MULTIDIMENSIONAL DATA ANALYSIS\textsuperscript{1}

Th. Naumann and H. Schiller
Institut für Hochenergiephysik der Akademie der Wissenschaften der DDR,
Berlin–Zeuthen

\textsuperscript{1}This is a reprint from “FORMULAE AND METHODS IN EXPERIMENTAL DATA EVALUATION” published by the European Physical Society (Computational Physics Group) at CERN in January 1984
# Contents

1 Introduction 2

2 Variables, Metrics, Similarity Measures 3

3 Some General Methods of Multidimensional Data Analysis 7
   3.1 Projection Pursuit 7
   3.2 Nonlinear Mapping 8
   3.3 Multidimensional Two Sample Test 9

4 Cluster methods 11
   4.1 Hierarchical Methods 11
      4.1.1 Hierarchical Clustering using the MST 12
      4.1.2 Selection of Jets by Hierarchical Clustering 13
   4.2 Non–hierarchical Techniques 14
      4.2.1 The Valley–Seeking Technique 14
      4.2.2 The Cluster Algorithm CLUCOV 17
      4.2.3 Gelsema’s Interactive Clustering Technique 19
Chapter 1

Introduction

The necessity for the application of multidimensional data analysis in high energy physics arises from at least two reasons:

- Events of a given reaction are described by a (multiplicity dependent) large number of kinematical variables
- The number of events is normally high so that the application of statistical methods becomes essential

Being confronted with a large amount of multidimensional data the physicist often chooses the traditional way of analyzing these data, i.e. to produce one- or two-dimensional projections of the data.

Low-dimensional projections of multidimensional data are often insufficient as they reveal only a small part of the information contained in the data. Multidimensional data analysis therefore aims to use the full experimental information.

If there exists a physical model of the measured process the parameters of this model can be fitted to the data. Such parametric methods as partial wave analysis, analytical multi-channel analysis or prism plot analysis rely heavily on a priori physical knowledge and will not be considered here.

Also beyond the scope of this survey are classification procedures starting from a known category structure within the data and sorting the data points into categories defined before. We describe methods to find out structure in multidimensional data with minimum a priori knowledge of the underlying physics and data structure.

In chapter 2 the importance of a skilled choice of the variables spanning up the multidimensional space and the definition of similarity in the case of cluster methods are discussed. Methods of lowering the dimensionality of the multidimensional space and of comparing multidimensional data sets are presented in chapter 3. Looking for an appropriate separation algorithm the analyst has to take into account the computing demands of different algorithms. The hierarchical algorithms presented in chapter 4.1 are only applicable for relatively small numbers of data points, while the non–hierarchical methods of chapter 4.2 can also be applied to large data sets.

For theoretical discussion and applications of different methods of multidimensional data analysis we refer the reader to the review of Kittel (KITT76) and the proceedings of the meetings on high energy data analysis (CERN76, NIJM78). The reader especially interested in cluster analysis should consult the monographies of Anderberg (ANDE73) and Duran and Odell (DURA74).
Chapter 2

Variables, Metrics, Similarity Measures

Information and structure do not exist by themselves but only in the context of a given application. Such, for example, the noise in a radio receiver contains no information for the ordinary listener, but it can well give some information on a defect in the receiver.

The existence of structure within multidimensional data sets depends largely on the variables that span the multidimensional space and on the metrics defined in this space. For classification procedures one furthermore has to decide on the meaning of similarity and dissimilarity of categories.

The importance of the choice of variables for the ultimate results of the analysis is well illustrated in fig. 2.1. Scaling one coordinate already suggests a new group structure within the four data points.

Figure 2.1: The effect of the scale of variables on the group structure

![Figure 2.1](image)

Leaving out relevant variables naturally makes a meaningful analysis impossible. Adding variables that are not very relevant to the purpose of the analysis but induce nevertheless a partition of the data is clearly misleading.

Another serious problem is the relative scale between variables of different origin. It is sometimes recommended to reduce all variables to standard form (zero mean and unit variance) at the beginning.

These considerations are meant as a warning: multidimensional data analysis cannot reveal some absolute information that preexists in the data. It can merely act as a heuristic tool to generate hypotheses on the structure of the data. The
answers obtained from the analysis depend largely on the questions asked by the analyst.

Multidimensional data in high energy physics most frequently consist of the momentum and energy variables of many particle final states. Therefore we shall consider the problem of the choice of variables in this context.

It would be desirable to find some physical requirements that make the choice of variables less arbitrary or even unique. One approach would be to look for a complete set of Lorentz–invariant variables which does not favour any of the final state particles.

The reaction

\[ a + b \rightarrow 1 + 2 + \ldots + n \]

can be described in terms of the following variables

\[
(a_i) = (p_a p_i) \\
(b_i) = (p_b p_i) \\
(i k) = (p_i p_k) \quad \text{with } i, k = 1, \ldots, n.
\]

These variables are relativistically invariant and invariant under permutations of the incoming particles \(a, b\) and the final state particles \(i, k\).

On the other hand they are directly related to momentum transfers and masses squared so that they are relevant for the dynamics of the reaction to be investigated.

However, the number of invariants exceeds the number of independent variables which is \(3n - 5\) for an \(n\)–particle final state at fixed total energy. To our knowledge there is no subset of variables that maintains the properties of Lorentz–invariance and permutation symmetry. Therefore we have to resort to the somewhat weaker property of quasi–permutation invariance which means permutation invariance up to a linear transformation. This leads to the demand for distance measures that are invariant under linear transformations.

Provided one has such a distance measure, a relativistically and quasi–permutation invariant subset of variables should be good for a multidimensional analysis.

For a three–particle final state one can simply choose the four invariants

\[
(a1), \quad (a2), \quad (b1), \quad (b2)
\]

For four–particle final states, Yang (BYER64, BYCK73) proposed the following seven variables

\[
(a1), \quad (a2), \quad (a3), \quad (b1), \quad (b2), \quad (b3), \quad \Delta_4(1234)
\]

where \(\Delta_4\) is the Gram determinant defined by

\[
\Delta_k(12\ldots k) = \begin{vmatrix} (11) & \ldots & (1k) \\ . & \ldots & . \\ . & \ldots & . \\ (k1) & \ldots & (kk) \end{vmatrix}
\]

4
In order to generalize the Yang variables one has to take into account the kinematic constraints on the invariants. Such a generalization has been performed in (BECK76). For this purpose we define a quantity

\[ Z_m^{(n)} = \left[ (-1)^{m+1} \sum_{i_1}^n \cdots \sum_{i_m}^n \Delta_m(i_1, \ldots, i_m) \right]^{1/m} \]

This quantity is relativistically invariant and quasi-permutationally invariant with respect to the final state particles. Thus it can be used as a generalization of \( \Delta_4 = Z_4^{(4)} \) for more than four particles in the final state. Furthermore, this invariant is constant for \( m = 1 \):

\[ Z_1^{(n)} = \sum_{i=1}^n m_i^2 \]

with \( m_i \) as the mass of particle \( i \) so that only \( Z_m^{(n)} \) with \( m > 1 \) can be incorporated in the variable sets.

For five particles in the final state we have ten invariants

\[
\begin{align*}
(a1) & \quad \ldots \quad (a4) \\
(b1) & \quad \ldots \quad (b4) \\
\text{and} \quad & Z_4^{(5)}, \quad Z_3^{(5)}.
\end{align*}
\]

For six particles we need thirteen variables

\[
\begin{align*}
(a1) & \quad \ldots \quad (a5) \\
(b1) & \quad \ldots \quad (b5) \\
\text{and} \quad & Z_4^{(6)}, \quad Z_3^{(6)}, \quad Z_2^{(6)}.
\end{align*}
\]

For a detailed analysis of many particle final states the full set of \((3n - 5)\) variables should be used. Experience has shown, however, that most of the global information on the reaction mechanisms is already contained in the \((2n - 2)\) four-momentum products (BECK76)

\[
\begin{align*}
(a1) & \quad \ldots \quad (a n - 1) \\
(b1) & \quad \ldots \quad (b n - 1)
\end{align*}
\]

We now turn our attention to the problem of metrics. A non-negative real function \( d(X_i, X_j) \) is called a metric in a \( p \)-dimensional Euclidean space \( E_p \) if

1. \( d(X_i, X_j) > 0 \) for all \( X_i \) and \( X_j \) in \( E_p \)
2. \( d(X_i, X_j) = 0 \) if and only if \( X_i = X_j \)
3. \( d(X_i, X_j) = d(X_j, X_i) \)
4. \( d(X_i, X_j) \leq d(X_i, X_k) + d(X_k, X_j) \)
where $X_i$, $X_j$ and $X_k$ are any three vectors in $E_p$.

The most popular and commonly used metric is the Euclidean metric

$$d_2(X_i, X_j) = \sqrt{\sum_{k=1}^{p} (X_{ki} - X_{kj})^2}$$

An absolute value norm

$$d_1(X_i, X_j) = \sum_{k=1}^{p} |X_{ki} - X_{kj}|$$

is computationally even cheaper.

A generalized Euclidean distance is the Mahalanobis metric (MAHA36)

$$D^2(X_i, X_j) = (X_i - X_j)^T C^{-1} (X_i - X_j).$$

The matrix $C^{-1}$ is usually the inverse of the covariance matrix of a class of data points. The Mahalanobis distance has a very useful property: it is invariant under any non-singular linear transformation. Thus it fits well to the Yang variables and their generalizations which are only quasi-permutationally invariant, that is up to linear transformations. In chapter 4.2.2 a cluster algorithm is described which uses this metric together with the Yang variables.

For a general approach to the problem of similarity and dissimilarity of groups or clusters one should consult (DURA74). In practice the similarity measure or clustering criterion will be chosen within the context of the given problem.

The number of possible subdivisions of real data sets is astronomically high. Therefore it is impossible to find the best among all possible partitions for a given clustering criterion. Consequently, one also needs an algorithm how to efficiently reach an approximately optimum solution of the clustering problem.

Some clustering criteria and cluster procedures that have found application in high energy physics will be presented in chapter 4.
Chapter 3

Some General Methods of Multidimensional Data Analysis

Most experiments in high energy physics (but not only there) lead to multidimensional data. The methods to analyze such data are much less developed than for the one-dimensional case. In some cases one has a model attempting to describe the process from which the data derive. Then the methods of parametric density estimation as maximum likelihood or moments can be used. If one is not so lucky to have a model for the data one normally will check if all the dimensions are really needed to describe the data, i.e. one will look at the intrinsic dimensionality of the data. For this the principal component analysis can be used in the linear case. In the nonlinear case the generalized principal component analysis or successive application of the principal component analysis (FRIE76, FUKU71) can be used.

In this section we shall describe some methods for the exploratory data analysis which have proven to be of broad applicability.

3.1 Projection Pursuit

Projection pursuit (FRIE74) is a mapping technique which searches for the one- or two-dimensional projection exhibiting as much structure of the data as possible. First the one dimensional case is discussed.

Let $\vec{X}_i$ ($i = 1 \ldots N$) be the data set. Then the aim is to search for that direction $\vec{a}$ with $|\vec{a}| = 1$ for which the so called projection index $I(\vec{a})$ is maximized. In (FRIE74) the following construction of $I(\vec{a})$ is proposed:

$$I(\vec{a}) = S(\vec{a})d(\vec{a})$$

with $S(\vec{a})$ measuring the spread of data (trimmed standard deviation)

$$S(\vec{a}) = \left[ \sum_{i=pN}^{(1-p)N} \frac{(\vec{X}_i \vec{a} - \vec{X}_a)^2}{(1 - 2p)N} \right]^{\frac{1}{2}}$$

$$\vec{X}_a = \sum_{i=pN}^{(1-p)N} \frac{\vec{X}_i \vec{a}}{(1 - 2p)N}$$
Supposing the $\vec{X}_i$ are ordered on their projected values ($\vec{X}_i\vec{a}$) and with $d(\vec{a})$ being an average nearness function of the form

$$d(\vec{a}) = \sum_{i=1}^{N} \sum_{j=1}^{N} f(r_{ij}) \Theta(R - r_{ij})$$

$$r_{ij} = |(\vec{X}_i - \vec{X}_j)\vec{a}|$$

$$\Theta(\eta) = \begin{cases} 1 & \text{if } \eta > 0 \\ 0 & \text{if } \eta \leq 0 \end{cases}$$

The function $f(r)$ should be monotonically decreasing in $[0, R]$. The algorithm is insensitive to the special form of $f(r)$ but

$$\bar{r} = \frac{\int_0^R r f(r) \, dr}{\int_0^R f(r) \, dr}$$

determines the size of the structure which is searched for. Finding the maximum of $I(\vec{a})$ is a nonlinear problem which can be solved by standard procedures.

If one aims to find two–dimensional projections the generalization is straightforward: The projection is given by the two orthogonal directions $\vec{a}, \vec{b}$ ($\vec{a} \cdot \vec{b} = 0$)

$$S(\vec{a}, \vec{b}) = S(\vec{a}) S(\vec{b})$$

$$r_{ij} = \left[ (\vec{X}_i - \vec{X}_j) \cdot \vec{a} \right]^2 + \left[ (\vec{X}_i - \vec{X}_j) \cdot \vec{b} \right]^2$$

$$\bar{r} = \frac{\int_0^R r f(r) \, dr}{\int_0^R f(r) \, dr}$$

### 3.2 Nonlinear Mapping

The idea of mapping is to construct a low dimensional (normally two–dimensional) map of the data points preserving approximately the interpoint distances as they are in the original data set. Due to the extraordinary human gift for pattern recognition the investigator can very competently detect the structure of the data, find clusters etc.

Let $\vec{X}_i$ ($i = 1 \ldots N$) be the original points which are mapped to $\vec{Y}_i$ and

$$D_{ij} = |\vec{Y}_i - \vec{Y}_j|$$

$$d_{ij} = |\vec{X}_i - \vec{X}_j|$$

Then the mapping algorithm of Sammon (Samm69) minimizes the error function

$$E(\vec{Y}_1 \ldots \vec{Y}_N) = \frac{1}{N} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \left( \frac{D_{ij} - d_{ij}}{d_{ij}} \right)^2$$

A modification of this algorithm is due to Manton (Mant76). The main disadvantage of these algorithms is the high computational load. So they are applicable only to a few hundred data points.
A much simpler but still powerful technique is the sequential nonlinear mapping (LEE74). It starts from the fact that if $Y_i$ and $Y_j$ are chosen to preserve the original distance

$$D_{ij} = d_{ij}$$

it is always possible to place a third $Y_k$ such as to preserve the distances to the points $i, j$:

$$D_{ik} = d_{ik}, \quad D_{jk} = d_{jk}$$

(triangulation). Thus at least $2N - 3$ interpoint distances of the $N(N - 1)/2$ total number of interpoint distances can be preserved. Since the edge lengths of the MST are known to carry much information about the structure of the point set it is only natural to use the $N - 1$ edge lengths of the MST as part of the distances to be preserved. Then still $N - 2$ interpoint distances can be preserved. For this purpose Lee suggested (LEE74):

- Preserve the distance of all points to a fixed reference point or
- Preserve the distances of each point to its nearest point already matched or
- Preserve the distance to its farthest point already matched.

As with the other nonlinear mappings the disadvantage is that the resulting transformation cannot be summarized with a few numbers and that adding a new data point demands complete recomputation.

### 3.3 Multidimensional Two Sample Test

A very interesting approach to the problem of two sample test for multidimensional data is due to Friedman and Rafsky (FRIE78). They generalize the Wald–Wolfowitz and the Smirnov tests. The Wald–Wolfowitz as well as the Smirnov test in the one-dimensional case start by sorting the pooled data in ascending order without regard to sample identity. Now the essential idea is to use the MST as a generalization of the sorting in the one-dimensional case. This is justified by two important properties of the MST:

1. The $N$ nodes are connected by $N - 1$ edges.
2. The edges connect closely lying points.

The Wald–Wolfowitz test is then generalized as follows. The two samples of size $m$ and $n$ respectively are merged to give the pooled sample of size $N = m + n$, for which the MST is constructed. The number of runs $R$ is defined by

$$R = \sum_{edges} Z_i + 1$$

where

$$Z_i = \begin{cases} 1 & \text{if the edge links nodes from different samples} \\ 0 & \text{otherwise} \end{cases}$$
It can be shown that under the null hypothesis (both samples drawn from the same distribution) and for large sample sizes the quantity $W$ has a standard normal distribution:

$$W = \frac{R - E\{R\}}{\sqrt{\sigma^2\{R\}}}$$

$$E\{R\} = \frac{2mn}{N} + 1$$

$$\sigma^2\{R\} = \frac{2mn}{N(N-1)} \frac{2mn - N}{N} + \frac{C - N + 2}{(N-2)(N-3)}[N(N-1) - 4mn + 2]$$

$C$ is the number of edge pairs sharing a common mode. It should be noted that a multidimensional two sample test can also be used to test factorization (FRIE73):

The hypothesis to be tested is

$$f(x_1, x_2, \ldots, x_d) = f_1(x_1, \ldots, x_i) \times f(x_{i+1}, \ldots, x_d)$$

For this one compares the original sample with a sample derived from the original one by randomly exchanging the components $x_{i+1}, \ldots, x_d$ within the point set.
Chapter 4

Cluster methods

In exploratory data analysis one often is confronted with the following problem: given a set of data one has to find out if there are groups of data such that members of a given group are ‘similar’ and different groups are ‘dissimilar’. Obviously the meaning of the terms ‘similar’ and ’dissimilar’ depends on the context in which the data analysis is performed.

Finding groups with the above stated properties has the following consequences:

• It shows that the data points build clusters, i.e. the data points do not occupy the available space randomly but are concentrated in some regions so that the clusters are distinguishable from each other.

• A data reduction results since the data set can be described by the features of the clusters.

• The investigator might try to interpret the resulting clusters, i.e. to explain the structure of the data in the context of the underlying process. The knowledge obtained this way can even be used to enter an iterative procedure of clustering (e. g. cutting some data, averaging or splitting some clusters, choosing another clustering criterion etc.)

There is a great variety of algorithms for finding clusters in a given data set. One class (hierarchical clustering methods) does not aim to find a unique solution but rather results in a set of solutions of different levels such that two clusters of a lower level belong to one cluster at some higher level. Applications of hierarchical clustering algorithms in high energy physics are discussed in section 4.1. Section 4.2 deals with non hierarchical clustering algorithms.

4.1 Hierarchical Methods

Hierarchical clustering methods applied to \( n \) data points result in \((n - 1)\) possible groupings of points normally represented as a tree as illustrated in fig. 4.1. At the lowest level each data point is a group for its own while at the highest level all the data set is united in one group. The tree could be constructed either by successive merging of groups starting at the lowest level or successive splitting of groups starting
at the highest level. In either case some measure is necessary to decide which groups 
are to be combined or which group has to be split.

One also needs some criterion to decide at which level the tree has to be cut to 
obtain the real clustering since obviously not all the possible groupings depicted by 
the tree are of real interest.

We shall now describe in some detail the application of a splitting algorithm 
(4.1.1) as well as a merging algorithm (4.1.2).

4.1.1 Hierarchical Clustering using the MST

Minimal spanning trees (MST, see 3.) can be used for finding clusters (ZAHN71).

Cutting one edge of the MST one is left with two unconnected parts. Proceeding 
this way one derives a hierarchy of groupings like in fig. 4.1. The first application of 
this technique to high energy data was by Schotanus (SCHO78). The data consisted 
of 1000 events of the final state $\pi^+p \rightarrow \pi^+\pi^0p$ at 5 GeV/c. The distance used in the 
construction of the MST was the Euclidean distance in the space of

$$m_{\pi^+p}^2, \quad m_{\pi^+\pi^0}^2, \quad \omega$$

The MST was cut at its ‘inconsistent edge’, i.e. at the edge with the greatest ratio 
edge length/averages of upward and downward neighbourhood, since this corresponds 
to a density minimum.

Studying the resulting grouping it was concluded that edges in the overlap regions 
are rather shorter than in neighbouring regions so that different clusters could not 
be separated one from another.

To achieve an improvement it was necessary to modify the criterion by taking 
into account additional information about the shape of the neighbourhood. For this 
the path collinearity was introduced. Path collinearity is a directional criterion which 
can be defined for any defined path through the MST — advantageously one uses a 
maximal path. The path collinearity $\Theta$ at a certain point $p$ with a leverage $l_p$ is the
angle between the straight lines connecting $p$ with the points $l_p$ places upward and downward the path respectively. Thus $\Theta$ measures if the data have linear pieces ($\Theta$ being $\pi$) or sharp bends. By means of the collinearity criterion the $\pi^+p \to \pi^+\pi^0p$ data at $5 GeV/c$ could be separated into

a) a backward production region (baryon exchange)

b) $\Delta^{++}$ resonance production

c) $\rho^+$ resonance production

d) diffraction dissociation and $\Delta^+$ production

e) elastic contamination of the reaction $\pi^+p \to \pi^+p$.

The collinearity criterion is applicable if the data is essentially one–dimensional. It could be extended to higher dimensions defining coplanarity etc. but it was suggested (SCHO76) that for this type of extensions it would be better to analyze the full covariance matrix of the neighbourhood.

4.1.2 Selection of Jets by Hierarchical Clustering

High energy quarks and gluons produced in storage ring collisions manifest themselves as hadronic jets observed in the final state. As the total energy increases QCD unambiguously predicts an increase of the jet multiplicity. Thus it becomes an important task to develop methods to recognize events of any jet multiplicity.

The most obvious feature of jets is the strong angular correlation, i.e. the appearance of a narrow cone built by the momenta of the particles belonging to a given jet. This quite naturally leads to the use of clustering algorithms for the study of jet events. For this the distance $d_{ik} = d(\vec{p}_i, \vec{p}_k)$ between the two particles $i$ and $k$ is defined such that small angles result in short distances. The first application of this idea is in (LANI81). For $d_{ik}$ they used:

$$d_{ik} = \frac{1}{2} (\frac{\vec{p}_i \cdot \vec{p}_k}{|\vec{p}_i||\vec{p}_k|} + 1)$$

The algorithm then combines $(N-1)$ times the two most similar (smallest $d_{ik}$) groups. For this it is necessary to state what the similarity of a combined group to the old ones is. If groups (= particles at the beginning) $i$ and $k$ have been combined to give a group called $m$ the similarity $d_{ml}$ of the new group $m$ to the remaining groups is chosen to be

$$d_{ml} = \min(d_{il}, d_{kl}) \quad \text{with} \quad l \neq i \quad \text{and} \quad l \neq k.$$

Using this definition one arrives at the ‘complete linkage’.

The cluster algorithm labels the particles according to their membership to a given group. But it does not answer the question of how many jets are in the event, i.e. at which level the tree has to be cut to yield the real clustering. Thus for each event one has to make all the possible hypotheses about the jet multiplicity and to decide which is the most probable one. In (LANI81) a straightforward generalization of the triplicity (BRAN79) was proposed that can be used in the decision procedure. It was concluded that
• the sketched clustering algorithm is well suited to find jets in multihadron final states
• it is applicable to higher jet multiplicities
• the particles are classified according to their membership to the jets.

Another approach due to Dorfan (DORF80) uses

\[ d_{ik} = \frac{\Theta_{ik}^2}{|\vec{p}_i||\vec{p}_k|} \]

\( \Theta_{ik} \) being the angle between the momenta \( \vec{p}_i \) and \( \vec{p}_k \).

Starting from this distance measure the MST (see 3.) is constructed and inconsistent edges are cut if they are larger than \((R_l \times \text{median of the edge lengths})\). A detailed Monte Carlo study again leads to the conclusion that hadronic jets are reliably reproduced by the found clusters which thereby allow for a meaningful study of the jet features.

4.2 Non–hierarchical Techniques

Non–hierarchical clustering techniques do not rely on a hierarchy of subsequent partitions of the data sample. They rather create in each iteration step a new assignment of the data points to the clusters according to some clustering criterion. We shall now describe in some detail three non–hierarchical clustering procedures, which all have been applied to high energy data:

• the non–parametric valley–seeking technique of Koontz and Fukunaga (KOON72)

• the cluster algorithm CLUCOV of Nowak and Schiller (NOWA75) and

• the interactive clustering technique of Gelsema (GELS74).

4.2.1 The Valley–Seeking Technique

We now describe the valley seeking technique of Koontz and Fukunaga (KOON72) in a version designed for the analysis of many–particle final states in high energy physics.

Results of the application of this technique to the reactions

\[ \pi^+ p \to p\pi^+\pi^+\pi^- \quad \text{at 8 and 16 GeV/c} \]

and \[ \pi^+ p \to p\pi^+\pi^+\pi^-\pi^- \quad \text{at 16 GeV/c} \]

can be found in (BOET74).

The construction of the algorithm starts from the loss of information \( J \) which arises if one replaces \( N \) given data vectors \( [X_1, \ldots, X_N] \) by labels or cluster numbers \( [w_1, \ldots, w_N] \):

\[ J = \sum_{i=1}^{N} \sum_{j=1}^{N} f(X_i, X_j) |d_X(X_i, X_j) - d_\omega(w_i, w_j)|^2 \]
Here, \( d_X(X_i, X_j) \) denotes the distance between two vectors \( X_i \) and \( X_j \), \( d_\omega \) is an appropriately defined metric for the distance between two classes or clusters and \( f(X_i, X_j) \) are weighting factors. The labels \( w_i \) can be integers from 1 to \( M \) (\( M < N \)) and denote the class to which \( X_i \) is assigned. The task of finding a meaningful partition of the data sample is now reformulated: one searches for the partition with minimal information loss \( J \).

We start with the assumptions

\[
    d_\omega(w_i, w_j) = \begin{cases} 
        D, & D > 0 \text{ for } w_i \neq w_j \\
        0, & \text{ for } w_i = w_j 
    \end{cases}
\]

\[
    f_X(X_i, X_j) = \begin{cases} 
        1, & \text{ if } d_X(X_i, X_j) < R \\
        0, & \text{ if } d_X(X_i, X_j) > R > 0 
    \end{cases}
\]

\[
    \equiv f_R[d_X(X_i, X_j)]
\]

Using the symmetry of \( f_R \) with respect to \( X_i \) and \( X_j \) and \( d_\omega(w_i, w_i) = 0 \) for all \( i \) (a property of any metric) and assuming sufficiently small \( R \) we obtain

\[
    J \sim 2D^2 \sum_{i=1}^{N} \sum_{j=i+1}^{N} f_R[d_X(X_i, X_j)](1 - \delta_{w_i, w_j})
\]

\[
    \equiv 2D^2 J_R
\]

\( J_R \) assigns a nonzero penalty for each pair of vectors closer together than \( R \) and classified into different classes. Hence the main contributions to \( J_R \) come from points near the boundary between two clusters.

Minimization of \( J_R \) consequently enforces the boundaries between the clusters to be defined across a region of minimum density of data points. The name valley seeking technique originates from this property.

This clustering criterion has the following advantages:

1. Computation: For a given classification, \( J_R \) is determined by counting rather than by difficult calculations.

2. Storage: The storage requirement is mainly governed by the number of pairs of vectors that are closer together than \( R \). This number can be kept small by choosing \( R \) sufficiently small.

3. The valley–seeking property makes the clustering criterion suitable for non–supervised classification.

Its disadvantages are:

1. Very distant clusters can receive the same label.

2. No account is taken of the inner structure of the clusters.

3. The cluster shape does not enter the cluster criterion.

As can be seen from fig. 4.2, pronounced V–shaped structures occur in practice which should be split at the edges.

Having defined a clustering criterion it remains to choose an algorithm of how to efficiently achieve an optimum classification. The minimization of the information loss \( J_R \) is performed by the following algorithm:
1. Choose an initial assignment of the $N$ points to $M$ classes.

2. For every point $i$ count the number of points belonging to a given class within a certain distance $R$ of $X_i$.

3. The point $i$ is assigned to the class having the maximum number of points within $R$.

4. If any point is placed in a new class, return to step 2. Otherwise, stop.

$M$ and $R$ have to be determined empirically.

For an application of the valley seeking technique to the reaction $\pi^+p \rightarrow p\pi^+\pi^+\pi^-$ the following variables were chosen:

$$M(p\pi_f^+), \ M(p\pi_s^+), \ M(p\pi^-), \ M(\pi_f^+\pi^-), \ M(\pi_s^+\pi^-)$$

$M$ denotes the invariant mass of the two particles in the bracket, the index $f/s$ stands for the $\pi^+$ with the greater and smaller longitudinal momentum.

As resonance production plays a decisive role in this reaction, this set of variables should permit to extract a significant part of the information contained in the data.

Inclusion of the four–momentum transfer variables $t(p/p)$ and $t(\pi^+/\pi^-)$ leads to a set of seven variables which completely describe this four–particle final state.

Other sets of independent invariants can be derived from these variables by linear transformations.

As linear transformations do not change the results of the analysis, this choice of variables guarantees a high degree of generality of the procedure.

The number $N$ of points $X_i$ led to storage problems in this application. Therefore the algorithm was simplified as follows.

$N_A$ arbitrarily chosen points were submitted to the algorithm described above. The remaining $N_B = N - N_A$ points were assigned to the existing clusters according to step 2 and 3 of the original algorithm. This improved the statistical significance of the existing clusters.

In this application an Euclidean metric was chosen for $d_X(X_i - X_j)$. 

Figure 4.2: $\pi^+ p \rightarrow p\pi^+\pi^0$ ($p_{lab} = 3.9 GeV/c$) a.) Prism plot for Monte–Carlo events (Lorentz–invariant phase space) b) Prism plot for experimental data
The distance parameter $R$ was chosen to be $R = 0.455 GeV$ with $M = 15$ initial clusters. The following methodic results were obtained:

1. There exist clusters in phase space.
2. They correspond to dynamical mechanisms.
3. Several production mechanism can contribute to one cluster.
4. Some mechanisms are well separated.

For detailed physical results see (B0ET74).

The inclusion of the four–momentum transfers did not lead to a cleaner separation. Possibly these variables do not contain additional information, or this information could not be extracted because of statistical limitations.

4.2.2 The Cluster Algorithm CLUCOV

Analyzing three and four particle hadronic final states it was found (BRAU71) that clusters are generally ellipsoids with arbitrary orientation in phase space. This is also illustrated in fig. 4.2. The cluster algorithm CLUCOV was specially designed to meet this situation. A detailed description of the algorithm can be found in (NOWA75).

In the cluster algorithm CLUCOV the $k$–th cluster $G^k$ is therefore characterized by the moments of order zero, one and two of the distribution of the $N$ points $X^m = (X^m_1, \ldots, X^m_{3n-5})$ contained in this cluster. Let these points have their experimental weights $w^m$. Then the three moments are

\[ I^k = \sum_{m=1}^{N_k} w^m \]

– the number $I^k$ of points $X^m$ contained in the cluster $G^k$

\[ Q^k_i = \frac{1}{I_k} \sum_{m=1}^{N_k} w^m X^m_i \quad \text{with } X^m \in G^k \]

– the centroid $Q^k$ of the cluster $G^k$

\[ C^k_{ij} = \frac{1}{I_k} \sum_{m=1}^{N_k} w^m (X^m_i - Q^k_i)(X^m_j - Q^k_j) \quad \text{with } X^m \in G^k \]

– the covariance matrix $C^k$ of the cluster $G^k$

The eigenvectors of the covariance matrix point into the direction of the main axes of the ellipsoids by which the shape of the clusters is approximated. The eigenvalues of the covariance matrix denote the lengths of the main axes, and the determinant measures the volume of the clusters.

All these three moments enter the definition of the distance of a point $m$ at $X^m$ from the $k$–th cluster $G^k$.

The number $I^k$ of points in cluster $k$ is included into the distance measure as a linear weight factor, so that big clusters ‘attract’ further points.

The Euclidean distance $(X^m - Q^k)$ of a given point $X^m$ from the centroid $Q^k$ of the $k$–th cluster enters the exponent of a gaussian containing also the covariance
matrix $C^k$:

$$f^k_m = \frac{I^k}{\sqrt{(2\pi)^{3n-5}|C^k|}} \exp[-\frac{1}{2}(X^m - Q^k)(C^k)^{-1}(X^m - Q^k)]$$

Thus, each cluster builds its 'own' metric.

In the direction of the main axes distances are measured in units of the corresponding eigenvalue of the covariance matrix (which is suggested by the quadratic form in the exponent of the gaussian).

The determinant in the denominator of $f^k_m$ favours compact clusters against voluminous clusters of the same content.

This distance measure is also invariant under linear transformations such as translation and rotation and — when applying the Yang variables — also against permutation of the final state particles.

We now describe an iteration step of the algorithm. A starting procedure will be given later.

1. Calculate for all points $X^m$ the distance measure $f^k_m$ with respect to all existing clusters $G^k$.

2. Assign the point $X^m$ to the cluster $G^l$ with the biggest $f^l_m$.

3. If $f^k_m$ is for all clusters lower than a certain limit (which is a parameter of the algorithm) assign $X^m$ to a 'garbage' cluster.

4. Update $I^k$, $Q^k$ and $C^k$ for all clusters. Goto 1.

In the algorithm CLUCOV it is also possible to split and merge clusters, that is to change the number of clusters.

To achieve this, a measure $t$ for the compactness of two clusters $G^k$ and $G^l$ is defined:

$$t = \frac{h_0}{\sqrt{h_k h_l}}$$

The quantities $h_k$ and $h_l$ are superpositions of the gaussians $f^k(X)$ and $f^l(X)$ of the clusters $G^k$ and $G^l$ in their centroids $Q^k$ and $Q^l$:

$$h_k = f^k(Q^k) + f^l(Q^k) \quad \text{and} \quad h_l = f^k(Q^l) + f^l(Q^l)$$

The quantity $h_0$ is the minimum of this superposition along the distance vector $(Q^k - Q^l)$ between the clusters:

$$h_0 = \min[f^k(X) + f^l(X)]$$

If for a pair of groups this measure $t$ exceeds a limit $t_{merge}$ being a parameter of the merging procedure this pair of groups is united into one group.

The compactness of the clusters is tested by arbitrarily subdividing the clusters with hyperplanes through the cluster centroids. If the relative compactness of the two parts of the clusters is smaller than $t_{split}$ being a parameter of the splitting procedure, this cluster is split by the corresponding hyperplane.
Figure 4.3: Test results obtained with the cluster algorithm CLUCOV applied to two-dimensional data

Other measures of distance and compactness are possible within the cluster algorithm CLUCOV and can be easily implemented.

Finally we describe a starting procedure. The contents of all clusters are set to one, and all covariance matrices are set to the unit matrix. To find a starting set of cluster centroids it is demanded that no single point has an Euclidean distance of more than \( R \) (a parameter of the starting procedure) from at least one cluster center. This is achieved by the following procedure:

1. Choose an arbitrary point as the first center.
2. Decide for each following point if its distance to all existing cluster centers is greater than \( R \). If yes, take this point as a new cluster center.

Fig. 4.3 demonstrates the capacities of this algorithm at some two-dimensional test examples. For applications of this algorithm to many-particle hadronic final states see (HONE79, NAUM79).

4.2.3 Gelsema’s Interactive Clustering Technique

This procedure developed in CERN (GELS74) starts from the following considerations:

Let the probability density distribution \( h(X|b^*) \) of the points \( X \) in phase space with the distribution parameter vector \( b^* \) consist of a mixture of \( M \) distributions
\[ f(X|b'_k) \] with weights \( p_k \)

\[
h(X|b^*) = \sum_{k=1}^{M} p_k f(X|b'_k)
\]

The task of clustering is now to find the distribution parameter vector \( b \) which is the best estimate of \( b^* \) and is consistent with a set of observations from the density \( h(X|b^*) \).

Now define the information function \( \eta(b, b^*) \) as the expectation value of the natural logarithm of the mixture density

\[
\eta(b, b^*) = E[\ln h(X|b)] = \int \ln[h(X|b)]h(X|b^*)dX.
\]

It can be shown (PATR72) that the vector \( b \) maximizing the information function corresponds to the asymptotic minimum-risk solution.

If \( h(X|b^*) \) is a superposition of \( M \) non–overlapping gaussians with relative weights \( p_k \) and covariance matrices \( C_k \), then maximizing \( \eta(b, b^*) \) corresponds to maximizing (PATR72)

\[
\eta_G(b) = \sum_{k=1}^{M} p_k \ln\left( \frac{p_k}{\sqrt{|C_K|}} \right).
\]

As \(|C_k|\) is related to the volume occupied by the category (or cluster) \( k \), maximizing \( \eta_G(b) \) leads to that subdivision of observation space which corresponds to maximum average probability density. A procedure that maximizes \( \eta_G(b) \) will therefore tend to locate clusters in the observation space.

The cluster procedure of Gelsema has the following general properties:

1. The number of clusters is fixed.
2. Initial cluster nuclei have to be defined on the basis of a priori knowledge.
3. Events may be left unclassified. This permits the treatment of clusters superimposed on a background.

The algorithm is an interactive one and works as follows. At the beginning, cluster nuclei have to be defined using some a priori knowledge. For each cluster nucleus \( i \) the fraction \( p_k \) of events in this nucleus and the covariance matrix \( C_k \) are calculated. This gives the starting value of \( \eta_G(b) \) qualifying the initial solution.

In each subsequent iteration step the events are assigned to all of the \( M \) existing clusters in turn and \( \eta_i(b) \) is calculated. No updating of the classes is performed at this stage, but for every event the sequence of improvements

\[
\Delta\eta_i(b) = \eta_i(b) - \eta_{old}(b)
\]

for a tentative assignment to all classes \( i = 1, \ldots, M \) is calculated and histogrammed separately for every class. The maximum improvement \( \Delta\eta_{max} \) and the corresponding
class number are stored. Events which really belong to class $i$ will have larger values of $\Delta \eta_i$.

The interaction between data analyst and program now consists in a visual inspection of these histograms on a display leading to the definition of a threshold value $\tau_i$ of $\Delta \eta_i$ above which events are assigned to class $i$. Now the clusters are updated. An event enters class $i$ if both conditions

$$\Delta \eta_{\text{max}} = \Delta \eta_i \quad \text{and} \quad \Delta \eta_i > \tau_i$$

are satisfied.

An event is omitted from class $i$ if at least one of the conditions

$$\Delta \eta_{\text{max}} \neq \Delta \eta_i \quad \text{or} \quad \Delta \eta_i \leq \tau_i$$

is satisfied.

A table of the number of reassignments to the clusters is displayed. If these numbers get small the procedure becomes stable and can be terminated. Applications of Gelsema’s interactive clustering technique can be found in (BAUB77,VAIS76).

In order to achieve a meaningful separation, in the histograms of $\Delta \eta_i$ the peaks of high values of $\Delta \eta_i$ have to be well separated from the rest of the events. Then, small changes in the cut values $\tau_i$ will not affect the final result. In such a case the procedure can even be run in an unsupervised way.
References

New York, San Francisco London 1973

(BAUB77) M. Baubillier et al., ‘Multidimensional Analysis of the Reaction $\pi^- n \rightarrow \pi^- \pi^+ n$ at 9 GeV/c’ and ‘Multidimensional Analysis of the reaction $\pi^- d \rightarrow \pi^-\pi^+ d$ at 9 GeV/c’, both submitted to Intern. Conf. on High Energy
Physics, Budapest 1977

(BECK76) L. Becker and H. Schiller, Possible Generalization of Yang Variables for
the Study of Many Particle Final States, Berlin preprint PHE 76–23 (1976)


(BRAN79) S. Brandt, H. D. Dahmen, Z. Phys. C1 (1979) 61

(BRAU71) J. E. Brau et al., Phys. Rev. Letts. 27 (1971) 1481

(BYCK73) E. Byckling and K. Kajantie, Particle Kinematics, J. Wiley and Sons
(1973) p. 202

(BYER64) N. Byers and C. N. Yang, Rev. Mod. Phys. 36 (1964) 595

(CERN76) Topical Meeting on Multidimensional Data Analysis, CERN (1976)


(DURA74) B. S. Duran, P. L. Odell, Cluster Analysis, Lecture Notes in Economics
and Mathematical Systems, Springer Verlag 1974

(FRIE73) J. H. Friedman, SLAC–PUB–1358 (1973)

(1974) 881

(FRIE76) J. H. Friedman, CERN/DD/76/23, pp 19


(1971) 176

(GELS74) E. S. Gelsema, Description of an Interactive Clustering Technique and its
Applications, CERN/DD/74/16


(KITT76) W. Kittel, Progress in Multidimensional Analysis of High Energy Data,
IVth International Winter Meeting on Fundamental Physics, Salardu (Spain)
1976


(MANT76) N. Manton, Two non–linear Mapping Algorithms for Use in Multidimensional Data Analysis, Proc. Topical Conf. on Multidimensional Data Analysis, CERN 1976


(NOWA75) W.D. Nowak and H. Schiller, Berlin preprint PHE 75–12 (1975)

(PATR72) E.A.Patrick, Fundamentals of Pattern Recognition, Prentice Hall Inc. 1972


(SCHO78) D.J. Schotanus, Topical Meetings on Multidimensional Data Analysis, CERN 1976 and Nijmegen 1978

(VAIS76) Ch. de la Vaissiere, Application of the CERN Interactive Cluster Analysis to Multibody pp Interactions, Topical Meeting on Multidimensional Data Analysis, CERN 1976

(ZAHN71) C.T. Zahn, IEEE Transactions on Computers C-20 (1971) 68