Application of a Genetic Algorithm to Variable Selection in Fuzzy Clustering

Christian Röver and Gero Szepannek

University of Dortmund*
Department of Statistics
44221 Dortmund, Germany
roever@statistik.uni-dortmund.de
szepannek@statistik.uni-dortmund.de

Abstract. In order to group the observations of a data set into a given number of clusters, an 'optimal' subset out of a greater number of explanatory variables is to be selected. The problem is approached by maximizing a quality measure under certain restrictions that are supposed to keep the subset most representative of the whole data. The restrictions may either be set manually, or generated from the data. A genetic optimization algorithm is developed to solve this problem.
The procedure is then applied to a data set describing features of sub-districts of the city of Dortmund, Germany, to detect different social milieus and investigate the variables making up the differences between these.

1 The problem

Before the observations are clustered, the data need to be reduced. A reduction is necessary to

1. avoid overfitting,
2. exclude noise and redundant variables and
3. keep the data perceptible and interpretable.

To achieve these goals, we would like to use a subset of the original variables rather than, for example, linear combinations (like principal components) that are harder to interpret.
To determine an 'optimal' subset of variables, some measure of cluster quality needs to be optimized; this measure should return comparable values regardless of the number or scale of variables in the subset. Also, some restrictions should be met to make sure that, for example, the subset has more than one element and, in some sense, most data features are reflected in the subset.

* This work has been supported by the Deutsche Forschungsgemeinschaft, Sonderforschungsbereich 475.

2 Tackling the problem

We focused on *fuzzy clustering methods*, that is, methods that do not assign fixed clusters to each observation, but that return posterior probabilities of cluster membership instead. These methods are often a more appropriate approach to clustering problems.
Validity measures are then computed from the membership matrix that is yielded by clustering with a specific variable set, and thus independently from the underlying variables themselves. In particular, they do not depend directly on the number or scales of variables. Assessment of clusterings with different variable sets can then be based on such measures.

Basing variable selection on the membership matrix alone still may lead to in some sense 'optimal', but still useless solutions. The final variable set may consist of a single, or some highly correlated variables, for example.
Instead, we try to keep the selected subset of variables as representative as possible of the complete data set. In order to achieve this, we are introducing subgroups of variables that have to be represented in the selected subset. These subgroups are either arranged 'by hand' (groups of variables with similar meaning or representing a certain aspect of the data set) or automatically (groups of correlated variables).

The selection itself is then performed by a genetic algorithm that can pretty easily be adapted to handle a parameter space of this kind (that is, a restricted space with varying dimension).

All computations will be performed using R, a free software for data analysis (Ihaka and Gentleman, 1996).

3 Methods

3.1 Fuzzy clustering

Usually, a clustering procedure returns specific assignments of clusters to all observations. Fuzzy clustering methods instead are those methods, that for each observation provide indices measuring the potential affiliation to *all* of the clusters.
The result of a fuzzy clustering then is a $(N \times k)$ membership matrix U, whith u_{ij} denoting the probability that observation i belongs to cluster j; or in other words: each row of U corresponds to one observation (i) and is the distribution of membership over clusters $1, \ldots, k$. An example with 3 clusters:

$$U = \begin{pmatrix} 0.95 & 0.02 & 0.03 \\ 0.50 & 0.30 & 0.20 \\ \vdots & \vdots & \vdots \end{pmatrix}$$

Both observations would be assigned to cluster 1, while the second assignment is not as clear as the first one.
We considered two different clustering methods, the `cmeans`-procedure from the `e1071` package, which is a fuzzy version of the known k-means clustering, and the `EMclust`-procedure from the package `mclust`. '`EMclust`' fits a Gaussian mixture model with k components to the data; in this case the components have the same covariance structure and differ by their means and a-priori-probabilities. The data is then clustered by assigning each observation to one of the k mixture components. We eventually decided in favour of the second method, mostly for interpretability reasons: while k-means-clustering by its nature carves the data into sphere-shaped clusters, model-based clustering is able to handle clusters with covariance structures even different to spheric shapes and does not require (and depend on) normalization (Fraley and Raftery, 2002).

3.2 Measuring the clustering quality

Let U be a $(N \times k)$ membership matrix, as above. All the u_{ij} should be close to one or zero, so the clustering yields distinct assignments. A measure of this feature is the *classification entropy*:

$$\mathrm{CE}(U) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{k}(u_{ij} \cdot \log_2 u_{ij})$$

$CE(U)$ is zero, if all elements of U are either 1 or 0 (a most crisp partitioning), and takes its maximum if all of them are $= \frac{1}{k}$ (the fuzziest partitioning) (Hall,1999).
The classification entropy allows comparison of clusterings based on different variable sets with different numbers of variables, but is sensitive to the number of clusters (k), so this quantity needs to be fixed beforehand.

While a most fuzzy clustering pretty obviously is a bad clustering, a crisp clustering does not necessarily have to be a good clustering. A variable subset leading to low entropy may in some sense not represent the data appropriately. In order to force some structure into the subset selection process, the concept of subgroups is introduced in the following section. This allows for the injection of expert knowledge or of further information on the data (correlations) into the procecedure.

3.3 Defining subgroups of variables

Subgroups can be defined manually, or they are constructed systematically as groups of *correlated variables*. These subgroups are generated by agglomerative hierarchical clustering (Kaufman and Rousseeuw, 1990); the *variables*

are clustered, and to do so, the 'distance' between two variables X_1 and X_2 is defined as:

$$d(X_1, X_2) = 1 - |\mathrm{Cor}(X_1, X_2)|$$

Thus, variables with a high (absolute) correlation are 'close' to each other, while uncorrelated variables are 'farther' from each other. With this definition, the correlation matrix can directly be transformed into a distance matrix, which is the only basis needed for the clustering. Using either *complete* or *single linkage* yields groups of variables with different interpretations:

complete linkage: the (absolute) correlation of variables from the same group is bounded below.
single linkage: the (absolute) correlation of variables from different groups is bounded above.

In both cases, the groups may be interpreted as variable sets with some common source of variability; and by picking variables from different groups, the intention is to cover these different sources.

3.4 Genetic optimization algorithms

Optimization problems, in general, are problems of finding the minimum of some function $f : \mathcal{M} \to \mathbb{R}$ that projects from some space $\mathcal{M}$ to the real line. Genetic algorithms are stochastic optimization algorithms to solve these kinds of problems by making use of evolutionary principles as known from biology, namely *mutation*, *recombination* and *selection ('survival of the fittest')*.
In nature, the fitness of individuals depends on their genes. Individuals with a greater fitness have a greater chance of survival and also the wider range of mating partners to choose between. 'New' individuals are generated by

mutation: single genes of an individual are changed, or
recombination: two genomes are combined to a new one.

Again, new individuals have to compete with the current population for partners and survival. The competitiveness of each individual is determined by its fitness.
Analogously, in genetic algorithms the goal function to be optimized corresponds to the fitness, and the individuals are parameter sets for the function. To start the algorithm, a starting population is generated. Then, generation by generation, the population is multiplied by mutating and breeding individuals and only the 'fittest' ones (as judged by the goal function) survive until the next generation. At some point, the procedure stops and the (so far) best parameter set is returned.
An advantage of genetic algorithms is, that the parameter space ($\mathcal{M}$) can literally be *any* space, as long as the mutation- and recombination procedures can be defined reasonably. Restrictions are implemented pretty easily as well (Goldberg, 1989).

3.5 Implementation

Parameters to be defined beforehand are: the subgroups of variables, the minimum numbers of representatives for each group that have to be in a variable subset (≥ 0), the (total) maximum number of variables in a subset, the 'population size' and the number of generations.
A 'genome' (an 'individual') is a vector of variable indices; its minimum length depends on the sum of minimum numbers for each variable group, and the maximum length is defined explicitly. The population is made up by a set of these individuals. The fitness of each individual is determined by clustering the data using the corresponding subset of variables and then computing the classification entropy as the measure of clustering quality that is achieved with this subset (the smaller the entropy, the greater the fitness).
In the beginning, a random starting population (of the given size) is created, and the fitness of each individual is determined. In each generation, individuals are mutated (a new individual is generated by changing, adding or deleting single indices from a given individual), and pairs of individuals are crossed (a new set of indices is selected from the union of parental indices). The chance of being mutated or crossed is proportional to the individuals' fitness. In each step (creation of starting population, mutation, recombination) it is made sure that the resulting individuals comply with the restrictions (given by the pre-defined subsets and referring minimum numbers). After each generation, the population is cut down again to the former population size (fittest individuals are kept).
After a given number of generations the fittest individual (the best subset of variables) is returned.

4 Applying the procedure

4.1 The Dortmund data

The data consisted of 170 observations of 200 variables, referring to 170 subdistricts of the city. All variables were total numbers (of inhabitants, females, births,...), so in order to make them comparable across districts, we first constructed normalized variables like '*fraction of female inhabitants*', '*birth rate*' and so on. The result was a set of 57 variables describing features like

i. age distribution
ii. births, deaths, migration
iii. motoring
iv. buildings, housing
v. employment, welfare
vi. some of the above broken down by sex or citizen/alien status

12 out of the 170 observations were considered as outliers; they showed extreme values in some variables, and by checking the corresponding district

on a city map, one could see that these were either extremely sparsely populated or contained some special feature like a boarding school, an old people's home, etc. These were then ignored in the further analysis.
The four groups that we considered should be represented are described by points i, ii, iv and v of the above enumeration. The remaining variables form a group that does not necessarily have to appear.
Grouping the variables by correlations in this case resulted in either huge numbers of subgroups, most of which containing only one variable, or the respective lower/upper correlation bound would be of insignificant order, leading to rather meaningless groupings. So we eventually dropped the automatic grouping approach and only used the subgroups arranged by variable meanings.
Each of the 4 groups should be represented by 1 variable in the final variable subset. In order to keep the data comprehensible, we set the maximum number of variables to 6. That forces the algorithm to choose 1 variable from each of the 4 groups, the remaining variables can then be picked arbitrarily.
Another quantity to be defined beforehand is the number of clusters. After some data exploration, repeated application of the procedure for different values and inspection of the resulting clusterings we found that the different city districts indicated the presence of 4 clusters that repeatedly showed up with a variety of variable sets.

4.2 Results

The 'optimal' set of variables, with respect to the clustering quality measure and restrictions, that we found, is shown in Table 1.

Table 1. Clustering variables and their means.

		Cluster			
Variable	**Group**	**1**	**2**	**3**	**4**
fraction of population of age 60–65	i.	0.057	0.065	0.064	0.083
moves to district per inhabitant	ii.	0.075	0.054	0.035	0.025
apartments per house	iv.	7.831	5.331	3.367	2.524
people per apartment	iv.	1.877	1.676	2.216	2.029
fraction of welfare recipients	v.	0.129	0.031	0.066	0.023
fraction of aliens of employed people	vi.	0.274	0.073	0.086	0.032

Figure 1 displays the distribution of the clusters across the city map. Clusters 1 and 2 roughly cover the city center, subdividing it into north (1) and south (2), while clusters 3 and 4 cover the remaining suburbs (roughly northwest and southeast).
The greatest differences are between clusters 1 (center north) and 4 (south-

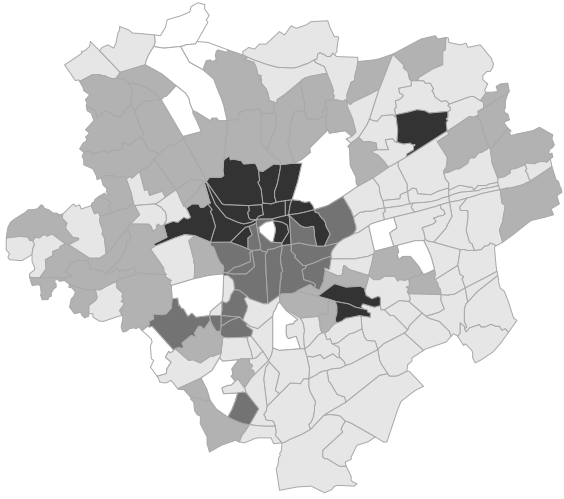

Fig. 1. Map of Dortmund showing the 4 clusters (from Cluster 1=darkgrey to Cluster 4=lightgrey; white districts are the outliers that were omitted).

east suburbs). Cluster 1 has a low fraction of older inhabitants, great fractions of aliens and welfare recipients, more migration and many apartments per house while cluster 4 takes the opposite extreme values. Clusters 2 and 3 are both more or less between these two extremes and differ by their buildings/housing structure: cluster 2 (center south) has more apartments per house and the fewest people per apartment while cluster 3 (northwest suburbs) has the most people per apartment.

4.3 Comparing the results

Clustering the data by *all* variables instead of a subset leads to pretty similar maps, for both the traditional k-means-algorithm and EM clustering based on gaussian mixture models.
Differences become evident when it comes to interpretation. When clustering with all variables, the different variable types (as indicated in the table in section 4.1) are weighted by the number of variables in each of the groups, which are rather random. In contrast, in the approach presented here these proportions are set manually. Also, a selection of necessary variables and elimination of noise variables does not take place. Using only a subset of variables, clusters can thus be easily characterized by the distribution of the (far fewer) variables that were actually used.

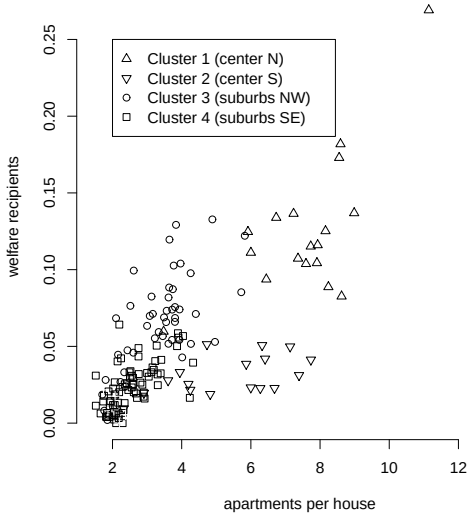

Fig. 2. Two of the variables discriminating the clusters.

5 Summary

The variable selection problem was approached by introducing a quality measure for clusterings and certain restrictions to retain as many information as possible from the complete data set in the variable subset. The optimal variable selection was then performed by a genetic optimization algorithm.
For the Dortmund data, the attempt to define variable subgroups based on correlations proved to be impractical, so the variables were only grouped manually by their respective meanings. Data exploration suggested the presence of 4 clusters. The application of the developed procedure resulted in a plausible set of discriminating variables and a reasonable distribution of the clustered districts across the map. While actual clustering results are similar to those of traditional methods, the necessary data was reduced to a minimum on which to focus any possible further investigation.

References

FRALEY, C. and RAFTERY, A.E. (2002): `mclust`: Software for model-based clustering, density estimation and discriminant analysis. *Technical Report, Department of Statistics, University of Washington*. See `http://www.stat.washington.edu/mclust`.

GOLDBERG, D.E. (1989): *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Boston.

HALL, M.A. (1999): Correlation-based feature subset selection for machine learning. *PhD thesis, Department of computer science, University of Waikato*.

IHAKA, R. and GENTLEMAN, R. (1996): R: A language for Data Analysis and Graphics. *Journal of Computational and Graphical Statistics 5, Nr. 3, 299-314.* See also `http//:www.r-project.org`

KAUFMAN, L. and ROUSSEEUW, P.J. (1990): *Finding Groups in Data: An Introduction to Cluster Analysis.* Wiley, New York.