

**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**



ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS

**SCHOOL OF INFORMATION SCIENCES  
& TECHNOLOGY**

**DEPARTMENT OF STATISTICS**

**POSTGRADUATE PROGRAM**

**‘Imputation Methods Based On  
Principal Component Analysis’**

By

**Xristos Konstadinos Siskas**

A THESIS

Submitted to the Department of Statistics  
of the Athens University of Economics and Business  
in partial fulfilment of the requirements for  
the degree of Master of Science in Statistics

Athens, Greece  
September 2016



**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**



ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS

## **ΣΧΟΛΗ ΕΠΙΣΤΗΜΩΝ & ΤΕΧΝΟΛΟΓΙΑΣ ΤΗΣ ΠΛΗΡΟΦΟΡΙΑΣ**

**ΤΜΗΜΑ ΣΤΑΤΙΣΤΙΚΗΣ**

**ΜΕΤΑΠΤΥΧΙΑΚΟ**

**‘Μέθοδοι που χρησιμοποιούνται για την  
αναπλήρωση ελλειπουσών τιμών και βασίζονται  
στην Ανάλυση Κύριων Συνιστωσών’**

Χρήστος Κωνσταντίνος Σίσκας

ΔΙΑΤΡΙΒΗ

Που υποβλήθηκε στο Τμήμα Στατιστικής  
του Οικονομικού Πανεπιστημίου Αθηνών  
ως μέρος των απαιτήσεων για την απόκτηση  
Μεταπτυχιακού Διπλώματος Ειδίκευσης στη Στατιστική

Αθήνα  
Σεπτέμβριος 2016



## ACKNOWLEDGEMENTS

This thesis has been written in the context of the postgraduate program in statistics of the Athens University of Economics and Business. Before the presentation of this thesis i would like to thank some people, because without their help it would be difficult to write it and complete it. First of all I would like to thank the supervisor professor of this thesis, Ioulia Papageorgiou for her important help and support. Also, i would like to thank the professors who agreed to be members of the evaluation committee. Last, i would like to thank my parents Konstadino and Maria and my brother Vretto who gave me patience and strength to complete this thesis.



## ABSTRACT

Principal Component Analysis is the oldest and most famous technique of Multivariate Analysis and can be used as a tool for researchers to deal with missingness in datasets. The aim of this thesis is the description, the analysis and the comparison of the techniques that belong in the category of Principal Component Analysis. All these available techniques are presented with respect to their theoretical framework and then a comparison of these methods in different percentages of missingness and for different types of datasets (simulated and real) follows in order to see which method responds better depending on the case and which is totally the most reliable.



## ΠΕΡΙΛΗΨΗ

Η Ανάλυση Κύριων Συνιστωσών είναι μέθοδος της Πολυμεταβλητής Ανάλυσης και μπορεί να χρησιμοποιηθεί από τους ερευνητές σαν ένα εργαλείο με το οποίο μπορεί να αντιμετωπιστεί η ύπαρξη ελλειπουσών τιμών σε αρχεία δεδομένων. Η διπλωματική αυτή έχει σαν στόχο την περιγραφή, την ανάλυση και την σύγκριση των τεχνικών εκείνων που βασίζονται στην Ανάλυση Κύριων Συνιστωσών και χρησιμοποιούνται για την δημιουργία ενός ολοκληρωμένου αρχείου δεδομένων. Όπως αναφέραμε, όλες αυτές οι τεχνικές αναλύονται σ' αυτήν τη διπλωματική αρχικά θεωρητικά και στη συνέχεια με τη χρήση προσομοιωμένων και πραγματικών δεδομένων συγκρίνονται μεταξύ τους σε περιπτώσεις με διαφορετικά ποσοστά ελλειπουσών τιμών, έτσι ώστε να δούμε ποια μέθοδος ανταποκρίνεται καλύτερα σε κάθε περίπτωση αλλά και ποια είναι συνολικά πιο αξιόπιστη.



# Contents

<b>1 Introduction</b> .....	8
<b>2 Non-Response</b>	
2.1 Non-Response and Consequenses .....	11
2.2 Unit and Item Non-Response.....	13
2.3 MCAR, MAR and MNAR Non-Response.....	15
2.4 Dealing with Non-Response.....	17
2.4.1 Weighting class adjustments.....	17
2.4.2 Post Stratification.....	19
2.4.3 Propensity Modelling.....	19
2.4.4 Imputation.....	20
<b>3 Imputation</b>	
3.1 Single Imputation and Methods.....	22
3.1.1 Mean Imputation.....	23
3.1.2 Hot Deck Imputation.....	24
3.1.3 Ratio Imputation.....	25
3.1.4 Regression Imputation.....	25
3.1.5 Cold Deck Imputation.....	26
3.1.6 Principal Component Analysis.....	26
3.1.7 Multiple Imputation.....	27
3.2 Applying Methods of Imputation.....	28
3.2.1 Applying Mean Imputation.....	29
3.2.2 Applying Hot Deck Imputation.....	29
3.2.3 Applying Regression Imputation.....	30
3.2.4 Applying Multiple Imputation.....	30
<b>4 Principal Components Analysis</b>	
4.1 Definition of Principal Components and Principal Component Analysis.....	33



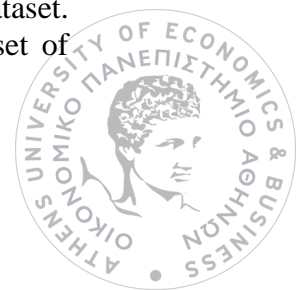
4.2 Non Linear Principal Component Analysis.....	35
4.3 Comparison of Principal Component Analysis and Non Linear Principal Component Analysis.....	36
4.4 Singular Value Decomposition.....	37
4.5 Main Models in Principal Component Analysis.....	37
4.5.1 The Fixed Effect Model.....	38
4.5.2 The Random Effect Model.....	39
4.6 Techniques of Imputation in Principal Component Analysis.....	40
4.6.1 The Iterative Principal Component Analysis Algorithm.....	40
4.6.2 The Multiple Imputation in Principal Component Analysis.....	42
4.6.3 The Forward Imputation Applying Principal Component Analysis.....	43
4.6.4 The Forward Imputation Applying Non Linear Principal Component Analysis.....	45
4.6.5 Factorial Analysis for Mixed Data.....	46
4.6.6 The Nipals Algorithm.....	49
<b>5 Experiments in R</b>	
5.1 Data with continuous variables.....	52
5.1.1 Uncorrelated Data From The Multivariate Normal.....	54
5.1.2 Correlated Data From The Multivariate Normal.....	56
5.1.3 Data From The Multivariate Skew Normal.....	58
5.1.4 A Real Dataset With Continuous Variables.....	60
5.2 Mixed Type Of Data.....	63
5.2.1 Mixed Real Dataset Iris.....	64
<b>6 Conclusions</b>	
6.1 Conclusions With A Critical View.....	67
<b>Appendix</b> .....	69
<b>Bibliography</b> .....	78



# Chapter 1

## Introduction

In many surveys the desirable information from the sample, or the population, is not completely collected. If we suppose that this information collects and stores in matrices, this means that the most matrices have empty cells. An incomplete dataset makes very difficult the job of a researcher, since he can't draw accurate results. So, the first thing that someone has to do, before continue with the statistical analysis, is to find the lost information and fill in the empty cells in order to create a fully observed dataset. In other words, either the survey must be repeated, either the lost information must be found in a kind of way. Beyond the cost and the difficulties that has a repetition of a survey, or the foundation of the lost information, even if this is achieved the dataset will never be complete, due to the fact, that non-response will occur for other reasons possibly. So, it must be implemented a technical approach, which has no sampling cost and help the analysis by creating fast and painless a full observed dataset. This very important approach is the technique of imputation. Imputation is the method where each missing value fills in with a substitute. There are many methods that could be used to create artificial values for the missing one, depending the survey and the conditions that we have in our analysis. The main aim of this thesis is a brief representation of the methods and the techniques of imputation that belong in the category of Principal Component Analysis. Principal Component Analysis is the oldest and most famous technique of Multivariate Analysis and it was used widely for data analysis. It was introduced by Pearson (1901) and developed by Hotelling (1933). The main idea of this technique is to reduce the dimensionality of a data set. The new dataset, has fewer variables, while retain as much as possible of the variation present in the initial dataset. This aim is achieved with a transformation of the original variables to a new set of



variables the principal components. The principal components are uncorrelated variables and there are ordered in descending order with respect to their variances.

Principal Component Analysis, as we mention, is a method of multivariate analysis that commonly used in data analysis, but can be also used as a tool for researchers to deal with missingness in datasets. In all imputation approaches we present in this thesis, the methodology originates from PCA. In many surveys, depending the percentage of missing values and the type of the variables, different approaches can be used to fill in the empty cells of a matrix and create a new full observed dataset. All the potential approaches, the advantages and the disadvantages of each approach, the way and the conditions that needed to implement all these methods and the results we take, will be discussed in detail. In the subparagraph below, is available the structure and the contents of each chapter in this thesis.

Chapter 2 of this thesis begins with the important problem of non-response. The meaning of non-response, how often happens, how much affect our analysis and the difficulties that a researcher has when information is missing, are some issues of discussion. Then a division of non-response follows. We separate non-response depending on the type of missingness we have in our analysis. First the different cases of missingness concerning whether we have Unit or Item non-response, i.e if the desirable information be missing from the entire unit of the population, or the data had collected partially from that unit. Also, the mechanisms that missing values could be produced, i.e if the missing data happen at random, or a pattern exists and missing values produced with a specific rule. At the same point, we give some examples in order to make all the above differences understandable. Then, we proceed with a reference to the ways that non-response can be handled accordingly. In Chapter 3, we analyze the main techniques of Single Imputation and we make a small reference to the approach of Multiple Imputation. The techniques of Mean Imputation, Hot Deck Imputation, Ratio Imputation, Regression Imputation and Cold Deck Imputation are analyzed further. At the end of the Chapter 3, we use some examples where the different methods of imputation are implemented and the results are annotated. In Chapter 4, which is the theoretical part of this thesis, we give first a brief description of PCA definition and component derivation. Subsequently, the definition of Non Linear Principal Component Analysis and the comparison of Principal Component Analysis and Non Linear Principal Component Analysis follows. Also, two main models are analyzed, the Fixed Effect model and the Random Effect model. The main theoretical part of the thesis follows where the available techniques which used to impute missing datasets affected by missing values and belongs in the category of Principal Component Analysis, are described in detail. The Iterative PCA Algorithm, the Forward Imputation, the Factorial Analysis for mixed datasets and the Nipals Algorithm are presented with respect to their theoretical framework, the properties they hold and a description of how these methods can be implemented (packages availability e.t.c). Chapter 5 consists of the computational part of this thesis, where the methods analyzed in Chapter 4 are implemented in different incomplete data scenarios. A comparison of the available approaches in different percentages of missingness and for different types of variables is examined. For the analysis in this chapter, we used simulated and real datasets in which we create missing values with a different percentage each time. More specifically we have simulated from the Multivariate Normal distribution and the Skew Normal distribution and we used two real datasets. The first real dataset contains only continuous variables and the second four continuous and one categorical variable. In all



the above datasets we will create 5%, 10% and 20% missingness. With boxplots and tables, we will summarize the results and we will compare the different approaches. Also, all the available packages and the commands of statistical package R that have been used to the analysis, are reported in detail. Chapter 6 contains all the conclusions of this thesis, from a critical point of view. Finally, appendices with R code and the references are provided in the end of the thesis.

After giving the purpose and the general structure of this thesis, we can refer to the required knowledge that someone must have, to read it and comprehend it. A mathematical background is necessary, especially for the Chapter 4, where the imputation methods originated from Principal Component Analysis are given using complicated, in most cases, mathematical equations. Also, a statistical background is needed. The knowledge of main characteristics of statistical analysis will help someone to read more easily this thesis.



# Chapter 2

## 2.1 Non-Response and Consequences

One of the most important problems in many censuses and sample surveys is non-response. It is the phenomenon that the desirable information is not collected due to the fact that some of the units contacted do not respond to at least some items being asked. The problem created by survey non-response is that data values intended by survey design to be observed, are in fact missing.

It has been demonstrated that the consequences of these missing values are very harmful for sample surveys and can have large effects on the results of the statistics to be computed. Firstly, one effect of non-response is that it reduces the sample size. The reduced size leads to less efficient estimates and the precision of estimators will be smaller, while at the same time margins of error will be larger. Also, a more serious effect of non-response is that it can be selective. This means that possible biases exist because the respondents are often systematically different from non-respondents. This occurs if specific groups are over or under represented in the survey. If these groups behave differently we get biased estimators. More generally, estimates are significantly too high or too low. For example, if men participate in a survey to a greater extent than women, then a random sample of persons will have an overrepresentation of men among the respondents. This will lead to biased estimates for the whole population total of any variable, such as income, where men have usually higher on average than women.

Suppose the population is being divided into two strata, the respondents and non-respondents. Let denote with  $N_R$  the number of population respondents and with  $N_M$  the population nonrespondents, where both  $N_R$  and  $N_M$  are unknown. Then in Table 2.1 we



have the size, the mean and the variance for the respondents, the non-respondents and the whole population.

Table 2.1: Table with the size, the mean and the variance of the respondents and the non-respondents.

Stratum	Size	Mean	Variance
Respondents	$N_R$	$\bar{y}_{RU}$	$S_R^2$
Nonrespondents	$N_M$	$\bar{y}_{MU}$	$S_M^2$
Population	$N$	$\bar{y}_U$	$S^2$

When we take a sample from the population will contain some respondents and some non-respondents. If the respondents differ from non-respondents, or more specifically the population mean of respondents  $\bar{y}_{RU}$  differ from the population mean of non-respondents  $\bar{y}_{MU}$ , the estimate of population mean using only the stratum of respondents will produce bias. We can write the below formula for the population mean:

$$\bar{y}_U = \frac{N_R}{N} \bar{y}_{RU} + \frac{N_M}{N} \bar{y}_{MU} \quad (2.1)$$

Also let  $\bar{y}_R$  be an approximately unbiased estimator of the mean in the respondent stratum. It results that approximately the producing bias from the difference of the two strata is:

$$E[\bar{y}_R] - \bar{y}_U \approx \frac{N_M}{N} (\bar{y}_{RU} - \bar{y}_{MU}) \quad (2.2)$$

The effect of Non-Response depends on the proportion of non-respondents and the difference between the means of the potential non-respondents and the respondents. From (2.2) perceived that if  $N_M/N$  is small, or the mean of respondents are close enough to the mean of non-respondents, then the resulting bias is small.

Example 2.1: Numerical example of produced bias.

We suppose that from an area containing 1000 households, a sample of 100 households obtained using Simple Random Sampling, for estimating the proportion of the households that do not have garden or yard. Also, suppose that 10% (i.e. 100 households) would refuse to cooperate in the survey, or would not be reachable and they would not included in the sample. Thus, between the 1000 households in the population there exist 100 potential nonresponding households and 900 potential responding households. Finally, suppose that 10% from the nonresponding households (i.e. 10 households) and 20% from the responding households (i.e. 180 households) do not have garden or yard. In the entire population of the 1000 households, 190 households do not have garden or yard. In this example:



- $N = 1000$  ,  $N_R = 900$  ,  $N_M = 100$
- $\bar{y}_{RU} = 0.2$  ,  $\bar{y}_{MU} = 0.1$

And from equation (2.2), the produced bias is:

$$Bias = \frac{100}{1000}(0.2 - 0.1) = 0.01$$

## 2.2 Unit and Item Non-Response

Initially we can categorize non-response in two types. The first type is called Unit non-response and occurs when data are missing from the whole population unit. In contrast with the first type, the second type of non-response, which called Item non-response, occurs when the population unit was reached but data has been collected only partially. In a survey of persons, Unit non-response means that the person provides no information for the survey and Item non-response that the person does not respond to a particular item on the questionnaire. Briefly, we present the main reasons for Unit and Item non-response.

The main reasons for Unit non-response are the followings:

1. Failure of the data collector to identify the sample unit.
2. Failure to make contact with the sample unit.
3. Refusal of the sample unit to participate.
4. Inability of the sample unit to participate (i.e. illness, health, absence)
5. Inability of the data collector and sample unit to communicate (e.g. language)
6. Accidental loss of the data.

The main reasons for Item non-response are the followings:

1. Refusal to provide an answer.
2. Inability to provide answer.
3. Other failure to answer.
4. Provided answer being of inadequate quality (e.g. incomplete, implausible, failing an edit, consistency, check)

Subsequently we will mention two examples as formulated in the book of Donald B. Rubin, “Multiple Imputation for Non-Response in Surveys”, where becomes comprehensible the difference between Unit and Item non-response.



### Example 2.2: Educational Testing Service's Sample Survey of Schools.

In 1971 conducted a sample survey of 660 schools by the Educational Testing Service (ETS), for the purpose of studying their compensatory reading programs. The desirable information on types of compensatory reading programs and the achievement levels of students in the schools were obtained from a questionnaire which sent to the principals. From the 660 principals only 472 had returned this questionnaire, despite the fact that all had been contacted by telephone and had been mailed with several reminds. Non-response on this questionnaire is called Unit non-response, because the unit in this survey, which is the non-responding principal, refused to respond to any items of the questionnaire. If the principals had responded to some items from the questionnaire, then we would say that the survey had suffered from Item non-response.

The high non-response rate was considered to be a serious problem and concern developed that the 188 non-responding principals were systematically different from the 472 responding principals. Maybe, the 188 non-respondents having students with more serious reading problems or having reading programs that were less effective.

Generally dealing with non-response presupposes adjusted estimates. Due to the fact, that most of the times respondents differ from non-respondents, expanded standard errors of estimates reflect the reduced sample size and the differences between respondents and non-respondents. Sensitive estimates and standard errors above groups maybe are different on unobserved variables.

In this example, a way to deal with non-response is to discard the 188 principals. But, this will result in bias, since respondents are different from non-respondents. For example, if the non-respondents have lower-achieving students, analysis based on the respondents only, will overestimate the typical achievement of students in compensatory reading programs. As we will see later in this chapter, another way to have efficient estimates is to weight the responding units to compensate for the non-responding units, but weighting approaches are confined to problems of Unit non-response. Last, another procedure which could be followed is to simply fill the missing data with the mean from the 472 responding schools for each corresponding variable. This technique may not be appropriate and could be lead to biased estimators, because there are exist differences between the values of one group and the others.

### Example 2.3: Current Population Survey and Missing Incomes.

The Current Population Survey (CPS), conducted by the Census Bureau, used 50.000 households to obtain information about the monthly income. The non-response rate was 15-20% on many income items. It is certainly possible that respondents differ systematically from non-respondents, because low or high income people tend to refuse to answer to many items of the survey than middle income people. The non-response in this example constitutes Item non-response, because each unit included in the survey produces information on most items, but some units fail to provide information to some particular items. Those who refused to answer at all, create Unit non-response.

In addition to the first example, a method that we could use to deal with non-response here and which we will analyze later in a separate paragraph, is the technique of Imputation. More specifically, a "hot deck" imputation procedure. This approach, finds for each non-respondent a matching respondent, where matching means close with respect to variables observed for both.



## 2.3 MCAR, MAR and MNAR Non-Response

Now suppose that we have a sample of  $n$  units from a population. Let  $Y_i = (Y_{i,1}, Y_{i,2}, \dots, Y_{i,p})^T$  denote the  $p$  variables that we intended to collect from the  $i^{th}$  unit, where  $i = 1, \dots, n$ . These data will be used to make inferences about a set of  $p$  population parameters  $\theta = (\theta_1, \theta_2, \dots, \theta_p)^T$ . Also, consider that  $Y_{i,M}$  denote the subset of  $p$  variables that are missing and  $Y_{i,O}$  the observed subset, for each unit  $i = 1, \dots, n$ . Furthermore for each individual of the sample ( $i = 1, \dots, n$ ) and variable ( $j = 1, \dots, p$ ) let  $R_{i,j} = 1$  if  $Y_{i,j}$  is observed and  $R_{i,j} = 0$  if  $Y_{i,j}$  is missing. Also, let  $R_i = (R_{i,1}, R_{i,2}, \dots, R_{i,p})^T$  a  $p$  dimension vector. Then, the missing value mechanism can be formulated as:

$$\Pr(R_i/Y_i) \quad (2.3)$$

We can separate two different cases of non-response:

1. When missing occurs at random and does not depend on the population unit (or Ignorable non-response). This category can be further separated in:
  - MCAR, when data are missing completely at random, and
  - MAR, when data are missing at random given an auxiliary variable.
2. When missing depends on the population unit, symbolized as MNAR and data missing not at random (or Non-Ignorable non-response).

In MCAR the probability of a value being missing, is unrelated to the observed and unobserved data on that unit. Therefore the observed data are representative of the population. Mathematically we can express this as:

$$\Pr(R_i/Y_i) = \Pr(R_i) \quad (2.4)$$

In MAR the probability distribution of  $R_i$ , given the observed data  $Y_{i,O}$  is independent of the unobserved data  $Y_{i,M}$ . This means that the chance of observing a variable under MAR will depend on its value. Mathematically we can express this as:

$$\Pr(R_i/Y_i) = \Pr(R_i/Y_{i,O}) \quad (2.5)$$

For the second category now, if the probability of an observation being missing depends on the underlying value and this dependence remains even when the observed data is given, we have a MNAR mechanism. Although, MNAR may be more plausible than MAR, inference and analysis is more difficult and hard under MNAR. We can write:

$$\Pr(R_i/Y_i) \neq \Pr(R_i/Y_{i,O}) \quad (2.6)$$



With the following example as set on the book of James R. Carpenter and Michael G. Kenward, "Multiple Imputation and its Application", it becomes more easily perceptible when we have a MCAR, MAR or a MNAR mechanism.

Example 2.4: Mandarin tableau.

At the frontage of senior Mandarin's house in the New Territories, Hong Kong there exist some figurines. We are interest to focus on the characteristics of those figurines, for example their number, height, facial characteristics and dress. So, Unit non-response corresponds to missing figurines and Item non-response to damaged or partially observed figurines. Here  $Y_i$  take the form of observations on the  $n = 4$  figurines, describing for example their size and dress.  $R_{i,j}$  indicates those observations that are missing on figurine because their heads are missing. Suppose that we want to summarize facial characteristics of the figurines, for example the average head circumference. If we have a MCAR mechanism and the missing heads missing completely at random then a valid estimate is obtained from the observed heads. With the assumption of MAR, the distribution of head characteristics given body characteristics, does not depend on whether the head is present. Thus, under the assumption of MAR we can estimate the distribution of characteristics of figurines with missing head from figurines with similar body characteristics. It should be mentioned that MAR is an assumption we make for the analysis and not a characteristic of the dataset. Maybe, MAR is plausible for headdress given necktie, but not for skin color given necktie. Lastly, the figurines with missing heads maybe were wearing a head dress that identified them as a member of a class and that was the reason that makes the heads to be damaged. This situation, it shows that we have a MNAR mechanism and we cannot say anything about the typical characteristics of the heads without making assumptions about the missing head dresses. Also, with MNAR arising that the distribution of head given body dress is different for figurines with missing and observed heads. Furthermore, analysis under MNAR is more complicated than the other cases, as it needs more assumptions about the form of the distribution of the missing data given the observed. But, as MAR , MNAR is an assumption for the analysis and not a characteristic of the data.



## 2.4 Dealing with Non-Response

One basic question is how to deal with non-response. We can work in two directions and try not having it at all, or at least to minimize it. Initially, we prevent non-response when we designed the survey, by doing the following actions:

- Well-designed questionnaires.
- Correct wording of the investigated subject of the survey.
- Correct times and conditions of conducting research.
- Choice, training and supervision of interviewers.

Then we proceed to the reduction of non-response, with:

- Follow-up calls.
- Different methods and different days and hour data collection.

In the most cases, despite the above actions, we will end up with a proportion of non-response and we must handle it. The next step is to ascertain if data are missing completely at random or not. If we believe that we have a MCAR mechanism, we can do nothing and cause no harm. Another alternative, assuming that we want to intervene in the proceeding, is to replace the unresponsive elements with other, or to take a smaller sample. We must say that rarely this case will appear. On the other hand, if we find out that missing data are not observed because of non-ignorable proceed, we do all the following actions which are summarized in subsections below.

### 2.4.1 Weighting Class Adjustments

Basic sampling weights  $w_i$ , reflect the probabilities of selection of the sampling units. They have been interpreted, as the number of the units in the population that represented by the unit  $i$  of the sample. It is a very important step the adjustment of the sampling weights for responders to account for the non-responding sample units. The more information we know about non-respondents, the more effectively we can adjust the weights of respondents in order to have similar characteristics to non-respondents. As non-response cannot be avoided, this adjustment of the sampling weights is necessary, as reduce resulting bias and minimizing the variance of the estimates. Specifically, one method for adjusting weights for non-response is to create homogeneous weighting classes of sample members that contain both respondents and non-respondents. Within each class, the responder weights are increased to take on the weights of the non-



respondents. We construct these weighting classes, so that in each weighting class the response variable is constant in class  $c$ , the probability of responding is the same for every unit in class  $c$  and the response variable is uncorrelated with the probability of responding. Unfortunately, there are some limitations to be able to implement this method. Firstly, the information that is used to create these classes must be available for both respondents and non-respondents. Secondly, in order to ensure relatively stable adjustments, there are some rules that are usually used when creating the class. Such as ensuring a certain number of respondents per class and a certain ratio of respondents to no respondents. In the simplest case, when we have Simple Random Sampling, the response probability for each class can be estimated by:

$$\hat{\Phi}_c = \frac{\text{number of respondents in class } c}{\text{number of units in class } c} \quad (2.7)$$

For example, using this probability and assuming that the sampling weight for each respondent in class  $c$  is  $1/(\hat{\Phi}_c)$ , it results with the assumption of SRS, that the estimator of the population total is given by:

$$\hat{t}_w = \frac{N}{n} \sum_i \frac{y_i}{\hat{\Phi}_c} \quad (2.8)$$

Example 2.5: Calculation of the response probability in Weighting Class Adjustments.

Suppose that we have 1000 individuals and we ask them about their monthly income. Also, suppose that we have three classes, the low incomes, the middle incomes and the high incomes. From the 1000 individuals, we take a sample of 50. From the chosen sample 20 persons belongs to the first class, 25 persons to the second class and 5 persons to the third class. Now, assume that from the 20 individuals that belong in the first class, only the 15 responded. From the 25 that belong to the second class only the 18 responded and from the 5 that belong to the third class only the 3 responded. So, we can easily calculate the response probabilities for each class as:

$$\hat{\Phi}_1 = 15/20$$

$$\hat{\Phi}_2 = 18/25$$

$$\hat{\Phi}_3 = 3/5$$

Then with the response probabilities and the values of the response variable, we can go to the equation (2.8) and find the estimator of the population total.



## 2.4.2 Post Stratification

It is called post-stratification because we can only compute weights after the collection of all the available data. The stratification part comes from the fact that we use various  $H$  known strata of the population to adjust our sample data, to conform more to the population parameters. It is similar approach with weighting class adjustment, but in this case population counts are used to adjust the weights. So, if we have information on population counts for post-strata and we have found that data are missing completely at random, within post-strata we use the following approach which is similar as for weighting classes. For example, under the assumption of SRS the post-stratified estimator for the population mean is given by:

$$\hat{t}_{post} = \sum_{h=1}^H \frac{N_h}{N} \bar{y}_h \quad (2.9)$$

where  $H$  the number of the strata,  $N_h$  the size of the  $h^{th}$  stratum and  $\bar{y}_h$  the sample estimate of the population mean from the  $h^{th}$  stratum.

Example 2.6: Calculation of the weights in Post Stratification.

Suppose that we have again the same sample as in the Example 2.5. Now in Post Stratification, the adjustment of the weights will be done with the use of the entire population. From the 1000 individuals, we suppose that 350 belong to the first class, 550 belong to the second class and 100 belong to the third class. So, the weights will be calculated as the number of the individuals that belong in each class divided with the total number of the individuals.

## 2.4.3 Propensity Modeling

Propensity modeling is an alternative method for adjusting for non-response and has become very popular, as the amount of information about non-respondents increases and the response percentage decreases. These models use logistic regression and then predict the likelihood of response versus non-response. This prediction is based to all available data and auxiliary information. Then the models are applied to the respondents and a log probability of responding is generated for each case. Using these probabilities the weights are calculated as the inverse of the associated probabilities. This method can be viewed as an extension of the weighting class adjustment from categorical to a response surface.



## 2.4.4 Imputation

It is the procedure whereby the missing values on one or more study variables are ‘filled in’ with substitutes. The main idea is to fill in a missing item with a substitute, expecting that this value is close enough in this which is missing. These substitutes can be constructed in a variety of methods and we will deal in detail with those in Chapter 3. Because imputed values are artificial, contain error. This error arises from the fact that the imputed values are not the “true values” but as we say before substitutes. The new values that constructed with the technique of Imputation can be classified in two categories:

1. Imputed values constructed by a statistical rule or a technique.
2. Imputed values constructed by expert skill and knowledge.

Even though imputation usually used for Item non-response, there is exist and the approach of Full Imputation, which used to treat both Item and Unit non-response. Another type of imputation is Mass Imputation. In Mass Imputation the artificial values are constructed not only for the sampled elements, but also for all the unobserved elements in the population. Imputation can be further separated in two major categories: i) The Single Imputation and ii) The Multiple Imputation. This distinction and the advantages and disadvantages of each category will be further analyzed in Chapter 3.

We must say that imputation may be preferable to weighting or other methods of treating non response in many cases. For example, in business surveys where the population is highly skewed. But as in the other methods which used to handle non-response, we hope that imputation will lead to estimates with small bias and small variances. Here’s an example regarding to the estimate of the population mean and which is made understandable the technique and the principle of imputation.

A very simple case is the estimating of the population mean that is given by:

$$\bar{Y} = \frac{1}{N} \sum_{i=1}^N y_i \quad (2.10)$$

An unbiased estimator of the population mean  $\bar{Y}$ , using the  $n$  observations of the sample is given by:

$$\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i \quad (2.11)$$

Assume now that non-response is present. So, we would not have some of the  $n$  observations from the sample. Let  $n_R$  denote the number of respondents and  $n_N$  the number of nonrespondents, while  $I_R$  the indices of the units that belong to respondents and  $I_N$  the indices of the units that belongs to non-respondents. Then, the sample mean can be written as:



$$\bar{y} = \frac{1}{n} (\sum_{i \in I_R} y_i + \sum_{i \in I_N} y_i) \quad (2.12)$$

In the above sum, the values of  $y_i$  for  $i \in I_N$  are missing. The method of imputation fill in these missing values with substitutes. So, if  $y_i^*$  for  $i \in I_N$ , is the imputed value for missing  $y_i$ , the sample mean can be written as:

$$\bar{y} = \frac{1}{n} (\sum_{i \in I_R} y_i + \sum_{i \in I_N} y_i^*) \quad (2.13)$$

As better are the values  $y_i^*$ , i.e. as closest to the missing values  $y_i$ , so much better for the calculation of estimates and variances because the bias is reducing, the estimates become more efficient and the results are more reliable.



# Chapter 3

## 3.1 Single Imputation and Methods

Often in surveys the desirable information is collected partially and as a result items of the dataset are missing. From the statistical inference point of view, it is necessary the replenishment of this missing information, not only to reduce bias and have efficient estimates, but also to create a clear rectangular dataset, as many of the statistical methodologies cannot handle non-response. The method which deal with this and help us with missing items is Single Imputation.

Single Imputation is the procedure whereby each missing value filling in with only one imputed value and it is the most commonly used method for handling item non-response in surveys. The basic advantages of this method is that standard complete methods of the analysis can be used and the final filled in dataset incorporate the data collector's knowledge. In addition to this positive features of Single Imputation, the basic disadvantage is that due to the fact the missing value is replaced only once, reflect neither sampling variability about the true value when one model for nonresponse is considered nor additional uncertainty when more than one model is being considered.

The imputed values can be created in a variety of methods according to the requirements of the research and the conditions that happens. Depending the method we are going to use, repeating the same approach to impute values we will be arrive to the same results or different. So, the methods that used to impute values are either deterministic, either random. Some deterministic methods of imputation are the Regression Imputation and the Nearest Neighbor Imputation, while a random method of imputation is the Random Hot Deck Imputation. Also, some of the methods make efficient use of auxiliary information and some others no. Because in many times different relationships are exist in the different subgroups of the sample, the same



method can not be implemented in every group. So, imputation can be carrying out according to a hierarchy of methods. That means first a method which is more likely to produce close enough values to the true is applied in one group, then if that method cannot be used to the second group, the second strongest method is applied and so on. The most famous and often used methods of imputation are listed in the next subparagraphs.

### 3.1.1 Mean Imputation

Mean Imputation is one of the most commonly used method of imputation and according to this the missing values of the data item are filled in with the mean value from the respondents sample units. Let  $r$  denote the response set for the sample,  $r_i$  the response set for the study variable and  $s$  the sample size. Also, suppose that  $y_k$  is the response value of the  $k^{th}$  respondent for the study variable and  $m$  is the number of respondents for the specific variable. It is easily understood that the number of the respondents for the study variable is less or equal with the number of the respondents for the sample which in turn is less or equal with the total size of the sample, i.e.  $r_i \subseteq r \subseteq s$ . So, if we want to fill in the missing values for our target variable with this method, we will compute the mean for the respondents from the sample by the following equation:

$$\bar{y}_{r_i} = \sum_{k \in r_i} \frac{y_k}{m} \quad (3.1)$$

Then we will replace each missing value, with the same value the sample mean value  $\bar{y}_{r_i}$ . This method maintains total and averages because the missing value is replaced with a value that has a relatively high degree of stability, but alters distributions and correlations among variables. Although, it maintains total and averages, causes artificial constrain on mean value and leading to artificial reduction of variance.

A refinement of the above method involves the separation of the sample into classes, as homogeneous as possible and apply the Mean Imputation in each class. In every subgroup we impute the missing values by using the mean value of the subgroup. Using classes allows to use information on differences among subdomains, something that leads to accurate results. The separation into classes is conducted by using an auxilliary variable highly correlated with the variable under study, so that homogeneous groups for the auxilliary variable to result in homogeneous group to the principal variable.



### 3.1.2 Hot Deck Imputation

The main principle of this method is to use a randomly selected donor to provide imputed values. The donor is selected by using different techniques and procedures depending the way we want to follow and fill in the missing items. However, it should be mentioned that the distribution of the completed data set after using Hot Deck Imputation may differ significantly from that which ideally would have been observed initially, because the imputed values come from respondents and usually respondents differ systematically from nonrespondents with regard to mean, variance and other characteristics. The most common techniques of Hot Deck Imputation listed below:

#### Random Hot Deck

Using this approach, a donor is randomly chosen from the sample respondents to give its value to the missing item and it is the simplest case of Hot Deck Imputation. So, the imputed values are  $y_j^* = y_m$  with  $1 \leq m \leq s$ . If  $r$  is the size of the group of the respondents in the sample, then the probability of choosing a specific donor among the sample and use its value for the imputation is given by the following equation:

$$P(y_j^* = y_m) = 1/r \quad (3.2)$$

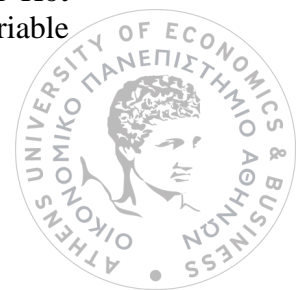
As in Mean Imputation the same idea can be implemented here that is the sample units can be divided into classes, using auxiliary variables in order to have less heterogeneity within a class and the classes are as much as possible different. Then, the missing values for each class are substituted from randomly selected donors who belong to the same class.

#### Sequential Hot Deck

In Sequential Hot Deck procedure the missing value is replaced by the corresponding value of the last observed responder unit. To be able to apply this method, we must separate the un-imputed data before the process starts and using the information from the auxiliary variables, a new array with the potential donors is created. When a missing value is found with a specific score on the auxiliary variables, we are looking for the corresponding score from the help array and the value of the donor with that score is returned. The advantage of this method is the ease of data processing, but it can be lead to multiple use of the same donor.

#### Nearest Neighbor Hot Deck

The main idea of this approach is to find a donor by minimizing the distance on an auxiliary variable for example the Manhattan or the Euclidean. Then the donor is used to substitute the missing value. The statistical idea behind the Nearest Neighbor Hot Deck Imputation is that two elements which have the x-values of the auxiliary variable



close, should also have y-values that are close. For example, suppose that we want to impute the missing value  $y_j$  by minimizing the Manhattan distance. The value that we are going to impute is  $y_j^* = y_m, m \in S$  when we have:

$$|x_j - x_m| = \min_{i \in S} |x_j - x_i| \quad (3.3)$$

Sometimes more than one response variable can be used. In this case, the Mahalanobis distance across a vector of several response variables is calculated between a specific non-respondent and each candidate donor. So, finding the smallest Mahalanobis distance is the way to determine the closest neighbor and the non-respondent missing values are imputed with the specific neighbor value.

### 3.1.3 Ratio Imputation

Ratio Imputation is a method of imputation that need auxiliary variables to be implemented. Let symbolize as  $y$  the values that takes the target variable with the missing values and  $x$  the values of the auxiliary variable. Then, we use only the responders group for the variables  $x$  and  $y$  to calculate the mean of each variable. We impute each missing value of the variable  $y$  with the ratio of the  $y$ -variable mean with the  $x$ -variable mean multiplied by the corresponding value of  $x$ . Mathematically, we can write:

$$y^* = x \bar{x}_R / \bar{y}_R \quad (3.4)$$

In the most cases, the auxiliary variable  $x$  is the same with the variable  $y$  measured in a previous survey. In that way, the two variables are correlated and the ratio estimator is more efficient.

### 3.1.4 Regression Imputation

In Regression Imputation more than one auxiliary variable can be used to create a regression equation that the right part will have the auxiliary variables and the left part the target variable. Using the information from the respondents, proceed to estimate parameters and then the regression equation is used to predict the missing values. Let



$x_1, \dots, x_q$  the auxiliary variables and  $\hat{b}_i, i = 1, \dots, q$  the regression estimates. The value of  $y_j$  is filled in with:

$$y_j^* = \hat{b}_0 + \hat{b}_1 x_{1j} + \dots + \hat{b}_q x_{qj} \quad (3.5)$$

A variation of this method is the stochastic regression imputation, in which the missing value is substituted by the predicted value of the above regression equation plus a randomly selected residual, i.e. the value of  $y_j$  is filled in with:

$$y_j^* = \hat{b}_0 + \hat{b}_1 x_{1j} + \dots + \hat{b}_q x_{qj} + e \quad (3.6)$$

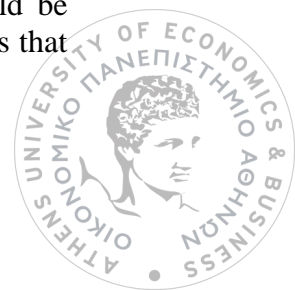
In this method, the hope for accurate imputed values becomes bigger with the assumption of the strong relationship between the target variable  $y$  and the auxiliary variables.

### 3.1.5 Cold Deck Imputation

According to Cold Deck Imputation, the imputed values come from previous similar surveys or other historical elements. The statistical idea of this method is that these available values will be closer than any other which can be calculated with one of the above methods. This method can be easily implemented, but in many cases the Cold Deck Imputation gives bias and inaccuracy in the research. This happens because the values of the variable that we want to impute maybe are not the same from survey to survey.

### 3.1.6 Principal Component Analysis

Principal Component Analysis is the most famous technique of Multivariate Analysis. It was first introduced by Karl Pearson in 1901 and used widely as a tool in exploratory data analysis. Essentially is a dimensionality reduction technique, where the original variables are transformed to a new set with fewer variables. The new variables are uncorrelated, retain the variability of the original variables and are ordered in ascending order with respect to their variances. The new transformed variables are usually named in the bibliography as scores, while the weights that each initial variable should be multiplied to get the scores are called loadings. Depending on the type of variables that



we have in our analysis and the general conditions, many existing approaches belong in Principal Component Analysis framework and could be used to fill in missing values. A more analytical presentation of Principal Component Analysis and the techniques that belongs to this method of analysis and help us with missingness, are available in Chapter 4.

### 3.1.7 Multiple Imputation

All the methods of imputation mentioned above replace each missing value with one only new imputed value without taking into consideration uncertainty about this missing value since any imputed value is treated as truly measured. A technique called Multiple Imputation has been developed by Rubin and other researchers to deal with this problem. The main idea here is to impute several  $m$  values for each missing datum and analyze each resulting data set by complete data methods with the analysis not take into consideration uncertainty about the missing values. Considering that these  $m$  values are ordered in a way that the first set of imputed values are used to create the first complete data set, the second set the second dataset and so on, we will have for our analysis  $m$  complete data sets with  $m$  in most cases be between 2 and 10.

As in Single Imputation, same here Multiple Imputation has the ability to use complete data methods of analysis and the ability to incorporate the data collector's knowledge. Furthermore it has and three other basic advantages compared with Single Imputation. First, Multiple Imputation increases the efficiency of estimation when many imputed values are randomly drawn to represent the distribution of the data. Second, when the imputed values are repeated draws under a model for non-response, we take the reflection of the additional variability due to the missing values under that model. Third, randomly repeated draws under more than one model, it helps us to understand and study the sensitivity of inferences to various models simply repeating complete data methods. The disadvantages of Multiple Imputation relative to Single Imputation are three. First of all more effort and work is needed to produce multiple imputations. Secondly, more storage space is needed to store multiple imputed dataset. A data analysis system using Multiple Imputation must be able to handle in addition to the full data matrix an auxiliary matrix with row size the number of missing values and column size the number of multiple imputations per missing value. Lastly, extra work and effort is needed since standard complete methods of analysis must be used on each completed dataset and the result under each model must be combined. Combining the results, give us the opportunity to take in account the uncertainty about the missing values and this combination usually requires the calculation of means and variances of the repeated complete dataset. However, the extra time and effort is not great and is measured in computer time and not on researcher time.



### 3.2 Applying Methods of Imputation

In this subsection we will use an example, as formulated in the book of Sharon L. Lohr to illustrate some of the above methods of Single Imputation. In the next table is observed an artificial small dataset which will be used to applicate and understand how a researcher deal with the missing values, by using some of the approaches of Single Imputation. The table consists of 6 columns, which have the variables and 20 rows, which have the individuals. The first column it is an indicator that identifies each person, the second column has the age of each person, the third column the sex, the fourth column the years of education and the fifth and sixth shows us if this person had been crime victim or violent crime victim, by using the number 1 if had been and the number 0 if not. The missing values that created completely at random, appear for the variables years of education, crime victim and violent crime victim and are 9 in total.

Table 3.1: Table with the measurements of 20 different individuals in 5 variables.

Person	Age	Sex	Years of Education	Crime Victim	Violent Crime Victim
1	47	M	16	0	0
2	45	F	?	1	1
3	19	M	11	0	0
4	21	F	?	1	1
5	24	M	12	1	1
6	41	F	?	0	0
7	36	M	20	1	?
8	50	M	12	0	0
9	53	F	13	0	?
10	17	M	10	?	?
11	53	F	12	0	0
12	21	F	12	0	0
13	18	F	11	1	?
14	34	M	16	1	0
15	44	M	14	0	0
16	45	M	11	0	0
17	54	F	14	0	0
18	55	F	10	0	0
19	29	F	12	?	0
20	32	F	10	0	0

### 3.2.1 Applying Mean Imputation

First we are going to construct four classes using the variables age and sex and each of the person of the above table it will be put in one of this classes. So, we make the classification and the results are presented in Table 3.2:

Table 3.2: Table with the classification of the individuals.

Sex / Age	$\leq 34$	$\geq 35$
Male	Persons 3, 5, 10, 14	Persons 1, 7, 8, 15, 16
Female	Persons 4, 12, 13, 19, 20	Persons 2, 6, 9, 11, 17, 18

We can observe that for the persons 2 and 6 the value for years of education is missing and these persons belong to the fourth class. These missing values will be imputed with the mean value of the women aged 35 or older, who responded to the question. So, the value that we will fill in is 12.25. It becomes easily understood that using the method of Mean Imputation the value which is used to impute, i.e. the mean for each class for the target variable, is the same as the mean of the respondents and is not one of the answers to the question. Due to the fact that, respondents may differ systematically from nonrespondents this method could be lead to wrong results and the estimated variance will be small because of the distortion and accumulation of the distribution to the sample mean.

This accumulation could be avoided simply by using stochastic Mean Imputation. Then the response variable will approximately normally distributed and the missing values will be imputed with a randomly generated value from a normal distribution with mean the average of the values for the responding units in class  $c$ , i.e. with mean the value  $\bar{y}_{cR}$  and standard deviation  $s_{cR}$ .

### 3.2.2 Applying Hot Deck Imputation

As in Mean Imputation within classes, same here the sample units are divided into classes. For each missing value, we substitute a value from a randomly chosen respondent unit of the same class. So, using Random Hot Deck Imputation in our example, person 10 was not responded for both variables of crime victimization. We can observe that in his class, the persons 3, 5, 14 have responses for both questions. Randomly chosen, the person 14 give his value to the missing items of person 10.

Also, we could apply the technique of Nearest Neighbor Hot Deck or Sequential Hot Deck Imputation. In the first case, we could use age and sex for the distance function, so the person with the closest age and the same sex select to be the donor, i.e. the person 3 will be used to impute the missing values of person 10. In the second case, person 5 has the last response recorded in the class that person 10 belongs. So, his values will be used and the value 1 is imputed in both questions for crime victimization.

### 3.2.3 Applying Regression Imputation

Suppose that in the same example, the aim is to predict the probability of victimization  $\hat{p}$ , using a logistic regression equation with explanatory variable the age. First, we estimate the parameters for our model, by using the 18 complete observations for the response of crime victimization and subsequently we take the following model:

$$\log \frac{\hat{p}}{1 - \hat{p}} = 2.56 - 0.08 * age$$

So, the predicted probability of being a crime victim for a 17 year old person is 0.74 and because that value is greater than 0.5, the value 1 is imputed.

### 3.2.4 Applying Multiple Imputation

Multiple Imputation is a method of imputation that differs from the other methods we analyzed above. For this reason, we will quote a second example to understand how this process operates with missing data. In the Table 3.3, we have a sample of 20 women, 75 years old at last birthday, selected from persons residing in three retirement communities for elderly persons. Missing values are observed for variables education (1, 2, 3, 4 denoting respectively elementary school, high school, some college, college graduate) and Mini Mental State Examination (MMSE) which is a screening test for cognitive impairment.



**Table 3.3:** Table with the characteristics collected on a sample of 20 women.

Retir. Community	Building	Education	MMSE
1	1	2	17
1	1	?	18
1	1	2	20
1	2	4	?
1	2	3	27
1	3	3	20
1	3	2	18
2	1	1	11
2	1	1	?
2	1	2	13
2	1	2	15
2	2	?	?
2	2	2	16
3	1	3	24
3	1	3	26
3	2	?	15
3	2	2	17
3	3	4	26
3	3	3	?
3	3	3	21

In this example suppose that we want to impute the missing values of MMSE (4 in total) and we have 3 primary sampling units denoted by each building. The first and third missing values belong in PSU 2 the second in PSU 1 and the fourth in PSU 3. So, each missing value can be replaced by fully observed values from donors that belong to the same PSU (Building) and stratum (Retirement Community). Finally, the subjects and the different combinations that can be used to impute the missing values is 6 and are available in Table 3.4:

**Table 3.4:** Table with all the possible imputed values for the missing values of the variable MMSE.

Combination	A	B	C	D
1	27	11	16	26
2	27	11	16	21
3	27	13	16	26
4	27	13	16	21
5	27	15	16	26
6	27	15	16	21



We conclude that the possible complete data sets that we can have in total are 6, as many as the possible combinations are. Each of the datasets yields a different estimation for the mean of the MSSE  $\bar{x}$  and for the standard error of the mean  $s(\bar{x})$ .



# Chapter 4

In this Chapter we are going to analyze all the imputation approaches that belong in the category of Principal Component Analysis methods. More specifically, first a definition and the aim of the Principal Component Analysis are presented with a description and a derivation of the principal components. Subsequently, follows a brief presentation of the Non Linear Principal Component Analysis and a comparison of the two methods with a reference to the positive and negative features. Also, the two main models, i.e. the fixed effect model and the random effect model are discussed. Last, all the proposed algorithmic techniques of Principal Component Analysis helping us to create a full data set that will be used in our statistical analysis are presented.

## 4.1 Definition of Principal Components and Principal Component Analysis

Principal component analysis is the oldest and most famous technique of multivariate analysis and it was introduced by Pearson (1901) and developed by Hotelling (1933). For the term “Principal Component Analysis” other terminology may be encountered outside of the statistical literature, for example “empirical orthogonal functions” in Meteorology or “eigenvector analysis” and “latent vector analysis”. The central idea of this technique is to reduce the dimensionality of a data set, which has a large number of variables, while retaining as much as possible of the variation present in the dataset. This aim is achieved with a transform of the original variables to a new set of variables the principal components. The main characteristic here is that the new variables are uncorrelated and there are ordered in such a way that the first new retain most of the variation present in all of the original variables.



As we mentioned above the aim of Principal Component Analysis is to reduce the number of variables from  $p$  to  $m$  where  $m \ll p$ . Suppose that  $x$  is the vector of  $p$  available variables of the data set. The principal components are optimal linear functions of  $x$  with respect to several and various optimality criteria. Although this maybe the simplest and most common use of Principal Component Analysis it is very useful because as already mentioned the original variation is reproduced with a new propose of variables which replaces a large number of variables with a much smaller and retains some interpretation with respect to the initial variables.

Since we have describe the purpose and general use of Principal Component Analysis we are going to express mathematically the way that principal components are defined. Firstly, we look for a linear function of the elements of  $x$   $a'_1x$  having maximum variance, where  $a_1$  is a unit measure vector of  $p$  constants  $a_{11}, a_{12}, \dots, a_{1p}$  so that:

$$a'_1x = a_{11}x_1 + a_{12}x_2 + \dots + a_{1p}x_p = \sum_{j=1}^p a_{1j}x_j \quad (4.1)$$

The maximization problem at this stage is:

$$\max_{a_1} \text{Var}(a'_1x) = \max_{a_1} a'_1 \Sigma a_1 \quad (4.2)$$

where  $\Sigma$  is the covariance matrix. After solving this maximization problem the constants  $a_{1j}$ ,  $j = 1, \dots, p$  will be known. In the next step we look for a linear function  $a'_2x$ , with maximum variance and uncorrelated with  $a'_1x$ . We obtain in this way the constants  $a_{2j}$ ,  $j = 1, \dots, p$ . We continue this approach in the same manner so that at the  $k_{th}$  stage we found a linear function  $a'_kx$ , which is the  $k_{th}$  principal component and has maximum variance and it is uncorrelated with  $a'_1x, a'_2x, \dots, a'_{k-1}x$ . The number of principal components that could be found is up to  $p$  as many as the original variables. So, after the definition and the mathematic form of the principal components we are interest to know how to find them. As we note from the equation (4.2) necessary condition to derive principal components is to know the covariance matrix  $\Sigma$  of the vector  $x$  of the random variables. The elements of the covariance matrix  $\Sigma$  is the known covariance between the  $i_{th}$  and  $j_{th}$  variable of the vector  $x$  when  $i \neq j$  and the variance of the  $j_{th}$  variable when  $i = j$ . The actual solution of the maximization problem in (4.2) is that  $a_1$  is equal with the first eigenvector of matrix  $\Sigma$ . The solution of the remaining maximization problems leads to the ordered in magnitude eigenvectors. Overall, the  $k_{th}$  principal component is given by:

$$z_k = a'_kx \quad (4.3)$$

where  $a_k$  is the eigenvector of covariance matrix  $\Sigma$ , which is corresponding to the  $k_{th}$  largest eigenvalue  $\lambda_k$ . The vector  $a_k$  can be mentioned in the bibliography as the vector of the coefficients or the vector with the loadings for the  $k_{th}$  principal component.



## 4.2 Non Linear Principal Component Analysis

Although Principal Component Analysis is maybe the most widely used method of multivariate analysis, suffers from two important limitations and becomes in many cases a non-appropriate method to perform in our analysis. Firstly, it assumes that the relationships between variables are linear and secondly that all the variables are assumed to be scaled at the numeric level. It is obviously that if our variables are categorical and especially categorical with unordered categories it makes no sense to compute weighted sums of the original variables, as Principal Component Analysis does. So, we must apply a method that have the positive features of Principal Component Analysis, but also can deal with the two above limitations. This method is the Non Linear Principal Component Analysis or else known as Categorical Principal Component Analysis.

Non Linear Principal Component Analysis is an alternative method of multivariate analysis which is suitable for all types of variables with mixed measurement levels and it can be used even if the variables are not linearly related. It was first described by Guttman (1941) and extra description of this method can be found in the literature from Kruskal (1965), Shepard (1966), Kruskal and Shepard (1974), Young et al (1978) and Winsberg and Ramsay (1983). The objective of Non Linear Principal Component Analysis is the same with the aim of Principal Component Analysis, i.e. to reduce a number of  $m$  continuous variables to a smaller number of  $p$  variables which are uncorrelated and ordered in such a way that the first new retain the largest percentage of the variability. If all the variables are numeric then the Non Linear Principal Component Analysis gives the same solution with Principal Component Analysis. If the variables are not only numeric and exist also categorical variables in our analysis, Non Linear Principal Component Analysis converting category numbers into numeric values by a technique called optimal quantification.

Optimal quantification, or optimal scaling, or optimal scoring, is a process during the ordered or unordered categories of categorical variables are assigned as numeric values. This process is very important and useful since variance can be calculated only for continuous numeric variables. In case of categorical variables the optimal quantified new variables are constructed in a way that as much possible of the variance is accounted for. Another reason for using quantification, is because Pearson correlations are used in the Principal Component Analysis and even if Pearson correlations may be calculated for ordinal variables, don't make sense for nominal variables. So, in Non Linear Principal Component Analysis the correlations are not computed between the original variables, but between the new quantified variables. Exactly for that reason, the solution in Non Linear Principal Component Analysis is not derived from the correlation matrix as in Principal Components Analysis, but it depends from the type of quantification that will be used in the analysis. It concludes that the method of optimal quantification maximizes the first  $p$  eigenvalues of the correlation matrix which is equivalent to maximize the variance accounted in the quantified variables.



### 4.3 Comparison of Principal Component Analysis and Non Linear Principal Component Analysis

After analyzing the methods of Principal Component Analysis and Non Linear Principal Component Analysis we are going to compare these two methods and to present the basic similarities and differences. Non Linear Principal Component Analysis has been developed after Principal Component Analysis and since then it has been used in cases where the last could not cope, i.e. data sets with categorical variables and in the presence of nonlinear relationships. Firstly, we are going to give the similarities between these two methods and then the main differences.

As we seen both methods provides eigenvalues, loadings and scores. Also, in both methods the first principal component is associated with the largest eigenvalue and accounts for most of the variance, the second principal component is associated with the second largest eigenvalue and accounts for the remaining variance and so on. The loadings are obtained with the same way, i.e. from the Pearson correlation between the principal components and observed variables for the Principal Component Analysis either the Pearson correlations between the principal component and the quantified variables for the Non Linear Principal Component Analysis. Last, the sum of squares of the loadings of a principal component is equal with the eigenvalue associated with that component and gives the variance accounted for the corresponding observed or quantified variable.

Although these two methods have many characteristics in common, have also main differences due to the fact that they need different conditions to be implemented. Initially, if there exists a nonlinear relationship between the variables or categorical variables are present in the problem and not only numeric, the Non Linear Principal Component Analysis performs better than Principal Component Analysis and leads to better results of accounting variance because apply the approach of the optimal quantification. Furthermore, in Principal Component Analysis the observed variables are directly analyzed, while in Non Linear Principal Component Analysis the quantified variables are analyzed and for that reason in the first case principal components are weighted sums of the original variables but in the second case principal components are weighted sums of the quantified variables. A last basic difference between the two methods is that in the simple Linear Principal Component Analysis the solutions are nested for the different values of the principal components  $p$ , something that does not happen in Non Linear Principal Component Analysis. More analytically, that means that the principal components of the  $p$  dimension will be equal with the corresponding principal components of the  $p + 1$  dimension in the case of Linear Principal Component Analysis.



## 4.4 Singular Value Decomposition

Before proceeding to analyze and describe the different approaches of imputation that exist in the category of Principal Component Analysis, it is necessary to report an alternative method which is relevant to PCA and can be used to find the principal components and there will be several references to this in the subparagraphs below. Singular Value Decomposition transforms the original correlated variables into a new set of uncorrelated variables fewer than the first one and suitable to expose the variability and the relationships that exist in the dataset. So, Singular Value Decomposition can be viewed as a data reduction method, since it transforms the data matrix  $X$  in a new matrix with fewer dimensions and through then identifying the principal components. The implementation of this method can be done in R with the package `svd`. Below, follow a small description with the way that principal components and principal axes can be found with the use of Singular Value Decomposition.

Let  $X$  a matrix of dimension  $n \times p$ , where  $n$  the number of individuals and  $p$  the number of the variables. Then the matrix  $X$  can be written as:

$$X = ULA' \quad (4.4)$$

where  $U$  a matrix of dimension  $n \times r$ ,  $L$  a matrix of dimension  $p \times r$ , so that  $U'U = I_r$  and  $A'A = I_r$  and  $L$  a diagonal matrix of dimension  $r \times r$ . So, if we can find the matrices  $U, L, A$  that satisfying relation (4.4), then the matrix  $A$  will give us the eigenvectors and the matrix  $L$  will give us the square roots of the corresponding eigenvalues, i.e. we will have the coefficients and the standard deviations of the principal components for the sample covariance matrix  $\Sigma$  or correlation matrix  $R$ . Also, the matrix  $U$  give us the scores of the principal components but in a rescaled form.

## 4.5 Main Models in Principal Component Analysis

In this subsection, in a model point of view, we are going to analyze two main models that exist and both developed and used for handling missing values, the Fixed Effect model and the Random Effect model. The Random Effect model is essentially a Bayesian treatment of the Fixed Effect model. These two models and a Bayesian procedure can be found in the articles of Julie Josse - Francois Husson and Alexander Ilin - Tapani Raiko.



### 4.5.1 The Fixed Effect Model

This model is a bilinear model where the data are generated as a fixed structure corrupted by noise and it may be found in the statistical literature as the “classical Principal Component Analysis model” or the “Fixed Factor Score model”. Applying this model the individuals have different expectations which are explained only from the error term and the maximum likelihood estimates derived from solving the least squares equations. The fixed effect model is applicable on data where each individual is an object of interest, i.e. the individuals are not a sample drawn from a population. For example, in sensory analysis where individuals can be food products and the aim of the analysis is to describe these food products. The mathematic form of this model is the following:

$$X_{ik} = m_k + (FU')_{ik} + \varepsilon_{ik}, \quad \varepsilon_{ik} \sim N(0, \sigma^2) \quad (4.5)$$

Let  $X_{I \times K}$  be the initial matrix of the data set with  $I$  individuals and  $K$  variables and  $M$  is an  $I \times K$  dimension matrix with each row equal to  $m$ , where  $m = (m_1, m_2, \dots, m_K)$  the vector with the mean of each variable.

The main aim of PCA is to reduce the dimensionality of a data set, i.e. to provide a best low rank  $S$  with  $S < K$ . This aim is achieved with defining the matrices of low rank  $S$ ,  $F_{I \times S}$  and  $U_{K \times S}$  with corresponding dimensions  $I \times S$  and  $K \times S$ , matrices that could be found by minimizing the following quantity:

$$\beta = \|X - M - FU'\|^2 \Leftrightarrow \quad (4.6)$$

$$\beta = \sum_{i=1}^I \sum_{k=1}^K (X_{ik} - m_k - \sum_{s=1}^S F_{is} U_{ks})^2 \quad (4.7)$$

Estimations of this two matrices are easy to be found with the additional condition that the columns of matrix  $U$  are orthogonal and of unit norm. Then the solution for the scores matrix or principal components matrix  $\hat{F}$ , where the  $\hat{F}$  is defined in a way that the variance of each column of the initial matrix  $X$  is equal to the corresponding eigenvalue of the covariance matrix  $\Sigma$  and the principal axes matrix or loadings-coefficient matrix  $\hat{U}$  with elements the eigenvectors of the covariance matrix  $\Sigma$ , are determined by the two followings equations:

$$\hat{U} = (\hat{F}'\hat{F})^{-1} \hat{F}'(X - \hat{M}) \quad (4.8)$$

&

$$\hat{F} = (X - \hat{M})\hat{U}(\hat{U}'\hat{U})^{-1} \quad (4.9)$$



## 4.5.2 The Random Effect Model

In this model, the data are generated as a random structure corrupted by noise and is implemented in these cases where Principal Component Analysis is performed on sample data drawn from a population, for example survey data. Here, in contrast with the previous model the individuals are independent and identically distributed. The Random Effect model is also known as the “Probabilistic Principal Component Analysis Model” or PPCA model and is special case of a Factor Analysis model in which the variance of noise is not free of distribution. The mathematic form of this model is the following:

$$X_{ik} = m_k + (ZB')_{ik} + \varepsilon_{ik}, \quad \varepsilon_{ik} \sim N(0, \sigma^2) \quad (4.10)$$

where  $B_{K \times S}$  a matrix with dimensions  $K \times S$  and elements the unknown coefficients and  $Z_{I \times S}$  the matrix of latent variables with dimensions  $I \times S$ . Under the selection of this model for our analysis it results a Gaussian distribution for the individuals with a specific structure of the covariance matrix  $\Sigma$  which is given below:

$$\Sigma = BB' + \sigma^2 \mathbb{I}_K, \quad i = 1, \dots, I \quad (4.11)$$

with  $\mathbb{I}_K$  the identity matrix of size  $K$ . In the Fixed Effect Model we present estimation for the matrices  $U$  and  $F$ , same here if we suppose that the individuals are independent we will have that:

$$X_{i.}/Z_{i.} \sim N(BZ_{i.}, \sigma^2) \quad (4.12)$$

Now, if  $\hat{U}$  is the matrix with the first  $S$  eigenvectors of the covariance matrix  $\Sigma$  and we take as an estimation for the scores the expectation of the latent variables given the observed variables, i.e.  $E(Z_{i.}/X_{i.})$ , the maximum likelihood estimates of the matrix  $B$ , of the matrix  $Z$  and the variance of noise  $\sigma^2$  are given by the following equations:

$$\hat{B} = \hat{U}(\hat{\Lambda} - \hat{\sigma}^2 \mathbb{I}_S)^{1/2} \mathbb{I}_S \quad (4.13)$$

&

$$\hat{\sigma}^2 = \frac{1}{K - S} \sum_{s=S+1}^K \hat{\lambda}_s \quad (4.14)$$

&

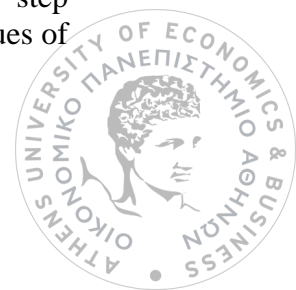
$$\hat{Z} = X B (B' B + \sigma^2 \mathbb{I}_S)^{-1} \quad (4.15)$$

## 4.6 Techniques of Imputation in Principal Component Analysis

As we mentioned in the previous chapters, in many surveys and especially in those with large data sets the desirable information is not completely collected and we are interested in applying a method to fill in the missing values of the variables. Assume that we are not approaching the problem by deleting any observation for which at least one of the variables has a missing value, something that will lead to waste of information rather than valid results if missing values are so many and respondents differ systematically from non-respondents. The very next alternative approach is to use the technique of imputation. In Chapter 3 we mentioned the most known and often used methods of imputation and we made a reference in the performance of Principal Component Analysis as a way to deal with missing values. Usually, the first step in this method is to compute the covariance or correlation matrix and the most effort is given on estimating the matrix  $\Sigma$  in the presence of missing data. Principal Component Analysis methods are dimensionality reduction techniques and often used to sum up data, study the similarities between individuals, the relationship between variables and to characterize the individuals by using the available variables. Their aim is the best estimation of the parameters and of their variance, the prediction of the missing values and the creation of a complete data set during the several algorithmic approaches that they have been developed in the category of Multivariate Exploratory Data Analysis methods. Methods of imputation that exist in the category of Principal Component Analysis and we are going to analyze in this chapter are: the Iterative PCA algorithm, the MI-PCA algorithm, the Forward Imputation, the Factorial Analysis for mixed datasets and the Nipals algorithm. The kind of method, as the way that will be used, depends from the type of variables in the problem, because different approaches are used for continuous variables and different approaches are used for categorical variables. However, there are methods that can deal with mixed type of data and can cope equally well with both types of variables.

### 4.6.1 The Iterative Principal Component Analysis Algorithm

The Iterative PCA is a procedure that corresponds to an expectation maximization algorithm which is associated to the Fixed Effect model that we have analyzed in subparagraph 4.5.1. During this algorithm and more specifically through the estimation process an imputation of the missing values is achieved. The expectation step corresponds to the imputation by the expectation of the missing values and the values of



the parameters given the observed values. The maximization step corresponds to the maximization of the likelihood. The Iterative PCA algorithm can be viewed in the bibliography as the EM-algorithm and because it is associated with the Fixed Effect model takes into account the similarities between individuals and the relationships between variables. So, this method it can be viewed as a single imputation method in the category of PCA, since a complete data set is creating during the implementation of the method. This algorithm is detailed analyzed in the articles of Julie Josse and Francois Husson and has first been proposed from Kiers (1997) who has shown that this algorithm minimizes the below criterion:

$$\beta = \|W * (X - M - FU')\|^2 \Leftrightarrow \quad (4.16)$$

$$\beta = \sum_{i=1}^I \sum_{k=1}^K w_{ik} \left( X_{ik} - m_k - \sum_{s=1}^S F_{is} U_{ks} \right)^2 \quad (4.17)$$

where  $X, F$  and  $U$  the matrices as defined in paragraph 4.3,  $W$  a weight matrix with elements  $w_{ik} = 0$  if the element  $ik$  of the initial matrix  $X$  is missing and  $w_{ik} = 1$  otherwise and  $*$  denote the Hadamard product. An interpretation of the Iterative PCA algorithm is that the weights of the variables are different from one individual to another and the weights of the individuals are different from one variable to another. The steps of the algorithm after minimizing the least squares criterion (4.17) using the non missing elements are presented in the next subparagraph.

First, in the beginning the missing elements of the data matrix  $X$  are replaced with initial values, for example the mean of each corresponding variable for the missing value. Then in the  $l$  step the Principal Component Analysis is implemented on the complete data matrix to estimate the parameters  $\hat{M}^l, \hat{F}^l, \hat{U}^l$  and after the estimation the missing values are imputed with the fitted values, i.e. we have:

$$\hat{X}^l = \hat{F}^l \hat{U}^{l'} + \hat{M}^{l-1} \quad (4.18)$$

and the new imputed dataset is given from:

$$X^l = W * X + (1 - W) * \hat{X}^l \quad (4.19)$$

according to which the imputed values for the non-missing entries are the same as the observed and the missing are completed with the fitted one. The approach is continued with the calculation of the  $\hat{M}^l$  on  $X^l$  and the estimation of parameters. The imputation is continued until we have convergence. So, the EM-algorithm improves the prediction of the missing values compared with the initial filled values using the mean imputation. The implementation of the Iterative PCA algorithm and generally the performing of Principal Component Analysis is achieved in the package of R “missMDA”.



## 4.6.2 The Multiple Imputation in Principal Component Analysis

As we have seen multiple imputation is a method that generates multiple imputed dataset in order to reflect the uncertainty of the prediction of the missing values, something that cannot be done with the method of single imputation. Multiple imputation which has proposed by Rubin (1987), consists of generating at a first stage  $D$  plausible values for each missing value and this leads to the creation of  $D$  imputed data sets. After the creation of the  $D$  imputed datasets the statistical analysis of each imputed dataset is performed in order to obtain the estimate of the quantity of interest  $\theta$ . Lastly, the results from each data set are combined to make a finally estimation for  $\theta$  and the variability of this estimation take into account the uncertainty due to the missing elements. The method that we are going to analyze here is a multiple imputation method which belongs into the category of Principal Component Analysis. This means that we focus on assessing the variability due to the missing values, i.e. we focus on assessing the influence of the different predicted values created from the approach of multiple imputation during the implementation of a Principal Component Analysis model.

The MI-PCA algorithm begins with the implementation of the EM-PCA algorithm on the matrix  $X$ . Estimation of the parameters  $\hat{M}$ ,  $\hat{F}$  and  $\hat{U}$  are therefore derived. The expectation-maximization algorithm is an iterative algorithm which gives a maximum likelihood estimate from an incomplete dataset when a solution is not directly available. The expectation step corresponds to the imputation of the missing values with the expectation of the missing values given the observed values and the values of the parameters at the  $l$  iteration. The maximization step corresponds to carry out Principal Component Analysis on the imputed data set. Finally, after the implementation of the EM-PCA algorithm and the estimation of the parameters we reconstruct the data as:

$$\hat{X} = \hat{M} + \hat{F}\hat{U}' \quad (4.20)$$

We next calculate the incomplete matrix of the residuals, since the matrix  $X$  is incomplete, from:

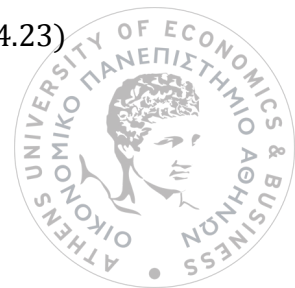
$$\hat{\varepsilon} = X - \hat{X} \quad (4.21)$$

For all the imputed datasets, the algorithm continues with a bootstrap approach, where the matrix of residuals  $\hat{\varepsilon}$  is replaced with a new matrix of residuals  $\varepsilon^*$ . A new matrix of data  $X^*$  that corresponds to  $\varepsilon^*$  is obtained and using this matrix the new estimated parameters  $\hat{M}^*$ ,  $\hat{F}^*$  and  $\hat{U}^*$  are obtained. The new data matrix is calculated as:

$$X^* = \hat{X} + \varepsilon^* \quad (4.22)$$

Then for  $d = 1, \dots, D$ , each missing value  $x_{ik}^d$  is imputed with the value of the conditional mean:

$$(\hat{M}^* + \hat{F}^*\hat{U}^{*'})_{ik}^d \quad (4.23)$$



Last, for all the imputed values in the datasets a residual  $\tilde{\varepsilon}$  drawn from the observed residuals  $\hat{\varepsilon}$  is added to take account the uncertainty for each missing value and the imputed matrix for the  $d$  dataset is:

$$X^d = W * X + (1 - W) * ((\hat{M}^* + \hat{F}^* \hat{U}^{*'}) + \tilde{\varepsilon}) \quad (4.24)$$

As we have seen there are different steps in the algorithm of multiple imputation and the main aim is to produce  $D$  different datasets that will reflect the uncertainty of the prediction due to the missing values. Also, the variance of the prediction it reflects the uncertainty in the estimation of the parameters, because  $D$  sets of parameters  $(\hat{M}, \hat{F}, \hat{U})$  are calculated and the variability of the noise since each missing value  $x_{ik}^d$  in the  $d$  dataset is imputed with:

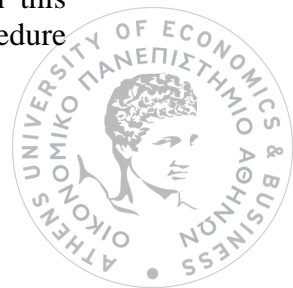
$$x_{ik}^d = \hat{m}_k^d + \sum_{s=1}^S \hat{f}_{is}^d \hat{u}_{ks}^d + \tilde{\varepsilon} \quad (4.25)$$

So, the proposed algorithm MI-PCA is a technique that belongs to the category of Principal Component Analysis methods and the reason that was developed is the generation of multiple imputed datasets. It must be said that this method is computationally efficient, fast and it can be easily implemented. A more analytical description of the method is available in the articles of Julie Josse and Francois Husson. The approach of MI-PCA is available in the package of R “missMDA”.

### 4.6.3 The Forward Imputation Applying Principal Component Analysis

Forward Imputation is based on an iterative algorithm, in which the method of Principal Component Analysis is performed in a complete matrix with elements the subset of data with no missing values and the imputation is achieved by the Nearest Neighbor approach, i.e. the missing data cells are filled in with the corresponding values of the nearest unit in the complete matrix measured by a distance. In this way, the entire process followed is a sequential procedure that imputes missing values “forward” by alternating the method of Linear Principal Component Analysis which is performed every time on the updated complete matrix and the Nearest Neighbor imputation process. In the particular case, by applying PCA, this technique can be only used when our dataset is consisting of continuous variables.

Suppose that  $X$  is the initial matrix of dimension  $n \times p$ , with  $n > p$ , where  $n$  denotes the individuals and  $p$  the quantitative measured variables. We consider that in this matrix there are existing missing values for the observed variables. Then the procedure



of Forward Imputation as described in the articles of Nadia Solaro, Alessandro Barbiero, Giancarlo Manzi and Pier Alda Ferrari begins with creating a matrix  $X_0^{(0)}$  of dimension  $n_0^{(0)} \times p$  from the initial matrix  $X$  with elements only the  $n_0$  completely responding units, with  $p \leq n_0^{(0)} < n$  and no missing values. Subsequently,  $K$  submatrices  $X_k$  of dimension  $n_k \times p$ , with  $k = 1, \dots, K < p$  expressing the number of missing values in each row created from the initial divided matrix  $X$ . In the next step for a fixed value of  $k$  and with the use of the Principal Component Analysis,  $p$  principal components are extract from the covariance matrix  $\Sigma_0^{(k-1)}$  or the correlation matrix  $R_0^{(k-1)}$  of the complete matrix  $X_0^{(k-1)}$  of dimension  $n_0^{(k-1)} \times p$ . We also use the covariance or corellation matrix to obtain the eigenvalues  $\lambda_s^{(k-1)}$  and the eigenvectors  $\omega_s^{(k-1)}$  with generic element  $\omega_{js}^{(k-1)}$ ,  $j, s = 1, \dots, p$ . The approach continues with the computation of the pseudo principal components for the submatrices  $X_k$  and  $X_0^{(k-1)}$ , using only the common variables with no missing values. If we denote as  $t$  the  $k$  combinations set of the  $p$  indices of variables which have missing values on the rows of  $X_k$ , the pseudo principal components  $\tilde{C}$  for the submatrix  $X_k$  symbolized with  $\tilde{C}^{(k)}$  and the pseudo principal components  $\tilde{C}$  for the submatrix  $X_0^{(k-1)}$  symbolized with  $\tilde{C}^{(k-1)}$  are given by:

$$\tilde{C}^{(k)} = \sum_{\substack{l=1 \\ l \notin t}}^p \omega_{ls}^{(k-1)} X_l^{(k)} \tag{4.26}$$

&

$$\tilde{C}^{(k-1)} = \sum_{\substack{l=1 \\ l \notin t}}^p \omega_{ls}^{(k-1)} X_l^{(k-1)} \tag{4.27}$$

After computing the pseudo principal components, the Minkowski distance of order  $r$ , with  $r \geq 1$ , is calculated between each incomplete unit  $u_i^{(k)}$  in matrix  $X_k$  and each complete unit  $u_c^{(k-1)}$  in matrix  $X_0^{(k-1)}$ . The Minkowski distance  $d_r$ , is given by:

$$d_r(u_i^{(k)}, u_c^{(k-1)}) = \left\{ \sum_{s=1}^p \left| \tilde{C}_i^{(k)} - \tilde{C}_c^{(k-1)} w_s^{(k-1)} \right|^r \right\}^{1/r}, c = 1, \dots, n_0^{(k-1)} \tag{4.28}$$

where  $w_s^{(k-1)}$  is the weight and calculated as the square root of the  $s$  eigenvalue of the covariance matrix  $\Sigma_0^{(k-1)}$  or the correlation matrix  $R_0^{(k-1)}$ , divided by the sum of all the eigenvalues. So, the weight  $w_s^{(k-1)}$  is given by the equation:



$$w_s^{(k-1)} = \sqrt{\lambda_s^{(k-1)} / \sum_{m=1}^p \lambda_m^{(k-1)}} \quad (4.29)$$

The Minkowski distance is used and more specifically the implementation of it helps to find the first corresponding complete units and select the donors  $u_{\delta,i}^{(k)}$  for the unit  $u_i^{(k)}$ . Then, if  $n_\delta$  is the total number of donors for the unit  $u_i^{(k)}$  and  $d_{\delta,i}$  is the distance between the  $\delta_{th}$  donor and the unit  $u_i^{(k)}$ , the missing values of the variable  $X_j, \forall j \in t$  are imputed with:

$$\tilde{x}_{ij}^{(k)} = \frac{\sum_{\delta=1}^{n_\delta} X_{\delta j}^{(k-1)} / d_{\delta i}}{\sum_{\delta=1}^{n_\delta} 1 / d_{\delta i}} \quad (4.30)$$

When the imputation is completed we take the imputed matrix  $\tilde{X}_k$  and this matrix is joined with the initial full observed matrix  $X_0^{(k-1)}$ . By merging these two matrices the new data matrix  $X_0^{(k)}$  is arising and then we set up  $k = k + 1$  and the approach is continued from the beginning until the initial matrix  $X$  is completely imputed. The method of Forward Imputation is available in the package of R “ForImp”.

#### 4.6.4 The Forward Imputation Applying Non Linear Principal Component Analysis

Non Linear Principal Component Analysis, as we have seen, is an alternative method of multivariate analysis which has the positive features of Principal Component Analysis but it is suitable for ordinal categorical variables and variables that are not linearly related. It can be used also to extract statistical indicators and measure a latent phenomenon which is lied in the categorical variables by the minimization of a loss function that consists from the scores, the loadings and the category quantifications for each observed categorical variable. For more details see Gifi (1990) and Michailidis-De Leeuw (1998). Now, if missing values are observed in the initial data matrix then the technique of Forward Imputation is implemented to create a new imputed dataset, but in a kind of a different way from the procedure which is followed in Principal Component Analysis. The method of Non Linear Principal Component Analysis is implemented on a subset of data with no missing values and the approach of optimal quantification is performing first and then the implementation of the Nearest Neighbor procedure follows for the imputation of missing cells with the corresponding nearest unit in the complete matrix. The proposal of the method that we are going to describe can someone find in the paper of Pier Alda Ferrari, Alessandro Barbiero and Giancarlo Manzi.



Let  $X$  be the data matrix affected by missing values, with  $n$  units and  $m$  ordinal variables with  $k_j$  categories for  $j = 1, \dots, m$  and  $G_j$  be the indicator matrix of dimension  $n \times k_j$  for the  $j$  variable. Then the procedure begins with the split of the matrix  $X$  into a matrix  $X_0^{(0)}$  of dimension  $n_0^{(0)} \times m$  with no missing values and  $K$  disjoint submatrices  $X_k$  of dimension  $n_k \times m$  for  $k = 1, \dots, K$  and  $K < m$ , where  $k$  denoting the number of missing elements for each row. Subsequently, for each matrix  $X_k$ , and for each row in matrix  $X_k$ , the approach of the Nearest Neighbor is implemented using the loadings and the quantifications which have observed from the performing of Non Linear Principal Component Analysis on the complete matrix  $X_0^{(k-1)}$ . So, using the Nearest Neighbor imputation the missing values are filled in with the values of the closest observation in the complete matrix  $X_0^{(k-1)}$ . To define the donors that are going to provide their values, the Euclidean distance as derive below in the equation (4.31) is used:

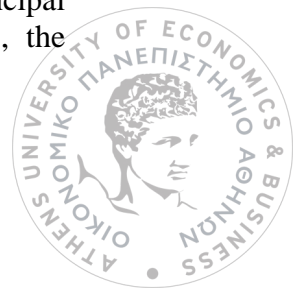
$$\min_z d(u_i^{(k)}, u_z^0) = \min_z \left( \sum_j \beta_j^{(k-1)} \left| G_j(i) q_j^{(k-1)} - G_j(z) q_j^{(k-1)} \right|^2 \right)^{\frac{1}{2}} \quad (4.31)$$

the indicator  $j$  is running only for the  $m - k$  variables that are observed on the both units  $u_i^{(k)}, u_z^0$  where  $u_z^0 \in X_0^{(k-1)}$  and  $q_j = (q_{j1}, \dots, q_{jk_j})'$  of dimension  $k_j \times 1$ , is a vector which contains the optimal category quantifications for variable  $j$ . After the above steps, the approach is continued by appending the new completed rows in the matrix  $X_0^{(k-1)}$  to create the new matrix  $X_0^{(k)}$ . The procedure is implemented for  $k = 1, \dots, K$  and finally the Non Linear Principal Component Analysis is performed on the complete matrix  $X_0^{(k)}$  to find the variable loadings  $\beta_j^{(k)}$  and the category quantifications  $q_j^{(k)}$  for  $j = 1, \dots, m$ .

The method that we have described due to the fact that it is implemented initial on the complete matrices it makes use of the whole information that is available and the variable loadings that gives can be interpreted as correlation coefficients and show the role of each variable in the process. Also, because the objects are included in the analysis in different steps and according to their number of observed variables, the role of each object in the interpretation of the results is connected with its number of observed values.

#### 4.6.5 Factorial Analysis for Mixed Data

In the majority of surveys and statistical researches the observed variables are not only continuous. Thus appeared the need to develop a method that could find the influence of all type of variables. The Factorial Analysis for mixed data responds well in all types of variables. As all the methods for imputation that belong in the category of Principal Component Analysis is used to study the relationships between the variables, the



similarities between the individuals and the connection between the existing measured variables and the individuals. This method, also named as PCAMIX, it operates like Principal Component Analysis in the case we have only quantitative variables and like Multiple Correspondence Analysis in the case we have only qualitative variables and it was first presented from Brigitte Escofier (1979) and then by Gilbert Saporta (1990) and Jerome Pages (2002). The article of Vincent Audigier, Francois Husson and Julie Rosse has an analytically description of the algorithm that we are present below.

Suppose now that  $I$  denote the individuals,  $K_1$  the number of the continuous variables and  $K_2$  the number of the categorical variables. It concludes that the total number of variables is  $K = K_1 + K_2$ . Then we note as  $X$  the matrix of dimension  $I \times J$  with  $(x_j)_{1 \leq j \leq K_1}$  continuous variables and  $(x_j)_{K_1+1 \leq j \leq J}$  the corresponding dummy variables of the categorical variables, where  $J$  is given by the equation:

$$J = K_1 + \sum_{k=1}^K q_k \quad (4.32)$$

and  $q_k$  is the number of categories of variable  $k$  for  $k > K_1 + 1$ . After the determination of the variables the next step is a weighting approach whereby each continuous variable  $x_j$  is divided with its standard deviation  $s_j$  and each dummy variable is divided with  $\sqrt{p_j}$  where  $p_j$  is the proportion of individuals that take the category  $j$ , with  $K_1 + 1 \leq j \leq J$ . Lastly, the implementation of the Principal Component Analysis on the weighted matrix  $XD_{\Sigma}^{-1/2}$ , where  $X$  the initial matrix and  $D_{\Sigma}$  the diagonal matrix, complete the process. The structure of the matrix  $D_{\Sigma}$  is available in the next equation:

$$D_{\Sigma} = \text{diag} \left( s_{x_1}^2, \dots, s_{x_{K_1}}^2, p_{K_1+1}, \dots, p_j, \dots, p_J \right) \quad (4.33)$$

We also define the matrix  $M$  of dimension  $I \times J$ . Each row of this matrix is equal with the vector of the means of each column of  $XD_{\Sigma}^{-1/2}$ .

After the general frame and the definitions of the variables we can move on and describe analytically the iterative algorithm of the Factorial Analysis for mixed data. First the missing values of the matrix  $X$  are substituted by initial values, for example the missing values of the continuous variables are filled in with the mean of the corresponding variable and the missing values of the dummy variables with the proportion of the corresponding category of each variable using non missing entry. It must be said that the sum of the entries of one individual and of one categorical variable is one and the initial values of the dummy variables can be no integer. Then we calculate the matrix  $D_{\Sigma}^0$  and the matrix  $M^0$  which is the matrix with the vector of the means of each column of  $X^0(D_{\Sigma}^0)^{-1/2}$ . In the  $l$  step of the algorithm is performing the Singular Value Decomposition of the matrix:

$$\left( X^{l-1}(D_{\Sigma}^{l-1})^{-1/2} - M^{l-1} \right) \quad (4.34)$$

The matrices  $\hat{V}^l, \hat{U}^l, (\hat{\Lambda}^l)^{1/2}$  with the corresponding eigenvectors, scores and eigenvalues are obtained from the above approach. Then by keeping the first  $S$

dimensions, we compute the fitted matrix  $\hat{X}_{I \times J}^l$  and we use it to find the new imputed data set  $X^l$ :

$$\hat{X}_{I \times J}^l = \left( \hat{U}_{I \times S}^l (\hat{\Lambda}_{S \times S}^l)^{1/2} (\hat{V}_{J \times S}^l)^T + M_{I \times J}^{l-1} \right) \left( (D_{\Sigma}^{l-1})_{I \times J} \right)^{1/2} \quad (4.35)$$

&

$$X^l = W * X + (1 - W) * \hat{X}^l \quad (4.36)$$

Where  $*$  it's a binary operation that takes two matrices of the same dimensions and produces another matrix where each element  $ij$  is the product of the elements  $ij$  of the original two matrices and  $W$  is a matrix with dimensions  $I \times J$  and elements equal to one if  $x_{ij}$  is observed and zero otherwise. The algorithm goes on until the change in the imputed matrix is smaller than a predefined threshold  $\varepsilon$ , for example when:

$$\sum_{ij} (\hat{x}_{ij}^{l-1} - \hat{x}_{ij}^l)^2 \leq 10^{-6} \quad (4.37)$$

The FAMD algorithm that has been described above and estimates the parameters via Singular Value Decomposition and goes on with the imputation of the missing values, suffer from overfitting problems. In order to overcome this problem and avoid instabilities an approach is proposed by Josse and Husson (2012) according to, is assumed that the first  $S$  dimensions are giving us information and noise and the last ones only noise, because the variance of the noise is estimated by the mean of the last eigenvalues. This is a regularized algorithm for which assumed that if the noise is small is similar with the initial algorithm and the eigenvalues  $\sqrt{\hat{\lambda}_s^l}, s = 1, \dots, S$  are replaced by the new regularized values:

$$\frac{\hat{\lambda}_s^l - \hat{\sigma}^2}{\sqrt{\hat{\lambda}_s^l}}, \quad s = 1, \dots, S \quad (4.38)$$

with

$$\hat{\sigma}^2 = \sum_{s=S+1}^{J-K_2} \frac{\lambda_s}{J - K_2 - S} \quad (4.39)$$

The method of Factorial Analysis for mixed datasets is a very reliable and computationally fast technique. It help us in many cases where the structure of data is really complicated. The approach of the Factorial Analysis is available in the package of R “missMDA”.



## 4.6.6 The Nipals Algorithm

The Nipals algorithm or Nonlinear Partial Least Squares is an iterative algorithm that performs Principal Component Analysis on datasets in the absence of missing values. It was first presented by Wold (1966) with the name NILES as an iterative procedure and it not well known because the method is not cooperate well in cases of data with very big percentage of missing elements. This algorithm can provide initially the desired principal components by applying an iterative procedure based on least squares regression and helps us with empty cells since moves on an imputation approach where all the empty cells are filled in.

Suppose that the data matrix  $X$  of dimension  $n \times p$  consists of  $n$  individuals and  $p$  quantitative variables. Also, if  $X = (X_1, \dots, X_p)'$  we suppose that each variable  $X_i$  is set in a way that  $E(X_i) = 0, \forall i = 1, \dots, p$ . Furthermore, it has been proved that in Principal Component Analysis the initial matrix  $X$  can be written as:

$$X = \sum_{h=1}^q t_h p_h' \quad (4.40)$$

where  $t_h$  denote the vector with the principal components for  $h = 1, \dots, q$  and  $p_h$  denote the vector with the principal axes for  $h = 1, \dots, q$ . The element  $p_h(i)$  of the vector  $p_h$  represents the slope coefficient in the linear regression of the variable  $X_i$  with the principal component  $t_h$ . The Nipals algorithm begins with an initialization step where the initial matrix  $X$  influenced by missing data is setted with a complete matrix  $X_0$  with no missing elements. The matrix  $X_0$  is a matrix that has been created with corresponding observed values of the matrix  $X$  and imputed values for the missing values of the matrix  $X$ . The approach of the imputation that can be used in the initialization step to create a complete matrix  $X_0$  is the Mean Imputation, i.e to fill in each missing value with the mean of the corresponding variable. After defining the matrix  $X_0$ , for  $h = 1, \dots, q$  we take as  $t_h$  the first column of the matrix  $X_{h-1}$  and until the convergence of the vector of principal axes  $p_h$ , we calculate the  $p_h$  as:

$$p_h = X_{h-1}' t_h / t_h' t_h \quad (4.41)$$

After the calculation of the vector  $p_h$ , we normalize this vector to a unit vector and then we calculate the vector of principal components  $t_h$  as:

$$t_h = X_{h-1} p_h / p_h' p_h \quad (4.42)$$

The algorithmic approach is completed with the creation of the new complete matrix  $X_h$ , which is given below:

$$X_h = X_{h-1} - t_h p_h' \quad (4.43)$$



Even if the Nipals algorithm is a not difficult technique, based on simple least squares regression and easy to implement in the most cases, it is not a widely used method. This is due to the fact that this algorithm requires strictly quantitative variables and the convergence of the algorithm depends on the percentage of missing data. The algorithm can be implemented in the R packages “plsdeplot” and “pcaMethods”.



## Chapter 5

In Chapter 5 we will deal with experiments in several datasets which will be analyzed by the statistical package R. The main aim is to apply the methods of imputation that we have reported in detail in Chapter 4 and to compare them, in order to see which method responds better depending on the case. The datasets that we will use vary, i.e. the variables can be all continuous and all the cells of the matrix are consist from numeric values, either the dataset may consist of continuous and categorical variables. Another sort for the data regard to whether is simulated and coming from a distribution, or there are truly collected and they are based on real situations. The data that we are going to use, in order to compare the methods of imputation that belongs in the category of Principal Component Analysis are listed below:

- Simulated data from the Multivariate Normal distribution where the variables are correlated.
- Simulated data from the Multivariate Normal distribution where the variables are not correlated.
- Simulated data from the Skew Normal distribution where the variables are correlated.
- A real dataset with only numeric variables that corresponds to 202 individuals and 11 continuous variables, which contains some measurements for athletes collected at the Australian Institute of Sport.
- A real mixed dataset, that consist of 1 categorical and 4 continuous variables and contains some measurements for 150 different plants.

In the above datasets, we create missing values with different percentage each time. In more detail, the rates of missingness will be 5%, 10% and 20%. Then for each case the application and the comparison of the methods, the packages, the corresponding commands, the advantages and disadvantages of each method and the conclusions from the analysis will be presented.



## 5.1 Data with continuous variables

In the case of continuous variables the available methods of imputation that belongs in Principal Component Analysis and were presented in the previous chapter are: a)The Iterative Algorithm, b)The Factorial Analysis for mixed data, c)The Forward Imputation and d)The Nipals Algorithm. All the above methods will apply in the datasets and will be compared. At this point, we must say that the Factorial Analysis is a method that it has been created for cases with data that have continuous and categorical variables, but it is plausible that we can use it if we have only continuous variables. Except for these four methods we may use in our analysis another general method of imputation, the method of the Mean Imputation. We will include this non-Principal Component imputation method to our comparison study as the standard technique usually used in practise.

Mean Imputation method can be implemented in the package of R called 'ForImp'. The implementation is very easy and it can be done with the following command:

```
>meanimp(mat)
```

where `mat` the matrix with the missing values. So, if the matrix with the missing values that we want to impute is named `mat`, we get the new imputed matrix using the above command. The next method that is belong to Principal Component Analysis is the Iterative Algorithm. This procedure is available in the package of R 'missMDA' and the corresponding command is:

```
>imputePCA(mat,ncp,threshold,maxiter)
```

The basic arguments of the command are a data frame with continuous variables containing missing values, an integer corresponding to the number of principal components that used to predict the missing values, the threshold for assessing convergence and an integer which denoting the maximum number of iteration for the algorithm. Next method in comparison is the Factorial Analysis for mixed data. This approach is also available in the package 'missMDA' and it can be implemented with the following command:

```
>imputeFAMD(mat,ncp)
```

The basic arguments here is as before a dataframe with continuous and categorical variables containing missing values and an integer corresponding to the number of principal components that used to predict the missing values. Subsequently, the next method is the Nipals Algorithm. The way that it can be used, is available in the package 'pcaMethods' and the main command is:

```
>completeObs(pca(mat,nPcs,method="nipals"))
```



The arguments are a matrix with continuous variables that has missing values and an integer denoting the number of principal components. The last method that we will use for the comparison is the Forward Imputation. It is available in the package ‘ForImp’ and the command is:

```
>ForImp(mat,p)
```

Same here, the arguments are a matrix with numeric values containing missing entries and an integer denoting the number of principal components. All the above methods are fast and can easily be implemented regardless to the total percentage of the missing values. In our analysis we will use and compare these five methods in datasets with percentage of missing values 5%, 10% and 20%.

We first create missing values in the existing dataset and then depending on the method that we are going to use we create the new imputed matrix. We repeat this procedure for each method and we obtain a complete dataset each time. The comparison among the imputation techniques is according to a measure of accuracy between the initial dataset and the imputed. The minimum value of this measure indicates the best imputation method. For continuous datasets the measure used for the comparison is the Root Mean Square Error. It is calculated as the square of the sum of the proposed value subtracting the true value that is missing for the purpose of the experiment and then we divide with the number of the individuals. The mathematic form of this measure is:

$$RMSE = \sqrt{\frac{\sum(x_i^t - x_i)^2}{n}} \quad (5.1)$$

where  $n$  is the number of the individuals,  $x_i^t$  the imputed value and  $x_i$  the initial true value. In the case that the variables are not all numeric, only the Root Mean Square Error is not enough and cannot be implemented for the categorical variables. So, if we have a dataset with continuous and categorical variables we use two measures. First we use the Root Mean Square Error for the continuous variables and the Percentage of Falsely Classified observations for the categorical variables. This is a measure that is calculated by dividing the number of the misclassified observations with the totally number of the missing for our categorical variables. Mathematically we can write:

$$PFC = \frac{n^m}{n^*} \quad (5.2)$$

where  $n^m$  the number of the misclassified observations and  $n^*$  the number of all the missing for the categorical variables.

As we will see in the next chapters, we will not have a single generation of datasets in the case of simulated data, but we will generate 200 datasets in each case. So, for each simulated dataset the corresponding measure is calculated and the comparison is based on the complete set of values.



### 5.1.1 Uncorrelated Data From The Multivariate Normal

In this paragraph we will use for the comparison of the methods simulated data from the Multivariate Normal. Let's assume a dataset of dimension  $150 \times 10$ , i.e we will have 150 individuals and 10 continuous variables generated from  $N(\mu, \Sigma)$ , where  $\mu=0$  and  $\Sigma = \text{diag}\{0.63, 6.65, 5.86, 7.71, 3.46, 7.84, 4.39, 9.87, 1.23, 1.58\}$ . Mathematically we can write about the structure of our data:

$$x \sim N(\mu, \Sigma) \quad (5.3)$$

$$\mu = [E(x_1), \dots, E(x_{10})] \quad (5.4)$$

$$\Sigma_{ij} = \text{cov}(x_i, x_j) \text{ with } i = 1, \dots, 10 \text{ and } j = 1, \dots, 10 \quad (5.5)$$

The procedure as described is repeated 200 times resulting into 200 datasets coming from the same  $N(\mu, \Sigma)$  distribution. The first 15 rows of the first produced dataset are available in the Table 5.1, in order to have an idea about the structure of our datasets.

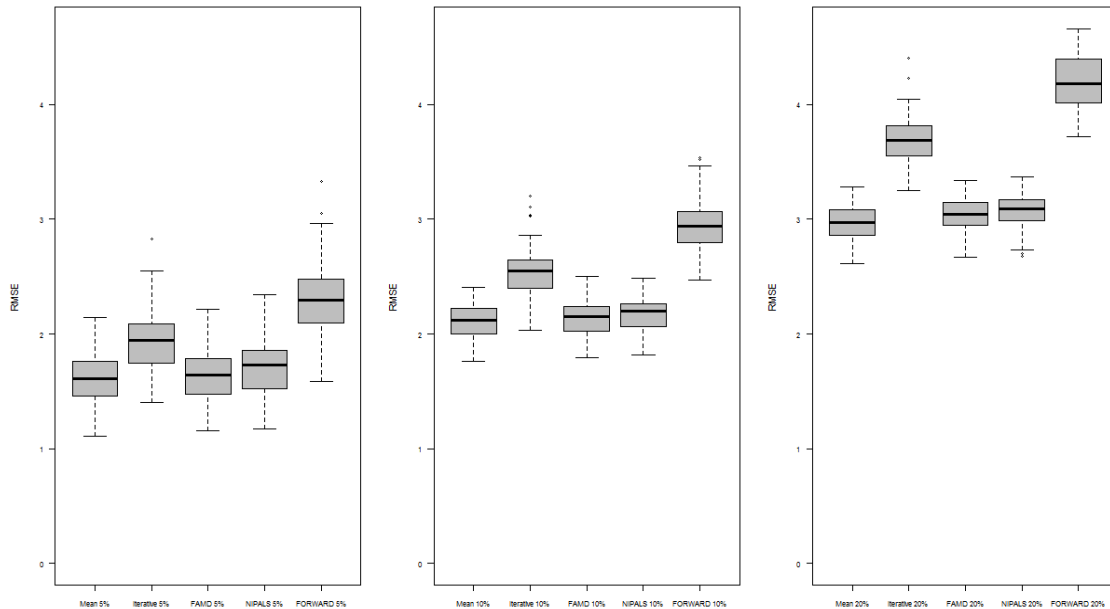
**Table 5.1:** The first 15 rows of the first produced dataset.

```

> mat
      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]      [,8]      [,9]     [,10]
[1,] -0.608303953 -0.610855646  0.20354893 -0.672638010 -1.707045417  0.961816908 -2.083520448 -0.87611480  1.921942228 -0.446261314
[2,]  0.338159655  0.201967808 -2.74975093 -5.702829059 -5.754602370  0.521058576 -3.991996897 -2.34722127  2.242306890  0.933750299
[3,] -0.748055177  0.061773623  0.58976279  0.279680033 -1.889610776 -1.816373856  2.080915388  3.62269739  0.327663131  0.842252631
[4,]  0.367240970 -0.830595089 -0.57532884 -0.774939770 -0.337364697  1.391322196  1.622079550  0.81821850 -1.142026713 -1.683384750
[5,]  0.256587971  0.586312365  0.90247056 -5.033990857 -1.223829186 -0.550989190  0.137460751  1.13038268 -1.980548474 -1.612926311
[6,]  1.108194298  0.481325342  4.72588353  1.085974312 -1.254117840 -1.798481475  1.625314230 -2.00473102  0.179910698 -1.244011522
[7,]  0.562401009  0.739760977  0.53441169  1.184889408 -4.759620530  1.454857849 -0.991644137 -0.77948507  0.969123418 -0.257007844
[8,] -0.726987657  0.195346931 -3.31077257 -3.626718413 -1.950677133  2.343498754 -3.633582749 -5.37370986  1.533596556  0.340568605
[9,]  0.089349783  0.712366715 -1.85019424 -4.789019428  2.509333167  0.939946209  2.142285594 -1.97124855 -2.265925350 -1.937974630
[10,] -0.333829383  0.924452503  3.62200565  3.340476954 -1.787939306 -1.131757872  1.452719536 -1.31048583 -1.141432499 -1.741938877
[11,] -0.754448811 -0.084559641 -1.03330418  3.696683192  2.565229612  0.555863988  0.734839848 -1.67708610 -2.250400855  0.953910158
[12,]  0.864042745  0.363388383  1.70645920 -1.485520163 -4.717797043  1.860510944  0.399711147  0.59417420  0.020082563  0.382048989
[13,] -0.864249092  0.603639955  4.98369400  2.716294145 -3.167887703  1.035865923 -7.875702964  0.98022866  7.170577744 -0.051974412
[14,] -0.150379819 -2.476238723  0.61520219  2.905045936  4.931087290 -2.391704601  4.192556313  2.37765753 -5.137025258  0.737662591
[15,]  0.603776618  0.899610493 -3.24972572 -0.520918152 -4.789138749 -0.430320612  3.968628219 -0.47800996  1.107242984 -1.186374831

```

**Figure 5.1:** Boxplots with the RMSE for each method and for the several scenarios of missing in case we have uncorrelated data from the Multivariate Normal.



We proceed to implement the five methods of imputation which have been listed in the beginning of the chapter, in three different cases of missing. As mentioned before, when the percentage of missing data is 5%, when the percentage of missing data is 10% and when is 20%. We must say that the missing values were generated completely at random. In the Figure 5.1 we can see the boxplots with the Root Mean Square Error. We have three graphs for each percentage of missingness and in each graph there are exist 5 boxplots. The first boxplot shows the value of the RMSE for the Mean Imputation, the second the value of the RMSE for the Iterative Algorithm, the third the value of the RMSE for the Factorial Analysis, the fourth the value of the RMSE for the Nipals Algorithm and the fifth the value of the RMSE for the Forward Imputation.

The first thing that we can notice in the above Figure is that as the percentage of the missing values increases, the value of the Root Mean Square Error is becoming bigger. This is reasonable to happen because the error becomes bigger while the number of non-respondents increases and therefore the information that gathered smaller. Also, we can see that for all the cases the simplest method of Mean Imputation responds better than any other method, although it uses only the mean of the datasets variable and not a sophisticate method for imputing the missing values. Now, for the methods that belong in the Principal Component Analysis, the most reliable is the Factorial Analysis, with the values of the RMSE to be very close to those that we take when we use the Mean Imputation. The method of Nipals Algorithm follows, then the Iterative Algorithm and last one the Forward Imputation. The same ordering among imputation methods holds in each of three percentages of missingness. The comparison is consistence and only the difference in RMSE among various methods and in general increases as the percentage of missingness increases. So, we conclude that the method of Mean Imputation is the best method in this case. The worst method is the Forward Imputation, since the RMSE

becomes much bigger. Also, the RMSE relevant to Factorial Analysis is very close in that which is relevant to Nipals Algorithm.

## 5.1.2 Correlated Data from the Multivariate Normal

Extending the previous case we simulate in this paragraph from the Multivariate Normal with a non-diagonal covariance matrix. The new covariance matrix is given in Table 5.2 and it has been generated with only condition to be positive definite. The dimension of each simulated matrix, same as before, will be  $150 \times 10$ , i.e. we will have 150 individuals and 10 continuous variables. The numbers of iterations will be 200 as in previous paragraph. In the Table 5.3, we take the first 15 rows of the first matrix in order to have a clear view of our data.

Table 5.2: The covariance matrix  $\Sigma$ , which used to produce our datasets.

```
> sigma
```

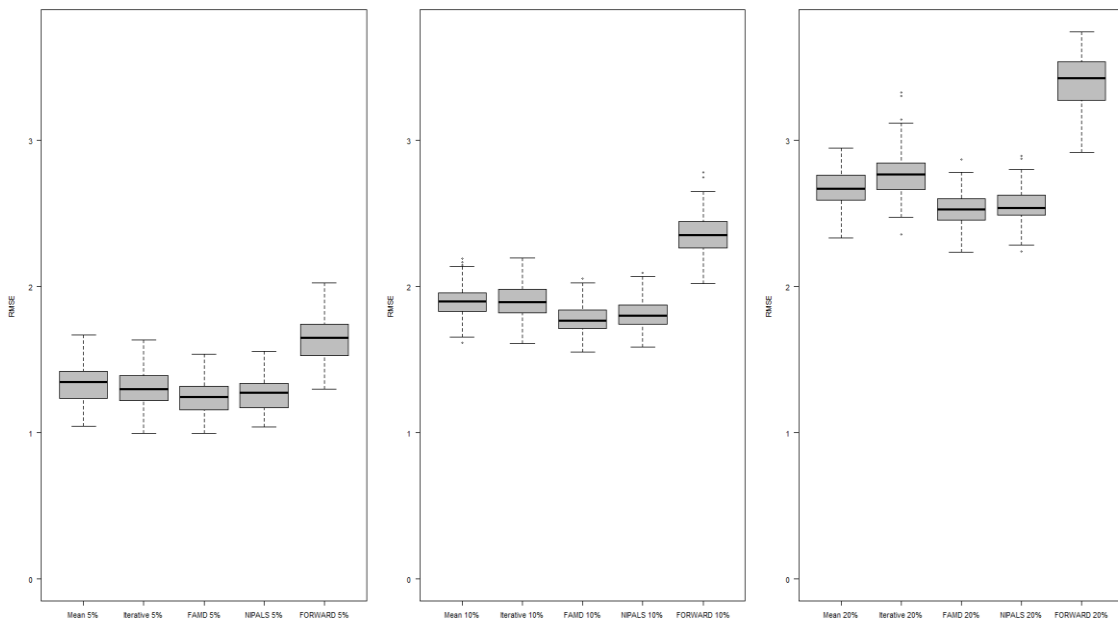
	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
[1,]	4.3846385	1.07377270	0.34090167	-0.1441265	-1.91363765	-0.29075190	-0.85112740	0.43967253	-0.7084080	0.9757300
[2,]	1.0737727	5.53013050	-0.14830685	0.5258131	1.08983811	2.22776831	-0.58513508	-0.04831601	0.5452776	-0.2301040
[3,]	0.3409017	-0.14830685	5.95793936	-0.7335754	-0.05816439	1.10394594	0.68802904	-0.25825427	-1.5069579	-0.4492987
[4,]	-0.1441265	0.52581309	-0.73357539	4.0477902	-1.82416116	0.66703115	-1.94444317	1.40617776	-1.5214683	-0.6548184
[5,]	-1.9136377	1.08983811	-0.05816439	-1.8241612	4.61099784	0.02801904	1.43887391	0.75580479	1.1447300	-0.3876350
[6,]	-0.2907519	2.22776831	1.10394594	0.6670312	0.02801904	3.09258565	-0.45107498	-1.00175624	-1.0300657	0.7978354
[7,]	-0.8511274	-0.58513508	0.68802904	-1.9444432	1.43887391	-0.45107498	4.75104636	-0.07225724	1.0559540	-0.4924725
[8,]	0.4396725	-0.04831601	-0.25825427	1.4061778	0.75580479	-1.00175624	-0.07225724	3.96898898	0.2075341	-0.3913062
[9,]	-0.7084080	0.54527765	-1.50695785	-1.5214683	1.14472997	-1.03006566	1.05595400	0.20753408	4.6391144	0.4900769
[10,]	0.9757300	-0.23010398	-0.44929867	-0.6548184	-0.38763504	0.79783536	-0.49247249	-0.39130622	0.4900769	3.4868749

Table 5.3: The first 15 rows of the first produced dataset.

```

> mat
      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]      [,8]      [,9]     [,10]
[1,] 0.68060784 4.74844026 -1.535240845 -0.163770136 -0.242530055 2.505905336 -0.350516692 -1.78763939 -1.33495193 -1.750511223
[2,] -1.51931652 -5.36088974 -0.708308738 -1.501121464 0.497395049 -4.800101504 2.175862969 1.84932793 1.09760212 -1.981567830
[3,] 0.70797540 -0.83681146 2.136194068 -2.064602310 0.898575654 0.320782371 1.431560763 -0.16381968 0.87783258 0.558159878
[4,] -0.49552587 -0.43724293 3.984088359 4.086495668 -1.120785879 1.076409263 -5.174623392 2.90845542 -4.76511588 0.516500447
[5,] -1.30937325 -2.53080602 1.658345967 2.586688066 -3.287194337 -0.060431702 -2.090120917 0.49707103 -1.96008248 -2.748779922
[6,] 0.83298744 0.16222049 2.456288074 1.928959026 1.457728175 -0.366027913 -2.496808356 2.98170377 -0.09634461 1.446913833
[7,] 0.04902123 1.11017525 -2.294018646 -2.270906398 0.148194802 -0.617319888 -0.445055670 -1.65759964 2.33765737 -3.304746248
[8,] 2.49554227 1.98063454 -2.002391256 0.022115927 -0.480986523 -0.293429402 1.732038338 2.87874865 4.13174903 0.507054497
[9,] 1.98114858 -2.83891576 1.668747345 -2.137357377 0.784390956 -3.415527199 1.186195177 -0.64725702 -3.86621015 -0.710358757
[10,] 2.72515667 2.72980617 0.139362175 0.934978630 -1.454265045 -0.020827618 -0.546628362 2.58321457 0.95029227 -2.887539639
[11,] -0.91428621 -1.95233465 -5.172268245 0.740869919 -0.746693749 -2.714719205 0.193401818 0.59756841 2.40539522 1.276517655
[12,] 1.05463958 3.53557555 2.423723860 -2.203018621 0.734118969 0.851960475 2.244169543 -0.47133545 0.45864663 -3.971031868
[13,] -0.61708191 1.10466104 -0.523756109 -0.970956944 1.770065864 0.007996194 0.236509818 -4.99432328 -0.30208280 0.532367359
[14,] 0.91936616 -2.52601397 3.322129639 2.906555135 -2.213286751 0.649923582 -2.566007802 1.49774884 -5.69981129 -0.851698174
[15,] -3.06076073 -1.43961092 -0.225779233 -1.809788082 1.838659463 -0.992215595 0.645521638 -2.50651410 -1.28299335 -0.930113159
    
```

Figure 5.2: Boxplots with the RMSE for each method and for the several scenarios of missing in case we have correlated data from the Multivariate Normal.



In Figure 5.2 we can see the boxplots with the RMSE presented in a similar manner as described in previous paragraph, in case we have 5%, 10% and 20% missing values for the Multivariate Normal distribution with correlated variables. Our main conclusions with respect to how the imputation techniques compare are: Firstly, a similarity is that in both cases the RMSE is growing up while the percentage of missing values becomes bigger. Secondly, the value of RMSE for the method of Forward Imputation is larger for



all the scenarios, something that is shows us that this method is less reliable either we have correlated variables either not.

With respect to the comparison of the methods in the case of datasets with uncorrelated variables, we note that in case of correlated variables the simple method of Mean Imputation is not the best anymore. Most reliable method according the analysis, is the method of Factorial Analysis with close competitor the method of Nipals Algorithm. Furthermore, we notice that the approach of Iterative Algorithm and the Mean Imputation seems to give comparable results, at least in the cases we have 5% and 10% missingness.

### 5.1.3 Data from the Multivariate Skew Normal

The second distribution we will use to simulate data is the Multivariate Skew Normal and we consider that the distribution is negative skewed. The number of the iterations will be 200 as before, with dimensions of each dataset  $150 \times 10$ . The Table 5.4 give the first 15 rows of the first produced dataset to get an indication of the numeric values that are exist in the specific datasets.

**Table 5.4:** The first 15 rows of the first produced dataset.

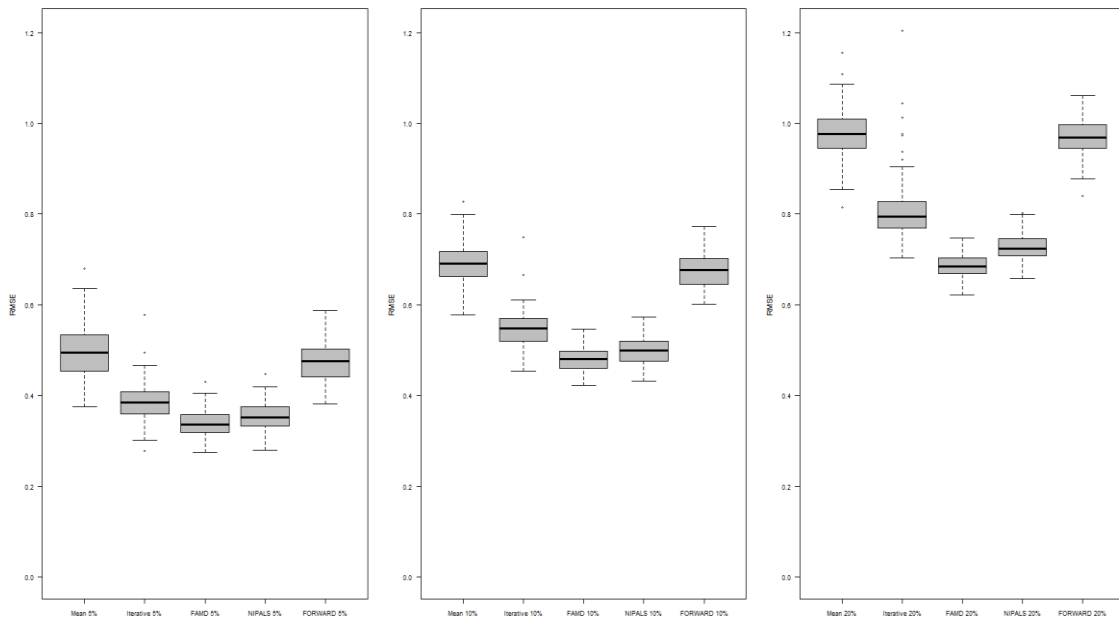
```

> mat
      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]      [,8]      [,9]      [,10]
[1,] -0.823086600 0.066228200 0.083708708 0.02124744 0.13134707 -0.0195168606 -0.48452399 -0.398052591 -0.05428746 0.241661523
[2,] -0.155805095 -0.425704415 -0.417588101 -0.60702418 0.14872044 0.1629053052 -0.73255480 -0.817467149 0.15650147 0.210799848
[3,] -0.731335036 -0.211874086 -0.929057301 -0.22177801 -0.80851277 -0.0330656130 -0.52375418 -0.023670764 -0.38257036 -0.616892002
[4,] -1.534233243 -0.847795403 -1.822749347 -2.38790255 -1.08576897 -1.7133343361 -1.32817437 -1.419463287 -0.99362811 -1.708023425
[5,] -0.952234317 -0.151992613 -1.027392843 -0.52992630 0.48079636 -0.6829377053 -0.47466034 0.126243765 -0.68911485 -0.343573951
[6,] -0.965943504 -0.618259064 -0.588493573 -1.47209208 -0.64071106 -0.4478956923 -0.85979967 -1.051013195 -0.19784138 -0.956058854
[7,] -0.323020725 -0.276801453 -1.540684725 -0.80808847 -1.09340829 -0.7697837285 -0.31982780 0.512797060 -1.70524571 -0.523983987
[8,] 0.010193369 -0.619445543 -0.473102581 -0.67427042 -0.88521385 -1.0732394924 0.42349078 -0.777476371 -0.03189843 -1.216552110
[9,] -2.280266819 -1.715446087 -1.631422314 -2.03981778 -1.35828990 -1.4605966662 -1.50579399 -2.180623181 -1.75006205 -1.301439825
[10,] 0.615535278 -0.331286009 -0.526537749 0.19915465 -0.50593250 0.1613683581 0.23970883 0.759928919 -0.68682272 -0.364380093
[11,] -0.019283989 -0.707237280 -0.803991329 -0.49384742 0.66840825 -0.5195475805 -0.02926041 -0.274144578 0.20194539 0.372799739
[12,] 0.506143112 -0.636630211 -0.427015905 -0.21000406 -0.41062642 -0.4088426651 -0.35022874 -0.003156198 0.45269203 -0.526288284
[13,] -0.423871791 -0.780217229 -1.823758074 -1.44504745 -1.19757334 -0.3178988236 -0.31050268 -0.497303674 -1.09025461 -0.858121987
[14,] 0.768125311 -0.272047876 0.176454587 -1.22432996 -0.24291877 -0.5934931328 -0.15295130 -1.422787938 -0.75386108 -0.543128252
[15,] -0.584739473 -0.936011809 -1.126396853 -1.13805920 -1.16769108 -0.1258976291 -0.93318814 -1.213145409 -1.08719227 -2.146052834
.....
.....

```



**Figure 5.3:** Boxplots with the RMSE for each method and for the several scenarios of missing in case we have data from the Multivariate Skew Normal.



We proceed as in previous cases with the implementation of each imputation method for each case of missingness percentage and the results we obtain for the RMSE are included in Figure 5.3. The most reliable method in the case we have simulated data from the Multivariate Skew Normal is the Factorial Analysis. But, we can also notice that the Nipals Algorithm is very close to that method in the scenarios of 5% and 10% missing, while it seems to be more distant when the percentage of missing is 20%. In addition with the Multivariate Normal the Forward Imputation is not the worst approach. Mean Imputation appears to have the biggest values of RMSE in all the cases of missing, something that informs us that it is the worst method. So, if we want to order the methods with respect to the minimum RMSE, we note that first comes the Factorial Analysis, next the Nipals Algorithm, then the Iterative Algorithm, after the Forward Imputation and last the technique of Mean Imputation. In general, as the dataset becomes more demanding (correlated variables or skewed shape) the standard technique of Mean Imputation becomes inadequate and consistently the Factorial Analysis performs better with close competitive the Nipals Algorithm. Correlated variables or skewed distributions is more realistic scenario for a dataset than uncorrelated and symmetrical distributed variables.

## 5.1.4 A Real Dataset with Continuous Variables

In this subparagraph we are going to implement the imputation methods under comparison in a real dataset. The data that we are going to use is available in the package 'sn'. The name of the dataset is `ais` and contains some measurements for athletes collected at the Australian Institute of Sport. More specifically the number of individuals is 202 and the variables that each individual is asked to give answers are:

- **RCC**: This is a variable that gives a measure that doctor use to find how many red blood cells have a person in his blood.
- **White blood cell count**: This is a variable that gives a measure that doctor use to find how many white cells have a person in his blood.
- **Hc**: A variable that gives the Hematocrit of each athlete.
- **Hg**: It is a variable that shows the levels of the Hemoglobin for each athlete.
- **Fe**: It is a variable that shows the plasma ferritin concertation in his blood.
- **BMI**: It is a variable that gives us the body mass index of each athlete. It is calculated by the following equation,  $BMI = weight / height^2$ .
- **SSF**: A variable which has the sum of skinfolds of each athlete.
- **Bfat**: A variable which contains the body fat percentage of each athlete.
- **LBM**: A variable which contains the lean body mass of each athlete.
- **Ht**: A variable which contains the height of each athlete in cm.
- **Wt**: A variable which contains the weight of each athlete in kg.

Table 5.5 contains the first 20 athletes and all the available values for the measurable variables. Table 5.6 provides with some descriptive statistics. Also, in Figure 5.4 we have histograms for the variables of our dataset. As we can see, the data are not coming from a symmetrical distribution.



Table 5.5: The first 20 rows of the real dataset ais.

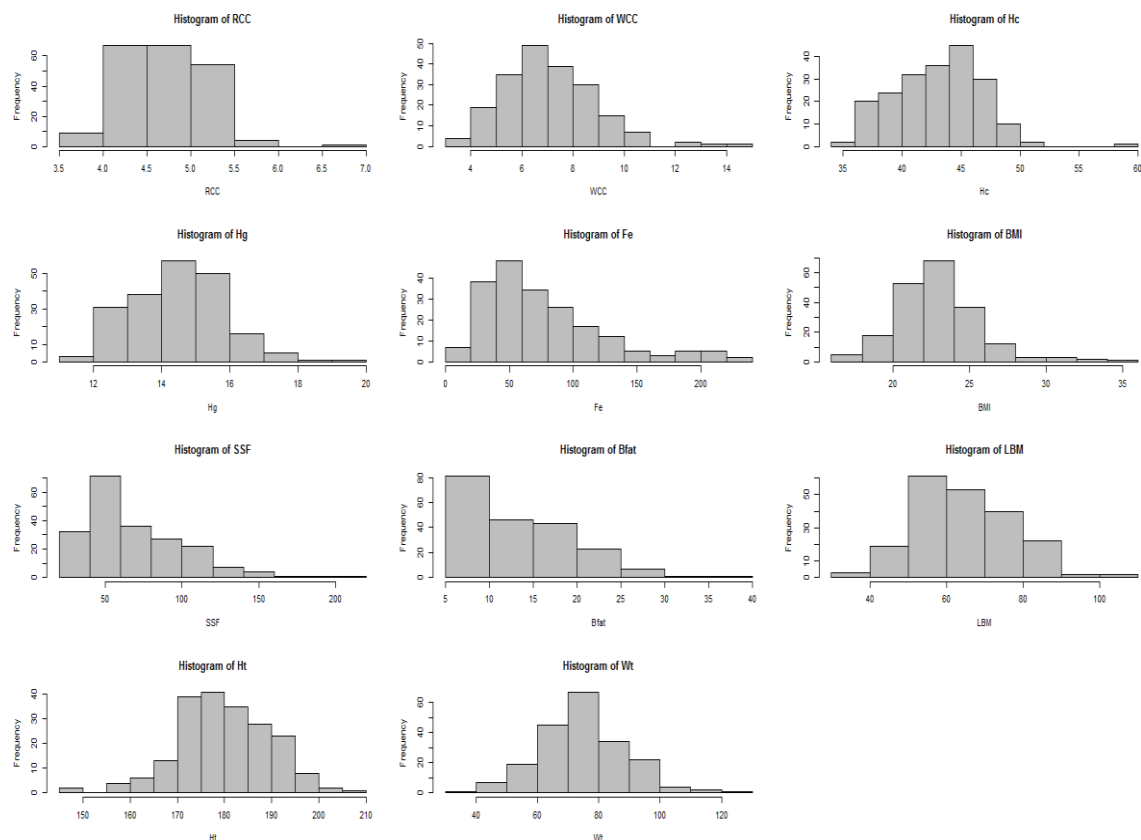
```

> ais
  RCC   WCC   Hc   Hg   Fe   BMI   SSF   Bfat   LBM   Ht   Wt
1  3.96  7.50 37.5 12.3  60 20.56 109.1 19.75  63.32 195.9 78.90
2  4.41  8.30 38.2 12.7  68 20.67 102.8 21.30  58.55 189.7 74.40
3  4.14  5.00 36.4 11.6  21 21.86 104.6 19.88  55.36 177.8 69.10
4  4.11  5.30 37.3 12.6  69 21.88 126.4 23.66  57.18 185.0 74.90
5  4.45  6.80 41.5 14.0  29 18.96  80.3 17.64  53.20 184.6 64.60
6  4.10  4.40 37.4 12.5  42 21.04  75.2 15.58  53.77 174.0 63.70
7  4.31  5.30 39.6 12.8  73 21.69  87.2 19.99  60.17 186.2 75.20
8  4.42  5.70 39.9 13.2  44 20.62  97.9 22.43  48.33 173.8 62.30
9  4.30  8.90 41.1 13.5  41 22.64  75.1 17.95  54.57 171.4 66.50
10 4.51  4.40 41.6 12.7  44 19.44  65.1 15.07  53.42 179.9 62.90
11 4.71  5.30 41.4 14.0  38 25.75 171.1 28.83  68.53 193.4 96.30
12 4.62  7.30 43.8 14.7  26 21.20  76.8 18.08  61.85 188.7 75.50
13 4.35  7.80 41.4 14.1  30 22.03 117.8 23.30  48.32 169.1 63.00
14 4.26  6.20 41.0 13.9  48 25.44  90.2 17.71  66.24 177.9 80.50
15 4.63  6.00 43.7 14.7  30 22.63  97.2 18.77  57.92 177.5 71.30
16 4.36  5.80 40.3 13.3  29 21.86  99.9 19.83  56.52 179.6 70.50
17 3.91  7.30 37.6 12.9  43 22.27 125.9 25.16  54.78 181.3 73.20
18 4.51  8.30 43.7 14.7  34 21.27  69.9 18.04  56.31 179.7 68.70
19 4.37  8.10 41.8 14.3  53 23.47  98.0 21.79  62.96 185.2 80.50
20 4.90  6.90 44.0 14.5  59 23.19  96.8 22.25  56.68 177.3 72.90

```

Table 5.6: Descriptive statistics for the real dataset ais.

	RCC	WCC	HC	Hg	Fe	BMI	SSF	Bfat	LBM	Ht	Wt
Min.	3.8	3.3	35.9	11.6	8.0	16.7	28.0	5.6	34.3	148.9	37.8
1 <sup>st</sup> Q.	4.3	5.9	40.6	13.5	41.2	21.0	43.8	8.5	54.6	174.0	66.5
Median	4.7	6.8	43.5	14.7	65.5	22.7	58.6	11.6	63.0	179.7	74.4
Mean	4.7	7.1	43.0	14.5	76.8	22.9	69.0	13.5	64.8	180.1	75.0
3 <sup>rd</sup> Q.	5.0	8.2	45.5	15.5	97.0	24.4	90.3	18.0	74.7	186.2	84.1
Max.	6.7	14.3	59.7	19.2	234.0	34.4	200.8	35.5	106.0	209.4	123.0

**Figure 5.4:** Histograms for the dataset ais.

The approach that has been followed is the same as in simulated data. In the observed matrix we make missing values in percentage 5%, 10% and 20%. Then all the methods of imputation that are available for the continuous variables were used to create a new complete matrix with the same values for the non-missing entries and with the proposed imputed values for the missing entries. The indicator of RMSE used again to show us which technique performs better in every case. Table 5.7 includes the resulting values of RMSE for each method, for each percentage of missingness. The first observation from Table 5.7 is that the values of the RMSE are very bigger from those we had when our data were simulated. Comparing the methods according to RMSE we notice that when we are in the case of 5% missingness the best method is the approach of Forward Imputation. As the percentage of missingness increases (10% and 20%) the result does not hold and in particular the method performs worst that all the others. The best method of imputation in these scenarios seems to be again the Factorial Analysis. The Iterative Algorithm technique is very close with the method of the Factorial Analysis. More less reliable results give the method of Nipals Algorithm and the Mean Imputation.

**Table 5.7:** Table with the values of the RMSE for each method of imputation and for the different scenarios of missing.

	5%	10%	20%
MEAN	14.42	19.61	29.47
ITERATIVE	10.97	17.59	23.42
FAMD	11.06	17.20	23.14
NIPALS	14.20	19.25	28.89
FORWARD	09.51	25.18	30.64

## 5.2 Mixed Type Of Data

In previous sub-sections we analyzed datasets with only continuous variables. But in the most cases the datasets that we will be invited to analyze in real situations will be mixed. There are not many methods belonging in the category of Principal Component Analysis. The main methods that can be found in bibliography and can deal with continuous and categorical variables are the Factorial Analysis and the Forward Imputation. Beyond these two methods, we will use in our analysis one more approach, the method of Random Forests. This is a classification method, but it can also help us to impute missing dataset. It is a non-parametric method, which does not belong in the category of Principal Component Analysis and it is not very commonly used. As all the non-parametric methods it makes no assumptions about the distribution of our variables. It performs efficiently on large datasets and maintains accuracy when a large proportion of the data are missing. For the comparison of the above three methods we will use both RMSE and PFC. Essentially, we will separate the dataset in two parts. The one part will be the categorical variables and the other part the continuous. For the continuous variables we will calculate the RMSE, whereas for the categorical variables we will calculate the PFC. As we had mentioned, PFC is an indicator which gives us the percentage of falsely classified observations in the part of categorical variables. It is calculated dividing the number of the missclassified observations with the total number of missing for the categorical variables. In the next paragraph we will analyze a real mixed dataset in order to see which from the above three methods performs better.



## 5.2.1 Mixed Real Dataset Iris

The dataset iris is a mixed type dataset which contains 4 continuous variables and 1 categorical variable. The categorical variable is an indicator that takes three values and show us in which species belong each of 150 different plants. There are three different types of plants, the setosa, the versicolor and the virginica. The continuous variables are 4 and are listed below:

- Sepal.Length: A variable that contains the sepal length of each plant.
- Sepal.Width: A variable that contains the sepal width of each plant.
- Petal.Length: A variable that contains the petal length of each plant.
- Petal.width: A variable that contains the petal width of each plant.

In Table 5.8 we have the first 20 rows of our dataset, whereas in Table 5.9 we can see some descriptive statistics about the 4 continuous variables of our data and the number of plants that belong in each type of our categorical variable. Furthermore, in Figure 5.5 we give histograms for the numeric variables of dataset iris.

Table 5.8: The first 20 rows of the real dataset ais.

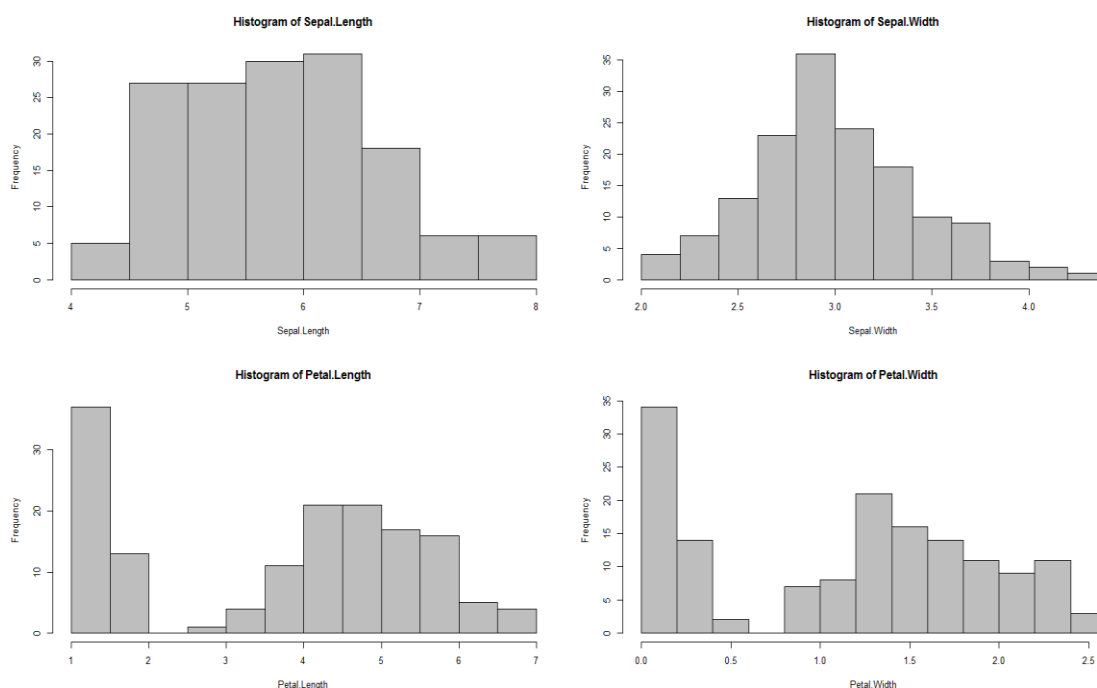
```
> iris
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1           5.1           3.5           1.4           0.2   setosa
2           4.9           3.0           1.4           0.2   setosa
3           4.7           3.2           1.3           0.2   setosa
4           4.6           3.1           1.5           0.2   setosa
5           5.0           3.6           1.4           0.2   setosa
6           5.4           3.9           1.7           0.4   setosa
7           4.6           3.4           1.4           0.3   setosa
8           5.0           3.4           1.5           0.2   setosa
9           4.4           2.9           1.4           0.2   setosa
10          4.9           3.1           1.5           0.1   setosa
11          5.4           3.7           1.5           0.2   setosa
12          4.8           3.4           1.6           0.2   setosa
13          4.8           3.0           1.4           0.1   setosa
14          4.3           3.0           1.1           0.1   setosa
15          5.8           4.0           1.2           0.2   setosa
16          5.7           4.4           1.5           0.4   setosa
17          5.4           3.9           1.3           0.4   setosa
18          5.1           3.5           1.4           0.3   setosa
19          5.7           3.8           1.7           0.3   setosa
20          5.1           3.8           1.5           0.3   setosa
```



Table 5.9: Descriptive statistics for the real dataset ais.

```
> summary(iris)
 Sepal.Length Sepal.Width Petal.Length Petal.Width Species
Min. :4.300 Min. :2.000 Min. :1.000 Min. :0.100 setosa :50
1st Qu.:5.100 1st Qu.:2.800 1st Qu.:1.600 1st Qu.:0.300 versicolor:50
Median :5.800 Median :3.000 Median :4.350 Median :1.300 virginica :50
Mean :5.843 Mean :3.057 Mean :3.758 Mean :1.199
3rd Qu.:6.400 3rd Qu.:3.300 3rd Qu.:5.100 3rd Qu.:1.800
Max. :7.900 Max. :4.400 Max. :6.900 Max. :2.500
```

Figure 5.5: Histograms for the continuous variables of dataset iris.



The procedure that we are going to follow is the same with the other chapters. In the real dataset we will create missingness and for the different percentage of missing values we will compare the two methods of Imputation that belong in the category of Principal Component Analysis, i.e. the Factorial Analysis and the Forward Imputation. As we refer above the comparison of the methods cannot be done only with the RMSE, because of the presence of a categorical variable. So, first we will see the value of the RMSE for the continuous variables and then the values of the PFC for the categorical variable. Table 5.10 presents the values taken for these two indicators for the different scenarios of missingness. The results of the approach of the non-parametric method Random Forests are available also in this table.

**Table 5.10:** RMSE and PFC for each method of imputation and for the different scenarios of missingness.

5%			
	FAMD	FORWARD	RANDOM FOREST
RMSE	0.165	0.354	0.113
PFC	0.1 (1/10)	0.2 (2/10)	0.1 (1/10)
10%			
	FAMD	FORWARD	RANDOM FOREST
RMSE	0.274	0.242	0.177
PFC	0.133 (2/15)	0.06 (1/15)	0.06 (1/15)
20%			
	FAMD	FORWARD	RANDOM FOREST
RMSE	0.318	0.536	0.257
PFC	0.275 (8/29)	0.310 (9/29)	0.06 (2/29)

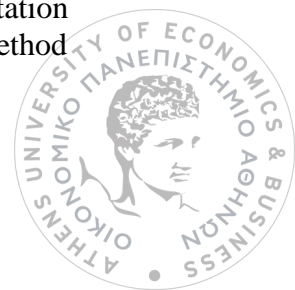
The RMSE and the PFC are becoming bigger while the number of the missing values is increasing. The best and more reliable method seems to be the approach of Random Forest because the values of error are smaller when we use it instead of the other two. Now, between the two methods that belong in the category of Principal Component Analysis, we can see that no one from these two methods responds better in all the cases of missingness. When the scenario that we create is that we have 5% missing data the approach of Factorial Analysis is more reliable with the smallest values of error. When we are in the case of 10% missing data the method with the smallest error is the technique of Forward Imputation. Last, when the proportion of missing values is 20% the best method is the Factorial Analysis. So, the only thing that we can say is that the best method in total is the non-parametric method of Random Forest.

# Chapter 6

## 6.1 Conclusions With A Critical View

In Chapter 5 we make several experiments in R. We have generated data from the Multivariate Normal and the Skew Normal distribution and we also use some real datasets. For all the cases, we create missing values to the initial matrices. The scenarios of missingness were different, i.e we have missing values with percentage 5%, 10% and 20%. Then all the available methods of imputation that belong in the category of Principal Component Analysis used to create a new full observed dataset. All these methods were compared with each other in each case, by using the RMSE and the PFC. The results that we take differ depending on the occasion. The differences in the effectiveness of each method, had to do with the type of the data and the different percentages of missingness.

Firstly, we take simulated data from the Multivariate Normal with uncorrelated variables. In this case, we immediately see that as the percentage of missing values is getting bigger the RMSE getting bigger too. Interestingly the simple method of Mean Imputation, seems to respond better than the other methods. This is interesting, because the other algorithmic approaches that were used, begin with the creation of the initial full observed matrix by substituting the missing values with the mean of the corresponding variable and continue until the algorithm converge. Now, for all the other methods of interest, the best results give the technique of Factorial Analysis for mixed datasets. While the worst results, give the method of Forward Imputation. We continued the experiments with simulated data from the Multivariate Normal, but with correlated variables. Same here, as the percentage of missing values is getting bigger the error increases. Although we had again data from the Multivariate Normal, Mean Imputation is not the best method as happened before. In this case, the best results give the method



of Factorial Analysis for mixed datasets, while the method of Nipals algorithm is very close. The worst method of imputation is the same as before, the Forward Imputation. Then, we proceed with simulated data from the Skew Normal distribution. The results that we obtain differ in comparison with the previous experiments regarding Multivariate Normal. The best method is still the Factorial Analysis with results similar with the Nipals Algorithm, but the worst method is not the Forward Imputation, but the Mean Imputation. So, we notice that the method of Mean Imputation, which was the approach with the best results in the case of Multivariate Normal with uncorellated variables, becomes here a not reliable method. Finally, we conclude our experiments with real data sets. The first dataset was the dataset ais, which contains only numeric variables. As in all the cases before, while the percentage of missing values increase the RMSE increases too. By noticing the values of RMSE, we observe that the Forward Imputation is the most appropriate method of imputation in the scenario with 5% missing values, while it becomes the worst method in the cases with 10% and 20% missing values. The best method in the scenarios of 10% and 20% missingness is the Factorial Analysis. Another remark is that the values of RMSE that we take with the use of Factorial Analysis are very close with the values of RMSE that we take by using the Nipals algorithm. The second real dataset that has been used is the mixed mode dataset iris. Due to the fact, that in this dataset exists continuous, but also a categorical variable, the RMSE measure has been used for the continuous variables and the PFC for the categorical variable. As in all the other cases, same here, as the percentage of missing values is getting bigger the values of error increases. We conclude that the best technique to deal with missingness in this dataset is the non-parametric method of Random Forests. From the methods that belongs in Principal Component Analysis the results are different depending of the percentage of missing values. In the scenario we have 5% or 20% missing values the approach of Factorial Analysis give better results. But, in the scenario we have 10% missing values the approach of Forward Imputation seems to be more reliable.

Summing all the results we draw in some main conclusions. The first is that in all the cases, regardless if we have simulated or real data, as expected when the percentage of missing values is getting bigger, the loss of information and the error either the RMSE, or the PFC increases. The main result, is that the method of Factorial Analysis performs better than all other competitors when the data are coming from Multivariate Normal distribution with correlated variables (a very common situation in real datasets) or in Skew Multivariate Normal distribution. Certainly, we can say that totally the Factorial Analysis seems to be the most reliable method. Even in the cases that is not appear to be the most appropriate method, is very close to that which gives the best results. Now, in the simulated datasets, the best method is reliable in all the scenarios, i.e it does not matter the percentage of missing values that we have. A fact which is not confirmed with the experiments based on real datasets. There, as we saw, the results are becoming different depending the percentage of missing values.



## Appendix

A1

```

fun<-function (ma,matr,maim,m4) {
  rm<-0
  p<-0
  for(i in 1:150){
    for(j in 1:10){
      if (m4[i,j]==TRUE){
        p<-p+1
        rm<-rm+(matr[i,j]-maim[i,j])^2
      }
    }
  }
  rm<-sqrt (rm/150)
  return(c (rm,p) )
}
library (ForImp)
library (missMDA)
library (pcaMethods)
iter<-200
RMSE1<-matrix(rep(0, iter*5), ncol=5)
for(i in 1:iter){
  sigma<-diag(runif(10,0,10))
  mat<-rmvnorm(n=150,mean=rep(0,10),sigma=sigma)
  m1<-missingmat(mat,nummissing=75,pattern="r")
  m2<-meanimp(m1)
  m4<-is.na(m1)
  RMSE1[i,1]<-fun(m1,mat,m2,m4)[1]
  MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
  mi2<-MI2$completeObs
  RMSE1[i,2]<-fun(m1,mat,mi2,m4)[1]
  mfa1<-as.data.frame(m1)
  imp<-imputeFAMD(mfa1,ncp=2)
  mfa<-imp$tab.disj
  RMSE1[i,3]<-fun(mfa1,mat,mfa,m4)[1]
  mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
  RMSE1[i,4]<-fun(m1,mat,mnip,m4)[1]
  mfo<-ForImp(m1,p=2)
  RMSE1[i,5]<-fun(m1,mat,mfo,m4)[1]
}
dimnames(RMSE1)[[2]]<-c("Mean 5%", "Iterative 5%", "FAMD 5%", "NIPALS
5%", "FORWARD 5%")
boxplot(RMSE1,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE1)),ylab="RMSE
")
RMSE2<-matrix(rep(0, iter*5), ncol=5)
for(i in 1:iter){
  mat<-rmvnorm(n=150,mean=rep(0,10),sigma=sigma)
  m1<-missingmat(mat,nummissing=150,pattern="r")
  m2<-meanimp(m1)
  m4<-is.na(m1)
  RMSE2[i,1]<-fun(m1,mat,m2,m4)[1]
  MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)

```



```

mi2<-MI2$completeObs
RMSE2[i,2]<-fun(m1,mat,mi2,m4)[1]
mfal<-as.data.frame(m1)
imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$tab.disj
RMSE2[i,3]<-fun(mfal,mat,mfa,m4)[1]
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
RMSE2[i,4]<-fun(m1,mat,mnip,m4)[1]
mfo<-ForImp(m1,p=2)
RMSE2[i,5]<-fun(m1,mat,mfo,m4)[1]
}
dimnames(RMSE2)[[2]]<-c("Mean 10%", "Iterative 10%", "FAMD 10%",
"NIPALS 10%", "FORWARD 10%")
boxplot(RMSE2,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE2)),ylab="RMSE
")
RMSE3<-matrix(rep(0, iter*5), ncol=5)
for(i in 1:iter){
mat<-rmvnorm(n=150,mean=rep(0,10),sigma=sigma)
m1<-missingmat(mat,nummissing=300,pattern="r")
m2<-meanimp(m1)
m4<-is.na(m1)
RMSE3[i,1]<-fun(m1,mat,m2,m4)[1]
MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
mi2<-MI2$completeObs
RMSE3[i,2]<-fun(m1,mat,mi2,m4)[1]
mfal<-as.data.frame(m1)
imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$tab.disj
RMSE3[i,3]<-fun(mfal,mat,mfa,m4)[1]
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
RMSE3[i,4]<-fun(m1,mat,mnip,m4)[1]
mfo<-ForImp(m1,p=2)
RMSE3[i,5]<-fun(m1,mat,mfo,m4)[1]
}
dimnames(RMSE3)[[2]]<-c("Mean 20%", "Iterative 20%", "FAMD 20%",
"NIPALS 20%", "FORWARD 20%")
boxplot(RMSE3,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")
par(mfrow=c(1,3))
boxplot(RMSE1,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")
boxplot(RMSE2,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")
boxplot(RMSE3,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")

```

A2

```

Posdef <- function (n, ev = runif(n, 0, 10))
{
Z <- matrix(ncol=n, rnorm(n^2))
decomp <- qr(Z)
Q <- qr.Q(decomp)
R <- qr.R(decomp)
d <- diag(R)
ph <- d / abs(d)
O <- Q %*% diag(ph)
Z <- t(O) %*% diag(ev) %*% O
return(Z)
}

```



```

}
library(ForImp)
library(missMDA)
library(pcaMethods)
iter<-200
RMSE1<-matrix(rep(0, iter*5), ncol=5)
sigma<-Posdef(10)
for(i in 1:iter){
mat<-rmvnorm(n=150,mean=rep(0,10),sigma=sigma)
m1<-missingmat(mat,nummissing=75,pattern="r")
m2<-meanimp(m1)
m4<-is.na(m1)
RMSE1[i,1]<-fun(m1,mat,m2,m4)[1]
MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
mi2<-MI2$completeObs
RMSE1[i,2]<-fun(m1,mat,mi2,m4)[1]
mfal<-as.data.frame(m1)
imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$tab.disj
RMSE1[i,3]<-fun(mfal,mat,mfa,m4)[1]
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
RMSE1[i,4]<-fun(m1,mat,mnip,m4)[1]
mfo<-ForImp(m1,p=2)
RMSE1[i,5]<-fun(m1,mat,mfo,m4)[1]
}
dimnames(RMSE1)[[2]]<-c("Mean 5%", "Iterative 5%", "FAMD 5%", "NIPALS
5%", "FORWARD 5%")
boxplot(RMSE1,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE1)),ylab="RMSE
")
RMSE2<-matrix(rep(0, iter*5), ncol=5)
for(i in 1:iter){
mat<-rmvnorm(n=150,mean=rep(0,10),sigma=sigma)
m1<-missingmat(mat,nummissing=150,pattern="r")
m2<-meanimp(m1)
m4<-is.na(m1)
RMSE2[i,1]<-fun(m1,mat,m2,m4)[1]
MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
mi2<-MI2$completeObs
RMSE2[i,2]<-fun(m1,mat,mi2,m4)[1]
mfal<-as.data.frame(m1)
imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$tab.disj
RMSE2[i,3]<-fun(mfal,mat,mfa,m4)[1]
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
RMSE2[i,4]<-fun(m1,mat,mnip,m4)[1]
mfo<-ForImp(m1,p=2)
RMSE2[i,5]<-fun(m1,mat,mfo,m4)[1]
}
dimnames(RMSE2)[[2]]<-c("Mean 10%", "Iterative 10%", "FAMD 10%",
"NIPALS 10%", "FORWARD 10%")
boxplot(RMSE2,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE2)),ylab="RMSE
")
RMSE3<-matrix(rep(0, iter*5), ncol=5)
for(i in 1:iter){
mat<-rmvnorm(n=150,mean=rep(0,10),sigma=sigma)
m1<-missingmat(mat,nummissing=300,pattern="r")
m2<-meanimp(m1)
m4<-is.na(m1)
RMSE3[i,1]<-fun(m1,mat,m2,m4)[1]

```



```

MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
mi2<-MI2$completeObs
RMSE3[i,2]<-fun(m1,mat,mi2,m4)[1]
mfa1<-as.data.frame(m1)
imp<-imputeFAMD(mfa1,ncp=2)
mfa<-imp$tab.disj
RMSE3[i,3]<-fun(mfa1,mat,mfa,m4)[1]
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
RMSE3[i,4]<-fun(m1,mat,mnip,m4)[1]
mfo<-ForImp(m1,p=2)
RMSE3[i,5]<-fun(m1,mat,mfo,m4)[1]
}
dimnames(RMSE3)[[2]]<-c("Mean 20%", "Iterative 20%", "FAMD 20%",
"NIPALS 20%", "FORWARD 20%")
boxplot(RMSE3,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")
par(mfrow=c(1,3))
boxplot(RMSE1,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")
boxplot(RMSE2,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")
boxplot(RMSE3,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")

```

A3

```

library(ForImp)
library(missMDA)
library(sn)
library(pcaMethods)
iter<-200
RMSE1<-matrix(rep(0, iter*5), ncol=5)
sigma<-matrix(rep(.8,100), nrow=10)
diag(sigma)<-1
alpha<-rep(-2,10)
for(i in 1:iter){
mat<-rmsn(n=150, rep(0, length(alpha)), Omega=sigma, alpha=alpha)
mat<-mat[1:150,1:10]
m1<-missingmat(mat,nummissing=75,pattern="r")
m2<-meanimp(m1)
m4<-is.na(m1)
RMSE1[i,1]<-fun(m1,mat,m2,m4)[1]
MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
mi2<-MI2$completeObs
RMSE1[i,2]<-fun(m1,mat,mi2,m4)[1]
mfa1<-as.data.frame(m1)
imp<-imputeFAMD(mfa1,ncp=2)
mfa<-imp$tab.disj
RMSE1[i,3]<-fun(mfa1,mat,mfa,m4)[1]
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
RMSE1[i,4]<-fun(m1,mat,mnip,m4)[1]
mfo<-ForImp(m1,p=2)
RMSE1[i,5]<-fun(m1,mat,mfo,m4)[1]
}
dimnames(RMSE1)[[2]]<-c("Mean 5%", "Iterative 5%", "FAMD 5%", "NIPALS
5%", "FORWARD 5%")
boxplot(RMSE1,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE1)),ylab="RMSE
")
RMSE2<-matrix(rep(0, iter*5), ncol=5)

```



```

for(i in 1:iter){
mat<-rmsn(n=150, rep(0, length(alpha)), Omega=sigma, alpha=alpha)
mat<-mat[1:150,1:10]
m1<-missingmat(mat,nummissing=150,pattern="r")
m2<-meanimp(m1)
m4<-is.na(m1)
RMSE2[i,1]<-fun(m1,mat,m2,m4)[1]
MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
mi2<-MI2$completeObs
RMSE2[i,2]<-fun(m1,mat,mi2,m4)[1]
mfal<-as.data.frame(m1)
imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$tab.disj
RMSE2[i,3]<-fun(mfal,mat,mfa,m4)[1]
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
RMSE2[i,4]<-fun(m1,mat,mnip,m4)[1]
mfo<-ForImp(m1,p=2)
RMSE2[i,5]<-fun(m1,mat,mfo,m4)[1]
}
dimnames(RMSE2)[[2]]<-c("Mean 10%", "Iterative 10%", "FAMD 10%",
"NIPALS 10%", "FORWARD 10%")
boxplot(RMSE2,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE2)),ylab="RMSE
")
RMSE3<-matrix(rep(0, iter*5), ncol=5)
for(i in 1:iter){
mat<-rmsn(n=150, rep(0, length(alpha)), Omega=sigma, alpha=alpha)
mat<-mat[1:150,1:10]
m1<-missingmat(mat,nummissing=300,pattern="r")
m2<-meanimp(m1)
m4<-is.na(m1)
RMSE3[i,1]<-fun(m1,mat,m2,m4)[1]
MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
mi2<-MI2$completeObs
RMSE3[i,2]<-fun(m1,mat,mi2,m4)[1]
mfal<-as.data.frame(m1)
imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$tab.disj
RMSE3[i,3]<-fun(mfal,mat,mfa,m4)[1]
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
RMSE3[i,4]<-fun(m1,mat,mnip,m4)[1]
mfo<-ForImp(m1,p=2)
RMSE3[i,5]<-fun(m1,mat,mfo,m4)[1]
}
dimnames(RMSE3)[[2]]<-c("Mean 20%", "Iterative 20%", "FAMD 20%",
"NIPALS 20%", "FORWARD 20%")
boxplot(RMSE3,col=8,cex.axis=0.8,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")
par(mfrow=c(1,3))
boxplot(RMSE1,col=8,cex.axis=0.9,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")
boxplot(RMSE2,col=8,cex.axis=0.9,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")
boxplot(RMSE3,col=8,cex.axis=0.9,las=1,ylim=c(0,max(RMSE3)),ylab="RMSE
")

A4

library(ForImp)
library(sn)

```



```

data (ais)
ai<-ais[,3:13]
ai<-as.matrix(ai)
m1<-missingmat(ai,nummissing=111,pattern="r")
m2<-meanimp(m1)
m4<-is.na(m1)
fun(m1,ai,m2,m4)
library(missMDA)
MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
mi2<-MI2$completeObs
fun(m1,ai,mi2,m4)
library(missMDA)
mfal<-as.data.frame(m1)
imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$tab.disj
fun(mfal,ai,mfa,m4)
library(pcaMethods)
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
fun(m1,ai,mnip,m4)
mfo<-ForImp(m1,p=2)
fun(m1,ai,mfo,m4)
m1<-missingmat(ai,nummissing=222,pattern="r")
m2<-meanimp(m1)
m4<-is.na(m1)
fun(m1,ai,m2,m4)
MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
mi2<-MI2$completeObs
fun(m1,ai,mi2,m4)
mfal<-as.data.frame(m1)
imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$tab.disj
fun(mfal,ai,mfa,m4)
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
fun(m1,ai,mnip,m4)
mfo<-ForImp(m1,p=2)
fun(m1,ai,mfo,m4)
m1<-missingmat(ai,nummissing=444,pattern="r")
m2<-meanimp(m1)
m4<-is.na(m1)
fun(m1,ai,m2,m4)
MI2<-imputePCA(m1,ncp=2,method="EM",threshold=1e-06,maxiter=1000)
mi2<-MI2$completeObs
fun(m1,ai,mi2,m4)
mfal<-as.data.frame(m1)
imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$tab.disj
fun(mfal,ai,mfa,m4)
mnip<-completeObs(pca(m1,nPcs=2,method="nipals"))
fun(m1,ai,mnip,m4)
mfo<-ForImp(m1,p=2)
fun(m1,ai,mfo,m4)

```

A5

```

library(missMDA)
library(ForImp)
library(missForest)
mat<-iris
mfal<-prodNA(mat,0.05)

```



```

summary(mfa1)
mfa11<-mfa1[,1:4]
mat2<-iris[,1:4]
mat22<-as.matrix(mat2)
imp<-imputeFAMD(mfa1,ncp=2)
mfa<-imp$tab.disj
mfa111<-as.data.frame(mfa11)
mf<-mfa[,1:4]
k<-rep(0,150)
p1<-0
p2<-0
p3<-0
for(i in 1:150){
if((mfa[i,5]>mfa[i,6]) & (mfa[i,5]>mfa[i,7]))
(p1<-p1+1)&(k[i]<-1)
if((mfa[i,6]>mfa[i,5]) & (mfa[i,6]>mfa[i,7]))
(p2<-p2+1)&(k[i]<-2)
if((mfa[i,7]>mfa[i,5]) & (mfa[i,7]>mfa[i,6]))
(p3<-p3+1)&(k[i]<-3)
}
fun<-function(ma,matr,maim,m4){
rm<-0
p<-0
for(i in 1:150){
for(j in 1:4){
if(m4[i,j]==TRUE){
p<-p+1
rm<-rm+(matr[i,j]-maim[i,j])^2
}
}
}
rm<-sqrt(rm/150)
return(c(rm,p))
}
m4<-is.na(mfa11)
fun(mfa111,mat22,mf,m4)
mfa2<-cbind(mfa1[,1],mfa1[,2],mfa1[,3],mfa1[,4],mfa1[,5])
mfaa<-ForImp(mfa2,p=2)
mfor<-mfaa[,1:4]
k<-rep(0,150)
p1<-0
p2<-0
p3<-0
for(i in 1:150){
if(mfaa[i,5]==1)
(p1<-p1+1)&(k[i]<-1)
if(mfaa[i,5]==2)
(p2<-p2+1)&(k[i]<-2)
if(mfaa[i,5]==3)
(p3<-p3+1)&(k[i]<-3)
}
fun(mfa111,mat22,mfor,m4)
mf5<-missForest(mfa1, maxiter = 10, ntree = 100, variablewise = FALSE,
decreasing = FALSE, verbose = FALSE,
mtry = floor(sqrt(ncol(mfa1))), replace = TRUE,
classwt = NULL, cutoff = NULL, strata = NULL,
sampsize = NULL, nodesize = NULL, maxnodes = NULL,
xtrue = NA, parallelize = c('no', 'variables', 'forests'))
mf55<-mf5$ximp

```



```

mixError(mf55,mfal,mat)
mrf<-mf55[,1:4]
fun(mfal111,mat22,mrf,m4)
mat<-iris
mfal<-prodNA(mat,0.1)
summary(mfal)
mfal1<-mfal[,1:4]
mat2<-iris[,1:4]
mat22<-as.matrix(mat2)
imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$stab.disj
mfal11<-as.data.frame(mfal1)
mf<-mfa[,1:4]
k<-rep(0,150)
p1<-0
p2<-0
p3<-0
for(i in 1:150){
if((mfa[i,5]>mfa[i,6]) & (mfa[i,5]>mfa[i,7]))
(p1<-p1+1)&(k[i]<-1)
if((mfa[i,6]>mfa[i,5]) & (mfa[i,6]>mfa[i,7]))
(p2<-p2+1)&(k[i]<-2)
if((mfa[i,7]>mfa[i,5]) & (mfa[i,7]>mfa[i,6]))
(p3<-p3+1)&(k[i]<-3)
}
m4<-is.na(mfal1)
fun(mfal111,mat22,mf,m4)
mfa2<-cbind(mfal[,1],mfal[,2],mfal[,3],mfal[,4],mfal[,5])
mfaa<-ForImp(mfa2,p=2)
mfor<-mfaa[,1:4]
k<-rep(0,150)
p1<-0
p2<-0
p3<-0
for(i in 1:150){
if(mfaa[i,5]==1)
(p1<-p1+1)&(k[i]<-1)
if(mfaa[i,5]==2)
(p2<-p2+1)&(k[i]<-2)
if(mfaa[i,5]==3)
(p3<-p3+1)&(k[i]<-3)
}
fun(mfal111,mat22,mfor,m4)
mf5<-missForest(mfal, maxiter = 10, ntree = 100, variablewise = FALSE,
decreasing = FALSE, verbose = FALSE,
mtry = floor(sqrt(ncol(mfal))), replace = TRUE,
classwt = NULL, cutoff = NULL, strata = NULL,
sampsize = NULL, nodesize = NULL, maxnodes = NULL,
xtrue = NA, parallelize = c('no', 'variables', 'forests'))
mf55<-mf5$ximp
mixError(mf55,mfal,mat)
mrf<-mf55[,1:4]
fun(mfal111,mat22,mrf,m4)
mat<-iris
mfal<-prodNA(mat,0.2)
summary(mfal)
mfal1<-mfal[,1:4]
mat2<-iris[,1:4]
mat22<-as.matrix(mat2)

```



```

imp<-imputeFAMD(mfal,ncp=2)
mfa<-imp$stab.disj
mfa111<-as.data.frame(mfa11)
mf<-mfa[,1:4]
k<-rep(0,150)
p1<-0
p2<-0
p3<-0
for(i in 1:150){
if((mfa[i,5]>mfa[i,6]) & (mfa[i,5]>mfa[i,7]))
(p1<-p1+1)&(k[i]<-1)
if((mfa[i,6]>mfa[i,5]) & (mfa[i,6]>mfa[i,7]))
(p2<-p2+1)&(k[i]<-2)
if((mfa[i,7]>mfa[i,5]) & (mfa[i,7]>mfa[i,6]))
(p3<-p3+1)&(k[i]<-3)
}
m4<-is.na(mfa11)
fun(mfa111,mat22,mf,m4)
mfa2<-cbind(mfal[,1],mfal[,2],mfal[,3],mfal[,4],mfal[,5])
mfaa<-ForImp(mfa2,p=2)
mfor<-mfaa[,1:4]
k<-rep(0,150)
p1<-0
p2<-0
p3<-0
for(i in 1:150){
if(mfaa[i,5]==1)
(p1<-p1+1)&(k[i]<-1)
if(mfaa[i,5]==2)
(p2<-p2+1)&(k[i]<-2)
if(mfaa[i,5]==3)
(p3<-p3+1)&(k[i]<-3)
}
fun(mfa111,mat22,mfor,m4)
mf5<-missForest(mfal, maxiter = 10, ntree = 100, variablewise = FALSE,
decreasing = FALSE, verbose = FALSE,
mtry = floor(sqrt(ncol(mfal))), replace = TRUE,
classwt = NULL, cutoff = NULL, strata = NULL,
sampsize = NULL, nodesize = NULL, maxnodes = NULL,
xtrue = NA, parallelize = c('no', 'variables', 'forests'))
mf55<-mf5$ximp
mixError(mf55,mfal,mat)
mrf<-mf55[,1:4]
fun(mfa111,mat22,mrf,m4)

```



## References

- 1) Levy, Paul S. , Lemeshow, Stanley (New York: Wiley 1999). Sampling of Populations: Methods and Applications.
- 2) Kalton, Graham (Beverly Hills: Sage 1983). Introduction to Survey Sampling.
- 3) Sandal, Carl-Eric, Lundstrom, Sixten (Chichester: Wiley 2005). Estimation in Surveys With Nonresponse.
- 4) Rubin, Donald B. (New York: Wiley 1987). Multiple Imputation for Nonresponse in Surveys.
- 5) Carpenter, James R. , Kenward, Michael G. (Wiley 2013). Multiple Imputation and its Application.
- 6) Sharon L. Lohr. (Advanced Series). Sampling: Design and Analysis, (Second Edition).
- 7) I. T. Jolliffe (Springer). Principal Component Analysis, (Second Edition).
- 8) Julie Josse, Jerome Pages, Francois Husson (Springer-Verlag 2011). Multiple Imputation in Principal Component Analysis.
- 9) Pier Alda Ferrari, Alessandro Barbiero, Giancarlo Manzi (Springer-Verlag Berlin Heidelberg 2011). Handling Missing Data in Presence of Categorical Variables: a New Imputation Procedure.
- 10) Julie Josse, Francois Husson (Societe Francaise de Statistique et Societe Mathematique de France 2012). Handling Missing Values in Exploratory Multivariate Data Analysis Methods.
- 11) Nadia Solaro, Alessandro Barbiero, Giancarlo Manzi and Pier Alda Ferrari (Department of Statistics, Milan, Italy). Algorithmic Imputation Techniques for Missing Data: Performance Comparisons and Development Perspectives.
- 12) Vincent Audigier, Francois Husson, Julie Josse (2013). A Principal Components Method to Impute Missing Values for Mixed Data.
- 13) Cristian Preda, Gilbert Saporta, Mohamed Hedi Ben Hadj Mbarek (AMS 2000). The Nipals Algorithm for Missing Functional Data.
- 14) Alessandro Barbiero, Pier Alda Ferrari, Giancarlo Manzi (2015). Imputation of Missing Values Through a Forward Imputation, Package ‘ForImp’.
- 15) Francois Husson, Julie Josse (2015). Handling Missing Values with Multivariate Data Analysis, Package ‘missMDA’.
- 16) Daniel J. Stekhoven (2013). Nonparametric Missing Value Imputation using Random Forest, Package ‘missForest’.
- 17) Gaston Sanchez. Principal Components with NIPALS, Package ‘plsdepot’.
- 18) Wolfram Stacklies, Henning Redestig (Max Plank Institute for Molecular Plant Physiology Potsdam, Germany and GAS-MPG Partner Institute for Computational Biology (PICB) Shanghai, P.R. China 2015). The ‘pcaMethods’ Package.
- 19) Leo Breiman (Berkeley 1999). Random Forests.
- 20) Pier Alda Ferrari, Paola Annoni, Alessandro Barbiero, Giancarlo Manzi (Elsevier B.V. 2011). An Imputation Method for Categorical Variables with Application to Nonlinear Principal Component Analysis.



- 21) Linting M. , Meulman J. J. , Groenen P. J. F. , Van der Kooij A. J. (American Psychological Association Analysis). Non Linear Principal Component Analysis: Introduction and Application.
- 22) Adelchi Azzalini (2015). The Skew-Normal and Skew-t Distributions. The Package ‘sn’.
- 23) Barbara Lepidus Carlson, Stephen Williams (Mathematica Policy Research, Inc. , Princeton, New Jersey 2001). A Comparison of Two Methods to Adjust Weights for Non Response: Propensity Modeling and Weighting Class Adjustments.
- 24) Pier Alda Ferrari, Paola Annoni, Sergio Urbisci ( Statistica and Applicazione, vol. IV, 2006). A Proposal for Setting up Vulnerability Indicators in the Presence of Missing Data.
- 25) Gabriele B. Durrant (University of Southampton 2005). Imputation Methods for Handling Item Nonresponse in the Social Sciences: A Methodological Review.
- 26) Bjorn Grung, Rolf Manne (University of Bergen 1998). Missing Values in Principal Component Analysis.
- 27) Abdelmounaim Kerkri, Zoubir Zarrouk, Jelloul Allal (Faculte des Science). A Comparison of NIPALS Algorithm with Two Other Missing Data Treatment Methods in a Principal Component Analysis.
- 28) Alexander Ilin, Tpani Raiko (Journal of Machine Learning, Aalto University of Science and Technology 2010). Practical Approaches to Principal Component Analysis in the Presence of Missing Values.
- 29) Henning Risvik (2007). Principal Component Analysis and NIPALS Algorithm.
- 30) Frank Critchley, Ana Pires, Conceicao Amado. Principal Axis Analysis.
- 31) Lindsay I. Smith (2002). A Tutorial on Principal Component Analysis.
- 32) Kirk Baker (2005). Singular Value Decomposition Tutorial.
- 33) Καρακώστας Γεώργιος, Εισαγωγή στην Πολυμεταβλητή Ανάλυση, Πανεπιστήμιο Ιωαννίνων, 1993.



