Special Issue Article

Journal of CHEMOMETRICS

Received: 21 September 2011, Revised: 3 January 2012, Accepted: 3 January 2012, Published online in Wiley Online Library: 2012

(wileyonlinelibrary.com) DOI: 10.1002/cem.1424

Coclustering—a useful tool for chemometrics

Rasmus Bro[a]*, Evangelos E. Papalexakis[b], Evrim Acar[a] and Nicholas D. Sidiropoulos[c]

Nowadays, chemometric applications in biology can readily deal with tens of thousands of variables, for instance, in omics and environmental analysis. Other areas of chemometrics also deal with distilling relevant information in highly information-rich data sets. Traditional tools such as the principal component analysis or hierarchical clustering are often not optimal for providing succinct and accurate information from high rank data sets. A relatively little known approach that has shown significant potential in other areas of research is *coclustering*, where a data matrix is simultaneously clustered in its rows and columns (objects and variables usually).

Coclustering is the tool of choice when only a subset of variables is related to a specific grouping among objects. Hence, coclustering allows a *select* number of objects to share a particular behavior on a *select* number of variables.

In this paper, we describe the basics of coclustering and use three different example data sets to show the advantages and shortcomings of coclustering. Copyright © 2012 John Wiley & Sons, Ltd.

Keywords: clustering; coclustering; L1 norm; sparsity

1. INTRODUCTION

The chemometric field is dealing with increasingly complex data, for instance, in omics, quantitative structure–activity relationships, and environmental analysis. It is not uncommon to use hyphenated methods for measuring thousands of chemical compounds. This is quite different from traditional chemometric applications, for instance, in spectroscopy where the number of variables (wavelengths) may be high but the actual number of chemicals reflected in the data—the chemical rank—is typically low. Approaches such as principal component analysis (PCA) are very well suited for analyzing fairly low rank data, especially when the gathered data are known to be relevant to the problem being investigated.

Traditional clustering techniques are more useful for exploratory analyses of "classical" data. However, with the increasing number of variables being measured nowadays, there is an interesting opposite trend toward not being interested in modeling the full data. Instead, the focus is often on finding few, so-called, biomarkers. A biomarker can be a specific chemical compound indicative of a pathological condition or indicative of intake of certain food stuff. Thus, even though the actual amount of data and "information" increases, at the same time, the need for simplifying the visualization, interpretation, and understanding increases.

In coclustering, a data matrix is simultaneously clustered in its rows and columns (objects and variables usually). Coclustering is by no means new [11], but it has attracted considerable interest in recent years because of some algorithmic developments and its promising performance in various applications—particularly in bioinformatics [15].

One of the main advantages of coclustering is that it clusters both objects (samples) and variables simultaneously. Suppose we have a data set that shows the food intake of various items for a group of people from Belgium and Korea. In order to find the clusters in this data set, we may use a simple approach where the samples are clustered first, and subsequently, the variables are clustered. It is conceivable that the main clusters could be exactly Asian and European because, overall, the main difference in intake relates to cultural differences. Hence, clustering among samples would split the samples into these two groups. It is also conceivable that there could be another grouping because of, for example, some people preferring fish. However, because fish-related items are only a small part of the variables and fish lovers appear in both populations, such a cluster cannot be realized. On the other hand, coclustering could capture both a country and a fish cluster because it considers which samples are related with which variables at the same time rather than one modality at a time.

Hence, coclustering is the tool of choice when subsets of subjects are related with respect to corresponding subsets of variables. For some coclustering methods it also holds that an individual subject (or variable) can belong to several (or no) clusters. This is so-called *overlapping* coclustering as opposed to non-overlapping coclustering where each variable is assigned to at most one cluster.

In the following, we describe the theory behind coclustering and subsequently exemplify coclustering on a toy data set reflecting different kinds of animals, on a data set of chromatographic measurements of olive oils, as well as on cancer gene expression data.

* *Correspondence to: R. Bro, Department of Food Science, Faculty of Life Sciences, University of Copenhagen, DK-1958 Frederiksberg, Denmark. E-mail: rb@life.ku.dk*

a *R. Bro, E. Acar*
Department of Food Science, Faculty of Life Sciences, University of Copenhagen, DK-1958 Frederiksberg, Denmark

b *E. E. Papalexakis*
School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA

c *N. D. Sidiropoulos*
Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA

2. THEORY

We assume that our data forms a matrix **X** of dimensions $I \times J$.

2.1. Coclustering with sparse matrix regression

Coclustering can be formulated as a constrained outer product decomposition of the data matrix, with sparsity on the latent factors of the bilinear model [17]. Each cocluster is represented by a rank-1 component of the decomposition. Instead of using a plain bilinear model, sparsity on the latent factors is imposed. Intuitively, latent sparsity selects the appropriate rows and columns that belong to each cocluster, rendering all other coefficients that do not belong to a certain cocluster exactly zero. Hence, each bilinear component represents a cocluster.

Mathematically, this coclustering scheme may be stated as the minimization of the following loss function:

$$\left\|\mathbf{X} - \mathbf{A}\mathbf{B}^T\right\|_F^2 + \lambda \sum_{i,k} |\mathbf{A}_{ik}| + \lambda \sum_{j,k} |\mathbf{B}_{jk}|$$

where **A** and **B** are matrices of size $I \times K$ and $J \times K$, respectively; K corresponds to the number of extracted coclusters. The sum of absolute values is used as a sparsity-inducing surrogate for the number of nonzero elements, for example, see Ref. [19], and λ is a sparsity-controlling parameter.

The loss function can be interpreted as a *constrained* version of a bilinear model such as PCA. Rotations such as varimax [12] also aim at simplicity and sparsity, but they do so in a lossless manner, where the actual bilinear approximation of the data is left unchanged. It is merely rotated toward a simpler view that will not usually lead to real sparsity.

Doubly sparse matrix factorization as shown has been proposed earlier [13,20]. Witten *et al.* [20] proposed adding sparsity-inducing hard one-norm constraints on both left and right latent vectors, as a variation of sparse singular value decomposition and sparse canonical correlation analysis. Although their model was not developed with coclustering in mind, it is similar to sparse matrix regression (SMR), which uses soft one-norm penalties instead of hard constraints (and possibly non-negativity when appropriate). Algorithmically, Witten *et al.* [20] use a deflation algorithm that extracts one rank-1 component at a time, instead of alternating optimization across rank-1 components as in SMR.

Lee *et al.* [13] proposed a similar approach specifically for coclustering. However, their algorithm is not guaranteed to converge because the penalties are not kept fixed during iterations. As a result, the algorithm in Lee *et al.* [13] does not monotonically reduce a tangible cost function, and instabilities are not uncommon.

In Papalexakis *et al.* [18], a coordinate descent algorithm is proposed in order to solve the given optimization problem. More specifically, one may solve this problem in an alternating fashion, where each subproblem is basically a least absolute shrinkage and selection operator problem [16,19]. We have to note that a global minimum for the bilinear problem may not be attained; the existing algorithms guarantee a local minimum or saddle point solution only.

The SMR coclustering algorithm [18] may be characterized as a soft or fuzzy coclustering algorithm, in the sense that cocluster membership is not merely a zero or one, but can be any value in between. Some rows and columns may not be assigned to any cocluster, and overlapping coclusters are allowed and can be extracted.

It follows that when sparsity is imposed to such an extent that rows and columns are completely left out, the concept of assessing residual sums of squares or fit values is not meaningful or at least not meaningful in the same sense as for ordinary least squares fitting. Therefore, other means for evaluating the usefulness of a model are needed. Such are described in the following section on metaparameters. Also, interpreting why certain samples or variables are left "orphan" may be useful for understanding the coclustering. This is usually an application-specific problem.

One may add non-negativity constraints to the given loss function formulation, which can be readily applied within the existing coordinate descent algorithm with minor modifications.

Although our focus here will be on SMR coclustering because of its appropriateness for chemometric applications, there are several types of coclustering models and algorithms that are popular in other areas and worth mentioning. Banerjee *et al.* [1,3,8] have introduced a class of coclustering algorithms that use *Bregman divergences*, unified in an abstract framework. Bregman coclustering is a hard coclustering technique, in the sense that it seeks to locate a non-overlapping "checkerboard" structure in the data. This type of coclustering is typically not of interest in chemometrics, where one often deals with data that contain large numbers of potentially irrelevant variables. Dhillon [7] has formulated coclustering as a bipartite graph partitioning problem originally in the context of coclustering of documents and words from a document corpus. This algorithm can also be classified as hard coclustering. In addition, this algorithm works for non-negative data only. Initial testing of various algorithms has shown that appearance of local minima is a common problem. In fact, most hard coclustering algorithms seem to have much more pronounced problems with local minima than soft coclustering ones. Furthermore, the possible local minima in soft coclustering are often distinct (e.g., rank deficient) and hence easier to spot. Other approaches that are more distantly related are methods presented by Damian *et al.* [4] and Friedman and Meulman [9], which do not account for sparsity, and the hard coclustering method of Hageman *et al.* [10], which uses a genetic algorithm that is sensitive to local minima.

2.2. Metaparameters

For SMR, there are certain meta-parameters, that is, the penalty λ and number of coclusters that need to be chosen. The number of coclusters must be selected in most coclustering methods, but for SMR, which is not based on hard clustering, it is found that in many cases, the clusters are exactly or approximately nested as we increase the number of clusters. Hence, for example, for a solution with five coclusters, it is often found that the first three coclusters is approximately equal the solution found using only three coclusters. The reason for this approximate nestedness is currently being investigated further. In any case, it greatly simplifies the use of the method. For hard coclustering methods, a similar behavior is naturally not observed.

In practice, the metaparameters are mostly determined in the following way: The penalty for a given number of components is chosen so that it is active. Choosing λ that is too small would give an inactive penalty, and choosing λ that is too big would lead to some components/coclusters with all zero values. A simple line search can be implemented to find a value of λ that is active without leading to all zeros. It is generally seen that the specific setting of λ is not critical, but of course, any automatically determined value of λ can be further refined. This has not been

Table 1. Animal data set used to illustrate coclustering

	Has eyes	Number of legs/ arms	Carnivore	Feather	Wings	Domesticized	Eaten by Caucasians	>100 kg	>2 m	Breathe under water	Extinct	Dangerous	Life expectancy	Random	Has a beak	Walk on two legs	Speed (MPH)
Giraffe	1	4	0	0	0	0	0	1	1	0	0	0	30	1	0	0	32
Cow	1	4	0	0	0	1	1	1	1	0	0	0	15	3	0	0	30
Lion	1	4	1	0	0	0	0	1	0	0	0	1	15	6	0	0	50
Gorilla	1	4	0	0	0	0	0	1	0	0	0	1	30	2	0	1	25
Fly	1	6	0	0	1	0	0	0	0	0	0	0	0,1	7	0	0	5
Spider	1	8	1	0	0	0	0	0	0	0	0	0	1	8	0	0	1
Shark	1	0	1	0	0	0	0	1	0	1	0	1	50	4	0	0	30
House	0	0	0	0	0	0	0	1	1	0	0	0	100	9	0	0	0
Horse	1	4	0	0	0	1	1	1	1	0	0	0	15	2	0	0	40
Elephant	1	4	0	0	0	0	0	1	1	0	0	0	35	6	0	0	25
Mammoth	1	4	0	0	0	0	0	1	1	0	1	0	35	5	0	0	25
Sabre Tiger	1	4	1	0	0	0	0	1	0	0	1	1	15	7	0	0	40
Pig	1	4	0	0	0	1	1	1	0	0	0	0	25	8	0	0	11
Cod	1	0	1	0	0	0	1	0	0	1	0	0	40	9	0	0	2
Eel	1	0	1	0	0	0	1	0	0	1	0	0	55	1	0	0	20
Jellyfish	1	0	0	0	0	0	0	0	0	1	0	0	0,7	3	0	0	1
Dolphin	1	0	1	0	0	0	0	1	1	1	0	0	30	5	0	0	35
Nemo	1	0	0	0	0	0	0	0	0	1	0	0	1	6	0	0	4
Shrimp	1	0	0	0	0	0	1	0	0	1	0	0	1	2	0	0	0,5
Dog	1	4	1	0	0	1	0	0	0	0	0	0	13	8	0	0	35
Cat	1	4	1	0	0	1	0	0	0	0	0	0	25	9	0	0	30
Fox	1	4	1	0	0	0	0	0	0	0	0	0	14	4	0	0	42
Wolf	1	4	1	0	0	0	0	0	0	0	0	1	18	3	0	0	25
Rabbit	1	4	0	0	0	1	1	0	0	0	0	0	9	8	0	0	35
Chicken	1	2	0	1	1	1	1	0	0	0	0	0	15	1	1	1	9
Eagle	1	2	1	1	1	0	0	0	0	0	0	0	55	3	1	1	60
Seagull	1	2	1	1	1	0	0	0	0	0	0	0	10	6	1	1	25
Blackbird	1	2	1	1	1	0	0	0	0	0	0	0	18	0	1	1	25
Bat	1	2	1	0	1	0	0	0	0	0	0	0	24	4	0	0	8
T. Rex.	1	4	1	0	0	0	0	1	1	0	1	1	40	9	0	1	25
Neanderthal	1	4	1	0	0	0	0	0	0	0	1	0	50	8	0	1	18
Triceratops	1	4	1	0	0	0	0	1	1	0	1	1	30	5	0	0	10
Man	1	4	1	0	0	0	0	0	0	0	0	0	80	2	0	1	28
Penguin	1	2	1	1	1	0	0	0	0	0	0	0	15	4	1	1	25

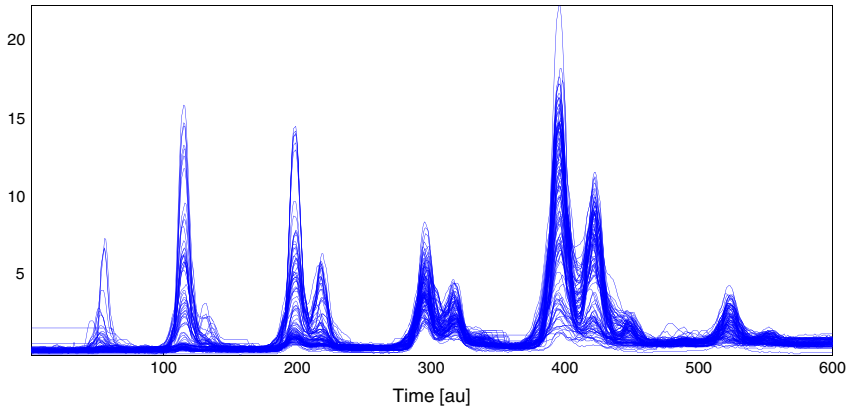

Figure 1. Preprocessed chromatographic data.

pursued here. In order to determine the number of coclusters, a fairly ad hoc approach has been used. Because coclustering is used for exploratory analysis and the solution is nested, we simply extract sufficiently many components to explain the main clusters. More rigorous approaches such as cross-validation could be implemented, but we do not see the predictive ability of coclustering as a very meaningful criterion to optimize. Rather, we find that interpretability of clusters is what is often sought and what we focus on here.

3. MATERIALS AND METHODS

A toy data set is constructed for illustrating the behavior of coclustering in general. This data set shows attributes of different animals, and the data were not made particularly meticulously. Several variables are not well defined, but this is of moderate consequence in this context. Also, the data were made from the authors' point of view, for example, in terms of which animals are domesticized. In Table I, the data set is tabulated. Note that the data also includes an outlying sample (house) and an outlying variable (random).

As another example, data from Refs [5,6] are analyzed. One hundred twenty-six oil samples are analyzed by HPLC coupled to a charged aerosol detector. Of the oil samples, 68 were various types, and grades of olive oils and the remaining were either non-olive vegetable oils or non-olive vegetable oils mixed with olive oil. The HPLC method is aimed at providing a triacylglyceride profile of the oils. The triacylglycerides are known to have a distinct pattern for olive oils. The data were baseline corrected and aligned as described in the original work, and the resulting data after removal of a few outliers is shown in Figure 1.

As a final data set, we looked at a typical gene expression data set. A total of 56 samples were selected from a cohort of lung cancer patients assayed by using the Affymetrix 95av2 GeneChip brand oligonucleotide array. The 56 patients represent four distinct histological types: normal lung, pulmonary carcinoid tumors, colon metastases, and small cell carcinoma. The data have been described in several publications [2,14] and also using coclustering [13]. The original data set contains 12 625 genes. Unlike most publications, no pre-selection to reduce the number of genes is performed here. Rather, coclustering is applied directly on the data. The data set holds information on 56 patients of which 20 are pulmonary carcinoid samples, 13 colon cancer metastasis samples, 17 normal lung samples, and 6 small cell carcinoma samples. The data set is fairly easy to cluster into these four groups.

The data and the algorithm can be found at www.models.life.ku.dk (January 2012).

4. RESULTS

4.1. Looking at the animal data set

It is interesting to investigate the outcome of a simple PCA model on the auto-scaled animal data. In Figure 2, a score plot of the first two components of a PCA model is shown. Component 1 seems to reflect birds, which is verified from the loading vector that has high values for the variables: feather, wings, has a beak, and walk on two legs. Component 2, though, is difficult to interpret and seems to reflect a mix of different properties. This is also apparent from the loading plot.

Looking at components 3 and 4 (Figure 3), similar complications arise in interpreting the meaning of different components. All but the first component reflect several phenomena in a contrast fashion, and often, it is difficult to extract and distinguish the important variation.

Turning to SMR, a model is fitted using six coclusters. Similar results are obtained with different numbers of coclusters, but we chose six here to exemplify the results. The data are scaled, not centered, and non-negativity is imposed. It is possible to plot the resulting components/clusters as ordinary PCA components in scatter or line plots. However, the semi-discrete nature of the clusters sometimes makes such visualizations less efficient. Instead, we have developed a plot where each cluster is shown by labels of all samples and variables larger than a threshold. This threshold was set to 20% of maximum but was inactive here because all elements smaller than 20% of maximum were exactly zero. Furthermore, the size of the label indicates the size of the element. This provides an intuitive visualization as shown in Figure 4 for the six-cocluster SMR model.

It is striking how easy it is to assess the meaning of this model compared with the PCA model. Looking at the coclusters one at a time, it is observed that cocluster 1 is a bird cocluster. Cocluster 2 is given by one variable (extinct) and is evident. Cocluster 3 comprises big animals. Note how several samples in coclusters 2 and 3 coincide. Animals in cocluster 4 are "grown" and eaten by people, and cocluster 5 captures animals living in water. Finally, cocluster 6 is too dense to allow an easy interpretation. It is apparently a cocluster relating to the overall variation and is in this sense taking care of the offsets induced by the lack of centering.

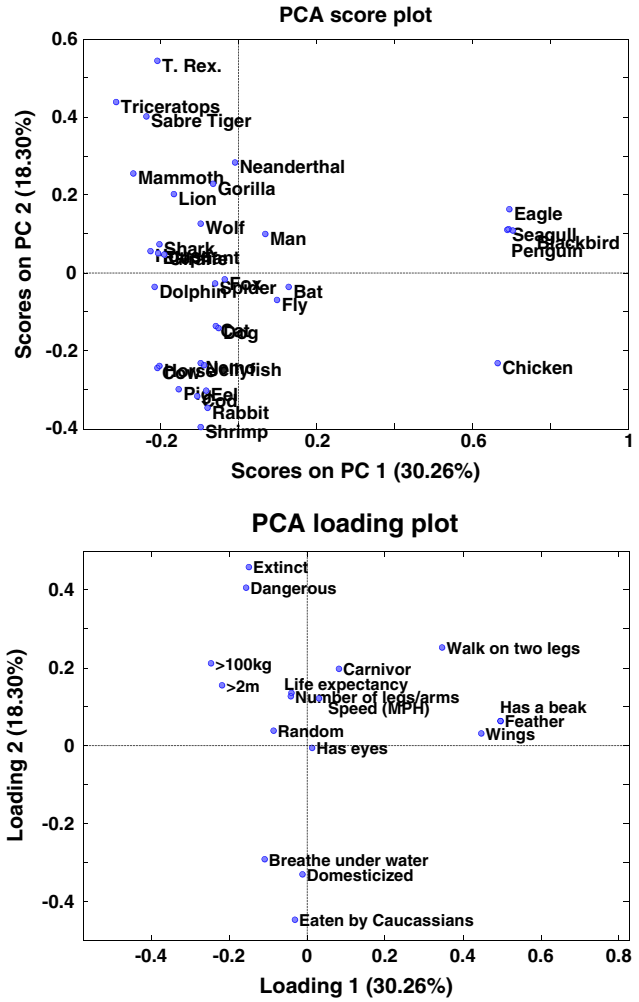

Figure 2. Top: score plot of PCA model. Bottom: corresponding loading plot.

There is a dramatic difference in how easy it is to visualize the results of PCA and SMR, but the data set is simple in the sense that there are no significant amounts of irrelevant variation. In order to see how SMR can deal with irrelevant variation, 30 random variables (uniformly distributed) were added to the original 17 variables. The data were scaled such that each variable had unit variance and SMR was performed.

In Figure 5, it is seen that the method very nicely distinguishes between the animal-related information and the random variables. All coclusters but cocluster 7 are easy to interpret. Cocluster 7 is not sparse at all—it comprises almost all variables and all samples. Also, note that the remaining coclusters are not identical to the coclusters found before, but they are indeed fairly similar.

4.2. Olive oils

For the olive oil data set, a nice separation is achieved with three coclusters. Adding more does not seem to change the coclusters obtained in the three cocluster model, and the added coclusters are not immediately meaningful. In Figure 6, it is seen that cocluster 1 reflects olive oils, whereas cocluster 2 reflects non-olive oils. The mixed samples containing some olive oils are placed in between. The third cocluster seems to reflect only a fraction of the olive oils. This is likely related to the olive oils being a very diverse class of samples spanning from pomace to extra virgin oil. The corresponding elution profiles of each cluster are meaningful. The first (olive oil) cocluster has peaks around 300 and 400 (arbitrary units), and those peaks represent the main olive oil triacylglycerides (triolein, 1,2-olein-3-palmitin, and 1,2-olein-3-linolein). Likewise, the non-olive oil cocluster represents trilinolein, 1,2-linolein-3-olein, and 1,2-linolein-3-palmitin, which are frequent in non-olive oils. It is satisfying to see that the olive oil samples are clustered together, as desired, even though SMR is an unsupervised approach that does not use any prior or side information.

The results obtained with coclustering are not too different from what would be obtained with PCA. In fact, it is somewhat disturbing that there is a distinct lack of sparsity. Although the model makes sense from a chemical point of view, little sparsity is seen, for example, in loading 1 and 2 (on the other hand, loading 3 is sparse, and so are scores 2 and 3 to a certain extent). As described in the theory section, the magnitude of the L1 penalty is automatically chosen, but it turns out that it is not possible to obtain more

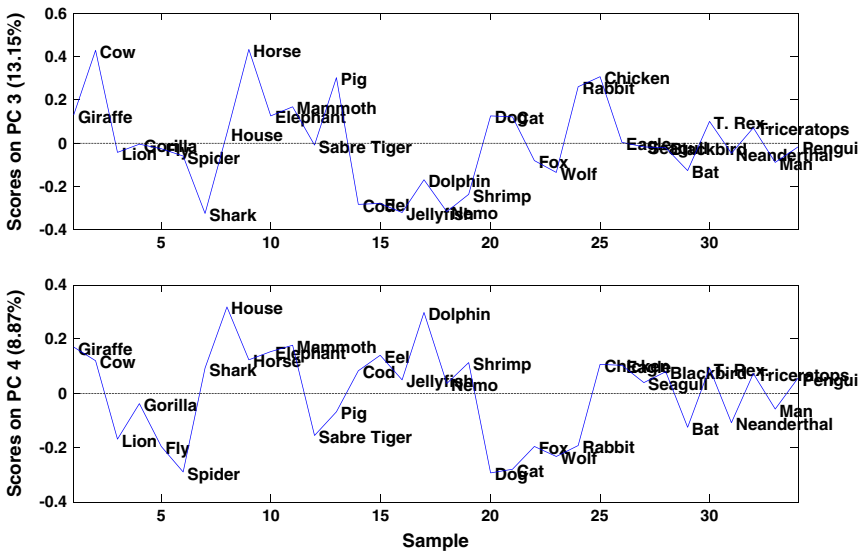

Figure 3. Scores 3 and 4 from a PCA model.

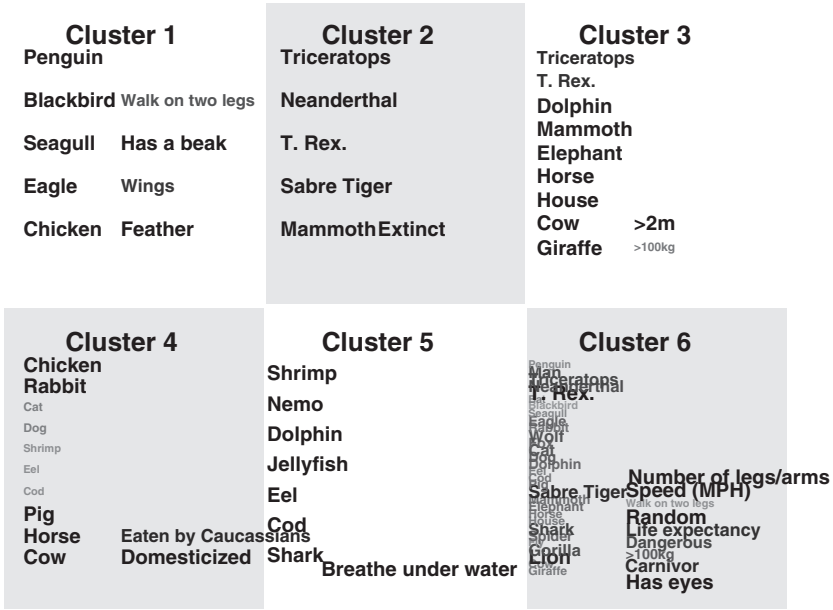

Figure 4. Sparse matrix regression coclusters of animal data. Font size indicates "belongingness" to the cluster.

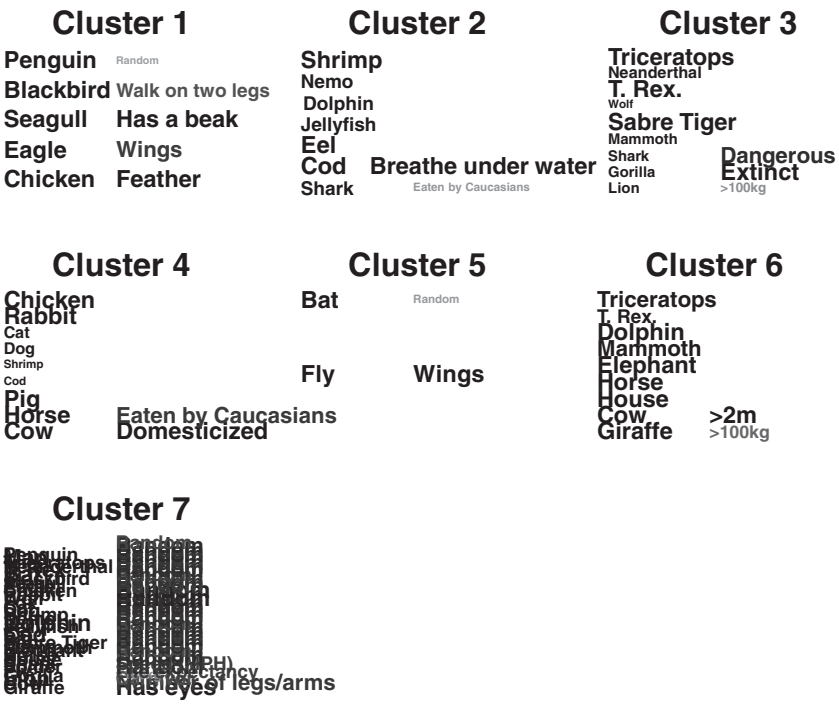

Figure 5. Sparse matrix regression coclusters with 30 random variables added to the data.

sparsity than shown here. Manually increasing λ leads to a model where one component/cocluster turns all zero and hence rank deficient. This points to a problem with the current coclustering approach. Because λ is the same for both the row and the column mode, problems or lack of sparsity may occur when the modes are quite different in dimension. The lack of sparsity is likely caused by the strong collinearity as well as by the lack of intrinsic sparsity in this type of data. It is questionable if coclustering as defined here is a suitable model for spectral-like data such as these. A more suitable approach could be an *elastic net*-type coclustering [21], which would allow the natural collinearities to be represented in the clusters. This seems like an interesting research direction.

Note that for this particular data set, it would be possible to integrate the chromatographic peaks and thereby obtain discrete data that would be more suitable for coclustering.

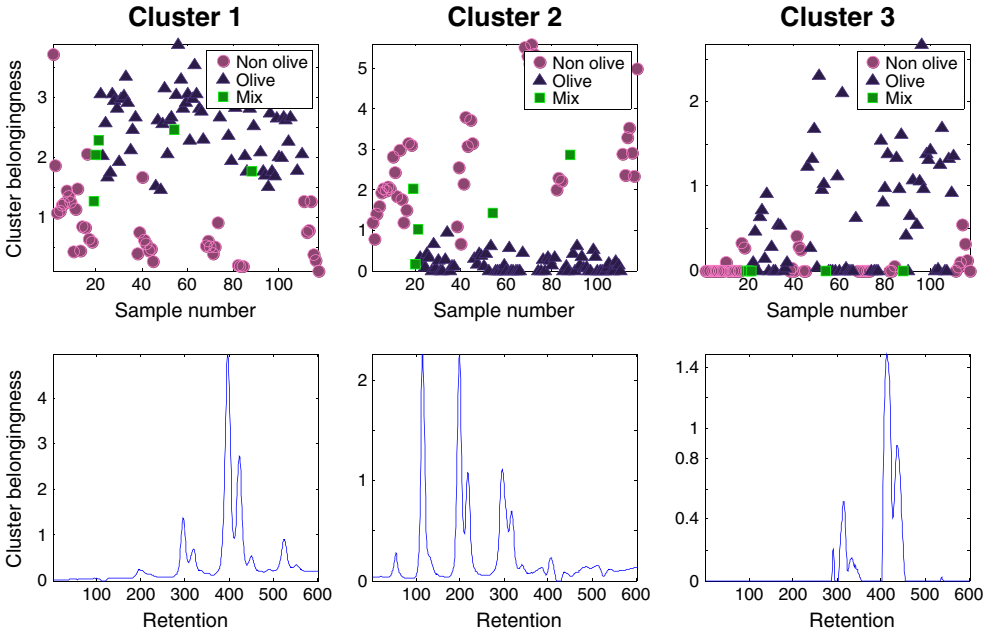

Figure 6. Three sparse matrix regression clusters are shown. Top plots show sample clusters and bottom plots show elution time clusters.

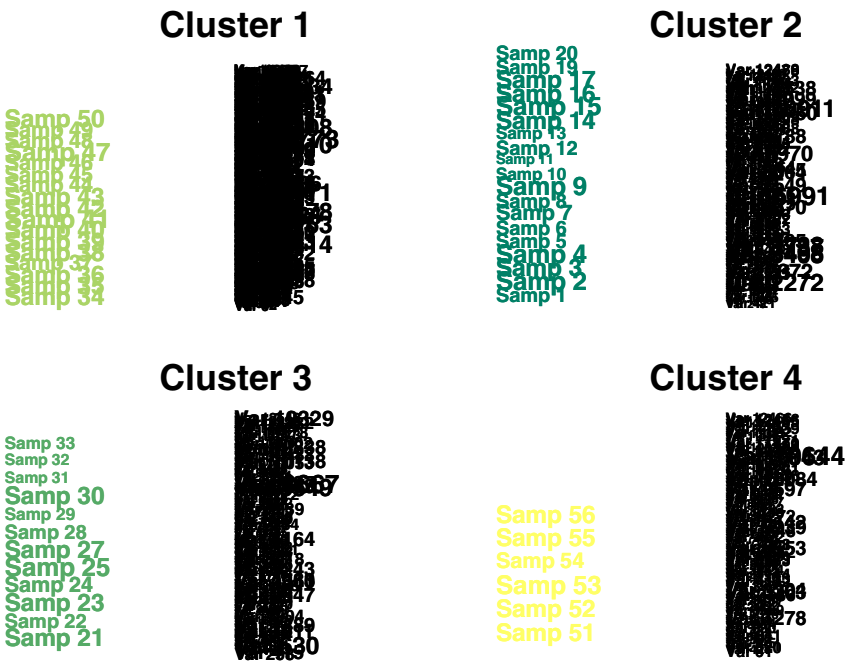

Figure 7. Sparse matrix regression coclusters of cancer data color-coded according to cancer class.

The intention though, with the given example, is to illustrate the behavior of coclustering on continuous data.

4.3. Cancer

When analyzing the gene expression data, the four different cancer types come out immediately when we fit a four-cocluster model as shown in Figure 7, where the four cancer classes are color coded. It is apparent that the four cancer classes are perfectly clustered, but it is also apparent that the gene mode shows little sparsity in comparison with patients. Hence, coclustering does not provide the sparsity desired in order to be able to talk meaningfully of specific biomarkers.

Performing a PCA on the same data (auto-scaled) provides a very clear grouping into the four cancer types (not shown). The separation is not perfect as in Figure 7, but the tendency is very clear. Lee *et al.* [13] also performed coclustering with an algorithm similar to the SMR algorithm. The coclustering in the sample space that they obtained resembles the one obtained using PCA more than the distinct coclustering obtained in Figure 7. This, however, can be explained by the fact that penalties are chosen differently by Lee *et al.* using a Bayesian

information criterion. Regardless, as also observed with the SMR algorithm, the algorithm of Lee *et al*. produces solutions that are not as sparse as expected in the gene mode.

5. CONCLUSION

The basic principles behind coclustering have been explained, and a new model and algorithm have been favorably compared with common methods such as PCA. It is shown that coclustering can provide meaningful and easily interpretable results on both fairly simple and complex data compared with more traditional approaches. Limitations were encountered when the number of irrelevant samples grew too high and when spectral-like data are analyzed. More elaborate algorithms need to be developed for handling such situations.

Acknowledgements

N. Sidiropoulos was supported in part by ARO grant W911NF-11-1-0500.

REFERENCES

1. Banerjee A, Merugu S, Dhillon IS, Ghosh J. Clustering with Bregman divergences. *J. Mach. Learn Res.* 2005; **6**: 1705–1749.
2. Bhattacharjee A, Richards WG, Staunton J, Li C, Monti S, Vasa P, Ladd C, Beheshti J, Bueno R, Gillette M, Loda M, Weber G, Mark EJ, Lander ER, Wong W, Johnson BE, Golub TR, Sugarbaker DJ, Meyerson M. Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. *Proc. Natl. Acad. Sci. U.S.A.* 2001; **98**: 13790–13795.
3. Cho H, Dhillon IS, Guan Y, Sra S. Minimum sum-squared residue co-clustering of gene expression data. *Proceedings of the Fourth SIAM International Conference on Data Mining* 2004; 114–125.
4. Damian D, Oresic M, Verheij E, Meulman J, Friedman J, Adourian A, Morel N, Smilde A, van der Greef J. Applications of a new subspace clustering algorithm (COSA) in medical systems biology. *Metabolomics* 2007; **3**: 69–77.
5. de la Mata-Espinosa P, Bosque-Sendra JM, Bro R, Cuadros-Rodriguez L. Discriminating olive and non-olive oils using HPLC-CAD and chemometrics. *Anal. Bioanal. Chem.* 2011a\; **399**: 2083–2092.
6. de la Mata-Espinosa P, Bosque-Sendra JM, Bro R, Cuadros-Rodriguez L. Olive oil quantification of edible vegetable oil blends using triacylglycerols chromatographic fingerprints and chemometric tools. *Talanta* 2011b; **85**: 177–182.
7. Dhillon IS. Co-clustering documents and words using bipartite spectral graph partitioning. *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining* 2001; 269–274.
8. Dhillon IS, Mallela S, Modha DS. Information-theoretic co-clustering. *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 2003; 89–98.
9. Friedman JH, Meulman JJ. Clustering objects on subsets of attributes. *J. Roy. Stat. Soc. B Stat. Meth.* 2004; **66**: 815–849.
10. Hageman JA, van den Berg RA, Westerhuis JA, van der Werf MJ, Smilde AK. Genetic algorithm based two-mode clustering of metabolomics data. *Metabolomics* 2008; **4**: 141–149.
11. Hartigan JA. Direct clustering of a data matrix. *J. Am. Stat. Assoc.* 1972; **67**: 123–129.
12. Kaiser HF. The varimax criterion for analytic rotation in factor analysis. *Psychometrika* 1958; **23**: 187–200.
13. Lee M, Shen H, Huang JZ, Marron JS. Biclustering via sparse singular value decomposition. *Biometrics* 2010; **66**: 1087–1095.
14. Liu Y, Hayes DN, Nobel A, Marron JS. Statistical significance of clustering for high-dimension, low-sample size data. *J. Am. Stat. Assoc.* 2008; **103**: 1281–1293.
15. Madeira SC, Oliveira AL. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Trans. Comput. Biol. Bioinform.* 2004; **1**: 24–45.
16. Osborne MR, Presnell B, Turlach BA. On the LASSO and its dual. *J. Comput. Graph. Stat.* 2000; **9**: 319–337.
17. Papalexakis EE, Sidiropoulos ND. Co-clustering as multilinear decomposition with sparse latent factors. *2011 IEEE International Conference on Acoustics, Speech and Signal Processing*, Prague, Czech Republic, 2011.
18. Papalexakis EE, Sidiropoulos ND, Garofalakis MN. Reviewer profiling using sparse matrix regression. *2010 IEEE International Conference on Data Mining Workshops*, 2010; 1214–1219.
19. Tibshirani R. Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc. B* 1996; **58**: 267–288.
20. Witten DM, Tibshirani R, Hastie T. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics* 2009; **10**: 515–534.
21. Zou H, Hastie T. Regularization and variable selection via the elastic net. *J. Roy. Stat. Soc. B* 2005; **67**: 301–320.

