study. Redundancy means that the features are dependent on each other, which causes (1) a reduction of the learning accuracy; (2) computational issues due to the hardware performance limitations; and (3) lack of information on features that characterise the data. To overcome these issues, many feature selection (FS) algorithms have been proposed aiming at reducing the dimensionality and redundancy in data [1, 2].

The literature of machine learning distinguishes two well-known techniques of FS according to the availability of the output (supervised and unsupervised feature selection (UFS)). UFS aims at reducing the redundancy that can be either linear or non-linear. To do that, there are two categories of UFS algorithms: wrappers and filters. The former includes a clustering algorithm during the dimensionality reduction, while the latter do not include a learning algorithm since they use a defined criterion to detect the existing redundancy.

In this work, a well-known criterion of space filling design, called the coverage measure (CM), is adapted to the unsupervised feature selection problems. The CM was used to choose the best space filling design by comparing it to the uniform distribution. Some of space filling designs produced using CM can be found in [3, 5].

The remainder of this paper is organised as follows. Section 2 presents the coverage measure, Section 3 describes the proposed algorithm based on the CM.

*The research was partly supported by the Swiss Government Excellence Scholarships.

ESANN 2018 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2018, i6doc.com publ., ISBN 978-287587047-6.
Available from http://www.i6doc.com/en/.

Numerical studies of simulated and real datasets are discussed in Section 4. Finally, the conclusions with future developments are given.

2 The coverage measure

Design and modelling of experiments have always been a fundamental task over years. One of the most important step is to check the coverage or the uniformity of the proposed design by applying several criteria. Among these criteria, the coverage measure was used to choose the best set of points that cover well the domain of interest [6].

Definition (from [7])

Let $X = \left\{x^1, \ldots, x^n\right\} \subset [0,1]^d$ be a set of n points of dimension d. The coverage measure is defined as follows:

$$\lambda = \frac{1}{\bar{\vartheta}} \left(\frac{1}{n} \sum_{i=1}^{n} (\vartheta_i - \bar{\vartheta})^2 \right)^{\frac{1}{2}} \tag{1}$$

where: $\vartheta_i = \min_{(k \neq i)} \left(dist\left(x^i, x^k\right)\right)$ is the minimal distance between x^i and the other points of the sequence, $\bar{\vartheta} = \frac{1}{n} \sum_{i=1}^{n} \vartheta_i$ is the mean of ϑ_i, and $dist$ is the Euclidean distance.

If the distribution of data points is close to the regular grid distribution, then $\vartheta_1 = \vartheta_2 = \ldots = \vartheta_n = \bar{\vartheta}$ and $\lambda = 0$.

The coverage of points can be quantified and detected by using a criterion based on a minimum Euclidean distance between them. In fact, the coverage measure λ represents the coefficient of variation of the ϑ_i, which is known as the relative standard deviation (the ratio of the standard deviation to the mean of ϑ_i). The best design should have the smallest coverage value λ [7] (Fig. 1). In the present research this metric is adapted to the selection of the best subset of features by reducing the existing redundancy in data (see algorithm 1). Another adaptation, of a space filling criterion, was proposed in [8].

Fig. 1: Different sequences of points with λ: 0.29, 0.52, and 0.65 for Halton, Sobol, and uniform set respectively

ESANN 2018 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2018, i6doc.com publ., ISBN 978-287587047-6.
Available from http://www.i6doc.com/en/.

3 Redundancy reduction using the coverage measure

This section describes the adaptation of the CM to reduce the existing redundancy in data. The particular FS strategy considered is implemented in a sequential forward search. Fig. 2 presents an illustration of a redundancy detection by the CM.

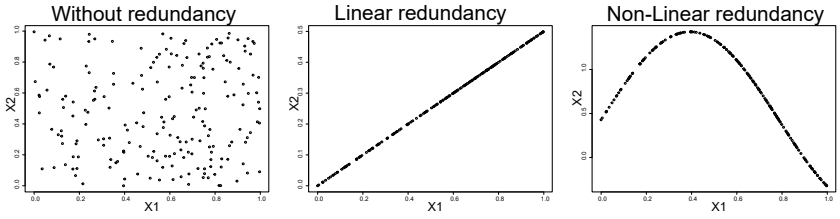

Fig. 2: Simulated data with random (or non-redundant), linear and non-linear redundancies, respectively. Corresponding CM values: $\lambda = [0.51, 1.06, 1.107]$

Algorithm 1 UfsCov algorithm

Input:
Dataset D with d features $X_{1,\ldots,d}$.
Empty vectors Id and CM
Output:
Id and CM respectively, the features id and the coverage values.

1: Rescale data to $[0, 1]$.
2: **for** $i = 1$ **to** d **do**
3: **for** $j = 1$ **to** $(d + 1 - i)$ **do**
4: $\lambda = Coverage(D[Id, j])$
5: **end for**
6: The lowest value of λ is stored in $CM[i]$
7: The corresponding id of the lowest λ is stored in $Id[i]$
8: **end for**

4 Experimental study

4.1 Simulated dataset

The simulated *Butterfly* dataset, introduced in [9], is composed of eight features $\{X_1, X_2, J_3, J_4, J_5, I_6, I_7, I_8\}$, where three of them $\{X_1, X_2, I_6\}$ are relevant and contain all the information of the dataset. The remaining five features are constructed basically from $\{X_1, X_2, I_6\}$ with linear and non-linear relations. In fact, these five features are redundant and do not bring new information. (See J. Golay et al. [10]).

Fig. 3 shows the results for the Butterfly datasets generated with different number of points N. In addition, the UfsCov algorithm is evaluated by injecting a Gaussian noise, with a mean fixed at 0 and a standard deviation set at: $1\%, 5\%, 10\%, 20\%$, and 50% of the original standard deviation of feature (see

ESANN 2018 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2018, i6doc.com publ., ISBN 978-287587047-6.
Available from http://www.i6doc.com/en/.

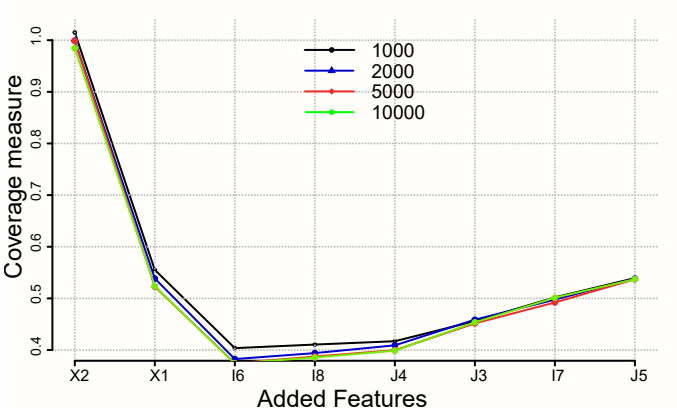

Fig. 3: The butterfly dataset results with different number of simulated points. The algorithm chooses the correct subset of features that characterise the dataset

Fig. 4). Table 1 illustrates the selected features for each standard deviation. The results demonstrate that UfsCov the algorithm is robust against noise.

It can be concluded, based on these experimental studies, that the proposed algorithm can easily detect the existing redundancy, both linear and linear, in data, and it is robust against a noise in data.

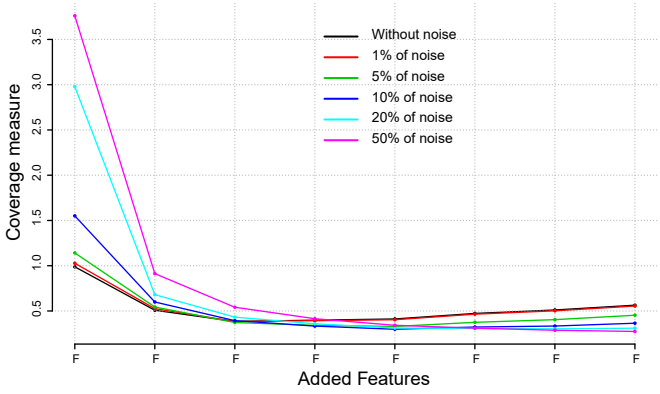

Fig. 4: The results of experiments with noise injected at different levels of standard deviation. See table 1 for the ranking of the selected features.

4.2 Real datasets

The proposed algorithm was applied to several well-known real world datasets, downloaded from the UCI machine learning repository (Parkinson, Ionosphere, Breast Cancer, and PageBlocks) [11]. In addition, the UfsCov algorithm was compared with three benchmark feature selection methods: Laplacian Score

ESANN 2018 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2018, i6doc.com publ., ISBN 978-287587047-6. Available from http://www.i6doc.com/en/.

	Without noise	1%	5%	10%	20%	50%
$F1$	X_1	X_1	X_2	X_1	X_1	I_6
$F2$	I_6	I_6	I_6	I_6	X_2	X_2
$F3$	X_2	X_2	X_1	X_2	I_6	X_1
$F4$	I_8	I_8	I_8	I_8	I_8	J_4
$F5$	J_4	J_4	J_4	J_4	J_4	J_5
$F6$	I_7	I_7	I_7	I_7	I_7	J_3
$F7$	J_5	J_5	J_5	J_5	J_5	I_8
$F8$	J_3	J_3	J_3	J_3	J_3	I_7

Table 1: Ranking of features selected by UfsCov algorithm.

Datasets	N	All feat.	# Sel. Feat.	Execution time (s)
Parkinson	195	22	7	0.22
Ionosphere	350	34	10	1.62
Breast Cancer	569	30	12	3.72
PageBlocks	5393	10	3	32.62

Table 2: Summary of the results obtained by the UfsCov algorithm on real datasets as well as the execution time (using R software on an Intel Core i7-7700K CPU @ 4.20 GHz with 32.0 GB of RAM under Windows 10). The column "# Sel. Feat." gives the number of feature selected by UfsCov.

(LScore) [12], Robust Unsupervised Feature Selection (RUFS) [13], and Multi-Cluster Feature Selection (MCFS) [14].

In order to evaluate the selected set of features, an external validation was applied by using random forest (RF) as a classifier. Such approach is a common practice used to evaluate an unsupervised feature selection algorithm. In fact, RF algorithm is applied twice, once with all features and once with only the selected features. RF was applied also on the features, selected by other methods. Tables 2 and 3 present a summary of the results.

The UfsCov algorithm contributes in reducing the redundancy in data. It can be concluded that it is a good tool for unsupervised feature selection problems, which makes a direct link between machine learning and sampling techniques.

Datasets	**UfsCov**	Lscore	RUFS	MCFS	All Feat.
Parkinson	0.91(0.03)	0.89(0.04)	0.92(0.05)	0.90(0.03)	0.91(0.04)
Ionosphere	0.91(0.05)	0.91(0.03)	0.91(0.03)	0.91(0.03)	0.91(0.03)
Breast Cancer	0.96(0.01)	0.96(0.01)	0.95(0.02)	0.93(0.02)	0.95(0.02)
PageBlocks	0.95(0.006)	0.96(0.004)	0.97(0.005)	0.97(0.006)	0.97(0.004)

Table 3: Overall accuracy and the standard deviation obtained after 20 repetitions of random forest.

5 Conclusion

In this work, a space filling metric is studied and adapted to the unsupervised feature selection problems, and implemented in a filter algorithm. The proposed filter algorithm was evaluated with simulated data and compared with

ESANN 2018 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 25-27 April 2018, i6doc.com publ., ISBN 978-287587047-6. Available from http://www.i6doc.com/en/.

benchmark methods using real datasets. The present work investigates the link between two well-known fields (space filling design and data mining). Further, preliminary results obtained are very promising. However, the proposed measure need to be studied in details to find the strengths and the limitations of its adaptation in data mining tasks. The UfsCov algorithm can be found in the R library SFtools, which is available on The Comprehensive R Archive Network (CRAN) [15] and on the GitHub repository (https://github.com/mlaib/SFtools).

References

[1] J. A. Lee, M. Verleysen, Nonlinear Dimensionality Reduction, Springer, New York, 2007.

[2] I. Guyon , S. Gunn , M. Nikravesh , L.A. Zadeh , Feature Extraction: Foundations and Applications, Springer, Berlin, 2006 .

[3] J. A. Royle, D. Nychka, An algorithm for the construction of spatial coverage designs with implementation in Splus, Computers and Geosciences 24 (1997) p. 479 - 488.

[4] D. Walvoort, D. Brus, J. de Gruijter, An R package for spatial coverage sampling and random sampling from compact geographical strata by k-means, Computers and Geosciences 36 (2010) p. 1261–1267.

[5] D. Dupuy, C. Helbert, J. Franco, DiceDesign and DiceEval: Two R Packages for Design and Analysis of Computer Experiments (2015).

[6] K. T. Fang, R. Li, A. Sudjianto, Design and Modeling for Computer Experiments, Taylor and Francis Group, 2006.

[7] J. Franco, Planification d'expériences numériques en phase exploratoire pour la simulation des phénomènes complexes, Thesis (2008) 282.

[8] D. Ballabio, V. Consonni, A. Mauri, M. Claeys-Bruno, M. Sergent, R. Todeschini, A novel variable reduction method adapted from space-filling designs, Chemometrics and Intelligent Laboratory Systems, Volume 136, 2014, p. 147-154.

[9] J. Golay, M. Laib, IDmining: Intrinsic Dimension for Data Mining, R package version 1.0.3 (2016). URL https://CRAN.R-project.org/package=IDmining.

[10] J. Golay, M. Leuenberger, M. Kanevski, Feature selection for regression problems based on the Morisita estimator of intrinsic dimension, Pattern Recognition 70 (2017) p. 126 - 138.

[11] M. Lichman, UCI Machine Learning Repository, University of California, School of Information and Computer Sciences, Irvine, 2013 . http://archive.ics.uci.edu/ml .

[12] X. He , D. Cai , P. Niyogi , Laplacian score for feature selection, Advances in Neu- ral Information Processing Systems, Vol. 18, MIT Press, Cambridge, USA, 2006.

[13] M. Qian , C. Zhai , Robust unsupervised feature selection, in: Proceedings of the 23rd International Joint Conference on Artificial Intelligence, 2013, p. 1621–1627.

[14] D. Cai , C. Zhang , X. He , Unsupervised feature selection for multi-cluster data, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2010, p. 333–342.

[15] M. Laib, M. Kanevski, SFtools: Space Filling Based Tools for Data Mining, R package version 0.1.0 (2017). URL https://CRAN.R-project.org/package=SFtools.