ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

Hierarchical, prototype-based clustering of multiple time series with missing values

Pekka Wartiainen and Tommi Kärkkäinen *

University of Jyvaskyla, Department of Mathematical Information Technology,
P.O. Box 35, 40014 University of Jyvaskyla, Finland

Abstract. A novel technique based on a robust clustering algorithm and multiple internal cluster indices is proposed. The suggested, hierarchical approach allows one to generate a dynamic decision tree like structure to represent the original data in the leaf nodes. It is applied here to divide a given set of multiple time series containing missing values into disjoint subsets. The whole algorithm is first described and then experimented with one particular data set from the UCI repository, already used in [1] for a similar exploration. The obtained results are very promising.

1 Introduction

In data mining and machine learning, division of problems, tasks, and methods into *supervised* and *unsupervised* is still common [2, 3], even if the so-called semisupervised scenarious (partially labelled data) have also emerged (see [4] and articles therein). Typically, if given inputs in data are supplemented with the supervised information in the form of predictive outputs or class labels, one turns the attention into a suitable technique to construct a (linear or nonlinear) model with the training data. An unsupervised technique, especially clustering (e.g., [5] and references therein), can be used as part of preprocessing to reduce the number of samples (replace a set of similar samples belonging to the same cluster with their prototype) or to imput the missing values (supplement the missing values with the available value of the cluster prototype, the so-called cold or hot deck imputation). The purpose of the present paper is to propose and test a novel combination of an unsupervised clustering algorithm in a supervised setting, i.e., with a set of multiple time series with given labelling.

A decision tree classifier ([2, 3]), on the nodal level, works in a supervised fashion when detecting the variables and the rules with their specific internals to divide the data into subtrees. The division is repeated as long as there are observations from more than one class in a node. One observes that the basic technique is both variable-oriented and univariate because only one variable at time determines the split rule. Furthermore, a tree-like structure to introduce a multiclass extension is typical for certain classification methods, such as support vector machines, although challenges appear with large number of classes [6]. Here, by using a clustering algorithm recursively to divide a data into subsets, also creates a dynamic tree-like structure but the rules are oriented along the observations dimension and are, then, multivariate (or multiobservative) by construction.

*The authors gratefully acknowledge the support from the OSER project.

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

Time series clustering can provide useful information in various settings. Liao [7] summarized the work investigating clustering of time series data into three groups depending on whether they work i) directly with the raw data either in the time or frequency domain, ii) indirectly with features extracted from the raw data, or iii) indirectly with models built from the raw data. However, an interesting and extremely important result in the area of time series clustering was presented in [8] stating that time series *subsequence* clustering, as it was approached then far (subsequence data extracted via a sliding window), was completely meaningless! Here, instead, we focus on the so-called whole clustering, i.e., in creation of new groups from the individual time series objects.

The purpose of the present paper is to propose and test a novel combination of a recursive use of a robust clustering algorithm, whose results are stored in a dynamic tree, that is, a tree whose number of childnodes varies. The approach applies unsupervised clustering in a supervised setting, i.e., with a set of multiple time series with given labelling. We introduce the overall algorithm in Section 2. The experimental results are shown in Section 3 and short conclusions provided in Section 4.

2 The Method

One of the oldest but still most popular partitional clustering algorithms is the *k-means*. Its popularity are due to efficiency, simplicity, and scalability when clustering large data sets. However, there are many specifics related to the use of partitional clustering and k-means: i) one needs to detect the number of clusters K, ii) an appropriate initialization is needed because the iterative relocations realize a local search, and iii) it is prone to outliers and nongaussian errors in data due to the use of mean as the statistical estimate for the prototypes.

Concerning the error, a missing value in data can be thought of as an ultimate outlier, because any value (in the variable's value range) could be the one unavailable. Hence, second order statistics which relies on the normally distributed error is not the best choice for a prototype, especially with a sparse data with missing values. Instead, one can and should use the so-called nonparametric, i.e., robust statistical techniques [9]. Two simplest and statistically robust location estimates are median and spatial median. Of these two, the spatial median is truly a multivariate location estimate and can take advantage of the pattern of available data as a whole. This can be clearly seen from the optimality conditions related to these estimates as provided and illustrated in [10]. The spatial median has many attractive statistical properties; especially it's breakdown point is 0.5 meaning that it can handle up to 50% of data contaminations.

In [5], a robust approach utilizing the spatial median to cluster sparse and noisy data was introduced: The *k-spatialmedians* clustering algorithm. It minimizes locally the score (cluster error) function of the form

$$\mathcal{J} = \sum_{k=1}^{K} \sum_{i \in I_k} \|\boldsymbol{P}_i(\boldsymbol{x}_i - \boldsymbol{c}_k)\|_2. \tag{1}$$

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

Algorithm 1: Hierarchical, prototype-based clustering.

Input: Dataset $\mathbf{X}$ with N observations (time series).
Output: A tree where $\mathbf{X}$ in the root node is divided into subsets hierarchically until the leaf nodes.
repeat
1. Create clusters for $k = 2, \ldots, K_{\max}$ and let cluster indices in mixture-of-experts fashion detect the number of clusters K;
2. Consider all K clusters, i.e., child nodes, recursively;

until *Nodal stopping criterion satisfied.*;

Here, I_k refers to the observations that are closest to the kth prototype c_k, which is determined using the projected distance in (1). The projections $\boldsymbol{P}_i, i = 1, \ldots, N$, capture the available values of the observations: $(\boldsymbol{P}_i)_j = 1$, if $(\boldsymbol{x}_i)_j$ exists, and 0 otherwise. Recomputation of the prototypes in (1) is based on the SOR (Sequential OverRelaxation) algorithm [5] with the overrelaxation parameter $\omega = 1.5$.

As explained above, a prototype-based clustering algorithm needs to determine the number of clusters K. The so-called cluster indices (see [5]) measure the quality of the final result of a relocation algorithm. We use here three well-known indices which all take into account the clustering error (1) (the whole intra-cluster error), by combining it with the distance between the prototypes (approximation of inter-cluster error). More precisely, the Ray-Turi index [11] and the Davies-Bouldin, as well as as the Davies-Bouldin*, indices [12] are used. For all the indices, smaller values are better, so they are applied here in a mixture-of-expert fashion with a simple but special gating function: enlarge the number of clusters until any index starts to increase (see Figure 2).

The proposed approach is summarized in Algorithm 1. In the actual realization of Step 1, we aim at maximum stability of the clustering result for each tested k. The k-spatialmedians algorithm, initialized with the well-known k-means++ algorithm with the complete subset of data, to ensure complete prototypes, is first used for clustering. This is then repeated ten times, because of randomness of the initialization and locality of the search. During the ten clusterings, we construct a data structure that captures the co-occurrences of all pairs of observations in the same cluster, denoted with $co(\boldsymbol{x}_i, \boldsymbol{x}_j)$. This information, representing pairwise similarities on the range $[0, 10]$, is then linearly scaled to a pairwise distance function by taking $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = |\, co(\boldsymbol{x}_i, \boldsymbol{x}_j) - \max_{i,j} co(\boldsymbol{x}_i, \boldsymbol{x}_j)|$. Finally, we apply a shortest distance hierarchical clustering algorithm with this distance matrix to create the final clustering result.

3 Experimental results

Next we apply Algorithm 1 to Dodgers data set from UCI repository representing a long time series. Data describes a five minute sampled traffic sensor storing the amount of cars passing a ramp on a freeway in Los Angeles. The learning

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

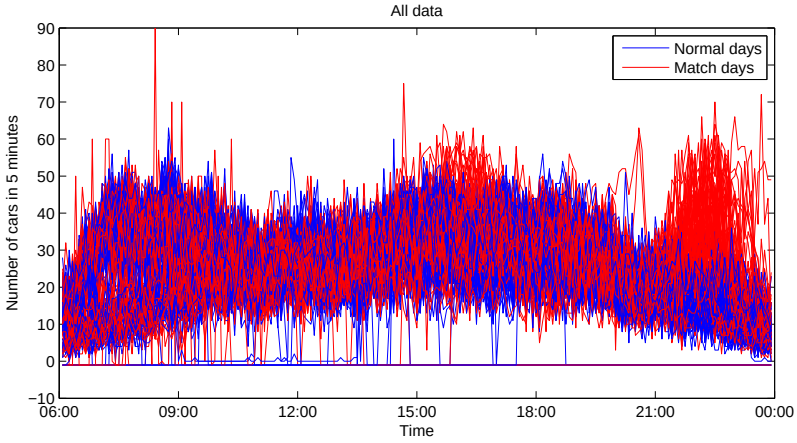


Fig. 1: Dodgers data preprocessed to daily format. Blue data is normal traffic and red data marks the days with matches. Notice missing data at zero level.

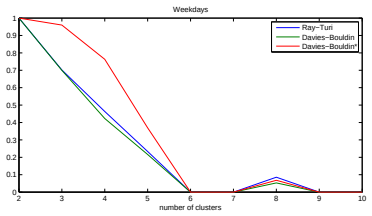
problem is to determine the days of football matches which are provided in another file. In the whole data, there is almost six months of measurements, but occasionally the sensor is off.

At first, preprocessing was performed. The data was cut to shorter vectors where each one described one day of traffic, totaling 175 days. Then we removed the early hours with very little traffic from each day and used only the data between 06:00 – 23:55. We also removed the first and the last day from data, since the sensor was off for the whole day. After these operations, we had separate time series for 173 days of which 81 match days, each day containing 215 observations. These are given in Figure 1.

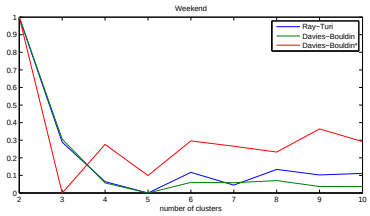
The daily time series were associated with the availability information. The time instances when the sensor was off were marked with -1 in data and, hence, defined as missing. Moreover, sometimes the sensor recordings were clearly unreliable. Thus, we also marked those recordings missing, which were contained in at least 12 consecutive zero observations (one hour traffic) during day time.

After the data preparation, Algorithm 1 was applied with $K_{\max} = 10$. First, we obtained two clusters, which were further divided into several subclusters and subsubclusters. The cluster was marked as final if either all of its data belonged to the same group (match days or normal days) or there were at most only two observations attached to the cluster. The obtained clustering tree is depicted in Figure 3. Prototypes for the second level of the tree are shown in Figure 2. All the created clusters are supplied with metadata of matches and days of week, which are used to characterize the result. We observe that the first clustering divides the days into weekdays and weekends (except the three Mondays in the weekends cluster). After the remaining two levels of clustering, we obtained a clear and overall accurate division (accuracies also given in Figure 3) of data into

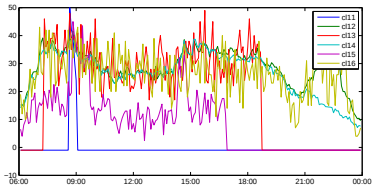
ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.



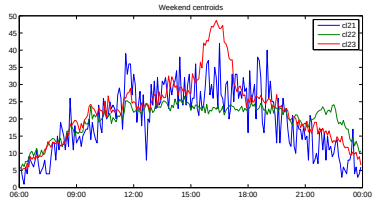
(a) Evaluation indices: Weekdays



(b) Evaluation indices: Weekend



(c) Prototypes: Weekdays



(d) Prototypes: Weekend

Fig. 2: Cluster evaluation indices and obtained prototypes for two main clusters.

natural subsets: evening matches on weekdays with high accuracy but daytime matches mixed with normal traffic patterns; matches during the days of weekend at the evening or during daytime separated from the normal daily traffic patterns without matches. The remaining clusters represented outliers, e.g., due to large number of missing values.

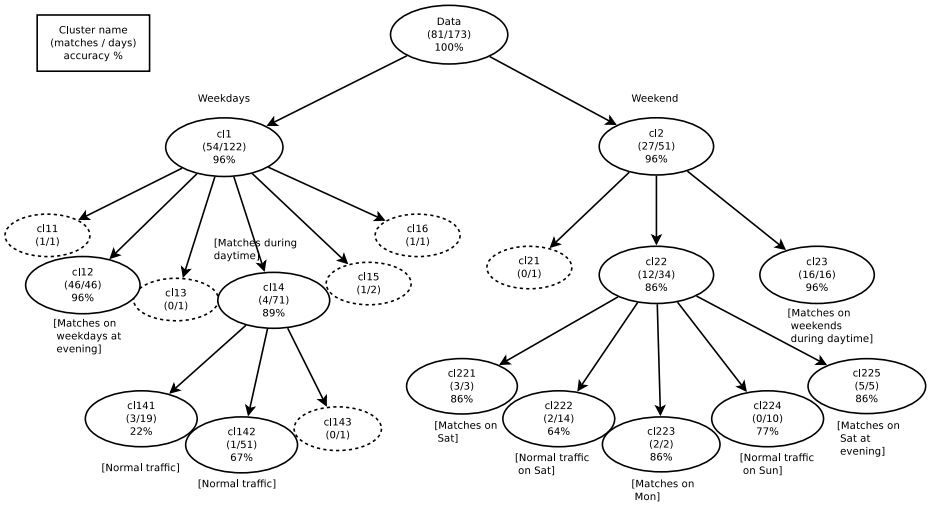


Fig. 3: Hierarchically clustered decision tree.

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

4 Conclusions

We proposed a hybrid of partitive and hierarchical clustering approach, suitable for both unsupervised and supervised use. We did not reach as perfect detection accuracy concerning the match days as reported in [13], but the method that was introduced and used here is simpler and more generally applicable without any modifications to other similar problems. When combined with metadata, one found distinct characterizations for the observations in clusters, such that the nodes revealed clear profiles of certain types of traffic patterns. Moreover, the depth of the obtained dynamic tree was low. However, the weekdays, where the match was played during the daytime, were not separated from the normal traffic pattern (cluster14). The proposed algorithm is the first step towards a dynamic decision tree using robust clustering hierarchically. The actual use of the constructed tree, e.g. in imputation and classification, is to be studied in the later work.

References

[1] T. Kärkkäinen, A. Maslov, and P. Wartiainen. Region of interest detection using MLP. In *Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning - ESANN 2014*, pages 213–218, 2014.

[2] E. Alpaydin. *Introduction to Machine Learning*. The MIT Press, Cambridge, MA, USA, 2nd edition, 2010.

[3] J. Han, M. Kamber, and J. Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011.

[4] B. Frénay and M. Verleysen. Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, 25(5):845–869, 2014.

[5] Sami Äyrämö. *Knowledge Mining Using Robust Clustering*, volume 63 of *Jyväskylä Studies in Computing*. University of Jyväskylä, 2006.

[6] A. Rocha and S. K. Goldenstein. Multiclass from binary: Expanding one-versus-all, one-versus-one and ecoc-based approaches. *IEEE Transactions on Neural Networks and Learning Systems*, 25(2):289–302, 2014.

[7] T. Warren Liao. Clustering of time series data - a survey. *Pattern Recognition*, 38:1857–1874, 2005.

[8] E. Keogh and J. Lin. Clustering of time-series subsequences is meaningless: implications for previous and future research. *Knowledge and Information Systems*, 8:154–177, 2004.

[9] T. P. Hettmansperger and J. W. McKean. *Robust nonparametric statistical methods*. Edward Arnold, London, 1998.

[10] Tommi Kärkkäinen and Erkki Heikkola. Robust formulations for training multilayer perceptrons. *Neural Computation*, 16:837–862, 2004.

[11] S. Ray and R. H. Turi. Determination of number of clusters in k-means clustering and application in colour image segmentation. In *Proceedings of the 4th international conference on advances in pattern recognition and digital techniques*, pages 137–143, 1999.

[12] Minho Kim and RS Ramakrishna. New indices for cluster validity assessment. *Pattern Recognition Letters*, 26(15):2353–2363, 2005.

[13] A. Ihler, J. Hutchins, and P. Smyth. Adaptive event detection with time-varying Poisson processes. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'06)*, pages 27–33, 2006.