Differences between generalised procrustes analysis and multiple factor analysis in case of projective mapping

Oliver Tomic

Norwegian University of Life Sciences

Department of Chemistry, Biotechnology and Food Science

Master Thesis 60 credits 2013

NORWEGIAN UNIVERSITY OF LIFE SCIENCES

Universitetet for miljø- og biovitenskap (UMB)

MASTER THESIS 2013

Differences between Generalised Procrustes Analysis and Multiple Factor Analysis in the case of projective mapping

Forskjeller mellom Generalised Procrustes Analysis og Multiple Factor Analysis ved bruk i projective mapping

Oliver Tomic

Ås, December 2013

Department of Chemistry, Biotechnology and Food Science

Norwegian University of Life Sciences

Preface

This master thesis was carried out at the Institute for Chemistry, Biotechnology and Food Science of the Norwegian University of Life Sciences. The master thesis accounts for 60 credits and was carried out in the period from January to December 2013 at Nofima AS.

The master thesis is structured in a different way than regular master theses. It consists mainly of an introduction and a scientific journal paper. The chapter with the scientific journal paper is meant to be self-contained and ready for submission to a scientific journal in the field of sensory and consumer science. Hence, the statistical methods used in this part are referenced as usual and explained only briefly to the reader in order to obey journal specific requirements for the format of the manuscript. What the scientific journal paper chapter lacks in coverage regarding sensory and statistical methods is then elaborated in more detail the introduction.

This thesis was accomplished as a part of my daily work at Nofima where I work as research scientist doing amongst other data analysis on sensory and consumer data. Not very surprising, when working full time as a research scientist and taking part in many different research and client projects it has been a struggle at times to get enough time to prioritise work and writing of the thesis. With my diploma degree in bio process engineering in 1998 in Germany and my Dr. scient. degree in gas-sensor array technology in 2004 here in Norway I felt I had all time I needed to fully concentrate on the writing. This time, however, things were a little bit different. I had to find the right balance between with my daily work, my work with this thesis while trying to spend also quality time with my wife and two small children.

I would like to thank Tormod Næs at Nofima and Trygve Almøy at the Norwegian University of Life Sciences for being my supervisors. Through a number of discussions I got valuable advice regarding the work presented in this thesis. I guess I haven't been a typical master student to them given my situation and I am very thankful that they gave me a lot of freedom while working with this thesis and being patient when progress has been slow. I am also very thankful to Øydis Ueland, who is the research director of the department for sensory and consumer science at Nofima where I work, for letting me attend statistics courses at UMB during work time. I am utmost thankful to my wife Heidi and my children Nina and Thomas for being patient with me during the busy times of my studies in statistics when it was time for exams and the writing of this thesis needed priority. Heidi has done a tremendous job in taking care of the children and keeping our house in order while I was occupied with completing this master. I promise I am done studying now!

Ås, December 2013

Oliver Tomic

Summary

Rapid sensory methods have become very popular in food science and especially the international food industry. Their major appeals are that they are more cost effective and quicker to carry out than some of the traditional sensory methods that are usually applied to get information about the consumer preferences. One method that has gained particular popularity is projective mapping where consumers place a number of products on a sheet or map according to their similarities or dissimilarities. Each consumer can use their own criteria to decide what makes some product similar or not and where to place them on the projective mapping sheet.

In order to get valuable information out of these individual product placings on the projective mapping sheets one needs to apply suitable statistical methods that can handle that type of data. Two methods that have established themselves for analysis of projective mapping data are generalised procrustes analysis and multiple factor analysis. Both of them take quite different approaches to handle and analyse the data, which triggers the question whether results from the two methods will be different or not. In addition, a combination of the two methods has been tested in order to see whether this could provide better results than generalised procrustes analysis and multiple factor analysis by themselves.

This thesis attempts to give some insight into what differences in results there may be by testing out generalised procrustes analysis, multiple factor analysis and the combination of the two on three types of data: random data in Monte Carlo simulations; on constructed or designed data that were manipulated in controlled ways to check what kind of isolated situations the methods can handle or not; on nine sets of real world data where different types of products were tested by varying number of individuals.

Analysis results give no clear answer to which method should be preferred over the other since in some cases generalised procrustes analysis performed better than multiple factor analysis and vice versa. The combination of the two methods gave the least satisfying results.

Sammendrag

Raske sensoriske metoder har blitt veldig populære i matvitenskap og spesielt i internasjonal matindustri. De er appelerende fordi de er kostnadseffektive og raskere å gjennomføre enn noen av de tradisjonelle sensoriske metoder som vanligvis anvendes for å få tak i informasjon om forbrukere. En metode som har blitt spesielt populær er projective mapping hvor hver forbruker plasserer et antall produkter på et ark basert på deres likheter og forskjeller. Hver forbruker bruker sine egne kriterier for å avgjøre hva som gjør at produkter er lignende eller ikke og hvor de skal plasseres på arket.

For å få nyttig informasjon fra disse individuelle produktplasseringene på projective mapping arket trengs det en passende statistisk metode som kan håndtere denne type data. To metoder som har etablert seg for analyse av projective mapping data er generalised procrustes analysis og multiple factor analysis. Begge benytter seg av veldig forskjellige tilnærminger for å håndtere og analysere dataene, noe som gjør at man stiller spørsmål om resultatene fra disse to metodene vil være forskjellige eller ikke. I tilllegg ble det testet en kombinasjon av disse to metodene for aa undersøke om dette kan gi bedre resultater enn generalised procrustes analysis og multiple factor analysis hver for seg.

Denne masteroppgaven prøver å gi noe insikt i hvilke forskjeller i resultatene det kan oppstå ved å teste ut generalised procrustes analysis, multiple factor analysis og kombinasjonen av disse to på tre type data: tilfeldige data i Monte Carlo simuleringer; konstruerte eller desginete data som ble manipulert på kontrollerte måter for å kunne undersøke hva slags isolerte situasjoner metodene kan håndtere; ni reelle datasett hvor forskjellige type produkter ble testet av et varierende antall individer.

Resultatene av analysene ga ingen kart svar om hvilken metode skulle foretrekkes framfor den andre siden i noen tilfeller generalised procrustes analysis fungerte bedre enn multiple factor analysis og omvendt. Kombinasjonen av de to metodene ga minst tilfredstillende resultater.

Contents

Chapter 1

Introduction

Sensory and consumer science - measuring, analysing and interpreting the responses of humans

Sensory and consumer science (Lawless and Heymann, 2010; Lawless, 2013) has been traditionally an integral part in the field of food science, but in the past decade it has been increasingly embraced also by other non-food related fields that involve evaluation of consumer products such as cars, cosmetics, entertainment electronics, services, etc. The aim of sensory and consumer science is to measure, analyse and understand human responses to external stimuli that are perceived by the senses of sight, smell, taste, touch and hearing (Martens and Martens, 2001). Many different sensory methods exist for measuring human responses generating measurement data of various kinds, most of them multivariate. Sensometrics (Næs et al., 2010), a small branch within the field of statistics, is dedicated to the statistical analysis of the sensory and consumer data, providing both univariate and multivariate statistical tools for exploration of the data as well as extraction of important and relevant information. In this regard sensometrics has become an nonexpendable complementary tool to sensory and consumer science and as sensory and consumer science keeps evolving the importance of sensometrics will only increase.

Emergence of new rapid sensory methods

Food industry, just as any other industry, is under increasing pressure to innovate itself and come up with new food products that will immediately appeal the consumer and provide new ways of making profits in a market that traditionally has low margins. At the same time costs for product innovations need to be cut and food industry is compelled to develop and apply new methods that yield more relevant information or data about the consumer at a lower price tag. Therefore, in recent years there has been a strong development of new so-called rapid sensory methods that fulfill these requirements. A number of these rapid sensory methods have gained attention in recent years (Dehlholm et al., 2012; Valentin et al., 2012; Varela and Ares, 2012), but this thesis will focus on only one specific method, which is *projective mapping* (Risvik et al., 1994). Projective mapping generates multiblock data, i.e. multiple data matrices, that require an adequate statistical method for analysis and information extraction. There are a number of multiblock methods available that could be used for analysis of projective mapping data, but the two most commonly used are generalised procrustes analysis (GPA) and multiple factor analysis (MFA).

Aim of the study: which multi-block method should one use?

Even though both MFA and GPA are conceptually very different (see details on each method in section 1.4 and 1.5) both are used regularly for analysis of projective mapping data. To the author's knowledge there exists no study that discusses in detail the differences between the two methods in general, and for projective mapping data in particular. Only one study (Nestrud and Lawless, 2008) briefly mentions that both methods have been tested on the same data set and reports that results were very similar. Without having any facts to prove it, through attendance at conferences and conversations with international colleagues within the field, the author gets the impression that many of those who analyse projective mapping data do not really reflect over which of the two methods they should use. Instead, it seems, they rather use the method they know or they were exposed to first or the one that is provided in the data analysis software they use regularly.

Therefore, in an attempt to give some attention to this issue, the aim of this study is to test both MFA and GPA extensively on projective mapping data and provide some new insight regarding differences between the two methods in the special case of projective mapping.

Limitations of this thesis

The amount of work needed to compare two statistical methods extensively is considerable. In order keep the volume of this thesis within limits some restrictions needed to be made. Main focus in this thesis has therefore been the similarity of the consensus configurations acquired by GPA and MFA and uncovering potential differences that might influence interpretation of the product related information.

1.1 Projective Mapping

Development of new food products is a very challenging task considering that it is a costly and time consuming process and the fact that most new products do not survive on the market for very long. It is without doubt the consumers who decide whether a new product will be a failure or a success. It is therefore of utmost importance for the product developer to get insight into consumer's preferences, i.e. finding out which products he or she likes or dislikes, and actively use this information in the further product development process.

Traditionally, consumer tests are the main tool to learn more about the consumers' preferences. Consumer tests are carried out in such a way that each consumer is served a number of products where he or she rates each product on a 5-, 7- or 9-point scale (Lawless and Heymann, 2010). As an example, a typical 5 point scale would be distributed like this: 1 - dislike the product very much; 2 - dislike the product; 3 - neither like nor dislike the product; 4 - like the product; 5 - like the product very much. With this type of data the product developer gains some insight into which products the consumers prefer the most or the least, but no information is provided explaining why they prefer one product over another. This information, the knowledge about the drivers of liking of a product, needs to be acquired in another way. The standard sensory method

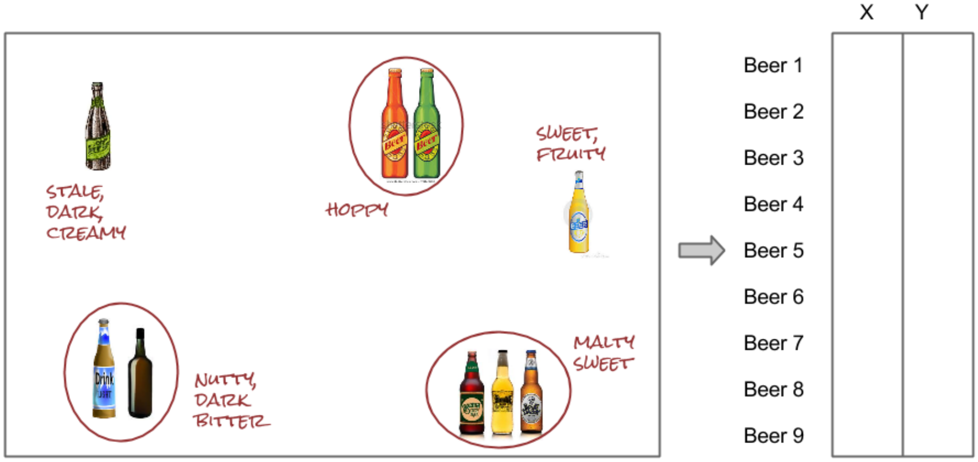

Figure 1.1: Example of a projective mapping sheet where one user placed 9 brands of beer according to own criteria. This is the *individual product configuration* of this user and describes how the products relate to one another. The data for each individual are created from the product coordinates, as illustrated by the figure.

to get this information is called preference mapping (Lawless and Heymann, 2010) and requires that a trained sensory panel provides a descriptive profile of the tested products and that their data is linked to the consumer ratings by use of a suitable statistical method. The problem is, however, that using a trained sensory panels is costly and that the food industry with their low profit margins are rather reluctant to use them in their product development. So the problem is the following: how can the food industry get trained sensory panel type of information at a lower price?

One option is to use projective mapping (Risvik et al., 1994, 1997), a sensory method that was developed in the early nineties at Nofima (called Matforsk at that time) and that aims to have untrained consumers do the profiling of the products. In projective mapping a number of individuals place the tested products on a sheet of paper according to their similarities and dissimilarities. Products that are similar are placed close to one another and products that are very different are placed far from one another. See Fig. 1.1 for an example of how one individual placed nine beers on a projective mapping sheet according to some sensory criteria.

It is important to note that the individuals participating in the test use their own criteria for how products are placed on the sheet. The criteria may vary somewhat from individual to individual since consumers are known to perceive sensory sensations quite differently (Lawless and Heymann, 2010). An important question a product developer could ask is: what is the overall perception of the products across all individuals? Note that the overall perception may also be called *consensus product configuration* as opposed to the individual product configuration from one person as displayed in Fig. 1.1. Other important questions a product developer might ask: how are individuals different from the consensus?; how should the product placing of all individuals be analysed? To get answers to these questions one needs to convert individual product placing into quantitative data. Here, the first step is to record the positions of each product on the sheet, i.e. their x- and y-coordinates, and collect them in own data tables or blocks as illustrated by Fig. 1.1. Then these data blocks need to be analysed with a suitable statistical method such as GPA and MFA, whose analysis results will provide answers to the questions above.

1.2 Notations and Preliminaries

Before the statistical methods used in this master thesis are introduced in the following sections some notations and preliminaries need to be considered. *Generalised Procrustes Analysis* and *Multiple Factor Analysis*, the two main statistical methods compared in this study, are both so-called multi-block methods. In general, both handle $k = 1, \ldots, K$ data matrices (also referred to as *blocks* or *individual configurations* from now on in this thesis), where each block $\mathbf{X}_{[k]}$ is of dimension $I \times J_{[k]}$. This means that the objects I are the common axis across all blocks $\mathbf{X}_{[k]}$ and that the number of variables $J_{[k]}$ may vary across the blocks. Thus, the total number variables across all blocks is $J = \sum J_{[k]}$. Moreover, blocks $\mathbf{X}_{[k]}$ may be concatenated horizontally into larger blocks $\mathbf{X}$ of dimension $I \times J$ formalised as follows:

$$\mathbf{X} = \left[\mathbf{X}_{[1]} | \ldots | \mathbf{X}_{[k]} | \ldots | \mathbf{X}_{[K]}\right]. \tag{1.1}$$

Note that the data collected from projective mapping tests represent a special case of multi-block data where the number of variables in each block $\mathbf{X}_{[k]}$ is exactly two. Hence, for the typical projective mapping data there are a total of K blocks $\mathbf{X}_{[k]}$, one for each individual who participated in the test, where all blocks are of dimension $(I \times 2)$.

1.3 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) (Martens and Næs, 1989; Abdi and Williams, 2010) is a multivariate statistical method and standard tool for explorative analysis of a single data block **X**. The main goals of PCA are:

- to find and summarise systematic variation or patterns in the data
- to investigate the object configuration, i.e. to visualise how matrix objects or rows relate to one another based on the variance from the measured variables
- to understand how variables in **X** are correlated to one another and how much each variable contributes to the variance in the data
- to reduce dimensionality, i.e. keep important information in a few latent variables instead of all variables in **X**
- to separate noise from information.

There are several ways of doing PCA for a data block **X**, but in this study focus will be on singular value decomposition (SVD) and generalised singular value decomposition (GSVD). Section 1.3.1 and 1.3.2 describe in detail how factors or principal components are acquired with SVD and GSVD, respectively, how the amount of variance they explain is computed as well as how factor or PCA scores, factor or PCA loadings are obtained. Even though SVD and GSVD appear to be very similar at first glance there are some essential differences. It is therefore important to describe them separately. Both approaches are integral parts of separate computation steps in MFA (see details in 1.4). An excellent and detailed review of PCA based on SVD is provided in a recent paper (Abdi and Williams, 2010) and this study provides a brief summary of the most important features used in this study.

1.3.1 Singular Value Decomposition (SVD)

When applying singular value decomposition to a matrix $\mathbf{X}$ of dimension $(I \times J)$ it is decomposed the following way:

$$\mathbf{X} = \mathbf{U}\Gamma\mathbf{V}^T \tag{1.2}$$

where $\mathbf{U}$ is an orthonormal matrix of dimension $(I \times L)$ holding normalised left singular vectors and L is the rank of matrix $\mathbf{X}$; Γ is a diagonal matrix of L singular values with $[\gamma_1, \gamma_2, \ldots, \gamma_L] = diag\{\Gamma\}$ with $\ell = 1, \ldots, L$; $\mathbf{V}$ is an orthonormal matrix of dimension $(J \times L)$ holding normalised right singular vectors. Since both $\mathbf{U}$ and $\mathbf{V}$ are orthonormal, it is true that

$$\mathbf{U}^T\mathbf{U} = \mathbf{V}^T\mathbf{V} = \mathbf{I} \tag{1.3}$$

where each column $\mathbf{u}_\ell$ in $\mathbf{U}$ and $\mathbf{v}_\ell$ in $\mathbf{V}$ represent the ℓth *principal component* (PC) in PCA. The *first* PC, typically abbreviated PC1, finds the direction of the largest variance in the multivariate space spanned by the variables of $\mathbf{X}$. The *second* PC, i.e. PC2, is orthogonal on PC1 and takes the direction of the second largest variation, etc.

Factor or PCA scores

Information on the measured objects of $\mathbf{X}$ is provided by the *factor scores* or so-called *PCA scores*. They are obtained by

$$\mathbf{G} = \mathbf{U}\Gamma \tag{1.4}$$

with $\mathbf{G}$ being of dimension $(I \times L)$. When plotting two columns of $\mathbf{G}$ in a scatter plot one gets a PCA scores plot that visualises the relation between the objects for these two specific principal components. If two objects are located close to each other in the PCA scores plot they are very similar within the space spanned by the two PC's and vice versa.

Factor or PCA loadings

Information on how the variables of $\mathbf{X}$ contribute to the variance in the data can be gained from

V, the *factor loadings* or so-called *PCA loadings*. When plotting two columns of **V** in a scatter plot one gets a PCA loadings plot that visualises the relation between the variables for these two specific principal components. If located close to each other two variables are highly correlated for the selected PC's and vice versa. Furthermore from the PCA loadings plot one can also visualise how much the variables contribute to the variation relative to each other for the two specific PC's. The further away from the origo, the more a variable contributes to variation for those PC's.

Explained variances

The amount of *calibrated explained variance* by the ℓth PC can be acquired from the the squared singular values γ_ℓ^2 (Wall et al., 2003) by computing:

$$explVar_\ell = \frac{\gamma_\ell^2}{\sum_{\ell=1}^{L} \gamma_\ell^2} \tag{1.5}$$

which provides a measure for how much systematic variance in the data has been captured by each PC. Note that the calibrated explained variances are non-negative and that their sum across all L PC's equals 1. *Validated explained variances* are discussed in section 1.6.1.

1.3.2 Generalised Singular Value Decomposition (GSVD)

Generalised singular valued decomposition (GSVD) is a generalisation of SVD that incorporates constraints to the singular vectors of **X** under the decomposition. When applying GSVD to a matrix **X** of dimension (IxJ) it is decomposed the following way:

$$\mathbf{X} = \mathbf{P}\Delta\mathbf{Q}^T \tag{1.6}$$

where **P** is a matrix of dimension $(I \times L)$ holding normalised left generalised singular vectors and L is the rank of matrix **X**; Δ is a diagonal matrix of L generalised singular values such that $[\delta_1, \delta_2, \ldots, \delta_L] = diag\{\Delta\}$; **Q** is an matrix of dimension $(J \times L)$ holding normalised right generalised singular vectors. Now similarly to Eq. 1.3 as in SVD the following is given for GSVD:

$$\mathbf{P}^T\mathbf{M}\mathbf{P} = \mathbf{Q}^T\mathbf{A}\mathbf{Q} = \mathbf{I}. \tag{1.7}$$

Here, however, two new matrices are included, i.e. **M** and **A** that implement some metric on the rows and columns of **X**, respectively, that are incorporated into the decomposition of **X**. **M** is a positive definite matrix of dimension $(I \times I)$ which is almost always diagonal (Abdi et al., 2013) and where each diagonal element holds a mass m_i for the corresponding object or observation i in **X**. The masses m_i are non-negative and their sum equals one. By changing the values of the masses m_i it is possible to assign an individual mass or "weight" to each object and influence its importance in the decomposition. This is, however, rarely of interest and most often all masses are set to $m_i = \frac{1}{I}$. **A** is also a positive definite matrix of dimension $(J \times J)$ that is often, but now always diagonal (Abdi et al., 2013). **A** may contain weights for the columns or variables in **X** allowing for weighting of those as part of the decomposition. Multiple Factor Analysis (see section 1.4) for example makes use of this feature when analysing multiple blocks of data. Similar to SVD in Eq. 1.3 column $\mathbf{p}_\ell$ in **P** and $\mathbf{q}_\ell$ in **Q** in Eq. 1.7 represent the ℓth *principal component* (PC) in PCA. Note that when $\mathbf{M} = \mathbf{A} = \mathbf{I}$ GSVD reduces to SVD.

Factor or PCA scores

Each column in **P** and **Q** represents a *principal component* in PCA. Information on the measured objects of **X** is provided by the *factor scores* or so-called *PCA scores*. They are obtained by

$$\mathbf{F} = \mathbf{P}\Delta. \tag{1.8}$$

Factor or PCA loadings

Information on how the variables of **X** contribute to the variance in the data can be gained from **Q**, the *factor loadings* or so-called *PCA loadings*.

Explained variances

In GSVD the amount of *calibrated explained variance* by the ℓth PC can be acquired from the the squared singular values δ_ℓ^2 by computing:

$$explVar_\ell = \frac{\delta_\ell^2}{\sum_{\ell=1}^{L} \delta_\ell^2} \tag{1.9}$$

Note that the calibrated explained variances are non-negative and that their sum across all L

PC's equals 1. *Validated explained variances* are discussed in 1.6.1.

1.4 Multiple Factor Analysis (MFA)

An excellent and very detailed review of MFA is given in a recent paper (Abdi et al., 2013) and this thesis provides a brief summary of the most important features of MFA used for analysis of the data described below in section 2.2.2.

Multiple Factor Analysis (Escofier and Pagès, 1994) was developed for analysis of multiple tables or blocks $\mathbf{X}_{[k]}$ that hold data from various measurements on the same objects I. Since these measurements may be of different origin (e.g. using different types of instruments to measure objects I) the blocks are of dimension $(I \times J_{[k]})$ where each bock can consist of a unique numbers of variables $J_{[k]}$. As a special case all $\mathbf{X}_{[k]}$ may consist of an equal number of variables by measuring the same objects with the same instrument at different points of time. MFA assumes all block $\mathbf{X}_{[k]}$ to be standardised, i.e. the variables $J_{[k]}$ have zero mean and a standard deviation that equals one. More concretely, the original x- and y-coordinates from projective mapping were stored in individual matrices $\mathbf{Z}_{[k]}$ (see Fig. 1.1) and their standardised values are stored in $\mathbf{X}_{[k]}$, which is then used by MFA. MFA can be described as a two-step procedure.

STEP 1

First each block $\mathbf{X}_{[k]}$ is decomposed using SVD as described in Eq. 1.2 which results in:

$$\mathbf{X}_{[k]} = \mathbf{U}_{[\mathbf{k}]} \Gamma_{[\mathbf{k}]} \mathbf{V}_{[k]}^{T} \tag{1.10}$$

with

$$\mathbf{U}_{[k]}^{T} \mathbf{U}_{[k]} = \mathbf{V}_{[k]}^{T} \mathbf{V}_{[k]} = \mathbf{I}_{[k]} \tag{1.11}$$

where the left and right singular vectors of $\mathbf{X}_{[k]}$ are stored in $\mathbf{U}_{[k]}$ and $\mathbf{V}_{[k]}$, respectively. The singular values $\gamma_{\ell,k}$ are stored in $\Gamma_{[k]}$ such that

$$[\gamma_{1,k}, \gamma_{2,k}, \ldots, \gamma_{\ell,k}, \ldots, \gamma_{L,k}] = diag\{\Gamma_{[k]}\}. \tag{1.12}$$

The main objective of *STEP 1* is to compute weights for each block $\mathbf{X}_{[k]}$ which are then used in *STEP 2* for normalisation of each $\mathbf{X}_{[k]}$. The weight α_k for a block $\mathbf{X}_{[k]}$ is computed from its first singular value acquired from SVD in the following way:

$$\alpha_k = \frac{1}{\gamma_{1,k}^2} = \gamma_{1,k}^{-2}. \tag{1.13}$$

Now all weights α_k can be collected in a vector $\mathbf{a}$ of dimension $(J \times 1)$ such that each variable in a block $\mathbf{X}_{[k]}$ is assigned to its corresponding weight α_k. This is summarised in

$$\mathbf{a} = [\alpha_1 \mathbf{1}_{[1]}^T, \ldots, \alpha_k \mathbf{1}_{[k]}^T, \ldots, \alpha_K \mathbf{1}_{[K]}^T] \tag{1.14}$$

where $\mathbf{1}_{[k]}$ is a vector of ones representing the $J_{[k]}$ variables in each block $\mathbf{X}_{[k]}$. Eventually, the weights can be stored in a diagonal matrix $\mathbf{A}$ of dimension $(J \times J)$

$$\mathbf{A} = diag\{\mathbf{a}\} = diag\{[\alpha_1 \mathbf{1}_{[1]}^T, \ldots, \alpha_k \mathbf{1}_{[k]}^T, \ldots, \alpha_K \mathbf{1}_{[K]}^T]\}. \tag{1.15}$$

This matrix $\mathbf{A}$ can now be used in *STEP 2* in GSVD as described in Eq. 1.7.

STEP 2

First all blocks $\mathbf{X}_{[k]}$ are concatenated horizontally into a grand matrix $\mathbf{X}$

$$\mathbf{X} = \left[\mathbf{X}_{[1]} | \ldots | \mathbf{X}_{[k]} | \ldots | \mathbf{X}_{[K]}\right] \tag{1.16}$$

then GSVD is applied to $\mathbf{X}$ according to Eq. 1.6. Now matrix $\mathbf{A}$ from *STEP 1* is used in Eq. 1.7 to impose MFA related constraints to the variables in grand matrix $\mathbf{X}$, i.e. weighting each block $\mathbf{X}_{[k]}$ by its corresponding weight α_k as part of the decomposition. Using Eq. 1.6 and Eq. 1.8 the GSVD decomposition can be rewritten as

$$\mathbf{X} = \mathbf{F}_{MFA} \mathbf{A} \mathbf{Q}_{MFA}^T. \tag{1.17}$$

now, however, based on PCA scores $\mathbf{F}_{MFA}$ and PCA loadings $\mathbf{Q}_{MFA}$. The PCA scores $\mathbf{F}_{MFA}$ are used to visualise the *compromise* or *consensus configuration*, i.e. the configuration of the objects based on information from all $\mathbf{X}_{[k]}$. Eq. 1.17 can be rearranged into

$$\mathbf{F}_{MFA} = \mathbf{XAQ}_{MFA} \tag{1.18}$$

and $\mathbf{Q}_{MFA}$ can be re-expressed as

$$\mathbf{Q}_{MFA} = \begin{bmatrix} \mathbf{Q}_{[1]} \\ \vdots \\ \mathbf{Q}_{[k]} \\ \vdots \\ \mathbf{Q}_{[K]} \end{bmatrix} = \left[\mathbf{Q}^T_{[1]}, \ldots, \mathbf{Q}^T_{[k]}, \ldots, \mathbf{Q}^T_{[K]}\right] \tag{1.19}$$

where each $\mathbf{Q}_{[k]}$ is a matrix of dimension $(J_{[k]} \times L)$ holding the right singular vectors corresponding to the variables of $\mathbf{X}_{[k]}$ and where L is the rank of the grand matrix $\mathbf{X}$. By using Eq. 1.19 one can rewrite Eq. 1.18 the following way:

$$\begin{aligned} \mathbf{F}_{MFA} &= \mathbf{XAQ}_{MFA} \\ &= \left[\mathbf{X}_{[1]} | \ldots, |\mathbf{X}_{[k]}| \ldots, |\mathbf{X}_{[K]}\right] \times \mathbf{A} \times \begin{bmatrix} \mathbf{Q}_{[1]} \\ \vdots \\ \mathbf{Q}_{[k]} \\ \vdots \\ \mathbf{Q}_{[K]} \end{bmatrix} \\ &= \textstyle\sum_k \mathbf{X}_{[k]}\mathbf{A}_{[k]}\mathbf{Q}_{[k]} \\ &= \textstyle\sum_k \alpha_k \mathbf{X}_{[k]}\mathbf{Q}_{[k]}. \end{aligned} \tag{1.20}$$

Now based on Eq. 1.20 the *partial factor scores* $\mathbf{F}_{[k]}$ for each $\mathbf{X}_{[k]}$are computed as follows:

$$\mathbf{F}_{[k]} = K \times \alpha_k \times \mathbf{X}_{[k]}\mathbf{Q}_{[k]}. \tag{1.21}$$

These scores are projections of the individual configuration of a block $\mathbf{X}_{[k]}$ into the space of the compromise or consensus configuration. By having both projected scores $\mathbf{F}_{[k]}$ and and consensus scores $\mathbf{F}_{MFA}$ in the same space one can create PCA scores plots to visualise differences between the consensus configuration and individual configurations and to compute the relative distance between them as it is done with the similarity ratios SR_k discussed in section 2.2.3.5.

Fig. 1.2 shows a graphical illustration of the MFA computation process that is slightly different from the one that is described above. Both approaches obviously produce the same statistical results. In section 2.2.3.2 of the scientific paper a third alternative is presented for computing MFA results. Also this one produces the same results as the two prior mentioned.

1.5 Generalised Procrustes Analysis (GPA)

Generalised procrustes analysis (Gower, 75) is the other multiblock method discussed in this thesis. Similar to MFA, GPA was designed for analysis of K blocks $\mathbf{Z}_{[k]}$ of dimension $(I \times J_{[k]})$ that hold data from measurements on the same objects I. In the same way as MFA, also here the goal is to find a consensus configurations $\mathbf{Y}$ and their PCA scores $\mathbf{F}_{GPA}$ based on all individual configurations $\mathbf{Z}_{[k]}$, PCA loadings $\mathbf{Q}_{GPA}$ and projected scores $\mathbf{F}_{[k]}$ of the individual configurations into the space of the consensus configuration. GPA, though, takes a quite different statistical approach to compute these results compared to MFA. In brief, the main steps of GPA are the so-called Procrustes transformation of blocks $\mathbf{Z}_{[k]}$ to make them all as similar as possible before PCA is applied on their average matrix. An excellent and detailed review of GPA in sensory science context is given in a book chapter (Dijksterhuis, 1996) and this thesis provides a brief summary of the content.

STEP 1 - Procrustes transformation

The GPA procedure starts with Procrustes transformation of the blocks $\mathbf{Z}_{[k]}$ to make them as alike as possible. The Procrustes transformation itself consist of several data transformations: (I) translation; (II) rotation and reflection; (III) isotropic scaling. Fig. 1.3 illustrates how these three transformations are carried out exemplified by two simple configurations with three objects.

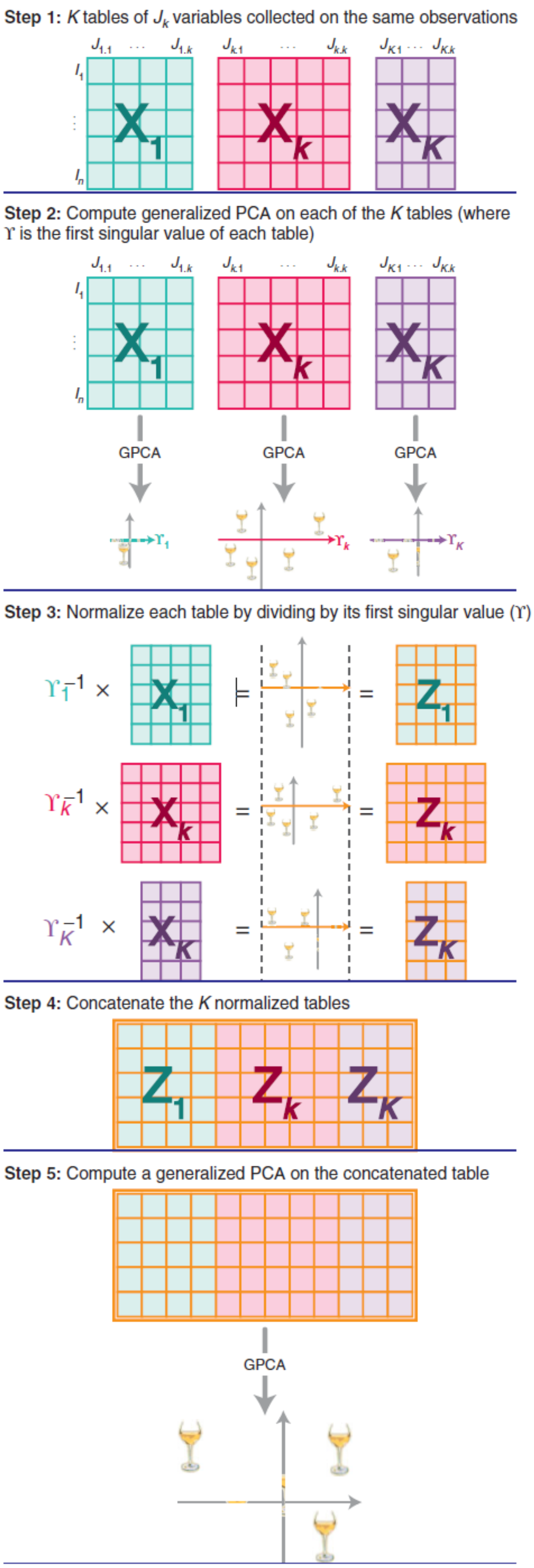
Step 1: K tables of J_k variables collected on the same observations
$J_{1.1}$ ··· $J_{1.k}$
$J_{k.1}$ ··· $J_{k.k}$
$J_{K.1}$ ··· $J_{K.k}$
I_1
I_n
X_1
X_k
X_K
Step 2: Compute generalized PCA on each of the K tables (where ϒ is the first singular value of each table)
GPCA
GPCA
GPCA
$ϒ_1$
$ϒ_k$
$ϒ_K$
Step 3: Normalize each table by dividing by its first singular value (ϒ)
$ϒ_1^{-1}$ × X_1 = = Z_1
$ϒ_k^{-1}$ × X_k = = Z_k
$ϒ_K^{-1}$ × X_K = = Z_K
Step 4: Concatenate the K normalized tables
Z_1
Z_k
Z_K
Step 5: Compute a generalized PCA on the concatenated table
GPCA

Figure 1.2: A graphical illustration of the MFA computation process (taken from (Abdi et al., 2013)).

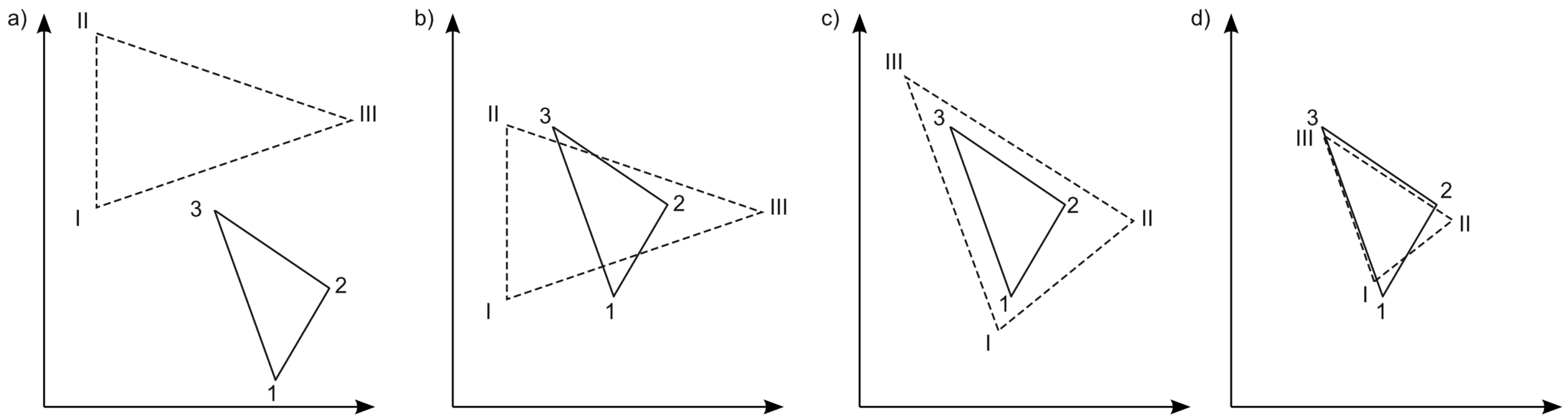

Figure 1.3: A graphical illustration of Procrustes transformation attempting to make two configurations as similar as possible using three transformation steps: *a)* original configurations before transformation; *b)* after translation; *c)* after rotation and reflection; *d)* after isotropic scaling.

In statistical terms the procedure is described by the following. After all three transformation steps are carried out the distances between the transformed individual configurations of $\mathbf{Z}_{[k]}$ can be written as

$$\sum_{k<l}^{K} \|\tau(\mathbf{Z}_{[k]}) - \tau(\mathbf{Z}_{[l]})\| \tag{1.22}$$

where $\tau(\mathbf{Z}_{[k]})$ represents the transformed version of block $\mathbf{Z}_{[k]}$. One important restriction of the Procrustes rotation is that it keeps intact the relative distances between the objects within a configuration. As it will be shown later, MFA does not preserve these relative distances between objects (see section 2.3.2). Minimising Eq. 1.22 is equivalent to

$$\sum_{k=1}^{K} \|\tau(\mathbf{Z}_{[k]}) - \mathbf{Y}\| \tag{1.23}$$

where $\mathbf{Y}$ is the consensus matrix computed as the average across all transformed blocks $\tau(\mathbf{Z}_{[k]})$ as described by

$$\mathbf{Y} = K^{-1} \sum_{k=1}^{K} \tau(\mathbf{Z}_{[k]}). \tag{1.24}$$

The three transformations steps in Procrustes transformation of a block $\mathbf{Z}_{[k]}$ can be summarised in

$$\tau(\mathbf{Z}_{[k]}) = \rho_k \mathbf{Z}_{[k]} \mathbf{H}_{[k]} + \mathbf{T}_{[k]} \tag{1.25}$$

where $\mathbf{T}_{[k]}$ represents the *translation step* (going from *a)* to *b)* in Fig. 1.3). Practically, this means that the variables in $\mathbf{Z}_{[k]}$ are zero mean centered. The *rotation and reflection* transformation (going from *b)* to *c)* in Fig. 1.3) is represented by the orthonormal rotation matrix $\mathbf{H}_{[k]}$ which is of dimension $(J \times J)$. Since $\mathbf{H}_{[k]}$ is orthonormal it is true that

$$\mathbf{H}_{[k]}^T \mathbf{H}_{[k]} = \mathbf{H}_{[k]} \mathbf{H}_{[k]}^T = \mathbf{I}. \tag{1.26}$$

The last step in Procrustes transformation, i.e. *isotropic scaling* (going from *c)* to *d)* in Fig. 1.3), is represented by the isotropic scaling factor ρ_k, which is a non-negative scalar. These three steps in Procrustes transformation minimise the sum of all squared distances between the transformed blocks $\tau(\mathbf{Z}_{[k]})$ such that Eq. 1.23 can be reformulated as

$$\sum_{k<l}^{K} \|\rho_k \mathbf{Z}_{[k]} \mathbf{H}_{[k]} - \rho_l \mathbf{Z}_{[l]} \mathbf{H}_{[l]}\| = K \sum_{k=1}^{K} \|\mathbf{Y} - \rho_k \mathbf{Z}_{[k]} \mathbf{H}_{[k]}\|. \tag{1.27}$$

In order to prevent a trivial solution where ρ_k minimises to zero the following constraint is imposed on Eq. 1.27

$$\sum_{k=1}^{K} \|\rho_k \mathbf{Z}_{[k]} \mathbf{H}_{[k]}\| = K \tag{1.28}$$

where the constraint scales the total variance to K, i.e. the total number of blocks $\mathbf{Z}_{[k]}$ included in the computation.

Through the Procrustes transformation the total variance across the individual configurations $\mathbf{Z}_{[k]}$ are minimised. In order to evaluate the contribution of each transformation to the reduction of the total variance a *PANOVA* (Procrustes ANOVA) table is computed (Gower, 75) using approximate F-tests for each of the three transformations. Table 1.1 shows an example of a PANOVA table produced from the commercial XLSTAT software for real world data set 3 which is described in section 2.2.2.4. From this example it can be observed that the largest reduction in variance is due to the rotation transformation (p < 0.0001). Reduction in variance through translation is considerable (p = 0.034), however, it is about 4.6 times smaller than for rotation. Isotropic scaling contributes the least to variance reduction (p = 0.540).

Table 1.1: Example of PANOVA table generated by the statistical software XLSTAT.

Source	DF	Sum of squares	Mean squares	F	Pr > F
Residuals after scaling	204	17711.084	86.819		
Scaling	17	1372.461	80.733	0.930	0.540
Residuals after rotation	221	19083.545	86.351		
Rotation	17	21137.916	1243.407	14.322	< 0.0001
Residuals after translation	238	40221.461	168.998		
Translation	34	4580.865	134.731	1.552	0.034
Corrected Total	272	44802.326	164.714		

STEP 2 - PCA on consensus matrix $\mathbf{Y}$

The second main step in GPA is to apply PCA to the average matrix $\mathbf{Y}$ from Eq. 1.24 by means of SVD as described in Eq. 1.2. Note that since $\mathbf{Y}$ is an average across K individual blocks $\mathbf{Z}_{[k]}$, $\mathbf{Y}$ is of the same dimension as the blocks $\mathbf{Z}_{[k]}$, i.e. $(I \times J_{[k]})$. With this $\mathbf{Y}$ is decomposed the following way

$$\mathbf{Y} = \mathbf{U}\Gamma\mathbf{V}^T. \tag{1.29}$$

From this the PCA scores of the consensus configuration $\mathbf{F}_{GPA}$ can be computed by

$$\mathbf{F}_{GPA} = \mathbf{G} = \mathbf{U}\Gamma \tag{1.30}$$

and the PCA loadings $\mathbf{Q}_{GPA}$ are represented by $\mathbf{V}$ in Eq. 1.29. Finally, the projected PCA scores $\mathbf{F}_{k,GPA}$ of the individual blocks $\mathbf{X}_{[k]}$ into the consensus space of $\mathbf{F}_{GPA}$ can be computed by

$$\mathbf{F}_{k,GPA} = \tau(\mathbf{Z}_{[k]})\mathbf{Q}_{GPA}. \tag{1.31}$$

By having both projected scores $\mathbf{F}_{[k]}$ and and consensus scores $\mathbf{F}_{GPA}$ in the same space one can create PCA scores plots to visualise differences between the consensus configuration and individual configurations and to compute the relative distance between them as it is done with the similarity ratios SR_k discussed in section 2.2.3.5.

1.6 Validation

The PCA scores $\mathbf{F}_{GPA}$ and $\mathbf{F}_{MFA}$ hold the consensus information on the tested products computed with GPA and MFA, respectively. By plotting the first two columns of either $\mathbf{F}_{GPA}$ or $\mathbf{F}_{MFA}$ in a scatter plot usually most of the systematic variance in the data is visualised, thus providing the product developer valuable insight into how the products relate to one another. This information is then used for further development or refinements of the products. The question now is whether the acquired statistical results are really valid and not only a random outcome? Can the product developer really trust them enough to base his or her decisions on them regarding the further development of a new product? Therefore, it is necessary to apply some suitable validation tools as part of the computations with GPA and MFA

1.6.1 Cross validation in PCA

Since both GPA and MFA do a PCA as the last step of computations obvious tools for validation would be *test set validation* and *cross validation* (Martens and Næs, 1989). With test set validation one needs to create a training set and a test set. The training set then would consists of the products tested by consumers and would be used to build or train the PCA model from. The test set should contain new untested products which would then be used to validate the PCA model. This approach is not feasible when working with consumers, since consumers would have to participate in a second trial testing the new products. This would be very costly and too time consuming for the product development process. The more suitable option would be to apply cross validation (sometimes also called *internal validation*), i.e. only the products tested by the consumers are used for training and testing the PCA model.

When cross validating a PCA model that was built from all I objects in a matrix $\mathbf{X}$ of dimension $(I \times J)$, a number N objects are left out and then a new PCA model is calibrated with the remaining $I-N$ objects. N may consist of one or more objects and making the right choice for which samples should be left out each time depends on the relationship between the objects. The matrix of left out objects can be denoted $\mathbf{X}_{out}$ and is of dimension $(N \times J)$. The N objects that were left out are then projected into that new PCA model space based on $I-N$ objects and PCA scores are obtained for those N left-out objects. These projected PCA scores are then used to predict or reconstruct the matrix $\mathbf{X}_{out}$. This reconstructed matrix is named $\widehat{\mathbf{X}}_{out}^{[\ell]}$ and is computed using the first ℓ PC's. The whole process is then repeated a number of times until each object has been left out once. Note that if the number of left out objects is $N=1$, it means that there are I such repetitions. The cross validation then becomes a full cross validation, which is also known as *leave-one-out* cross validation. Eventually, when all repetitions are done the $\widehat{\mathbf{X}}_{out}^{[\ell]}$ from all repetitions can be utilised to construct $\widehat{\mathbf{X}}$ and compute PRESS (PRediction Error Sum of Squares), also called residuals

$$PRESS = \|\mathbf{X} - \widehat{\mathbf{X}}\|^2. \tag{1.32}$$

From $PRESS$ the mean square error of cross validation (MSECV) can be acquired by computing $MSECV = PRESS/I$ for each PC. This $MSECV$ can then be used to compute the *validated explained variance* (Martens and Næs, 1989), a measure that provides information on the validity or robustness of the PCA model based on all I objects. In general terms, the closer the validated explained variance to the calibrated explained variance (see Eq. 1.5 and 1.9 for SVD and GSVD, respectively), the more robust the PCA model.

In case of the projective mapping data, which are the main focus of this thesis, full cross validation was used in MFA for where the grand matrix $\mathbf{X}$ from Eq. 1.16 is decomposed by GSVD. For GPA, full cross validation was used where projective mapping data the consensus matrix $\mathbf{Y}$ (see Eq. 1.24) was decomposed by SVD.

1.6.2 Permutation testing for Procrustes transformation consensus

As part of the GPA process all individual matrices $\mathbf{Z}_{[k]}$ are transformed using Procrustes transformations (see section 1.5 and Fig. 1.3) and a consensus matrix $\mathbf{Y}$ is acquired by averaging across these transformed individual configurations $\tau(\mathbf{Z}_k)$ (see Eq. 1.24). But this consensus matrix or consensus configuration $\mathbf{Y}$ also needs to be subjected to validation in order to determine whether the consensus is real or if it a product of chance. Earlier research has shown that even by using random numbers for individual configurations $\mathbf{Z}_{[k]}$ GPA may obtain a consensus that is in close agreement with the individual configurations (King and Arents, 1991). The authors suggest that the level of agreement can be measured with the R_c statistic which represents the proportion of the total variance explained by the consensus $\mathbf{Y}$. This R_c statistic may then be utilised in a permutation test (Wakeling et al., 1992) to determine the validity of the consensus.

In the permutation test the rows in each individual matrix $\mathbf{Z}_{[k]}$ are permuted before Procrustes transformations are applied to the permuted data and a new consensus $\mathbf{Y}_{perm}$ is computed. In this way all information regarding the objects or products is lost, yet the original configuration of the data points is preserved. By running a Monte Carlo simulation and doing these permutations a high number of times (e.g. 10 000 times) a distribution of R_c is found. By comparing for example the 95th percentile of this distribution with the R_c of the original data it is

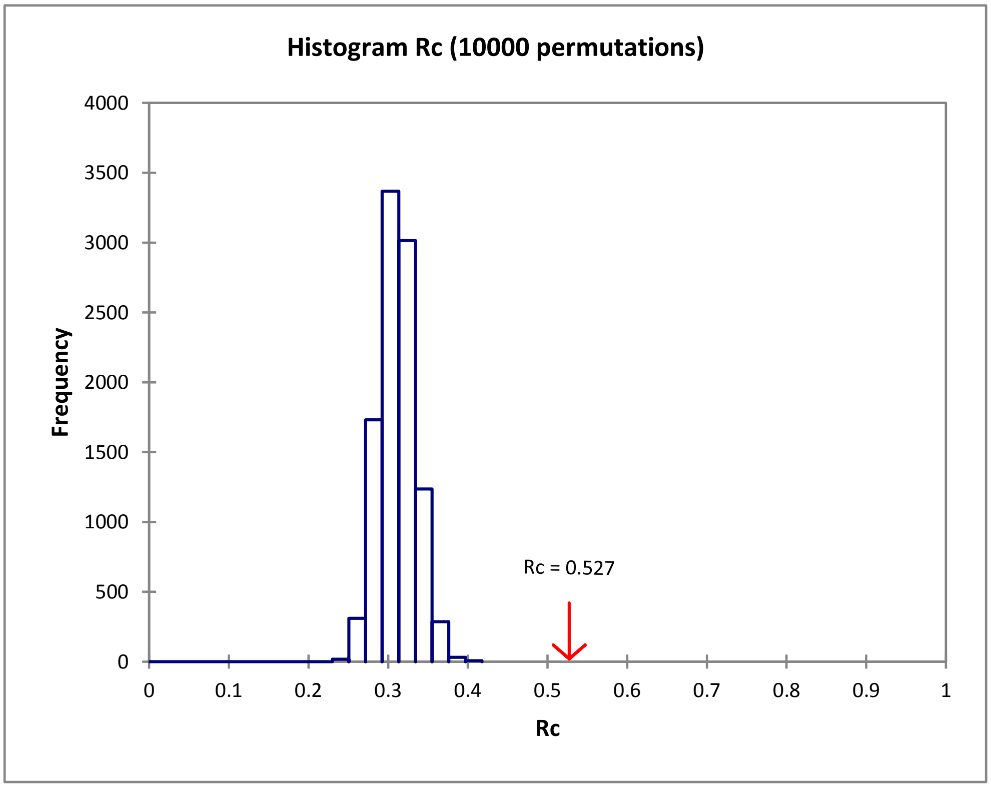

Figure 1.4: Histogram showing the R_c distribution from 1000 permutations. The R_c value of the original data is 0.527 which is higher than any of the R_c values from permutation.

possible to determine whether the consensus of the original data has been acquired by chance or not. Fig. 1.4 shows an example of such an permutation test with 10000 permutations for the real world data set 3. The distribution suggests that the consensus matrix **Y** is not acquired by chance and that the reduction in variance by Procrustes transformation is significant.

Chapter 2

Scientific Journal Paper

Abstract

Generalised procrustes analysis and multiple factor analysis are multivariate statistical methods that belong to the family of multiblock methods. Both methods are often used for analysis of data from projective mapping (a.k.a. Napping®), a method from sensory science where untrained individuals are asked to place products on a sheet according to product similarities or dissimilarities. In this study generalized procrustes analysis, multiple factor analysis as well as a combination of the two were applied on a number of data sets and their statistical results were compared with one another. The type of data used in this study were (I) random data from Monte Carlo simulations; (II) constructed data that were manipulated according to different criteria; (III) real world data from nine Napping® experiments. Results have shown that in some cases due to somewhat different outcomes interpretation problems may arise and that the user faces a difficult decision where one statistical method should be preferred over the other. Some simple guidelines are suggested that may help making this decision.

Keywords: projective mapping, Napping®, generalized procrustes analysis, GPA, multiple factor analysis, MFA, consumer test, combination GPA MFA

2.1 Introduction

In recent years rapid sensory methods have gained a lot of interest in the field of sensory science (Dehlholm et al., 2012; Valentin et al., 2012; Varela and Ares, 2012). An important method that belongs to this category is projective mapping (Risvik et al., 1994), also known as Napping (Pagès, 2005), where untrained individuals are asked to place a number of products on a two-dimensional sheet according to their similarities or dissimilarities using their own criteria (i.e. different types of sensory sensations) they consider to be important. Despite being documented to be less precise than descriptive sensory analysis (Valentin et al., 2012) projective mapping has gained much popularity especially with the food industry because it is less time consuming and more cost effective than traditional methods.

By placing products on a sheet each individual generates a two-dimensional data matrix holding the coordinates of all placed products. Projective mapping is carried out with a number of individuals typically ranging from 10 to 100, meaning that a multiblock or multimatrix situation is given after all data are collected. These data need to be analysed with a suitable statistical method where analysis results will provide important information about the tested products and where this information can be utilised for further product development or product optimisation.

For the analysis of projective mapping data the two most established statistical methods are generalised procrustes analysis (GPA) (Gower, 75) and multiple factor analysis (MFA) (Escofier and Pagès, 1994). Even though both GPA and MFA are conceptually very different (see details on each method in section 2.2.3.1 and 2.2.3.2) both belong to the family of the so-called multiblock methods (Abdi et al., 2013). They provide amongst other information regarding the consensus product configuration which is derived from product placing of all individuals, contribution of each block and its variables to the variation in the data, etc. In general, both GPA and MFA can be applied on blocks or data coming from any type of measurement (sensory, chemical, physical, etc.) when carried out on the same objects, since they allow for the numbers of variables in each block to be different. This is an important property when different types of measurement are available for analysis. In this study focus will be on analysis of data acquired from projective mapping, i.e. data blocks with only two variables holding x- and y-coordinates of product placings, which defines a special case of multiblock situation.

To the author's knowledge there exists no study that discusses in detail the differences between the two methods in general, and when applied on the same set of projective mapping data in particular. Only one study (Nestrud and Lawless, 2008) briefly mentions that both methods have been tested on the same data set and reports that results were very similar. In this study GPA and MFA were applied on data that were generated from a single experiment where 13 citrus juices were evaluated by a group of experienced chefs and a group of untrained consumers. Configurations within each group acquired from GPA and MFA were compared using the RV coefficient (Robert and Escoufier, 1976) and found to be similar (see section 2.2.3.4 for details on RV coefficient).

This study attempts to give more insight into differences in results acquired with GPA and MFA in the case of projective mapping. This is done through comparisons of results from GPA and MFA computed from: (I) random data in Monte Carlo simulations; (II) constructed data that were manipulated using some specific criteria; (III) real world data from nine Napping experiments. Furthermore, for case (II) and (III) a combination of the two methods will be investigated, where Procrustes rotation is carried out first for all data matrices followed by MFA.

2.2 Materials and Methods

2.2.1 Projective Mapping

Projective mapping is a method where individuals evaluate the overall perception of a number of products and place them on a sheet according to the products similarities or dissimilarities (Risvik et al., 1994; Pagès, 2005). Placement can be done either by putting products directly on a sheet of paper or by indicating their position on a computer screen sheet. Individuals are instructed to place products close to each other if they are perceived to be similar and vice versa using their own criteria they consider being important for the products. Other than that individuals are not given further instructions. If the placement of the products needs to be refined, the individuals may taste the products again (provided that they are food products) until placement is considered to be satisfying. Optionally, individuals may be asked to write down sensory descriptors on the sheet close to the tested products, that best describe each group of products. By doing so the projective map is turned into an Ultra Flash Profile method as described previously (Perrin et al., 2008). In this study, however, focus will be only on the product sheet coordinates (two-dimensional data blocks in form of x- and y-coordinates) coming from each individual. These multiblock data are then transformed into a consensus product configuration using GPA, MFA and a combination of the two, representing the all individuals.

A well know critique regarding projective mapping worth mentioning is that complex multidimensional products may be difficult to place on the two-dimensional sheet since the two dimensions of the sheet may not be enough to distinguish the products properly and may the leave the user with a non-satisfying placement of the products. Recent research (Nestrud and Lawless, 2011), however, refutes this criticism by claiming that subsampled dimensions and configurations could be recovered using MFA and multidimensional scaling.

2.2.2 Projective Mapping data used in study

This section describes the data that were used in this study and served the purpose for comparison of outcomes from GPA and MFA. Three types of data were used: (I) random data generated with different settings to simulate a specific number of consumers evaluating a specific number of products; (II) constructed data that simulate certain situations and that allow for checking how GPA and MFA handle translations, rotations and scaling of configurations; (III) real world data from nine Napping experiments. The aim was to compare statistical results provided by GPA and MFA in these three specific situations and gain insight into potential differences. However, before going into detail regarding the three data types it is important to understand the general structure of projective mapping data. This will be done briefly in the next section 2.2.2.1.

2.2.2.1 General Structure of Projective Mapping Data

Since every individual taking part in the projective mapping trial is supposed to place a number of products on a projective mapping sheet the resulting individual data block $\mathbf{Z}_k$ is of dimension $(I \times J)$ with $J = 2$. Here I represents $i = 1, \ldots, I$ number of objects or products tested by the individuals and $j = 1, 2$ represent the two variables represent the x- and y-coordinates of the products on the projective mapping sheet. Provided that a total number of $k = 1, \ldots, K$ individuals have participated in the projective mapping trial there are K individual data blocks $\mathbf{Z}_k$. This is a special case of multi-block data where all blocks are yielded from the same type of measurement and are of type "short thin" consisting of many objects compared to the number of variables. The consensus product configurations from GPA, MFA and PrMFA (details on the methods provided in section 2.2.3.1, 2.2.3.2 and 2.2.3.3) are always computed from the same set of individual data blocks $\mathbf{Z}_k$ and are named $\mathbf{Y}_{GPA}$, $\mathbf{X}_{MFA}$ and $\mathbf{X}_{PrGPA}$, respectively. The PCA scores of these three consensus configurations are named $\mathbf{F}_{GPA}$, $\mathbf{F}_{MFA}$ and $\mathbf{F}_{PrGPA}$, respectively. The final product maps that provide information on how the tested products relate to one another, i.e. the PCA scores plots, are based on them. When speaking of PCA scores of consensus product configuration in general, i.e. independent of the used statistical method used to compute the consensus configuration, they will be referred to as $\mathbf{F}_c$ with $c = GPA, MFA, PrMFA$. When comparing the product maps from GPA, MFA and PrMFA using the RV coefficient, only the first two components of $\mathbf{F}_{GPA}$, $\mathbf{F}_{MFA}$ and $\mathbf{F}_{PrGPA}$ were considered. This restriction was imposed on the MFA and PrMFA data in order to have a common base with the GPA scores which only have two components (details in section 2.2.3.1)

2.2.2.2 Random data for Monte Carlo simulations

The main objective in this part of the study was to investigate the similarity of consensus product configurations $\mathbf{F}_{GPA}$ and $\mathbf{F}_{MFA}$ over a large number of simulations. For each fictive individual taking part in the projective mapping trial random data were generated that fell within a standard projective mapping sheet of size 60 x 40 cm. The amount of random data used in each Monte Carlo simulation depended on the number of individuals $k = 20, 40, 60$ taking part in the trial and the number of products $i = 4, \ldots, 16$ simulated for all blocks $\mathbf{Z}_k$. The random data were generated using a uniform distribution. The upper limit of 16 products was chosen based on one of the real world data sets used in section 2.2.2.4 that compared 16 products (data set 4 in Table 2.1). The upper limit of 100 consumers was also chosen based on another real world data set (data set 7 in Table 2.1) from section 2.2.2.4 using 97 consumers to evaluate the products.

Using all possible pair-wise combinations of number of individuals and number of products a total of 39 Monte Carlo simulations (3 levels of individuals x 13 levels of products) were carried out. For each of the 39 Monte Carlo simulations 1000 runs were carried out, i.e. 1000 consensus product configurations $\mathbf{F}_{GPA}$ and $\mathbf{F}_{MFA}$ were computed and for each run their similarity was measured using the RV coefficient. The average across the 1000 RV coefficient was then computed and applied as an indicator for general similarity between $\mathbf{F}_{GPA}$ and $\mathbf{F}_{MFA}$.

2.2.2.3 Constructed data

In this part of the study the aim was to investigate in a controlled setting how GPA, MFA and PrMFA would handle individual product configurations $\mathbf{Z}_k$ from a number of fictive individuals that were initially identical (see Fig. 2.1) before they were subject to one or more targeted manipulations. These targeted manipulations included off-sets from the projective mapping sheet centre, different degrees of rotations and reflections, and variation of relative product distances. Those are exactly the types of situations that GPA can handle with its Procrustes transformations to make individual product configurations as similar as possible. The question at hand was whether MFA could handle such situations and whether a combination of GPA and MFA would produce a consensus configuration that better represents all individuals. The PCA scores $\mathbf{F}_{GPA}$, $\mathbf{F}_{MFA}$ and $\mathbf{F}_{PrGPA}$ of the resulting consensus product configurations were then compared with one another using the RV coefficient to make a statement regarding their similarity. For this purpose five scenarios were created with manipulated data sets for 8 fictive individuals. The data in each scenario were manipulated by applying at least one or a combination of the three manipulations mentioned above.

As mentioned above, a constructed individual product configuration of five products (see Fig. 2.1), from now on called initial product configuration, was used as a starting point for all in-

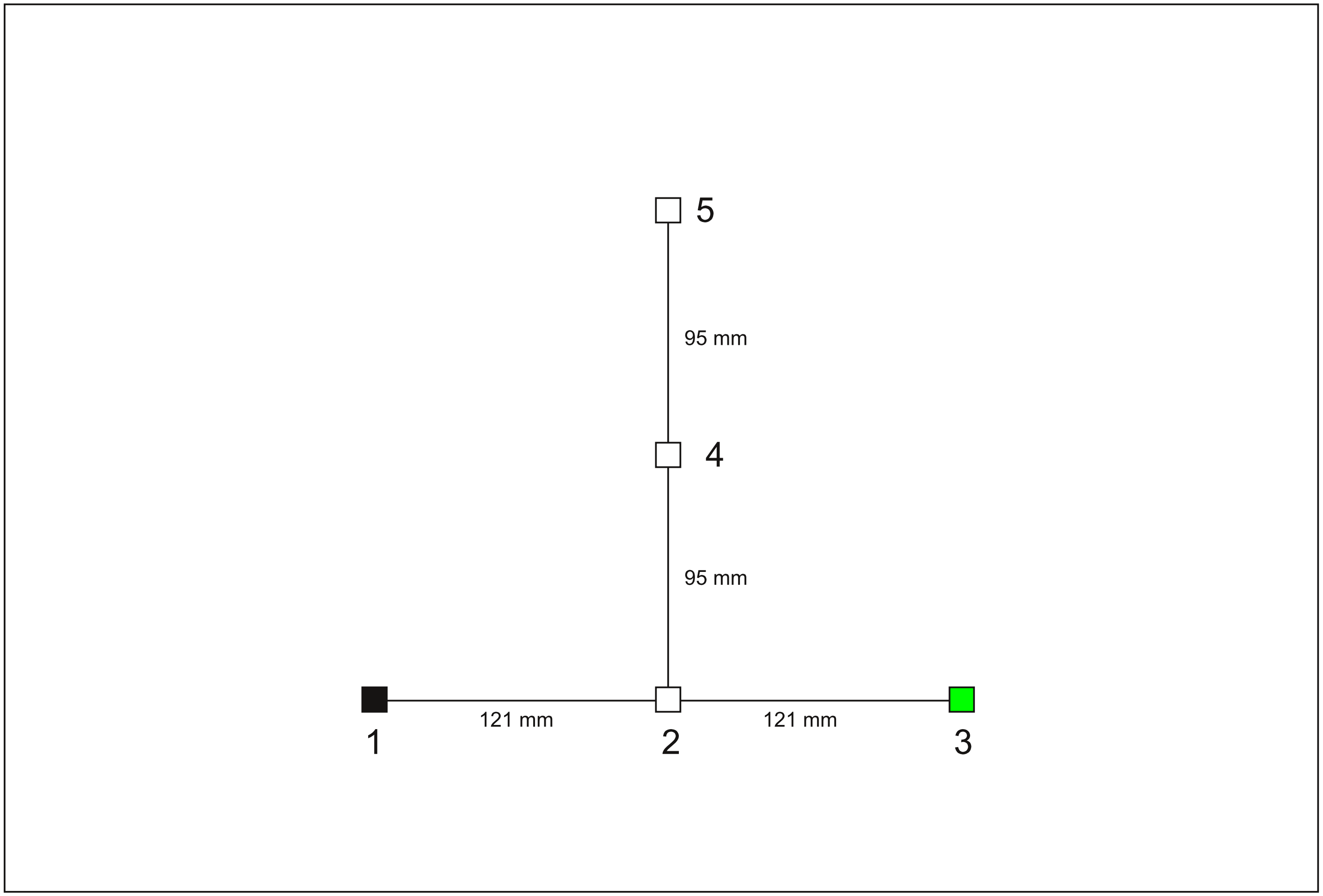

Figure 2.1: The plot shows a projective mapping sheet and a constructed individual product configuration $\mathbf{Z}_k$ with 5 products numbered 1 to 5. Product 4 is located in the middle of the projective mapping sheet. This individual product configuration was used as a starting point for all fictive individuals prior to targeted manipulation.

dividuals before controlled manipulation. This initial configuration represents a triangle shape and is centred in the middle of the projective map (600 mm x 400 mm) pointing “north”. Product 4 is located exactly in the middle of the projective map. The distance between products 1 and 2 was set to 121 mm and is identical to the distance between products 2 and 3. The distance between products 2 and 4 was set to 95 mm and is identical to the distance between products 4 and 5. The axis formed by products 1, 2 and 3 is orthogonal to the axis formed by products 2, 4 and 5. In order to avoid numerical computation problems with MFA, 1 % random noise was added to each of the individual configurations after they were manipulated according to the five constructed data situations as described below.

Constructed data 1: rotation only

For all 8 fictive consumers the initial individual product configuration (see Fig. 2.1) was used as a starting point. Before and after rotation all configurations were centred in the middle of the projective map, i.e. product 4 kept its position on the projective map. No translation or reflection took place and distances between all objects were preserved, meaning there was neither isotropic scaling nor deformation of the triangle shape. The only manipulation undertaken was clock-wise rotation of the initial individual product configuration in fixed steps of 45 degrees with each individual. More specifically the rotations were the following: fictive individual 1 (0°, pointing north); fictive individual 2 (45°, pointing north-east); fictive individual 3 (90°, pointing east); fictive individual 4 (135°, pointing south-east); fictive individual 5 (180°, pointing south); fictive individual 6 (225°, pointing south-west); fictive individual 7 (270°, pointing west); fictive individual 8 (315°, pointing north-west).

Constructed data 2: translation only

For all 8 fictive consumers the initial product configuration (see Fig. 2.1) was used as a starting point. Before and after translation all product configurations pointed “north”. No rotation or reflection took place and distances between all objects were preserved meaning there was neither isotropic scaling nor deformation of the triangle shape. The only manipulation each individual product configuration underwent was 40 mm translations away from the projective mapping sheet centre. More specifically translation directions were the following: fictive individual 1 (south); fictive individual 2 (south-west); fictive individual 3 (west); fictive individual 4 (north-west); fictive individual 5 (north); fictive individual 6 (north-east); fictive individual 7 (east); fictive individual 8 (south-east).

Constructed data 3: translation and rotation

These constructed data are a combination of constructed data 1 and 2, where all individual product configurations were moved away 40 mm from the projective map centre and rotated clock-wise in 45 degrees steps. The product configurations of fictive individual 2, as an example, was moved away 40 mm southwest from the projective map centre and was rotated 45 degrees clock-wise such that the vertex pointed north-east.

Constructed data 4: translation, rotation and reflection

These data are constructed in the same way as constructed data 3, however, the product configuration of fictive consumer 2, 4, 6 and 8 (those whose product configurations were moved away diagonally from the centre) are pointing away from the projective map sheet centre and product configuration being mirrored along the axis formed by products 2, 4 and 5. Fig. 2.2 visualises schematically what the individual product placements of the 8 fictive consumers would look like if they were placed on the same projective mapping sheet.

Constructed data 5: changing relative distances between products

For all 8 fictive individuals the initial product configuration (see Fig. 2.1) was used as a starting point. Before and after stretching and shrinking all individual product configurations were centred in the middle of the projective map and pointing north, meaning that no translation, rotation or reflection took place. This time distances between products were changed by stretching and shrinking the individual product configurations in different ways and combinations. Stretching and shrinkage were applied along either the axis spanned by product 1, 2 and 3 or the axis spanned by product 2, 4 and 5 or both. Changes applied to distance between product 1 and 2 were identical to those of distance between product 2 and 3. Furthermore, changes of the distance between product 2 and 4 were identical to those of distance between products 4 and 5. This resulted in 8 differently shaped isosceles triangles.

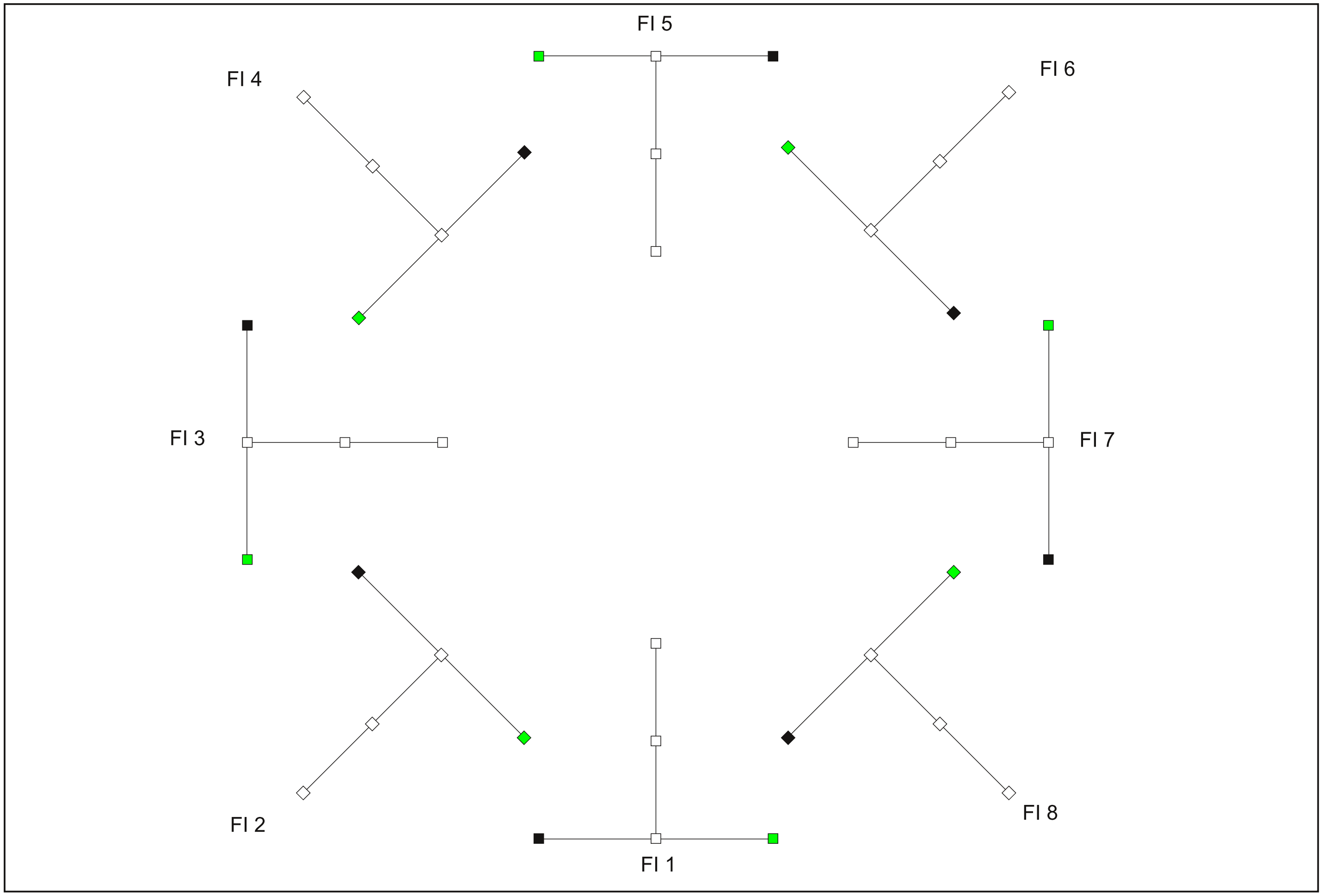

Figure 2.2: The plot shows schematically how the individual product configurations from the 8 fictive individuals (in plot abbreviated with FI) are placed in relation to one another if they were placed on one single projective mapping sheet. Note that those are not the real placement coordinates but just a simplified visualisation. If the real placement coordinates were used the individual product configurations would overlap to a great extent.

2.2.2.4 Real world data

In this part of the study nine Napping data sets from real experiments were analysed. Table 2.1 provides a short summary over what kind of and how many products were used in each experiment, how many individuals tested the products and whether there is a connection to any of the other data sets. Note that data set 1 and 2 were acquired through experiments carried out at Nofima and that the remaining data sets were kindly provided by Husson on his own web site. The tested products are of varying sensory complexity ranging from relative low complexity products like apple and orange juices to relatively high complexity products such as wine. From experience it is know that sensory-wise complex products generate more variation across consumers and experts than products of rather low sensory complexity would do.

Table 2.1: Overview over the real world data sets used in this study.

Data set #	Product type	# of products	# of individuals	Remarks
1	apple juices	8	16	Same products as in data set 2
2	apple juices	8	11	Same products as in data set 1
3	biscuits	8	18	
4	cocktails	16	10	
5	orange juices	12	20	same products as in data set 6
6	orange juices	12	28	same products as in data set 5
7	perfumes	12	97	same products as in data set 8
8	perfumes	12	23	same products as in data set 7
9	wines	10	18	

2.2.3 Statistical methods and measures

GPA and MFA have in common that they are multivariate statistical methods designed for analysis of multiple blocks of data measured on the same set of objects. Both methods allow the number of variables across data blocks to vary; a feature that makes them suitable to incorporate any kind of measurements in the analysis. GPA and MFA are both known to be used for analysis of projective mapping data, a special case of multiblock data. Both methods provide many of the same types of results as for example consensus scores, loadings, block loadings, contribution of each block to the total variance, etc. But despite these similarities regarding their general frameworks the way results are computed are very different. In the following sections GPA and MFA will be introduced briefly in order to emphasise the differences between their computational approaches. A combination of GPA and MFA, named PrMFA in this study, was also applied to the data in section 2.2.2.3 and 2.2.2.4 to investigate whether the combination of the two methods would bring any additional benefits beyond what is provided by GPA and MFA.

2.2.3.1 Generalised Procrustes Analysis (GPA)

GPA (Gower, 75; Dijksterhuis, 1996) is a multivariate statistical method that is applied for analysis of multiple data blocks. The main goal is to acquire a consensus from those blocks after they have undergone Procrustes transformations that reduce individual differences by means of translation, rotation and reflection as well as isotropic scaling. GPA is therefore well suited for analysis of projective mapping data given our goal to find a consensus product configuration from all individuals who took part in the mapping. Note that GPA consists of two steps: (I) Procrustes transformation followed by (II) Principal Component Analysis (Martens and Næs, 1989) on those transformed data blocks.

A data block $\mathbf{Z}_k$ contains the individual product configuration of individual k based on how this individual places the tested products on the projective mapping sheet and the resulting product x- and y-coordinates. Clearly, there will always be differences across individuals how individuals place the products on the sheet. This might be because of individuals perceiving the products differently, but also due to their different ways of placing products on the mapping sheet. Regarding the former, these are the sensory differences that are relevant for computation of the consensus product configuration since they contain important sensory information regarding the products. For the latter these differences arise because of individuals using more or less space on the mapping sheet; individual product configurations may be very similar but mirrored and/or rotated against one another; they may have varying degrees of shift off the origo (middle of mapping sheet). These are the differences one would like to eliminate from the individual data since they are not really product related but merely a result from individuals using the mapping sheet in different ways. GPA accounts for these non-sensory related differences through its Procrustes transformation.

In general, the three steps of Procrustes transformation can be summarised like this: (a) translation, meaning that all individual configurations are moved to the middle of the mapping sheet. In statistical terms this is the mean-centring of the x- and y-coordinates; (b) rotation and reflection of individual configurations until they are in best possible agreement with one another; (c) isotropic scaling, i.e. shrinking or stretching of individual configurations until they are alike as possible but, however, without changing the relative distances between the products in each configuration.

Statistically, the three steps of Procrustes transformation may be summarised in the following way:

$$\tau(\mathbf{Z}_{[k]}) = \rho_k \mathbf{Z}_{[k]} \mathbf{H}_{[k]} + \mathbf{T}_k \tag{2.1}$$

where $\tau(\mathbf{Z}_k)$ represents the Procrustes transformation of block $\mathbf{Z}_k$, $\mathbf{T}_k$ is the matrix of translation constants (step (a)), $\mathbf{H}_k$ represent the rotation matrix (step (b)) and ρ_k represents the scalar from isotropic scaling (step (c)). Note that $\mathbf{H}_k$ is an orthogonal matrix $\mathbf{H}^T\mathbf{H} = \mathbf{H}\mathbf{H}^T = I$. $\mathbf{H}_k$ and ρ_k of each data block are obtained by minimising:

$$\sum_{k=1}^{K} \|\rho_k \mathbf{Z}_{[k]} \mathbf{H}_{[k]} - \mathbf{Y}_{GPA}\| \tag{2.2}$$

where $\mathbf{Y}_{GPA}$ represents the mean or so-called consensus matrix across all transformed blocks $\tau(\mathbf{Z}_{[k]})$. $\mathbf{Y}_{GPA}$ is of dimension $(I \times 2)$, i.e. exactly the same dimension as the individual data blocks $\mathbf{Z}_{[k]}$. As a final step $\mathbf{Y}_{GPA}$ is then analysed with PCA where the scores plot represents the final consensus sensory map containing the consensus product configuration based on the placements of all individuals.

Note that since $\mathbf{Y}_{GPA}$ is of dimension $(I \times 2)$, only two principal components (PC) may be extracted from the data. As a consequence all information in the resulting consensus product configuration $\mathbf{F}_{GPA}$, which also is of dimension $(I \times 2)$, will be contained in the space spanned by these two PC's (PC1 and PC2 will cumulatively always reach 100% explained variance). This property of GPA may be regarded as a major drawback since most products are usually more than only two-dimensional when seen from a sensory perspective and therefore the resulting consensus product configuration $\mathbf{F}_{GPA}$ may provide an oversimplified picture of the situation.

2.2.3.2 Multiple Factor Analysis (MFA)

There are several ways to describe mathematically how MFA (Escofier and Pagès, 1994) works. A thorough review of the alternatives is provided elsewhere (Abdi et al., 2013). To keep this section brief MFA is presented as a 'simple PCA', one of three alternatives mentioned above (Abdi et al., 2013). Note that for the computation of the consensus matrix $\widetilde{\mathbf{X}}_{MFA}$ the variables of each individual block $\mathbf{Z}_{[k]}$ are standardised such that they become $\mathbf{X}_{[k]}$. The following steps are carried out: (I) singular value decomposition (SVD) for each individual block $\mathbf{X}_{[k]}$ separately where amongst other the first singular value $\gamma_{1,k}$ of each $\mathbf{X}_{[k]}$ is acquired; (II) new blocks $\widetilde{\mathbf{X}}_{[k]}$ are computed, i.e. normalised versions of $\mathbf{X}_{[k]}$ that were divided by their respective first singular value $\gamma_{1,k}$ acquired in step (I). This is formalised the following way:

$$\widetilde{\mathbf{X}}_{[k]} = \gamma_{1,k}^{-1}\mathbf{X}_{[k]} \tag{2.3}$$

(III) All $\widetilde{\mathbf{X}}_{[k]}$ are concatenated horizontally forming the consensus matrix $\widetilde{\mathbf{X}}_{MFA}$

$$\widetilde{\mathbf{X}}_{MFA} = [\widetilde{\mathbf{X}}_{[1]}|\ldots|\widetilde{\mathbf{X}}_{[k]}|\ldots|\widetilde{\mathbf{X}}_{[K]}] \tag{2.4}$$

(IV) PCA is applied on $\widetilde{\mathbf{X}}_{MFA}$ which results in consensus product configuration $\mathbf{F}_{MFA}$. Note that, unlike with GPA, more components are available for visualisation the consensus product configuration in MFA, since $\mathbf{F}_{MFA}$ is of dimension $(I \times L)$ where L represents the number of PC's extracted by PCA. Typically, the first two components will be the most relevant to look at since they explain most of the systematic variance in the data. But one can also investigate component 3 and further, which is not possible in GPA. With MFA it is therefore possible that for complex products each component could be related to a sensory perception or modality. This is not possible with GPA since there are always only two dimensions in the sensory consensus map available.

2.2.3.3 Combination of Generalised Procrustes Analysis and Multiple Factor Analysis (PrMFA)

In addition to applying GPA and MFA on the projective mapping data it was of interest to investigate whether a combination of the two methods would bring any benefits. In the combined approach the data were first treated by Procrustes transformation as with GPA followed by MFA on the transformed data. This results in the consensus product configuration $\mathbf{F}_{PrMFA}$. This approach was applied only on the constructed and the real world data from section 2.2.2.3 and 2.2.2.4, respectively.

Note that the combined approach mentioned above is not to be confused with another approach called Procrustes Multiple Factor Analysis (Morand and Pagès, 2006) which applies the two methods in reverse order. In that study first MFA was applied on the individual data $\mathbf{X}_{[k]}$ followed by Procrustes transformations of individual data $\mathbf{F}_{k,MFA}$ to make them as similar as possible to the MFA consensus $\mathbf{F}_{MFA}$ for the purpose of interpretation.

2.2.3.4 RV coefficient

The variance across all variables (columns) in a data matrix $\mathbf{X}$ provides information on the configuration of its objects (rows), i.e. describing how the objects relate to one another in the multidimensional space spanned by the variables. In our case with projective mapping, the object configuration is a result of how an individual has placed the products on the two-dimensional sheet. Now If one needs to determine the degree of similarity of how two individuals have placed the products on their respective sheets one needs to analyse their data matrices $\mathbf{X}_1$ and $\mathbf{X}_2$. The similarity of two object configurations can be measured with the RV coefficient (Robert and Escoufier, 1976). It is computed as follows:

$$RV = (\mathbf{X}_1, \mathbf{X}_2) = \frac{tr[\mathbf{X}_1\mathbf{X}_1^T\mathbf{X}_2\mathbf{X}_2^T]}{\sqrt{tr[\mathbf{X}_1\mathbf{X}_1^T\mathbf{X}_1\mathbf{X}_1^T]\,tr[\mathbf{X}_2\mathbf{X}_2^T\mathbf{X}_2\mathbf{X}_2^T]}} \tag{2.5}$$

Note that both $\mathbf{X}_1$ and $\mathbf{X}_2$ are assumed to be column centred. The RV coefficient is a scalar that varies between 0 and 1. The higher the RV coefficient, the more similar are the object configurations in $\mathbf{X}_1$ and $\mathbf{X}_2$. Important properties of the RV coefficient are scale and rotation invariance which is very convenient when analysing data from projective mapping. Individuals may place the products in a similar manner, yet they might use more or less space on the projective mapping sheet. The RV coefficient accounts for this type of individual difference through its scale invariance. Furthermore, individuals' placements of the products may be similar, yet rotated in relation to one another or mirrored in either one or both sheet coordinate axis. Also this problem the RV coefficient accounts for through its rotation invariance. In general, when computing the RV coefficient for two data matrices $\mathbf{X}_1$ and $\mathbf{X}_2$ the number of variables in each matrix may be different. However, with the projective mapping data dealt with in this study all data matrices have only two variables, i.e. the first two columns of $\mathbf{F}_{GPA}$, $\mathbf{F}_{GPA}$ and $\mathbf{F}_{GPA}$.

Below in section 2.3.1 the RV coefficient will be used to compute similarity of consensus product configurations that were acquired with GPA and MFA from the same set of individual data. In section 2.3.2 and 2.3.3 the RV coefficient will be applied to compute similarity of consensus product configurations that were acquired with GPA, MFA and PrMFA from the same set of individual data. Note that the value of the RV coefficient is dependent on the number of objects and variables in $\mathbf{X}_1$ and $\mathbf{X}_2$ (Smilde et al., 2009) and that it may be subject to a centering effect (Tomic et al., 2013). This is especially true for "short fat" matrices with few objects and many variables. In this study, however, these problems are of less relevance since all matrices are of type "long thin". This means typically that in $\mathbf{X}_1$ and $\mathbf{X}_2$ there will be many more objects (products on the sheet) relative to the two variables (x- and y-coordinates).

2.2.3.5 Similarity ratio for projected individual data

When applying GPA, MFA or PrMFA on a projective mapping data set each method computes PCA scores $\mathbf{F}_c$ from a consensus product configuration. The validity of the consensus product configuration can be evaluated by measuring how well $\mathbf{F}_c$ represents each individual product configuration. This can be done by inspecting the PCA scores of the consensus product configuration $\mathbf{F}_c$ and the sub-space it spans together with the PCA scores $\mathbf{F}_{[k],c}$ of individual product configurations projected into that same sub-space. Differences between the consensus scores $\mathbf{F}_c$ and the individual projected scores $\mathbf{F}_{[k],c}$ can be utilised to compute a measure of how well the particular individual is represented by the consensus. This measure, named similarity ratio $SR_{k,c}$ in this study, may be computed for each individual i in the following way:

$$SR_{k,c} = \sum_{k=1}^{K} \frac{\|\mathbf{F}_c - \mathbf{F}_{[k],c}\|^2}{\|\mathbf{F}_c\|^2} \tag{2.6}$$

where $k = 1, \ldots, K$ represents the individuals participating in the projective mapping; $c = GPA, MFA, PrMFA$ represents the method the ratio is computed for; $\mathbf{F}_{[k]}$ represents projected scores of an individual k. Note that only the first two columns for each $\mathbf{F}_{[k]}$ and $\mathbf{F}_c$ are used. Computing the similarity ratio $SR_{k,c}$ in this way is convenient, since it is independent of the scores units provided by each GPA, MFA and PrMFA and thus makes comparisons across the three methods possible. From Eq. 2.6 can be seen that the larger the difference between the projected scores $\mathbf{F}_{[k]}$ and the consensus product configuration $\mathbf{F}_c$ is, the higher is $SR_{k,c}$ for that particular individual. There is no upper limit for the similarity ratio; however for the real world data used in this study the highest value reached was about 4.5. If the projected scores $\mathbf{F}_{[k]}$ are exactly the same as the consensus scores $\mathbf{F}_c$ then $SR_{k,c}$ will be zero since the nominator in Eq 2.6 will be zero. To get a measure of how well the consensus product configuration $\mathbf{F}_c$ represents the whole group of individuals one can compute the total sum of all individual similarity indices. This is done as follows:

$$SR_{tot,c} = \sum_{k=1}^{K} SR_{k,c} \tag{2.7}$$

The sum and the standard deviation of the similarity ratio then allow for comparison of the methods, where one would prefer the method with: (I) the lowest $SR_{tot,c}$, meaning that overall individuals are represented by the consensus in a best possible manner in combination with; (II) lowest possible standard deviation across $SR_{k,c}$ meaning that the consensus product configuration is representative for most individuals.

2.2.4 Data Analysis Software

Monte Carlo simulations of random data in subsection 2.2.2.2 were carried out in a Python programming language environment using the numerical package Numpy (Oliphant, 2007). From the Python environment GPA and MFA functions were called to do the computations on the random projective mapping data. The GPA and MFA functions are part of the FactoMineR package (Lê et al., 2008) coded in R and were accessed through PypeR (Xia et al., 2010), an interface between the Python and R programming languages.

For the constructed and real world data of subsection 2.2.2.3 and 2.2.2.4 the commercial XLSTAT software was used for computation of results. Both GPA and MFA are part of the XLSTAT-MX add-on package for market research and sensory analysis. In particular, the Gower implementation of GPA in XLSTAT was used for analysis.

For all three types, random, constructed and real world data, a Python implementation of the RV coefficient was used to compute values for configuration similarities from GPA, MFA and PrMFA where applicable.

2.3 Results and Discussion

2.3.1 Monte Carlo simulations with random data

Fig. 2.3 summarises results from Monte Carlo simulations on generated random data as described in section 2.2.2.2. The figure shows three box plots visualising the distribution of RV coefficient computed for $\mathbf{F}_{GPA}$ and $\mathbf{F}_{MFA}$ with 20, 60 and 100 individuals, respectively. In each box plot simulation results from 4 to 16 products are shown for 1000 simulation runs each. In general one may conclude that on average similarity between $\mathbf{F}_{GPA}$ and $\mathbf{F}_{MFA}$ is high and that the distribution of the RV coefficients is quite similar whether computations were done for 20, 60 and 100 individuals.

Fig. 2.4 is based on the same data, however, this time only the average RV coefficient is shown for each individuals-product combination. Each data point represents the average across RV coefficients from 1000 simulations for a specific combination of number of individuals and number of products. As can be seen on average the highest similarities between the scores $\mathbf{F}_{GPA}$ and $\mathbf{F}_{MFA}$ from GPA and MFA are present with a low number of products. For all tested numbers of individuals, i.e. 20, 60 and 100 individuals, RV coefficients are the highest for 4 products and decreasing with an increasing number of products. The only exception is a slight increase taking

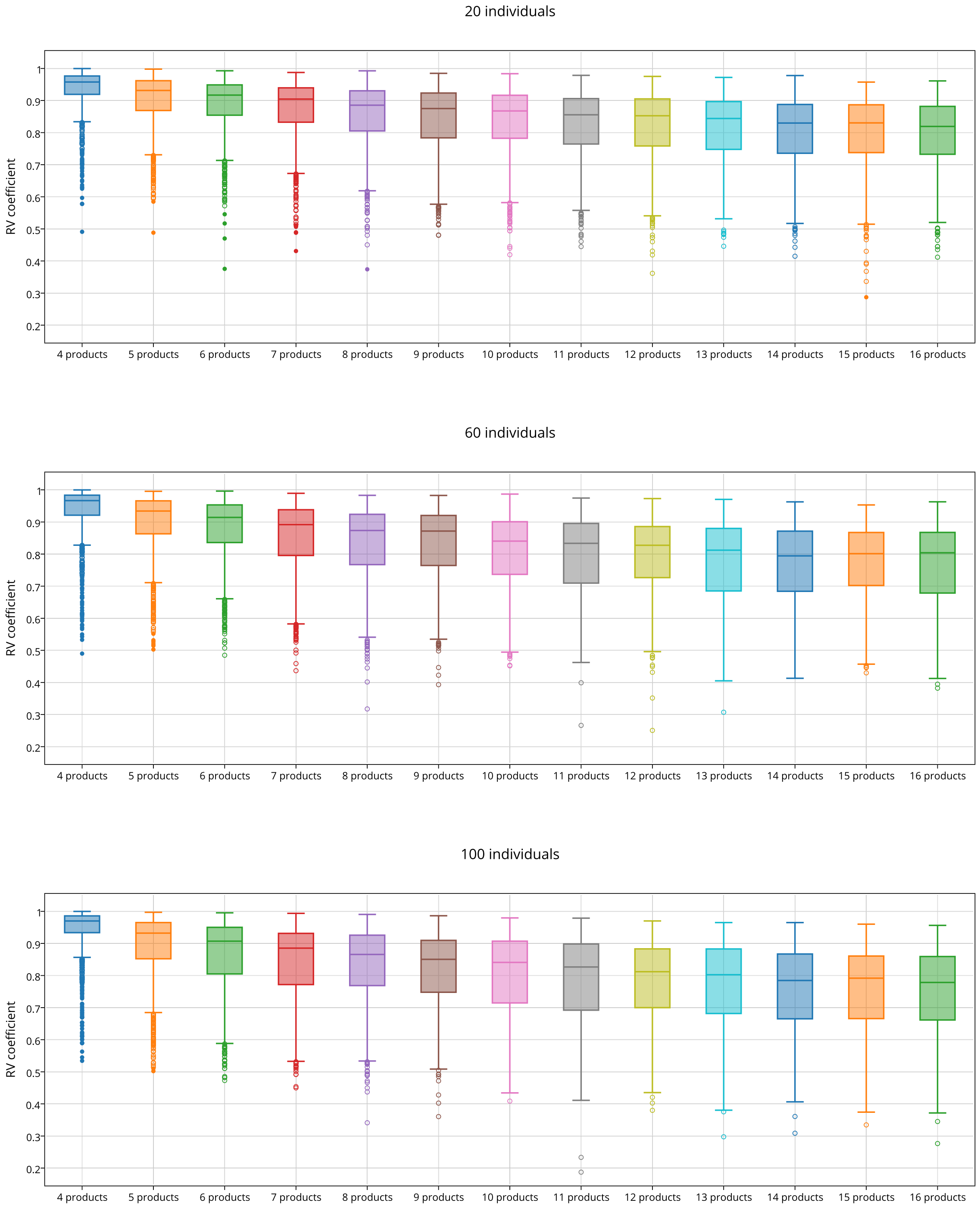

Figure 2.3: Box plots are shown summarising the distribution of the RV coefficients for 20, 60 and 100 consumers for a given number of products. Each column in the box plots is based on 1000 simulations. The RV coefficients were computed for the PCA scores (first two PC's) $\mathbf{F}_{GPA}$ and $\mathbf{F}_{MFA}$ of the consensus product configurations acquired with GPA and from MFA, respectively.

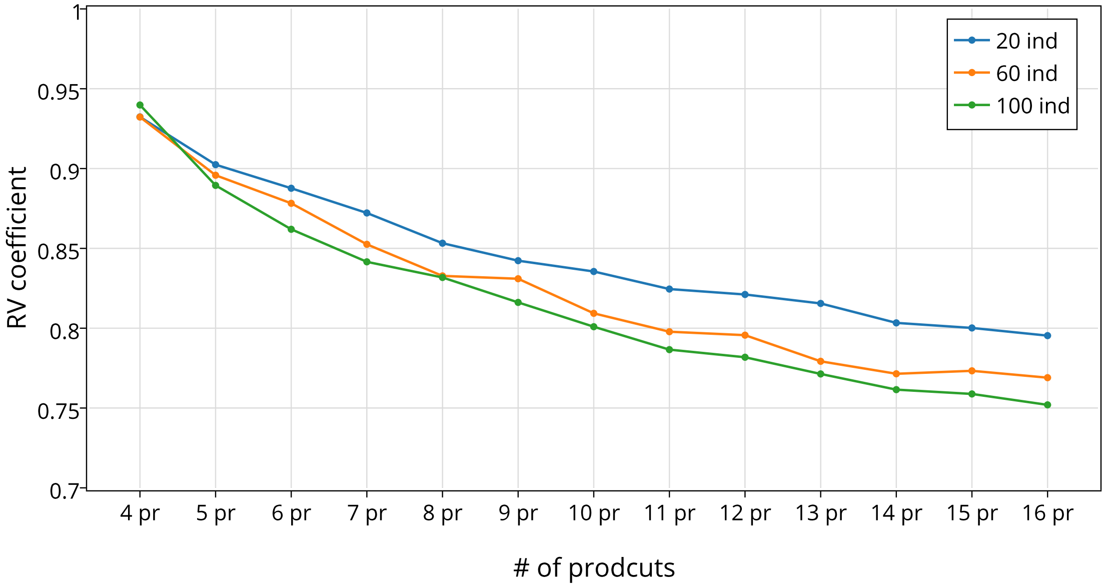

Figure 2.4: These are the same data as shown in Fig. 2.3. Here, however each data point displays average across 1000 simulations of a specific consumer-product combination.

place between 14 and 15 products for 60 consumers where the average RV coefficient is 0.7715 and 0.7733, respectively. As can be seen the decrease of RV coefficient value by adding another product to simulations is larger when the number of products is low. From what can be seen in Fig. 2.4 in general is that the changes are getting continuously smaller when the number of products is high. Furthermore, it seems that GPA and MFA consensus product configurations are less similar when the number of consumers increases.

The simulations have been computationally extremely expensive, mainly because GPA took a very long time to compute. As an example, using a standard 4-core processor laptop it took GPA between 9 and 10 minutes to compute one single simulation out of 1000 for 4 products and 100 consumers. For this particular consumer-product combination GPA computation times were longer than for any other combinations and about 120 longer than for one single MFA simulation. With a higher number of products GPA computation time typically decreased.

Although RV coefficients on average were relatively high (with RV > 0.7) it must be noted that the consensus product configurations $\mathbf{F}_{GPA}$ and $\mathbf{F}_{MFA}$ may be different to such a degree that interpretation regarding the tested products vary to some extent. An example of this is given in section 2.3.3 and Fig. 2.6 for real world data set 5 (see Table 2.1).

2.3.2 Constructed data

For constructed data 1 to 4 (as described in section 2.2.2.3) GPA, MFA and PrMFA returned almost identical results, which indicates that in such a simple setting the three methods handle translation, rotation and reflection very similarly. The resulting consensus configurations and their PCA scores $\mathbf{F}_{GPA}$ and $\mathbf{F}_{MFA}$ are practically identical considering that RV coefficients never fall below 0.9985 for any comparison (see Table 2.2). Note that an RV = 1 would indicate that the configurations are identical and that almost all of the computed RV coefficients in Table 2.2 are very close to this value. At this point it is of interest to take into account individual product configurations $\mathbf{F}_{[k],c}$ from the 8 fictive consumers, project them into the consensus configuration space and investigate how closely they fit $\mathbf{F}_{GPA}$ of their respective consensus configuration. The closer an individual product configuration is to the consensus configuration, the better it is represented by the consensus configuration. The closeness can be measured by the similarity ratio $SR_{k,c}$, which was described above in section 2.2.3.5. The lower SR_k is, the closer the individual product configuration is to the consensus product configuration. Table 2.2 shows $SR_{tot,c}$ for GPA, MFA and PrMFA for the constructed data 1 to 5.

Since constructed data 4 (see Fig. 2.2) is the most complex of the first four, containing translation, rotation and reflection, this one will be discussed here in more detail. Fig. 2.5 shows the PCA scores $\mathbf{F}_{GPA}$, $\mathbf{F}_{MFA}$ and $\mathbf{F}_{PrMFA}$ three consensus product configurations acquired with GPA, MFA and PrMFA as well as their respective projected scores $\mathbf{F}_{[k],GPA}$, $\mathbf{F}_{[k],MFA}$ and $\mathbf{F}_{[k],PrMFA}$. The plots clearly visualise how all three methods succeeded in handling translation, rotation and reflection of the individual product configurations. In each case the individual product configurations are very close to their respective consensus product configuration. Although very small one can observe that the projected individual product configurations of product 1 and 3 are somewhat more scattered for in case of MFA than they are for GPA and PrMFA. This is reflected by $SR_{tot,c}$ for each method c, where $SR_{tot,MFA}$ is about 23 times higher than $SR_{tot,GPA}$

Table 2.2: Numerical results for each of the five constructed data scenarios as described in section 2.2.2.3. The middle part of the table shows RV coefficients for the first two PC's of consensus configurations from GPA, MFA and PrMFA in each scenario. The right part of the table summarises computations of the similarity ratio $SR_{tot,c}$ and $SR_{k,c}$ from the projected individual product configurations compared to the consensus configuration.

Data set	**RV**				**SR**		
construtced data 1		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
rotation only	*GPA*	1	0.9997	0.9987	*GPA*	5.04E-04	3.21E-05
	MFA	0.9997	1	0.9997	*MFA*	1.17E-02	7.96E-04
	PrMFA	0.9987	0.9997	1	*PrMFA*	4.73E-04	1.39E-05
constructed data 2		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
translation only	*GPA*	1	0.9985	0.9985	*GPA*	5.09E-04	2.90E-05
	MFA	0.9985	1	1.0000	*MFA*	7.32E-04	7.38E-05
	PrMFA	0.9985	1.0000	1	*PrMFA*	7.11E-04	7.08E-05
constructed data 3		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
translation and	*GPA*	1	0.9997	0.9987	*GPA*	5.04E-04	3.21E-05
rotation	*MFA*	0.9997	1	0.9997	*MFA*	1.17E-02	7.96E-04
	PrMFA	0.9987	0.9997	1	*PrMFA*	4.73E-04	1.39E-05
constructed data 4		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
translation, rotation	*GPA*	1	0.9997	0.9988	*GPA*	4.63E-04	1.26E-05
and reflection	*MFA*	0.9997	1	0.9997	*MFA*	1.08E-02	5.87E-04
	PrMFA	0.9988	0.9997	1	*PrMFA*	5.10E-04	2.18E-05
constructed data 5		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
changing relative	*GPA*	1	0.9994	0.9994	*GPA*	4.07E-01	4.66E-02
distances between	*MFA*	0.9994	1	1.0000	*MFA*	1.45E-03	1.65E-04
products	*PrMFA*	0.9994	1.0000	1	*PrMFA*	1.32E-03	1.70E-04

and $SR_{tot,PrMFA}$ (see Table 2.2).

Note that GPA, MFA and PrMFA provide very similar results for $SR_{tot,c}$ for constructed data 2 where only translation is taking place. GPA provides the highest $SR_{tot,c}$ in constructed data 5 where the initial configuration used for all 8 fictive individuals were “deformed”. This is due to the fact that GPA preserves the relative distance between the products while MFA is changing relative distances between the products during computation. Intuitively, one would like the relative distances between the products to remain the same which makes interpretation of how the products relate to one another much simpler. MFA, though, transforms the individual configurations of all 8 fictive individuals in such a way, that they are practically identical when looking at $\mathbf{F}_{MFA}$ and $\mathbf{F}_{[k],MFA}$ together in a PCA scores plot. This makes interpretation of the relationship between the products more difficult.

2.3.3 Real World Data

It is important to mention that the nine data sets (see Table 2.1) discussed in this section are of varying degree of complexity and size. The degree of complexity is mainly attributed to sensory dimensionality of the tested products and the possibility that the individuals perceive the products may vary greatly. It was expected that there would be less agreement across individuals for complex products such as perfume (data set 7 and 8) and wine (data set 9) than for example apple juice (data set 1 and 2). Table 2.3 confirms this assumptions with the R_c values from the GPA permutation tests (Wakeling et al., 1992). R_c describes the proportion of total variance explained by the found consensus $\mathbf{Y}_{GPA}$ after Procrustes transformations. While for the simpler products in sensory context, as for example the apple juices in data set 1 and 2 the R_c is as relatively high with 0.713 and 0.577 respectively, the R_c for the wine data set (data set 9) is the lowest with 0.304. This is a clear indication that there is far less agreement across individuals for complex products as in the wine data than there is for less complex products as in the apple juice data. The percentiles next to the R_c in Table 2.3 are computed from permutation tests with 10000 permutations and indicate at which level the R_c of the original value is compared to the distribution of 10000 R_c from the permutation test. The R_c from the wine data indicates that the found consensus configuration $\mathbf{Y}_{GPA}$ might have been a product of chance if level of significance

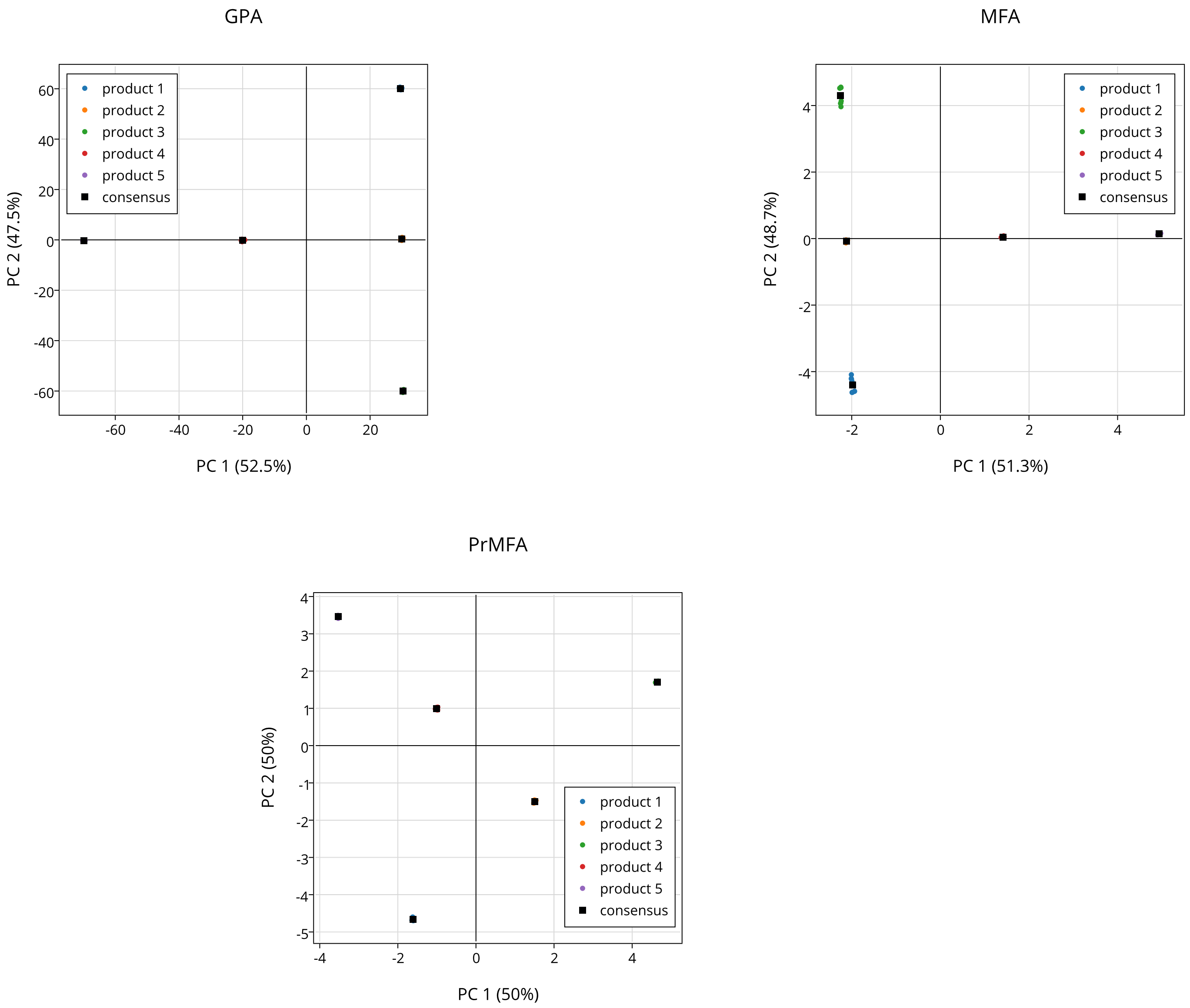

Figure 2.5: PCA scores $\mathbf{F}_{GPA}$, $\mathbf{F}_{MFA}$ and $\mathbf{F}_{PrMFA}$ of the consensus product configurations acquired with GPA, MFA and PrMFA and the respective individual projected scores $\mathbf{F}_{[k],GPA}$, $\mathbf{F}_{[k],MFA}$ and $\mathbf{F}_{[k],PrMFA}$ of the projected individual product configurations.

at the 95th percentile. Note that for the first eight data sets the consensus configuration $\mathbf{Y}_{GPA}$ is considered significant with each of their R_c being at the 100th percentile, which means that none of the R_c from permutations is larger than R_c from the found consensus configurations.

Table 2.3: Overview over the R_c values for each of the nine real world data sets and their percentile from 10000 permutations.

Data set	**Rc**	**Quantile**
1	0.713	100.000
2	0.577	100.000
3	0.527	100.000
4	0.524	100.000
5	0.492	100.000
6	0.427	100.000
7	0.335	100.000
8	0.326	100.000
9	0.304	91.210

The middle part of Table 2.4 shows the RV coefficients between the first two PC's of $\mathbf{F}_{GPA}$, $\mathbf{F}_{MFA}$ and $\mathbf{F}_{PrMFA}$ for consensus product configurations from GPA, MFA and PrMFA for the nine data sets. The RV coefficients in general are relatively high indicating that very often GPA, MFA and PrMFA compute very similar consensus configurations. Many of the RV coefficients across the nine data sets are well above 0.9, some of them closer to 1. The lowest single RV coefficient is given for data set 9 for the wine products with RV = 0.838 between the consensus configurations from GPA and PrMFA which is still relatively high considering that $RV = 1$ indicates perfect agreement between the two.

The next important step is to investigate how well individuals are represented by the consensus product configurations from GPA, MFA and PrMFA. As in section 2.3.2 similarity ratios $SR_{k,c}$ will be used to investigate which one represents individuals in the best manner. The right part of Table 2.4 shows $SR_{tot,GPA}$, $SR_{tot,MFA}$ and $SR_{tot,PrMA}$ as well as standard deviation of $SR_{k,c}$ for each of the nine data sets. It can be seen that for data set 1 to 7 GPA achieves the lowest $SR_{tot,c}$ values and lowest standard deviation across SR_k. Although differences between the three $SR_{tot,c}$ are small, differences between the standard deviation of SR_k appear to be relatively large. With

Table 2.4: RV coefficients for consensus configurations from GPA, MFA and PrMFA and summaries of computations of the similarity ratio $SR_{tot,c}$ and $SR_{k,c}$.

Data set	**RV**				**SR**		
Data set 1		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
apple	*GPA*	*1*	0.966	0.996	*GPA*	5.740	0.128
juices	*MFA*	0.966	*1*	0.979	*MFA*	6.802	0.195
	PrMFA	0.996	0.979	*1*	*PrMFA*	7.427	0.358
Data set 2		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
apple	*GPA*	*1*	0.983	0.985	*GPA*	6.513	0.109
juices	*MFA*	0.983	*1*	0.986	*MFA*	8.115	0.180
	PrMFA	0.985	0.986	*1*	*PrMFA*	9.676	0.468
Data set 3		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
biscuits	*GPA*	*1*	0.974	0.981	*GPA*	12.822	0.092
	MFA	0.974	*1*	0.994	*MFA*	13.259	0.198
	PrMFA	0.981	0.994	*1*	*PrMFA*	13.953	0.405
Data set 4		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
cocktails	*GPA*	*1*	0.975	0.978	*GPA*	7.508	0.046
	MFA	0.975	*1*	0.973	*MFA*	8.787	0.537
	PrMFA	0.978	0.973	*1*	*PrMFA*	9.405	0.677
Data set 5		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
orange	*GPA*	*1*	0.874	0.878	*GPA*	17.758	0.052
juices	*MFA*	0.874	*1*	0.974	*MFA*	21.437	0.382
	PrMFA	0.878	0.974	*1*	*PrMFA*	21.607	0.386
Data set 6		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
orange	*GPA*	*1*	0.947	0.963	*GPA*	32.612	0.071
juices	*MFA*	0.947	*1*	0.980	*MFA*	33.069	0.306
	PrMFA	0.963	0.980	*1*	*PrMFA*	34.296	0.517
Data set 7		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
perfumes	*GPA*	*1*	0.953	0.957	*GPA*	162.589	0.357
	MFA	0.953	*1*	0.989	*MFA*	165.853	0.633
	PrMFA	0.957	0.989	*1*	*PrMFA*	177.018	0.816
Data set 8		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
perfumes	*GPA*	*1*	0.900	0.911	*GPA*	44.420	0.440
	MFA	0.900	*1*	0.953	*MFA*	39.706	0.385
	PrMFA	0.911	0.953	*1*	*PrMFA*	43.710	0.375
Data set 9		*GPA*	*MFA*	*PrMFA*		$SR_{tot,c}$	$std(SR_{k,c})$
wines	*GPA*	*1*	0.875	0.838	*GPA*	38.004	0.576
	MFA	0.875	*1*	0.917	*MFA*	29.327	0.446
	PrMFA	0.838	0.917	*1*	*PrMFA*	34.520	0.539

GPA achieving a standard deviation across its SR_k that are up to 11 and 14 times smaller than those of MFA and PrMFA (see Table 2.4 for data set 4) it seems that the GPA consensus product configuration provides the best representation of the individuals.

Fig 2.6 shows a typical problem that may arise when analysing projective mapping data with different statistical methods. In the figure three consensus product configurations are shown, i.e. from GPA, MFA and PrMFA left, middle and right side of the figure, respectively. The RV coefficients between the three consensus product configurations indicate high similarity (0.874, 0.878 and 0.974; see Table 2.4) but there is no doubt that conclusions regarding the products may be different depending on which statistical method is used for analysis of the projective mapping data. All three consensuses separate the products in a very similar manner along the first component. Products 4, 7, 8, 9 and 12 are on one side of the plot, while products 1, 2, 3, 6, 10 and 11 are found on the opposite side along component 1. Product 5 is placed about in the middle in each of the consensus maps. RV coefficients comparing the similarity of the three are high because the first component is responsible for a large part of the explained variance. Problems however arise when the placement of the products are compared along component 2. One can see substantial differences, as for example the placing of product 1 and 11. In the GPA consensus the two products may be considered quite different regarding the second component whereas in the MFA and PrMFA consensuses they may be interpreted to be very similar overall. The position of product 9 is another example of where interpretation is obviously very dependent on the choice of statistical method. If a user should decide to compare the consensuses from GPA and MFA, which in practise is quite unlikely, he or she may face a dilemma of how to properly interpret the findings. One solution that may help to pick one of the methods is to investigate the residuals for each individual, i.e. the SR_k computed from each method.

Fig. 2.7 shows an example of what the $SR_{k,c}$ look like for each of the discussed methods. They are computed from data set 5 where 20 consumers tested 12 orange juice products. The plot clearly illustrates that GPA finds a consensus product configuration that represents all individuals well. It is important to remember that from permutation tests the GPA consensus configuration can be considered valid. The $SR_{k,GPA}$ of GPA vary very little compared to $SR_{k,MFA}$

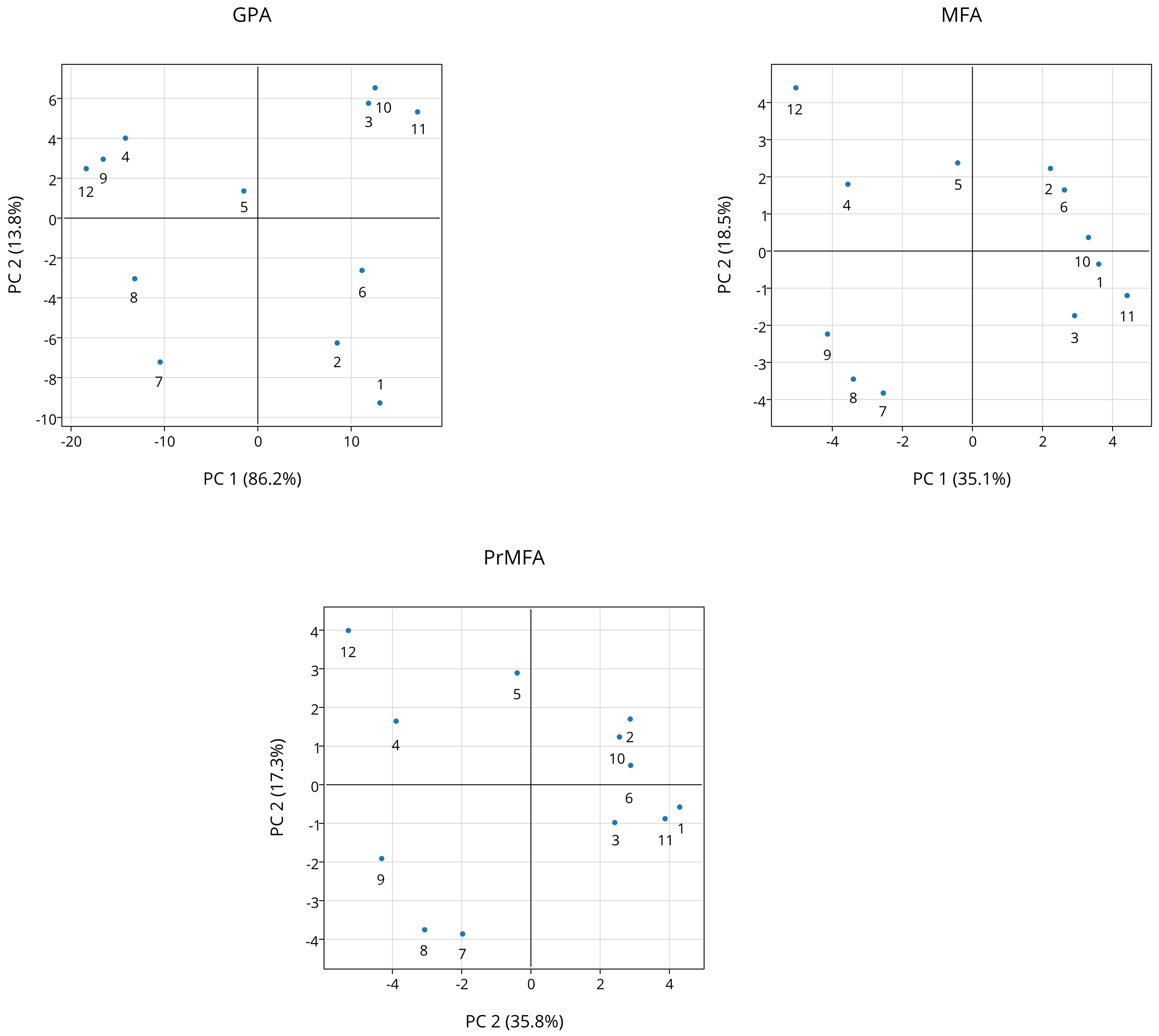

Figure 2.6: PCA scores $\mathbf{F}_{GPA}$, $\mathbf{F}_{MFA}$ and $\mathbf{F}_{PrMFA}$ of consensus product configurations from GPA, MFA and PrMFA for real world data set 5 and the respective individual projected scores $\mathbf{F}_{[k],GPA}$, $\mathbf{F}_{[k],MFA}$ and $\mathbf{F}_{[k],PrMFA}$ of the projected individual product configurations.

and $SR_{k,MFA}$ computed from MFA and PrMFA which means that for GPA every individual the difference or distance between the consensus product configuration and individual product configurations is relative similar. One could claim that every individual is about equally well represented by GPA. For $SR_{k,MFA}$ of MFA one can observe that the individual product configurations of consumer 3 and 4 are relatively different from the MFA consensus product configuration because $SR_{3,MFA}$ and $SR_{4,MFA}$ are very high compared to the other individuals. So are those of consumer 9 to 14, however, to a somewhat lesser degree. It is obvious that for the MFA consensus product configuration there are much stronger variations across the $SR_{k,MFA}$, which is reflected by its standard deviation (0.052 for GPA compared to 0.382 for MFA and 0.386 for PrMFA). The combination of the fact that $SR_{tot,c}$ and the standard deviation of $SR_{k,c}$ for GPA are smaller than those of MFA and PrMFA suggests that the GPA consensus product configuration may be the best choice for this data set in particular. A mixed model two-factor ANOVA was carried out to investigate whether differences between $SR_{k,c}$ from the three methods are significant or not. One of the two main factors represents the multiblock methods (GPA, MFA and PrMFA) and was set to be fixed. The other main factor represented the individuals (consumer 1 to 20) and was set to random. The ANOVA table (not shown) reports that the method factor is significant (p=0.033). A Tukey's test reveals, however, that differences are only close to being significant at 5% level (comparison between $SR_{k,c}$ of GPA and MFA: p=0.066; comparison between SR_k of GPA and PrMFA: p=0.052). Even though the differences between GPA, MFA and PrMFA are not significant one may argue that because of the lower $SR_{tot,c}$ and far lower standard deviation of $SR_{k,GPA}$ the GPA consensus product configuration might be the best choice for interpretation of the products. For the other data sets the effect of method was non-significant.

2.4 Conclusion

When analysing data from projective mapping experiments it is necessary to use a suitable statistical methods that provides answers regarding how products relate to each other and how the consensus product configuration fits overall across all consumers that were involved in the testing of the products. Projective mapping data is often analysed by either GPA or MFA, both being multiblock methods that handle data from many individuals. To the author's knowledge, there

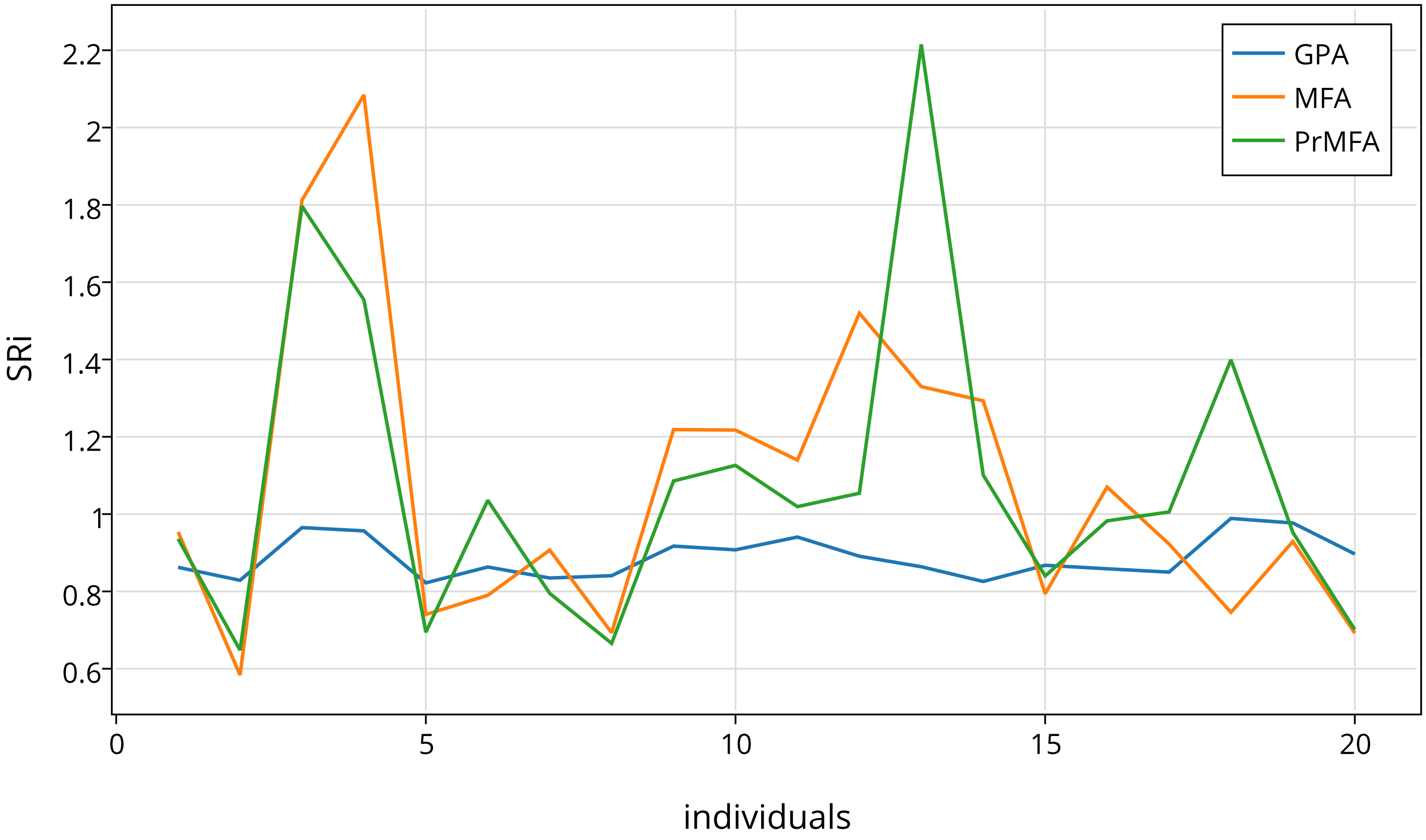

Figure 2.7: Individual $SR_{k,c}$ for for GPA, MFA and PrMFA for real world data set 5 (orange juice).

are no studies comparing the outcomes of the two methods in the special case of projective mapping and its typical two-variable multiblock data. This study attempts to provide insight into how results may vary when using one method or the other.

Monte Carlo simulations with random data suggest that GPA and MFA provide very similar consensus product configurations when only few products are tested. An average RV > 0.93 for 1000 simulations with only 4 products suggests that findings and interpretations may be very similar and that the choice of statistical method matters less. But with an increasing number of products the average RV coefficient drops and similarities decrease. This is where the choice of statistical method may lead to a different interpretation regarding the investigated products and problems arise. Furthermore, the simulations indicate that differences grow even larger the more consumers are included in the simulations.

Constructed data that were manipulated in specific ways gave insight to how GPA, MFA and PrMFA (a combination of GPA and MFA) handle different situations. In these simple cases GPA, MFA and PrMFA handled translation, rotation and reflection as well as the combination of these equally well. One important difference between GPA and MFA is that GPA obeys a restriction to keep relative object distances constant for each individual whereas MFA does not have such a restriction. This is an important difference between the two methods that may be responsible for varying outcomes of the respective consensus product configurations. In worst case this may lead to different conclusions regarding the tested products.

Analysis of nine real world data sets indicate that there is no clear conclusion regarding which methods is to be preferred over the other. GPA provided in 7 out of 9 data sets the consensus product configuration that best fit individual product configurations. For all of these 7 data sets the consensus configurations from GPA has been validated with a permutation test. Furthermore it must be noted that the similarity ratios SR_k between the consensus configuration and individual configurations were significantly smaller for GPA in only one out of these 7 cases. Tukey's test however showed that in this one case the pair-wise comparisons did not show a significant effect of method with p values just above 0.05. For two out of 9 data sets, MFA provided a consensus product configuration that best fit the individuals product configurations. Here for only one (wine data) out of the two cases the effect of methods was significant at 5% level. It important to mention that this was the case where the consensus configuration of GPA was not significant. PrMFA was least effective in finding a consensus configuration that represents all individuals in a best possible manner, although differences from GPA and MFA were never large or significant.

Since it is important to make the correct interpretation of the tested products, one may suggest to run both GPA and MFA and pick the method that provides the significantly lowest $SR_{k,c}$. If $SR_{k,c}$ from each method are not significant different from each other, one may go for the method that has the lowest total $SR_{tot,c}$ across individuals combined with the lowest standard deviation of $SR_{k,c}$. If none of the two described situations is given it is difficult to decide which statistical method to choose. For the nine real world data sets analysed in only one case none of the two above described situations was given. Eventually, due to practical reasons, computing time for each method might be decisive. During the Monte Carlos simulations it became apparent that GPA computations took substantially longer than MFA. In the most extreme cases it took about up to 120 times longer to compute the consensus product configuration for GPA (up to 10 minutes) than for MFA (approx. 5 seconds).

Acknowledgements

The author would like to thank Francois Husson for kindly providing Napping data on his web page at Agrocampus Ouest.

Chapter 3

Code for Computations

Python and R code for Monte Carlo simulations with random data linked through PypeR

```
# -*- coding: utf-8 -*-
#==============================================================================
# Author:
# Oliver Tomic
#
# Purpose:
# 1. Generate random projective mapping data for different situations:
# - different number of products
# - different number of consumers
#
# 2. Compute consensus with GPA and MFA (with R FactoMineR via PypeR)
#
# 3. Check with RV coefficient how consensus from GPA and MFA are alike or
#    different for different situations
#==============================================================================

#==============================================================================
# Import necessary modules
#==============================================================================
import numpy as np
import random as rd
import statTools as st
import pyper
import time
import datetime
```

```
#==============================================================================
# Clock in starting time
#==============================================================================
startTime = time.time()


#==============================================================================
# Set parameters for generating random napping data for a number of individuals
# and a number of products.
#==============================================================================

# Set range of products (min to max)
minProd = 4
maxProd = 16

# Set range of consumers (min to max)
minConsum = 20
maxConsum = 100

# Print out for convenience
print ’minimum number of consumers: ’, minConsum
print ’minimum number of products: ’, minProd
print
print ’maximum number of consumers: ’, maxConsum
print ’maximum number of products: ’, maxProd

# Set number of simulation runs
numRuns = 1000


#==============================================================================
# Set up pyper (Python - R interface) and load R packages
#==============================================================================

# Set up Pyper accessing R packages and their functions
r = pyper.R()

# Import R FactoMineR
r(’library(FactoMineR)’)
print r(’sessionInfo()’)


#==============================================================================
# STEP 1: Run simulation for varous numbers of products and consumers
#==============================================================================

# Define lists with range of how many consumers and products are to be used in
# simulation. Later the script goes systematically through all possible
# combinations between these two lists.
consumRange = range(minConsum, maxConsum + 1)
prodRange = range(minProd, maxProd + 1)

# Initiate zero-arrays that will hold all average RV-coefficients across
# chosen number of simulations for each product-consumer combination
rv_coeffArr = np.zeros((len(consumRange), len(prodRange)))
totalSims = np.size(rv_coeffArr)

# Loop through all possible combiantions of products and consumers.
for ind, element in np.ndenumerate(rv_coeffArr):
    print; print
    print ind, 'consumers', consumRange[ind[0]], '- products', prodRange[ind[1]]

    # Generate column names for R data frame
    colNamesList = []
    for cons in range(consumRange[ind[0]]):

        xName = 'X{0}'.format(cons+1)
        yName = 'Y{0}'.format(cons+1)
        colNamesList.append(xName)
        colNamesList.append(yName)

    # Generate specific strings for R command that will be run through pyper
    # Make string for group
    comma = ','
    tempGroup = '2' * consumRange[ind[0]]
    group = comma.join(tempGroup)

    # Make string for group and name.group (used later in R command)
    nameGroup = '"Gc1"'
    sGroup = '"s"'
    for con in range(2,consumRange[ind[0]]+1):
        partStr = ',"Gc{0}"'.format(str(con))
        nameGroup += partStr
        sPartStr = ',"s"'
        sGroup += sPartStr

    # Collect GPA and MFA results
    gpaConsList = []
    gpaRVList = []

    mfaConsList = []
    mfaRVList = []

    # Collect RV coefficient from all simulation runs so that average RV
    # coefficient across all runs can be computed and stored in rv_coeffArr.
```

```
  rvSimList = []

  # Do 100 runs to get a representative number of coefficients.
  for run in range(numRuns):

    arrList = []
    for cons in range(consumRange[ind[0]]):

      arr = np.zeros((prodRange[ind[1]], 2))
      for prod in range(prodRange[ind[1]]):

        x = rd.uniform(1,59)
        y = rd.uniform(1,39)

        arr[prod,0] = x
        arr[prod,1] = y

      arrList.append(arr)


#=========================================================================
# STEP 2: compute consensus with GPA and MFA via R and pyper
#=========================================================================

    # First stack data from all consumers into one array
    data = np.hstack(arrList)

    # Construct data frame in R. GPA and MFA only take data frames as
    # input, not matrices.
    r[’data’] = data
    r[’varNames’] = colNamesList
    r(’dataDF <- as.data.frame(data)’)
    r(’colnames(dataDF) <- varNames’)

    # Run GPA
    r(’resGPA <- GPA(dataDF, group=c({0}), \
      name.group=c({1}))’.format(group, nameGroup))
    GPA_cons = r[’resGPA$consensus’]
    GPA_RV = r[’resGPA$RV’]

    gpaConsList.append(GPA_cons)
    gpaRVList.append(GPA_RV)

    # Run MFA
    r(’resMFA<-MFA(dataDF, group=c({0}), type=c({1}), ncp=5, \
      name.group=c({2}), num.group.sup=c(), \
      graph=FALSE)’.format(group, sGroup, nameGroup))
    MFA_cons = r[’resMFA$ind$coord’]
```

```
        MFA_RV = r['resMFA$group$RV']

        mfaConsList.append(MFA_cons)
        mfaRVList.append(MFA_RV)


#==============================================================================
# STEP 3: compute RV coefficient between GPA and MFA consensus
#==============================================================================

        # Compute RV coefficient between GPA and MFA consensus
        res_RV = st.RVcoeff([GPA_cons,MFA_cons[:,:2]])
        rvSimList.append(res_RV[0,1])

        np.savetxt('rv_simlist.txt', np.array(rvSimList), fmt='%.7f', \
            delimiter='\t')

    # Store average RV in array for different combinatios and consumers
    meanRV = np.average(rvSimList)
    rv_coeffArr[ind] = meanRV

    # Compute how far computations have progressed
    currentSims = float(ind[0] * (maxProd - minProd + 1) + ind[1]+1)
    progress = currentSims / totalSims * 100
    print 'PROGRESS: ', progress

    # Save rv_coeffArr after finishing one simulation in case of crash. In
    # this way at least the computed results are preserved.
    np.savetxt('RESULT.txt', rv_coeffArr, fmt='%.3f', delimiter='\t')


#==============================================================================
# Clock in end time and compute total time used
#==============================================================================
endTime = time.time()
totalTime = endTime - startTime
print 'Total time used:', totalTime
```

Python Code for Computation of Similarity Ratios $SR_{k,c}$ and $SR_{tot,c}$

```
# -*- coding: utf-8 -*-
#==============================================================================
# Author:
# Oliver Tomic
#
# Purpose:
# Compute individual similarity ratios SRk, their sum SRtot and STD for GPA,
# MFA and  PrMFA. Plot SRk and SRtot.
#==============================================================================

#==============================================================================
# Import necessary modules
#==============================================================================
import numpy as np
import pandas as pd
import statTools as st
import matplotlib.pyplot as plt


#==============================================================================
# Set parameters for number of products and consumers and load data from file
#==============================================================================
# Set parameters for number of products and number of users
numProds = 10
numUser = 18

# Load consensus product configurations data from text file
df_cons_GPA = pd.read_table(’GPA_cons.txt’, sep=’\t’, index_col=0)
cons_GPA = df_cons_GPA.values

df_cons_MFA = pd.read_table(’MFA_cons.txt’, sep=’\t’, index_col=0)
cons_MFA = df_cons_MFA.values

df_cons_PrMFA = pd.read_table(’PrMFA_cons.txt’, sep=’\t’, index_col=0)
cons_PrMFA = df_cons_PrMFA.values

# Compute RV for all consensus arrays
rv = st.RVcoeff([cons_GPA, cons_MFA, cons_PrMFA])

# Load individual GPA factor scores text file and store in a list
df_ind = pd.read_table(’GPA_ind.txt’, sep=’\t’, index_col=0)
temp_ind = df_ind.values

tempArr = np.hstack(np.split(temp_ind, numProds))
indArrList = []
```

```
for userInd in range(numUser):
    indArr = tempArr[userInd,:].reshape(numProds,2)
    indArrList.append(indArr)

indArrList_GPA = indArrList[:]

# Load individual MFA factor scores text file and store in a list
df_ind = pd.read_table(’MFA_ind.txt’, sep=’\t’, index_col=0)
temp_ind = df_ind.values

tempArr = np.hstack(np.split(temp_ind, numProds))
indArrList = []

for userInd in range(numUser):
    indArr = tempArr[userInd,:].reshape(numProds,2)
    indArrList.append(indArr)

indArrList_MFA = indArrList[:]

#==============================================================================
# Import individual projected scores, reorganise for further computations
#==============================================================================
# Load individual PrMFA factor scores from text file and store in a list
df_ind = pd.read_table(’PrMFA_ind.txt’, sep=’\t’, index_col=0)
temp_ind = df_ind.values

tempArr = np.hstack(np.split(temp_ind, numProds))
indArrList = []

for userInd in range(numUser):
    indArr = tempArr[userInd,:].reshape(numProds,2)
    indArrList.append(indArr)

indArrList_PrMFA = indArrList[:]

# Now compute ratios between consensus and individual product configurations
ss_cons_GPA = np.sum(np.square(cons_GPA))
ss_cons_MFA = np.sum(np.square(cons_MFA))
ss_cons_PrMFA = np.sum(np.square(cons_PrMFA))

# for GPA
ratioList = []
for userInd in range(numUser):
    indRatio = np.sum(np.square(cons_GPA - indArrList_GPA[userInd])) / \
        ss_cons_GPA
    print userInd, indRatio
    ratioList.append(indRatio)
```

```
ratioList_GPA = ratioList[:]

# for MFA
ratioList = []
for userInd in range(numUser):
    indRatio = np.sum(np.square(cons_MFA - indArrList_MFA[userInd])) / \
        ss_cons_MFA
    print userInd, indRatio
    ratioList.append(indRatio)

ratioList_MFA = ratioList[:]

# for PrMFA
ratioList = []
for userInd in range(numUser):
    indRatio = np.sum(np.square(cons_PrMFA - indArrList_PrMFA[userInd])) / \
        ss_cons_PrMFA
    print userInd, indRatio
    ratioList.append(indRatio)

ratioList_PrMFA = ratioList[:]


#===============================================================================
# Plot results
#===============================================================================
# Plot individual SRi ratios in matplotlib
ratioArr_GPA = np.array(ratioList_GPA)
ratioArr_MFA = np.array(ratioList_MFA)
ratioArr_PrMFA = np.array(ratioList_PrMFA)

fig = plt.figure()
ax = fig.add_subplot(111)
ax.plot(ratioArr_GPA, 'b', label='GPA', linewidth=2)
ax.plot(ratioArr_MFA, 'r', label='MFA', linewidth=2)
ax.plot(ratioArr_PrMFA, 'g', label='PrMFA', linewidth=2)

ax.legend()
ax.set_ylim(0)

plt.show()

# Plot SRtotals in matplotlib
total_GPA = np.sum(ratioList_GPA)
total_MFA = np.sum(ratioList_MFA)
total_PrMFA = np.sum(ratioList_PrMFA)
totalList = [total_GPA, total_MFA, total_PrMFA]
```

```
std_GPA = np.std(ratioList_GPA)
std_MFA = np.std(ratioList_MFA)
std_PrMFA = np.std(ratioList_PrMFA)
stdList = [std_GPA, std_MFA, std_PrMFA]

fig = plt.figure()
ax = fig.add_subplot(111)

methods = ('GPA', 'MFA', 'PrMFA')
y_pos = np.arange(len(methods))
performance = np.array(totalList)
error = np.random.rand(len(methods))

plt.barh(y_pos, performance, xerr=stdList, align='center', alpha=0.4)
plt.yticks(y_pos, methods)
plt.xlabel('sum of ratios')
plt.title('Sum and STD of ratios')

plt.show()
```

Bibliography

Abdi, H. and Williams, L. J. (2010). Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2:433–349.

Abdi, H., Williams, L. J., and Valentin, D. (2013). Multiple factor analysis: principal component analysis for multitable and multiblock data sets. *Wiley Interdisciplinary Reviews: Computational Statistics*, 5:149–179.

Dehlholm, C., Brockhoff, P. B., Meinert, L., Aaslyng, M. D., and Bredie, W. L. (2012). Rapid descriptive sensory methods - comparison of free multiple sorting, partial napping, napping, flash profiling and conventional profiling. *Food Quality and Preference*, 26:267–277.

Dijksterhuis, G. (1996). *Procrustes Analysis in Sensory Research*, volume 16 of *Data Handling in Science and Technology*. Elsevier Science, multivariate analysis of data in sensory science edition.

Escofier, B. and Pagès, J. (1994). Multiple factor analysis. *Computational Statistics & Data Analysis*, 18:121–140.

Gower, J. C. (75). Generalized procrustes analysis. *Psychometrika*, 40:33–51.

Husson, F. Data sets. http://math.agrocampus-ouest.fr/infoglueDeliverLive/membres/Francois.Husson/data.sets. Accessed: 2013-12-07.

King, B. and Arents, P. (1991). A statistical test of consensus obtained from Generalized Procrustes Analysis of sensory data. *Journal of Sensory Studies*, 6:37–48.

Lawless, H. T. (2013). *Quantitative Sensory Analysis - Psychophysics, Models and Intelligent Design*. Wiley, Chichester, UK.

Lawless, H. T. and Heymann, H. (2010). *Sensory Evaluation of Food - Principles and Practices*. Springer, NY, USA, 2nd edition edition.

Lê, S., Josse, J., and Husson, F. (2008). FactoMineR: An R package for multivariate analysis. *Journal of Statistical Software*, 25:1–18.

Martens, H. and Martens, M. (2001). *Multivariate analysis of Quality: An Introduction*. Wiley, Chichester.

Martens, H. and Næs, T. (1989). *Multivariate Calibration*. John Wiley & Sons Ltd, Chichester.

Morand, E. and Pagès, J. (2006). Procrustes multiple factor analysis to analyse the overall perception of food products. *Food Quality and Preference*, 17:36–42.

Næs, T., Brockhoff, P. B., and Tomic, O. (2010). *Statistics for Sensory and Consumer Science*. Wiley, Chichester.

Nestrud, M. and Lawless, H. (2011). Recovery of subsampled dimensions and configurations derived from napping data by mfa and mds. *Attention, Perception and Psychophysics*, 73:1266–1278.

Nestrud, M. A. and Lawless, H. T. (2008). Perceptual mapping of citrus juices using projective mapping and profiling data from culinary professionals and consumers. *Food Quality and Preference*, 19:431–438.

Oliphant, T. E. (2007). Python for scientific computing. *Computing in Science and Engineering*, 9:10–20.

Pagès, J. (2005). Collection and analysis of perceived product inter-distances using multiple factor analysis: Application to the study of 10 white wines from the Loire valley. *Food Quality and Preference*, 16:642–649.

Perrin, L., Symoneaux, R., Maitre, I., Asselin, C., Jourjon, F., and Pagès, J. (2008). Comparison of three sensory methods for use with the napping procedure: Case of ten wines from loire valley. *Food Quality and Preference*, 19:1–11.

Risvik, E., McEwan, J. A., Colwill, J. S., Rogers, R., and Lyon, D. H. (1994). Projective mapping: A tool for sensory analysis and consumer research. *Food Quality and Preference*, 5:263–269.

Risvik, E., McEwan, J. A., and Rødbotten, M. (1997). Evaluation of sensory profiling and projective mapping data. *Food Quality and Preference*, 8:63–71.

Robert, P. and Escoufier, Y. (1976). A unifying tool for linear multivariate statistical methods: The RV-coefficient. *Applied Statstics*, 25:257–265.

Smilde, A. K., Kiers, H. A. L., Bijlsma, S., Rubingh, C. N., and van Erk, M. J. (2009). Matrix correlations for high-dimensional data: the modified RV-coefficient. *Bioinformatics*, 25:401–405.

Tomic, O., Forde, C., Delahunty, C., and Næs, T. (2013). Performance indices in descriptive sensory analysis - a complimentary screening tool for assessor and panel performance. *Food Quality and Preference*, 28:122–133.

Valentin, D., Chollet, S., Lelièvre, M., and Abdi, H. (2012). Quick and dirty but still pretty good: a review of new descriptive methods in food science. *International Journal of Food Science and Technology*, 47:1563–1578.

Varela, P. and Ares, G. (2012). Sensory profiling, the blurred line between sensory and consumer science. A review of novel methods for product characterization. *Food Research International*, 48:893–908.

Wakeling, I., Raats, M., and MacFie, H. M. (1992). A new significance test for consensus in generalized Procrustes analysis. *Journal of Sensory Studies*, 7:91–96.

Wall, M. E., Rechtsteiner, A., and Rocha, L. M. (2003). *A Practical Approach to Microarray Data Analysis*. Springer.

Xia, X.-Q., McClelland, M., and Wang, Y. (2010). PypeR, a Python package for using R in Python. *Journal of Statistical Software*, 35:1–8.