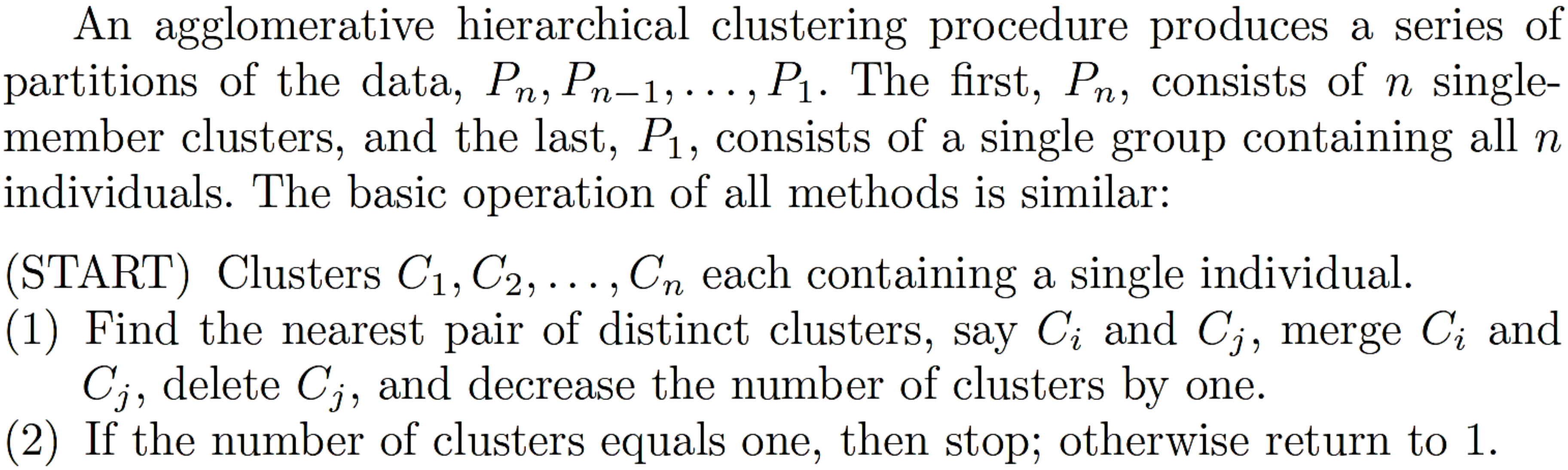
Introduction to Applied Multivariate Analysis with R

Cluster Analysis

Agglomerative Hierarchical Clustering

Produces a *hierarchical classification* of data.

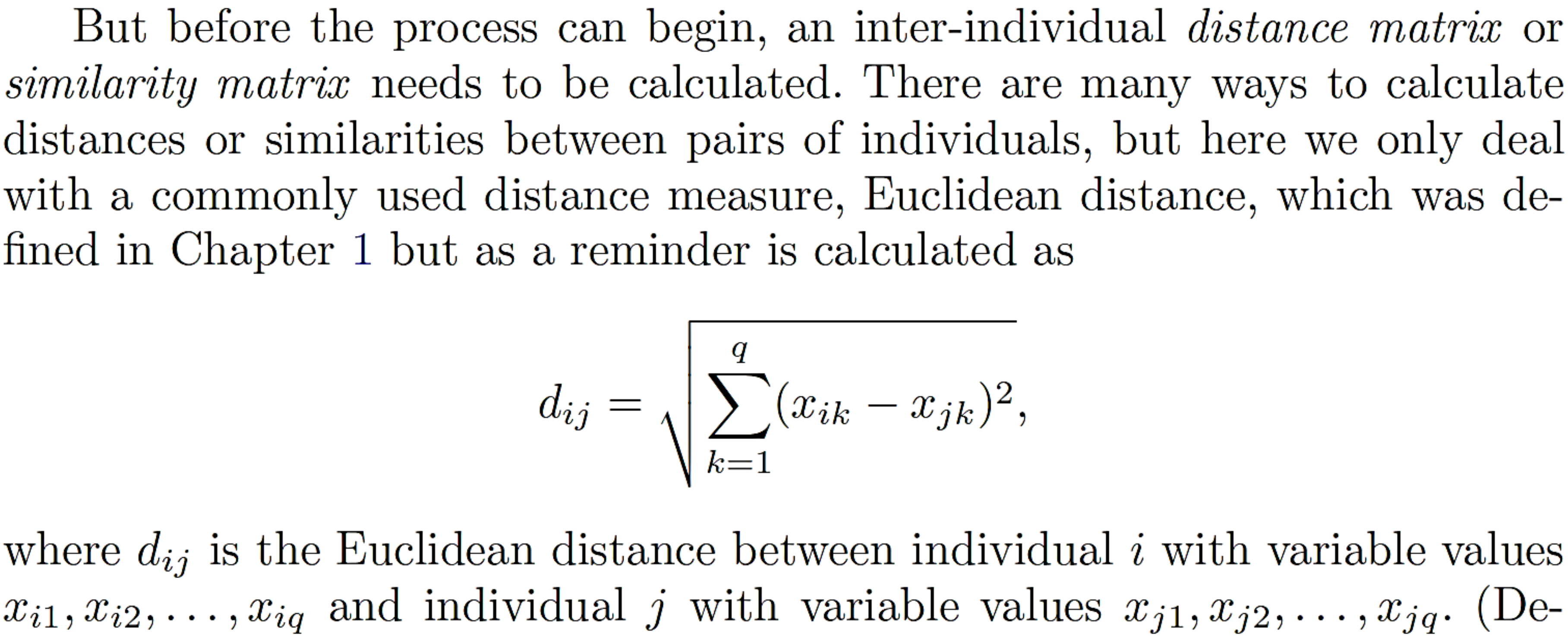


An agglomerative hierarchical clustering procedure produces a series of partitions of the data, $P_n, P_{n-1}, \ldots, P_1$. The first, P_n, consists of n single-member clusters, and the last, P_1, consists of a single group containing all n individuals. The basic operation of all methods is similar:

```
(START) Clusters $C_1, C_2, \ldots, C_n$ each containing a single individual.
(1)  Find the nearest pair of distinct clusters, say $C_i$ and $C_j$, merge $C_i$ and
     $C_j$, delete $C_j$, and decrease the number of clusters by one.
(2)  If the number of clusters equals one, then stop; otherwise return to 1.
```

Agglomerative Hierarchical Clustering

Produces a *hierarchical classification* of data.

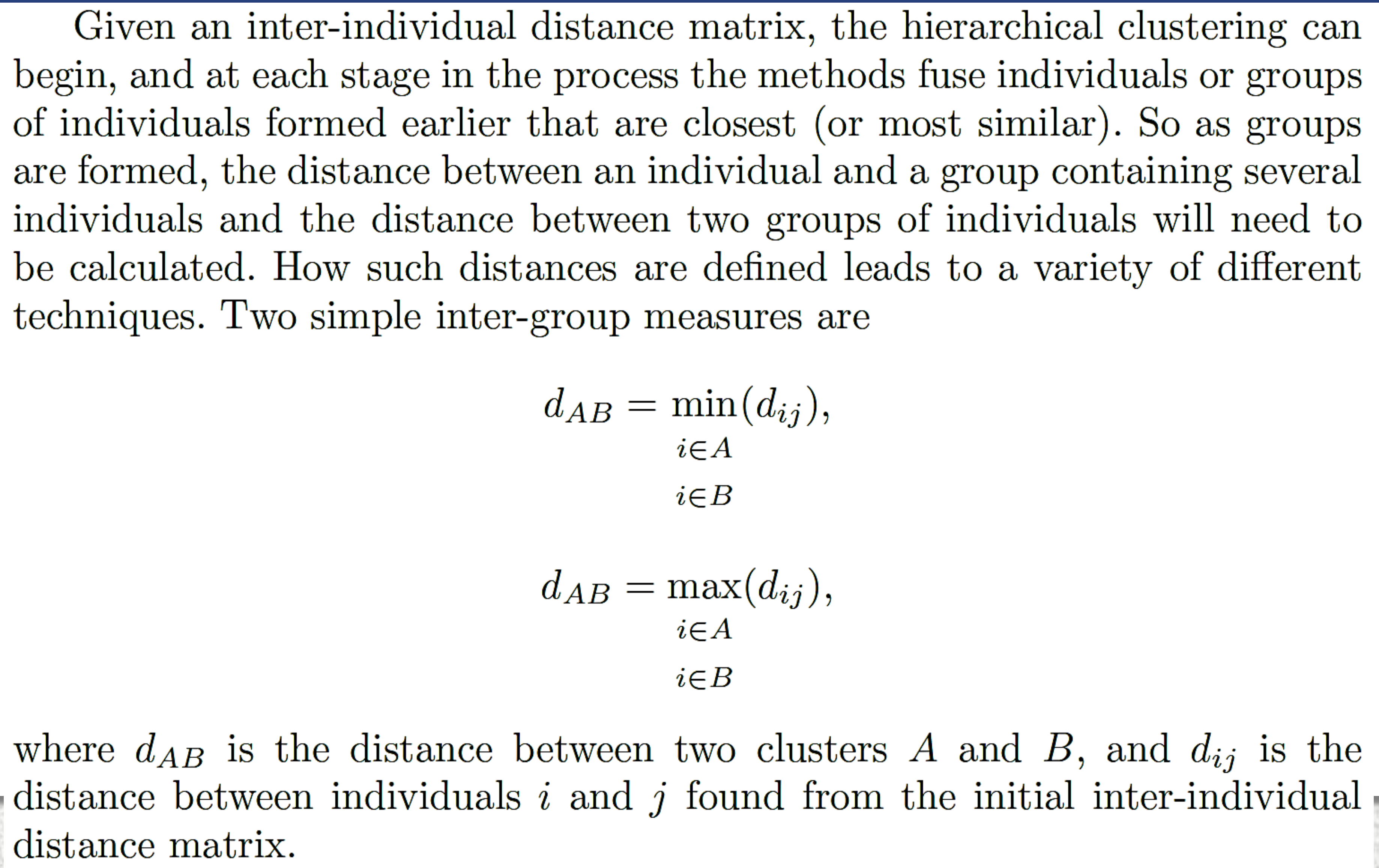


But before the process can begin, an inter-individual *distance matrix* or *similarity matrix* needs to be calculated. There are many ways to calculate distances or similarities between pairs of individuals, but here we only deal with a commonly used distance measure, Euclidean distance, which was defined in Chapter 1 but as a reminder is calculated as

$$d_{ij} = \sqrt{\sum_{k=1}^{q}(x_{ik} - x_{jk})^2},$$

where d_{ij} is the Euclidean distance between individual i with variable values $x_{i1}, x_{i2}, \ldots, x_{iq}$ and individual j with variable values $x_{j1}, x_{j2}, \ldots, x_{jq}$. (De-

Agglomerative Hierarchical Clustering



Given an inter-individual distance matrix, the hierarchical clustering can begin, and at each stage in the process the methods fuse individuals or groups of individuals formed earlier that are closest (or most similar). So as groups are formed, the distance between an individual and a group containing several individuals and the distance between two groups of individuals will need to be calculated. How such distances are defined leads to a variety of different techniques. Two simple inter-group measures are

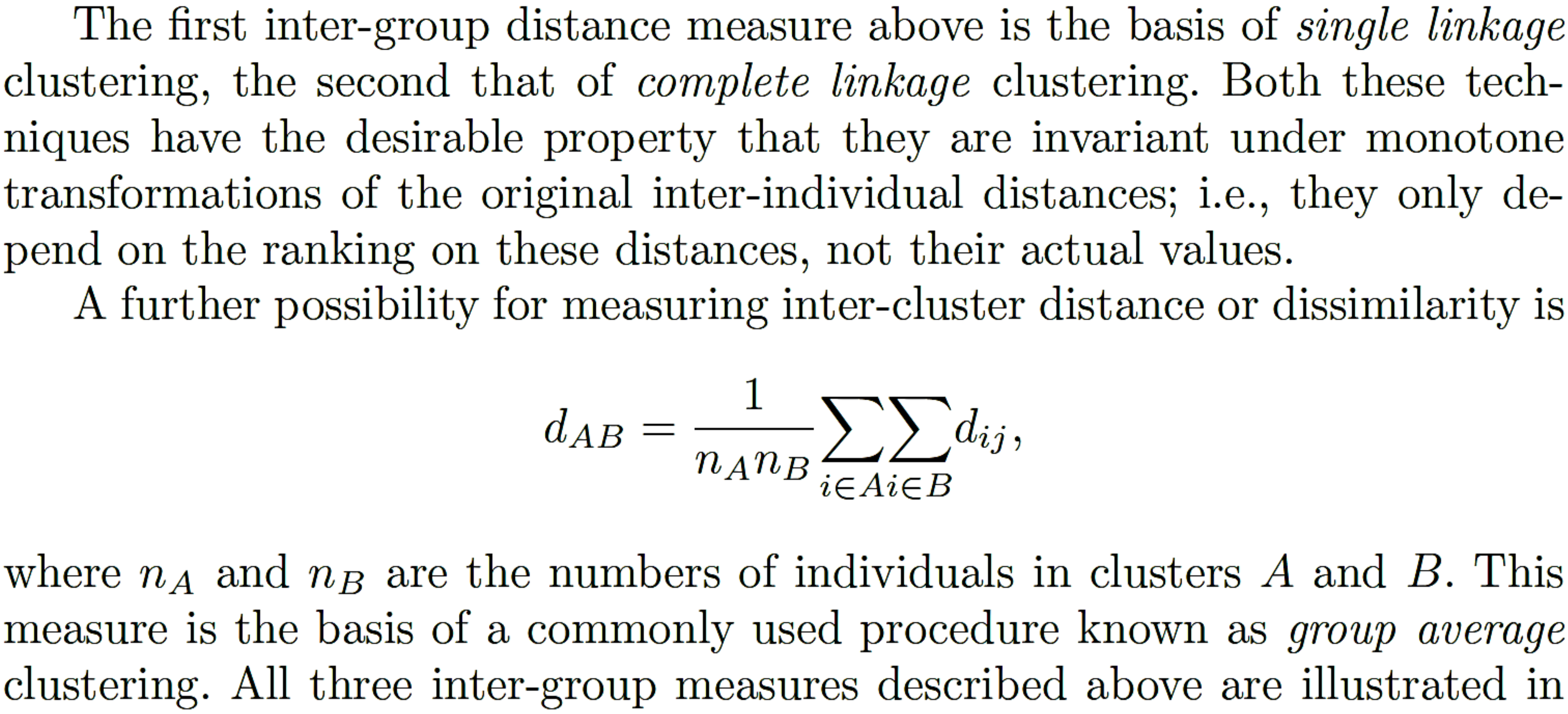
$$d_{AB} = \min_{\substack{i \in A \\ i \in B}} (d_{ij}),$$

$$d_{AB} = \max_{\substack{i \in A \\ i \in B}} (d_{ij}),$$

where d_{AB} is the distance between two clusters A and B, and d_{ij} is the distance between individuals i and j found from the initial inter-individual distance matrix.

Agglomerative Hierarchical Clustering

Produces a *hierarchical classification* of data.



The first inter-group distance measure above is the basis of *single linkage* clustering, the second that of *complete linkage* clustering. Both these techniques have the desirable property that they are invariant under monotone transformations of the original inter-individual distances; i.e., they only depend on the ranking on these distances, not their actual values.

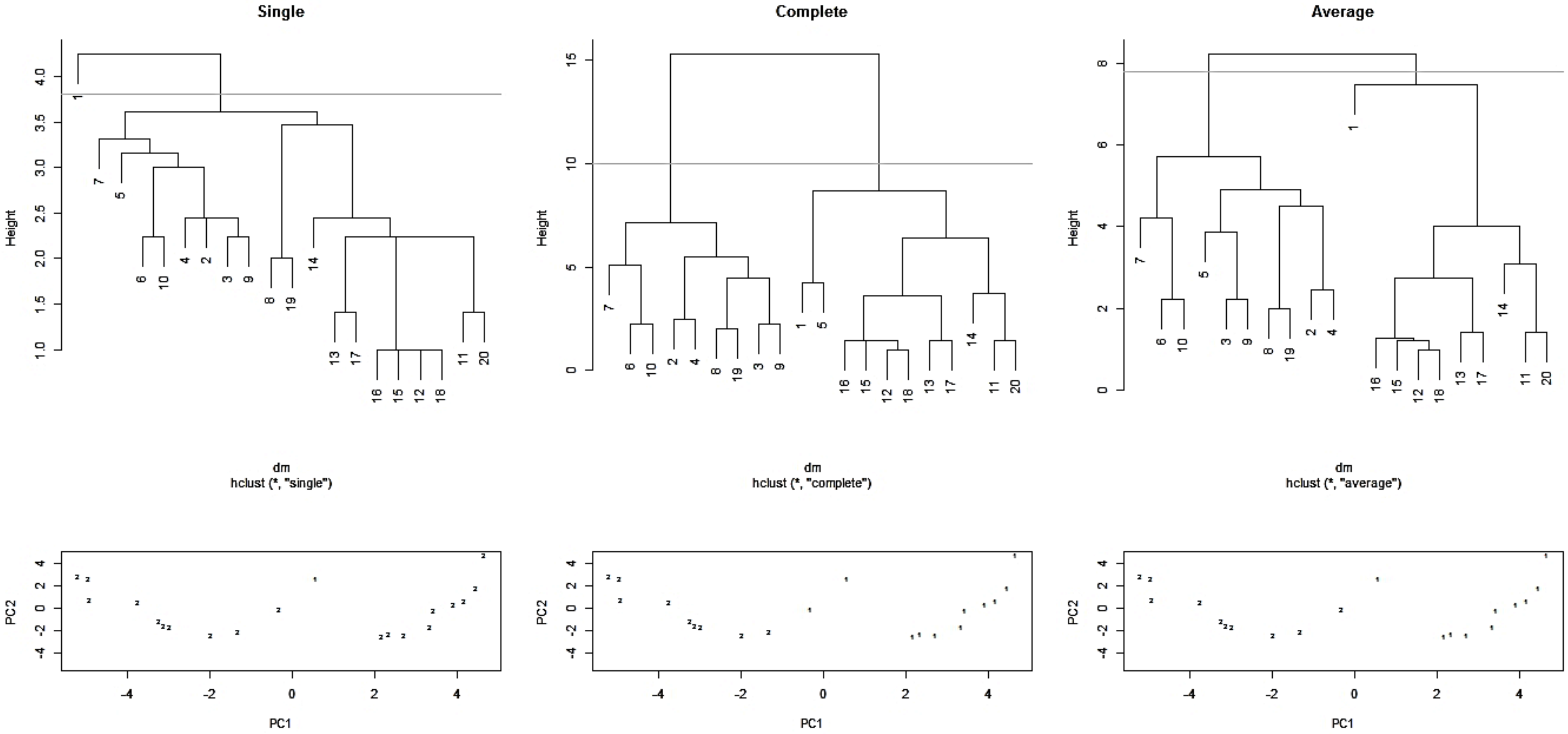
A further possibility for measuring inter-cluster distance or dissimilarity is

$$d_{AB} = \frac{1}{n_A n_B} \sum_{i \in A} \sum_{i \in B} d_{ij},$$

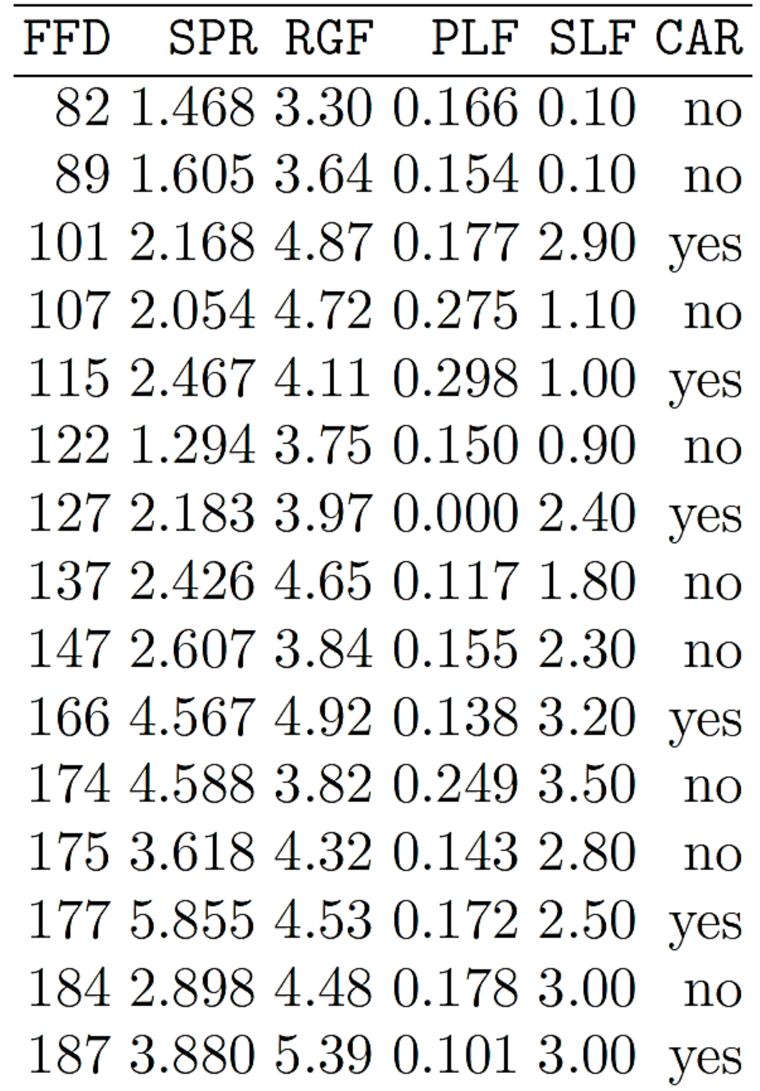
where n_A and n_B are the numbers of individuals in clusters A and B. This measure is the basis of a commonly used procedure known as *group average* clustering. All three inter-group measures described above are illustrated in

Agglomerative Hierarchical Clustering

Cluster Solutions for Measure Data:



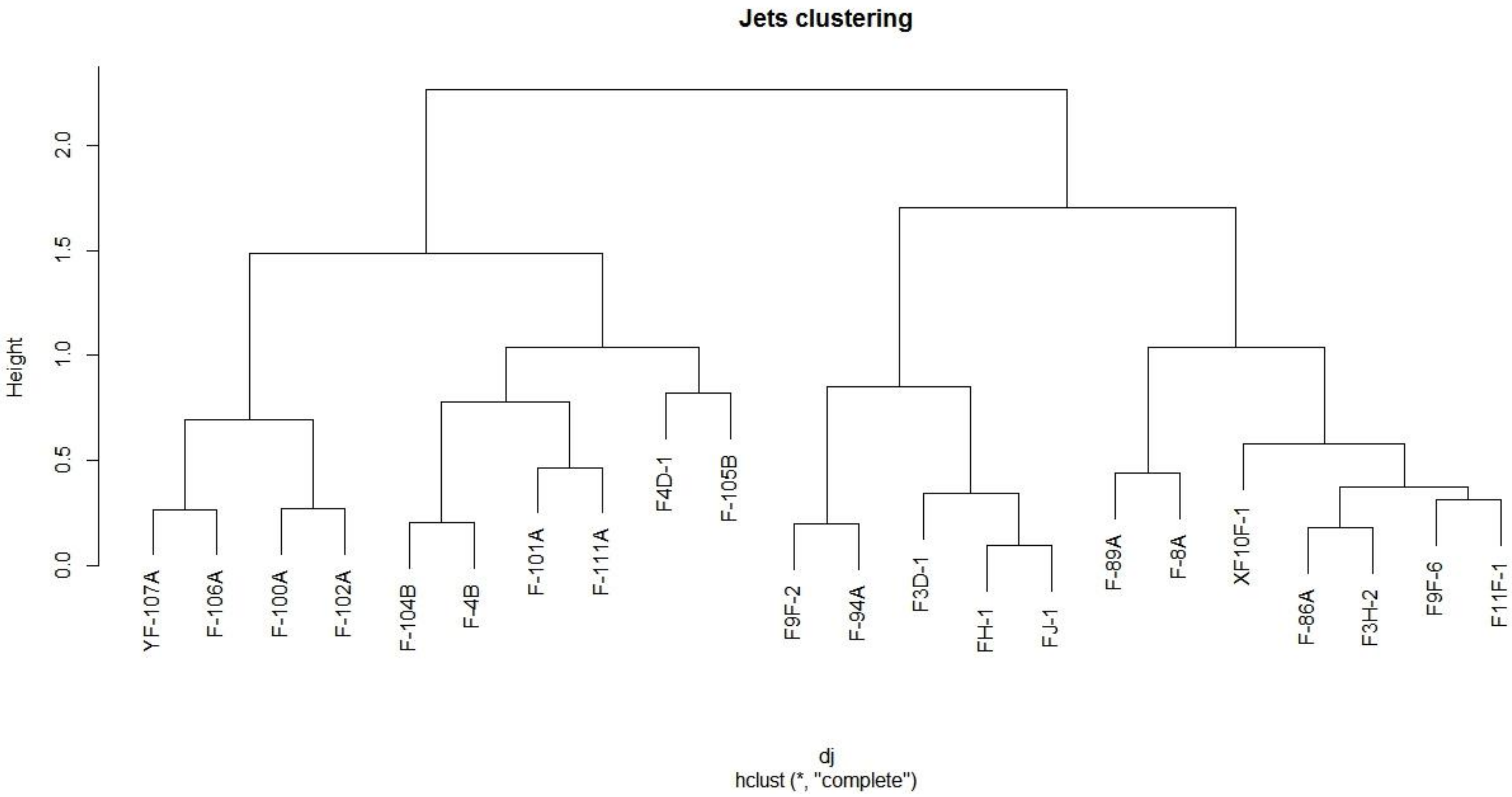
Jet Fighters Data (partial)



FFD	SPR	RGF	PLF	SLF	CAR
82	1.468	3.30	0.166	0.10	no
89	1.605	3.64	0.154	0.10	no
101	2.168	4.87	0.177	2.90	yes
107	2.054	4.72	0.275	1.10	no
115	2.467	4.11	0.298	1.00	yes
122	1.294	3.75	0.150	0.90	no
127	2.183	3.97	0.000	2.40	yes
137	2.426	4.65	0.117	1.80	no
147	2.607	3.84	0.155	2.30	no
166	4.567	4.92	0.138	3.20	yes
174	4.588	3.82	0.249	3.50	no
175	3.618	4.32	0.143	2.80	no
177	5.855	4.53	0.172	2.50	yes
184	2.898	4.48	0.178	3.00	no
187	3.880	5.39	0.101	3.00	yes

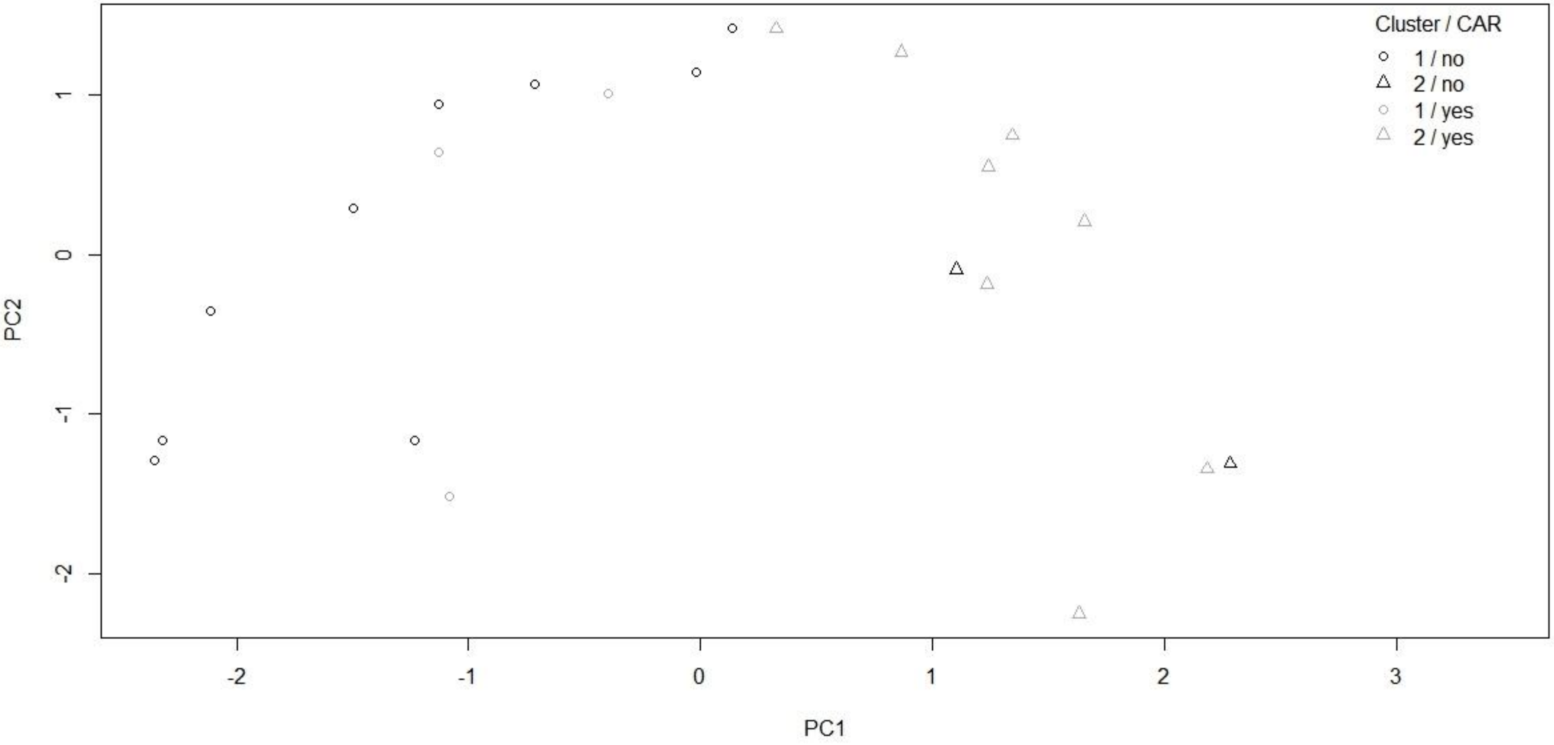
Jet Fighters Data

Hierarchical Clustering (complete linkage) of Jet Data:



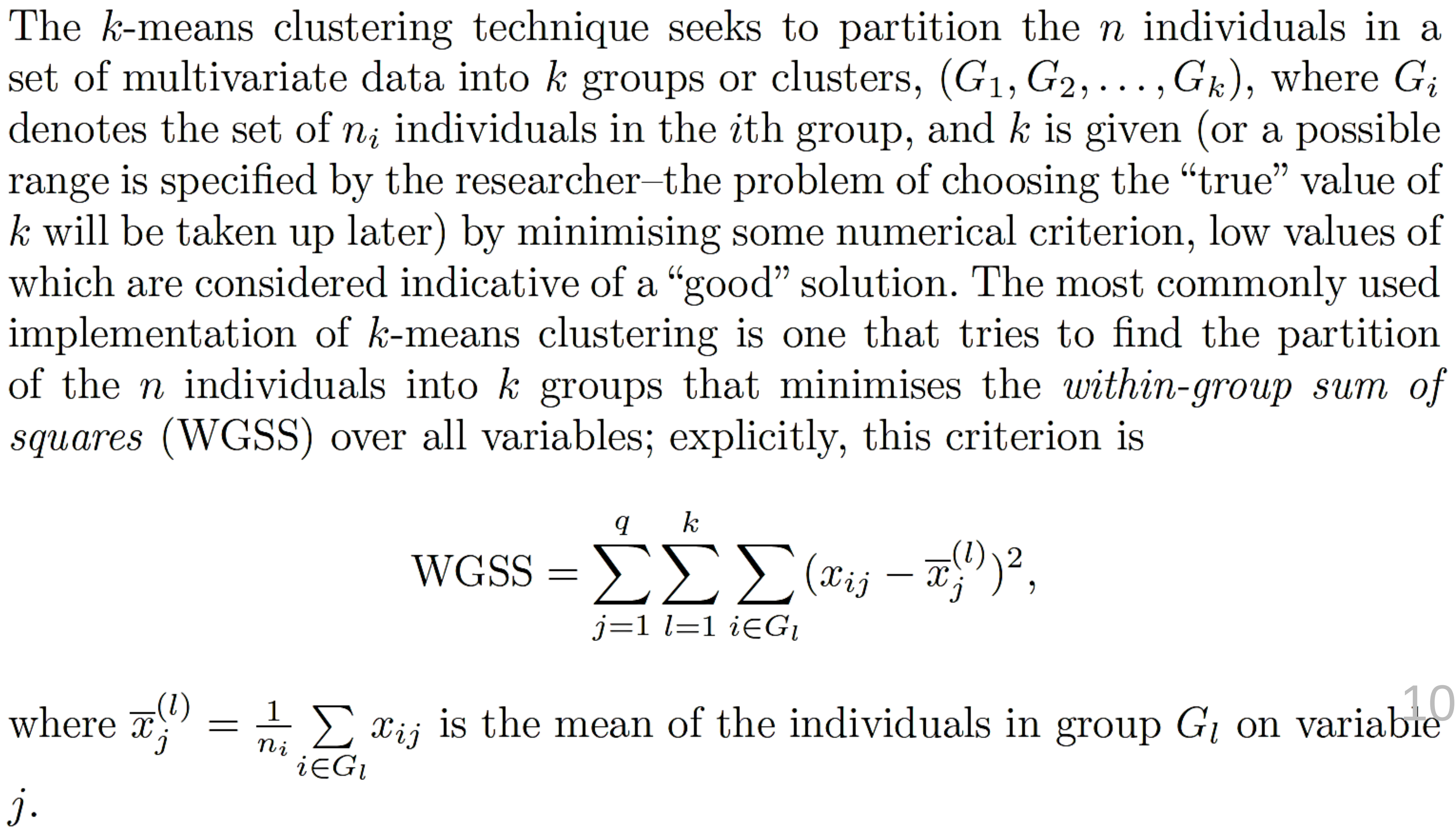
Jet Fighters Data

Hierarchical Clustering (complete linkage) of Jet Data Plotted in PCA Space:



K-Means Clustering

Partitions the *n* individuals in a set of multivariate data into *k* groups or clusters.



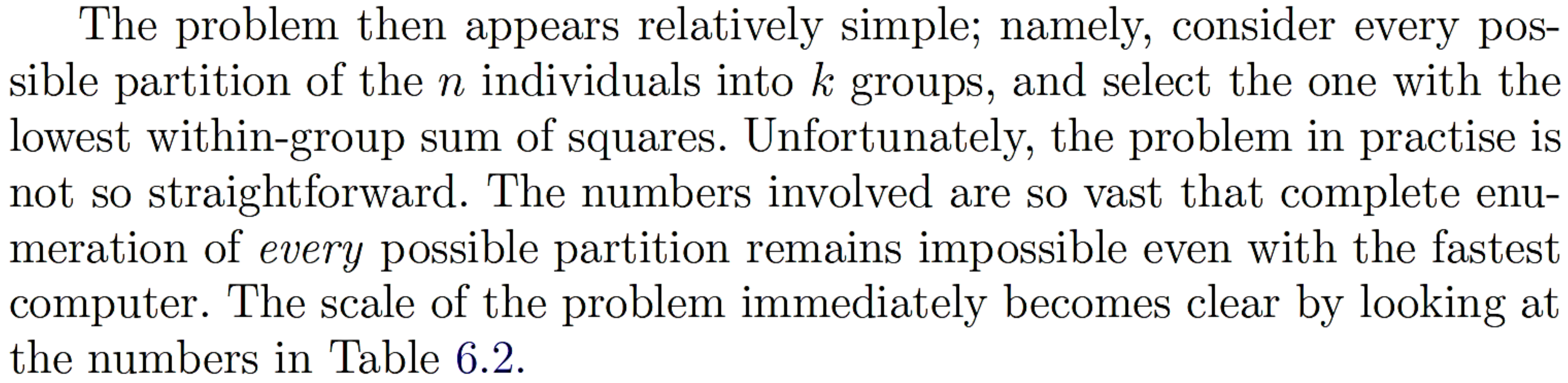
The k-means clustering technique seeks to partition the n individuals in a set of multivariate data into k groups or clusters, $(G_1, G_2, \ldots, G_k)$, where G_i denotes the set of n_i individuals in the ith group, and k is given (or a possible range is specified by the researcher–the problem of choosing the "true" value of k will be taken up later) by minimising some numerical criterion, low values of which are considered indicative of a "good" solution. The most commonly used implementation of k-means clustering is one that tries to find the partition of the n individuals into k groups that minimises the *within-group sum of squares* (WGSS) over all variables; explicitly, this criterion is

$$\text{WGSS} = \sum_{j=1}^{q} \sum_{l=1}^{k} \sum_{i \in G_l} (x_{ij} - \overline{x}_j^{(l)})^2,$$

where $\overline{x}_j^{(l)} = \frac{1}{n_i} \sum_{i \in G_l} x_{ij}$ is the mean of the individuals in group G_l on variable j.

K-Means Clustering

Partitions the *n* individuals in a set of multivariate data into *k* groups or clusters.



The problem then appears relatively simple; namely, consider every possible partition of the n individuals into k groups, and select the one with the lowest within-group sum of squares. Unfortunately, the problem in practise is not so straightforward. The numbers involved are so vast that complete enumeration of *every* possible partition remains impossible even with the fastest computer. The scale of the problem immediately becomes clear by looking at the numbers in Table 6.2.

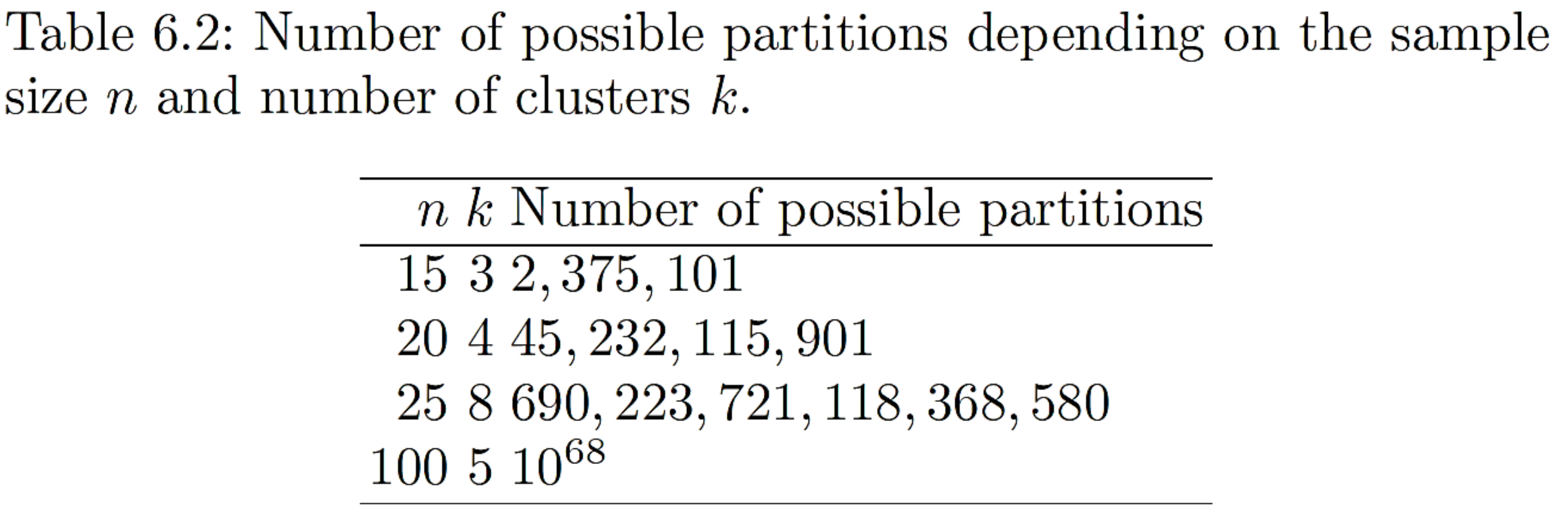
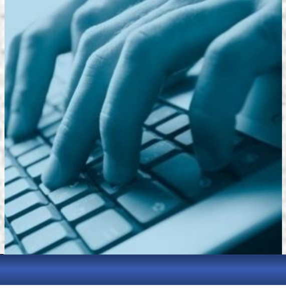


Table 6.2: Number of possible partitions depending on the sample size n and number of clusters k.

n	k	Number of possible partitions
15	3	$2,375,101$
20	4	$45,232,115,901$
25	8	$690,223,721,118,368,580$
100	5	10^{68}

Crime Data (partial)



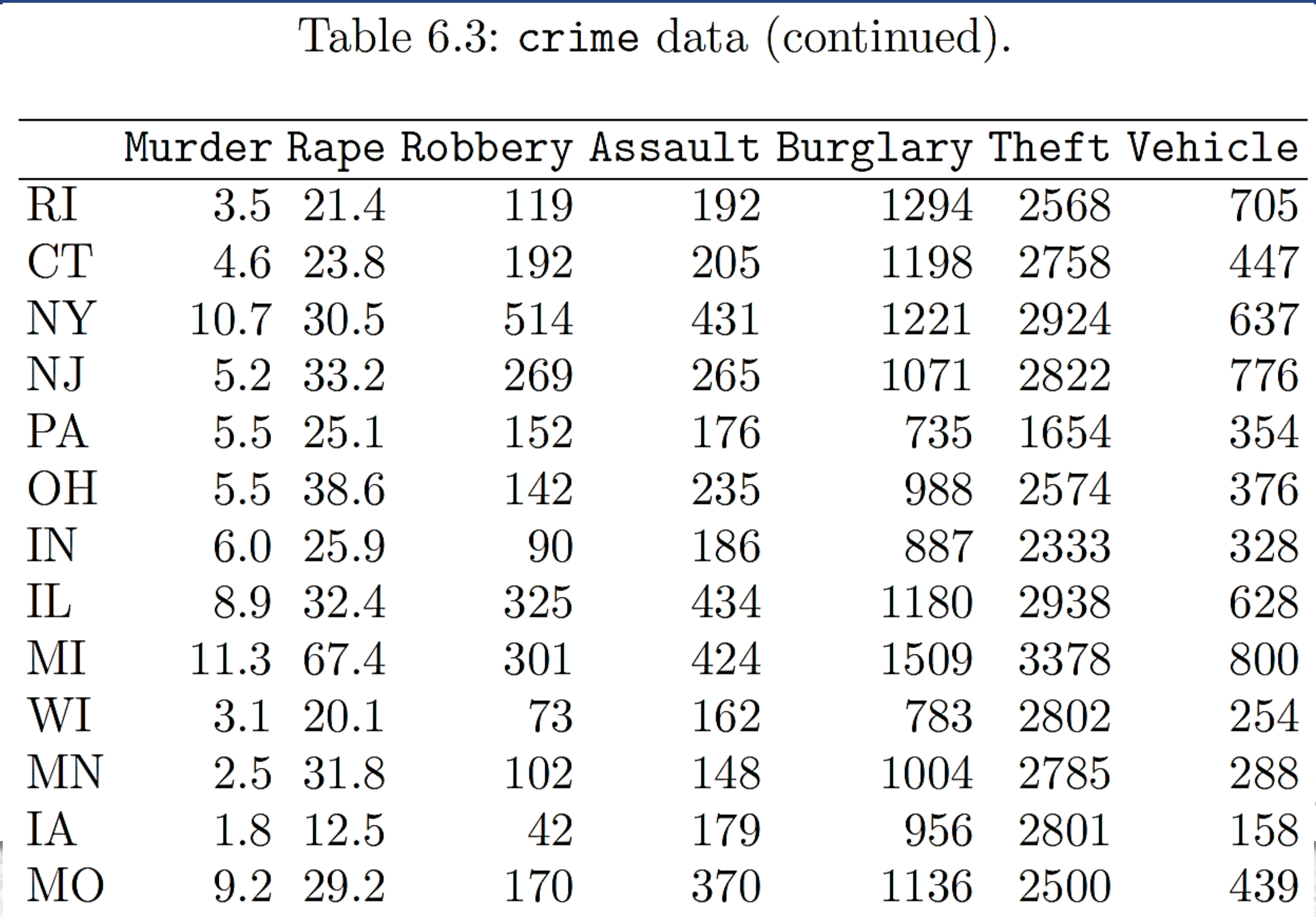
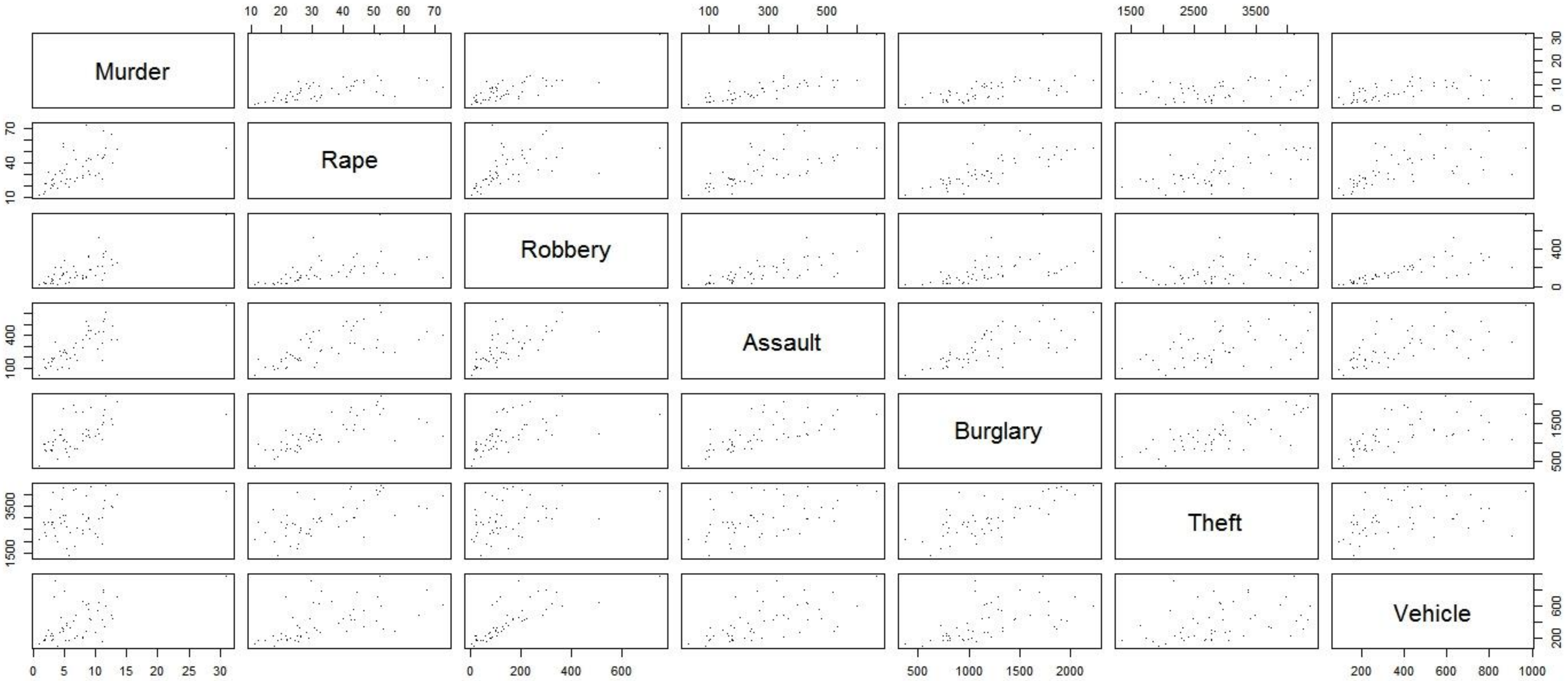


Table 6.3: crime data (continued).

	Murder	Rape	Robbery	Assault	Burglary	Theft	Vehicle
RI	3.5	21.4	119	192	1294	2568	705
CT	4.6	23.8	192	205	1198	2758	447
NY	10.7	30.5	514	431	1221	2924	637
NJ	5.2	33.2	269	265	1071	2822	776
PA	5.5	25.1	152	176	735	1654	354
OH	5.5	38.6	142	235	988	2574	376
IN	6.0	25.9	90	186	887	2333	328
IL	8.9	32.4	325	434	1180	2938	628
MI	11.3	67.4	301	424	1509	3378	800
WI	3.1	20.1	73	162	783	2802	254
MN	2.5	31.8	102	148	1004	2785	288
IA	1.8	12.5	42	179	956	2801	158
MO	9.2	29.2	170	370	1136	2500	439

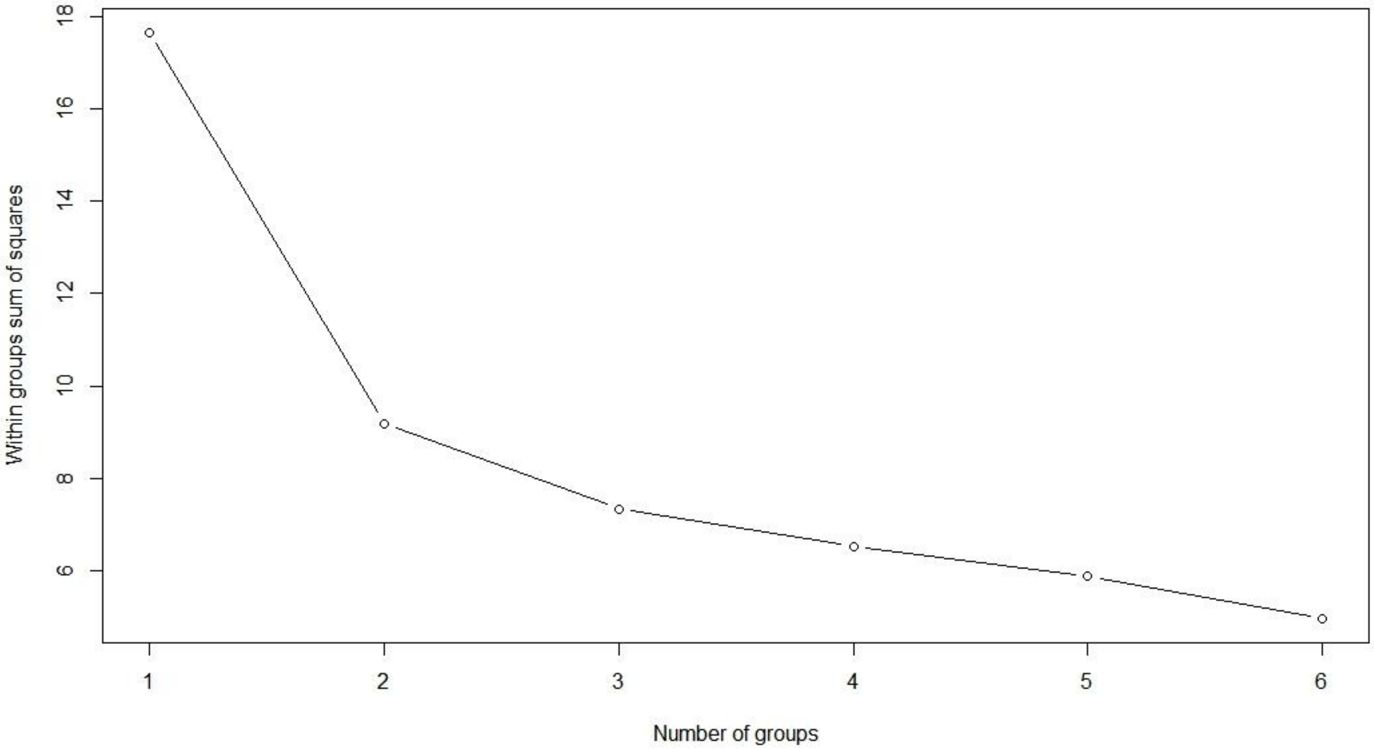
Crime Data

Crime Data Scatterplot Matrix:



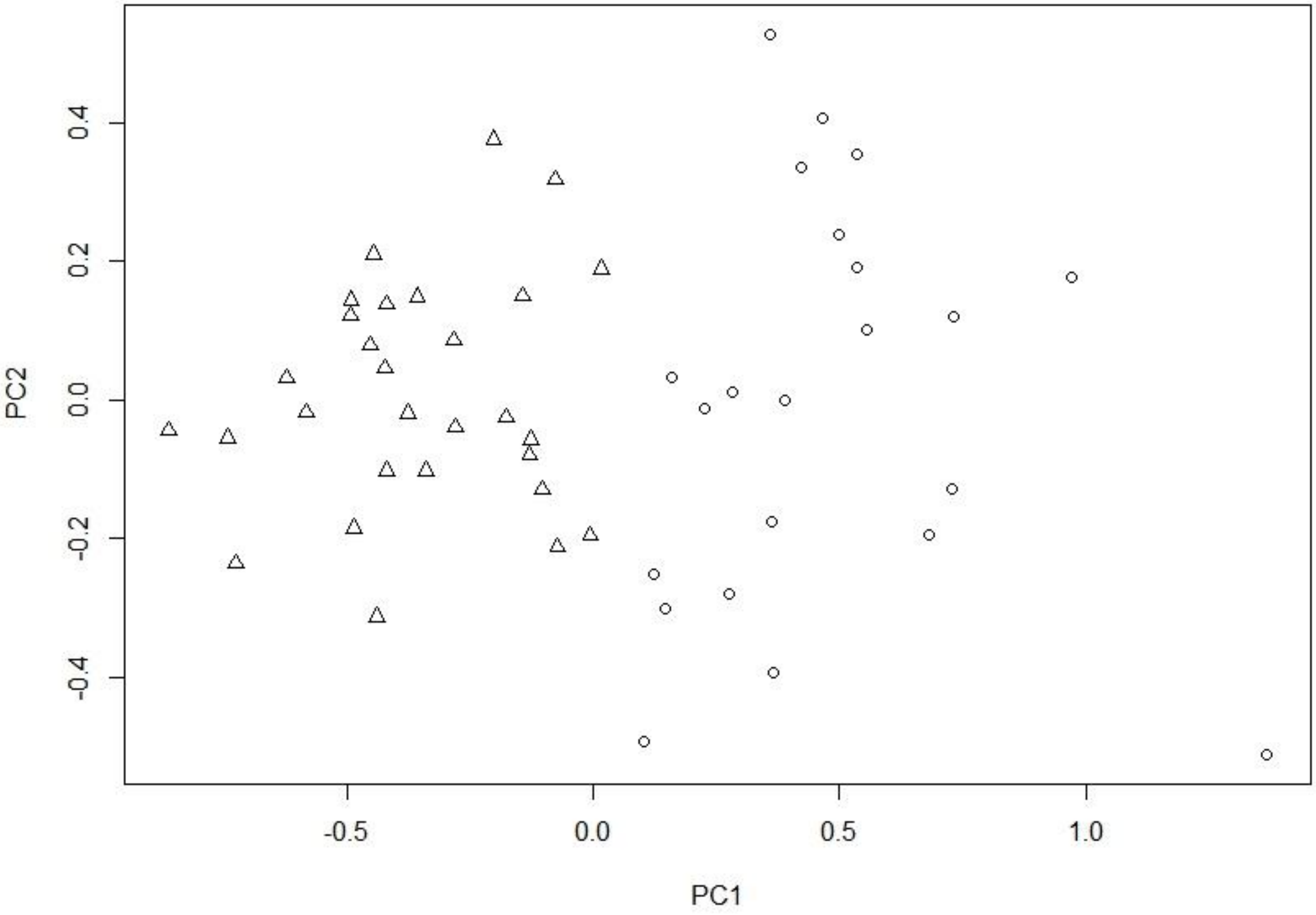
Crime Data

Within-Groups Sums of Squares for 1-6 Group Solutions:



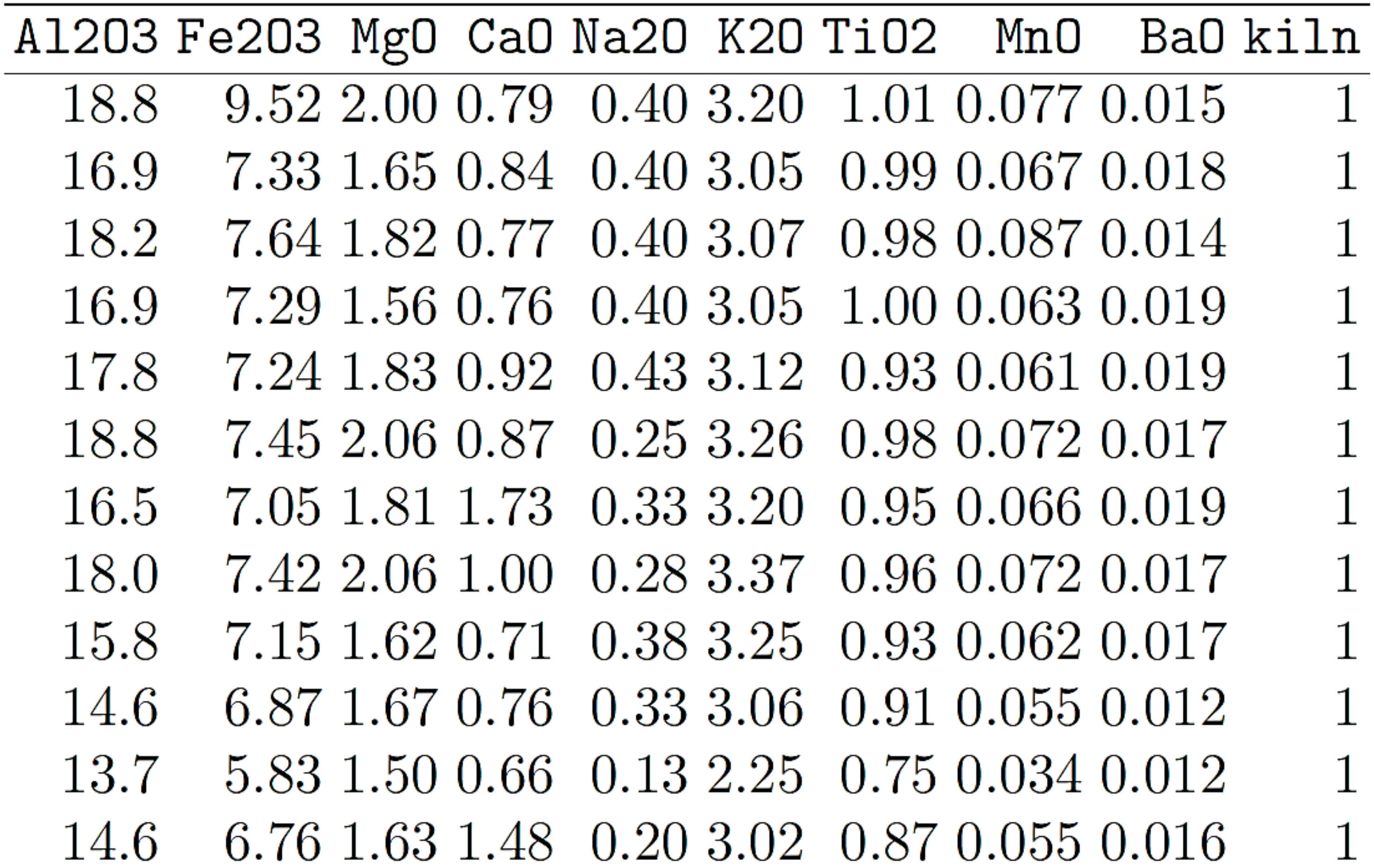
Crime Data

Two-Group Solution Simply Based on First Principal Component Score:



pottery Data

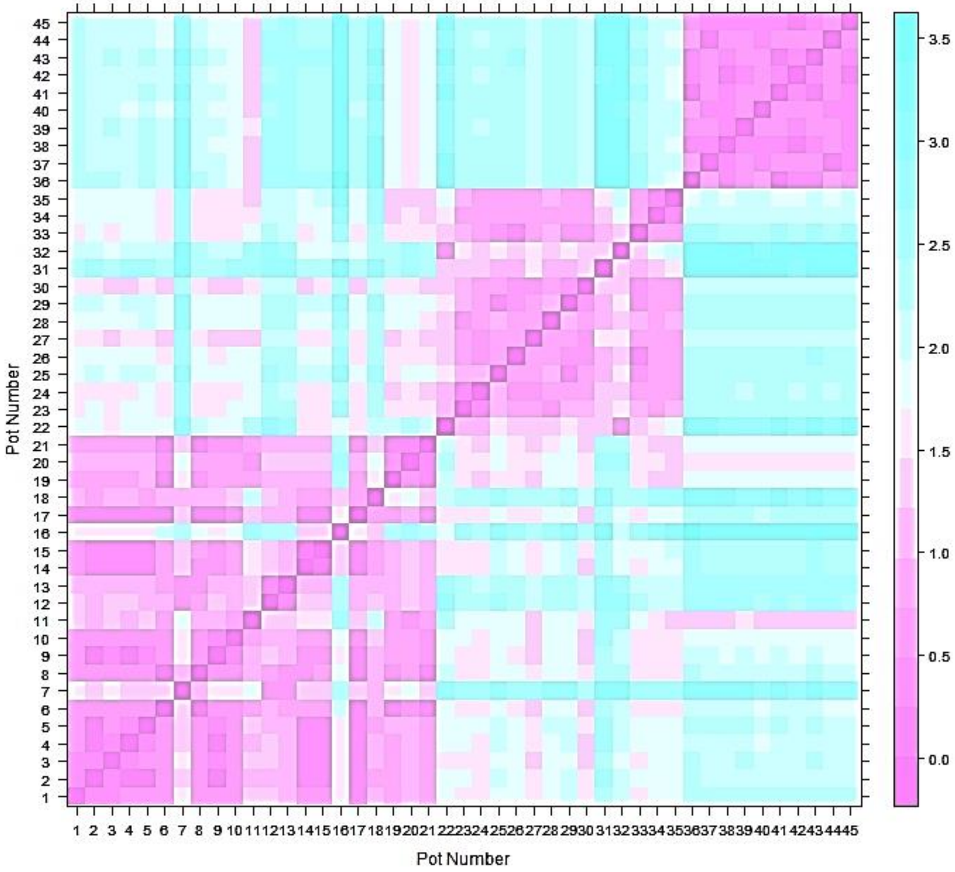
pottery data (partial). Romano-British pottery data:



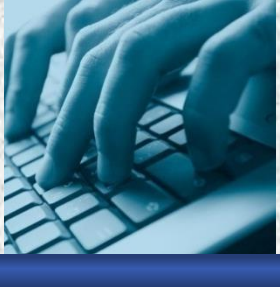
Al2O3	Fe2O3	MgO	CaO	Na2O	K2O	TiO2	MnO	BaO	kiln
18.8	9.52	2.00	0.79	0.40	3.20	1.01	0.077	0.015	1
16.9	7.33	1.65	0.84	0.40	3.05	0.99	0.067	0.018	1
18.2	7.64	1.82	0.77	0.40	3.07	0.98	0.087	0.014	1
16.9	7.29	1.56	0.76	0.40	3.05	1.00	0.063	0.019	1
17.8	7.24	1.83	0.92	0.43	3.12	0.93	0.061	0.019	1
18.8	7.45	2.06	0.87	0.25	3.26	0.98	0.072	0.017	1
16.5	7.05	1.81	1.73	0.33	3.20	0.95	0.066	0.019	1
18.0	7.42	2.06	1.00	0.28	3.37	0.96	0.072	0.017	1
15.8	7.15	1.62	0.71	0.38	3.25	0.93	0.062	0.017	1
14.6	6.87	1.67	0.76	0.33	3.06	0.91	0.055	0.012	1
13.7	5.83	1.50	0.66	0.13	2.25	0.75	0.034	0.012	1
14.6	6.76	1.63	1.48	0.20	3.02	0.87	0.055	0.016	1

pottery Data

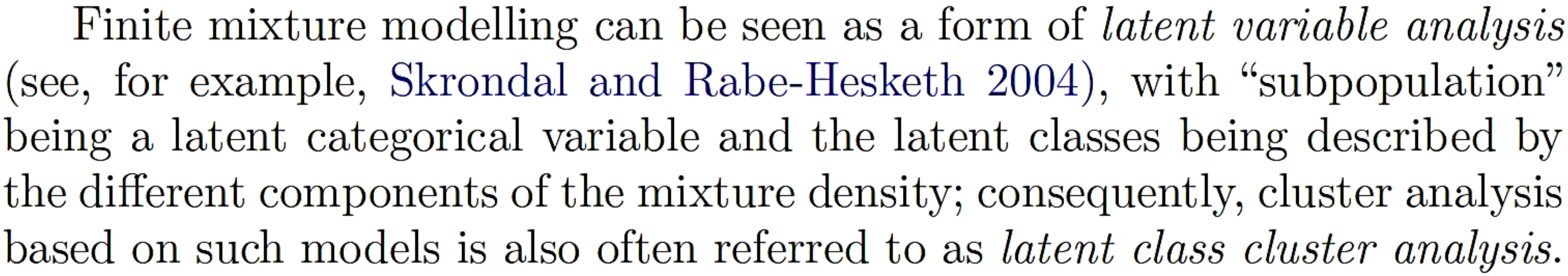
pottery data levelplotRomano-British pottery data:



Model-Based Clustering

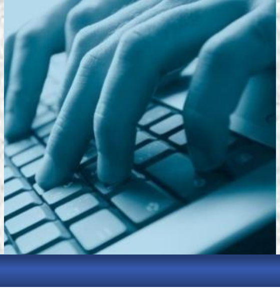


Finite mixture density approach; also known as *latent variable analysis*.

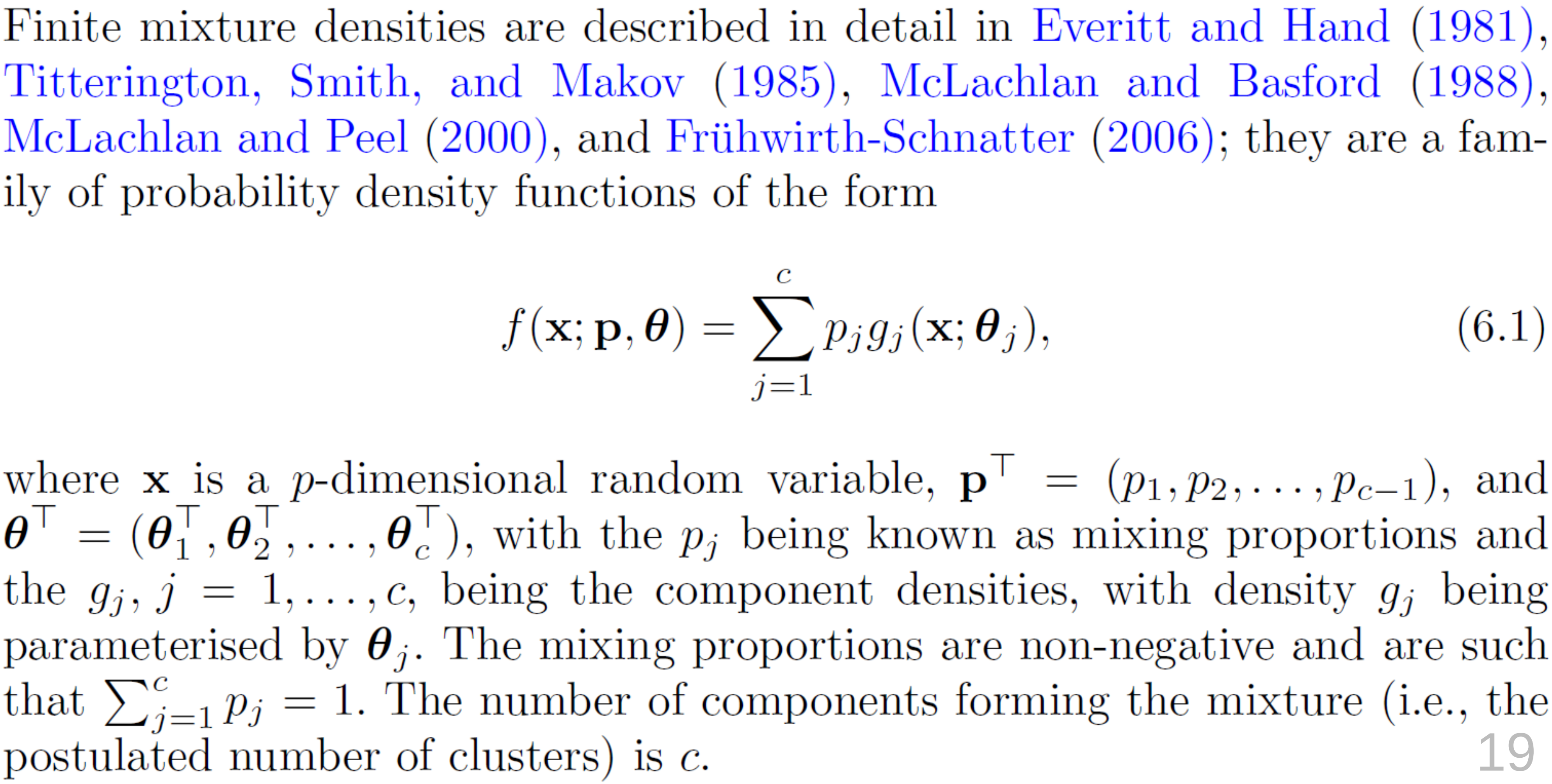


Finite mixture modelling can be seen as a form of *latent variable analysis* (see, for example, Skrondal and Rabe-Hesketh 2004), with “subpopulation” being a latent categorical variable and the latent classes being described by the different components of the mixture density; consequently, cluster analysis based on such models is also often referred to as *latent class cluster analysis*.

Model-Based Clustering



Finite mixture density approach; also known as *latent variable analysis*.



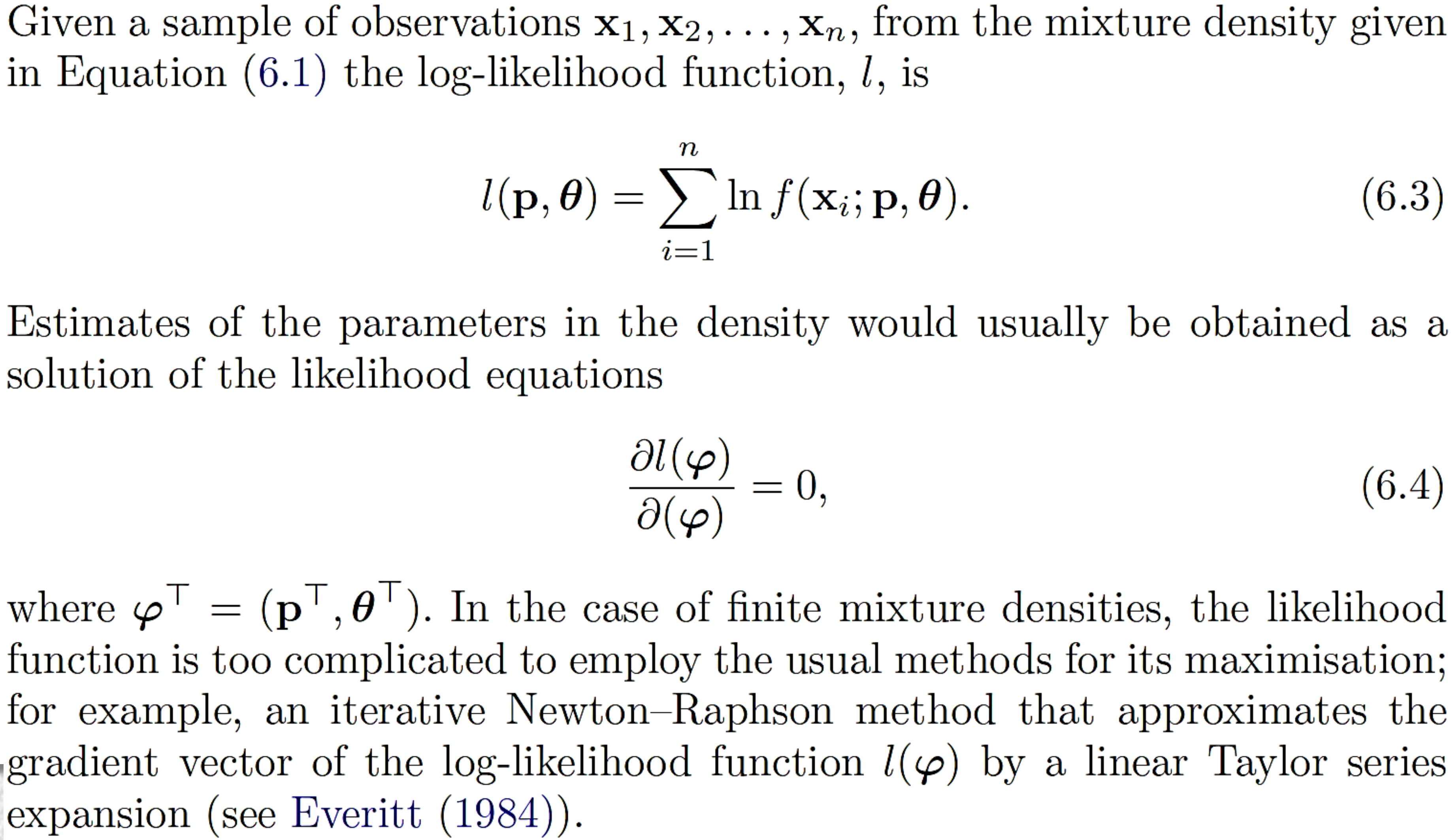
Finite mixture densities are described in detail in Everitt and Hand (1981), Titterington, Smith, and Makov (1985), McLachlan and Basford (1988), McLachlan and Peel (2000), and Frühwirth-Schnatter (2006); they are a family of probability density functions of the form

$$f(\mathbf{x}; \mathbf{p}, \boldsymbol{\theta}) = \sum_{j=1}^{c} p_j g_j(\mathbf{x}; \boldsymbol{\theta}_j), \tag{6.1}$$

where $\mathbf{x}$ is a p-dimensional random variable, $\mathbf{p}^\top = (p_1, p_2, \ldots, p_{c-1})$, and $\boldsymbol{\theta}^\top = (\boldsymbol{\theta}_1^\top, \boldsymbol{\theta}_2^\top, \ldots, \boldsymbol{\theta}_c^\top)$, with the p_j being known as mixing proportions and the g_j, $j = 1, \ldots, c$, being the component densities, with density g_j being parameterised by $\boldsymbol{\theta}_j$. The mixing proportions are non-negative and are such that $\sum_{j=1}^{c} p_j = 1$. The number of components forming the mixture (i.e., the postulated number of clusters) is c.

MLE in a Finite Mixture Density with Multivariate Normal Components

Finite mixture density approach; also known as *latent variable analysis*.



Given a sample of observations $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$, from the mixture density given in Equation (6.1) the log-likelihood function, l, is

$$l(\mathbf{p}, \boldsymbol{\theta}) = \sum_{i=1}^{n} \ln f(\mathbf{x}_i; \mathbf{p}, \boldsymbol{\theta}). \qquad (6.3)$$

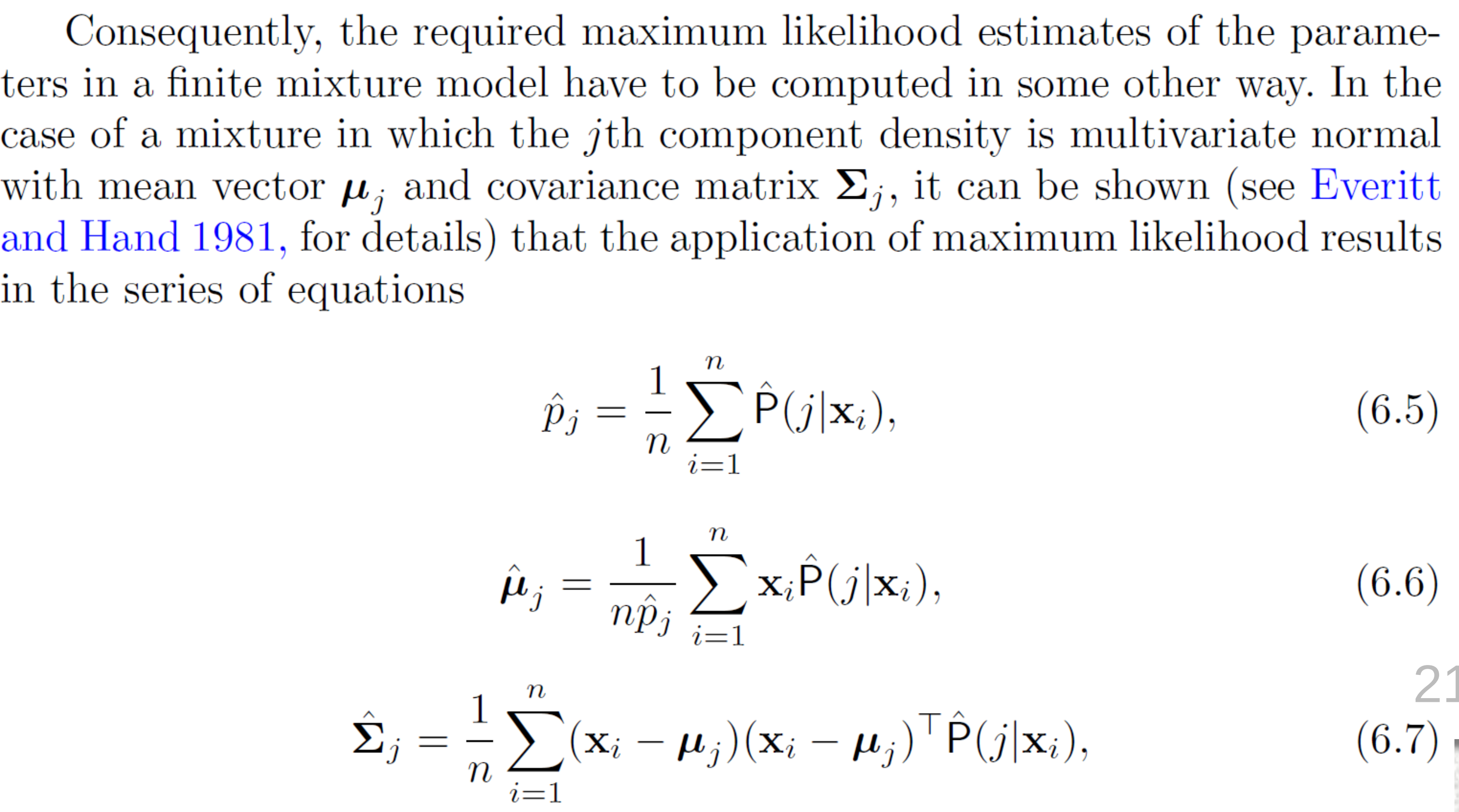
Estimates of the parameters in the density would usually be obtained as a solution of the likelihood equations

$$\frac{\partial l(\boldsymbol{\varphi})}{\partial(\boldsymbol{\varphi})} = 0, \qquad (6.4)$$

where $\boldsymbol{\varphi}^\top = (\mathbf{p}^\top, \boldsymbol{\theta}^\top)$. In the case of finite mixture densities, the likelihood function is too complicated to employ the usual methods for its maximisation; for example, an iterative Newton–Raphson method that approximates the gradient vector of the log-likelihood function $l(\boldsymbol{\varphi})$ by a linear Taylor series expansion (see Everitt (1984)).

MLE in a Finite Mixture Density with Multivariate Normal Components

Finite mixture density approach; also known as *latent variable analysis*.



Consequently, the required maximum likelihood estimates of the parameters in a finite mixture model have to be computed in some other way. In the case of a mixture in which the jth component density is multivariate normal with mean vector $\boldsymbol{\mu}_j$ and covariance matrix $\boldsymbol{\Sigma}_j$, it can be shown (see Everitt and Hand 1981, for details) that the application of maximum likelihood results in the series of equations

$$\hat{p}_j = \frac{1}{n}\sum_{i=1}^{n} \hat{\mathsf{P}}(j|\mathbf{x}_i), \qquad (6.5)$$

$$\hat{\boldsymbol{\mu}}_j = \frac{1}{n\hat{p}_j}\sum_{i=1}^{n} \mathbf{x}_i\hat{\mathsf{P}}(j|\mathbf{x}_i), \qquad (6.6)$$

$$\hat{\boldsymbol{\Sigma}}_j = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu}_j)(\mathbf{x}_i - \boldsymbol{\mu}_j)^\top\hat{\mathsf{P}}(j|\mathbf{x}_i), \qquad (6.7)$$

`mclust` Family of Mixture Models

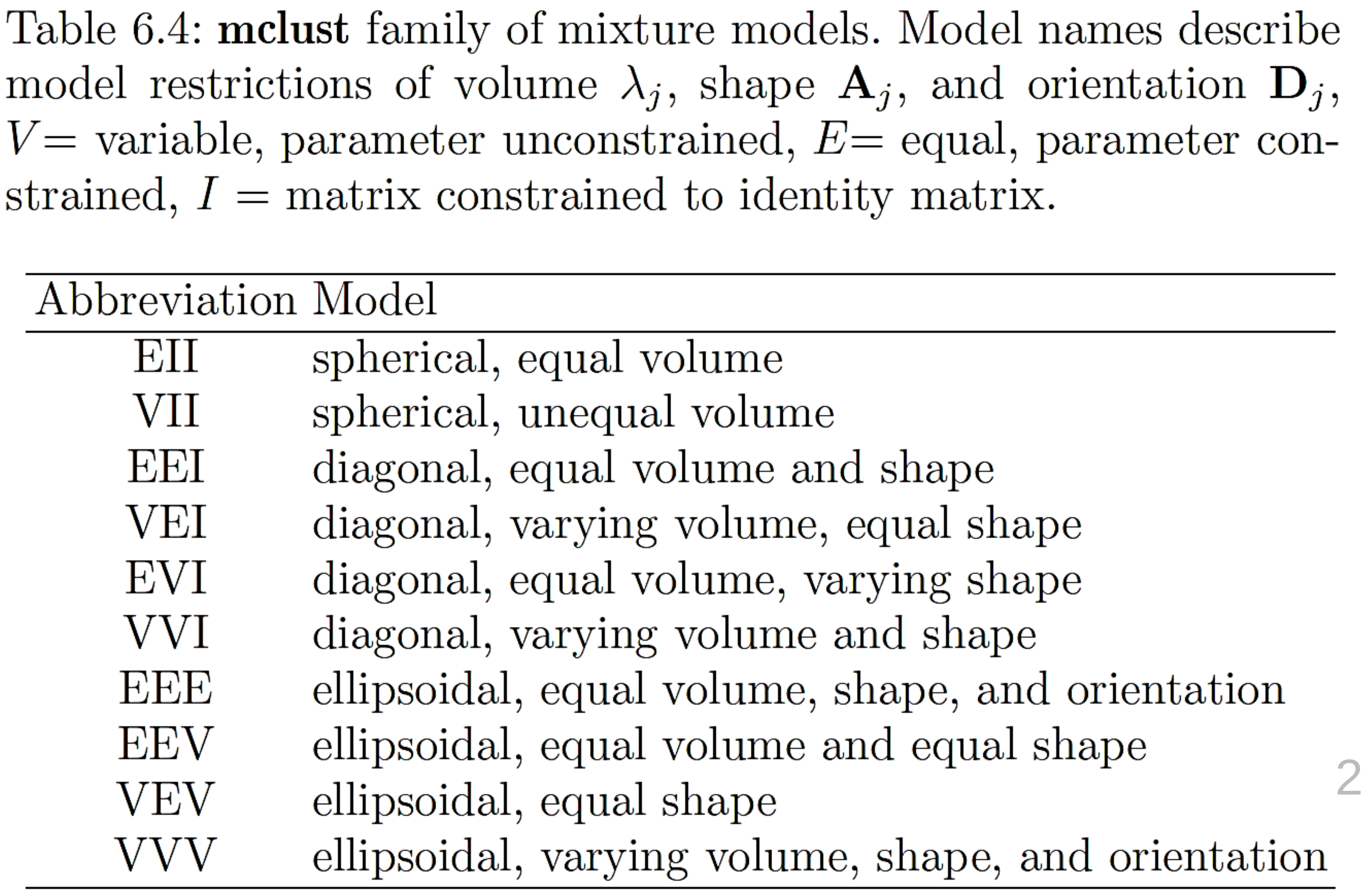


Table 6.4: **mclust** family of mixture models. Model names describe model restrictions of volume λ_j, shape $\mathbf{A}_j$, and orientation $\mathbf{D}_j$, $V=$ variable, parameter unconstrained, $E=$ equal, parameter constrained, I = matrix constrained to identity matrix.

Abbreviation	Model
EII	spherical, equal volume
VII	spherical, unequal volume
EEI	diagonal, equal volume and shape
VEI	diagonal, varying volume, equal shape
EVI	diagonal, equal volume, varying shape
VVI	diagonal, varying volume and shape
EEE	ellipsoidal, equal volume, shape, and orientation
EEV	ellipsoidal, equal volume and equal shape
VEV	ellipsoidal, equal shape
VVV	ellipsoidal, varying volume, shape, and orientation