CHAPTER 3

ANALYSIS OF THREE-WAY DATA BY WEIGHTED PRINCIPAL COMPONENTS ANALYSIS

A constrained variant of PARAFAC has been considered by Van der Kloot and Kroonenberg (1985, p. 483) and by Ten Berge, De Leeuw and Kroonenberg (1987, pp. 189–190). Recently, this constrained variant of PARAFAC has been applied by Van IJzendoorn and Kroonenberg (1990). In this variant of PARAFAC, the data are supposed to be based on the same components, but now these are to have the same relative importances from occasion to occasion; only the joint importance of the components differs from occasion to occasion. Accordingly, this method, called Weighted Principal Components Analysis (Weighted PCA) here, minimizes

$$\mathrm{WPCA}(A,B,\mathbf{c})=\sum_{k=1}^{p}\|X_k-c_kAB'\|^2, \tag{3.1}$$

where the component matrix A and the pattern matrix B are of the same orders as in PARAFAC, $\mathbf{c}$ is a p–vector such that $\mathbf{c}=(c_1,...,c_p)'$, and c_k denotes the weight for occasion k, $k=1,...,p$. From (3.1) it is clear that Weighted PCA minimizes the PARAFAC loss function subject to the constraint that $D_k=c_kI$, $k=1,...,p$. The main purpose of the present chapter is to compare this constrained variant of PARAFAC with unconstrained PARAFAC, and to give some special properties of Weighted PCA.

An obvious difference between PARAFAC and Weighted PCA is that PARAFAC will usually give the better fitting representation of the data, whereas Weighted PCA gives the more parsimonious representation (using fewer parameters than PARAFAC). As a result, Weighted PCA may be preferred over PARAFAC in cases where the discrepancy in fit is small. One such situation arises if the PARAFAC solution yields matrices $D_1,\ldots,D_p$ that are exactly proportional, because then Weighted PCA will represent the data exactly as well as PARAFAC does, as is readily verified. In case the PARAFAC solution yields $D_1,\ldots,D_p$ that are *nearly* proportional, it can be expected that the representation $c_1AB',\ldots,c_pAB'$ fits the data almost as well as the unconstrained PARAFAC representation.

Another difference between Weighted PCA and PARAFAC is related to uniqueness. In Weighted PCA, which is PARAFAC with the matrices $D_1,\ldots,D_p$ constrained to be proportional, Kruskal's (1977) sufficient condition $k_a+k_b+k_c \geq 2q+2$ for uniqueness is not satisfied. In fact, Weighted PCA does not have a unique solution, as follows at once from $c_kAB'=c_kATT^{-1}B'=c_k\overset{*}{A}\overset{*}{B}'$, $k=1,\ldots,p$, where $\overset{*}{A}=AT$, $\overset{*}{B}=B(T')^{-1}$, for any non-singular T.

In the previous chapter, it has been proposed to use PARAFAC with two proportional columns in, for instance, C to detect weak uniqueness. This raises the question how PARAFAC with two proportional columns in C is related to Weighted PCA, which can be seen as PARAFAC with proportional columns throughout the matrix C. From (3.1) it is clear that, for dimensionality 2, Weighted PCA and PARAFAC with two proportional columns in C coincide and, for dimensionality greater than 2, Weighted PCA is a constrained variant of PARAFAC with two proportional columns in C. Apart from this relation of Weighted PCA with the constrained PARAFAC methods in the previous chapter, one may wonder how Weighted PCA is related to other methods to analyze three-way data. Kiers (1991) has presented a hierarchy for a number of three-way methods, where every method in the hierarchy is a constrained variant of its predecessor. Weighted PCA is not mentioned in this hierarchy. In the present chapter, it will be shown how Weighted PCA fits into this hierarchy.

Van IJzendoorn and Kroonenberg (1990) used the TUCKALS3 algorithm (Kroonenberg & De Leeuw, 1980; Kroonenberg, Ten Berge, Brouwer, & Kiers, 1989) for Weighted PCA. It will be shown in the present chapter that a simpler and more efficient algorithm than the TUCKALS3 algorithm can be used for Weighted PCA.

In the case of exploratory data analysis, one is interested in finding a representation for the data that fits the data satisfactorily and consists of components that allow an easy interpretation. In this chapter, it will be illustrated that Weighted PCA may fit the data almost as well as PARAFAC, and that the Weighted PCA representation may be preferred because its components allow an easier interpretation. In addition, it will be illustrated that, in case PARAFAC yields a degenerate solution, Weighted PCA may be used in addition to PFORTA, to avoid a degenerate solution. Specifically, it will be illustrated that from a PFORTA and a Weighted PCA representation different types of conclusions can be drawn. First, however, a reinterpretation of Weighted PCA will be given.

3.1 Weighted PCA interpreted as PCA of a weighted sum of data matrices

An important property of the Weighted PCA method is that, for fixed $\mathbf{c}$ scaled to unit length, Weighted PCA is equivalent to PCA of $\sum_{k=1}^{p} c_k X_k$. That is, for optimal A and B the product AB' optimally represents $\sum_{k=1}^{p} c_k X_k$, so Weighted PCA can be interpreted as PCA of a weighted sum of the frontal slabs, hence its name. Specifically, for fixed $\mathbf{c}$ with $\mathbf{c}'\mathbf{c}=1$, and from (3.1) it follows that

$$\mathrm{WPCA}(A,B,\mathbf{c})=\sum_{k=1}^{p}\|X_k\|^2-2\mathrm{tr}\sum_{k=1}^{p}c_kX_k'AB'+\|AB'\|^2,$$

$$=\|\sum_{k=1}^{p}c_kX_k-AB'\|^2+\sum_{k=1}^{p}\|X_k\|^2-\|\sum_{k=1}^{p}c_kX_k\|^2. \qquad (3.2)$$

Because $\mathbf{c}$ is considered fixed, $\sum_{k=1}^{p}\|X_k\|^2-\|\sum_{k=1}^{p}c_kX_k\|^2$ is constant. So for the A and B that minimize WPCA(A,B,$\mathbf{c}$), the product AB' must optimally represent $\sum_{k=1}^{p} c_k X_k$, and can therefore be interpreted as PCA of $\sum_{k=1}^{p} c_k X_k$.

3.2 How Weighted PCA fits in a hierarchy of three-way methods

Kiers (1991) has presented the following hierarchical relations between a number of three-way methods: SUMPCA, a method to be discussed shortly, is a constrained variant of PARAFAC subject to orthonormality constraints on A and B (called PFORTAB here); PFORTAB is a constrained variant of PARAFAC; PARAFAC can be seen as constrained TUCKALS3 (see Carroll & Chang, 1970, p. 312). We will now show that Weighted PCA fits in this hierarchy at a position in between SUMPCA and PFORTAB. Specifically, it will be shown that Weighted PCA is constrained PFORTAB and that SUMPCA is constrained Weighted PCA.

It will first be shown that Weighted PCA is a constrained variant of PFORTAB. For this purpose, the Weighted PCA loss function may be written as

$$\text{WPCA}(A,B,\mathbf{c},D)=\sum_{k=1}^{p}\|X_k-Ac_kDB'\|^2, \tag{3.3}$$

where D is a diagonal matrix. Analogously to the derivation of (3.2), it can be shown that in (3.3) A and B can be constrained to be column-wise orthonormal without loss of fit. It follows at once that Weighted PCA can be seen as PFORTAB with D_k constrained to be c_kD, k=1,...,p.

It will secondly be shown that SUMPCA is a constrained variant of Weighted PCA. From Kiers (1991), it can be seen that SUMPCA minimizes

$$\text{SUMPCA}(A,B,D)=\sum_{k=1}^{p}\|X_k-ADB'\|^2, \tag{3.4}$$

where A and B are of the same order as in (3.1) and D is a diagonal matrix. To show that SUMPCA is a constrained variant of Weighted PCA, it merely has to be noted that imposing the constraint $c_k=c$, k=1,...,p, in Weighted PCA and absorbing c in D gives the description of SUMPCA.

Clearly, SUMPCA consists of less parameters than Weighted PCA. Therefore, SUMPCA provides a more parsimonious representation of the three-way data than Weighted PCA and might be preferred if it represents the data (almost) as well as Weighted PCA. Similarly, Weighted PCA provides a more parsimonious representation of the three-way data than PFORTAB and PARAFAC and it might be preferred if it represents the data (almost) as well as PFORTAB and PARAFAC.

3.3 An Algorithm for Weighted PCA

Van IJzendoorn and Kroonenberg (1990) used the TUCKALS3 algorithm for Weighted PCA. The TUCKALS3 algorithm minimizes

$$\text{TUCKALS3}(A,B,C,G_1,\ldots,G_{q_3})=\sum_{k=1}^{p}\|X_k-A\sum_{l=1}^{q_3}c_{kl}G_lB'\|^2, \tag{3.5}$$

over column-wise orthonormal matrices $A(n\times q_1)$, $B(m\times q_2)$, $C(p\times q_3)$, and arbitrary matrices $G_1,\ldots,G_{q_3}$ of order $q_1\times q_2$, where q_1, q_2 and q_3 denote the dimensionality of the solution for A, B, and C, respectively. The so-called core array consists of the matrices $G_1,\ldots,G_{q_3}$, which contain interactions between the components collected in the columns of the matrices A, B, and C. As noted by Ten Berge et al. (1987, pp. 189–190), TUCKALS3 reduces to Weighted PCA in case $q_1=q_2=q$ and $q_3=1$. In the context of Weighted PCA the $q_1\times q_2$ core array has only one frontal slab, which will be denoted by G. It is proposed to use an alternative algorithm, which is more efficient. The alternative is based on considering Weighted PCA as a special case of TUCKALS3, with $q_1=q$, $q_2=m$ and $q_3=1$, so B is a square matrix and C is a vector, denoted by $\mathbf{c}$. Then, the TUCKALS3 algorithm minimizes

$$\text{TUCKALS3}(A,B,\mathbf{c},G)=\sum_{k=1}^{p}\|X_k-c_kAGB'\|^2. \tag{3.6}$$

Because B is a square orthonormal matrix, the rotational freedom of B in TUCKALS3 can be used, without loss of fit, to fix B to I_m. Then, the TUCKALS3 loss function coincides with the Weighted PCA loss function with G' fulfilling the role of B from Weighted PCA. For our special case, the TUCKALS3 algorithm can be elaborated as follows. According to Kiers, Kroonenberg and Ten Berge (1992) the TUCKALS3 update for G, using the fact that $B=I_m$, is

$$G=A'\sum_{k=1}^{p}c_kX_kB=A'\sum_{k=1}^{p}c_kX_k, \tag{3.7}$$

and the update for A is

$$A=\text{GS}\left(\sum_{k=1}^{p}c_kX_kBG'\right)=\text{GS}\left(\sum_{k=1}^{p}c_kX_kG'\right), \tag{3.8}$$

where $\text{GS}\left(\sum_{k=1}^{p}c_kX_kG'\right)$ denotes the Gram–Schmidt orthogonalization of $\sum_{k=1}^{p}c_kX_kG'$. Finally, the update for $\mathbf{c}$ is GS($\mathbf{d}$), where d_k, element k of $\mathbf{d}$, is equal to $\text{tr}A'X_kBG'=\text{tr}A'X_kG'$. In the TUCKALS3 algorithm, one iteration consists of updating G, A, G, B, G, and C, respectively (see Kiers et al., 1992). These iterative steps can be simplified without affecting the monotonical convergence of the TUCKALS3 algorithm. Note that B is constant, so the update for B and the consecutive update for G can be left out. Also note that, because C has rank 1, the above GS–update equals the optimal Bauer–Rutishauser update (see Kiers et al., 1992), hence updating G may be skipped, without affecting the monotonical convergence of the algorithm. In conclusion, it is proposed to update A for fixed G and $\mathbf{c}$, update G for fixed A and $\mathbf{c}$, and update $\mathbf{c}$ for fixed A and G, in each full cycle of the algorithm. This algorithm will be called the Weighted PCA algorithm. After the same stopping criteria of section 1.2 are satisfied, the optimal B in (3.1) is found as G' in the Weighted PCA algorithm.

In Appendix A it is proven that the above Weighted PCA algorithm and the TUCKALS3 algorithm yield the same series of function values. Weighted PCA is based on fewer (matrix) multiplications than TUCKALS3, as can be seen, for instance, by comparing the expressions for A^1 (see equation A.1 in Appendix A) and for E^1 in Appendix A. Therefore, Weighted PCA uses less computation time. In a few test runs on empirical data it was found that the Weighted PCA algorithm was about 3 times as fast as the TUCKALS3 algorithm.

A number of test trials on empirical data revealed that the Weighted PCA algorithm is not sensitive to local minima. Because the danger of local minima cannot be ruled out completely, it is suggested to run more than one Weighted PCA analysis on the same data with different starting configurations for the matrix A and the vector $\mathbf{c}$.

As a useful additional result, it is found that, if the A parameters have converged, the Weighted PCA algorithm gives the principal components of $\sum_{k=1}^{p} c_k X_k$. This can be seen as follows. Upon convergence of the A parameters in the algorithm, it follows that $A=\mathrm{GS}\left(\sum_{k=1}^{p}\sum_{l=1}^{p} c_l c_k X_k X_l' A\right) = \sum_{k=1}^{p}\sum_{l=1}^{p} c_l c_k X_k X_l' A U$, for a certain upper triangular matrix U. Hence, $I_q=A'A=\left(A'\sum_{k=1}^{p}\sum_{l=1}^{p} c_l c_k X_k X_l' A\right)U$. For all practical purposes it may be assumed that $\left(A'\sum_{k=1}^{p}\sum_{l=1}^{p} c_l c_k X_k X_l' A\right)$ has full rank. Then, $U=\left(A'\sum_{k=1}^{p}\sum_{l=1}^{p} c_l c_k X_k X_l' A\right)^{-1}$, which is symmetric. Because U is also upper triangular, it follows that U

is diagonal. So, $\left(\sum_{k=1}^{p}\sum_{l=1}^{p} c_l c_k X_k X_l'\right) A = AU^{-1}$, and hence A contains eigenvectors of $\left(\sum_{k=1}^{p}\sum_{l=1}^{p} c_l c_k X_k X_l'\right)$ and therefore the principal components of $\left(\sum_{k=1}^{p} c_k X_k\right)$, which completes the proof.

Recently, Kiers et al. (1992) proposed a modification of the TUCKALS3 algorithm that handles three-way arrays of order $n \times m \times p$ for any n. The modified algorithm is efficient in the sense that, if $n > mp$, it needs less work space to store the data during the iterative process on a computer and it has a higher execution speed. Above, the Weighted PCA algorithm was derived as a special case of the TUCKALS3 algorithm, and hence the modified TUCKALS3 algorithm can also be used for Weighted PCA. This modified TUCKALS3 algorithm is based on updating A implicitly. For Weighted PCA, this modified TUCKALS3 algorithm can be further simplified in the sense that updating A can be skipped from the iterative process. This has the advantage over the modified TUCKALS3 algorithm that even less storage space is needed and that computational speed is increased. This can be seen as follows. Without loss of generality, for the TUCKALS3$(A,B,\mathbf{c},G) = \sum_{k=1}^{p} \|X_k - c_k AGB'\|^2$ loss function Weighted PCA may be subjected to the constraints $B = I_m$, $\mathbf{c}'\mathbf{c} = 1$ and $G'G = I_q$. It can be verified that, for fixed $\mathbf{c}$ and G, it follows from regression theory (Draper & Smith, 1981) that the optimal A satisfies $A = \left(\sum_{k=1}^{p} c_k X_k BG'\right) = \left(\sum_{k=1}^{p} c_k X_k G'\right)$. Upon substituting $\left(\sum_{k=1}^{p} c_k X_k G'\right)$ for A into TUCKALS3$(A,B,\mathbf{c},G)$ and using the fact that $B = I_m$, it follows that minimizing TUCKALS3$(A,B,\mathbf{c},G)$ is equivalent to maximizing

$$g(G,\mathbf{c}) = \mathrm{tr}G'\left(\sum_{k=1}^{p} c_k X_k\right)'\left(\sum_{l=1}^{p} c_l X_l\right)G = \sum_{k=1}^{p}\sum_{l=1}^{p} c_k c_l \mathrm{tr}G' X_k' X_l G = \mathbf{c}'V\mathbf{c}, \tag{3.9}$$

where V is a positive semi-definite matrix with $v_{kl}=\text{tr}G'X_k'X_lG$ as element (k,l), subject to the constraints $\mathbf{c}'\mathbf{c}=1$ and $G'G=I_q$. From (3.9) it can be seen that G and $\mathbf{c}$ can be updated by using Bauer–Rutishauser or by using Gram–Schmidt in the same way as in the TUCKALS3 algorithm (Kroonenberg & De Leeuw, 1980; Kroonenberg et al., 1989). This shows that, in order to minimize the TUCKALS3 loss function for Weighted PCA, only G and $\mathbf{c}$ need to be updated, and only the matrix with $X_k'X_l$, $k=1,...,p$ and $l=1,...,p$ need to be stored by the computer.

3.4 Interpretation of the Weighted PCA components

In chapter 1 the structure matrix for PARAFAC was defined as $S=\sum_{k=1}^{p} X_k'AD_k$, and it was proposed to use S for the interpretation of the PARAFAC components. Analogously, a structure matrix for Weighted PCA can be defined as $S\equiv\sum_{k=1}^{p} c_kX_k'A$, which follows from substituting c_kI_q for D_k in $S=\sum_{k=1}^{p} X_k'AD_k$.

It is well-known that in PCA, when the components are orthonormal, the structure matrix and the pattern matrix are equal. With our definition of the structure matrix this property is retained in the Weighted PCA generalization of PCA. That is, if $A'A=I_q$ and $\mathbf{c}'\mathbf{c}=1$, which can always be arranged, then $B=S$. This can be proven by using the substitution c_kI_q for D_k analogously to the proof for $S=B$ in case of PFORTA, see section 1.5. Therefore, using S or B as a basis for interpretation of the Weighted PCA components leads to the same interpretation if $A'A=I_q$ and $\mathbf{c}'\mathbf{c}=1$.

The occasion parameters in $\mathbf{c}$ of Weighted PCA can be interpreted from the knowledge of the occasions. In addition to this, it may be important to know how well the Weighted PCA model fits the data for each occasion separately. For this reason, the constraint $\mathbf{c}'\mathbf{c}=1$ will be dropped and the constraint $\|B\|^2=1$ will be imposed instead, as can be arranged without loss of fit. If X_k is centered column-wise, and if c_k is optimal, then c_k^2 is the amount of variance in X_k that the Weighted PCA components explain, $k=1,...,p$. This can be proven as follows. From the optimality of c_k, it follows that c_k minimizes $\mathrm{f}(c_k)=\|X_k-c_kAB'\|^2$. From this, $A'A=I_q$ and $\|B\|^2=1$, it follows that $c_k=\mathrm{tr}A'X_kB$, because

$$\mathrm{f}(c_k)=\|X_k-c_kAB'\|^2=\|X_k\|^2-2\mathrm{tr}A'X_kBc_k+c_k^2+(\mathrm{tr}A'X_kB)^2-(\mathrm{tr}A'X_kB)^2$$

$$=(\mathrm{tr}A'X_kB-c_k)^2+\|X_k\|^2-(\mathrm{tr}A'X_kB)^2. \tag{3.10}$$

This function is minimal for $c_k=\mathrm{tr}A'X_kB$. From the substitution of $\mathrm{tr}A'X_kB$ for c_k into (3.10) it follows that

$$\mathrm{f}(c_k)=\|X_k\|^2-c_k^2. \tag{3.11}$$

From (3.11), it can be seen that for occasion k the amount of residual variance, written as $\|X_k-c_kAB'\|^2$, is partitioned into the amount of variance to be explained $\|X_k\|^2$, and the squared element of the occasion parameter c_k^2. Therefore, c_k^2 is the amount of variance that Weighted PCA explains in X_k, $k=1,...,p$, which completes the proof.

3.5 An empirical example

To illustrate the above hierarchical relations and Weighted PCA, a data set, denoted as GOS* data, with scores of 34 children of age 2–3 on 10 variables that measure intelligence on two occasions, was analyzed. The variables are divided into four groups: Simultaneous Processing (Magic Window, Face Recognition, and Gestalt Closure), Achievement (Expressive Vocabulary and Faces and Places), Sequential Processing (Hand Movements and Number Recall), and Motor Skills (Gross Motor Skills, Fine Motor Skills, and Figure Movement in Disc), see Van Eldik, Neutel, Van der Meulen, and Spelberg (1990). Prior to the analysis, the variables were centered column-wise within the occasions, and scaled to unit length over the occasions.

To illustrate the above hierarchical relations the GOS data were analyzed by SUMPCA, Weighted PCA and PARAFAC, all with dimensionality 2. The matrix C that resulted from the PARAFAC analysis, with A and B scaled to unit length column-wise, was $\begin{pmatrix} 1.14 & .61 \\ 1.61 & 1.27 \end{pmatrix}$. The value of Tucker's (1951) coefficient of congruence between the two columns of C is .99. So it can be concluded that the columns in the matrix C are nearly proportional. From this it can be expected that the discrepancy between PARAFAC and Weighted PCA is small. The percentage of variance explained by PARAFAC, Weighted PCA, and SUMPCA was 47.5, 47.2, and 45.5, respectively. The small discrepancy in fit between PARAFAC and Weighted PCA indicates that these

---

*The author is obliged to Ria Neutel and Henk Lutje Spelberg who kindly made the data available.

PARAFAC components are weakly unique. The SUMPCA model, or, equivalently, PCA of the average matrix, seems not very restrictive for these data.

To illustrate Weighted PCA and to compare Weighted PCA with PARAFAC, more detailed results of the analyses are given now. The occasion parameters that resulted from Weighted PCA (with $\|B\|^2=1$) for occasion 1 and 2 were 1.18 and 1.82, respectively. By simply squaring these values, see section 3.4, it can be found that the amount of variance explained for the first occasion is 1.40 and for the second occasion 3.32. The amounts of variance to be explained were 4.01 and 5.99, for occasion 1 and 2, respectively. So the Weighted PCA components explain 34.6% and 55.4% of the variance for occasion 1 and 2, respectively. For PARAFAC, the corresponding percentages are 34.6% and 56.2%, respectively. From this it can be concluded that for both methods the data for occasion 2 are represented better than those for occasion 1. The structure matrices resulting from PARAFAC and Weighted PCA are depicted in Table 3.1. Before the structure matrices were computed A, C, and $\mathbf{c}$ were scaled to unit length column-wise.

From Table 3.1 it can be seen that the structure matrices from Weighted PCA (without rotation) and PARAFAC are quite similar. The variables are assumed to belong to two clusters, but the PARAFAC components, as well as the Weighted PCA components, do not correspond to these clusters of variables. Now the fact that Weighted PCA can be seen as PCA of $\sum_{k=1}^{p} c_k X_k$ allows one to use all available rotation techniques for PCA in Weighted PCA analysis, as if $\sum_{k=1}^{p} c_k X_k$ were analyzed by PCA. It was chosen to rotate the components to oblique components that approximate simple structure, by using the independent cluster rotation proposed by Harris and Kaiser (1964). This resulted in rotated components that correlate 0.34. In Table

3.1 the structure matrix from rotated Weighted PCA analysis is depicted with elements greater than .40 in bold face. It can be seen from the rotated Weighted PCA structure matrix, that the variables of the first two groups load mainly on the first rotated component, and the variables of the second two groups load mainly on the second rotated component. So the components may be interpreted as Simultaneous Processing and Sequential Processing, respectively. Apparently, by using the rotational freedom that Weighted PCA has, components can be identified that correspond better to the clusters of variables and therefore allow for a simpler interpretation than the PARAFAC components.

Table 3.1 *The structure matrices from PARAFAC, and unrotated and rotated Weighted PCA analysis of the GOS data.*

	PARAFAC		WPCA		ROTATED	
Variable	I	II	I	II	I	II
Magic Window	.76	.48	.68	.40	**.78**	.19
Face Recognition	.62	.40	.56	.29	**.63**	.19
Gestalt Closure	.63	.36	.58	.27	**.64**	.21
Expressive Vocabulary	.78	.37	.73	.27	**.77**	.31
Faces and Places	.75	.40	.70	.29	**.76**	.28
Hand Movements	.49	−.33	.63	−.41	.42	**.73**
Number Recall	.31	−.53	.48	−.60	.20	**.76**
Gross Motor Skills	.28	−.54	.46	−.62	.17	**.77**
Fine Motor Skills	.32	−.21	.41	−.29	.26	**.49**
Figure Movement in Disc	.27	−.02	.31	−.13	.23	.30

In order to assess the stability of the PARAFAC and the rotated Weighted PCA components the data were subjected to five different splithalf analyses. After determining the two sets randomly the data were preprocessed per set as before. For PARAFAC only one component in one splithalf analysis was stable. The lowest congruence values encountered in the five splithalf analyses were .62, .57, .56, .67, and .26. For Weighted PCA, Tucker's congruence coefficient was computed between the corresponding **c** vectors and between the corresponding columns of the rotated matrix BT, were T is the rotation matrix, proposed by Harris and Kaiser (1964). For Weighted PCA with rotation, the five **c** vectors and the five first columns of B were stable and two of the second columns of B were stable. For the unstable columns of B the congruence values encountered were .65, .81, and .75. These results suggest that neither the PARAFAC nor the rotated Weighted PCA components are stable in the splithalf sense, and that the rotated Weighted PCA components are less unstable than the PARAFAC components.

3.6 Using Weighted PCA to arrive at a non-degenerate solution

Here it will be illustrated, by reporting the results of various analyses of the 32×6×8 Affective Response data (Eckblad, 1981), that PARAFAC may yield degenerate components and that for the same data Weighted PCA may be preferred over PFORTA. The Affective Response data consist of scores of 32 students who judged eight tasks of increasing complexity on six scales: Simple–Complex, Monotone–Varied, Boring–Interesting, Unpleasant–Pleasant, Uncomfortable–Comfortable and Disorderly–Clear. The tasks are manipulated such that they should only differ with respect to complexity, which is one–dimensional. So in advance it is at least doubtful that the tasks show distinct stretching and contraction for two PARAFAC components. Eckblad (1981, p. 3–6) predicted and found, by analyzing $\begin{pmatrix} X_1 \\ \vdots \\ X_p \end{pmatrix}$ by PCA, that the variables are represented (more or less) along a semi–circle. In addition, Eckblad (1981, pp. 3–4) predicted that the rank order of the tasks is the same for all the subjects. As was done by Eckblad (1981), the data array X was centered and scaled to unit length over the persons and the occasions. The Affective Response data were analyzed by PARAFAC in two dimensions. After 2000 iterative steps the value –.78 was found for cosABC. By taking more iterative steps until machine precision broke down the monotonical convergence the value of cosABC decreased further, but did not reach –.85. After 2000 iterative steps the cosines between the columns of A, B, and C are .96, .84, and –.98, respectively. This solution can be seen as borderline degenerate.

To avoid a degenerate solution, Harshman and Lundy (1984b, p. 274) suggest imposing column–wise orthonormality on one of the parameter matrices. It will now be illustrated that another way to arrive at a non–degenerate solution is to apply Weighted PCA. The Weighted PCA solution will be compared to that of PARAFAC with column–wise orthonormality on A. The Affective Response data were analyzed by PFORTA, Weighted PCA (without rotation), and SUMPCA. The percentage of explained variance by PFORTA, Weighted PCA, and SUMPCA is 44.3, 42.9, and 8.0, respectively. Clearly, SUMPCA is overly restrictive for the Affective Response data, whereas the Weighted PCA fit to the data is 1.4% of variance explained less than that of PFORTA, see also section 2.3. This rather small discrepancy between PFORTA and Weighted PCA suggests that these data can be fitted reasonably well by the more parsimonious Weighted PCA model. The structure matrices that resulted from PFORTA and from Weighted PCA (without rotation) are depicted in Table 3.2. Before the structure matrices were computed C and $\mathbf{c}$ were scaled to unit length column-wise.

From the structure matrices in Table 3.2, it can be seen that the variables are indeed (more or less) represented on a semi-circle. On the basis of the structure matrices, the PFORTA and the Weighted PCA components have almost identical interpretations, with the first component interpreted as Complexity and the second component as Pleasantness.

Table 3.2 *The structure matrices from PFORTA and Weighted PCA analysis of the Affective Response data.*

	Structure Matrices			
	PFORTA		Weighted PCA	
Variable	I	II	I	II
Simple–Complex	.80	–.10	.80	.03
Monotone–Varied	.70	.05	.69	.16
Boring–Interesting	.29	.40	.27	.43
Unpleasant–Pleasant	–.09	.48	–.12	.36
Uncomfortable–Comfortable	–.46	.35	–.48	.33
Disorderly–Clear	–.82	.14	–.83	.06

The matrix C (containing the task parameters) that resulted from PFORTA, and the vector $\mathbf{c}$ that resulted from Weighted PCA are reported in Table

3.3. For the task parameters in PFORTA, the matrix B was scaled to unit length column-wise and in Weighted PCA B was scaled such that $\|B\|^2=1$. As was explained in sections 1.3 and 3.4, by squaring these (task) parameter values it can be seen how much variance the corresponding component(s) explain(s) for the corresponding task.

From Table 3.3 it can be seen that the task parameters of Weighted PCA increase, except for one task, with the complexity of the tasks. Hence, Weighted PCA recovered, up to one exception, the rank order of the tasks consistently with the one-dimensional manipulation of the tasks.

Table 3.3 *The Task parameters from PFORTA and Weighted PCA analysis of the Affective Response data.*

	Task Parameters		
	PFORTA		Weighted PCA
Task	I	II	
1	−.72	−.17	−.76
2	−.59	−.04	−.64
3	−.44	−.10	−.48
4	−.20	.13	−.20
5	.27	.45	.38
6	.53	.29	.61
7	.41	.29	.46
8	.70	.35	.76

The first column of the task parameter matrix of PFORTA resembles the task parameters of Weighted PCA and can therefore be interpreted accordingly.

As explained in section 3.4, by squaring the elements of the second column of the task parameter matrix, it can be computed that the second component explains 0.1 of the sum of squares for the first four and 0.5 of the sum of squares for the last four tasks. Apparently, the second component accounts mainly for tasks 5 through 8, which are the more complex tasks.
In order to assess the stability of the PFORTA and the Weighted PCA components, the data were subjected to five different splithalf analyses. After determining the two sets randomly the data were preprocessed per set as before. For Weighted PCA, Tucker's congruence coefficient was computed between the corresponding **c** vectors and between the corresponding columns of the matrix B. The first PFORTA component was stable but the second was unstable over the five analyses. The lowest congruence values over B and C encountered for PFORTA in each of the five splits were .07, .34, .41, .60, and .76. The lowest congruence value encountered for Weighted PCA was .90. These results suggest that the PFORTA components are not stable, whereas the Weighted PCA components are stable in the splithalf sense.
Several conclusions can be drawn from the SUMPCA, the Weighted PCA, PFORTA and PARAFAC analysis of the Affective Response data. The SUMPCA model is overly restrictive for these data. The PARAFAC solution is borderline degenerate and is therefore unacceptable. PFORTA and Weighted PCA represent the Affective Response data almost equally well and can both be used to arrive at a non-degenerate solution. However, only the Weighted PCA components are stable in the splithalf sense. Up to one exception, the Weighted PCA task parameters recovered the rank order of the tasks.

3.7 Conclusions

A great discrepancy in fit between Weighted PCA and PARAFAC implies that the proportional columns constraint in Weighted PCA is overly restrictive. In such a case it seems appropriate for the components to have distinct relative importances from occasion to occasion. On the other hand, in case of a small discrepancy in fit between Weighted PCA and PARAFAC it seems appropriate for the components to have the same relative importances. In this sense, the Weighted PCA solution provides a more parsimonious representation of the three-way data. An even more parsimonious representation of the three-way data can be found by using the rotational freedom in Weighted PCA to approximate simple structure, as has been illustrated. It can be concluded that, in order to analyze three-way data, Weighted PCA can be useful in addition to PARAFAC.

In case one's goal is to find components that allow an easier interpretation, imposing other constraints than the proportional columns constraint on the occasion parameters, as is the case in Weighted PCA, may be useful. For instance, in case of three-way positive manifold data, components without contrasting signs allow an easier interpretation, as was explained in section 1.9. The subject of non-contrast components will be treated in the next chapter.