THE EFFECTIVENESS OF

MISSING DATA TECHNIQUES

IN PRINCIPAL COMPONENT ANALYSIS

Huibrecht Elizabeth Maartens

Supervisor: Elsabé Smit

School of Statistics and Actuarial Science
University of the Witwatersrand

A dissertation submitted to the Faculty of Science, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Master of Science.

Johannesburg, 2015

DECLARATION

I declare that this research report is my own, unaided work. It is being submitted for the Degree of Master of Science by dissertation to the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination in any other University.

4 November 2015

ABSTRACT

Exploratory data analysis (EDA) methods such as Principal Component Analysis (PCA) play an important role in statistical analysis. The analysis assumes that a complete dataset is observed. If the underlying data contains missing observations, the analysis cannot be completed immediately as a method to handle these missing observations must first be implemented. Missing data are a problem in any area of research, but researchers tend to ignore the problem, even though the missing observations can lead to incorrect conclusions and results. Many methods exist in the statistical literature for handling missing data. There are many methods in the context of PCA with missing data, but few studies have focused on a comparison of these methods in order to determine the most effective method. In this study the effectiveness of the Expectation Maximisation (EM) algorithm and the iterative PCA (iPCA) algorithm are assessed and compared against the well-known yet flawed methods of case-wise deletion (CW) and mean imputation. Two techniques for the application of the multiple imputation (MI) method of Markov Chain Monte Carlo (MCMC) with the EM algorithm in a PCA context are suggested and their effectiveness is evaluated compared to the other methods. The analysis is based on a simulated dataset and the effectiveness of the methods analysed using the sum of squared deviations (SSD) and the R_v coefficient, a measure of similarity between two datasets. The results show that the MI technique applying PCA in the calculation of the final imputed values and the iPCA algorithm are the most effective techniques, compared to the other techniques in the analysis.

Dedicated with all my heart to my Lord and Saviour in Whom I trust always.
May all the glory and praise be unto Your Name.

ACKNOWLEDGEMENTS

First and foremost I would like to thank my Lord and Saviour for making all that I do possible and for constantly reminding me that miracles still happen. Then I would like to thank my family, friends and colleagues that God blessed me with for always being there for me. A special word of thanks goes to my parents, Lilly, the Van Zyl family and the Fourie family who literally carried me with their love, prayers, support and guidance. To my supervisor, Elsabé Smit, thanks for all the effort, guidance, time and support you have given me over all these years. Finally, Prof. Jacky Galpin thanks for your time and knowledge with the final editing and review.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY OF ACRONYMS

CW	Case-wise deletion
EDA	Exploratory Data Analysis
EM	Expectation Maximisation
iPCA	Iterative Principal Component Analysis
MAR	Missing At Random
MCAR	Missing Completely At Random
MCMC	Markov Chain Monte Carlo
MI	Multiple Imputation
ML	Maximum Likelihood
MNAR	Missing Not At Random
NIPALS	Non-linear Iterative Partial Least Squares
PC	Principal Component
PCA	Principal Component Analysis
SI	Single Imputation
SSD	Sum of Squared Deviations
SVD	Singular Value Decomposition

1. INTRODUCTION

1.1 Background

Many researchers and analysts use statistical analyses to answer research questions and make important decisions based on the results from the analyses. Behrens (1997, p.132) states that the role of a data analyst is: "*to listen to the data in as many ways as possible until a plausible 'story' of the data is apparent*". According to Jeffers (1994) it is important to start with an exploratory data analysis (EDA) as a first step in any statistical analysis. One EDA technique used frequently is principal component analysis (PCA). Wold, Esbensen and Geladi (1987, p.46) state that PCA is: "*recommended as an initial step of any multivariate analysis to obtain a first look at the structure of the data, to help identify outliers [and] delineate classes*". PCA as an EDA technique has many attractive features such as the ease of reducing a high dimensional dataset into a lower dimensional dataset (Chen, 2002). This is done in order to simplify the structure of a dataset with many variables to analyse, making it difficult to extract all the information given in the variables (Bro and Smilde, 2014). Focusing only on the simplified structure of the data can reveal relationships between variables that were overlooked when the full dataset was observed (Johnson and Wichern, 1998). Another advantage of PCA is that it requires no distributional assumptions regarding the data (Chen, 2002).

However, these advantages are only useful if the dataset is complete, without any missing observations. The presence of missing data causes a loss in efficiency when estimating parameters since there are fewer observations to analyse (Rubin, 1987). Furthermore, missing data may cause biased results since observed data are often systematically different from unobserved data (Barnard and Meng, 1999). The major problem of missing data in the context of PCA, however, is that PCA as a standard complete-data method cannot be immediately used to evaluate the results as the method was not designed to take missing observations into consideration (Schafer and Graham, 2002), and it is unclear how to handle missing values in the case of PCA (Chen, 2002). The output of a PCA model is also required to be complete as PCA is usually applied as an intermediate tool before performing other statistical analyses such as regression analysis or a cluster analysis (Johnson and Wichern, 1998). Despite this, Schlomer,

Bauman and Card (2010) state that many researchers are still unaware of the importance of correctly dealing with missing data.

A number of different approaches exist for dealing with missing data, and the most widely used methods, according to Schlomer *et al* (2010), include case-wise deletion (CW), single imputation (SI) and multiple imputation (MI). CW discards all observations with at least one missing value across the variables and as such is easy to implement but also causes a loss of observations (Knol *et al*, 2010). Mean imputation is an example of a SI method which replaces the missing observations with a single value, the mean of the observed values in this case. Known SI methods used explicitly in a PCA framework include the Expectation Maximisation (EM) algorithm (Dempster, Laird and Rubin, 1977) and the iterative PCA (iPCA) algorithm (Josse and Husson, 2012a). The drawback of SI methods, however, is that these methods are known to underestimate the variance since the single imputed value is assumed to be the only possible value of the missing observation (Little and Rubin, 2002). A well-known method used to overcome this shortcoming is MI, developed by Rubin (1987). This method replaces the missing value with a set of plausible values so as to account for the uncertainty of what the missing value should be, resulting in multiple datasets. Rubin (1987) provides formulae for combining the multiple datasets into a single dataset, but this is only defined for the calculation of means, variances and regression parameters. The practical application of these formulae in the context of PCA is not mentioned.

1.2 Problem Statement

The objective of this study is to investigate how the missing data problem can be addressed in the context of PCA by considering five imputation algorithms, namely CW, mean imputation, the EM algorithm, iPCA and MI. The EM algorithm is an effective method known to have changed the way statisticians handle missing data (Schafer and Olsen, 1998). iPCA is a more recently proposed algorithm, with the effectiveness compared to other methods still to be determined (Josse and Husson, 2012a). MI will be explored as an alternative method that has not yet been applied in PCA, and different ways to combine the sets of output are investigated. CW

and mean imputation are included only as benchmarks to compare the other methods against since the drawbacks of these methods are well documented (Schafer and Graham, 2002).

In order to assess the effectiveness of the algorithms, a simulated dataset is used as the baseline dataset and different scenarios of missing data (with the percentages of missingness varying from 10% to 40%) are introduced. The values are then imputed by applying the five aforementioned imputation algorithms, and PCA is performed on the baseline data, as well as the imputed datasets. From the different PCA models, the cumulative proportion of variation explained, the eigenvalues and the resultant output principal components (PC) are calculated. The results are then evaluated by looking at the descriptive statistics and the sum of squared deviations (SSD) of the baseline PC's compared to the imputed PC's. The sets of PC's are also compared using the R_V coefficient (Escoufier, 1973), a measure of similarity between two datasets such that a value of 1 indicates complete similarity and 0 indicates complete dissimilarity.

The specific research objectives addressed in this study are:

1. Is there an effective way to combine the PCA results from MI?
2. How efficient are the EM algorithm, iPCA and MI methods for handling missing data in PCA compared to CW and mean imputation?
3. Which of the methods under consideration are the most effective? This is evaluated by assessing which of the methods result in the lowest SSD and a R_V coefficient closest to 1?

1.3 Structure of the report

The report is structured as follows: Chapter 2 presents a review of the literature concerning PCA, missing data, imputation algorithms and the application of imputation algorithms in PCA. Chapter 3 outlines the methodology for the data simulation, the imputation methods and the evaluation of the effectiveness of the imputation algorithms. The results of these analyses are presented and discussed in Chapter 4, with conclusions and recommendations in Chapter 5.

2. LITERATURE REVIEW

This chapter reviews some of the published literature on the concept of PCA, missing data and how to deal with missing data especially in the context of PCA. Section 2.1 outlines the history, derivation and applications of PCA as an EDA technique with the more recent studies described in Section 2.2. Section 2.3 provides a description of the causes of missing data related to Rubin's missing data mechanisms (Rubin, 1987). An overview of the most common methods for handling missing data as well as some of the problems associated with these methods are given in Section 2.4 with specific focus on methods used specifically for PCA in Section 2.5. Finally, a synthesis of the literature review is presented in Section 2.6.

2.1. History of Principal Component Analysis

PCA is defined as the calculation of linear combinations of the variables from a dataset, termed principal components (PC's), such that further analyses can be performed on the PC's instead of the original variables (Jolliffe, 2002). Through the use of PCA, a dataset with many variables that are highly correlated with each other, can be represented by a fewer number of variables that retain the same information as the original variables (Bro and Smilde, 2014).

The total number of PC's that can be constructed is equal to the number of variables in the dataset, but the main use of PCA is data reduction, in which only a selected number of PC's are retained for further analyses (Jolliffe, 2002). Shlens (2014) refers to data reduction as a challenge for experimenters to extract the most meaningful information from the data, so as to reveal any hidden data structures, without filtering out the noise as well. Dimension reduction techniques, such as PCA, are used to compare individuals from a multidimensional viewpoint, detect the relationship between variables and use the variables to describe the individuals (Josse and Husson, 2012a). According to Wold *et al* (1987) other goals of PCA include: simplification, modelling, outlier detection, variable selection and prediction.

Over the years, numerous research articles on PCA have been published, with the earliest research deriving the technique of singular value decomposition (SVD), which was done

independently by both Beltrami (1873) and Jordan (1874) as noted by Jolliffe (2002). The idea of SVD is that any arbitrary matrix can be decomposed into three matrices consisting of two orthogonal matrices and one diagonal matrix (Shlens, 2014). In the context of PCA, these orthogonal matrices refer to the PC's and the eigenvectors whereas the diagonal matrix contains the singular values, also termed the eigenvalues (Shlens, 2014). The eigenvectors are defined as the matrix indicating the weights of the linear combinations in the PC's (Jolliffe, 2002) and the eigenvalues are a vector indicating the amount of variation each PC explains multiplied by the number of variables in the dataset (Wold *et al*, 1987). The research on SVD later developed into one of the techniques used to calculate the PC's from a dataset (Shlens, 2014). Jolliffe (2002) notes, however, that the importance of SVD not only lies in the calculation of the PC's, but also in being an aid to understanding what PCA does, providing geometric and algebraic ways of presenting the results.

The derivation of the SVD represents the start of the journey with the actual technique of PCA being introduced by Pearson (1901) through means of a geometric explanation followed by an algebraic derivation given several years later by Hotelling (1933). Pearson's analysis focuses on finding the optimal fit of lines and planes to a set of points in a *p*-dimensional space while Hotelling's analysis focuses on finding the optimal linear combinations of the original *p* variables, termed PC's, that maximise the contribution to the variances of the *p* variables. Hotelling was the first to introduce the term "principal component" such that it is not confused by the term "factor" used in mathematics (Jolliffe, 2002). Hotelling (1933) also defines the principal axes property which states that the first calculated PC will explain most of the variation of the original variables and the variation then decreases from the second PC onwards (Bro and Smilde, 2014). Being able to quantify the contribution that each dimension of the dataset adds to the variability of the dataset is stated by Shlens (2014) to be the most important benefit of PCA.

In order to determine the PC's, Hotelling's (1933) research proceeds to define the power method as a method to determine the PC's with a faster version of the method given in Hotelling (1936). The power method consists of calculating the first PC by determining the largest eigenvalue of a

covariance or correlation matrix and the resultant eigenvector. Morrison (1976) presents an adjustment to the power method such that either the first few PC's or the last few PC's can be calculated and also provides some worked out examples. Although this is a simple method for finding the first few PC's, the method becomes problematic if the eigenvalues for the PC's are very close to each other and becomes less accurate the more PC's have to be calculated (Jolliffe, 2002).

If all the PC's have to be calculated, Wilkinson (1965) recommends the QL-algorithm which is based on the idea that any non-singular matrix can be written as the product of an orthogonal matrix Q and a lower triangular matrix L. By iteration, the non-singular matrix will then converge to a diagonal matrix with the diagonal consisting of the eigenvalues in decreasing order (Jolliffe, 2002). The eigenvectors are then calculated by adding the transformations in the QL-algorithm as is done by Smith *et al* (1976). Jolliffe (2002) notes that, similar to the power method, convergence of the QL-algorithm will depend on the distance between consecutive eigenvalues.

As mentioned before, SVD also provides a method for calculating the PC's and is recommended by Chambers (1977) as well as Gnanadesikan (1977). Mandel (1982) states that SVD is especially beneficial if the PCA is followed by a regression analysis since the SVD presents the PC scores as an output that would otherwise have to be calculated before applying the regression analysis. Other methods for calculating the PC's that are documented by Jolliffe (2002) consist of the EM algorithm that is used especially in the presence of missing data (Tipping and Bishop, 1999) and neural networks that are used for datasets that require regular updates such that the PC's have to be re-calculated every time (Diamantaras and Kung, 1996). Neural networks provide a variety of different algorithms depending on a number of factors including whether the first few PC's or last few PC's are calculated and the number of PC's to be calculated (Jolliffe, 2002).

An important question to answer at this stage is that of the number of PC's that should be extracted in order to produce an effective PCA model. The selection of the number of PC's,

denoted by *S*, to use in PCA is a very important part of the analysis, as it would have an influence not only on the model but also on the analyses performed such as outlier detection (Bro and Smilde, 2014). Audigier, Husson and Josse (2013) note that if *S* is too small, relevant information is lost whereas if *S* is too large, unwanted noise is introduced into the model, which could influence the model results. Many different methods for selecting *S* exist in literature but the methods do not always produce definite results of what *S* should be (Bro and Smilde, 2014). Due to the many uses of PCA and the variety of different underlying data structures, there is no single method for selecting *S* that is preferred above the other (Josse and Husson, 2012a) and sometimes even a combination of the different methods are used in order to determine the most effective *S* to use (Bro and Smilde, 2014).

According to Jolliffe (2002), the cumulative percentage of variation explained by the PC's is considered the most obvious criterion to determine *S*, where the percentage of variation explained is calculated as the sum of variations that the individual PC's explain, divided by the number of variables. The criterion states that, for a selected threshold of the percentage of variation explained, *S* is selected to be the smallest number of PC's for which the accumulated percentage variation explained is greater or equal to the threshold. The threshold usually ranges between 70% and 90% (Jolliffe, 2002).

A visual method for selecting *S* is a scree plot test, which consists of a plot of the eigenvalues on the Y-axis against the number of PC's on the X-axis. *S* is then selected to be the value at which the graph forms an "elbow" such that the curve is steep to the left of *S* and forms an almost horizontal line to the right of *S* (Jolliffe, 2002). The method was developed by Cattel (1966) and is based on the assumption that the relevant information given by the data is more than the random noise and as such the size of the variation of the random noise tends to smooth out linearly as *S* increases (Bro and Smilde, 2014). Jolliffe (2002, p.116) notes that the name of the plot originated from having a similar shape as "*the accumulation of loose rubble, or scree, at the foot of a mountain slope*". If the eigenvalues are too large to distinguish an "elbow", an alternative is to plot the logarithm of the eigenvalues instead of the actual eigenvalues (Farmer,

1971). It is, however, still difficult to determine *S* if the eigenvalues decrease gradually (Jolliffe, 2002).

An additional method that also depends on the eigenvalues is Kaiser's rule (Kaiser, 1960), that selects *S* as the number of PC's that have eigenvalues greater than one. The rule is based on the argument that every PC would have an eigenvalue of one if all the variables are orthogonal. As such an eigenvalue greater than one indicates that the PC explains the variation of more than one variable (Bro and Smilde, 2014). In practice, however, it is sometimes possible for PC's to have eigenvalues below one and still make a significant contribution to the model (Bro and Smilde, 2014).

The aforementioned criteria are all ad-hoc methods that are very subjective, rather than based on statistical principles (Jollife, 2002). Often there exists a requirement for *S* to be determined more accurately than simply based on subjective criteria (Bro and Smilde, 2014). For example, when a PCA model is applied to detect outliers, the model will produce different results for different number of components and as such it is important to select the correct number of components. A popular method to use is cross-validation. This was initially developed by Mosier (1951) but introduced in the context of PCA by Wold (1978). The idea of cross-validation methods is to leave out part of the data, build a PCA model on the data that are left and then apply the model to predict the left-out observations. The SSD between the actual and predicted observations is then calculated and *S* is selected to be the number of PC's in the model with the lowest SSD (Josse and Husson, 2012b).

2.2. Recent Studies

Apart from the few literature references such as Girshick (1939), who focused on a study of the asymptotic sampling distribution of the coefficients and variances of the PC's, it is only from the 1960's that the literature in PCA expanded. Jolliffe (2002) comments that this expansion coincides with the advances in computer technology and quotes four important references that have contributed to the advancement in PCA, namely Anderson (1963), Rao (1964), Gower (1966) and Jeffers (1967). Anderson (1963) adds to the work done by Girshick (1939) by

studying the asymptotic sampling distribution of the coefficients and variance of the sample PC's. Rao (1964) introduces many alternative uses, interpretations and extensions of PCA, including the use of PCA in cluster determination and significance testing. Rao (1964) also provides a clear distinction between PCA and factor analysis. Gower (1966) links the use of PCA with other statistical techniques such as principal coordinate analysis.

Jeffers (1967) offers a more practical study of PCA with the aim to provide the practical objectives that can be achieved with PCA as well as a clearer understanding of the interpretation underlying the outcome of PCA. In Jeffers' (1967) study PCA is applied as more than a reduction tool in two case studies. For the first case study, the aim of the analysis is to determine whether pit props cut from Corsican pine in East Anglia are strong enough to be used in the mines. PCA is applied to a dataset consisting of 13 highly correlated variables and 180 observations, in order to reduce the dataset to 5 significant PC's. The PC's are then applied in a regression analysis, known as principal component regression, to determine the strength of the pit props. In the second case study, the analysis focuses on determining the number of distinct taxa within a sample of 40 winged aphids, with 19 variables measuring different characteristics of the insects. Based on the results of PCA, the dataset is reduced to 2 significant PC's which, when plotted, indicate 4 different groups of insects.

Jolliffe (2002) provides an in-depth summary of the history, the derivations and the uses of PCA. Abdi and Williams (2010) claim that PCA can most likely be seen as the most popular multivariate statistical technique that is used across many different statistical fields. They also commend PCA on its versatile nature mentioning the use of PCA in neural network models, correspondence analysis, multiple factor analysis and many more. PCA can therefore be seen as a very important and widely-used methodology. To give an illustration of the versatile uses of PCA, three recent studies each applying PCA in a different context will now be discussed.

Ndiaye and Gabriel (2011) focus on principal component regression in order to determine the electricity consumption of housing units in Oshawa, Canada, which help local electricity distribution companies to better develop conservation and demand management projects. The

data consist of 59 highly correlated variables and 62 observations, gathered from energy audits, phone surveys and smart meter readings. Applying PCA results in 9 PC's extracted and a linear regression model is then applied to the PC's with the annual electricity consumption per square foot of floor area as the dependent variable. The model yields an R^2 value of 79% indicating the good fit. All the parameter estimates are significant at the 5% level of significance.

In a different study by Tomazzoni *et al* (2014), PCA is used in a classification analysis to aid in diesel quality control. Fuel alterations are sometimes made by adding low value-added products in order to make illegal financial profits and the authors suggest the use of a fluorescence spectrophotometer in gas stations or fuel distributors, due to its practicality, cost effectiveness and reliability. The fluorescence spectrophotometer, however, only produces effective results in differentiating between pure samples containing no mixture and as such the application of PCA to the results from the fluorescence spectrophotometer is recommended in order to differentiate the pure samples from the mixtures. Their analysis is based on three different samples of data, each consisting of the 1136 variables measured by the fluorescence spectrophotometer and 17 different mixtures of diesel, oil and biodiesel including a pure diesel mixture. PCA is then applied to the different datasets and the results indicate the successful classification according to the type of product and concentration of oil or biodiesel added to it.

The third study includes outlier detection with the aid of PCA done by Bro and Smilde (2014). Their data consist of 44 different samples of red wine (Cabernet Sauvignon), produced from the same grape variety, distributed as follows: 6 from Argentina, 15 from Chile, 12 from Australia and 11 from South Africa. Fourteen different characteristics, such as the ethanol content and the density, are then measured for each wine using the Foss WineScan instrument. PCA is then applied for different types of analyses including outlier detection. The authors identify outliers as unusual observations not similar to the majority of the observations and identify these outliers by visual inspection with the aid of score plots, score contribution plots and influence plots. The results indicated one outlier from South Africa with very high levels of volatile and lactic acids.

2.3. Missing Data

One of the requirements, before PCA can be applied, is that the dataset must be complete, since PCA is a multivariate statistical technique based on all the observations in the dataset and thus cannot be directly applied if missing values are present (Rubin and Schenker, 1991). The presence of missing data, however, is a well-known problem across all areas of research (Horton and Lipsitz, 2001).

Groves *et al* (2009) define two main types of missing data, namely unit non-response and item non-response. Unit non-response follows when all the information for a unit or object is missing. For example, when a respondent cannot be contacted to complete a survey due to incorrect contact information, it is classified as unit non-response. Item non-response is when only some information for the unit is missing, such as a half completed survey due to the respondent's lack of knowledge on the subject of the survey. The two types are analysed differently as both of them have different reasons for the nonresponse and are different problems with different solutions (Wagner and Kemmerling, 2010). Durrant (2005) notes that the general methods for handling unit non-response are weighting methods, while the methods for item non-response include weighting methods, imputation methods and maximum likelihood based methods such as the EM algorithm. Since imputation and maximum likelihood are the topics of this study, the focus will be on item non-response only.

Item non-response is caused by what Rubin (1987) mathematically defines as the three missing data mechanisms:

- Missing Completely At Random (MCAR)
- Missing At Random (MAR)
- Missing Not At Random (MNAR)

2.3.1. Missing Completely At Random

Missing observations are said to be MCAR when the missing observation is completely random with respect to the variable being measured as well as the other variables (Rubin, 1987). The missing values occur completely by chance and not because of a specific reason. For example, in survey research, the interviewer may accidentally turn over two pages instead of just one and thus cause a whole page of missing observations. The advantage of missing observations being MCAR is that almost any method for handling these missing observations will produce effective results (Bennett, 2001). Although MCAR does occur, Stuart, Azur, Frangakis and Leaf (2009) note that missing observations are generally more likely to be MAR than MCAR.

2.3.2. Missing At Random

Missing observations are MAR when the missing observation does not depend on the variable being measured but depends on another variable (Rubin, 1987). In this case there is a specific reason why an observation is missing, but the reason is unrelated to the variable that contains the missing value. For example, males tend to know more about cars than females. If a female is asked a technical question about the inner workings of an engine, she may leave that question unanswered. Here the reason for the non-response has nothing to do with the question asked but rather with the lack of knowledge of the female population regarding the particular subject area.

2.3.3. Missing Not At Random

Missing observations will be MNAR if the missing observation depends on the variable being measured (Rubin, 1987). For example, a person refuses to answer a question about personal income as he/she feels uncomfortable in disclosing such information. In this example, the non-response is explicitly because of the question asked. According to Nakagawa and Freckleton (2010), MNAR is the mechanism with the most problems but no general method exists to appropriately deal with data that are MNAR (Donders, Van der Heijden, Stijnen and Moons, 2006).

MCAR and MAR are considered to be ignorable in the sense that the distribution of the missingness caused by either of these mechanisms does not need to be modelled, since the distribution has no effect on the analysis (Rubin, 1987). With MNAR, however, the missing data are influenced by the distribution of the missingness and must thus be modelled before the missing data can be handled. MNAR is therefore termed as non-ignorable.

The distinction between the underlying missing data mechanisms is important when dealing with missing data, since if the mechanism can be correctly identified then the researcher will be able to identify which strategy is the most appropriate to use (Nakagawa and Freckleton, 2010). In practice it is possible to determine whether missing data are MCAR or not MCAR by using either statistical tests (Little, 1988) or visual inspection (Nakagawa and Freckleton, 2010). The distinction between MAR and MNAR, however, is more difficult (Nakagawa and Freckleton, 2010) since MNAR requires more information regarding the distribution of the missing data, in order to be distinguished from MAR (Schafer, 1997).

Little (1988) proposes that each variable with missing values be split into the observed part and the missing part. A series of t-tests are then performed, comparing the differences in the means of the other variables within these two groups. If no significant difference exists, then the missing data can be classified as MCAR, otherwise it is classified as not MCAR. Nakagawa and Freckleton (2010) suggest a simpler method of distinction, by recoding the data matrix into binary variables to indicate whether the observation is missing or observed. It is then possible to visually assess whether the missing observations are MCAR or not MCAR from the data patterns of the bivariate plots between the binary variables and the variables on the original scale.

2.4. Handling Missing Data

2.4.1. Case-wise deletion

In this section commonly used techniques to handle missing data are discussed. The most commonly used method for handling missing data that is implemented in most software packages as the default procedure is CW, also known as complete-case analysis (Knol *et al*, 2010). CW discards all observations with at least one missing value and as such results in a dataset with fewer observations to analyse. A major concern with CW is the fact that information is lost in the process, which leads to a loss in the statistical power of the analysis (Little and Rubin, 2002). Also, if missing observations are not MCAR, CW can produce biased parameter estimates leading to incorrect conclusions for the analysis (Klebanoff and Cole, 2008). As such, Jolliffe (2002) notes that CW in the context of PCA is only acceptable if there are a few missing values, with no specific threshold value indicated.

2.4.2. Single Imputation

In order to avoid deletion of observations and hence reducing the number of observations, an alternative method is to impute or replace the missing observations with a certain value. This method is known as SI, and found its origin in survey research (Little and Rubin, 2002). SI is computationally easy to use and available in most standard software packages (Schlomer *et al*, 2010). Since important information is retained instead of deleted, SI is known to perform better than CW (Schafer and Graham, 2002). The advantage of SI in the context of PCA is that the output is a single dataset which can easily be used in further analyses. However, the substitution of a single value for the missing observation completely ignores the uncertainty of what that missing value should be, causing overconfident precision and biased parameter estimates (Little and Rubin, 2002, Donders *et al*, 2006). Three of the most widely used methods of SI include mean substitution, regression substitution and hot-deck imputation (Schlomer *et al*, 2010).

Mean Imputation

In mean imputation, the missing observations are replaced by the mean of the observed data. This method produces biased means if missing observations are not MCAR (Bennett, 2001). Researchers strongly advise against the use of mean imputation, as it underestimates the variance even in the case where missing observations are MCAR (Bennett, 2001; Schafer and Graham, 2002).

Regression Substitution

Regression substitution predicts the value of the missing observation using a regression analysis that models the (generally linear) relationship between the variable containing missing data (Y) and one or more other variables (**X**). Even though regression substitution produces unbiased means for MCAR and MAR (Schlomer *et al*, 2010), it still has the disadvantage of underestimating the variance (Bennett, 2001). The main concern with regression substitution, however, is that this method cannot be used if the covariance or correlation between the variables are analysed, as regression substitution is known to overemphasize the strength of the relationship between them (Schafer and Graham, 2002).

Hot-deck Imputation

Hot-deck imputation is a nonparametric method in which the values of the missing observations are predicted by using other 'similar' observations. For example, in survey research, a person that has the same response to the answered questions as another person, will tend to have the same response as that person for the questions he/she did not respond to (Rubin, 1987). Even though this method produces biased results when missing observations are MNAR, it produces less biased results than CW and mean substitution (Bennett, 2001). However, hot-deck imputation also underestimates the variance (Rubin, 1987).

2.4.3. Multiple Imputation

To overcome the problem of the variance being underestimated, Rubin (1987) developed the method of MI which is recommended above all other methods by researchers such as Klebanoff and Cole (2008) and Janssen *et al* (2010). Instead of replacing the missing observations with a single value, MI consists of replacing every missing observation by a vector of plausible values (Rubin, 1987). This vector of multiple values takes the uncertainty of what the missing value should be into consideration, and thus produces more accurate and efficient estimates of the variance (Little and Rubin, 2002).

The first step in the MI process is the imputation step, where missing values are imputed using an appropriate imputation method. This step is replicated *m* times in order to impute each missing observation with *m* plausible values. With modern computing power it is not much effort to make *m* as large as possible, but Rubin (1987) shows that usually *m* between 3 and 10 is sufficient. Collins, Schafer and Kam (2001) show that $m = 5$ produce results that are efficient enough. Parametric Bayesian models or nonparametric methods are typically used to impute missing values. After the imputation step, each of the *m* datasets is analysed using the normal complete-data methods in the analysis step. The final step in the process is termed the pooling step, where the *m* sets of estimates are combined using the formulae developed by Rubin (1987) in order to produce a single set of estimates that are easy to understand and interpret.

Parametric Bayesian Models

Little (2011) states that Bayesian models, in particular, can be used to handle missing data. The parametric Bayesian model (Rubin, 1987, Schafer, 1999) is based on using the observed data (X_{obs}) and the prior distribution of the vector of unknown parameters (θ) to compute the posterior distribution of θ using Bayesian theory. The missing observations (X_{mis}) are then imputed by simulating values from this posterior distribution. An example of this method is the Markov Chain Monte Carlo (MCMC) with the EM algorithm.

MCMC originated in the 1950s as a means for physicists to study molecules (Yuan, 2000), but the technique was not used in Bayesian inference until the 1980s (Tanner and Wong, 2010). The MCMC algorithm requires starting values as initial estimates. According to Yuan (2000) the parameters from the EM algorithm, a model first formulated by Dempster *et al* (1977), are good starting values to use for MCMC. MCMC and the EM algorithm are known for their flexibility and reliability when dealing with missing data (Schafer, 1997). However, Nakagawa and Freckleton (2010) state that MCMC is not widely used due to implementation difficulties and slow convergence speed.

Nonparametric Methods

In contrast to parametric Bayesian models, nonparametric methods require minimal information about the distribution of θ. These methods are known to work well for large samples (Schafer, 1999) as long as the between-imputation variance is taken into account (Rubin, 1987). One example of a nonparametric method is the Approximate Bayesian Bootstrap imputation method, where missing values are imputed by sampling with replacement, as discussed in Rubin (1987).

Many researchers have studied the effects of using MI for different percentages of missingness according to the three missing data mechanisms. Although the methods of MI used were different, they all reached similar conclusions regarding the effectiveness of MI.

Knol *et al* (2010) compare CW with MI for missing observations that are MCAR and MAR, according to five different percentages of missingness ranging from 2.5% to 30%. Five variables are selected from the Dutch part of a European prospective cohort study aimed at developing a multifactor risk algorithm for onset of major depression (n = 1075). CW produces unbiased results for all five percentages of MCAR. For MAR, the results are unbiased only if there is less than 5% missing data. MI, however, results in effective, unbiased estimates for both mechanisms and for all percentages and is thus identified as the method of choice.

Janssen *et al* (2010) use 500 simulated datasets from a cross-sectional study among 804 adult patients with suspicion of deep venous thrombosis, with percentages of missingness ranging from 10% to 90%. They show that MI performs better than CW when considering missing observations that are MAR. Donders *et al* (2006) compare MI with SI on 1000 simulated samples of 500 observations with approximately 40% MAR missingness and confirm that MI is better.

Marshall, Altman and Holder (2010) investigate the MAR mechanism by looking at five percentages of missingness ranging from 5% to 75% and applying the methods of CW, SI and MI. Using a dataset consisting of 7507 patients from a randomised colorectal cancer trial between May 1994 and September 2003, they obtain 500 replications of 1000 observations each by sampling with replacement from this dataset, according to the MAR mechanism. They conclude that for 5% of missingness there are very few differences between the three methods. For percentages of missingness greater than 10%, they advise against the use of both CW and SI. MI proves to be the most useful method for handling 10% to 50% MAR missingness. However, for percentages greater than 50% even MI leads to biased and misleading results.

Burns *et al* (2011) use MI to estimate the missing Mini-Mental State Examination results for 17303 participants drawn from the Dynamic Analyses to Optimise Aging Project. The data were collected between 1990 and 2006. For percentages of missingness ranging from 0.5% to 95% under the MAR mechanism, the researchers show that MI is only effective for up to 50% MAR .

MI, just as any other method, has its disadvantages. A possible limitation of the MCMC algorithm is that the algorithm assumes that data are from a multivariate normal distribution. However, Schafer (1997) produces evidence that valid inferences will be obtained even if the assumption of normality is violated. The EM algorithm, on the other hand, assumes that the missing observations are ignorable, i.e. either MCAR or MAR. Also, MI can become computationally intensive (Schlomer *et al*, 2010). However, the increasing number of articles on the method of MI, and the practical implementation thereof, have made this method more accessible (Knol *et al*, 2010).

After the missing observations are imputed using a suitable MI method, the intended analysis (such as regression analysis) is then performed on each imputed dataset and the results combined using formulae developed by Rubin (1987). A drawback of MI in the context of PCA is that the output consists of multiple datasets that should ideally be pooled or combined for use in possible subsequent analyses such as PCA or cluster analysis.

2.4.4. Maximum Likelihood Approaches

Except for MI, Josse and Husson (2012a) state that the other group of highly recommended methods for dealing with missing data are the maximum likelihood (ML) approaches. A ML based approach known for its effectiveness, under the assumption of multivariate normality, is the EM algorithm first introduced by Dempster *et al* (1977). The EM algorithm is a method that calculates the ML estimates of parameters in the case of missing observations. Based on the ML estimates, the missing observations are then imputed using the expected log-likelihood. Since the missing observations are imputed with a single value, the EM algorithm can also be seen as an SI method (Little and Rubin, 2002).

2.5. Algorithms Specifically for PCA

Considering missing data algorithms used specifically for PCA, Jollife (2002) notes that most research articles generally do not consider missing data in PCA but rather focus on the estimation of the covariance or correlation matrix in the presence of missing data. This allows for the derivation of PCA models, but the resultant PC's for individual respondents cannot be determined due to the observations still being missing. A range of methods handling missing data in the context of PCA have been studied in the past, each with their own advantages and disadvantages. Jolliffe (2002) and Ilin and Raiko (2010) provide a general overview of the missing data algorithms used historically in PCA, with Ilin and Raiko (2010) focusing more on Bayesian methods.

2.5.1. The Non-linear Iterative Partial Least Squares Algorithm

The problem of missing data in PCA was first considered by Dear (1959) whose analysis is based on the minimum mean-square error (MSE) formulation of PCA derived by Young (1941). Dear (1959) splits the data matrix into the observed values and the missing values. The PC and loadings from a one component PCA model based on the observed values are then used to estimate the missing values. As an equivalent to Dear's (1959) method, Wold (1966) develops the non-linear iterative partial least squares (NIPALS) algorithm, first defined by Fisher and Mackenzie (1923), for a single PC model. Starting with an initial value for the PC, say the variable with the highest variance, the NIPALS algorithm consists of estimating the eigenvector and normalising it to length one. The PC is then re-estimated given the normalised eigenvector and the process is repeated until a measure of convergence is reached, for example when the SSD between the elements of two consecutive PC's is 0 or below a certain threshold. Wold *et al* (1987, p.50) suggest that the iteration should be stopped if convergence is not reached within 25 iterations as the data will then be "*almost (hyper)spherical with no strongly preferred direction of maximum variance*".

Christofferson (1970) extends the NIPALS algorithm to handle missing values and mentions the application for two PC's. The extended algorithm to calculate more than one PC is given in Wold *et al* (1987) and is based on estimating the PC's on a sequential basis. The first PC is calculated using the NIPALS algorithm and the residuals are estimated by subtracting the single PCA model estimates from the dataset. The second PC is then calculated by applying the algorithm to the residuals instead of the original dataset and the process is then repeated for the following PC's. Josse, Husson and Pagès (2011) note that the NIPALS algorithm consists of alternating two weighted simple regression models, namely the regression of the eigenvector on the PC and the regression of the PC on the eigenvector. Wold *et al* (1987) comment that the NIPALS algorithm is a faster alternative for calculating PC's compared to the SVD method if only the first few PC's are calculated, but if all the PC's have to be calculated, SVD is more efficient.

Although the NIPALS algorithm is a commonly used algorithm for handling missing data in PCA (Josse *et al*, 2011), there are many literature studies that criticise the algorithm including Gabriel and Zamir (1979), Grung and Manne (1998) and Josse and Husson (2012a). The criticism includes that the algorithm produces different solutions for different software programs (Grung and Manne, 1998), the algorithm does not always converge to a solution especially if many missing observations are present (Josse and Husson, 2012a), and the algorithm only produces reasonable results for a few missing values (Josse *et al*, 2011).

2.5.2. Criss-cross Multiple Regression

Due to the shortcomings of the NIPALS algorithm, researchers such as Grung and Manne (1998) and Kroonenberg (2008) suggest the use of criss-cross multiple regression developed by Gabriel and Zamir (1979). The criss-cross multiple regression is an extension to the NIPALS algorithm in the sense that instead of using two weighted simple regression models, the method uses two weighted multiple regression models that are alternated until convergence (Josse *et al*, 2011). While the NIPALS algorithm proceeds dimension by dimension, the criss-cross multiple regression considers all the dimensions at once (Gabriel and Zamir, 1979).

2.5.3. The Iterative PCA Algorithm

As an alternative to criss-cross multiple regression, Kiers (1997) studies the method of iPCA. In the EDA framework, Josse *et al* (2011) note that iterative imputation was first proposed in correspondence analysis by Nora-Chouteau (1974) and Greenacre (1984). Their work is based on the missing data method designed by Healy and Westmacott (1956) to handle missing observations in block experiments. The method consists of proposing initial values for the missing observations and then running the analysis on the completed data matrix for a predetermined number of PC's. The missing observations are then imputed based on the output from the model and the model is re-estimated given the new completed data matrix. This process is repeated until the total change in the matrix is less than a certain threshold level. The iPCA algorithm is also considered to be an SI method (Josse and Husson, 2012a).

Kiers (1997) compares the iPCA algorithm to the criss-cross multiple regression method and concludes that both methods provide similar results. In Kiers' (1997) study, 30 datasets consisting of 20 variables each are simulated from a uniform distribution with parameters -0.2 and 0.8 with the number of observations selected as either 100 or 500 and the number of dimensions set to either 3 or 6. The weight matrix used for the criss-cross multiple regression method is simulated from a uniform distribution with parameters 0 and 1 such that the weights are all greater than 0. A PCA model is then fitted to the datasets using both the iPCA algorithm and the criss-cross multiple regression method and the final loss function value, i.e. the sum of the squared differences between the actual and predicted values, recorded. The results show that there are no significant differences between the loss function values of the two methods in 29 out of the 30 cases. For the last case, the loss function value for the iPCA algorithm is more than 1% lower than that of the criss-cross multiple regression method.

A study by Walczak and Massart (2001) applies the iPCA algorithm for different percentages of missingness ranging from 2% to 25% that are removed from the data based on the MCAR missing data mechanism. Their complete dataset consists of 10 variables and 45 observations. The number of PC's is set to 2 which accounts for 96.7% of the variation in the data. Score plots of the PC's, comparing the scores from the complete dataset to the scores from the imputed datasets, indicate that for low percentages of missingness (2% to 9%) the imputed scores are very similar to the complete scores. For high percentages of missingness ($\geq$ 22%) the differences between the imputed and complete scores are more evident but the data pattern is still preserved.

As sourced from Josse and Husson (2012a), Josse, Husson and Pagès (2009) present a regularised iPCA algorithm by introducing a shrinking parameter in the imputation step of iPCA algorithm. The shrinking parameter is used to reduce the effect of noise in the model and thus to reduce the problem of overfitting. In a simulation study they determine the effectiveness of the regularised iPCA algorithm by comparing it against the normal iPCA algorithm, the NIPALS algorithm, the EM algorithm and the mean imputation method. A dataset comprising of 7 normalised variables and 21 observations is simulated from a two dimensional signal model with a noise component added that is simulated from a normal distribution with a mean of 0 and a

variance of σ^2, σ^2 ranging from 0.1 to 0.75. Missing observations are introduced according to the MCAR mechanism with the missingness varying between 10% and 40%. The number of PC's to be used in the iPCA algorithms is selected *a priori* to be either 2 or 3 PC's.

In their study, Josse *et al* (2009) conclude that the regularised iPCA algorithm outperforms the other algorithms, especially for noisy data with a high percentage of missingness. When there is no noise in the data, the regularised iPCA algorithm is similar to the normal iPCA algorithm. If the incorrect number of PC's is selected, the regularised iPCA algorithm still presents acceptable results as opposed to the incorrect results from the EM algorithm. The NIPALS algorithm is shown to be very unstable for high percentages of missingness.

Josse *et al* (2011) provide an MI version of the iPCA algorithm to account for the variability due to the missing observations. In their study, they create multiple imputed datasets from the generated PCA model, but do not combine the results into a single dataset. Instead they project the multiple possibilities onto a reference configuration, in order to visually observe the uncertainty of the missing values. In other words, they plot the first two PC's from an SI iPCA algorithm and then overlay the multiple predicted observations from the MI iPCA algorithm onto these observations. The multiple values are then summarised using ellipses or convex hulls and the bigger, the ellipse the more variability is caused by the missing observations. Although the effectiveness of the methodology still needs to be tested (Josse and Husson, 2012a), the authors comment that the proposed MI iPCA algorithm can be considered as an alternative to other MI methods such as the MCMC algorithm by Schafer (1997).

Further recent developments for the iPCA algorithm include multi-level simultaneous component analysis (Timmerman, 2006), multiple factor analysis (Escofier and Pagès, 2008), multiple correspondence analysis (Josse and Husson, 2012a; Josse, Chavent, Liquet and Husson, 2012) and an algorithm to impute mixed data (Audigier *et al*, 2013).

2.5.4. The EM Algorithm

The EM algorithm (Dempster *et al*, 1977) is a widely used missing data algorithm that can also be used in the context of PCA (Jolliffe, 2002). Popov (2006) presents such an analysis in which the EM algorithm is compared to mean substitution. The analysis is based on a real-world biomedical dataset consisting of blood tests with 35 measured variables for 26 patients, before and after taking treatment. An incomplete dataset is constructed by deleting 20% of the observations using MCAR. A visual analysis, comparing the complete dataset with the imputed datasets, is then performed. The results indicate that the EM algorithm performs better than the mean substitution.

Schafer and Olsen (1998, p.546) note that the importance of the EM algorithm goes beyond the implementation of the algorithm as *"the ideas underlying EM signalled a fundamental shift in the way statisticians viewed missing data".* They comment that, before the EM algorithm was developed, missing observations were either deleted or imputed, whereas the EM algorithm now provided a method of averaging over the predictive probability distribution calculated from the observed values.

Since the development of the EM algorithm, there has been an increase in the number of missing data algorithms that make use of the EM algorithm. For example, in the robust estimation of the mean and covariance matrix (Little, 1988), the regularised EM algorithm is used specifically if the number of variables exceeds the number of observations (Schneider, 2001), and the robust PCA model developed for datasets containing missing observations as well as outliers (Serneels and Verdonck, 2008). Other examples include a parallel factor analysis model in the presence of missing data (Tomasi and Bro, 2005) as well as a robust parallel factor analysis model (Hubert, Van Kerckhoven and Verdonck, 2012). These are only a few of the adaptations of the EM algorithm in order to indicate the versatile application of the algorithm.

2.6. Synthesis of the Literature Review

PCA is a widely known EDA method used for multiple purposes including data reduction, outlier detection and variable selection. In many instances, PCA is used in an initial analysis, before applying other statistical analyses such as regression analysis or cluster analysis. Hence, a complete dataset is required as an outcome of the PCA in order to do the subsequent analyses. The presence of missing data, however, makes it impossible for PCA to be applied and since missing data are in most cases unavoidable, a suitable solution for handling these missing values needs to be found.

Many different methods exist to handle missing observations but only some of these methods are used specifically for PCA. The EM algorithm is known for its effectiveness but has the disadvantages of requiring the data to be multivariate normally distributed and underestimates the variance by imputing the missing observations with a single value. The more recently proposed iPCA algorithm is quite well researched by authors such as Josse *et al* (2009) and has the advantage of requiring no assumptions regarding the data. However, the effectiveness of the iPCA algorithm compared to other effective methods such as the EM algorithm still needs to be determined and the algorithm is still an SI method.

Although SI methods underestimate the variance, the methods produce a single, complete dataset that can be easily and directly applied in subsequent analyses such as cluster analysis. In order to overcome the underestimation of the variance, MI methods can be applied but the disadvantage is that the methods produce multiple datasets. It is, however, unclear which dataset should essentially be used or how to combine the datasets in order to have a single, complete dataset. The important research question thus becomes whether the variation in the imputed observations is of such importance to make it worth the extra effort of finding an effective method to combine the multiply imputed datasets from an MI method.

3. METHODOLOGY

This analysis consists of four parts, namely data simulation, handling of missing data, PCA and evaluation of the results. A number of different techniques are used to deal with the missing data, either through exclusion or through imputation. The SVD method is used to derive the eigenvalues and eigenvectors in the PCA. A detailed discussion of PCA is given in Section 3.1 and the imputation algorithms are outlined in Section 3.2. Section 3.3 describes the specific analytic approach followed in the analysis, with a summary of the expected results given in Section 3.4.

3.1. Principal Component Analysis

3.1.1. Complete Dataset

Consider a data matrix X with *n* observations and *p* variables. The goal of PCA is to determine the best linear transformation of the *p* variables into *S* new variables, $S \leq p$, such that the variance of the *p* variables is maximised (Ilin and Raiko, 2010). Mathematically, PCA consists of approximating X as the product of two smaller dimensional matrices V (*n* x *S*) and U^t (*S* x *p*) that contain the most relevant data patterns of X (Jolliffe, 2002). Using the method of least squares, this amounts to finding V and U that minimises the following criterion (Josse *et al*, 2011):

$$C = \|X - A - VU^t\|^2$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{p}\left(X_{ij} - a_j - \sum_{s=1}^{S} V_{is}U_{js}\right)^2 \quad (1)$$

Where $A = (a_1, a_2, \ldots, a_p)$ and a_j is a *n* x 1 vector with the mean of variable *j*, $j = 1,2,\ldots,p$.

Given the additional constraints that the columns of U are orthogonal and of unit form, SVD is used to obtain $\widehat{V}$ and $\widehat{U}$, where (Josse *et al*, 2011):

$\widehat{V}$ = The principal components (or score matrix) such that the variance of each column equals the corresponding eigenvalue.

$\widehat{U}$ = Eigenvectors (or loadings matrix) of the correlation matrix.

Singular Value Decomposition

Given the data matrix X with *n* observations and *p* variables, the following quantities are defined (Shlens, 2014):

- $\hat{u}_i^* = \{\hat{u}_1, \hat{u}_1, \dots, \hat{u}_p\}$ is the set of orthonormal *p* x 1 eigenvectors with corresponding eigenvalues $\{\lambda_1, \lambda_2, \dots \lambda_p\}$ for the symmetric matrix $X^T X$:

$$X^t X\, \hat{u}_i = \lambda_i\, \hat{u}_i$$

- $\sigma_i = \sqrt{\lambda_i}$ are the singular values
- $\hat{v}_i^* = \{\hat{v}_1, \hat{v}_1, \dots, \hat{v}_n\}$ is the set of *n* x 1 vectors such that:

$$\hat{v}_i = \frac{1}{\sigma_i} X\, \hat{u}_i$$

The singular values are then ordered from the largest value to the smallest value:

$$\sigma_{\tilde{1}} \geq \sigma_{\tilde{2}} \geq \cdots \geq \sigma_{\tilde{p}}$$

The diagonal matrix Σ is constructed by placing the ordered singular values on the diagonal from left to right. Similarly the matrices U and V are constructed:

$$U = \left[\hat{u}_{\tilde{1}}^*, \hat{u}_{\tilde{2}}^*, \dots, \hat{u}_{\tilde{p}}^*\right]$$

$$V = \left[\hat{v}_{\tilde{1}}^*, \hat{v}_{\tilde{2}}^*, \dots, \hat{v}_{\tilde{n}}^*\right]$$

Hence, SVD is the decomposition of the data matrix X into:

$$XU = V\Sigma$$

Since matrix U is orthogonal, the equation can be written as:

$$X = V\Sigma U^T$$

Assumptions

Even though PCA has the advantage of being non-parametric, Shlens (2014) notes the following three assumptions that must hold for PCA:

- PCA assumes linear combinations of the variables in calculating the PC's so as to restrict the number of possible transformations and thus simplifying the calculation slightly
- PC's with larger variances (and thus larger eigenvalues) are assumed to contain important information while PC's with smaller variances represent noise
- The assumption of orthogonality when calculating PC's based on SVD

3.1.2. Incomplete Datasets

When missing observations are present in the data matrix X, the criterion C in Equation 1 cannot be calculated directly (Josse *et al*, 2011). Given that some of the observations are missing, a weight matrix (W) is introduced such that $W_{ij} = 0$ if X_{ij} is missing and $W_{ij} = 1$ otherwise. The criterion to minimise is then expressed as:

$$C_{miss} = \sum_{i=1}^{n}\sum_{j=1}^{p} W_{ij}\left(X_{ij} - a_j - \sum_{s=1}^{S} V_{is}U_{js}\right)^2$$

Contrary to the complete dataset, there exists no explicit solution to minimise C_{miss} and as such an iterative algorithm is necessary (Josse and Husson, 2012a). One such algorithm is the iPCA algorithm (Section 3.2.2) that proceeds to determine the missing observations and the parameters of the PCA model simultaneously. Alternatively, imputation methods can be applied before calculating C as the imputation will result in a complete dataset.

3.2. Missing Data Algorithms

3.2.1. EM Algorithm

The EM algorithm is a method that calculates the maximum likelihood estimates of parameters when dealing with incomplete data and consists of two steps, namely the expectation step and the maximisation step (Dempster *et al*, 1977):

- *Expectation step:* Given a current estimate $\theta^{(k)}$ at iteration *k*, the expected values for X_{mis} are calculated by finding the expected complete-data log-likelihood ($Q(\theta|\theta^{(k)})$) if θ were $\theta^{(k)}$

$$Q\left(\theta\middle|\theta^{(k)}\right) = E[\ell(\theta|X)]$$
$$= \int \ell(\theta|X) f(X_{mis}|X_{obs}, \theta = \theta^{(k)}) dX_{mis}$$

- *Maximisation step:* $\theta^{(k+1)}$, the estimate of θ at iteration *k+1* is calculated by maximizing the complete-data log-likelihood

$$Q\left(\theta^{(k+1)}\middle|\theta^{(k)}\right) \geq Q\left(\theta\middle|\theta^{(k)}\right), \text{for all } \theta$$

Starting with an arbitrary value for θ, as suggested by Schafer (1997), and iterating between the expectation and maximisation steps until the parameter estimates converge, leads to the required parameter estimates (Little and Rubin, 2002). The imputed dataset will be the dataset based on the parameter estimates from the last maximisation step after convergence has been reached. The only assumption of the EM algorithm is that the data should be multivariate normally distributed.

3.2.2. iPCA Algorithm

The iPCA algorithm is an iterative algorithm for calculating the PCA model in the presence of missing data. The algorithm consists of an estimation step where the PCA parameters are estimated and the imputation step where the missing values are imputed given the PCA model:

- *Estimation step*: Given initial values for the missing observations, such as the mean of the variables (Josse and Husson, 2012a), and the dimension *S*, the parameter estimates of the PCA model, namely: $\hat{A}^k, \hat{V}^k, \hat{U}^k$ are calculated for iteration *k*
- *Imputation step*: The missing observations are imputed with the fitted values based on the estimated PCA model calculated as follows:

$$\hat{X}^k = \hat{A}^k + \hat{V}^k \hat{U}^{tk}$$

The new imputed dataset is constructed by:

$$X^k = W \times X + (1 - W) \times \hat{X}^k$$

The algorithm iterates between the estimation and imputation step until convergence is reached and the imputed dataset is the dataset from the last imputation step.

The iPCA algorithm can be seen as an EM algorithm and is often referred to as the EM-PCA algorithm (Josse *et al*, 2011). The expectation step of the EM algorithm corresponds to the imputation step of the iPCA algorithm by imputing the expectation of the missing observations given the observed observations and the estimated parameters at iteration *k*:

$$\hat{x}_{ij}^k = \sum_{s=1}^{S} \hat{a}_j^{k-1} + \hat{v}_{is}^k \hat{u}_{js}^k$$

Furthermore, maximising the complete likelihood in the maximisation step of the EM algorithm corresponds to estimating the PCA model on the imputed dataset in the estimation step of the iPCA algorithm. As opposed to the EM algorithm, the iPCA algorithm does not have any assumptions regarding the data.

3.2.3. Multiple Imputation

MI consists of three steps, namely the imputation step, the analysis step and the pooling step (Rubin, 1987). In the imputation step, the missing observations are imputed by applying a Bayesian MCMC with the EM algorithm. Bayesian imputation methods use the prior distribution of θ, $\pi(\theta)$, and the likelihood of X_{obs}, $L(\theta|X_{obs})$, to determine the posterior distribution of X_{obs}:

$$P(\theta|X_{obs}) \propto \pi(\theta) \times L(\theta|X_{obs})$$

Simulating from this posterior distribution consists of an imputation step and a posterior step (Schafer, 1997):

- *Imputation step:* Given a current value $\theta^{(k)}$ for θ at iteration k, values for X_{mis} are drawn from the conditional predictive distribution of X_{mis}

$$X_{mis}^{(k+1)} \sim P(X_{mis}|X_{obs}, \theta^{(k)})$$

- *Posterior step:* The values from the imputation step are used to draw values for θ from the complete-data posterior

$$\theta^{(k+1)} \sim P(\theta|X_{obs}, X_{mis}^{(k+1)})$$

According to Schafer (1999), the complete-data posterior is not always one of the standard distributions that are easy to apply and draw values from. As such, MCMC is used as a method for approximating draws from the posterior distribution. An initial estimate for θ, $\theta^{(0)}$, is obtained by means of the EM algorithm (Section 3.2.1). The MCMC algorithm then iterates through the imputation and posterior step to form a Markov Chain consisting of the estimates for θ and the imputed values for X_{mis}:

$$\left\{\left(\theta^{(k)}, X_{mis}^{(k)}\right) : k = 1,2, \ldots\right\}$$

The process is iterated until the Markov Chain converges, i.e. the difference between the values of two successive iterations is very small (Gilks, Richardson and Spiegelhalter, 1996). The missing observations are then imputed by the observations from the last imputation step.

The imputation step is replicated five times, in order to impute each missing observation with five plausible values, and each of the imputed datasets is then analysed using PCA. The results from the five PCA models are then combined according to two techniques. The first technique (denoted MI Ave) is based on Rubin's (1987) methodology of applying a simple average to the five estimates:

$$\hat{V}_{is} = \frac{1}{M}\sum_{m=1}^{M} \hat{V}_{is}^{m}$$

Where $\hat{V}_{is}^{m}$ is the $(i, s)^{\text{th}}$ observation of the estimated score matrix at imputation m, $m = 1,2,\ldots,M$ with $i = 1,2,\ldots,n$ and $s = 1,2,\ldots,S$.

The second technique (MI PCA) applies the idea of taking an average of the five estimates, but instead of a simple average, the technique calculates a weighted average with the weights (w_m) determined by a one component PCA model:

$$\hat{V}_{is} = \frac{1}{M}\sum_{m=1}^{M} w_m \hat{V}_{is}^{m}$$

Where $\hat{V}_{is}^{m}$ is the $(i, s)^{\text{th}}$ observation of the estimated score matrix at imputation m, $m = 1,2,\ldots,M$ with $i = 1,2,\ldots,n$ and $s = 1,2,\ldots,S$.

Since MI makes use of the EM algorithm to determine the initial values of the algorithm, MCMC with the EM algorithm also requires the assumption of multivariate normally distributed data.

3.3. Analytic Approach

3.3.1. Data Simulation

A baseline dataset is simulated using a multivariate random normal simulation function (*mvrnorm*) in the **MASS** library (Venables and Ripley, 2002) of the R software package (www.R-project.org). A multivariate normal distribution is selected in order to ensure that the EM algorithm's assumption of multivariate normality holds. The *mvrnorm* function uses the mean vector and covariance matrix as input. An arbitrary total of $n = 300$ observations across

p = 6 variables are simulated from a standard normal distribution, $f(X_1, X_{2, \ldots,} X_6)$, with a mean vector $\boldsymbol{\mu} = \underline{\mathbf{0}}$ and covariance matrix structured such that the 6 variables will yield two PC's when the PCA model is applied, as shown in Table 3.1. Typically a low correlation refers to a value lower than 0.35 in absolute terms and a high correlation to a value higher than 0.7 in absolute terms. Fixing the number of PC's to be extracted allows for a comparative assessment across the different PCA models that result from the imputation algorithms. Since the variables are standard normally distributed, it follows that the covariance matrix is also the correlation matrix.

Table 3.1: Structure of the correlation matrix for the baseline dataset

	$\mathbf{X_1}$	$\mathbf{X_2}$	$\mathbf{X_3}$	$\mathbf{X_4}$	$\mathbf{X_5}$	$\mathbf{X_6}$
$\mathbf{X_1}$	1	**High**	**High**	Low	Low	Low
$\mathbf{X_2}$	**High**	1	**High**	Low	Low	Low
$\mathbf{X_3}$	**High**	**High**	1	Low	Low	Low
$\mathbf{X_4}$	Low	Low	Low	1	**High**	**High**
$\mathbf{X_5}$	Low	Low	Low	**High**	1	**High**
$\mathbf{X_6}$	Low	Low	Low	**High**	**High**	1

A requirement for using the *mvrnorm* function is that the covariance matrix has to be positive definite. This can be checked using the *is.positive.definite* function in the **corpcor** library (Schaefer *et al*, 2011) of R, and changed, if necessary, using the *make.positive.definite* function, also in the **corpcor** library (Schaefer *et al*, 2011).

The baseline dataset is then subjected to the influence of the missing data mechanisms by deleting randomly selected observations based on three possible scenarios according to the definition of each of the three missing data mechanisms. Let *pmiss* = (0.1, 0.2, 0.3, 0.4) indicate the proportion of missing data that will be removed from the baseline dataset. The process of removing observations is then defined as follows:

Missing Completely At Random

- Assume X_i is the variable to be subjected to missing data
- Randomly select *pmiss*% observations from variable X_i
- Remove selected observations

Missing At Random

- Assume X_i is the variable to be subjected to missing data based on variable X_j
- Define Set A = $\{X_i: X_j > 0\}$ and Set B = $\{X_i: X_j \leq 0\}$
- Randomly select 80% of *pmiss*% observations from Set A
- Randomly select 20% of *pmiss*% observations from Set B
- Remove selected observations

Missing Not At Random

- Assume X_i is the variable to be subjected to missing data
- Define Set C = $\{X_i: X_i > 0\}$ and Set D = $\{X_i: X_i \leq 0\}$
- Randomly select 80% of *pmiss*% observations from Set C
- Randomly select 20% of *pmiss*% observations from Set D
- Remove selected observations

The cut-off value of zero used to define the subsets for MAR (A vs. B) and MNAR (C vs. D) ensures that there are enough observations (approximately 150) available for deletion. The reason for deleting only 80% of *pmiss*% observations for MAR and MNAR is because the missing observations do not always occur according to only one specific mechanism. Eighty percent is an arbitrary percentage selected to effectively represent the corresponding mechanism but also account for observations caused by another mechanism.

The three possible scenarios that are being considered in this study are defined in Table 3.2 and consist of a scenario applicable to each of the three missing data mechanisms. In all three scenarios it is assumed that the missing data are not only as a result from one specific missing data mechanism and hence the variables X_1 and X_4 are subjected to an arbitrary 15% MCAR missingness. It is also assumed that the missingness can only occur within the two defined PC's meaning that the MAR missingness for X_2 is based on X_3 and X_5 is based on X_6. The variables X_3 and X_6 are left complete with no observations missing.

Table 3.2: Missing data scenarios under consideration in this study

	X_1	X_2	X_3	X_4	X_5	X_6
Scenario 1: MCAR	15% MCAR	MCAR	-	15% MCAR	MCAR	-
Scenario 2: MAR	15% MCAR	MAR|X3	-	15% MCAR	MAR|X6	-
Scenario 3: MNAR	15% MCAR	MNAR	-	15% MCAR	MNAR	-

This process is repeated 1000 times for each each percentage of missingness such that a total of 4000 incomplete datasets are simulated per scenario.

3.3.2. Analysis

In the analysis, the missing values of each incomplete dataset are imputed using CW, mean imputation, the EM algorithm, the iPCA algorithm and MI imputation, applying MCMC with the EM algorithm. CW is applied in R using the function *na.omit* in the **stats** library (R Core Team, 2012). For mean imputation, the missing values for each variable are determined using the *is.na* function and imputed with the mean of the observed values for that variable using the *mean* function, with both functions found in the **base** library (R Core Team, 2012). The EM algorithm is located in the **norm** library (Schafer, 2010) and consists of R manipulating the data with the function *prelim.norm* as an input to the function *em.norm* and *imp.norm* that are used to calculate the ML estimates of the algorithm and impute the missing values. The **missMDA** library (Husson and Josse, 2013) contains the iPCA algorithm and is applied using the *imputePCA* function. Similarly to the EM algorithm, the MCMC algorithm for MI is found in the **norm** library (Schafer, 2010) and applies the EM algorithm as an intermediate step to obtain initial values that are then used in the *da.norm* and *imp.norm* functions that perform the MCMC imputation. Except for MI, all the algorithms produce a total of 4000 imputed datasets for each scenario. MI produces a set of 5 imputed datasets for each incomplete dataset and thus a total of 20000 imputed datasets for each scenario.

PCA is then performed on the baseline dataset and each of the imputed datasets using the *princomp* function in the **stats** library (R Core Team, 2012). For MI this results in 5 sets of results for each imputed dataset, which are then combined into a single set of results by applying

two different techniques. MI Ave applies the *mean* function in the **base** library (R Core Team, 2012) and MI PCA the *princomp* function in the **stats** library (R Core Team, 2012). This produces 4000 sets of results for each scenario and each missing data algorithm.

The next step is to compare the baseline results with the imputed results in order to determine the most effective imputation algorithm. According to Peres-Neto, Jackson and Somers (2003) there are two problems that need to be considered before the results of the PCA models can be compared, namely axis reflection and axis reordering. Axis reflection involves the situation where the signs of the eigenvectors are arbitrarily swapped around and results in a positive value being compared to a negative value, when the values should be similar. Axis reordering involves the change in the order of the eigenvectors such that the first PC in the one simulation is not comparable to the first PC in another simulation, but rather the second or the third PC. In order to overcome these problems, Peres-Neto, Jackson and Somers (2003) make use of the correlation matrices between two sets of eigenvectors in the sense that the sign will indicate whether the PC's are reflected or not and by multiplying the reflected eigenvectors with a factor of -1, the problem is solved. Also, the maximum absolute value of the correlations will indicate whether the PC's are reordered and the correct PC's can be compared with each other. Similar to the approach used by Peres-Neto, Jackson and Somers (2003), correlation matrices are applied in this study. However, instead of using the eigenvectors, the correlations are determined between the resultant PC's since the eigenvectors only consist of 6 observations and can thus cause the correlations to be distorted and lead to incorrect results.

3.3.3. Evaluation

In the comparison analysis of the baseline and imputed results, the results to be evaluated are threefold:

- *PCA model results*: Analysis of the cumulative proportion of variation explained by the PC's and the eigenvalues of the PCA model to determine the number of PC's extracted
- *PCA output results*: A variable level comparison regarding the descriptive statistics of the imputed PC's compared to the baseline PC's
- *Goodness-of-fit results:* A respondent level comparison in order to assess the overall effectiveness of the imputed PCA models compared to the baseline PCA model.

PCA Model

As mentioned in Section 2.1, an important parameter considered in the PCA model is the number of PC's, *S*, to extract. In this study, *S* is determined by the two criteria cumulative proportion of variation explained (Jolliffe, 2002) and Kaiser's rule (Kaiser, 1960).

The cumulative proportion of variation explained by the first *S* PC's is calculated as:

$$cumPVar_S = \frac{100}{p}\sum_{k=1}^{S} l_k$$

Where l_k is the variance of the k^{th} PC and the sum of the variances of the PC's equals the sum of the variances of the *p* variables in the data matrix X. *S* is then selected to be the smallest number for which $cumPVar_S \geq 80\%$ (Jolliffe, 2002). The proportion of variation accounted explained by each PC is calculated using the *princomp* function in the **stats** library (R Core Team, 2012).

Kaiser's rule considers the eigenvalues as calculated by the SVD composition explained in Section 3.1.1 and selects *S* as the number of PC's for which the eigenvalues are ≥ 1. The eigenvalues for each PC is calculated using the *eigen* function in the **base** library (R Core Team, 2012). Both the criteria are calculated for the baseline dataset as well as the imputed datasets. An effective imputation algorithm must result in the same value for *S* as the baseline dataset.

PCA Output

The effectiveness of the imputation algorithms on a variable level is analysed by comparing the descriptive statistics of the imputed PC's to the descriptive statistics of the baseline PC's using the sum of the squared deviation (SSD). The SSD for the *k*th PC is calculated as follows:

$$SSD_{d,k} = \sum_{i=1}^{n} \left(d_{ik}^{base} - d_{ik}^{imp}\right)^2$$

Where d^{base} is the descriptive statistic of the baseline PC, d^{imp} the descriptive statistic of the imputed PC and $k = 1,2,\ldots,S$.

The descriptive statistics under consideration include the median and the standard deviation. A lower SSD implies that the predicted descriptive statistic is closer to the actual descriptive statistic and thus results in a more effective imputation algorithm.

Goodness-of-fit

The effectiveness of the imputation models is also analysed on a respondent level by using the goodness-of-fit measures consisting of the R_V coefficient and the SSD. The R_V coefficient is a measure of similarity introduced by Escoufier (1973) to study the relationship between two sets of variables given that they have the same number of observations. Let $Vbase$ denote the baseline PC's and $Vimp$ the imputed PC's. The R_V coefficient is then defined as:

$$R_v = \frac{tr\{Vbase\ Vbase^T\ Vimp\ Vimp^T\}}{\sqrt{tr\{(Vbase\ Vbase^T)^2\} \times tr\{(Vimp\ Vimp^T)^2\}}}$$

Where $tr\{.\}$ denotes the sum of the diagonal elements of the matrix.

The values of the R_V coefficient ranges between 0 and 1 where 0 implies that all the variables of V is uncorrelated to the variables of V and 1 implies that the structures are similar (Josse, Husson and Pagès, 2008). In order to test the statistical significance of the R_V coefficient, Josse *et al* (2008) applies a Pearson type III distribution (or gamma distribution) which is implemented in the **FactoMineR** library (Husson, Josse, Le and Mazet, 2013) in R. Both the R_V coefficient and the p-value are calculated using the function *coeffRV*.

The SSD comparing the baseline PC's to the imputed PC's is as follows:

$$SSD = \sum_{i=1}^{n}\sum_{j=1}^{S}\left(Vbase_{ij} - Vimp_{ij}\right)^2$$

A more effective imputation algorithm will have a R_V coefficient that is significant at a significance level of 1% and closer to 1 than the other imputation algorithms. The algorithm should also have a lower SSD between the imputed PC's and the baseline PC's.

3.4. Expected Results

The comparison of the effectiveness of the EM algorithm, the iPCA algorithm and MI in the context of PCA has not been documented in the past. However, since research has proven the success of the three methods in their own right, even if not in the context of PCA, it is possible for all three methods to handle missing data in PCA effectively.

The effectiveness of the EM algorithm as a missing data algorithm not only in the context of PCA but in general as well is commended by many researchers. As such it is expected that the results will indicate the superiority of the EM algorithm compared to the other algorithms. The iPCA algorithm has only recently received attention in literature. Although MI is known to perform well as an imputation algorithm, there is no study that indicates the performance of MI in the context of PCA. It can therefore be expected that the iPCA algorithm and MI will perform better than CW and mean imputation as these algorithms are known for their drawbacks. However, it is unclear whether the performance will exceed that of the EM algorithm.

Many literature studies indicate that missing observations that are MCAR and MAR can be imputed effectively for as high as 50% missingness and as such, irrespective of the imputation algorithm, the imputation should be satisfactory for all percentages of MCAR and MAR missingness. It can be expected that the missing observations that are MNAR will only be imputed effectively for the low percentages of missingness, if at all.

4. ANALYSIS

Section 4.1 describes the simulation of the complete baseline dataset with application to a real-world example from Psychology and the baseline PCA. Section 4.2 provides the results from a single execution using 40% MAR missingness. This serves as an illustration of how the simulation, imputation algorithms and evaluation of results are performed. The overall simulation with the corresponding results for the different percentages of missingness and the different missing data scenarios are given in Section 4.3. Section 4.4 completes this section with a summary of all the results.

4.1. Baseline Data Simulation

The baseline data is simulated based on a practical application in Psychology regarding extrinsic and intrinsic motivation (Lepper, Corpus and Iyenger, 2005). The research involves identifying factors that have an influence on the motivation of school learners to do their schoolwork. These factors can either be externally related factors such as motivation from the teacher or the parents (extrinsic motivation), or internally related factors where the learner motivates himself or herself to work (intrinsic motivation). An example of these factors taken from Lepper *et al* (2005) is given in Table 4.1 as well as the possible survey question underlying the corresponding factor.

Table 4.1: Practical application in Psychology research of simulated baseline dataset

Motivation	Factors	Possible questions related to factors
Extrinsic	Easy work	Do you like school subjects where it's easy to just learn the answers?
	Pleasing teacher	Do you do your schoolwork because your teacher tells you to?
	Dependence on teacher	When you make a mistake, do you like to ask the teacher how to get the right answer?
Intrinsic	Challenge	Do you like difficult problems because you enjoy trying to figure them out?
	Curiosity	Do you work really hard because you like to learn new things?
	Independent mastery	When you make a mistake, do you like to figure out the right answer by yourself?

Since it is known that extrinsic and intrinsic motivation are weakly, negatively correlated (Lepper *et al*, 2005), the data are simulated using the mean vector $\boldsymbol{\mu}$ and correlation structure as discussed in Section 3.3.1 such that the bounds for the correlation structure are defined as follows:

- Low: [-0.35; -0.15]
- High: [0.7; 0.8]

The resultant sample mean vector and sample correlation matrix are given in Table 4.2 and Table 4.3 with the colours in Table 4.3 indicating the high correlations within the two defined factors. The required distribution of extrinsic versus intrinsic motivation is thus reflected in the sample as is evident from the scatter-plot matrix in Figure 4.1. From the figure it follows that there exists a strong linear relationship between the variables within the same motivational subset and a weak negatively linear relationship between the variables from different subsets.

Table 4.2: Means of baseline dataset

X_1	X_2	X_3	X_4	X_5	X_6
0.049	0.048	0.016	0.070	0.037	0.092

Table 4.3: Covariance matrix of baseline dataset

	X_1	X_2	X_3	X_4	X_5	X_6
X_1	**1**	**0.826**	**0.742**	-0.329	-0.285	-0.260
X_2	**0.826**	**1**	**0.708**	-0.209	-0.337	-0.179
X_3	**0.742**	**0.708**	**1**	-0.335	-0.304	-0.170
X_4	-0.329	-0.209	-0.335	**1**	**0.784**	**0.813**
X_5	-0.285	-0.337	-0.304	**0.784**	**1**	**0.788**
X_6	-0.260	-0.179	-0.170	**0.813**	**0.788**	**1**

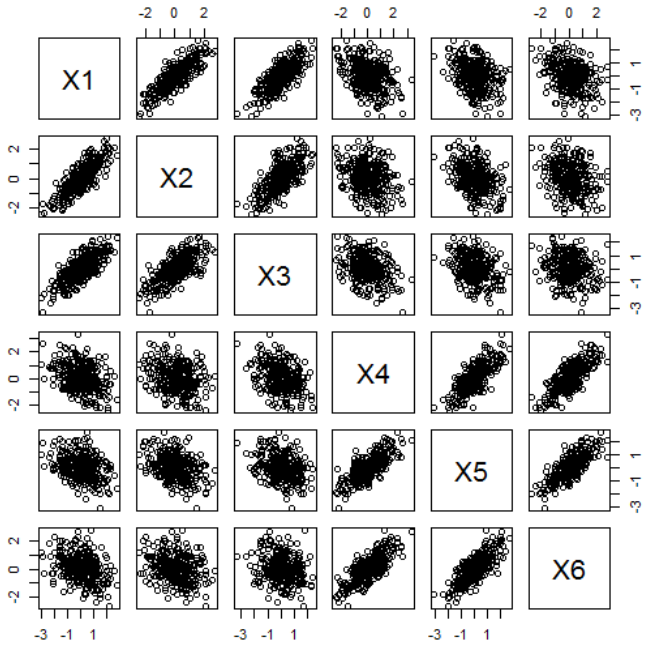

Figure 4.1: Scatter-plot matrix of baseline dataset

In order to test the assumption of separating the baseline dataset into two PC's, Table 4.4 presents the results from the selection criteria of the cumulative proportion of variation and Kaiser's rule. The values highlighted in red indicate that the assumption holds and a PCA model with two PC's can be fitted to the dataset.

Table 4.4: Selection criteria of the number of PC's for the baseline dataset

	PC1	**PC2**	**PC3**	**PC4**	**PC5**	**PC6**
Cumulative proportion of variation	57.32%	**84.63%**	90.70%	95.36%	98.24%	100%
Eigenvalues	3.44	**1.64**	0.36	0.28	0.17	0.11

The eigenvectors of the estimated PCA model for the baseline data are given in Table 4.5. The table shows that the first PC is a differentiation between the extrinsic and intrinsic motivational factors whereas the second PC is a weighted average of the different factors. Table 4.6 presents the rotated loadings matrix of the baseline PCA model as a visual illustration of the factors underlying the two PC's.

Table 4.5: Eigenvectors of the estimated PCA model for the baseline dataset

	PC1	PC2
$\mathbf{X_1}$	0.4258	-0.3967
$\mathbf{X_2}$	0.4027	-0.4220
$\mathbf{X_3}$	0.3907	-0.4084
$\mathbf{X_4}$	-0.4116	-0.3991
$\mathbf{X_5}$	-0.4223	-0.3744
$\mathbf{X_6}$	-0.3952	-0.4453

Table 4.6: Rotated loadings matrix for the baseline PCA model

	PC1	PC2
$\mathbf{X_1}$		-0.581
$\mathbf{X_2}$		-0.583
$\mathbf{X_3}$		-0.565
$\mathbf{X_4}$	-0.573	
$\mathbf{X_5}$	-0.564	
$\mathbf{X_6}$	-0.594	

The estimated median and standard deviation of the resulting baseline PC's are given in Table 4.7. These values will be used in the comparison analysis between the imputed datasets and the baseline dataset.

Table 4.7: Descriptive statistics for the baseline PC's

	PC1	PC2
Median	-0.0375	0.0268
Standard Deviation	1.8576	1.2823

4.2. Single Simulation Analysis

As discussed in Section 2.3, missingness within the survey can occur according to three different missing data mechanisms. The differences between the three mechanisms are illustrated in Figure 4.2, Figure 4.3 and Figure 4.4 for 40% missingness. The observed values and the missing values are given in the same scatter-plot with the missing values being the coloured blocks and the observed values the empty diamonds.

Figure 4.2 presents a random scatter in the missing values and as such shows the case of MCAR missingness. The straight lines in Figure 4.3 and Figure 4.4 indicate the cut-off value for being MAR and MNAR, respectively. With MAR (Figure 4.3) it can be observed that majority of the missing values in X_2 have high values for X_3, but are randomly scattered with respect to X_1. Hence, the missing values in X_2 depend on X_3. Considering MNAR (Figure 4.4), the majority of the missing values are in both instances observed to have high values for X_2 and thus depend on the variable with the missingness (X_2).

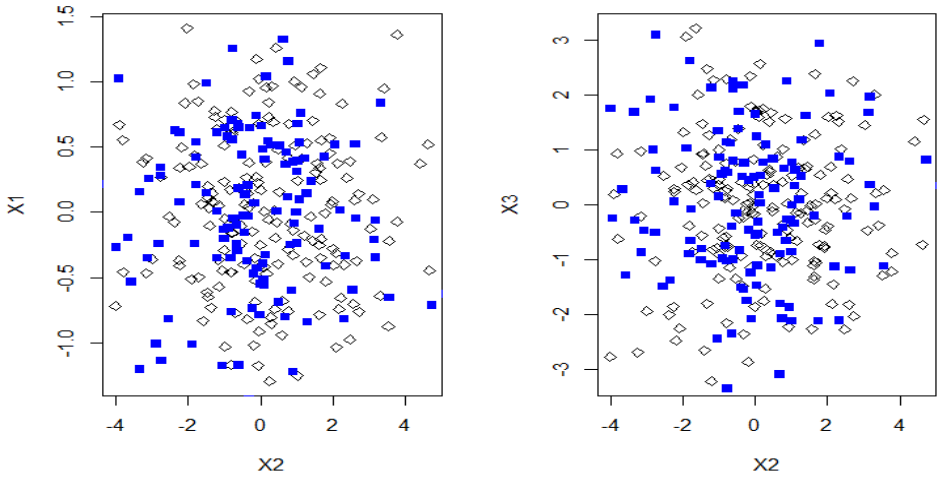

Figure 4.2: Scatter-plots for MCAR

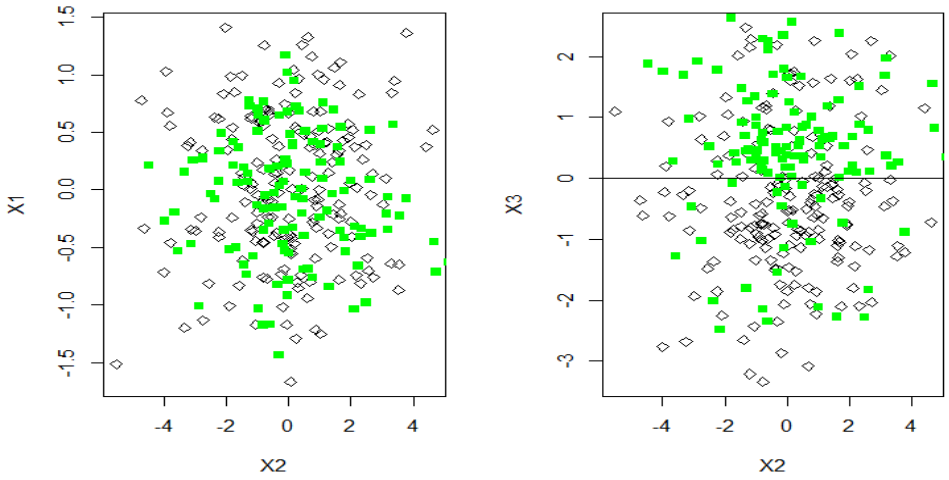

Figure 4.3: Scatter-plots for MAR

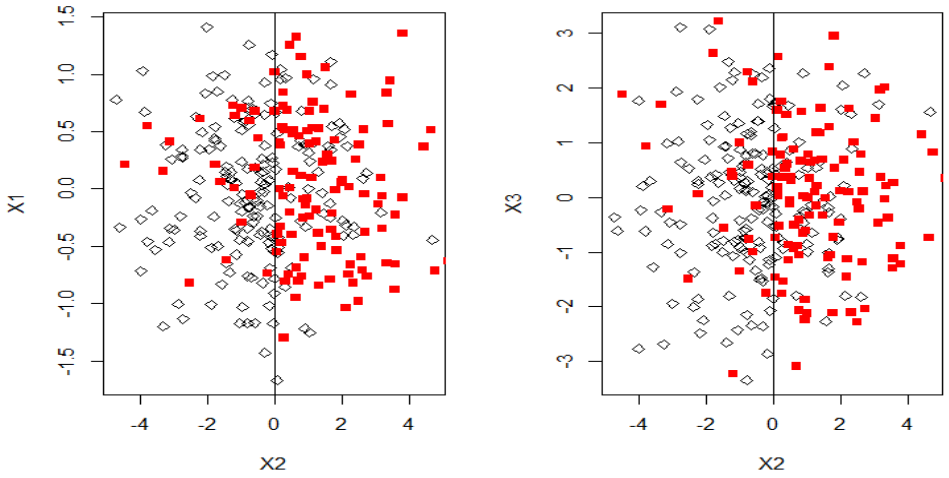

Figure 4.4: Scatter-plots for MCAR

Given the survey in Psychology, it is possible that the school learners might leave some questions unanswered as a result of the teacher being in the room while the survey is conducted. Hence, the single simulation analysis is performed on an assumption of 40% MAR missingness. The incomplete data are then imputed by the different imputation algorithms and a PCA model applied to each of the imputed datasets. The results from the imputed datasets are then compared to the results from the baseline dataset.

4.2.1. PCA Model Results

The first step is to compare the number of PC's selected from the imputed datasets based on the cumulative proportion of variation (Table 4.8) and Kaiser's rule (Table 4.9), with the results highlighted in red in the tables. According to Table 4.8, the only imputed dataset that incorrectly identified the number of PC's to select based on cumulative proportion of variation was the Mean imputed dataset. However, according to Kaiser's rule in Table 4.9, the mean imputation selected the correct number of PC's. Hence, the cumulative proportion of variation may be more influenced by the missing data than the eigenvalues.

Table 4.8: Cumulative proportion of variation for the single simulation analysis

	PC1	PC2	PC3	PC4	PC5	PC6
Baseline	57.32%	**84.63%**	90.70%	95.36%	98.24%	100%
CW	56.97%	**83.30%**	89.44%	94.67%	98.14%	100%
Mean	49.13%	73.21%	**82.41%**	90.44%	96.28%	100%
EM	58.59%	**84.86%**	91.06%	95.33%	98.13%	100%
iPCA	61.62%	**88.97%**	93.46%	96.09%	98.52%	100%
MI	57.96%	**85.11%**	90.90%	95.34%	98.27%	100%

Table 4.9: Eigenvalues for the single simulation analysis

	PC1	PC2	PC3	PC4	PC5	PC6
Baseline	3.44	**1.64**	0.36	0.28	0.17	0.11
CW	3.42	**1.58**	0.37	0.31	0.21	0.11
Mean	2.95	**1.44**	0.55	0.48	0.35	0.22
EM	3.52	**1.58**	0.37	0.26	0.17	0.11
iPCA	3.70	**1.64**	0.27	0.16	0.15	0.09
MI	3.48	**1.63**	0.35	0.27	0.18	0.10

4.2.2. PCA Output Results

The next step is to compare the resultant PC's from the imputed PCA models to the baseline PC's. However, as mentioned in Section 3.3.2 the effect of axis reflection and axis rotation should first be considered by looking at the correlation matrices between the respective PC's. This calculation is demonstrated in Table 4.10 that shows the correlations between the resultant PC's from an EM imputed dataset and the baseline PC's, with the maximum absolute correlation for each column highlighted in red. The maximum correlations thus indicate that in this specific scenario there were no PC's that swapped around. Also, based on the sign of the maximum correlations, axis reflection is observed for PC3 with a correlation of -0.85. Hence, the values for imputed PC3 are multiplied with -1 before being compared to the baseline PC's. The same process is followed when combining the 5 MI imputed results.

Table 4.10: Correlation matrix between EM imputed PC's and baseline PC's

		EM Imputed					
		PC1	**PC2**	**PC3**	**PC4**	**PC5**	**PC6**
Baseline	**PC1**	**0.98**	-0.02	-0.03	0.01	-0.05	-0.01
	PC2	0.03	**0.96**	0.01	0.03	0.01	-0.04
	PC3	0.04	-0.09	**-0.85**	-0.03	0.02	-0.01
	PC4	0.03	0.07	0.09	**0.72**	-0.11	-0.06
	PC5	-0.02	0.01	-0.06	0.17	**0.89**	0.03
	PC6	-0.01	-0.05	0.09	0.09	0.03	**0.49**

By looking at the median and standard deviation of the PC's (the minimum, mean and maximum are excluded from the analysis as it showed similar results to the median), Table 4.11 presents the squared deviation between the baseline and imputed statistics, with the lowest error highlighted in green and the highest in red. From the table it is observed that for the medians, the EM algorithm is the only algorithm that produced high errors relative to the other algorithms with CW having the lowest error for PC1 and MI PCA for PC2. Considering the standard deviation, MI PCA had very high errors compared to the other algorithms and MI Ave and iPCA had the lowest errors for PC1 and PC2, respectively. Based on the results of the descriptive

statistics for the single simulation, it is thus evident that MI PCA is not an effective imputation algorithm, compared to the other algorithms that have errors close to 0.

Table 4.11: Descriptive statistics for single simulation analysis

	Median		Standard Deviation	
	PC1	**PC2**	**PC1**	**PC2**
CW	**0.0003**	0.0000	0.0000	0.0003
Mean	0.0010	0.0003	0.0190	0.0061
EM	**0.0138**	**0.0048**	0.0004	0.0006
iPCA	0.0029	0.0002	0.0047	**0.0000**
MI Ave	0.0015	0.0001	**0.0000**	0.0005
MI PCA	0.0029	**0.0000**	**0.1337**	**0.8570**

4.2.3. Goodness-of-fit Results

The goodness-of-fit of the imputed datasets are calculated on the respondent level output by looking at the R_V coefficient (Figure 4.5) and the SSD (Figure 4.6). Since CW results in a dataset of different size compared to the baseline, it is excluded from the goodness-of-fit analysis. From Figure 4.5 it follows that MI PCA performs worse than the mean imputation with the lowest R_V coefficient of 0.91. iPCA and MI Ave are the most effective imputation algorithms with the EM algorithm slightly below them. Each of the R_V coefficients was statistically significant at a 1% significance level.

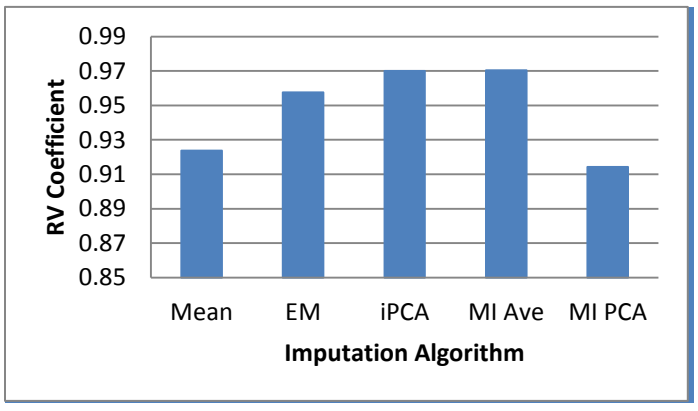

Figure 4.5: R_V coefficient for the single simulation analysis

Similar conclusions follow from the SSD given in Figure 4.6. MI Ave performs slightly better than iPCA. The figure also shows the big difference between the SSD from MI PCA compared to the other algorithms.

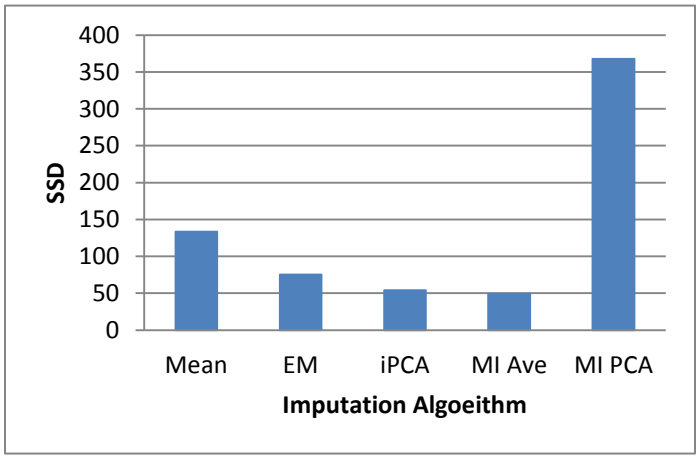

Figure 4.6: SSD deviation for the single simulation analysis

Although the study is only focused on the first two PC's, it is interesting to note from Table 4.12 that the goodness-of-fit measures decrease if all six PC's are added to the model since the extra PC's add more noise to the data. MI PCA is the most impacted by the extra noise in the data with the SSD almost ten times more than the model with only two PC's.

Table 4.12: Goodness-of-fit for the single simulation analysis based on all six PC's

	Mean	**EM**	**iPCA**	**MI Ave**	**MI PCA**
R_V Coefficient	0.832	0.954	0.968	0.968	0.668
SSD	790.72	226.24	488.91	139.02	3484.13

4.3. Overall Simulation Analysis

The single simulation analysis in Section 4.2 provides an indication of the effectiveness of the different imputation algorithms. However, the conclusions are made based on a single execution and a single percentage of missingness. This can often be misleading, as these results may occur by chance or specifically for that percentage. The analysis is therefore extended to four different percentages of missingness and three different scenarios of missing data with 1000 incomplete datasets created for each percentage and each scenario.

4.3.1. Scenario 1: Missing Completely At Random

PCA Model Results

The PCA model results for MCAR are given in Table 4.13 and indicate the error rates as a percentage of the 1000 simulations that selected the incorrect number of PC's to be extracted for the different algorithms at the different percentages of missingness. CW and mean imputation are the only algorithms mentioned in the table as all the other algorithms had a 0% error rate across the simulations. Based on the cumulative proportion of variation, there is an increase in the error rate for mean imputation as the percentage of missingness increases such that at 30% and 40% missingness every simulation identifies the incorrect number of PC's. CW only had a 0.5% error rate at 40% missingness. In each of these instances the results identified that three PC's must be selected as opposed to the two PC's from the baseline model. Applying Kaiser's rule decreased most of the error rates such that only 0.1% of the simulations from CW at 40% missingness stated that one PC should be extracted from the data.

Table 4.13: PCA model results for MCAR

		10%	20%	30%	40%
Cumulative proportion of variation	**CW**	0%	0%	0%	0.5%
	Mean	39%	97%	100%	100%
Eigenvalues	**CW**	0%	0%	0%	0.1%
	Mean	0%	0%	0%	0%

PCA Output Results

The output of the resulting PCA models for the different algorithms is shown in Figure 4.7 to Figure 4.10 for the median and standard deviation of both of the PC's. The output is presented as boxplots indicating the range of the statistics across the simulations for the different algorithms as well as the different percentages of missingness, with the red dot indicating the observed statistic from the baseline PC. Ideally, the statistics from the imputed datasets must be as close to the baseline as possible. This, however, is not the case for CW considering the median of PC1 (in Figure 4.7) since most of the simulations overestimated the baseline median of PC1 (-0.0375) with a wide range across all the percentages of missingness. As the percentage of missingness increase, there is a significant increase in the length of the boxplot for CW. The other algorithms indicate that up until 30% missingness most of the algorithms predict the baseline median quite accurately. At 30% and 40% missingness there is also a slight underestimation for both iPCA and MI Ave.

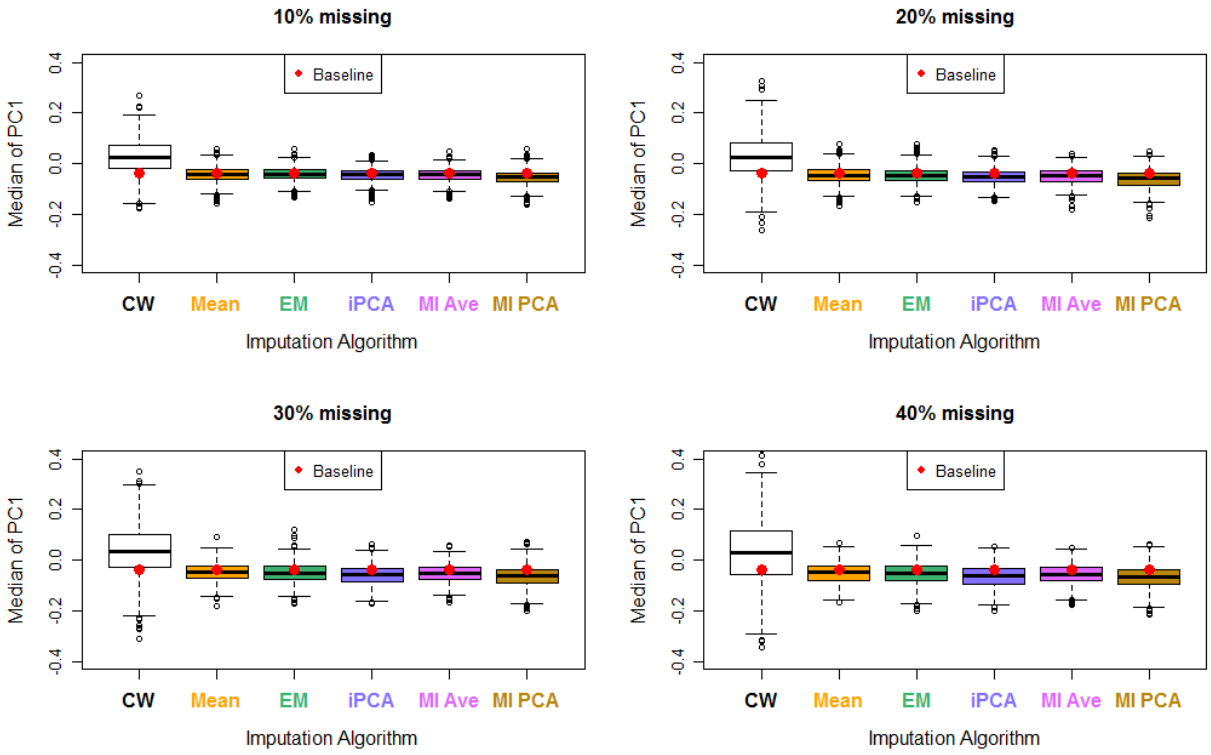

Figure 4.7: PCA output results of the median of PC1 for MCAR

Figure 4.8 presents the results for the median of PC2 and indicates that even though the range of CW increase as the percentage of missingness increase, the baseline median for PC2 (0.0268) is still close to the median of the distributions. At 10% missingness MI PCA slightly overestimates the baseline median and from 30% missingness all of the algorithms slightly underestimates the baseline median.

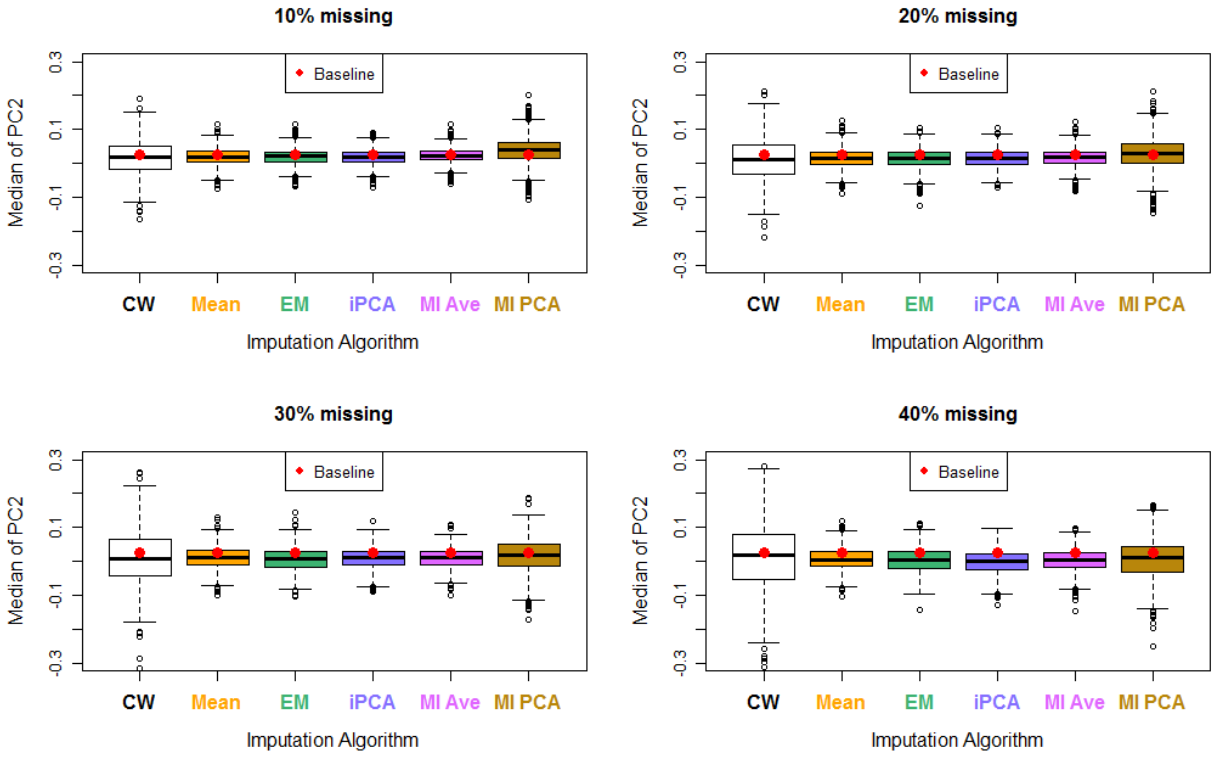

Figure 4.8: PCA output results of the median of PC2 for MCAR

From Figure 4.9 it follows that the EM algorithm and MI Ave estimate the baseline standard deviation for PC1 (1.8576) quite accurately. The range of CW increases only slightly and a gradual increase in the underestimation of the baseline standard deviation is observed for mean imputation. MI PCA completely overestimates the baseline standard deviation even at 10% missingness. The mean imputation underestimates the baseline standard deviation for all percentages of missingness.

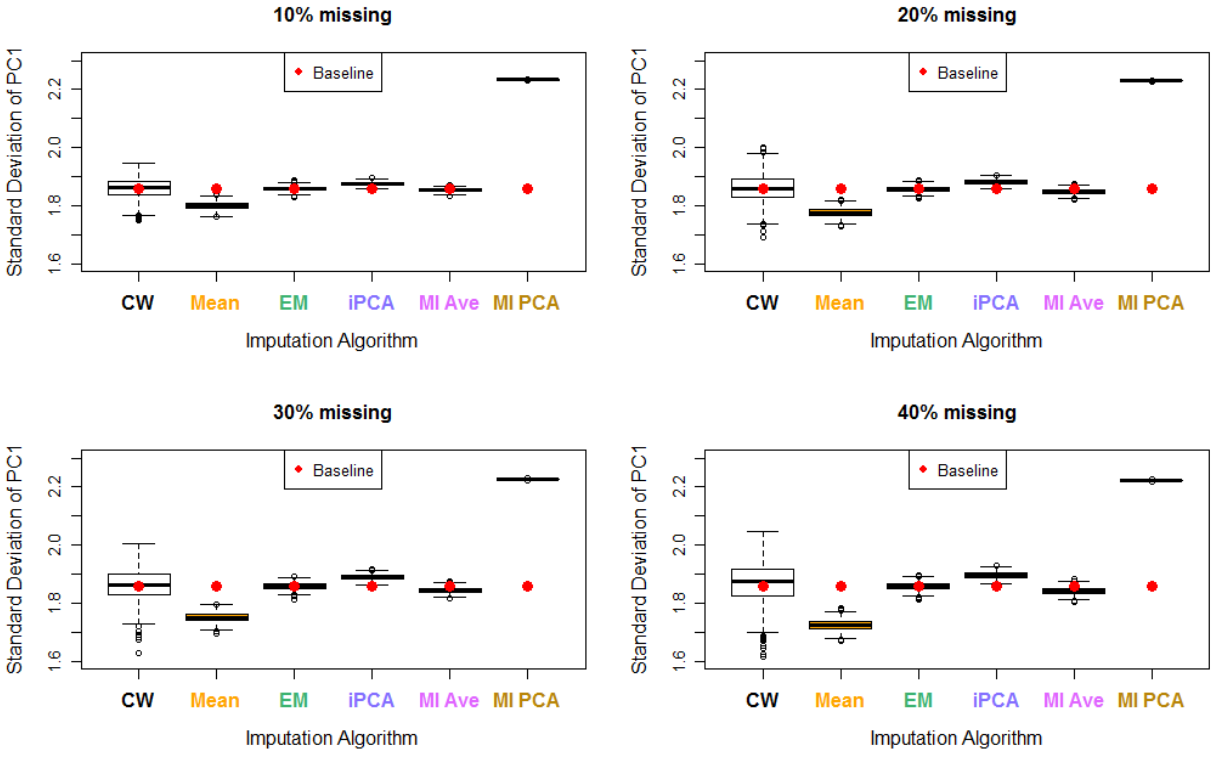

Figure 4.9: PCA output results of the standard deviation of PC1 for MCAR

Looking at the standard deviation of PC2 (Figure 4.10), MI PCA again completely overestimates the baseline standard deviation (1.2823) for all the percentages of missingness. The other algorithms present accurate results with slight deviations from 20% missingness for mean imputation and iPCA especially.

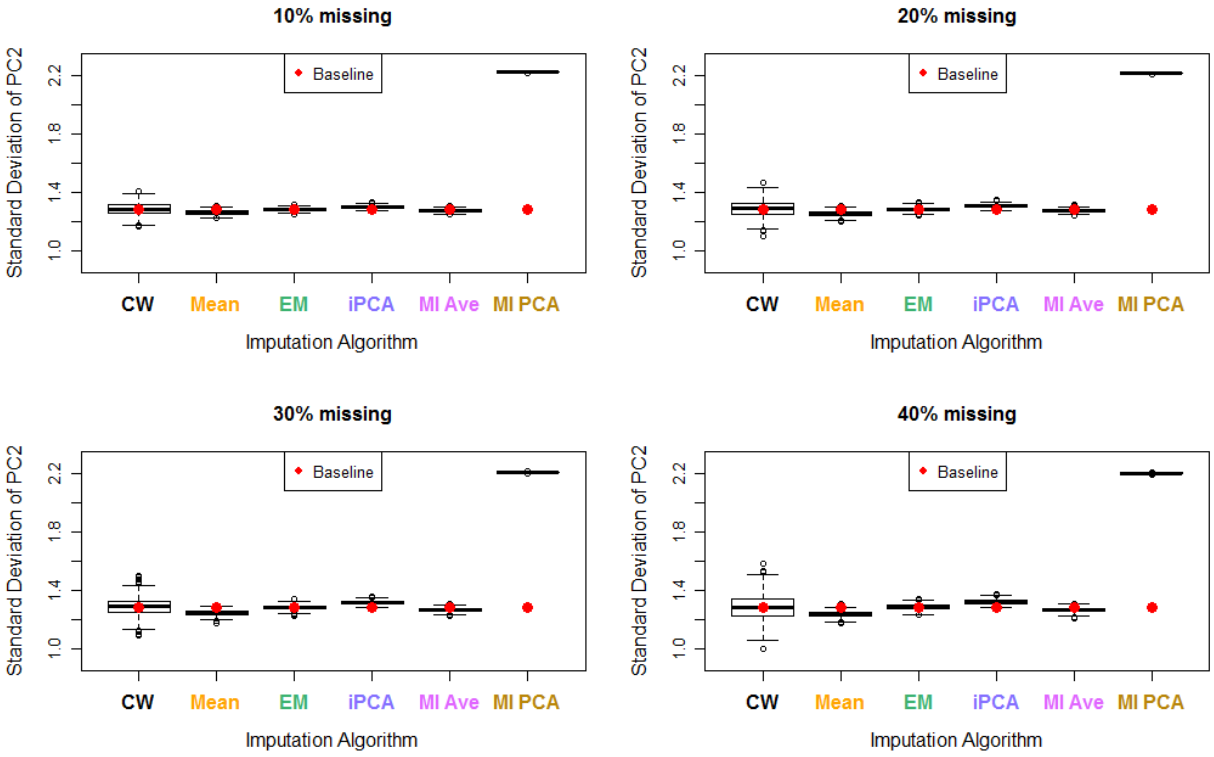

Figure 4.10: PCA output results of the standard deviation of PC2 for MCAR

Goodness-of-fit Results

The ranges of the R_V coefficient for the different imputation algorithms across the different percentages of missingness for MCAR are given in Table 4.14. All of the R_V coefficients proved statistically significant at a 1% significance level. It is notable from the table that as the percentage missingness increases, the upper and lower bounds of the ranges decrease indicating a decline in effectiveness. The ranges highlighted in red show that at 10% and 20% missingness, the upper bounds for iPCA suggest superiority in effectiveness compared to the other algorithms, as the bounds are the closest to 1. At 30% and 40% missingness, MI Ave is the most effective. The ranges for MI PCA show the poor performance of the algorithm compared to the other algorithms suggesting less effectiveness than observed for mean imputation. On average, the R_V coefficient shows a rapid deterioration for mean imputation as the percentage of missingness increases and still the results outperforms MI PCA (Figure 4.11). The average R_V coefficients for iPCA and MI Ave are almost exactly the same.

Table 4.14: Ranges of the R_V coefficient for MCAR

	10%	20%	30%	40%
Mean	(0.956; 0.98)	(0.932; 0.97)	(0.92; 0.956)	(0.905; 0.948)
EM	(0.973; 0.989)	(0.955; 0.982)	(0.946; 0.976)	(0.936; 0.97)
iPCA	**(0.983; 0.994)**	**(0.975; 0.99)**	(0.966; 0.984)	(0.962; 0.98)
MI Ave	(0.982; 0.993)	(0.976; 0.989)	**(0.967; 0.985)**	**(0.96; 0.981)**
MI PCA	(0.925; 0.937)	(0.919; 0.933)	(0.91; 0.928)	(0.904; 0.925)

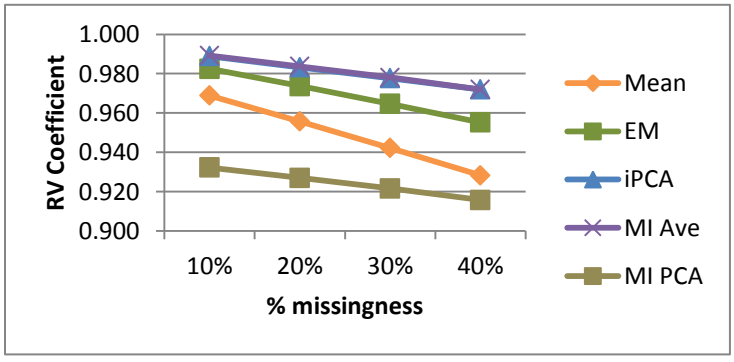

Figure 4.11: Line graph of the average R_V coefficient for MCAR

Similar results follow from the average SSD across the 1000 simulations for the different imputation algorithms and percentages of missingness in Figure 4.12. The size of the bubble in the figure reflects the size of the SSD relative to the other algorithms at each percentages of missingness. On average, the SSD of MI PCA is approximately 4 times the size of mean imputation and approximately 10 times the size of iPCA and MI Ave. The effectiveness of the EM algorithm sits on average between that of the mean imputation and iPCA or MI Ave.

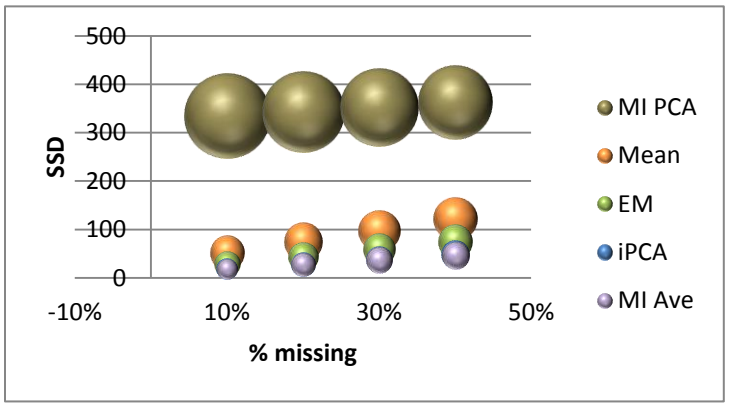

Figure 4.12: Bubble plot of the average SSD for MCAR

4.3.2. Scenario 2: Missing At Random

PCA Model Results

Table 4.15 shows the PCA model results for MAR and similar to the results for MCAR (Table 4.13), only CW and mean imputation resulted in incorrect selections of the number of PC's. There is an increase in the error rate based on the cumulative proportion of variation as the percentages of missingness increase with mean imputation resulting in a 100% error rate at 30% and 40% missingness. In every instance the number of PC's to select was predicted to be three PC's. However, as opposed to MCAR, by applying Kaiser's rule to MAR both CW and mean imputation selected the correct number of PC's.

Table 4.15: PCA model results for MAR

		10%	20%	30%	40%
Cumulative proportion of variation	**CW**	0%	0%	0.1%	0.9%
	Mean	36%	97%	100%	100%
Eigenvalues	**CW**	0%	0%	0%	0%
	Mean	0%	0%	0%	0%

PCA Output Results

The output of the PCA models for the medians is given in Figure 4.13 and Figure 4.14 and the output for the standard deviations in Figure 4.15 and Figure 4.16. From Figure 4.13 it follows that most of the time mean imputation, the EM algorithm and MI Ave accurately predicts the baseline median for PC1 (-0.0375) for all the percentages of missingness. CW overestimates the baseline median completely with a widening range of values as the percentages of missingness increase. iPCA and MI PCA start to slightly underestimate the baseline median from 30% missingness.

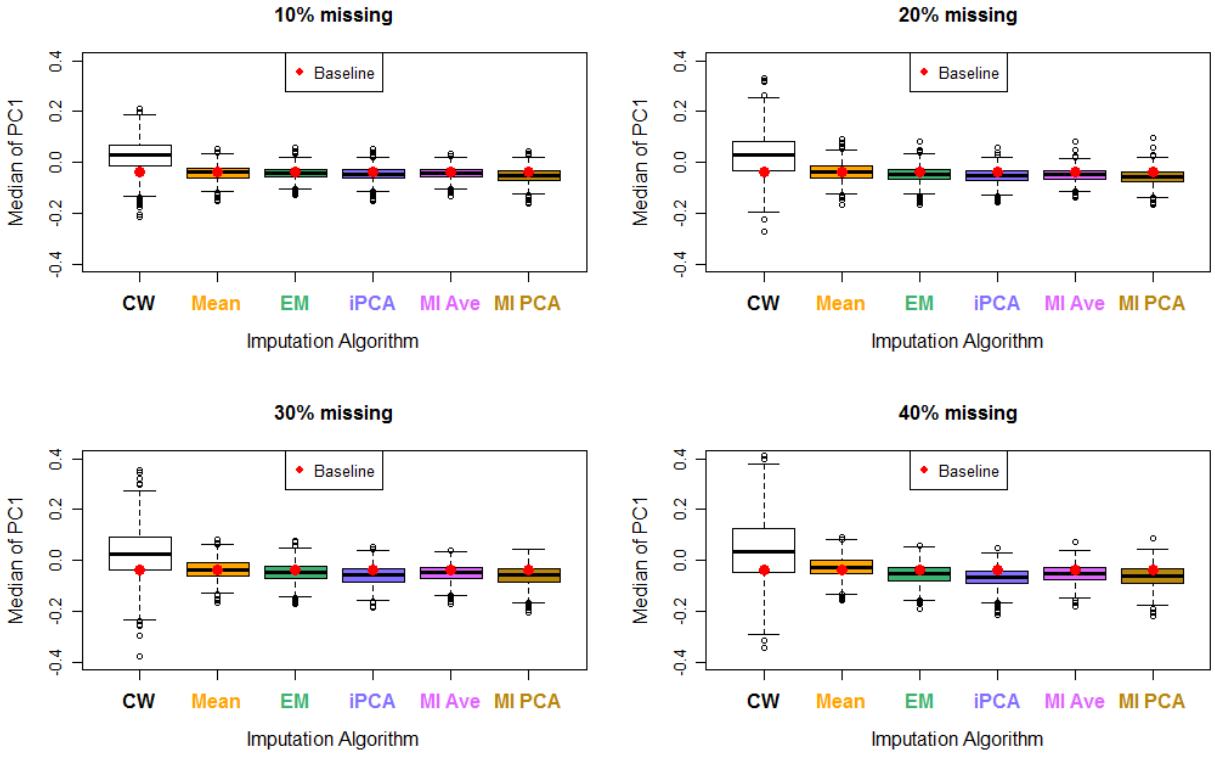

Figure 4.13: PCA output results of the median of PC1 for MAR

From the results for the median of PC2 (Figure 4.14), it can be observed that the baseline median for PC2 (0.0268) is quite aligned to the median from CW. Mean imputation underestimates the baseline median from 20% missingness. MI PCA slightly overestimates the baseline median up until 40% missingness.

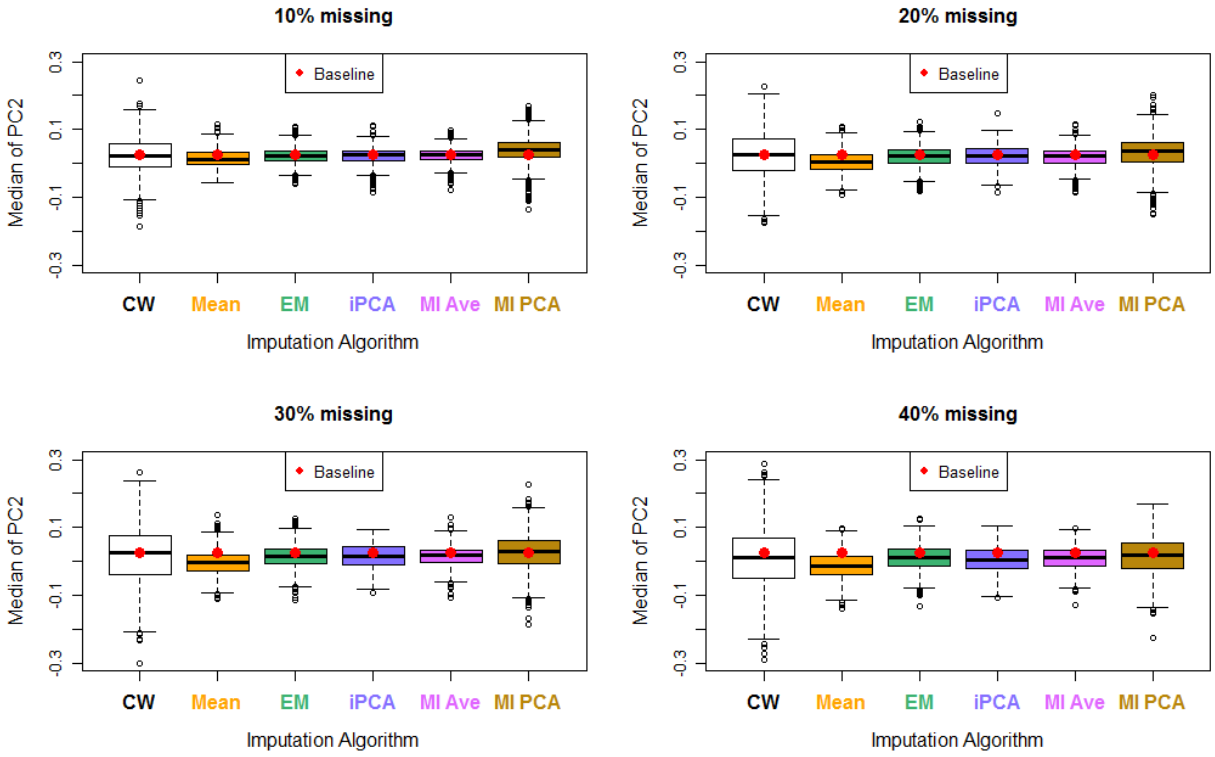

Figure 4.14: PCA output results of the median of PC2 for MAR

The results of the standard deviation for PC1 (Figure 4.15) compared to the baseline standard deviation for PC1 (1.8576) indicate that CW, the EM algorithm and MI Ave accurately predicted the baseline standard deviation as the medians of their distributions. From 10% missingness, the underestimation of the baseline standard deviation observed in mean imputation rapidly increases as the percentages of missingness. MI PCA completely overestimates the baseline standard deviation from 10% missingness.

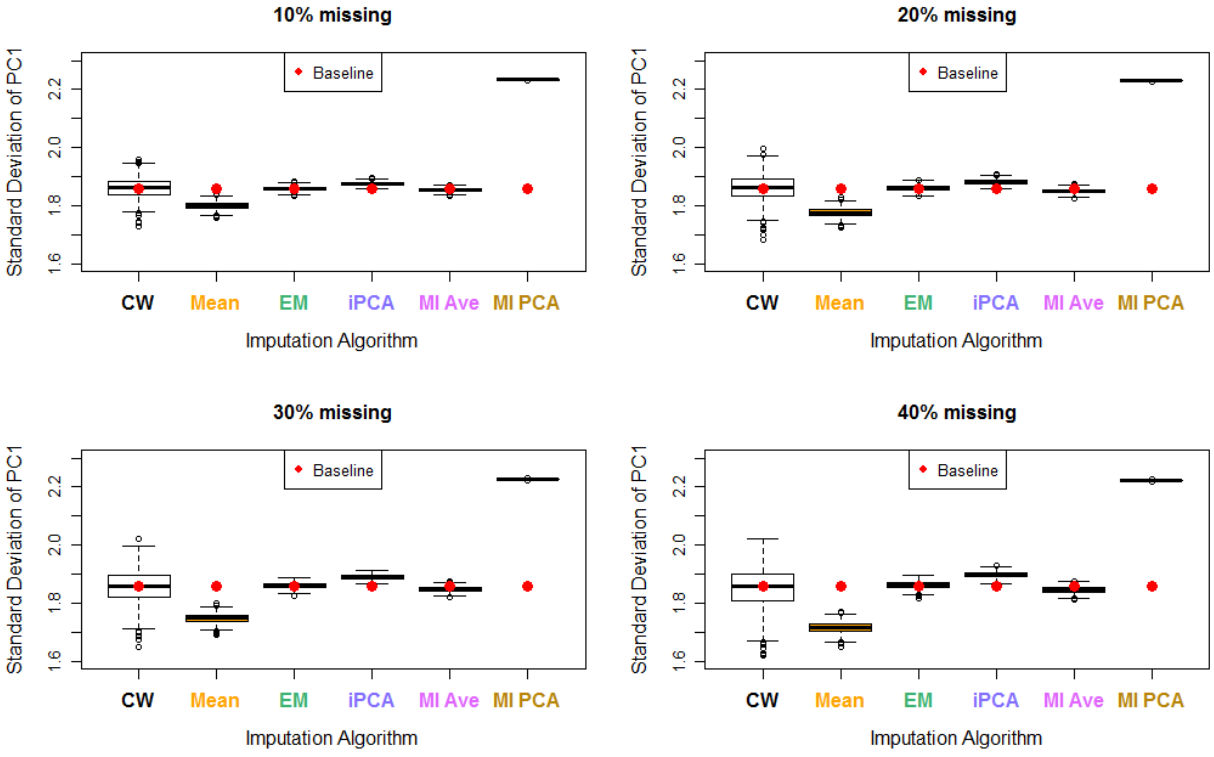

Figure 4.15: PCA output results of the standard deviation of PC1 for MAR

Figure 4.16 shows the results of the standard deviation for PC2 compared to the baseline standard deviation of 1.2823. The EM algorithm and MI Ave reflects the baseline standard deviation quite accurately compstrf to mean imputation and iPCA that start deviating from 20% missingness.

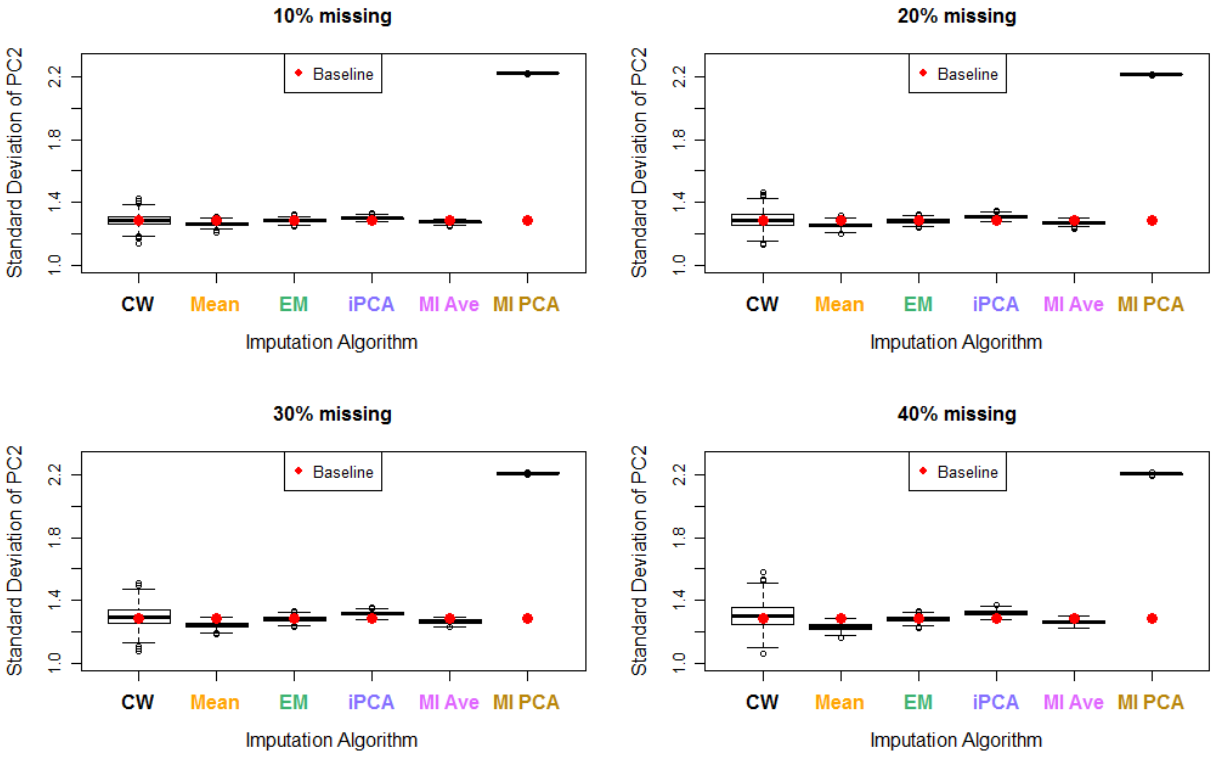

Figure 4.16: PCA output results of the standard deviation of PC2 for MAR

Goodness-of-fit Results

The ranges of the R_V coefficient given in Table 4.16 consist of statistically significant R_V coefficients for MAR at a 1% significance level. The table shows similar results as the R_V coefficients for MCAR in Table 4.14. However, for MAR, MI Ave proves to be the most effective for every percentage of missingness with iPCA equal to MI Ave at 20% missingness and slightly below MI Ave for the other percentages of missingness. MI PCA have the lowest R_V coefficients for most percentages of missingness except at 40% missingness. Figure 4.17 shows that on average, the EM algorithm is more effective than mean imputation and MI PCA but still worse than iPCA and MI Ave, that have almost exactly the same average R_V coefficients.

Table 4.16: Ranges of the R_V coefficient for MAR

	10%	20%	30%	40%
Mean	(0.954; 0.981)	(0.933; 0.969)	(0.914; 0.959)	(0.902; 0.948)
EM	(0.972; 0.99)	(0.96; 0.984)	(0.951; 0.977)	(0.938; 0.969)
iPCA	(0.982; 0.993)	**(0.975; 0.989)**	(0.963; 0.984)	(0.961; 0.98)
MI Ave	**(0.983; 0.994)**	**(0.975; 0.989)**	**(0.97; 0.985)**	**(0.962; 0.98)**
MI PCA	(0.926; 0.937)	(0.917; 0.933)	(0.913; 0.929)	(0.907; 0.924)

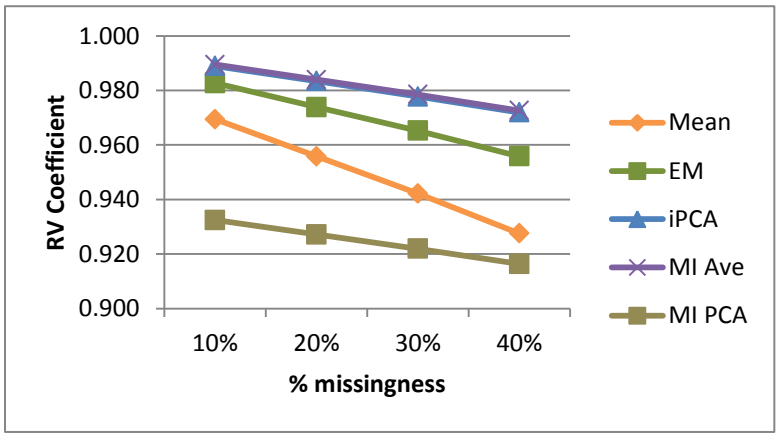

Figure 4.17: Line graph of the average R_V coefficient for MAR

The results above are confirmed by the average SSD observed in Figure 4.18. The figure indicates the big gap between the average SSD for MI PCA and the other algorithms. Similar to Figure 4.12, it follows that the average size of the SSD for MI PCA is approximately 4 times the size of mean imputation and approximately 10 times the size of iPCA and MI Ave.

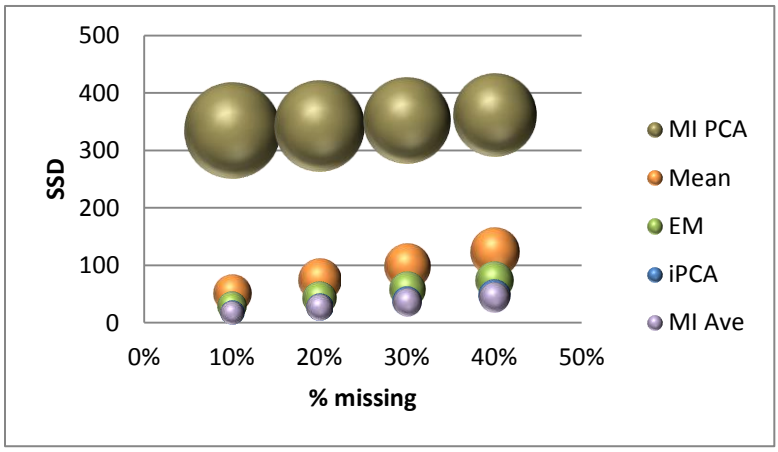

Figure 4.18: Bubble plot of the average SSD for MAR

4.3.3. Scenario 3: Missing Not At Random

PCA Model Results

Looking at the PCA model results for MNAR in Table 4.17, similar trends to MCAR (Table 4.13) and MAR (Table 4.15) can be observed. Given the cumulative proportion of variation, the increase in the error rates is not only observed as the percentages of missingness increase, but also when comparing the error rates for MNAR to MCAR and MAR. The results for MNAR indicate somewhat higher error rates for CW at 30% and 40% missingness as was observed for the other two scenarios. Still, the application of Kaiser's rule decreased the error rates such that only CW has an error of 0.1% at 40% missingness. The incorrect number of PC's selected was three PC's based on the cumulative proportion of variation and one PC based on Kaiser's rule.

Table 4.17: PCA model results for MNAR

		10%	20%	30%	40%
Cumulative proportion of variation	**CW**	0%	0%	0.2%	2.4%
	Mean	37%	98%	100%	100%
Eigenvalues	**CW**	0%	0%	0%	0.1%
	Mean	0%	0%	0%	0%

PCA Output Results

Figure 4.19 to Figure 4.22 shows the output of the PCA model for MNAR. The baseline median for PC 1 (-0.0375) is accurately predicted by most of the algorithms for the different percentages of missingness in Figure 4.19. CW and mean imputation are the only algorithms that deviated from the baseline median. The mean imputation slightly overestimates the baseline median at 40% missingness whereas CW overestimates the baseline median from 10% missing.

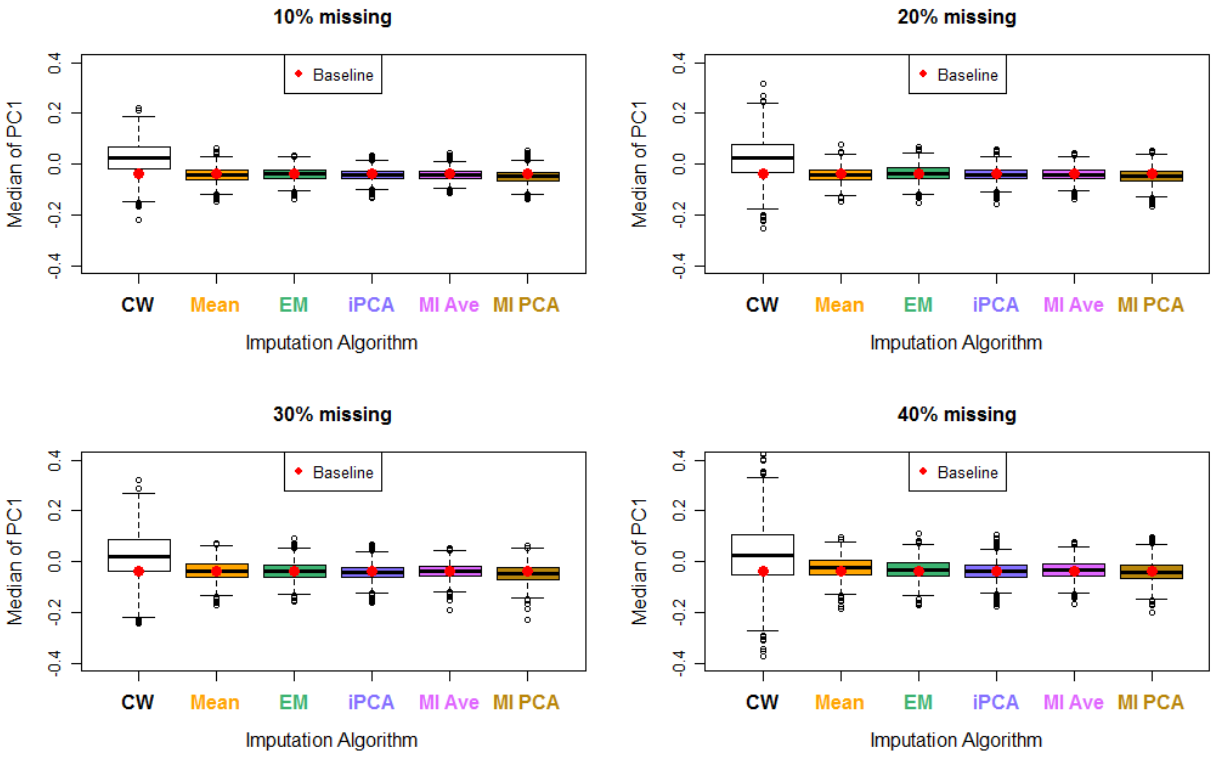

Figure 4.19: PCA output results of the median of PC1 for MNAR

Based on the output for the median of PC2 in Figure 4.20 compared to the baseline median of PC2 (0.0268), it follows that up until 30% missingness majority of the algorithms predicted the baseline median well. At 40% missingness, there is a general slight underestimation observed in all the algorithms.

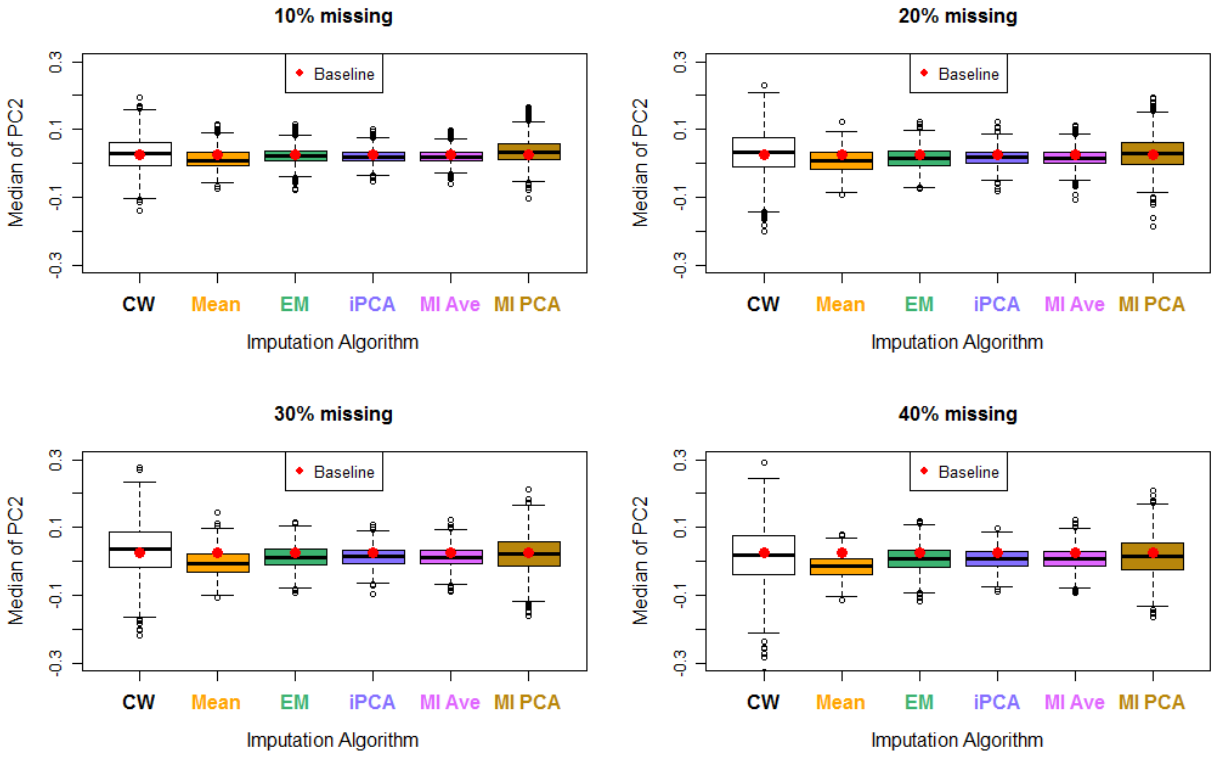

Figure 4.20: PCA output results of the median of PC2 for MNAR

From Figure 4.21 comparing the standard deviation of PC1 to the baseline standard deviation for PC1 (1.8576), it can be observed that for all percentages of missingness CW, the EM algorithm and MI Ave predicted the baseline standard deviation accurately. While the overestimation observed in iPCA gradually increases as the percentages of missingness increase, the underestimation in mean imputation also increases at a faster rate. MI PCA still completely overestimates the baseline standard deviation from 10% missingness.

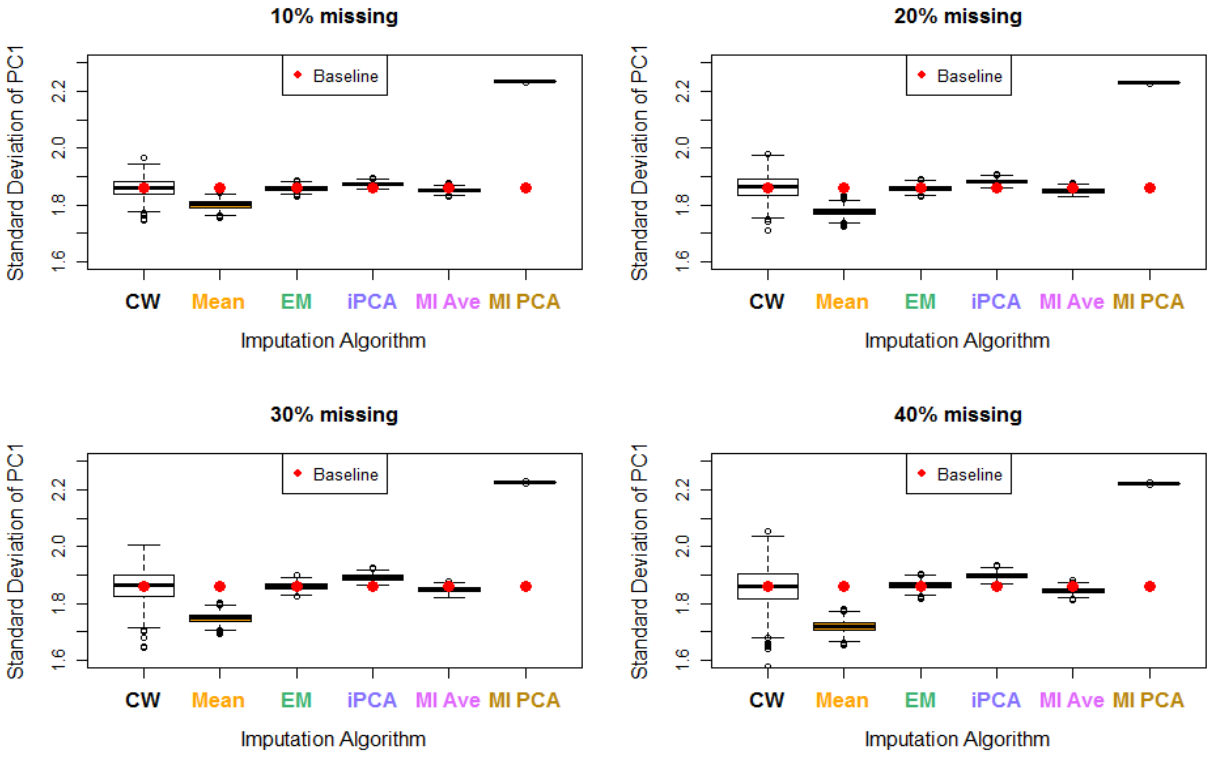

Figure 4.21: PCA output results of the standard deviation of PC1 for MNAR

The results of the standard deviation of PC2 is presented in Figure 4.22 and compared to the baseline standard deviation of PC 2 (1.2823). Except for MI PCA that overestimates the baseline standard deviation at all percentages of missingness, the algorithms generally produce accurate estimates. It is only from 30% missingness that the mean imputation and iPCA start to deviate slightly.

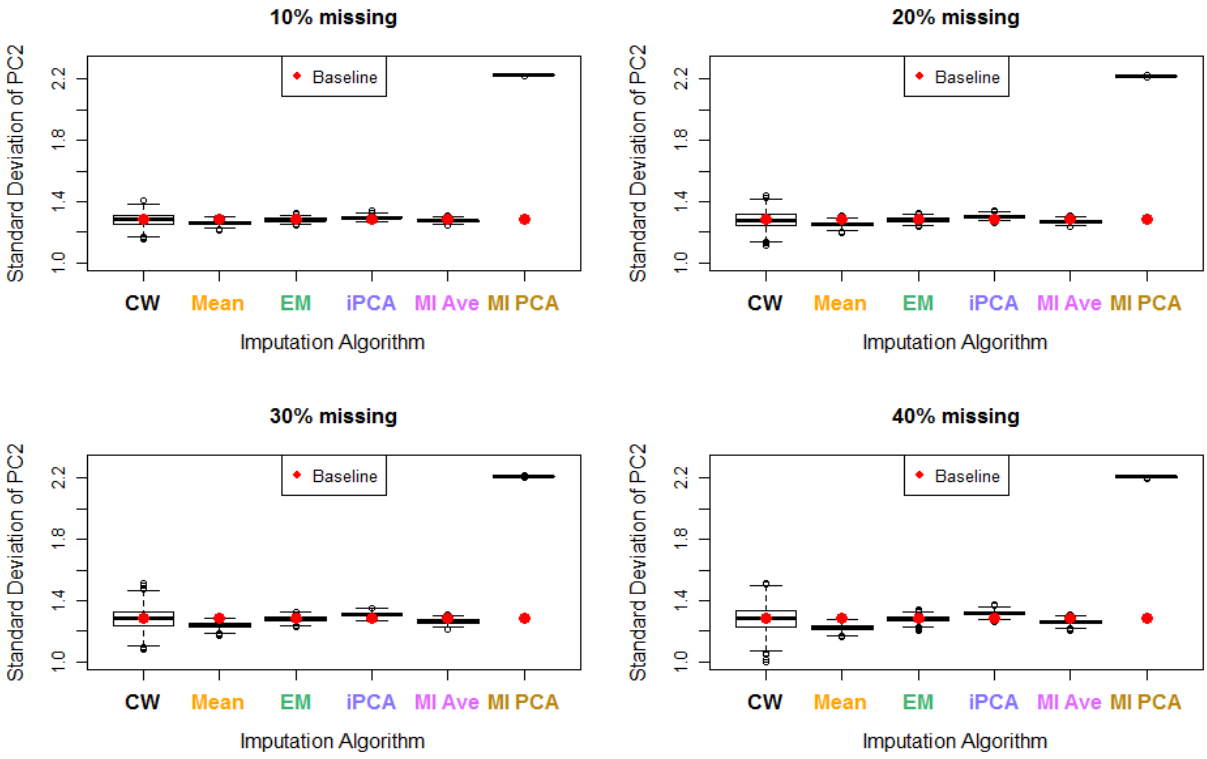

Figure 4.22: PCA output results of the standard deviation of PC2 for MNAR

Goodness-of-fit Results

Table 4.18 shows the ranges of the R_V coefficient for MNAR of which all of the values proved to be statistically significant at a 1% significance level. Contrary to the results that were observed for MCAR (Table 4.14) and MAR (Table 4.16), it follows that iPCA and MI Ave are very similar for all percentages of missingness. Similarly to MAR, MI PCA is the least effective for all percentages of missingness except at 40% where the mean imputation has a lower minimum bound than MI PCA. On average, the R_V coefficient for mean imputation at 40% missingness is almost as low as the R_V coefficient for MI PCA but MI PCA is still the least effective imputation algorithm.

Table 4.18: Ranges of the R_V coefficient for MNAR

	10%	20%	30%	40%
Mean	(0.954; 0.981)	(0.937; 0.972)	(0.912; 0.955)	(0.897; 0.942)
EM	(0.973; 0.989)	(0.961; 0.983)	(0.945; 0.975)	(0.933; 0.968)
iPCA	**(0.983; 0.993)**	**(0.974; 0.989)**	(0.967; 0.984)	**(0.962; 0.979)**
MI Ave	**(0.983; 0.993)**	**(0.974; 0.989)**	**(0.969; 0.984)**	(0.96; 0.979)
MI PCA	(0.926; 0.937)	(0.917; 0.933)	(0.913; 0.928)	(0.903; 0.923)

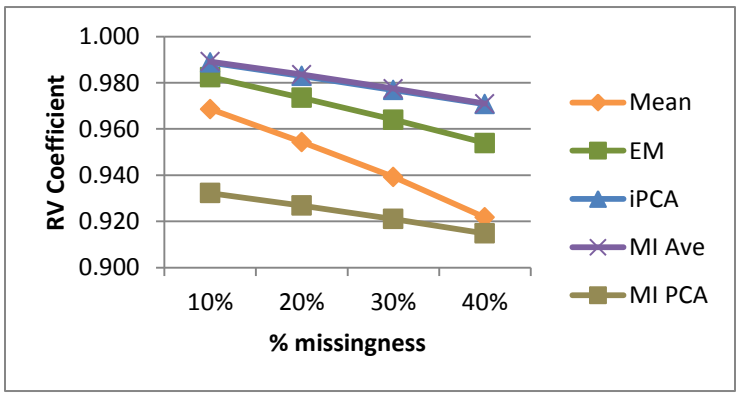

Figure 4.23: Line graph of the average R_V coefficient for MNAR

The bigger size of the average SSD for MI PCA that was observed for MCAR (Figure 4.12) and MAR (Figure 4.18), can also be observed for MNAR in Figure 4.24. The figure shows a clear distinction between the average SSD for MI PCA and mean imputation at 40% missingness even though the average R_V coefficient for mean imputation at 40% missingness is close to the average R_V coefficient for MI PCA (Figure 4.23). It also follows that the EM algorithm, iPCA and MI Ave are on average more effective than Mean Imputation.

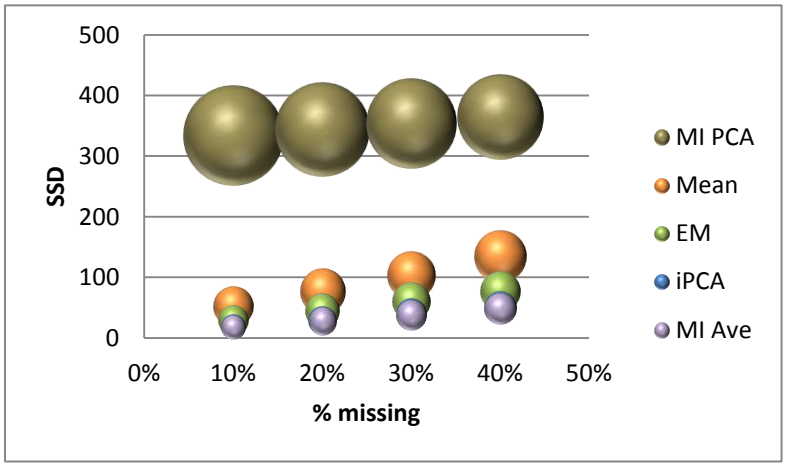

Figure 4.24: Bubble plot of the average SSD for MNAR

4.4. Summary

The results for the overall simulation analysis in Section 4.3 coincide with the results that were observed for the single simulation analysis in Section 4.2. In both analyses, there are three parts to the evaluation of the results, namely the PCA model, the PCA output and the goodness-of-fit measures.

The first objective of the analysis was to determine whether the imputation algorithm selected the same number of PC's to be extracted in the PCA model as the baseline PCA model. For this purpose both the criteria cumulative proportion of variation and Kaiser's rule were applied. The results showed that only CW and Mean Imputation were affected with instances for which the incorrect number of PC's was selected. Majority of these instances were observed for the cumulative proportion of variation.

The second part of the analysis focused on a variable level analysis to compare the imputed PC's to the baseline PC's. For this step of the analysis, the descriptive statistics consisting of the median and the standard deviation were calculated for both the PC's from the imputed and the baseline PCA models. The biggest deviations from the baseline medians were observed for CW, and the biggest deviations from the baseline standard deviations for Mean Imputation and MI PCA.

Finally, a goodness-of-fit analysis was performed in order to compare the imputed results with the baseline results on a respondent level. The measures that were used for the analysis include the R_V coefficient and the SSD. A significance test performed on the R_V coefficients indicated that all the values are statistically significant at a 1% significance level. Both the R_V coefficients and SSD concluded that iPCA and MI Ave were the most effective algorithms and MI PCA the least effective.

5. CONCLUSIONS

5.1. Conclusions

Missing data are a problem that needs to be considered but researchers tend to ignore the problem. As Schafer and Graham (2002, p.147) state: *"Missingness is usually a nuisance, not the main focus of inquiry"*. In this study, the focus of the analysis is missing data specifically in the context of PCA. Grung and Manne (1998) mention that if missing data are handled incorrectly, the results of the PCA may be severely affected.

The study considered a range of imputation algorithms that have not yet been compared including CW, mean imputation, the EM algorithm, iPCA and MI. In the case of MI, two methods for combining the multiple datasets into a single dataset were considered, namely the application of a simple average across the PC's from the multiple imputed datasets (MI Ave) and the application of a weighted average (MI PCA). The efficiency of the methods for imputing missing data was assessed using a baseline dataset simulated from a multivariate normal distribution and introducing the three missing mechanisms as three possible scenarios for four percentages of missingness. A PCA was performed and the results from the imputed datasets are compared with the results from the baseline data, including the cumulative proportion of variation and the eigenvalues. The goodness-of-fit of the PCA models were assessed using the descriptive statistics, the SSD and the R_V coefficient.

The first observation is that based on Kaiser's rule it can be concluded that mean imputation, the EM algorithm, iPCA and MI had no problem with selecting the same number of PC's as the baseline PCA model. Secondly, a comparison of the descriptive statistics showed that the only algorithms that produced descriptive statistics comparable to the baseline were the EM algorithm, iPCA and MI Ave. Finally, the goodness-of-fit results revealed that iPCA and MI Ave were the most effective imputation algorithms from this study. Similar to literature, the effectiveness decreased as the percentages of missingness increased. Contrary to research and expectations, the study showed that the influence of the MNAR missingness, specifically, on the results was minimal. A possible explanation can be because the missingness was only

introduced in a single variable for each PC and as such the effect is minimal as it is overshadowed by the existing relationships among the rest of the variables.

Based on the research questions asked at the beginning of this study, the conclusions are as follow: Taking a simple average of the multiple values provided an effective way to combine the results from MI. The EM algorithm, iPCA and MI Ave methods proved more effective than CW and mean imputation with both iPCA and MI Ave being the most efficient imputation algorithms.

While only a few methods are investigated in this study, there are many more methods each with their own advantages and disadvantages. Schlomer *et al* (2010, p.3) state that *"There is not one best strategy; the strategy will depend on the data and the analyses."* Even though the methods for handling missing data are distinct, all the methods have the same goal: To impute the missing observations as accurately as possible in order to obtain results and conclusions that the study would have given if all the observations were observed.

5.2. Recommendations

Although a growing field, the research in missing data is still far from complete, specifically for missing data in the context of statistical techniques other than linear regression, such as PCA. This analysis has shown that MI Ave produced efficient results and should be investigated further to determine whether the variability within and between imputations should be incorporated in the final PCA derivation, similar to what Rubin (1987) did for linear regression. MI PCA on the other hand produced variable results and should also be investigated as to the reason for the variability. The efficient results of iPCA also suggest further research into the most recent mention of an MI version of iPCA. The influence of the missing data mechanisms, with specific focus on MNAR missingness, on the results of PCA also deserves further research. This can be achieved by analysing the effect on the results when the missingness is introduced into more of the variables. Another focus can be to compare the correlation matrices between the baseline and imputed PC's using the determinant of matrices, or any other measure that can summarise the correlation matrices.

6. REFERENCES

Abdi, H. and Williams, L.J. (2010). Principal component analysis, *Wiley Interdisciplinary Reviews: Computational Statistics*, **2**, 433–459.

Anderson, T.W. (1963). Asymptotic theory for principal component analysis, *Annals of Mathematical Statistics*, **34**, 122–148.

Audigier, V., Husson, F., and Josse, J. (2013). A principal components method to impute missing values for mixed data, *arXiv preprint arXiv:1301.4797*.

Barnard, J. and Meng, X.L. (1999). Applications of multiple imputation in medical studies: from AIDS to NHANES, *Statistical Methods in Medical Research*, **8**, 17–36.

Behrens, J.T. (1997). Principles and procedures of exploratory data analysis, *Psychological Methods*, **2**, 131–160.

Beltrami, E. (1873). Sulle funzioni bilineari, *Giornale di Mathematiche di Battaglini*, **11**, 98–106.

Bennett, D.A. (2001). How can I deal with missing data in my study? *Australian and New Zealand Journal of Public Health*, **25**, 464–469.

Bro, R. and Smilde, A.K. (2014). Principal component analysis, *Analytical Methods*, **6**, 2812–2831.

Burns, R.A., Butterworth, P., Kiely, K.M., Bielak, A.A.M., Luszcz, M.A., Mitchell, P., Christensen, H., Von Saden, C. and Anstey, K.J. (2011). Multiple imputation was an efficient method for harmonizing the Mini-Mental State Examination with missing item-level data, *Journal of Clinical Epidemiology*, **64**, 787–793.

Cattell, R.B. (1966). The scree test for the number of factors, *Multiv. Behav. Res.*, **1**, 245–276.

Chambers, J.M. (1977). *Computational Methods for Data Analysis*, New York: Wiley, Inc. 268 pages.

Chen, H. (2002). Principal component analysis with missing data and outliers, *URL: http://www. cmlab. csie. ntu. edu. tw/~ cyy/learning/papers/PCA_Tutorial. pdf*.

Christoffersson, A. (1970). *The One Component Model with Incomplete Data*, PhD thesis, Uppsala University.

Collins, L.M., Schafer, J.L. and Kam, C.M. (2001). A comparison of inclusive and restrictive strategies in modern missing data procedures, *Psychological Methods*, **6**, 330–351.

Dear, R E. (1959). A principal components missing data method for multiple regression models, *Technical Report SP-86, Santa Monica: Systems Development Corporation*.

Dempster, A.P., Laird, N.M. and Rubin, D.B. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion), *Journal of the Royal Statistical Society, Series B (Methodological)*, **39**, 1–38.

Diamantaras, K.I. and Kung, S.Y. (1996). *Principal Component Neural Networks Theory and Applications*, New York: Wiley, Inc., 255 pages.

Donders, A.R.T., Van der Heijden, G.J.M.G., Stijnen, T. and Moons, K.G.M. (2006). Review: A gentle introduction to imputation of missing values, *Journal of Clinical Epidemiology*, **59**, 1087–1091.

Durrant, G.B. (2005). Imputation methods for handling item-nonresponse in the social sciences: a methodological review, *ESRC National Centre for Research Methods and Southampton Statistical Sciences Research Institute. NCRM Methods Review Papers NCRM/002*.

Escofier, B. and Pagès, J. (2008). *Analyses factorielles simples et multiples: objectifs, méthodes et interprétation*, Dunod.

Escoufier, Y. (1973). Le traitement des variables vectorielles. *Biometrics*, **29**, 751–760.

Farmer, S.A. (1971). An investigation into the results of principal component analysis of data derived from random numbers, *Statistician*, **20**, 63–72.

Fisher, R. and MacKenzie, W. (1923). Studies in crop variation. II. The manurial response of different potato varieties, *Journal of Agricultural Science*, **13,** 311–320.

Gabriel, K.R. and Zamir, S. (1979). Lower rank approximation of matrices by least squares with any choice of weights, *Technometrics*, **21**, 489–498.

Gilks, W.R., Richardson, S. and Spiegelhalter, D.J. (1996). *Markov Chain Monte Carlo in practice*, 1st ed., London: Chapman and Hall, 486 pages.

Girshick, M.A. (1939). On the sampling theory of roots of determinantal equations, *Annals of Mathematical Statistics*, **10**, 203–224.

Gnanadesikan, R. (1977). *Methods for Statistical Data Analysis of Multivariate Observations*, New York: Wiley, Inc., 311 pages.

Gower, J.C. (1966). Some distance properties of latent root and vector methods used in multivariate analysis, *Biometrika*, **53**, 325–338.

Greenacre, M. (1984). *Theory and applications of correspondence analysis*, London: Acadamic Press, pages.

Groves, R.M., Fowler Jr., F.J., Couper, M.P., Lepkowski, J.M., Singer, E. and Tourangeau, R. (2009). *Survey Methodology*, 2nd ed., New Jersey: John Wiley and Sons, Inc., 461 pages.

Grung, B. and Manne, R. (1998). Missing values in principal component analysis, *Chemometrics and Intelligent Laboratory Systems*, **24**, 125–139.

Healy, M.J.R. and Wesmacott, M. (1956). Missing values in experiments analyzed on automatic computers, *Applied statistics*, **5**, 203–206.

Horton, N.J. and Lipsitz, S.R. (2001). Multiple imputation in practice: comparison of software packages for regression models with missing variables, *American Statistical Association*, **55**, 244–254.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components, *Journal of Educational Psychology*, **24**, 417–441, 498–520.

Hotelling, H. (1936). Simplified calculation of principal components, *Psychometrika*, **1**, 27–35.

Hubert, M., Van Kerckhoven, J., and Verdonck, T. (2012). Robust PARAFAC for incomplete data, *Journal of Chemometrics*, **26**, 290-298.

Husson, F. and Josse, J. (2013). *missMDA: Handling missing values with/in multivariate data analysis (principal component methods)*, R package version 1.7.1, http://CRAN.R-project.org/package=missMDA.

Husson, F., Josse, J., Le, S. and Mazet, J. (2013). *FactoMineR: Multivariate Exploratory Data Analysis and Data Mining with R*, R package version 1.25, http://CRAN.R-project.org/package=FactoMineR.

Ilin, A. and Raiko, T. (2010). Practical approaches to principal component analysis in the presence of missing values, *Journal of Machine Learning Research*, **11**, 1957–2000.

Janssen, K.J.M., Donders, A.R.T., Harrell Jr., F.E., Vergouwe, Y., Chen, Q., Grobbee, D.E. and Moons, K.G.M. (2010). Missing covariate data in medical research: to impute is better than to ignore, *Journal of Clinical Epidemiology*, **63**, 721–727.

Jeffers, J.N.R. (1967). Two case studies in the application of principal component analysis, *Applied Statistics*, **16**, 225–236.

Jeffers, J.N.R. (1994). The Importance of Exporatory Data Analysis Before the use of Sophisticated Procedures. *Biometrics*, **50**, 881-883.

Johnson, R.A. and Wichern, D.W. (1998). *Applied multivariate statistical analysis*, 4th ed., Prentice-Hall, Inc., New Jersey, 288 pages.

Jolliffe, I.T. (2002). *Principal component analysis*, 2nd ed., Springer-Verlag New York, Inc., 487 pages.

Jordan, M.C. (1874). Mémoire sur les Formes Bilinéaires, *Journal de Mathématiques Pures et Appliquées*, **19**, 35–54.

Josse, J., Chavent, M., Liquet, B. and Husson, F. (2012). Handling missing values with regularized iterative multiple correspondence analysis, *Journal of Classification*, **29**, 91–116.

Josse, J. and Husson, F. (2012a). Handling missing values in exploratory multivariate data analysis methods, *Journal de la Société Française de Statistique*, **153**, 79–99.

Josse, J. and Husson, F. (2012b). Selecting the number of components in principal component analysis using cross-validation approximations, *Computational Statistics and Data Analysis*, **56**, 1869–1879.

Josse, J., Husson, F. and Pagès, J. (2008). Testing the significance of the RV coefficient, Computational Statistics and Data Analysis, **53**, 82–91.

Josse, J., Husson, F. and Pagès, J. (2009). Gestion des données manquantes en Analyse en Composantes Principales, *Journal de la Société Française de Statistique*, **150**, 28–51.

Josse, J., Husson, F. and Pagès, J. (2011). Multiple imputation in principal component analysis, *Advances in data analysis and classification*, **5**, 231–246.

Kaiser, H.F. (1960). The application of electronic computers to factor analysis, *Educ. Psychol. Meas.*, **20**, 141–151.

Kiers, H.A.L. (1997) Weighted least squares fitting using ordinary least squares algorithms, *Psychometrica*, **62**, 251–266.

Klebanoff, M.A. and Cole, S.R. (2008). Use of multiple imputation in the epidemiologic literature, *American Journal of Epidemiology*, **168**, 355–357.

Knol, M.J., Janssen K.J.M., Donders A.R.T., Egberts A.C.G., Heerdink, E.R., Grobbee, D.E., Moons, K.G.M. and Geerlings M.I. (2010). Unpredictable bias when using the missing indicator method or complete case analysis for missing confounder values: an empirical example, *Journal of Clinical Epidemiology*, **63**, 728–736.

Kroonenberg, P.M. (2008). *Applied multiway data analysis*, Wiley series in probability and statistics, USA: John Wiley and Sons, Inc., 579 pages.

Lepper, M. R., Corpus, J. H., and Iyengar, S. S. (2005). Intrinsic and extrinsic motivational orientations in the classroom: age differences and academic correlates, *Journal of educational psychology*, **97**, 184–196.

Little, R.J.A. (1988). A test of missing completely at random for multivariate data with missing values, *Journal of American Statistical Association*, **83**, 1198–1202.

Little, R. (2011). Calibrated Bayes, for Statistics in general, and missing data in particular, *Statistical Science*, **26**, 162–174.

Little, R.J.A. and Rubin, D.B. (2002). *Statistical analysis with missing data*, 2nd ed., New York: John Wiley and Sons, Inc., 381 pages.

Mandel, J. (1982). Use of the singular value decomposition in regression analysis, *American Statistical Association*, **36**, 15–24.

Marshall, A., Altman, D.G. and Holder, R.L. (2010). Comparison of imputation methods for handling missing covariate data when fitting a Cox proportional hazards model: a resampling study, *BMC Medical Research Methodology*, **10**, 112–122.

Morrison, D.F. (1976). *Multivariate Statistical Methods*, 2nd ed., Tokyo: McGraw-Hill Kogakusha, Inc., 338 pages.

Mosier, C.I. (1951). Problems and designs of cross-validation, *Educ. Psychol. Meas.*, **11**, 5–11.

Nakagawa, S. and Freckleton, R.P. (2010). Model averaging, missing data and multiple imputation: a case for behavioural ecology, *Behavioural Ecology Sociobiology*, **65**, 103–116.

Ndiaye, D., and Gabriel, K. (2011). Principal component analysis of the electricity consumption in residential dwellings, *Energy and buildings*, **43**, 446-453.

Nora-Chouteau, C. (1974). Une méthode de reconstitution et d'analyse de données incomplètes. PhD thesis, Université Pierre et Marie Curie.

Pearson, K. (1901). On lines and planes of closest fit to systems of points in space, *Philosophical Magazine*, **2**, 559–572.

Peres-Neto, P.R., Jackson, D.A. and Somers, K.M. (2003). Giving meaningful interpretation to ordination axes: assessing loading significance in principal component analysis, *Ecology*, **84**, 2347–2363.

Popov, S. (2006). Large-scale data visualization with missing values, *Technological and Economic Development of Economy*, **12**, 44-49.

R Core Team (2012). *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0, URL http://www.R-project.org/.

Rao, C.R. (1964). The use and interpretation of principal component analysis in applied research, *Sankhya A*, **26**, 329–358.

Rubin, D.B. (1987). *Multiple imputation for nonresponse in surveys*, 1st ed., New York: John Wiley and Sons, Inc., 258 pages.

Rubin, D.B. and Schenker, N. (1991). Multiple imputation in health-care databases: An overview and some applications, *Statistics in Medicine*, **10**, 585–598.

Schaefer, J., Opgen-Rhein, R., Zuber, V., Silva, A.P.D. and Strimmer, K. (2011). *corpcor: Efficient Estimation of Covariance and (Partial) Correlation*, R package version 1.6.0, http://CRAN.R-project.org/package=corpcor.

Schafer, J.L. (1997). *Analysis of incomplete multivariate data*, 1st ed., London: Chapman and Hall, 430 pages.

Schafer, J.L. (1999). Multiple imputation: a primer, *Statistical Methods in Medical Research*, **8**, 3–15.

Schafer, J.L. (2010). *norm: Analysis of multivariate normal datasets with missing values*, R package version 1.0-9.2, http://CRAN.R-project.org/package=norm.

Schafer, J.L. and Graham, J.W. (2002). Missing data: our view of the state of the art, *American Psychological Association*, **7**, 147–177.

Schafer, J.L. and Olsen, M.K. (1998). Multiple imputation for multivariate missing-data problems: A data analyst's perspective, *Multivariate Behavioral Research*, **33**, 545–571.

Schlomer, G.L., Bauman, S. and Card, N.A. (2010). Best practices for missing data management in counseling psychology, Journal of Counseling Psychology, **57**, 1–10.

Schneider, T. (2001). Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values, *Journal of Climate*, **14**, 853–871.

Serneels, S., and Verdonck, T. (2008). Principal component analysis for data containing outliers and missing elements, *Computational Statistics and Data Analysis*, **52**, 1712-1727.

Shlens, J. (2014). A tutorial on principal component analysis, *arXiv preprint arXiv:1404.1100*.

Smith, B.T., Boyle, J.M., Dongarra, J.J., Garbow, B.S., Ikebe, Y., Klema, V.C., and Moler, C.B. (1976). *Matrix Eigensystem Routines: EISPACK guide*, 2nd ed., Berlin: Springer-Verlag, Inc., 387 pages.

Stuart, E.A., Azur, M., Frangakis, C. and Leaf, P. (2009). Multiple imputation with large data sets: A case study of the Children's Mental Health Initiative, *American Journal of Epidemiology*, **169**, 1133–1139.

Tanner, M.A. and Wong W.H. (2010). From EM to data augmentation: the emergence of MCMC Bayesian computation in the 1980s, *Statistical Science*, **25**, 506–516.

Timmerman, M.E. (2006). Multilevel component analysis, *British Journal of Mathematical and Statistical Psychology*, **59**, 301–320.

Tipping, M. and Bishop, C.M. (1999). Probabilistic principal component analysis, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **61**, 611–622.

Tomasi, G., and Bro, R. (2005). PARAFAC and missing values, *Chemometrics and Intelligent Laboratory Systems*, **75**, 163-180.

Tomazzoni, G., Meira, M., Quintella, C. M., Zagonel, G. F., Costa, B. J., de Oliveira, P. R., Pepe, I.M. and da Costa Neto, P. R. (2014). Identification of vegetable oil or biodiesel added to diesel using fluorescence spectroscopy and principal component analysis, *Journal of the American Oil Chemists' Society*, **91**, 215-227.

Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*, 4th ed, Springer, New York, ISBN 0-387-95457-0.

Wagner, S.M. and Kemmerling, R. (2010). Handling nonresponse in logistics research, *Journal of Business Logistics*, **31**, 357–381.

Walczak, B., and Massart, D.L. (2001). Dealing with missing data: Part I, *Chemometrics and Intelligent Laboratory Systems*, **58**, 15-27.

Wilkinson, J.H. (1965). *The Algebraic Eigenvalue Problem*. Oxford: Oxford University Press, 662 pages.

Wold, H. (1966). Nonlinear estimation by iterative least squares procedures, in F. David (Editor), *Research Papers in Statistics*, New York: Wiley, 411–444.

Wold, S. (1978). Cross-validatory estimation of the number of components in factor and principal components models, *Technometrics*, **20**, 397–405.

Wold, S., Esbensen, K. and Geladi, P. (1987). Principal component analysis, *Chemometrics and Intelligent Laboratory Systems*, **2**, 37–52.

Yuan, Y.C. (2000). Multiple imputation for missing data: concepts and new development, *Proceedings of the twenty-fifth annual SAS Users Group International conference* (Paper no. 267), Cary, NC: SAS Institute.

Young, G. 1941. Maximum likelihood estimation and factor analysis, *Psychometrika*, **6**, 49–53.

APPENDIX: R CODE

Load libraries needed:

```
library(corpcor)
library(MASS)
library(norm)
library(Matrix)
library(FactoMineR)
library(missMDA)
```

Simulate baseline dataset:

```
simulateDataRandomCorr=function(n, seedValue){
        #Set random generator seed
        rngseed(seedValue)
        #Set correlation bounds
        LowMin=-0.35;       LowMax=-0.15
        HighMin=0.7;        HighMax=0.85
        #Set correlations
        x1x1=1;
        x1x2=runif(1,min=HighMin,max=HighMax);
        x1x3=runif(1,min=HighMin,max=HighMax);
        x1x4=runif(1,min=LowMin,max=LowMax);
        x1x5=runif(1,min=LowMin,max=LowMax);
        x1x6=runif(1,min=LowMin,max=LowMax);
        x2x1=x1x2;
        x2x2=1;
        x2x3=runif(1,min=HighMin,max=HighMax);
        x2x4=runif(1,min=LowMin,max=LowMax);
        x2x5=runif(1,min=LowMin,max=LowMax);
        x2x6=runif(1,min=LowMin,max=LowMax);
```

```
x3x1=x1x3;
x3x2=x2x3;
x3x3=1;
x3x4=runif(1,min=LowMin,max=LowMax);
x3x5=runif(1,min=LowMin,max=LowMax);
x3x6=runif(1,min=LowMin,max=LowMax);
x4x1=x1x4;
x4x2=x2x4;
x4x3=x3x4;
x4x4=1;
x4x5=runif(1,min=HighMin,max=HighMax);
x4x6=runif(1,min=HighMin,max=HighMax);
x5x1=x1x5;
x5x2=x2x5;
x5x3=x3x5;
x5x4=x4x5;
x5x5=1;
x5x6=runif(1,min=HighMin,max=HighMax);
x6x1=x1x6;
x6x2=x2x6;
x6x3=x3x6;
x6x4=x4x6;
x6x5=x5x6;
x6x6=1;
#Construct correlation matrix
cor_mat=matrix(c(x1x1,x1x2,x1x3,x1x4,x1x5,x1x6,
                 x2x1,x2x2,x2x3,x2x4,x2x5,x2x6,
                 x3x1,x3x2,x3x3,x3x4,x3x5,x3x6,
                 x4x1,x4x2,x4x3,x4x4,x4x5,x4x6,
                 x5x1,x5x2,x5x3,x5x4,x5x5,x5x6,
```

```
                    x6x1,x6x2,x6x3,x6x4,x6x5,x6x6),6,6)
    #Calculate mean vector of 0
    mu=c(0)
    for (i in 1:6){
        mu[i]=0
    }
    #Make sure correlation matrix is positive definite
    if(is.positive.definite(cor_mat)=="FALSE"){
        cor_mat=make.positive.definite(cor_mat)
    }
    is.positive.definite(cor_mat)
    #Simulate data
    data_sim=mvrnorm(n, mu, cor_mat)
    #Name the variables
    colnames(data_sim)=c("X1","X2","X3","X4","X5","X6")
    return(list(data=data_sim,
            corr=cor_mat))
}
```

Create the missing data mechanisms:

```
MCAR=function(dataset,miss,var_MCAR){
    n_row=nrow(dataset)  #Number of rows in dataset under consideration
    num_miss=round(miss*n_row,0)   #Number of missing values to create
    #Number of variable in dataset that must contain the missing values
    variable=var_MCAR
    i=c(0)
    #Random rows that will contain the missing values
    i=sample(1:n_row,num_miss,replace=F)
    #Create the missing observation
    for (x in 1:num_miss){
```

```
            dataset[i[x],variable]=NA
        }
        return(dataset)
}

MAR=function(dataset,miss,pMAR,var_MAR,var_createMAR,cutoff){
        variable=var_MAR  #Variable that should contain MAR missingness
        n=nrow(dataset)   #Number of observations in original dataset
        #Only create pMAR% of miss as MAR and the rest as MCAR
        num_specific=round(miss*pMAR*n,0)        #Number of MAR to create
        num_other=round(miss*n,0)-num_specific   #Number of MCAR to create
        #Randomly select values to delete conditional on the cutoff value for var_createMAR
         specific=which(dataset[,var_createMAR]>cutoff)
        i=c(0)
        i=sample(specific,num_specific,replace=F)
        for (x in 1:num_specific){
            dataset[i[x],variable]=NA
        }
        #Randomly select the values that will MCAR
        other=which(dataset[,var_createMAR]<=cutoff)
        j=c(0)
        j=sample(other,num_other,replace=F)
        for (x in 1:num_other){
            dataset[j[x],variable]=NA
        }
        return(dataset)
}
```

```
MNAR=function(dataset,miss,pMNAR,var_MNAR,cutoff){
        #MNAR will be the same as MAR but the missingness will be conditional on
        #the same variable that will contain the missing values
        dataset_MNAR=MAR(dataset,miss,pMNAR,var_MNAR,var_MNAR,cutoff)
         return(dataset_MNAR)
}
```

Apply the imputation algorithms:

```
imputeMean=function(dataset){
        #Initialise the imputed dataset
        imp_x=dataset
        #For each variable in the dataset, determine if the value is missing and impute it with
        #the mean of the observed values for that variable
        for (i in 1:NCOL(dataset)){
                imp_x[is.na(dataset[,i]),i]=mean(dataset[,i],na.rm=TRUE)
        }
        return(imp_x)
}

imputeEM=function(dataset, seedValue){
        #Do preliminary manipulations
        s=prelim.norm(dataset)
        #Find the MLE parameters
        thetahat = em.norm(s,showits=FALSE)
        #Set random number generator seed
        rngseed(seedValue)
        #Impute missing data under the MLE
        imp_x = imp.norm(s,thetahat,dataset)
        #Return the imputed dataset
        return(imp_x) }
```

```
imputeiPCA=function(dataset, S, seedValue){
	#Set random number generator seed
	rngseed(seedValue)
	#Impute data using iterative PCA
	#Method set to using EM algorithm instead of regularised EM
	#seed=NULL implies initial missing values set to mean of variables
	imp_data=imputePCA(dataset,ncp=S,method="EM",seed=NULL)
	#Return the imputed dataset
	imp_x=imp_data$completeObs
	return(imp_x)
}

imputeMCMC=function(dataset,seedValue){
	#Set random number generator seed otherwise the function repeats the values
	rngseed(seedValue)
	#Create matrix to use in functions
	data_matrix=prelim.norm(dataset)
	#Compute ML estimates using EM-algorithm
	par_hat=em.norm(data_matrix,showits=F)
	#Create new par. taking 10 steps
	new_par=da.norm(data_matrix,start=par_hat,steps=10,showits=F)
	#Impute under new par.
	imp_x=imp.norm(data_matrix,new_par,dataset)
	#Return imputed dataset
	return(imp_x)
}
```

Run the PCA:

```
pcaModel=function(dataset, S){
        #Number of variables
        p=NCOL(dataset)
        #Fit PCA model to dataset
        fit_model= princomp(dataset, cor=TRUE)
        #Get the cumulative proportion of variation accounted for
        cumpVar=cumsum(fit_model$sdev^2/sum(fit_model$sdev^2))
        #Get the eigenvalues and vectors
        eSummary=eigen(cor(dataset))
        eValues=eSummary$values
        #The eigenvectors are only extracted for the first S PC's
        eVecs=eSummary$vectors[1:p,1:S]
        #Extract the first S PC's and calculate the scores:
        #Note: the scores are calculated based on the scaled and centered data matrix
        PCs=with(fit_model, scale(dataset,center=center,scale=scale))%*%eVecs
        #Return the results
        return(list(cumpVar=cumpVar,
                eig=eValues,
                eigvec=eVecs,
                scores=PCs))
}
```

Determine the correlations between the SI imputed PC's and baseline PC's:

```
#Function to determine if imputed PC's swapped around and swapping them around if they did
corrPCs=function(observed,imputed,S){
        #Determine the correlation
        corMatrix=cor(observed,imputed)
        #Initialise
        n=0
```

```
        maxCor=c(0,0,0,0,0,0)
        maxCorInd=c(0,0,0,0,0,0)
        swapInd=0
        #For each imputed PC determine which PC has the max correlation with baseline PC's
        for (i in 1:S){
                maxCorInd[i]=which(abs(corMatrix[,i])==max(abs(corMatrix[,i])))
                maxCor[i]=corMatrix[maxCorInd[i],i]
        }
        #Test if imputed dataset swapped PC's
        if (sum(maxCorInd==c(1,2,3,4,5,6))!=6)  swapInd=1
        #Swap PC's if necessary otherwise keep PC as is
        scoresFinal=cbind(sign(maxCor[1])*imputed[,maxCorInd[1]],
                        sign(maxCor[2])*imputed[,maxCorInd[2]],
                        sign(maxCor[3])*imputed[,maxCorInd[3]],
                        sign(maxCor[4])*imputed[,maxCorInd[4]],
                        sign(maxCor[5])*imputed[,maxCorInd[5]],
                        sign(maxCor[6])*imputed[,maxCorInd[6]])
        return(list(scores=scoresFinal,
                indPCs=maxCorInd,
                swapInd=swapInd))
}
```

Determine the correlations between the MI imputed PC's and baseline PC's:

```
corrMI=function(observed,impdata1,impdata2,impdata3,impdata4,impdata5,S){
        #Create a combined dataset of 5 imputed datasets
        scores_combined=cbind(impdata1,impdata2,impdata3,impdata4,impdata5)
        #Initialise
        maxCorMI=matrix(0,S,4)
        maxCor_PCs=matrix(0,S,4)
        sumMiss=0
```

```
scoresMIave=matrix(0,NROW(observed),S)
scoresMIpca=matrix(0,NROW(observed),S)
MIcumpVar=matrix(0,1,30)
MIeig=matrix(0,1,30)
#Calculate which PC's match the first imputed dataset's PC's
for (j in 1:4){
        cor_PCs=cor(scores_combined[,1:6],scores_combined[,(6*j+1):(6*j+6)])
        for (i in 1:S){
                maxCor_PCs[i,j]=which(abs(cor_PCs[,i])==max(abs(cor_PCs[,i])))
                maxCorMI[i,j]=cor_PCs[maxCor_PCs[i,j],i]
        }
}
for (k in 1:S){
        #Check number of PC's swapped in 5 imputed datasets
        sumMiss=sum(maxCor_PCs[k,]!=k)+sumMiss
        #Create the swapped PC's of 5 imputed datasets
        PCtemp=cbind(impdata1[,k],
                sign(maxCorMI[k,1])*impdata2[,maxCor_PCs[k,1]],
                sign(maxCorMI[k,2])*impdata3[,maxCor_PCs[k,2]],
                sign(maxCorMI[k,3])*impdata4[,maxCor_PCs[k,3]],
                sign(maxCorMI[k,4])*impdata5[,maxCor_PCs[k,4]])
        #Calculate MI Ave by taking average of imputed values
        scoresMIave[,k]=rowMeans(PCtemp)
        #Calculate MI PCA by calculating one component PCA model
        pcaMI=pcaModel(PCtemp,1)
        scoresMIpca[,k]=pcaMI$scores
        #To check if one PC should be extracted
        MIcumpVar[(5*(k-1)+1):(5*k)]=pcaMI$cumpVar
        MIeig[(5*(k-1)+1):(5*k)]=pcaMI$eig
}
```

```
        #Determine the correlation between the combined MI PC's and baseline PC's
        MIaveAdj=corrPCs(observed,scoresMIave,S)
        MIpcaAdj=corrPCs(observed,scoresMIpca,S)
        return(list(swapMI=sumMiss,
                    aveScores=MIaveAdj$scores,
                    aveIndPCs=MIaveAdj$indPCs,
                    aveSwapInd=MIaveAdj$swapInd,
                    MIcumpVar=MIcumpVar,
                    MIeig=MIeig,
                    pcaScores=MIpcaAdj$scores,
                    pcaIndPCs=MIpcaAdj$indPCs,
                    pcaSwapInd=MIpcaAdj$swapInd))
}
```

Evaluation of results:

```
summaryStatsPCs=function(datasetScores){
        #Initialise
        statsPCs=matrix(0,NCOL(datasetScores),2)
        #Get summary stats per PC
        for (i in 1:NCOL(datasetScores)){
                statsPCs[i,1]=mean(datasetScores[,i])
                statsPCs[i,2]=sd(datasetScores[,i])
        }
        return(statsPCs)
}

comparePCs=function(completeScores,imputedScores){
        #Initialise
        results=matrix(0,1,3)
        #Get the RV Coefficient and p-value of complete vs imputed scores
```

```
        results[1]=coeffRV(completeScores,imputedScores)$rv
        results[2]=coeffRV(completeScores,imputedScores)$p.value
        #SSD between complete and imputed scores
        results[3]=sum((completeScores-imputedScores)^2)
        return(results)
}
```