

CLeVer: A Feature Subset Selection Technique for Multivariate Time Series*

Kiyoung Yang, Hyunjin Yoon, and Cyrus Shahabi

Computer Science Department, University of Southern California,
Los Angeles, CA 90089, U.S.A
{kiyoungy, hjy, shahabi}@usc.edu

Abstract. Feature subset selection (FSS) is one of the data pre-processing techniques to identify a subset of the original features from a given dataset before performing any data mining tasks. We propose a novel FSS method for Multivariate Time Series (MTS) based on Common Principal Components, termed *CLeVer*. It utilizes the properties of the principal components to retain the correlation information among original features while traditional FSS techniques, such as Recursive Feature Elimination (RFE), may lose it. In order to evaluate the effectiveness of our selected subset of features, classification is employed as the target data mining task. Our experiments show that *CLeVer* outperforms RFE and Fisher Criterion by up to a factor of two in terms of classification accuracy, while requiring up to 2 orders of magnitude less processing time.

1 Introduction

Feature subset selection (FSS) is one of the techniques to pre-process the data before we perform any data mining tasks, e.g., classification or clustering. FSS is to identify a subset of original features from a given dataset while removing irrelevant and/or redundant features [1]. The objectives of FSS are to improve the prediction performance of the predictors, to provide faster and more cost-effective predictors, and to provide a better understanding of the underlying process that generated the data [2].

The FSS methods choose a subset of the original features to be used for the subsequent processes. Hence, only the data generated from those features need to be collected ignoring all the other features. This makes FSS different

* This research has been funded in part by NSF grants EEC-9529152 (IMSC ERC), IIS-0238560 (PECASE) and IIS-0307908, and unrestricted cash gifts from Microsoft. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. The authors would like to thank Dr. Carolee Winstein and Jarugool Tretiluxana for providing us the BCAR dataset and valuable feedbacks, and Thomas Navin Lal for providing us the BCI MPI dataset. The authors would also like to thank the anonymous reviewers for their valuable comments.

from feature *extraction*, where the correspondence information to the original features are in general not maintained, so all the original variables are required to be measured.

A time series is a series of observations, $x_i(t); [i = 1, \dots, n; t = 1, \dots, m]$, made sequentially through time where i indexes the measurements made at each time point t [3]. It is called a univariate time series when n is equal to 1, and a multivariate time series (MTS) when n is equal to, or greater than 2. An MTS item is naturally represented in an $m \times n$ matrix, where m is the number of observations and n is the number of *variables*, e.g., sensors. However, the state of the art FSS techniques, such as Recursive Feature elimination (RFE) [2], require each item to be represented in one row. Consequently, to utilize these techniques on MTS datasets, each MTS item needs to be first transformed into one row or column vector. However, since each of variables is considered separately during this *vectorization*, the correlation information among the features might be lost in the previous FSS method for MTS datasets [4].

In this paper, we propose a novel feature subset selection method for multivariate time series (MTS)¹ named *CLeVer* (descriptive **C**ommon principal component **L**oading based **V**ariable subset selection method). *CLeVer* utilizes the property of the principal components and common principal components to retain the correlation information among the variables.

2 Proposed Method

CLeVer is a novel variable subset selection method for multivariate time series (MTS) based on common principal component analysis (CPCA) [5, 6]. Figure 1 illustrates the entire process of *CLeVer*, which involves three phases: (1) principal components (PCs) computation per MTS item, (2) descriptive common principal components (DCPCs) computation per label² and their concatenation, and (3) variable subset selection using K -means clustering on DCPC loadings of variables. Each of these phases is described in the subsequent sections. Table 1 lists the notations used in the remainder of this paper, if not specified otherwise.

2.1 PC and DCPC Computations

The first and second phases (except the concatenation) of *CLeVer* are incorporated into Algorithm 1. It obtains both PCs and then DCPCs consecutively. The required input to Algorithm 1 is a set of MTS items with the same label.

Though there are n PCs for each item, only the first $p (< n)$ PCs, which are adequate for the purpose of representing each MTS item, are taken into consideration. Algorithm 1 takes in the threshold (δ) to determine p . That is,

¹ For multivariate time series, each *variable* is regarded as a feature [4]. Hence, the terms *feature* and *variable* are interchangeably used throughout this paper, when there is no ambiguity.

² The MTS datasets considered in our analysis are composed of labeled MTS items.

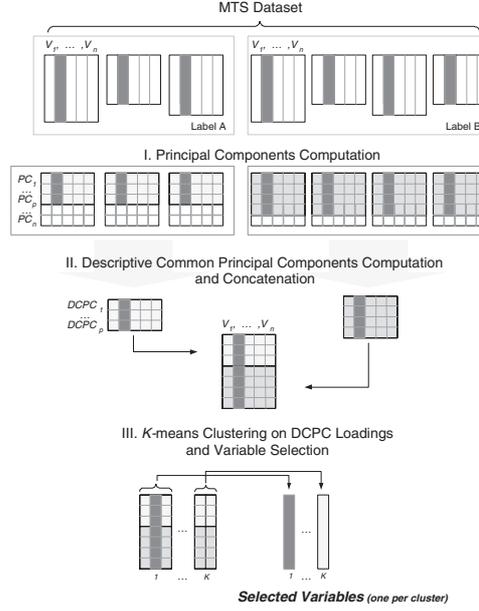


Fig. 1. The process of *CLeVer*

Table 1. Notations used in this paper

Symbol	Definition
N	number of MTS items in an MTS dataset
n	number of variables in an MTS item
K	number of clusters for K -means clustering
p	number of PCs for each MTS item to be used for computing DCPCs

for each input MTS item, p is determined to be the minimum value such that the ratio of the variances explained by its first p PCs to the total variance exceeds the provided threshold δ for the first time (Lines 3~10). Since the MTS items can have different values for p , p is finally determined as their maximum value (Line 11).

All MTS items are now described by their first p principal components. Let them be denoted as L_i ($i = 1, \dots, N$). Then, the DCPCs that *agree most closely* with all N sets of p PCs are successively defined by the eigenvectors of the matrix $H = \sum_{i=1}^N L_i^T L_i$ [5]. That is, $SVD(H) = SVD(\sum_{i=1}^N L_i^T L_i) = V\Lambda V^T$, where rows of V^T are eigenvectors of H and the first p of them define p DCPCs for N MTS items. This computation of DCPC is captured by Lines 16~17.

Algorithm 1. *ComputeDCPC*: PC and DCPC Computations

Require: MTS data groups with N items and δ {a predefined threshold}

- 1: $DCPC \leftarrow \emptyset$;
- 2: $H[0] \leftarrow \emptyset$;
- 3: **for** $i=1$ to N **do**
- 4: $X \leftarrow$ the i th MTS item;
- 5: $[U, S, U^T] \leftarrow$ SVD(correlation matrix of X);
- 6: $loading[i] \leftarrow U^T$;
- 7: $variance \leftarrow diag(S)$;
- 8: $percentVar \leftarrow 100 \times (variance / \sum_{j=1}^n variance_j)$;
- 9: $p_i \leftarrow$ number of the first p $percentVar$ elements whose cumulative sum $\geq \delta$;
- 10: **end for**
- 11: $p \leftarrow max(p_1, p_2, \dots, p_N)$;
- 12: **for** $i=1$ to N **do**
- 13: $L[i] \leftarrow$ the first p rows of $loading[i]$;
- 14: $H[i] \leftarrow H[i-1] + (L[i]^T \times L[i])$;
- 15: **end for**
- 16: $[V, S, V^T] \leftarrow$ SVD(H);
- 17: $DCPC \leftarrow$ first p rows of V^T ;

2.2 Variable Subset Selection

CLeVer utilizes a clustering method to group the *similar* variables together and finally to select the least redundant variables. First, the DCPCs per label are computed by Algorithm 1 and are concatenated column-wise if the MTS dataset has more than one label. Subsequently, K -means clustering is performed on the columns of the concatenated DCPC loadings, each column of which holds the one-to-one correspondence to the original variables. Then, the column vectors with the similar pattern of contributions to each of the DCPCs will be clustered together.

The next step is the *actual* variable selection, which decides the representatives of clusters. Once the clustering is done, one column vector closest to the centroid vector of each cluster is chosen as the representative of that cluster. The other columns within each cluster therefore can be eliminated. Finally, the corresponding original variable to the selected column is identified, which will form the selected subset of variables with the least redundant and possibly the most related information for the given K . For details, please refer to [6].

3 Performance Evaluation

We evaluate the effectiveness of *CLeVer* in terms of classification performance and processing time. We conducted several experiments on three real-world datasets: HumanGait, BCAR, and BCI MPI. After obtaining a subset of variables using *CLeVer*, we performed classification using Support Vector Machine (SVM) with linear kernel and leave-one-out cross validation for BCAR and 10 fold stratified cross validation for the other two datasets. Subsequently, we com-

all the variables, which is 99.4% accuracy. The 27 variables selected by *CLeVer* are from only 16 markers (marked with a filled circle in Figure 2(b)) out of 22, which would mean that the values generated by the remaining 6 markers does not contribute much to the identification of the person. The performances by RFE and FC is much worse than the ones using *CLeVer*. Even when using all the variables, the classification accuracy is around 55%. Figure 2(c) illustrates *CLeVer* consistently outperforms RFE and FC when the number of selected features is more than 4. The 7 variables selected by *CLeVer* produce about 100% classification accuracy, which is even better than using all the 11 variables which is represented as a horizontal solid line. This implies that *CLeVer* never eliminates useful information in its variable selection process. Figure 2(d) represents the performance comparison using the BCI MPI dataset³. It depicts that when the number of selected variables is less than 10, RFE performs better than *CLeVer* and FC technique. When the number of selected variables is greater than 10, however, *CLeVer* performs far better than RFE. Using the 17 variables selected by *CLeVer*, the classification accuracy is 72.85%, which is very close to the performance of *MIC 17*, i.e., the known 17 motor imagery channels, whose accuracy is 73.65%.

Processing Time. The processing time for *CLeVer* includes the time to perform Algorithm 1 and the average time to perform the clustering and obtain the variable subsets while varying K from 2 to the number of all variables for each dataset. The processing time for RFE and FC includes the time to obtain 3 autoregressive fit coefficients and perform the feature subset selection. Overall, *CLeVer* takes up to 2 orders of magnitude less time than RFE, while performing better than RFE up to about a factor of two. The detailed results are provided in [6].

References

1. Liu, H., Yu, L., Dash, M., Motoda, H.: Active feature selection using classes. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. (2003)
2. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** (2003) 1157–1182
3. Tucker, A., Swift, S., Liu, X.: Variable grouping in multivariate time series via correlation. *IEEE Trans. on Systems, Man, and Cybernetics, Part B* **31** (2001)
4. Lal, T.N., Schröder, M., Hinterberger, T., Weston, J., Bogdan, M., Birbaumer, N., Schölkopf, B.: Support vector channel selection in BCI. *IEEE Trans. on Biomedical Engineering* **51** (2004)
5. Krzanowski, W.: Between-groups comparison of principal components. *Journal of the American Statistical Association* **74** (1979)

³ Unlike in [4] where they performed the feature subset selection per subject, the whole items from the 5 subjects were utilized in our experiments. Moreover, the regularization parameter \mathcal{C}_s was estimated via 10 fold cross validation from the training datasets in [4], while we used the default value, which is 1.

6. Yang, K., Yoon, H., Shahabi, C.: Clever: a feature subset selection technique for multivariate time series. Technical report, University of Southern California (2005)
7. Chang, C.C., Lin, C.J.: Libsvm – a library for support vector machines. <http://www.csie.ntu.edu.tw/~cjlin/libsvm/> (2004)
8. Moon, T.K., Stirling, W.C.: Mathematical Methods and Algorithms for Signal Processing. Prentice Hall (2000)