Machine Learning, 35, 25–39 (1999)

Concept Learning and Feature Selection Based on Square-Error Clustering

BORIS MIRKIN mirkin@dimacs.rutgers.edu
Center for Discrete Mathematics & Theoretical Computer Science (DIMACS), Rutgers University, 96 Frelinghuysen Road, Piscataway, NJ 08854-8018
and Central Economics-Mathematics Institute (CEMI), Moscow, Russia.

Editor: Douglas Fisher

Abstract. Based on a reinterpretation of the square-error criterion for classical clustering, a "separate-and-conquer" version of K-Means clustering is presented and a contribution weight is determined for each variable of every cluster. The weight is used to produce conjunctive concepts that describe clusters and to reduce or transform the variable (feature) space.

Keywords: Clustering, variable weights, conjunctive concepts, feature selection, feature space transformation

1. Introduction

The classical approach to clustering based on the entity-to-entity similarities or distances has significant limitations in concept learning contexts. Michalski and Stepp (1992) indicate that a major difficulty is that the "classical approach has no mechanisms for selecting and evaluating attributes *in the process* of generating clusters" (p.169).

This paper presents a modified approach to classical cluster analysis that exploits a natural mechanism for evaluating the cluster-specific importance of variables. The approach is based on a known, but reinterpreted decomposition of the variance of the data explained by the classification structure. The approach determines a cluster-specific contribution of each variable to the variance of the data. The contribution weight of a variable is proportional to the deviation (squared) of the variable's within-cluster mean from its grand mean, which goes in line with suggestions in data mining that the more deviant a feature is from a standard (the grand mean, in this case), the more interesting it is (Fayyad, Piatetsky-Shapiro & Smyth, 1996). Each contribution weight is a part of a clustering criterion to be maximized, not a posterior quality measure. Based on these insights, a "separate-and-conquer" version of the K-Means clustering method produces clusters one by one, not simultaneously, and relaxes the problem of defining a partition size in advance.

Contribution weights can be evaluated for any category structure, both for a determined one (e.g., through clustering) and a predefined one (e.g., as in supervised learning). The variables with greatest contribution towards a cluster (or cluster structure) may be used to form a conjunctive concept that approximately describes the cluster. When the clusters are based on the square-error criterion, a small number of the variables in a concept is sufficient to well approximate the clusters. When the clusters are predefined, as in a learning-from-examples task, some of them can be spread over the variable space so they "interfere" with each other. In this case, no simple conjunctive concepts can describe the clusters distinctively, and it may be desirable to transform the variable space to allow a better conjunctive description of the classes.

The most contributing variables can be used in variable space reduction or transformation. The approach utilized is related to the so-called Forward/Backward Search Selection (FSS/BSS) algorithms for feature selection in machine learning (e.g., Aha & Bankert, 1995). There are some distinctions, however, based on the evaluation function and intermediate criteria involved.

In Section 2 the clustering approach and a separate-and-conquer version of K-Means clustering are described. In Section 3, the contribution weights are defined and the directions outlined above for exploiting them are fleshed out. Section 4 concludes by highlighting some positive and questionable aspects of the approach.

2. A Separate-and-Conquer Partitional Clustering Procedure

This section discusses the square-error clustering criterion in the context of the square data scatter decomposition. The section then describes and illustrates a separate-and-conquer version of K-Means clustering that sequentially extracts clusters from the "main body" of the entities.

2.1. Data Representation

Let us consider a data matrix $X = (x_{ik}), i \in I, k \in K$, where rows $x_i = (x_{ik})$ correspond to entities (instances), i, and their components x_{ik} are corresponding values of quantitative variables (features), k. Moreover, assume these entities are partitioned into groups (classes, clusters).

Such a data set is presented in Table 1, where the entities are "archetypal psychiatric patients" fabricated by experienced psychiatrists from Stanford University. Each datum is described by seventeen variables: [w1.] Somatic concern. [w2.] Anxiety. [w3.] Emotional withdrawal. [w4.] Conceptual disorganization. [w5.] Guilt feelings. [w6.] Tension. [w7.] Mannerisms and posturing. [w8.] Grandiosity [w9.] Depressive mood. [w10.] Hostility. [w11.] Suspiciousness. [w12.] Hallucinatory behavior. [w13.] Motor retardation. [w14.] Uncooperativeness. [w15.] Unusual thought content. [w16.] Blunted effect. [w17.] Excitement.

The values of the variables are 0-6 severity ratings. The patients are partitioned into four classes of mental disorders: depressed (manic-depressive illness), manic (manic-depressive illness), simple schizophrenia, and paranoid schizophrenia. Each class contains eleven consecutive individuals that are considered typical of that class. The table is published in Mezzich and Solomon (1980, pp. 60-63), along with a detailed description of the data.

Such a data set is traditionally standardized into data matrix $Y = (y_{ik})$,

$$y_{ik} = \frac{x_{ik} - a_k}{b_k}, \; i \in I, k \in K, \tag{1}$$

where a_k is the mean of observed values of variable k (that is, $a_k = \sum_{i \in I} x_{ik}/|I|$) and b_k is equal to unity or the standard deviation, $\sigma_k = (\sum_{i \in I}(x_{ik} - a_k)^2/|I|)^{1/2}$, depending on the user's decision.

Table 1. Disorder data: archetypal patients measured on 17 psychopathological items from Mezzich and Solomon (1980), p.62.

No	w1	w2	w3	w4	w5	w6	w7	w8	w9	w10	w11	w12	w13	w14	w15	w16	w17
1	4	3	3	0	4	3	0	0	6	3	2	0	5	2	2	2	1
2	5	5	6	2	6	1	0	0	6	1	0	1	6	4	1	4	0
3	6	5	6	5	6	3	2	0	6	0	5	3	6	5	5	0	0
4	5	5	1	0	6	1	0	0	6	0	1	2	6	0	3	0	2
5	6	6	5	0	6	0	0	0	6	0	4	3	5	3	2	0	0
6	3	3	5	1	4	2	1	0	6	2	1	1	5	2	2	1	1
7	5	5	5	2	5	4	1	1	6	2	3	0	6	3	5	2	3
8	4	5	5	1	6	1	1	0	6	1	1	0	5	2	1	1	0
9	5	3	5	1	6	3	1	0	6	2	1	1	6	2	5	5	0
10	3	5	5	3	2	4	2	0	6	3	2	0	6	1	4	5	1
11	5	6	6	4	6	3	1	0	6	2	0	0	6	4	4	6	0
12	2	2	1	2	0	3	1	6	2	3	3	2	1	4	4	0	6
13	0	0	0	4	1	5	0	6	0	5	4	4	0	5	5	0	6
14	0	3	0	5	0	6	0	6	0	3	2	0	0	3	4	0	6
15	0	0	0	3	0	6	0	6	1	3	1	1	0	2	3	0	6
16	3	4	0	0	0	5	0	6	0	6	0	0	0	5	0	0	6
17	2	4	0	3	1	5	1	6	2	5	3	0	0	5	3	0	6
18	1	2	0	2	1	4	1	5	1	5	1	1	0	4	1	0	6
19	0	2	0	2	1	5	1	5	0	2	1	1	0	3	1	0	6
20	0	0	0	6	0	5	1	6	0	5	5	4	0	5	6	0	6
21	5	5	1	4	0	5	5	6	0	4	4	3	0	5	5	0	6
22	1	3	0	4	1	4	2	6	3	3	2	0	0	4	3	0	6
23	3	2	5	2	0	2	2	1	2	1	2	0	1	2	2	4	0
24	4	4	5	4	3	3	1	0	4	2	3	0	3	2	4	5	0
25	2	0	6	3	0	0	5	0	0	3	3	2	3	5	3	6	0
26	1	1	6	2	0	0	1	0	0	3	0	1	0	1	1	6	0
27	3	3	5	6	3	2	5	0	3	0	2	5	3	3	5	6	2
28	3	0	5	4	0	0	3	0	2	1	1	1	2	3	3	6	0
29	3	3	5	4	2	4	2	1	3	1	1	1	4	2	2	5	2
30	3	2	5	2	2	2	2	1	2	2	3	1	2	2	3	5	0
31	3	3	6	6	1	3	5	1	3	2	2	5	3	3	6	6	1
32	1	1	5	3	1	1	3	0	1	1	1	0	5	1	2	6	0
33	2	3	5	4	2	3	0	0	3	2	2	0	0	2	4	5	0
34	2	4	3	5	0	3	1	4	2	5	6	5	0	5	6	3	3
35	2	4	1	1	0	3	1	6	0	6	6	4	0	6	5	0	4
36	5	5	5	6	0	5	5	6	2	5	6	6	0	5	6	0	2
37	1	4	2	1	1	1	0	5	1	5	6	5	0	6	6	0	1
38	4	5	6	3	1	6	3	5	2	6	6	4	0	5	6	0	5
39	4	5	4	6	2	4	2	4	1	5	6	5	1	5	6	2	4
40	3	4	3	4	1	5	2	5	2	5	5	3	1	5	5	1	5
41	2	5	4	3	1	4	3	4	2	5	5	4	0	5	4	1	4
42	3	3	4	4	1	5	5	5	0	5	6	5	1	5	5	3	4
43	4	4	2	6	1	4	1	5	3	5	6	5	1	5	6	2	4
44	3	5	5	5	2	5	4	5	2	4	6	5	0	5	6	5	5

For the data in Table 1, all the variables are expressed in the same 7-grade scale; there is no need for further normalization; all $b_k = 1$ in this case.

Two other data sets considered below are taken from the Irvine public repository. These are Iris (150 by 4 matrix partitioned in three 50-element classes) and small Soybean (47 by 35 data matrix partitioned in four classes and reduced to 47 by 21 format since 14 of the columns [variables] are constant and are not used in the following computations; in the subsequent computations, one more variable, 27, has been excluded as coinciding with another one, 26). Although the Soybean data are represented primarily by categorical variables, their codes will be considered quantitative in this paper. Both of the data sets involve rather different variables and will be normed by $b_k = \sigma_k$.

2.2. *Representation of Clusters and Clusterings*

Assume that the clustering structure for a data set $Y = (y_{ik}),\ i \in I,\ k \in K$, is a partition of I into m nonoverlapping clusters, S_t. Each cluster S_t is presented along with its “standard” point $c_t = (c_{tk})$, which is a vector of variable value means within cluster S_t: $c_{tk} = \sum_{i \in S_t} y_{ik}/|S_t|$ $(t = 1, ..., m)$. Each subset S_t represents an extensional description of a clustering, while each mean vector, c_t, is an intensional cluster representation.

It is well known (e.g., Jain & Dubes, 1988, p.95; Mirkin, 1990) that the following equality holds for any clustering $(S, c) = \{S_t, c_t|\ t = 1, ..., m\}$.

$$\sum_{i \in I} \sum_{k \in K} y_{ik}^2 = \sum_{t=1}^{m} \sum_{k \in K} c_{tk}^2 |S_t| + \sum_{t=1}^{m} \sum_{i \in S_t} \sum_{k \in K} (y_{ik} - c_{tk})^2 \tag{2}$$

The left part in this equation represents the total variance of the data (up to the constant factor $1/|I|$) since the variables have been centered by (1). Obviously, it is the sum of the variances of the individual variables; each of the variables contributes the same value $|I|$ in the sum when $b_k = \sigma_k$ in (1), which implies that

$$\sum_{i} \sum_{k} y_{ik}^2 = |K||I|$$

when $b_k = \sigma_k$.

The two terms of equation (2) are usually interpreted according to terminology of analysis of variance (ANOVA) in statistics: the inter-group and within-group variance, respectively (Hand & Taylor, 1987; Jain & Dubes, 1988). Yet another interpretation is possible: the first term is the contribution of the cluster structure to the total variance while the second is the unexplained part of the variance (Mirkin, 1990).

The unexplained (or within-group) variance in the right part of (2) is the well-known square-error clustering criterion (Jain & Dubes, 1988). This is to be minimized with respect to the clustering, (S, c), that is sought. The square-error criterion can be rewritten in terms of the Euclidean distances between row-vectors $y_i = (y_{ik})$ and corresponding standard points:

$$L(S, c) = \sum_{t=1}^{m} \sum_{i \in S_t} d^2(y_i, c_t) \tag{3}$$

where $d(x, y) = (\sum_{i \in I}(x_i - y_i)^2)^{1/2}$ is Euclidean distance between vectors $x = (x_i)$, $y = (y_i)$, $i \in I$.

2.3. The Clustering Procedure

The moving-center method (K-Means, ISODATA) is one of the most popular classical clustering techniques for alternating minimization of the square-error criterion (3). Starting with a m class partition of I, or with m tentative standard points or 'seeds', c_t, selected somehow, the algorithm repeatedly performs the following two-step iteration: (1) update the partition based on the standard points: when all c_t are given, make each S_t the set of y_i that are nearest (by Euclidean distance) to $c_t,\ t = 1, ..., m$; (2) update the standard points: when all S_t are given, compute c_t as the mean of the within-cluster vectors. The algorithm stops when the updating procedure does not change the clustering. The method also can be employed incrementally by updating the centers and clusters after each particular instance is processed.

The moving-center method requires prior information about the number of clusters and initial standard points (seeds) or clusters. A method that exploits many of the same mechanisms as the moving-center method, but which mitigates the need for prior knowledge, separates clusters from the set of instances one by one. In machine learning this has been called a "separate-and-conquer" strategy (Pagallo & Haussler, 1990). The separate-and-conquer procedure extracts an initial cluster $S_1 \subset I$ with its standard point c_1; the complementary set represents the main "body" of the instances, which serves as the source for separating additional clusters one by one. This is reflected in that fact that the main body's standard point is fixed at 0, given the normalization of (1), and it is not changed during the entire clustering computation.

Table 2. Algorithm SCC (Separate-and-Conquer Clustering).

```
Step 0. $t \leftarrow 1$.

Step 1. (Selecting an extreme). Pick a point, $y_{i^*}$, maximizing $d(0, y_i)$, $i \in I$.
Take $c_t = y_{i^*}$ as the initial center (seed) of $S_t$.

Step 2. (Updating the cluster). Define cluster $S_t = \{i : d(y_i, c_t) \leq d(y_i, 0)\}$
of points $y_i$ around $c_t$ to separate them from the origin.

Step 3. (Updating the center). Compute gravity center $c'_t$ for $S_t$.
Compare $c'_t$ with the previous center $c_t$.
If there is no difference, $c_t$ and $S_t$ are $t$-th cluster.
Else let $c_t \leftarrow c'_t$ and go to Step 2.

Step 4. (Excluding the entities). Set $I \leftarrow I - S_t$.
If $I = \emptyset$, end. Else set $t \leftarrow t + 1$ and go to Step 1.
```

Algorithm SCC of Table 2 fleshes out this separate-and-conquer procedure. Clustering terminates when all objects have been placed in a cluster (i.e., separated out).

As an intuitive example, let us consider when there is only one variable, uniformly distributed across its range. Then, having the zero point in the midrange, SCC initially separates one fourth of the range at one extreme, then one fourth at the other extreme, with one half of the range left to be cut at extremes again. In contrast, a traditional divisive version of the method, with both of the standard points updated, will initially produce a split just in the midrange, splitting then each of the clusters by half, and so forth.

This example reflects a general property that the size of a SCC-designed cluster depends on its distance from the origin as stated in Step 2: the nearer to the origin, the less the diameter of the cluster! Thus, SCC could be modified to allow the user to specify the origin based on user's knowledge of the variable space: the better the knowledge, the smaller the classes. This bias can be useful, for instance, for a robot-planning system: the robot must learn and classify the nearest part of the world in more detail than more distant objects.

Placing the origin as the grand mean point initially causes SCC to separate the subset of instances corresponding to extreme combinations of the variable values. Thus, the separated subset can be considered on its own as "interesting" in the sense of Fayyad, Piatetsky-Shapiro & Smyth (1996).

This interpretation can be supported with the mathematics underlying the method. At any step of SCC, for two classes found around 0 and c_t, decomposition (2) looks like

$$\sum_{i \in I} \sum_{k \in K} y_{ik}^2 = \sum_{k \in K} c_{tk}^2 |S_t| + \sum_{i \in I - S_t} d^2(y_i, 0) + \sum_{i \in S_t} d^2(y_i, c_t).$$

SCC alternately minimizes the sum of the last two items in that expression; thus, it maximizes the contribution of the cluster separated, $\sum_{k \in K} c_{tk}^2 |S_t| = d^2(c_t, 0)|S_t|$, to the total variance of the data (for the current I).

This suggests yet another stopping rule for SCC: the separate-and-conquer process may terminate when the cumulative contribution of the separated clusters becomes greater than a user-specified threshold (e.g., 70% of the variance), rather than continuing until no unclassified entities remain in the main body. This stopping rule leads to discovery of "extreme" clusters and a residue around the grand mean. When the user wants no entities unclassified, the standard points of discovered clusters can be considered as the tentative standard points (seeds) for follow-up application of the general moving-center method.

It can be proved that SCC is a nonoverlapping version of the principal cluster analysis method (Mirkin, 1987; Mirkin, 1990). In these publications, however, each cluster is separated out by an instance-by-instance addition method, not the version of the moving-center method described here.

EXAMPLE: When SCC was applied to the Disorder data, the algorithm produced four clusters coinciding with the four mental disorder classes in Table 1 (in the same order) except that entity 21 was clustered with the fourth class; the same phenomenon has been reported in Mezzich and Solomon (1980), p. 69 - 73, using complete linkage, ISODATA and K-Means clustering.

For the small Soybean data set SCC produced six clusters; two of them coincide with predefined classes. The other four clusters are splits of the other two predefined classes in two subclasses each. With the number of clusters fixed at 4^1, SCC and subsequent application of the moving-center (K-Means) method gives exactly the four predefined classes.

For the Iris data set, the algorithm sequentially finds 6 clusters, some of which are parts of the predefined ones while the others are mixed. A confusion matrix for the 6 SCC discovered clusters and 3 predefined classes is presented in Table 3. The fact that predefined classes

2 and 3 are spread over 4 SCC clusters confirms the well-known property that they are not compact. □

Table 3. Confusion matrix for the SCC clusters and predefined classes of the Iris data.

Predefined classes	SCC discovered clusters						
of the Iris data	1	2	3	4	5	6	Total
1		49	1				50
2			12	17	2	19	50
3	26		2	7	15		50
Total	26	49	15	24	17	19	150

3. Contribution Weights and Their Uses

In this section, a cluster-specific measure of variable salience is suggested based on decomposition (2). The measure is employed for approximate conceptual description of clusters and for two-stage forward/backward feature selection along with transformation of the variable space when necessary.

3.1. Determining Variable Weights

An "importance weight" of a variable as a term in an overall additive evaluation measure is not uncommon in machine learning: Gennari (1989) and Fisher, Xu, Carnes, Reich, Fenves, Chen, Shiavi, Biswas and Weinberg (1993) show how to exploit each variable's contribution toward a category utility score function as its "salience". Based on the formula in (2), we suggest an extension of this idea that determines a cluster-specific salience weight for each variable.

The total contribution of the clustering towards variance is equal to

$$V = \sum_{t=1}^{m} \sum_{k \in K} c_{tk}^2 |S_t|, \tag{4}$$

which can be presented as the sum of cluster contributions, $v_t = \sum_{k \in K} c_{tk}^2 |S_t|$, or as the sum of variable contributions, $v(k) = \sum_{t=1}^{m} c_{tk}^2 |S_t|$. These contributions can be employed as salience weights of either clusters or variables. Still smaller items in the sum, $c_{tk}^2|S_t|$, lead to the relative contributions of variables, k, due to clusters, t:

$$v(k/t) = c_{tk}^2 |S_t| / v_t = c_{tk}^2 / \sum_{k \in K} c_{tk}^2. \tag{5}$$

EXAMPLE: In Table 4, the standard point values (means) c_{tk} are presented for class 2 of the Disorder data (Table 1), along with corresponding relative contribution weights, $v(k/t)$. □

Table 4. Cluster 2 described with 17 psychopathological variables: the central value in the original scale, the central value in the standardized scale, and the relative contribution, per cent.

Variable	Mean value (original scale)	Mean value c_{tk} (standardized scale)	Contribution weight (%)
w1	1.27	-0.95	8.20
w2	2.27	-0.61	3.42
w3	0.18	-1.45	19.08
w4	3.18	0.03	0.01
w5	0.45	-0.71	4.58
w6	4.82	0.90	7.42
w7	1.09	-0.41	1.52
w8	5.82	1.15	11.94
w9	0.82	-0.81	6.00
w10	4.00	0.52	2.49
w11	2.36	-0.29	0.76
w12	1.45	-0.34	1.07
w13	0.09	-0.85	6.62
w14	4.09	0.35	1.13
w15	3.18	-0.32	0.96
w16	0.00	-0.97	8.54
w17	6.00	1.34	16.26

The meaning of these variable-to-cluster contribution weights can be seen from Table 4: as far as the standardized value c_{tk} measures deviation of the mean of the variable k in the t-th cluster from the grand mean (in the standard deviation scale), its contribution is proportional to the deviation squared (which is equivalent to what the analysis of variance methodology does when the variables are considered uncorrelated, see Hand and Taylor (1987)). Loosely speaking, the farther c_{tk} from zero (which is the grand mean here), the more separated the cluster from the other entities based on the variable k. In terms of Fayyad, Piatetsky-Shapiro and Smyth (1996), $v(k/t)$ measures the "degree of interestingness" of the variable k in cluster t with respect to its "standard" mean value.

EXAMPLE: Decomposition (4) of the explained part of the data scatter by 17 variables and 4 classes in Table 1 is shown in Table 5. The entries are of format "contribution of a variable-to-class pair, $c_{tk}^2|S_t|$ / its relative contribution, $v(k/t)$, per cent", obviously extended in the marginal column and row according to their contents. □

Weights can be employed in various learning problems including (a) concept learning, (b) feature selection, and (c) space transformation, which will be considered in subsequent sections.

3.2. *Representing Clusters Approximately by Conjunctive Concepts*

There are many systems that learn logical descriptions of a subset of the instances (e.g., Michalski, 1992; Quinlan, 1986; Wnek & Michalski, 1994). Cluster-specific contribution weights give yet another way of finding approximate conjunctive descriptions for every cluster separately.

Table 5. Decomposition of the explained part of the data scatter according to Table 1 by 17 variables and 4 classes; the entry 12.35/8.06 in the left corner means: pair Variable w1/Class 1 contributes 12.35 to the scatter and w1 contributes 8.06% to "explanation" of the data by Class 1.

Variable	Class 1	Class 2	Class 3	Class 4	General, $v(k)$
w1	12.35/8.06	9.94/8.20	0.40/0.48	0.07/0.06	22.76/4.86
w2	6.59/4.31	4.15/3.42	7.83/6.59	4.15/3.78	21.48/4.59
w3	3.68/2.40	23.15/19.08	7.43/8.83	0.03/0.03	34.29/7.32
w4	6.71/4.38	0.01/0.01	0.85/1.00	2.52/2.30	10.09/2.15
w5	25.73/16.81	5.56/4.58	1.15/1.37	2.70/2.46	35.14/7.50
w6	3.50/2.28	9.00/7.42	7.50/8.92	2.59/2.36	22.59/4.82
w7	3.68/2.40	1.84/1.52	3.33/3.96	2.10/1.92	10.95/2.34
w8	11.60/7.58	14.49/11.94	9.38/11.15	7.09/6.45	42.56/9.09
w9	25.88/16.91	7.28/6.00	0.62/0.73	2.58/2.35	36.35/7.76
w10	8.40/5.49	3.02/2.49	6.59/7.83	13.88/12.64	31.88/6.81
w11	3.40/2.22	0.92/0.76	3.40/4.04	21.58/19.66	29.30/6.26
w12	3.61/2.36	1.30/1.07	1.30/1.54	17.47/15.91	23.67/5.05
w13	24.37/15.92	8.04/6.62	0.12/0.15	6.01/5.48	38.55/8.23
w14	4.61/3.01	1.37/1.13	6.44/7.66	12.35/11.25	24.78/5.29
w15	1.57/1.02	1.16/0.96	1.16/1.38	11.62/10.59	15.52/3.31
w16	0.00/0.00	10.36/8.54	18.34/21.80	1.20/1.09	29.90/6.39
w17	7.37/4.81	19.73/16.26	9.52/11.31	1.84/1.68	38.46/8.21
Total	153.05/32.68	121.33/25.91	84.11/17.96	109.78/23.44	468.26/100.00

Based on the right column in Table 4, let us pick the features that most contribute to class 2 in Table 1, to form a conjunctive conceptual description of the cluster. Initially, let us take the range of the most salient variable, w3 (contribution 19.08%), within the cluster 2: it is interval $[0, 1]$, the boundary points included. Conceptual description $W : 0 \leq w3 \leq 1$ covers all 11 individuals belonging to class 2; however, there are two other individuals, 4 from class 1 and 35 from class 4, which also satisfy condition W. This relates to what could be called 'precision error', PE, of the concept W with respect to a class $S \subset I$. The PE is defined as the number of elements from outside S satisfying W, divided by the number of all elements outside S. Actually, PE is just the proportion of false positives for the concept W as a description of S.

To decrease $PE(W) = 2/33$, let us pick the next most contributing variable, w17 (contribution 16.26%), and consider the conjunctive concept formed by the within-cluster ranges of both, w3 and w17: $W : 0 \leq w3 \leq 1 \ \& \ w17 = 6$. Obviously, precision error of this combined category equals zero. Moreover, it is easy to see that the first term of the concept is not necessary; concept $W : w17 = 6$ corresponds to all 11 individuals from class 2 and no one else.

This is an example of the situation when a less contributing variable (w17) gives a better conceptual description than a more contributing variable (w3), which shows that the statistics-based contribution weights reflect only tendencies of the logical relations rather than exact patterns of them.

A general algorithm, ACCL, for approximate conjunctive conceptual description $W(S)$ of a class $S \subset I$ is presented in Table 6. The data matrix is assumed normalized with formula

Table 6. Algorithm ACCL (Approximate Conjunctive Concept Learning).

Step 0. Find the means, c_k, of the variables $k \in K$ within S and consider list L of the variables in descending order by their contribution weights, c_k^2.
Let $W(S)$ be empty and $PE = 1$.

Step 1. Remove the first variable, x_k, from list L and consider combined concept $W' = W(S)$ & $m_k \leq x_k \leq M_k$ where m_k and M_k are minimum and maximum of x_k within S, respectively. Compute $PE(W')$.

Step 2. If $PE(W') < PE(W(S))$, put $W(S) \Leftarrow W'$ and $PE(W(S)) \Leftarrow PE(W')$.
If $PE(W') = PE(W(S))$, then $W(S)$ and $PE(W(S))$ are left unchanged.

Step 3. If $L = \emptyset$ or $PE(W(S)) = 0$, go to Step 4. Otherwise, go to Step 1.

Step 4. For every conjunctive term w_h of $W(S)$ ($h = 1, ..., H$ where H is the number of terms in $W(S)$), consider conjunctive concept $W(S)|w_h$ which is $W(S)$ with w_h removed. Pick that h which makes $p_h = PE(W(S)|w_h)$ minimum over all H terms (if there are several such hs, take that one corresponding to the least contributing variable).

Step 5. If $p_h = PE(W(S))$ or $H > n$ (where n is a user-specified threshold for the number of conjunctive terms), go to Step 6; otherwise, end.

Step 6. Remove w_h from $W(S)$ by putting $W(S) \Leftarrow W(S)|w_h$, $PE(W(S)) \Leftarrow p_h$ and $H \Leftarrow H - 1$. Go to Step 4.

(1). The degree of approximation is characterized by the precision error $PE(W(S))$. A standard stopping criterion, the maximum number of conjunctive terms in the concept $W(S)$, may be also involved as a user-defined integer n.

In its steps 1 through 3, the algorithm ACCL in Table 6 one-by-one adds features to produce a conjunctive description $W(S)$ (forward-selection strategy FSS). Then, in steps 4 through 6, it one-by-one removes those terms that minimally affect the precision error of $W(S)$ (backward-selection strategy BSS). If no maximum number of terms, n, is prespecified, only those terms are removed that do not change $PE(W(S))$ at all.

Since ACCL is a local search algorithm, its resulting conjunction $W(S)$ may have a non-minimum precision error.

EXAMPLE: Let us apply the algorithm ACCL to class 3 of Table 1 based on the variable weights presented in the corresponding column of Table 5. The maximum weight variable with regard to class 3 is w16 (contribution 21.80%). Its within-cluster range $W : 4 \leq w16 \leq 6$ covers 5 individuals in the other classes (one in class 4 and four in class 1), which makes $PE(W) = 5/33$. Adding the within-cluster range of the next most contributing variable, w17 (contribution 11.31%), we have $W : 4 \leq w16 \leq 6$ & $0 \leq w17 \leq 2$, which makes $PE(W) = 4/33$ since the previously covered individual from class 4 does not satisfy the combined condition. Variable w16 cannot be removed from the concept (Step 2) since this makes precision error grow. Then, considering each of the next most contributing variables, w8, w6, and w3, we can see that adding none of them decreases $PE(W)$. For example, the within-cluster range of w6 is $[0, 4]$ which is compatible with values of w10 for all the four individuals from class 1 (2, 9, 10, and 11) satisfying W. However, the next contributing variable, w2, has its within-cluster range, $[0, 4]$, incompatible with the values of w2 for individuals 2, 10, and 11, which makes the concept $W : 4 \leq w16 \leq 6 \;\&\; 0 \leq w17 \leq 2 \;\&\; 0 \leq w2 \leq 4$ have $PE(W) = 1/33$. Moreover, $w17$ now can be removed from W at Step 2, with $PE(W) = 1/33$ unchanged. This leads to $W : 4 \leq w16 \leq 6 \;\&\; 0 \leq w2 \leq 4$ as a two-term solution to the problem. Subsequently adding w5 to W reduces $PE(W)$ to zero.

The concepts found for the other Disorder classes are: $w9 = 6$ (class 1, PE=0), $w17 = 6$ (class 2, PE=0), $5 \leq w11 \leq 6 \;\&\; 4 \leq w10 \leq 6$ (class 4, PE=1/33).

Predefined classes of the small Soybean data set are exactly described (with PE=0) by the within-class ranges of variables v23 (class 1), v26 (class 2), v4 & v24 (class 3), and v35 & v12 (class 4).

In the Iris data set, predefined classes can be described by the following concepts found with algorithm ACCL: $1 \leq w3 \leq 1.9$ (class 1, PE=0), $3.0 \leq w3 \leq 5.1 \;\&\; 1.0 \leq w4 \leq 1.8$ (class 2, PE=0.08), and $1.4 \leq w4 \leq 2.5 \;\&\; 4.5 \leq w3 \leq 6.9$ (class 3, PE=0.18). Precision errors of the two latter conjunctions cannot be reduced by adding other variables' ranges. Large precision error for two of the Iris classes supports the conclusion that they are dispersed in the variable space.

Table 7 overviews the results found by algorithm ACCL for all the clusterings considered, with $n = 2$ (i.e., with only two conjunctive terms permitted). □

Table 7. Mean precision error (per cent) over all clusters in each of the six clusterings considered.

	Disorder		Soybean		Iris	
	SCC clusters	Predefined classes	SCC clusters	Predefined classes	SCC clusters	Predefined classes
Number	4	4	6	4	6	3
Mean PE, %	1.5	1.5	0.0	0.0	4.6	8.7

3.3. Feature Selection

The problem of reducing the space dimensionality by selecting a most informative variable (feature) subset has attracted considerable attention in machine learning (see John, Kohavi & Pfleger, 1994; Aha & Bankert, 1995). Feature (variable) selection algorithms for learning a class or partition (clustering) involve two major components: an evaluation function, which evaluates performance of feature subsets, and a search algorithm, which searches the space of feature subsets.

The algorithm ACCL can be considered as a procedure of forward/backward feature space search to learn a class (cluster) along with the precision error as an evaluation function. The set of variables occurring in the approximate conjunction describing the class forms a feature space. The set-theoretic union of these sets over all clusters of a partition forms a feature space selected for learning the partition.

EXAMPLE: Predefined classes of the small Soybean data set have been exactly learnt (with PE=0) by the variables v23 (class 1), v26 (class 2), v4 and v24 (class 3), and v35 and v12 (class 4), which leads to a six-dimensional subspace (generated by these variables) in the original 20-dimensional space.

Subset $\{w2, w8, w9, w11, w16, w17\}$ corresponds to the conjunctive concepts found for four predefined Disorder classes (up to $PE = 1/33$) in the original 17-dimensional space.

Three predefined classes of the Iris set have been described, in the previous example, with two variables, $w3$ and $w4$, as the only ones occurring in the conjunctive descriptions found with algorithm ACCL. Although the precision errors of the conjunctions describing classes 2 and 3 are high, addition of the remaining variables, $w1$ and $w2$, does not improve the quality of approximate conjunctive description. □

3.4. Transforming the Feature Space

The ACCL based method for finding approximate conjunctive descriptions of classes and selecting feature subspaces does rather well when the classes are located in different zones of the original feature space. The method works poorly in domains like the Iris data where classes are intermingled in the feature space so that a class cannot be separated into that box-like cylinder volume which corresponds to an ACCL output conjunction.

However, the method's performance can be improved by transforming and combining the variables. A generator of the compound variables can be utilized as follows.

Denote the set of original variables by B and the set of ACCL selected variables by A. For every $x \in A$ and $y \in B$, compute $x * y$, and, also, if $x \neq y$, $x + y$, $x - y$, x/y and y/x. Consider all the variables found (plus those in A) as the resulting feature space $F(A, B)$.

Applying ACCL to the resulting space $F(A, B)$ may only improve the quality of conjunctive descriptions of the classes. Reiterating the procedure (with A being the set of ACCL selected variables on the previous iteration, while maintaining B as the set of original variables), we can arrive at conjunctive descriptions of the classes with as small a precision error as needed. This should be considered a hypothesis-driven (actually, cluster-driven) constructive induction system (Wnek & Michalski, 1994).

EXAMPLE: Let us consider the set of four original Iris variables as B and the set of two variables found with ACCL, w3 and w4, as A. Algorithm ACCL, applied to $F(A, B)$ for description of classes 2 and 3 (do not forget, class 1 has been distinctively separated by $w3$ alone), produces conjunctions $1.18 \leq w1/w3 \leq 1.70$ & $3.30 \leq w3 * w4 \leq 8.64$ (class 2, PE=0.04) and $7.50 \leq w3 * w4 \leq 15.87$ & $1.80 \leq w3 - w2 \leq 4.30$ (class 3, PE=0.02) (with the number of conjunctive terms restricted to be not larger than 2). The errors have become much smaller, but still they may be considered too high. Can the precision errors be reduced to just 1 percent?

Putting the compound variables involved, $w1/w3$, $w3 * w4$, and $w3 - w2$, as A and leaving B as it is, an update $F(A, B)$ is computed to give rise to the following ACCL produced conjunctions: $2.86 \leq w1 * w2/w3 \leq 4.77$ & $3.30 \leq w3 * w4^2 \leq 15.55$ (class 2, PE=0.04) and $3.24 \leq (w3 - w2) * w4 \leq 9.89$ & $1.35 \leq w3 * w4 - w1 \leq 8.70$ (class 3, PE=0.02). Although the errors of the two-term conjunctions have not changed, the new feature space leads to a better four-term description of class 2 (with PE decreased to 0.01).

Taking A as consisting of the four new variables, $w1 * w2/w3$, $w3 * w4^2$, $(w3 - w2) * w4$, and $w3 * w4 - w1$, and B unchanged, the algorithm ACCL applied to $F(A, B)$ leads to the following final conjunctions: $0.64 \leq w2 * (w3 - w2) * w4 \leq 4.55$ & $0.21 \leq w2/(w3 * w4^2) \leq 0.74$ (class 2, PE=0.01) and $4.88 \leq w3 * w4^2 - w1 \leq 31.20$ & $-2.85 \leq (w3 - w2) * w4 - w1 \leq 2.19$ (class 3, PE=0.01). It should be added that class 1 can be distinctively separated with one of these variables, $w2 * (w3 - w2) * w4 \leq -3.07$ (class 1, PE=0.00).

The process of combining of variables is stopped at this point to underscore a trade-off needed between the exactness and complexity of cluster descriptions, which parallels similar trade-offs in other description techniques such as regression analysis.

The predefined three Iris classes are somewhat more “compact” in the final four-dimensional feature space. Not only have the precision errors of conjunctive descriptions been reduced, but also clustering results have been improved. Reclustering the Iris data set in the final feature space with the algorithm SCC produces 4 clusters with the confusion matrix in Table 8. Obviously, the predefined classes are more visible here than in the original-space clustering shown in Table 3. □

Table 8. Confusion matrix for the SCC clusters in the transformed feature space and predefined classes of the Iris data.

Predefined classes	SCC discovered clusters				
of the Iris data	1	2	3	4	Total
1	5		45		50
2				50	50
3		42		8	50
Total	5	42	45	58	150

4. Conclusion

Classical square-error clustering can be employed as an effective tool for machine learning since there are additive items in the criterion that can be interpreted as cluster-specific contributions of the variables to explain the data. Based on properties of the least-squares criterion, a separate-and-conquer clustering procedure is suggested to design clusters one by one while keeping the center of the “data body” unchanged. Both the contribution weights of the variables and the clustering procedure appear compatible with the concept of “interestingness” in data mining since both concentrate on the items that are farthest from the average.

The cluster-specific contribution weight is utilized as an intermediate easy-to-calculate scoring function to address the following problems: (a) cluster description; (b) feature selection; (c) feature space transformation. The approach is illustrated with small datasets from public domains. Fisher (1996) and Hanson, Stutz, and Cheeseman (1991) describe very different ways of identifying variables relevant to a cluster’s description, though neither approach is concerned with formulating conjunctive descriptions from such variables.

There is no explicitly expressed relationship between the least-squares criterion and the evaluation criterion utilized, the precision error. Thus, the approach should be considered as a useful heuristic rather than a comprehensive methodology in solution the machine learning tasks. Another approach to post-clustering characterization is to apply a well-established supervised learning system such as C4.5 (Quinlan, 1993) to the clusters. An approach like this, but using AQ15 (Michalski, Mezetic, Hong & Lavrac, 1986) as the supervised system, was taken by Lu and Chen (1987).

In this paper, only the quantitative data case has been considered, although the square-error approach could be extended to the case when the variables may be Boolean or nominal, via the so-called bilinear clustering model (Mirkin, 1990). However, such a development is associated with some supplementary aspects (as the problem of standardization of mixed data) which should be treated in another place.

Finally, feature selection and transformation have been used primarily in post-clustering characterization of clusters. An obvious direction for future work is to integrate selection and transformation into the clustering process itself (as demonstrated informally with the Iris data, Table 8). Similar, but distinct ideas of tightly coupling automatic feature selection and unsupervised learning have been explored by Devaney and Ram (1997).

Acknowledgments

The author is indebted to the referees for their helpful comments and to Dr Doug Fisher for his thorough revising suggestions going well beyond the usual editing style. The author thanks the Office of Naval Research for its support under grants number N00014-93-1-0222 and N00014-96-1-0208 to Rutgers University.

Notes

1. Here we assume that yet another termination condition is used for SCC; the algorithm terminates when a user-specified number of clusters has been separated out – 4 clusters in this case (i.e., when $t \leftarrow 5$ in step 4 of SCC).

References

1. Aha, D.W., & Bankert, R.L. (1995). A comparative evaluation of sequential feature selection algorithms. In D. Fisher and H.-J. Lenz (Eds.), *Learning from Data: AI and Statistics*. Springer-Verlag: New York.
2. Devaney, M., & Ram, A. (1997). Efficient feature selection in conceptual clustering. *Proceedings of the Fourteenth International Conference on Machine Learning* pages 92-97, Morgan Kaufmann: Nashville, TN.
3. Fayyad, U.M., Piatetsky-Shapiro, G., & Smyth P. (1996). From data mining to knowledge discovery: An overview. In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy (Eds.), *Advances in Knowledge Discovery and Data Mining*. AAAI Press/The MIT Press: Menlo Park, CA.
4. Fisher, D. (1996). Iterative optimization and simplification of hierarchical clusterings. *Journal of Artificial Intelligence Research, 4*: 147-180.
5. Fisher, D., Xu, L., Carnes, J.R., Reich, Y., Fenves, S.J., Chen, J., Shiavi, R., Biswas, G., & Weinberg, J. (1993). Applying AI clustering to engineering tasks. *IEEE Expert, 8*: 51-60.
6. Gennari, J.H. (1989). Focused concept formation. *Proceedings of the Sixth International Workshop on Machine Learning* pages 379-382, Morgan Kaufmann: Ithaca, NY.
7. Hand, D.J., & Taylor, C.C. (1987). *Multivariate Analysis of Variance and Repeated Measures*. Chapman & Hall: London.

8. Hanson, R., Stutz, J., & Cheeseman, P. (1991). Bayesian classification with correlation and inheritance. *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence* (pp. 692-698). Morgan Kaufmann: San-Mateo, CA.
9. Jain, A.K., & Dubes, R.C. (1988). *Algorithms for Clustering Data*. Prentice Hall: Englewood Cliffs, NJ.
10. John, G., Kohavi, R., & Pfleger, K. (1994). Irrelevant features and the subset selection problem. *Proceedings of the Eleventh International Machine Learning Conference* pages 121-129, Morgan Kaufmann, New-Brunswick, NJ.
11. Lu, S.C., & Chen, K. (1987). A machine learning approach to the automatic synthesis of mechanistic knowledge for engineering decision making. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 1*: 109-118.
12. Mezzich, J.E., & Solomon, H. (1980). *Taxonomy and Behavioral Science*. Academic Press: London
13. Michalski, R.S. (1992). Concept learning. In S.C. Shapiro (Ed.), *Encyclopedia of artificial intelligence*. J. Wiley & Sons: New York.
14. Michalski, R.S., Mozetic, I., Hong, J., & Lavrac, N. (1986). The multipurpose learning system AQ15 and its testing application to three medical domains. *Proceedings of the Fifth National Conference on Artificial Intelligence* (pp. 1041–1045). Morgan Kaufmann: Philadelphia, PA.
15. Michalski, R.S., & Stepp, R.E. (1992). Clustering. In S.C. Shapiro (Ed.), *Encyclopedia of artificial intelligence*. J. Wiley & Sons: New York.
16. Mirkin, B.G. (1987). Method of principal cluster analysis. *Automation and Remote Control, 48(10)*: 1379-1386.
17. Mirkin, B.G. (1990). A sequential fitting procedure for linear data analysis models. *Journal of Classification, 7*: 167-195.
18. Pagallo, G., & Haussler, D. (1990). Boolean feature discovery in empirical learning. *Machine Learning, 5*: 71-99.
19. Quinlan, J.R. (1986). Induction of decision trees. *Machine Learning, 1*: 81-106.
20. Quinlan, J.R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA.
21. Wnek, J., & Michalski, R.S. (1994). Hypothesis-driven constructive induction in AQ17-HCI: A method and experiments. *Machine Learning, 14*: 139-168.

Received September 26, 1994
Accepted August 17, 1998
Final Manuscript August 17, 1998