Analysis of Olympic Heptathlon Data

Brian P. DAWKINS, Peter M. ANDREAE, and Paul M. O'CONNOR*

An incremental clustering algorithm is described and applied to 1992 Olympic heptathlon data to produce characterizations of groups of the leading athletes. Results are compared with those obtained by classical clustering techniques. The incremental technique is of order $n \log n$, where n is the number of observational units and can be combined with Classification and Regression Trees (CART) to obtain descriptions of the clusters. Brief reference is made to other possible ways of analyzing the data, including the use of correspondence analysis. The analysis can be used to show which events are most critical in determining the better class of heptathlete.

KEY WORDS: CART; Clustering; Correspondence analysis; Data analysis; Incremental algorithm.

1. INTRODUCTION

Small, multidimensional data sets are hard to analyze, because any structure may be only weakly suggested. When the data set is also fairly homogeneous, any structure in the data is quite subtle and can be difficult to elucidate. Olympic heptathlon and decathlon data are interesting examples of such data sets. This article describes the application of an incremental clustering technique to heptathlon data and compares the results with three standard techniques. We demonstrate that the incremental clustering technique is an effective method for uncovering interesting structure in small data sets.

The heptathlon competition consists of seven events run over two consecutive days. The events of the first day, in order, are the 100 meter hurdles (hurd), the high jump (hiju), the shot put (shot), and the 200 meter run (m200). The events of the second day are the long jump (loju), the javelin (jave), and the 800 meter run (m800). (The codes in parentheses are used in tables and graphics throughout the article). The athletes are placed according to their total score, which is the sum of the point scores for each event. The point score for an event is computed from the raw performance measure using International Amateur Athletic Federation (IAAF) formulas that are not readily available.

In this article we consider only the 1992 women's heptathlon data; moreover, we consider data only for those athletes for whom a complete set of observations was recorded. The data set that we analyze consists of the raw performance measures of the athletes; we have not used the point scores. Table 1 shows the full data set, together with the total scores and placings of the athletes. The events occur in a specified order, so the variables have a natural ordering. But this ordering has not been used in the following analysis. Except where noted, the data set was standardized so that each column had mean 0 and standard deviation 1.

2. METHODS

The principle methods we used are two standard, classical, agglomerative clustering methods, a standard fast algorithm, and a technique that we developed from an algorithm published in the artificial intelligence literature. Further details on the methods used are provided in the Appendix.

Standard classical methods come in many different flavors; we use only two: compact clustering and average clustering, such as that implemented in Proc CLUSTER (SAS Institute, Inc. 1985) or *hclust* (Becker, Chambers, and Wilk 1988). Because these methods use distance matrices or surrogates, they tend to be of order n^2 or worse.

Because classical methods become inefficient as the sample size increases, various faster methods have been developed, such as the procedure FASTCLUS as implemented by SAS, the KMEANS algorithm of Spath (1980), and the CLARA routine of Kaufman and Rousseeuw (1990). These routines are roughly $O(n)$ or $O(n \log n)$.

The last method, called DySect, has been adapted from the CLASSIT algorithm that appeared first in the artificial intelligence literature (Gennari, Langley, and Fisher 1989) and that uses novel local maximization methods not available in standard statistical software. Details of the algorithm involved are given in the Appendix and are available as a technical report by the authors (Andreae, Dawkins, and O'Connor 1993). We intend to submit suitable software to statlib as soon as possible.

3. ANALYSIS

3.1 Compact Clustering

The dendrogram based on compact clustering is shown in Figure 1. A reasonable interpretation of the dendrogram is that there are basically two clusters, each containing exactly half of the instances (observational units). One could postulate further substructure in the data. For example, the cluster of the better athletes may contain a distinguished subclass consisting of Belova and Joyner-Kersee, and the cluster of the worse athletes may contain at least two individuals (Barber and Chouaa) who are significantly different from the others. But the reliability of such distinctions is dubious, and these would normally require additional support, possibly from another clustering method, such as the one described in the following subsection.

3.2 Average Clustering

The dendrogram from average clustering is shown in Figure 2. An interpretation does not seem easy to come by.

* Brian P. Dawkins is Senior Lecturer, Institute of Statistics and Operations Research, Peter M. Andreae is Senior Lecturer, Computer Science Department, and Paul M. O'Connor is a graduate student, Institute of Statistics and Operations Research, Victoria University of Wellington, New Zealand. The authors thank the associate editor and two anonymous referees for sound suggestions and advice. They also acknowledge partial support from Telecom NZ in the development of the DySect clustering algorithm.

Table 1. Pentathlon Results for the 1992 Olympics

TE: Name	Hurd (sec)	Hiju (m)	Shot (m)	m200 (sec)	Loju (m)	Jave (m)	m800 (sec)	Total (points)	Place
Aro	13.87	1.70	13.11	25.44	6.23	45.42	14.31	6,030	18
Atroshchenko	14.03	1.79	13.05	24.39	6.22	45.18	10.90	6,251	12
Barber	14.79	1.55	10.71	25.66	5.76	.00	21.59	4,530	26
Beer	13.48	1.82	13.23	23.93	6.01	48.10	9.49	6,434	6
Belova	13.25	1.88	13.77	23.34	6.82	41.90	5.08	6,845	2
Bond-Mills	14.31	1.76	12.96	25.01	6.01	43.30	18.84	5,897	21
Braun	13.25	1.94	14.23	24.27	6.02	51.12	14.35	6,649	3
Carter	13.97	1.85	14.35	24.54	6.10	37.58	8.62	6,256	11
Chouaa	16.62	1.64	12.24	25.44	5.88	44.40	24.30	5,278	25
Clarius	14.10	1.82	15.33	24.86	6.13	45.14	8.83	6,388	7
Court	13.48	1.58	13.85	23.95	6.10	52.12	31.21	5,994	19
Dimitrova	13.23	1.70	14.68	23.31	6.11	44.48	7.90	6,464	5
Greiner	13.59	1.79	14.35	24.60	6.38	40.78	14.16	6,300	9
Joyner-Kersee	12.85	1.91	14.13	23.12	7.10	44.98	11.78	7,044	1
Kaljurand	13.64	1.73	12.83	25.29	6.35	47.42	19.61	6,095	17
Kamrowska	13.48	1.70	14.49	24.40	6.12	44.12	10.96	6,263	10
Lesage	13.75	1.88	13.48	25.24	5.99	41.28	15.57	6,141	15
Marxer	13.94	1.67	12.40	24.43	5.74	41.08	17.53	5,749	24
Nastase	12.86	1.82	14.34	23.70	6.49	41.30	11.22	6,619	4
Nazaroviene	13.75	1.76	14.49	25.20	6.03	44.42	14.95	6,142	14
Rattya	13.96	1.70	12.97	25.09	5.90	49.02	15.18	5,993	20
Skjaeveland	13.73	1.82	12.07	24.48	6.08	35.42	22.19	5,869	22
Teppe	14.06	1.79	12.69	26.13	5.65	52.58	24.42	5,847	23
Vaidianu	14.04	1.73	14.96	25.28	6.11	49.00	18.40	6,152	13
Wlodarczyk	13.57	1.82	13.91	24.18	6.20	43.46	14.96	6,333	8
Zhu	13.64	1.82	14.26	23.83	5.99	45.12	31.84	6,123	16

NOTE: The 800 meters value gives the number of seconds in excess of 2 minutes.

There is certainly no clear indication of meaningful clusters. In particular, there is no support for the two major groupings suggested by the previous method, although there are some weak consistencies between the two dendrograms. For example, Barber and Chouaa are singled out in both, and many of the initial pairings are the same in each.

3.3 FASTCLUS

FASTCLUS is not hierarchical—it simply supplies a partition of the original data. Moreover, the user generally needs to consider various "tuning" options, including the maximum number of clusters allowed. We have chosen to illustrate only a few of the many possible partitions that FASTCLUS can generate. With such a small data set, it seems reasonable to start small and specify at most two or three clusters. Table 2 shows the results of six runs, each performed with different random seeds.

One of the more disturbing aspects of Table 2 is the lack of consistency between the partitions produced by the different runs. The instances 5, 14, and 19 making up the first

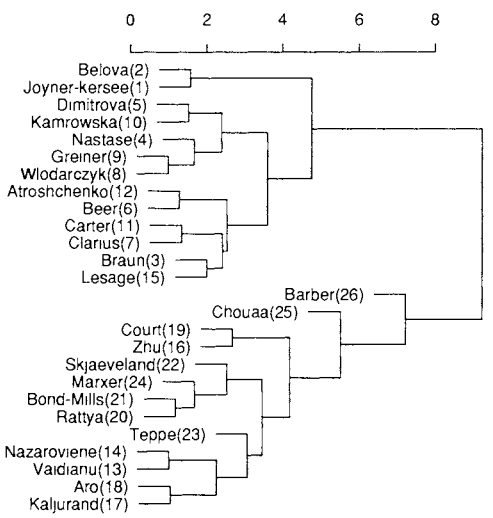

Figure 1. Compact Clustering Dendrogram of the 1992 Olympic Heptathlon Data. The final placings are shown in brackets. The strong suggestion of two clusters is probably rather misleading and is possibly a manifestation of the known propensities of the method to form clusters with approximately equal diameters. Note that despite the linking of Braun (3) and Lesage (15), on the whole the initial pairings are between athletes with closer final placings.

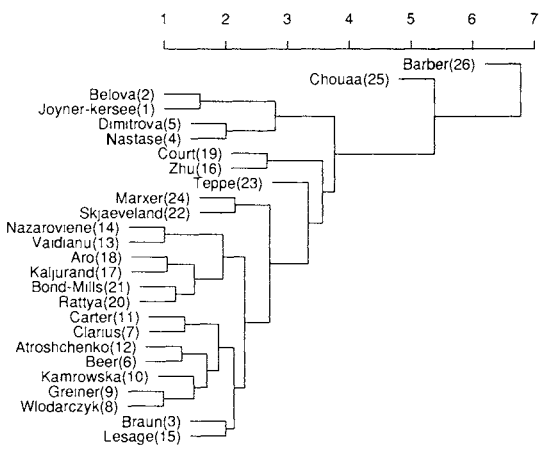

Figure 2. Average Clustering Dendrogram of the 1992 Olympic Heptathlon Data. The final placings are shown in brackets. This should be compared with Figure 1. There is no clear suggestion of clustering in the data. Note that there are similar but not identical initial pairings to those in Figure 1, and that in particular Braun and Lesage are linked. Barber and Chouaa are prominent here, even more so than in Figure 1, and are evidently quite clearly separated in the appropriate metric from the other athletes.

Table 2. Examples of Partitions Using FASTCLUS on the Pentathlon Data

Sample	Cluster 1	Cluster 2	Cluster 3
1	(5 14 19)	(the rest)	
2	(3 9)	(the rest)	
3	(3 9 11 23)	(the rest)	
4	(4 5 7 11 13 14 16 19 25 26)	(the rest)	
5	(1 2 4 5 6 7 9 10 12)	(3 7 11 13 14 15 16)	(17 18 19 20 21 22 23 24 25 26)
6	(1 2 4 7 8 10 11 12 13 16 20 21 25)	(3 6 9 15 17 18 22 23 26)	(5 14 19)

NOTE: The numbers denote the final placings.

cluster in the first run do occur again in run 6 as a cluster, so that the partition of run 6 could be viewed as a refinement of the partition of the first run. But this cluster of 5, 14, and 19 is split in the partitions of two of the other runs, and there appears to be almost nothing that is consistent across all partitions. There are certainly no clear clusterings. Normal use of this method could require many runs to discover structure, and by running the algorithm many more times with varying numbers of clusters, some consistencies may appear.

3.4 Incremental Clustering

The DySect algorithm produces a multiway tree as part of its output. This is one of the fundamental ways in which it differs from most classical techniques. Another difference is its facility for at least partially backing out of poor choices (see Andreae et al. 1993 for more details).

The Appendix briefly reviews the terminology appropriate to multiway trees. Figure 3 represents such a structure and shows schematically part of the tree produced by an application of the DySect algorithm to the heptathlon data. It should be stressed here that DySect will generally produce a different tree for different orderings of the data, given its incremental nature. This should be compared with the variability of the partitions determined by FASTCLUS, although that arises from a different source. But two or three repetitions are often sufficient to settle on the salient features of the data, and repeated application of the algorithm tends to show certain invariant features on the whole. We have chosen to illustrate this with a suitable structure. In such a structure, the root node at the top of the tree contains all the instances—26 in this case. Below the root are two nodes, labeled Class 2 and Class 3. Referred to as the children of the root, these nodes contain 3 and 23 members. Each of these nodes has 3 children; each child of Class 2 has a single member, whereas the children of Class 3 have 10, 10, and 3 members. DySect generates further substructure below Classes 7, 8, and 9, but we have not shown them here, because we are not concerned with such fine structure for the purposes of this discussion. They are available for the analyst if required.

Figure 4 shows more detail of the classes found by the incremental clustering algorithm. For each node, it displays boxplots for each of the variables, with lines connecting the medians of each boxplot. These lines form a profile of each class (similar to the profile plots discussed, for example, in Spath 1980 and Wegman 1990), giving a graphical characterization of the classes that enables easy comparison of the different classes.

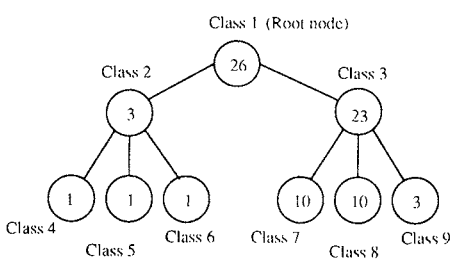

Figure 3. Schematic Display of a DySect Clustering of the 1992 Olympic Heptathlon Data. This is a display of the top-level nodes derived from an application of the incremental algorithm to the 1992 Olympic heptathlon data. Note that the structure is multiway in the sense that a given node may have more than two children. The figures within the nodes give the number of elements in each node. Thus Class 2 has 3 elements and Class 8 has 10 elements.

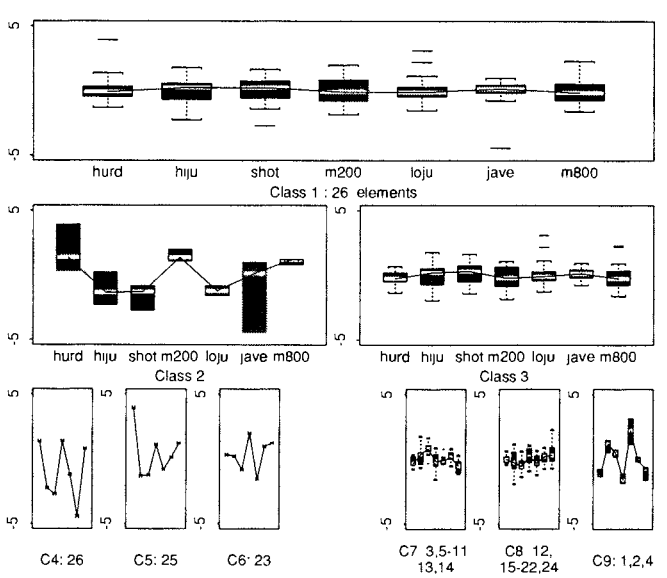

Figure 4. Boxplot Display of a DySect Clustering of the 1992 Olympic Heptathlon Data. This is essentially a more detailed version of the tree in Figure 3. The text beneath the bottom row of plots indicates the class number and the final placings of the athletes assigned to that node. Thus the extreme left node is Class 4, which has exactly one member (Barber), who placed 26th. The lines in each boxplot connect the medians of the variables, producing a profile for the class. Note the characteristic shapes of the profiles. For convenience we refer to Classes 4, 5, 6, and 8 as of type W and Classes 7 and 9 as of type M, because the profiles suggest these letters, particularly in the more extreme cases.

The profile plots in Figure 4 clearly show the qualitative difference between Classes 3, 7, and 9 on the one hand and the remaining classes on the other hand. If the sequences of boxplots are taken in the chronological order of events (as they are in the figure), then the shapes of the profiles are either "W" like or "M" like. We thus refer to the two groups of classes as type W and type M.

Note that the actual shapes of the profiles are not particularly significant, because they arise from the fact that lower values are advantageous in running events (i.e., hurdles, 200 meters, and 800 meters) and high values are advantageous in the other events. Because the data are normalized so that the mean is 0, one would expect the better athletes to tend to have an M profile and the worse athletes to tend to have a W profile. This is confirmed by examination of the membership of the classes. Class 9, consisting of three of the top four athletes, has a very pronounced M profile, and Class 7, consisting of all of the remaining top 14 athletes apart from the 12th place getter (Atroshchenko), has a less pronounced M profile. If the performance of the athletes had been very inconsistent across the various events, these M and W patterns would not have been as marked. Thus the patterns make clear that good performance across all events is essential to success in the heptathlon.

Incidentally, note from the display of the boxplots for the root node the variability of the medians and interquartile ranges for the standardized data. This suggests a possible inequity in the scoring system. The medians of the actual point values for the different events (with interquartile range in the parentheses) are 1,015(77), 966(148), 782(92), 932(114), 880(69), 752(106), and 893(118).

The spread in these values constitutes a prima facie case that some of the events may carry an unduly high weighting in the total score. One might argue that the median performer should score roughly 1,000 points in each event, and that the spread of values should also be standardized as appropriate. But a detailed evaluation of this claim would require knowledge of the formulas and rationales by which the IAAF computes the points from the raw performances, and we will not pursue this question any further.

Class 2 contains three of the worst performing athletes, each of whom has a distinctively poor performance in at least one event. They are distinguished from the athletes in Class 3 by the unevenness of their performances. Note that Barber (26th place) scored 0 in the javelin, presumably because she did not achieve a legal throw.

Class 9 contains three of the four best performers. An obvious question is why Braun (3rd place) did not join the others in Class 9. Similarly, one can ask why Atroshchenko (12th place) was assigned to Class 8 rather than Class 7. The clustering structure is useful in highlighting these two athletes as not fitting the regular pattern. But the structure does not immediately show why these two individuals are different.

One method is to examine the raw data directly. In the case of a very small data set such as this one, one can suggest reasons fairly easily. In the case of Braun, it would appear to be because she performed spectacularly well in the javelin and high jump, but relatively poorly in the 200 meter run and long jump. This is quite different from the members of Class 9 who performed well in the 200 meter run and long jump. In the case of Atroshchenko, it might appear that a relatively poor performance in the hurdles has relegated her to the lower class.

But Classification and Regression Trees (CART) (Breiman, Friedman, Olshen, and Stone 1984) offers an alternative explanation. It can be used to generate a different kind of characterization of the classes that can answer this kind of question more readily. CART will construct a classification tree to distinguish between two (or more) classes of data. The variables used for the splits in the tree can be interpreted as the most significant variables for the distinction. We used the CART software as implemented in S-PLUS (Clark and Pregibon 1992) to produce a classification tree to distinguish the type M classes from the type W classes. The result is shown in Figure 5. The tree suggests that type M and type W athletes can be distinguished largely on the basis of their performances in the shot put and the 800 meter run. To the extent that these are the events that most characterize the better athletes, we can then explain Atroshchenko's exclusion from Class 7 by her poor performance in these two events. We can do the same kind of analysis for Class 9 alone, which results in a trivial classification tree, with a single split based on the long jump. We thus can explain the exclusion of Braun from Class 9 by her much poorer performance in the long jump.

A CART tree constructed on the basis of the two obvious clusters in the compact clustering dendrogram was slightly more complex than the tree for type M and type W classes, but it also used the same two events (800 meter run, shotput) for the top-level splits. This suggests that these two events are the most effective for distinguishing the better half of the athletes.

This kind of analysis is not restricted to the classes generated by the clustering methods; we also constructed CART trees for several other classes of athletes, based solely on their placings. These trees showed that the top few athletes were

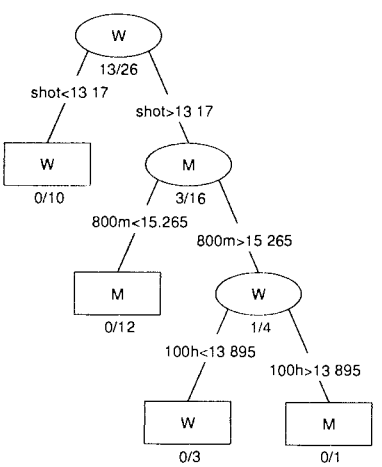

Figure 5. A CART Classification Tree Derived from Clusters Defined by DySect and as Displayed in Figure 4. The figures under each node represent the misclassification rates. Thus in the right child of the root node, 3 of the 16 elements are of type W. The elements have been assigned classification of type M or W based on the type of node to which they belong, as discussed in Figure 3.

best distinguished by their performances in the 200 meter run, hurdles, high jump, and long jump. It is interesting that this set of events is disjoint from the set of events that are most effective for identifying the better half of the athletes.

CART enabled us to explore a range of alternative clusterings, from which it could be ascertained that there was no clear or definitive split into two classes of better and worse athletes. The apparently strong distinction suggested by the dendrogram of Figure 1 thus is somewhat misleading; DySect's split at the second level of the tree is perhaps more indicative of the data. It is important to note, however, that CART's analysis is based on treating the variables quite independently, so that each split in the tree is based on a single event. It also requires a response variable (in this case, a binary partition of the athletes). The clustering methods, on the other hand, take all the variables into account for each split and do not require any response variable. CART and the clustering methods provide complementary kinds of analysis. We suggest that the combination of using the clustering methods to suggest classes based on the data itself and using CART to provide characterizations of classes leads to a more effective analysis.

4. DISCUSSION

4.1 Comparison of Results of Clustering Methods

The different clustering methods produce different kinds of clusterings, and it is difficult to compare them directly. (For further discussion in this connection, see Gordon 1987 and Milligan 1980.) The FASTCLUS clusterings seem to be less helpful on this data set than the other kinds of clusterings, partly because they are not internally consistent and partly because it is difficult to discover what the clusters represent. As far as the more conventional distance-based methods are concerned, apart from suggesting that Braun is in some way separate from the other top place getters because she tends to not be grouped with them, there appears to be little other consistency.

It is much easier to relate the clusterings generated by DySect and the compact clustering method. In particular, the two major classes in the compact clustering correspond fairly closely to the type M and type W categories of the DySect clustering. Furthermore, two of the three members of Class 2 in the DySect clustering (Barber and Chouaa) are also distinguished to some extent in the dendrograms, although the distinction is made much more clearly in the DySect clustering. Class 9 of the DySect clustering, which contains three of the top four athletes, does not have a clear parallel in the standard clusterings, although the standard clusterings do link the top two athletes closely.

The one feature common to all three clustering methods is the distinction between Braun (3) and the other top four athletes. Each method grouped her with different athletes, but none of them put her with the other top four. This seems to provide strong evidence that Braun is somewhat unusual, and it suggests that a more detailed investigation of her performance might lead to insight into the nature of successful heptathlon athletes.

Regardless of the detailed similarities and differences of the different clusterings of the data, we would argue that the multiway tree produced by DySect elucidates the class structure of a data set more effectively than do the dendrograms produced by the classical clustering methods. Further, we would argue that the boxplot and profile representation of the classes in the hierarchy provides a graphic characterization of the classes that is particularly helpful in understanding the nature of the clustering. We note that this representation could be used in conjunction with any of the clustering methods and is not restricted to the incremental clustering algorithm. Note, however, that clusterings based on standard dendrograms tend to be subjective.

All of the clustering was done on the standardized data rather than on the points scored. This was largely because the formulas by which the points are generated is not known, and we were uncertain of their effects. But clustering on the points scored and comparing with the other clusterings might give some insight into the "fairness" of the scoring system; any substantial difference would seem to imply a significant change in the metric from the raw to point scores. Justification for this change would need to be sought from those responsible for devising the system.

4.2 Correspondence Analysis

As suggested by one of the referees, we have used correspondence analysis to identify some kinds of structure in the data. As a simple example, consider Figure 6, which displays a standard analysis as described, for example, by Greenacre (1984). In these diagrams one can interpret a vector from the origin to a point representing an event as defining, at least approximately, a gradient direction for that event. This is somewhat similar to the biplot (Gabriel 1971); see Dawkin's (1989) discussion in the context of athletics data. Recalling that in the running events low times are advantageous, whereas in the field events greater distances and heights are better, we can see that Joyner-Kersee (1) was clearly domi-

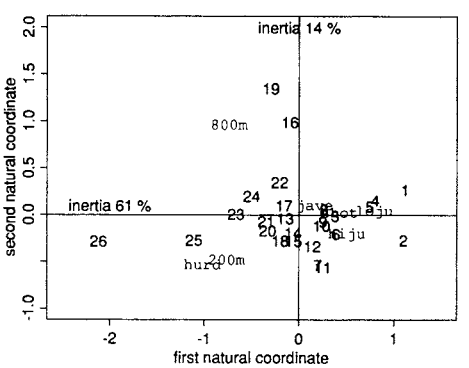

Figure 6. Correspondence Analysis of the 1992 Olympic Heptathlon Data. The numbers denote final placings, as noted earlier. The data have been standardized, so that the observations for each variable range from 0 to 1. The total inertia of the two axes is approximately 75%, indicating that although much of the variability in the original data is captured in the plot, there remains a significant proportion left. Note the prominent positions of the running events, which clearly contribute a large amount to the inertia. Less prominent is the javelin, indicating that this event contributes least to the variability of the data when approximated in two dimensions. Note the very clear dominance of Joyner-Kersee (1) in the hurdles, 200 meter run, and the long jump. See the text and the book by Greenacre (1984) for further details on the interpretations possible.

nant in the hurdles and the 200 meter run, whereas Belova (2) was preeminent in the 800 meter run.

It is interesting to note also that the plot clearly suggests that, at least in the best two-dimensional approximation to the data, the 800 meter run is an excellent criterion by which to divide the class into a top half and a bottom half, consistent with the CART analysis, and that the 200 meter run and hurdles have much the same discriminatory power in the sense that they lead to virtually the same rankings. It also appears that the javelin makes a rather weak contribution to the data in this representation, given that it is so close to the origin, and thus contributes little to the inertia about each axis. It is interesting to note that the correlation between the ranks based on the javelin and the final ranks is not significantly different from 0; in fact, the magnitudes of the correlations between ranks based on the individual events and final placings are approximately .72(hurd), .65(hiju), .70(shot), .70(200m), .69(loju), .07(jave), and .78(800m).

Other aspects of the plot could be explored. For example, one could conjecture from the plot that the running events actually dominate the orderings of the individuals. This could be explored with further plots and analyses, which are outside the scope of this article. A similar consideration applies to deciding which single event, pair of events, and so on would best approximate the final result using regression methods.

Correspondence analysis could also be combined with the clustering methods in the same way that we used CART. Applying correspondence analysis to each of the classes found by the clustering methods could provide further insight into the nature of the classes and could possibly be used to explain anomalous individuals.

A further avenue to explore might be to use correspondence analysis to impute a value for Barber in the javelin and to redo the analysis.

4.3 Other Possible Analyses

Other kinds of analysis could provide insight into small data sets such as this one by identifying kinds of structure other than clusterings of the data. One example would be to regress the final points scored on various subsets of the raw data to generate predictors of performance.

Canonical correlation analysis of first day events versus second day events could possibly shed light on the change of level of performance between the two days. Given that the first canonical variates probably can be interpreted as a general level of performance measure on the two successive days, it would be interesting to compare the performances of the best and worst competitors. This could conceivably give some insight into the performers' stamina as opposed to their strength and speed. The other canonical variates may give insight into subtler differences between performers. Similarly, one could compare the throwing events (shot put and javelin) with the running events or with the jumping events.

Further analyses could include examination of the results from the decathlon and results from previous Olympics from the same viewpoints. We have not yet pursued these ideas in any detail.

4.4 Incremental Methods

One of the main objectives in the development of incremental methods such as DySect is to facilitate the analysis of data sets that are too large to handle by more conventional methods. DySect shares with FASTCLUS a considerable speed advantage over the standard distance-based classical techniques, because it is order n or $n \log n$. On larger data sets, this gives DySect a great advantage. The speed advantage was not apparent on the very small data set considered here, but DySect has been used effectively to analyze a data set with more than 1 million cases. (This analysis is under embargo for reasons of commercial secrecy, however.)

DySect also shares with FASTCLUS the limitation that it can produce different clusterings, depending on the order in which the instances are presented. But it is evident that DySect can detect interesting structure that is meaningful to the analyst to the extent that regularities and anomalous individuals can be identified, despite the lack of robustness to the order of presentation of cases.

One additional characteristic of DySect not evident from the previous analysis is that it can largely ignore noise variables while more conventional methods are seriously compromised (see Andreae et al. 1993 for more details).

4.5 Computation

All computation and graphing was carried out on the ISOR Sun network using S-PLUS and C. The C code implementing the incremental clustering technique was written by Paul O'Connor, who also devised the display of Figure 4.

We expect to submit suitable software to statlib in the last quarter of 1994.

APPENDIX: TREES AND CLUSTERING ALGORITHMS

We first introduce briefly the terminology of tree structures, and then give some details of the clustering methods used. More details on the merits of the various classical clustering methods have been provided by Milligan (1980) (see also Gordon 1987).

A.1 Tree Notation

A multiway tree is a structure consisting of nodes. A given node may be a parent node, a child node, or both. If it is a parent node, then there are at least two other nodes designated as its children. If it is a child node, then there is exactly one node specified as its parent. In a nontrivial tree there are at least three nodes consisting of one parent and two children, and in any nontrivial tree there is always one node that is not a child. This is referred to as the root node. A leaf node is a node with a single member (or instance). Thus in Figure 3, Class 1 is the root node, and it has two children: Class 2 and Class 3. Class 2 itself is a parent and has three children: Classes 4, 5, and 6, each of which is a leaf node. (For more details of trees and associated ideas see Clark and Pregibon 1992.)

A.2 Compact Clustering

Compact clustering, also known as *complete linkage*, is an agglomerative method. In such a method, the initial configuration involves treating each observation as a cluster of size one. At any subsequent stage, two clusters are amalgamated into one using the two clusters judged closest by some suitable distance measure. Clustering ceases when all observations are combined into one cluster. In the case of compact clustering, the distance between two

clusters is taken as the maximum of the distances between any point of the first cluster and any point of the second cluster. Because these methods involve a distance matrix, they are generally unsuited to large data sets. Here large may mean more than a couple of hundred observations, depending on the implementation. In view of the huge data sets currently being accumulated, this seems somewhat limited; thus methods such as FASTCLUS have been developed.

A.3 Average Clustering

Average clustering is similar to compact clustering, differing only in the criterion by which extant clusters are to be fused at any given stage. Here the criterion is the average of the distances between points in the two clusters.

A.4 FASTCLUS

FASTCLUS is intended to produce a partition into disjoint clusters. For full details, the SAS Institute, Inc. (1985) documentation should be consulted. The algorithm, minus a few of the more sophisticated details, is roughly as follows. A set of points called *cluster seeds* is chosen by some method, possibly at random. Each observation is assigned to the nearest seed, forming interim clusters. The initial seeds are replaced by the means of these clusters, and the process is repeated until convergence is achieved. One prime advantage of an algorithm of this sort is that it allows much larger data sets to be handled. Two other characteristics that may be seen as drawbacks are that the end results will generally depend on the initial set of cluster seeds chosen and that the result is a partition rather than a hierarchy.

Regarding the first of these characteristics, it is usual to observe that no single clustering is likely to capture the essence of a real data set. The customary advice is that several clusterings using different methods should be applied. In the case of FASTCLUS or its near relatives, several runs may be made using different sets of seeds and the output compared. Any reasonable structure in the data should be evident on carrying out such a comparison. The $O(n)$ nature of the process makes this procedure feasible even for large data sets.

Depending on the nature of the data set, a hierarchical classification structure may be advantageous, in that one can observe "clusters of clusters." The production of a hierarchy comes at a cost, of course, and whether such a structure is desirable depends on the context.

A.5 Incremental Clustering

The method used in this article is derived directly from the CLASSIT algorithm as given by Gennari, Langley, and Fisher (1989). Depending on the implementation, the technique is between $O(n)$ and $O(n \log n)$. At present the implementation is essentially $O(n \log n)$, because it classifies down to leaf node level. After k instances have been processed by the algorithm, a hierarchical structure exists that completely classifies all previous instances. A new instance is then recursively incorporated into the structure, which may change radically in the process. When an instance is to be incorporated into a given node (starting with the root node), various options are explored:

- The new instance may become a new child node of the given node; that is, a leaf node.
- It may be incorporated (recursively) into an existing child node.
- Two of the current children of the given node are merged.
- A child of the given node is replaced by its children.

In the case of the last two options, should one be chosen, the incorporation procedure is reinvoked until the instance is added to a child node. These somewhat novel features are intended to allow a form of backing out from unsuitable choices made earlier. The choice of option to pursue depends on the value of a criterion function. The value for each of the options is calculated (incrementally) and the best one is chosen. Roughly speaking, the criterion chosen is based on favoring clusterings that tend to minimize the combined measure of spread in individual variables relative to the parent node. This means that, at least at the upper levels of the hierarchy, noise variables tend to have little effect on the clustering. Specific details have been given by Andreae et al. (1993).

It is indeed true that the results depend on the order in which the instances are presented, but the same remarks pertain here as to FASTCLUS—that several independent runs should be made and compared. When there is a natural order of presentation of the instances, such as in medical cases being presented for diagnosis, the procedure seems eminently suitable.

It is worth noting that the full tree should be grown only rarely and some form of limitation would normally be imposed, such as setting a minimum node size or perhaps limiting the lengths of the branches of the main tree. This would naturally lead to a decrease in the processing time required, becoming more comparable with the $O(n)$ of FASTCLUS.

[Received April 1993. Revised January 1994.]

REFERENCES

Andreae, P. M., Dawkins, B. P., and O'Connor, P. M. (1993), "DySect: An Incremental Clustering Algorithm," Technical Report 33, Victoria University of Wellington, ISOR.

Becker, R. A., Chambers, J. M., and Wilk, A. R. (1988), *The New S Language,* Pacific Grove, CA: Wadsworth and Brooks/Cole.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984), *Classification and Regression Trees,* Belmont, CA: Wadsworth.

Chambers, J. M., and Hastie, T. J. (1992), *Statistical Models in S.* Pacific Grove, CA: Wadsworth and Brooks/Cole.

Clark, L. A., and Pregibon, D. (1992), "Tree-Based Models," in *Statistical Models in S,* eds. J. M. Chambers and T. J. Hastie. Pacific Grove, CA: Wadsworth and Brooks/Cole.

Dawkins, B. P. (1989), "Multivariate Analysis of National Track Records," *The American Statistician,* 43, 110–115.

Fisher, D. (1987), "A Hierarchical Conceptual Clustering Algorithm," *Machine Learning,* 2, 139–172.

Gabriel, K. R. (1971), "The Biplot-Graphic Display of Matrices With Applications to Principal Components Analysis," *Biometrika,* 58, 453–467.

Gennari, J. H., Langley, P., and Fisher, D. (1989), "Models of Incremental Concept Information," *Artificial Intelligence,* 40, 11–61.

Gordon, A. D. (1987), "A Review of Hierarchical Classification," *Journal of the Royal Statistical Society,* Ser. A, 150, 119–137.

Greenacre, M. J. (1984), *The Theory and Applications of Correspondence Analysis,* London: Academic Press.

Kaufman, L., and Rousseeuw, P. J. (1990). *Finding Groups in Data,* New York: John Wiley.

Milligan, G. W. (1980), "An Examination of the Effect of Six Types of Error Perturbation on Fifteen Clustering Algorithms," *Psychometrica,* 45, 325–342.

SAS Institute, Inc. (1985), *SAS User's Guide: Statistics,* Cary, NC: Author.

Spath, H. (1980), *Cluster Analysis Algorithms,* Chichester, U.K.: Ellis Horwood.

Wegman, E. J. (1990), "Hyperdimensional Data Analysis Using Parallel Coordinates," *Journal of the American Statistical Association,* 85, 664–675.