8-2010

Mixture of Factor Analyzers with Information Criteria and the Genetic Algorithm

Esra Turan
eturan@utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Esra Turan entitled "Mixture of Factor Analyzers with Information Criteria and the Genetic Algorithm." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Statistics.

Hamparsum Bozdogan, Major Professor

We have read this dissertation and recommend its acceptance:

Michael W. Berry, Mohammed Mohsin, Russell Zaretzki, Bogdan C. Bichescu

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:
I am submitting herewith a thesis written by Esra Turan entitled "Mixture of Factor Analyzers with Information Criteria and the Genetic Algorithm." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Statistics.

Hamparsum Bozdogan, Major Professor

We have read this thesis
and recommend its acceptance:

Michael W. Berry

Mohammed Mohsin

Russell Zaretzki

Bogdan C. Bichescu

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Mixture of Factor Analyzers with Information Criteria and the Genetic Algorithm

A Thesis Presented for

The Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Esra Turan

August 2010

This dissertation is dedicated to my parents, Saime-Recep Turan. I have learned much about life from them. They have been my role-models for hard work, persistence and instilled in me the inspiration to set high goals and the confidence to achieve them. I strived to become an honorable person for them.

Acknowledgements

Since I came to the University of Tennessee in Knoxville (UTK) during the year of 2007, I received tremendous amount of support from my advisor, Dr. Hamparsum Bozdogan. I would like to take this opportunity to extend my deepest gratitude and thanks to Dr. Bozdogan for his guidance, encouragement, and his support not just during my doctoral thesis research but also during my academic program at UTK. This work would have not been completed without his precious ideas and time. I also would like to thank my doctoral committee members Dr. Michael Berry, Dr. Mohammed Mohsin, Dr. Russell Zaretzki, and Dr. Bogdan Bichescu for their time and support and useful comments.

I express my gratitude to Dr. Andrew Howe and Dr. Eylem Deniz who by their support and their words of advice throughout my doctoral studies. I thanked them for their contributions and special help, and for their friendship. My special thanks go to Martin Taylor who always showed support and encouragement. Most importantly, I wish to express my gratitude and thanks to my family who gave me strength, encouragement, understanding, and love throughout my life and education. I also would like to thank to all my friends in the United States who have encouraged and supported me to complete my dissertation. Finally, I acknowledge the receipt of the Summer Graduate Research Award (GRA) during the summer 2009 which partially supported the research of this thesis at the University of Tennessee in Knoxville.

Abstract

In this dissertation, we have developed and combined several statistical techniques in Bayesian factor analysis (BAYFA) and mixture of factor analyzers (MFA) to overcome the shortcoming of these existing methods. Information Criteria are brought into the context of the BAYFA model as a decision rule for choosing the number of factors m along with the Press and Shigemasu method, Gibbs Sampling and Iterated Conditional Modes deterministic optimization. Because of sensitivity of BAYFA on the prior information of the factor pattern structure, the prior factor pattern structure is learned directly from the given sample observations data adaptively using Sparse Root algorithm.

Clustering and dimensionality reduction have long been considered two of the fundamental problems in unsupervised learning or statistical pattern recognition. In this dissertation, we shall introduce a novel statistical learning technique by focusing our attention on MFA from the perspective of a method for model-based density estimation to cluster the high-dimensional data and at the same time carry out factor analysis to reduce the curse of dimensionality simultaneously in an expert data mining system. The typical EM algorithm can get trapped in one of the many local maxima therefore, it is slow to converge and can never converge to global optima, and highly dependent upon initial values. We extend the EM algorithm proposed by Ghahramani and Hinton (1997) for the MFA using intelligent initialization techniques, K-means and regularized Mahalabonis distance and introduce the new Genetic Expectation

Algorithm (GEM) into MFA in order to overcome the shortcomings of typical EM algorithm. Another shortcoming of EM algorithm for MFA is assuming the variance of the error vector and the number of factors is the same for each mixture. We propose Two Stage GEM algorithm for MFA to relax this constraint and obtain different numbers of factors for each population. In this dissertation, our approach will integrate statistical modeling procedures based on the information criteria as a fitness function to determine the number of mixture clusters and at the same time to choose the number factors that can be extracted from the data.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

As is well known, one of the major difficulties in multivariate analysis is to choose an appropriate model, estimating and determining the dimension of a model. In recent years, the statistical literature has placed more and more emphasis on model selection criteria. The goal of model selection is to find the *best approximating model* among a set of candidate models for given a data set. This dissertation will explore and develop new model selection techniques in modern latent variable modeling. Namely, we shall study the Standard factor analysis (SFA) model, Bayesian factor analysis (BFA) model, and Mixture of factor analyzers (MFA) model using information-theoretic model selection criteria and the genetic algorithm (GA) as our optimization workhorse. Our approach integrates multivariate statistical methods with modern computational tools. Hence, this thesis is an interdisciplinary endeavor. This dissertation is composed of seven chapters.

The Second and Third Chapters address the Standard factor analysis (SFA) and Bayesian factor analysis (BFA) models. Generally, the main purpose of factor analysis as an important multivariate statistical technique is to determine whether or not the correlations among a large number of observed variables can be explained in terms of a relatively small number of factors and what the best number of factors is to fit to a

given dataset. For situations resulting in negative unique variances in the Standard factor analysis model, which is referred to as a *Heywood case*, Bayesian factor analysis can be used. Another advantage of Bayesian factor analysis over the Standard factor analysis model is to be able to incorporate prior information into the model. However, Bayesian factor model often is sensitive to the prior information on the factor pattern structure (Λ_0). Because of this, it is important to intelligently determine the initial prior hyperparameters (especially Λ_0) by eliminating subjective specification of the factor loading structure. Furthermore, we compare the performance of Bayesian factor analysis (BFA) model in large samples which is originally introduced by Press and Shigemasu (1989). For small samples, we use the Gibbs Sampling (GS) and Iterated Conditional modes (ICM) algorithms in BFA model which are developed by Rowe and Press (1998) by introducing the information criteria within such methods to choose the best fitting model.

The Fourth Chapter of this dissertation is about Mixture of factor analyzers (MFA) model. MFA model classifies the high-dimensional data into different clusters and at the same time carries out factor analysis to reduce the dimensionality. MFA in this sense, models the covariance (or the correlation) structure of high dimensional data using a small number of latent variables (i.e., factors). It is because of this, MFA can be considered as an expert data mining technique. In this dissertation, we introduce information criteria within MFA to select the best approximating model for a given dataset to carry out a simultaneous decision in choosing the best number of factors and the number of mixtures to fit the data. Ghahramani and Hinton (1997) used the Expectation and Maximization (EM) algorithm to estimate the parameters of the MFA model. As is well known, the EM algorithm can get trapped in one of many local maxima of the likelihood function without robust starting values. The EM algorithm is too slow to converge and sometimes may never converge to global optima. It is highly dependent upon initial values. Although the typical EM algorithm has these shortcomings, especially on its dependence on the initial values, Ghahramani

and Hinton (1997) use random initialization to start the EM algorithm. In this thesis, we improve the EM algorithm in MFA by using several intelligent initialization schemes. These include: K-means initialization and GA for Regularized Mahalabonis (GARM) distance initialization. In Chapter Five introduce and derive the several forms of the information criteria in Standard factor (SFA), in Bayesian factor (BFA), and the MFA models to be scored in Chapter Seven.

In Chapter six, we introduce a new Genetic Expectation (GEM) algorithm in MFA in order to overcome the strong dependence on initialization of the traditional EM algorithm. Another major issue in MFA model is the assumption that covariance matrix of the random error is the same across the mixture of clusters and that one extracts the same number of factors. In practice, this is not a viable assumption. Therefore, one may ask the question: "Why does the number of factors or the covariances have to be the same for each population?" To be able to answer such an important question, we further propose a new method for MFA model to achieve flexibility in our assumptions in order to be able to obtain different number of factors across mixture of clusters. In this new method, we develop a Two Stage GEM algorithm. In the first stage, we discover the number mixture clusters, and then for each mixture we obtain the best approximating number of factors. In the second stage, we maximize the log likelihood function of the MFA model and using the information criteria we obtain the final number of factors and the covariance matrix of the random errors for each mixture cluster. In Chapter Seven, we provide simulated and many real world data numerical examples of our proposed technique. We show the accuracy of the parameter estimates using GEM in MFA model under a true structure using a simulation protocol. Further, we provide a comparison and the performance of the three initialization schemes to select the best fitting MFA model for simulated and real datasets. The dissertation will conclude with Chapter Eight.

Chapter 2

Standard Factor Analysis

Let $x_1, .., x_n$ denotes a random sample of size n on a p dimensional random vector with mean vector μ and the dispersion matrix Σ. In the Standard factor analysis (SFA) model x is modeled as

$$x = \mu + \Lambda f + \varepsilon, \tag{2.1}$$

where f is a k dimensional ($k < p$) vector of latent or unobservable variables called factors, and Λ is unknown factor loading matrix. The factors f are assumed to be independently and identically distributed as $N(0, I_q)$ where I_q denotes the $q \times q$ identity matrix. The random errors or the disturbance term ε is distributed as $N(0, \Psi)$, where Ψ is a diagonal matrix. According to this model, it can easily be shown that the random vector has a Gaussian distribution with the mean vector μ and the covariance matrix Σ. It is assumed that the factors account for all the correlation structure so that random errors $\varepsilon = x - \mu - \Lambda f$ and $\Sigma = \Lambda\Sigma_f$ are uncorrelated. That is, $\Psi = D[\varepsilon]$ is a diagonal matrix, $diag(\Psi_1^2, \Psi_2^2, ..., \Psi_k^2)$. Therefore,

$$\begin{aligned} \Sigma_x &= D[\Lambda f + \varepsilon] \\ &= D[\Lambda f] + D[\varepsilon] \\ &= \Lambda D[f]\Lambda' + D[\varepsilon] \\ &= \Lambda\Sigma_f\Lambda' + \Psi. \end{aligned} \tag{2.2}$$

When it is assumed that the factors are standardized and uncorrelated, that is, when $\Sigma_f = I_m$, then

$$\Sigma = \Lambda\Lambda' + \Psi. \tag{2.3}$$

Given a random sample of observations $x_1, x_2, ..., x_n$, the goal of the factor analysis is to decide whether Σ can be expressed in the form (2.3) for a reasonably small value of k and to estimate Λ and Ψ for obtaining the best covariance structure of x. Although there is no close form to estimate Λ and Ψ, they can be obtained by using the Expectation-Maximization (EM) algorithm to maximize the factor analytic likelihood function.

The EM algorithm is an iterative algorithm. Each cycle consists of an E step followed by an M step, which increases the likelihood fucntion of the parameters. It's commonly applied for models in which there is missing information. Incomplete data are the values of the latent variables f in the FA. Rubin and Thayer (1982) developed the EM algorithm for FA. In the E step, the expectation of the complete data log-likelihood given the observed data x_i and current estimated values of the parameters are computed. We wish to estimate the parameters Λ and Ψ. The complete data log-likelihood to do this is given by

$$\begin{aligned} l(\Lambda, \Psi) &= \log \prod_i^n p(x_i, f_i|\Lambda, \Psi) \\ &= \sum_i^n \log p(x_i, f_i|\Lambda, \Psi) \\ &= \sum_i^n \log p(x_i|f_i, \Lambda, \Psi) p(f_i|\Lambda, \Psi) \\ &= \sum_i^n \log p(x_i|f_i, \Lambda, \Psi) + \sum_i^n \log p(f_i|\Lambda, \Psi). \end{aligned} \tag{2.4}$$

However, the distribution of x is independent of Λ and Ψ, then the log-likelihood is

$$l(W, \Psi) = \sum_{i}^{n} \log p(x_i|f_i, \Lambda, \Psi) + \sum_{i}^{n} \log p(f_i). \tag{2.5}$$

Since the second term in (2.5) is independent of Λ and Ψ, it suffices (for the purpose of estimating Λ and Ψ) to only deal with the term

$$L = \sum_{i}^{n} \log p(x_i|f_i, \Lambda, \Psi).$$

We can expand L as

$$\begin{aligned} L &= \log \prod_i (2\pi)^{p/2} |\Psi|^{-1/2} \exp\{-\frac{1}{2}[x_i - \Lambda f]' |\Psi|^{-1/2} [x_i - \Lambda f]\} \\ &= c - \frac{n}{2} \log |\Psi| - \sum_i \left[\frac{1}{2} x_i' \Psi^{-1} x_i - x_i' \Psi^{-1} x_i \Lambda f + \frac{1}{2} f' \Lambda' \Psi^{-1} \Lambda z\right]. \end{aligned} \tag{2.6}$$

Taking the expectation of L according to $p(f_i|x_i, \Lambda, \Psi)$, we get

$$E(L) = c - \frac{n}{2} \log |\Psi| \sum_i \left(\frac{1}{2} x_i' \Psi^{-1} x_i - x_i' \Psi^{-1} x_i \Lambda E(z|x_i) + \frac{1}{2} trace[\Lambda' \Psi^{-1} \Lambda E(zz'|x_i)\right), \tag{2.7}$$

where c is a constant and independent of the parameters. The expected value of the factors are computed through linear projection.

Maximizing Equation (2.7) with respect to Λ, we have

$$\begin{aligned} \frac{\partial E(L)}{\partial \Lambda} &= -\sum \Psi^{-1} x_i E[f|x_i]' + \sum \Psi^{-1} \Lambda^{new} E[ff'|x_i] = 0 \\ \Lambda^{new} &= \sum x_i E[f|x_i]' \left(\sum E[ff'|x_i]\right)^{-1}. \end{aligned} \tag{2.8}$$

We can compute the second moment of the factors as follows:

$$\begin{aligned} E(f|x) &= \beta x \\ E(ff'|x) &= Var(f|x) + E(f|x)E(f|x)', \end{aligned}$$

where $\beta = \Lambda'(\Psi + \Lambda\Lambda')^{-1}$. $(\Psi + \Lambda\Lambda')^{-1}$ is a $p \times p$ matrix and can be inverted by using the matrix inversion lemma:

$$(\Psi + \Lambda\Lambda')^{-1} = \Psi^{-1} - \Psi^{-1}\Lambda(I + \Lambda'\Psi^{-1}\Lambda)^{-1}\Lambda'\Psi^{-1}. \tag{2.9}$$

Maximizing Equation (2.7) with respect to Ψ^{-1}, we have

$$\frac{\partial Q}{\partial \Psi^{-1}} = \frac{n}{2}\Psi^{new} - \left(\frac{1}{2}x_i'x_i - \Lambda^{new}E(f|x_i)x_i' + \frac{1}{2}\Lambda^{new}E(ff'|x_i)\Lambda^{new\prime}\right) = 0$$

$$\Psi^{new} = \frac{1}{n}diag\{\sum x_i'x_i - \Lambda^{new}E(f|x_i)x_i'. \tag{2.10}$$

One of the more difficult and delicate tasks of the Standard factor analysis is the selection of m, the number of common factors, based on a finite set of available data. A common approach is to use the scree plot. This plots the eigenvalues of the correlation matrix in descending order. We determine the number of factors equal to the number of eigenvalues that occur prior to the last major drop in eigenvalue magnitude. As a result, this approach involves a certain amount of subjective judgment. Another approach is the *Kaiser criterion*. It states that a number of factors equal to the number of the eigenvalues of the correlation matrix which are greater than 1. But this approach produces large number of factors (Newsom, 2005). Another task of the standard factor analysis model is to choose the method for estimation. There are several ways to estimate the parameters of the Standard factor model. The EM algorithm, which is explained above, is one of them. Based on our simulation applications, the estimation is closest to the real parameters using the EM algorithm.

In this dissertation, we combine information criteria to determine the best number of factors to fit to the dataset with the EM algorithm to estimate the parameters of the model.

Chapter 3

Bayesian Factor Analysis

3.1 Bayesian Factor Model

Consider that we have p variate observation vectors x with mean vector μ and covariance matrix Σ. For a given factor model in (2.1) ϵ_j is a vector of disturbances whose variance Ψ_j represent the uniqueness of x_j. The ϵ_j's are assumed to be mutually uncorrelated and normally distributed as $N(0, \Psi_j)$ for Ψ a symmetric positive matrix $\Psi > 0$. When Ψ is not positive-definite, the solution is said to be *improper* or a *Heywood case*. Sampling errors or an inappropriate factor model might cause this case (Bozdogan and Ramirez, 1986). If communality equals 1 which means that at least one unique variance is zero but the rest are positive, the situation is referred to as an *exact Heywood case*. If communality exceeds 1, the situation is referred to *ultra-Heywood case*. An ultra-Heywood case implies that some unique factor has a negative variance. The SFA model should not be applied in this case. The communalities for the j^{th} variable are computed by taking the sum of the squared loadings for that variable. This is expressed by

$$h_j = \sum_{q=1}^{m} {\lambda_{jq}}^2. \tag{3.1}$$

Martin and McDonald (1975) discuss the use of a Bayesian approach to overcome Heywood cases. They propose finding posterior joint modal estimators of the factor

loading and disturbance covariance matrix. They point out the importance of choosing a reasonable prior distribution for the the disturbance covariance matrix and the use of Jeffrey's type vague prior. Akaike (1987) dealt with the occurrence of improper solutions in the likelihood caused by over parametrization of the model. He approached the standard spherical prior of factor loadings to handle this problem by proper Bayesian modeling. Early works on Bayesian Factor analysis (BFA) include Martin and McDonald (1975) and Lee (1981). A new type of BFA was introduced by Press and Shigemasu (1989) to overcome improper solutions in the maximum likelihood estimation (MLE) methods in FA model in large samples. Their method is easy to apply and no iteration is needed to obtain the point estimates. They developed the method for obtaining large sample interval estimators discussed in Press (1997).

In Bayesian factor analysis, the important issue is how to assess the prior hyperparameters (Λ_0, ν, B, H). Press and Lee (2008) propose an empirical method for assessing the hyperparameters. Since the BFA model is most sensitive to the prior information on the factor pattern structure (Λ_0), it is important to determine the initial prior hyperparameter Λ_0 for the BFA. It is needed to estimate the factor pattern structure by eliminating subjective specification of the factor loading structure. Bozdogan and Shigemasu (1998) applied the Sparse Root algorithm on a training data set to obtain the best approximating factor pattern structure data adaptively. As in the standard factor model, a difficult and delicate task is the selection of the uncertain number of factors in the BFA model. Bozdogan and Shigemasu (1998) applied the information theoretic measure of complexity criterion (ICOMP), to decide on the best fitting number of factors m in the Bayesian factor model. Lopes and West (2004) worked on the same problem of uncertainty of the number of latent factors and combined their research with MCMC methods to estimate the parameters in BFA model. In addition to this, they use AIC, BIC and ICOMP to choose the model and compare their performance with other BFA methods.

Rowe and Press (1998) derived the conditional posterior distribution to use Gibbs Sampling (GS), and Iterated Conditional Modes (ICM) to estimate the parameters for both small and large samples. In the literature, there are several ways to estimate the parameters in the model. Such as Arminger and Muthen (1998) and Ansari and Jedidi (2000) implement Markov Chain Monte Carlo procedures such as Gibbs sampling and the Metropolis- Hastings methods for inference. Consider p variate observation vectors X with mean vector μ and covariance matrix Σ. The Bayesian factor model is written as

$$x_{j(p\times 1)} = \Lambda_{(p\times m)} f_{j(m\times 1)} + \mu + \epsilon_{j(p\times 1)} \quad m < p, \tag{3.2}$$

for $j = 1, \ldots, n$, where Λ denotes a matrix of constants called the factor loading matrix; f_j denotes the scores vector for subject j; $F' = (f_1, \ldots, f_n)$. The ε_j's are assumed to be mutually uncorrelated and normally distributed as $N(0, \Psi)$ for Ψ a symmetric positive definite matrix, i.e $\Psi > 0$. Further, the likelihood function is written as

$$p(X \mid \Lambda, F, \Psi) \propto |\Psi|^{-n/2} e^{-\frac{1}{2} tr \Psi^{-1}(X - F\Lambda')'(X - F\Lambda')}, \tag{3.3}$$

where "$\propto$" denotes the constant of proportionality which depends on (p, n) but not (Λ, F, Ψ). The joint prior density is obtained using the natural conjugate distributions of (Λ, F, Ψ) as priors. So, the joint prior density is

$$p(\Lambda, F, \Psi) = p(\Lambda \mid \Psi) p(\Psi) p(F), \tag{3.4}$$

where

$$\begin{aligned} p(\Lambda \mid \Psi) &\propto |\Psi|^{-m/2} e^{-\frac{1}{2} tr \Psi^{-1}(\Lambda - \Lambda_0) H (\Lambda - \Lambda_0)'} \\ p(\Psi) &\propto |\Psi|^{-v/2} e^{-\frac{1}{2} tr \Psi^{-1} B} \\ p(F) &\propto e^{-\frac{1}{2} tr F'F}, \end{aligned} \tag{3.5}$$

with B a diagonal matrix and $H > 0$, a positive definite matrix. Λ conditional on Ψ is normally distributed, with hyperparameters Λ_0 a $(p \times m)$ prior factor loading matrix and H am a $(m \times m)$ prior inter-factor correlation matrix. Both of them are assessed. $|\Psi|^{-1}$ follows Wishart distribution with hyperparameters ν (degrees of freedom) and B is a $(p \times p)$ scale matrix of hyperparameters. Both of them are assessed. If we consider a vague prior density for the factor scores F: if $p(F) \propto$ constant, then $p(F \mid X)$ follows a *matrix* T distribution. Since we assume $f_j \sim N(0,1)$, and f'_{j_s} are mutually independent, then $p(F)$ is a spherical prior density.

When applying Bayes theorem, we obtain the joint posterior density of the parameters by combining the likelihood function and joint prior density.

$$\begin{aligned} p(\Lambda, F, \Psi \mid X) &\propto p(X \mid \Lambda, F, \Psi) p(\Lambda \mid \Psi) p(\Psi) p(F) \\ &\propto e^{-\frac{1}{2} tr F'F} \left|\Psi\right|^{-\frac{(n+m+\nu)}{2}} e^{-\frac{1}{2} tr \Psi^{-1} G}, \end{aligned} \tag{3.6}$$

where $G = (x_j - F\Lambda')'(x_j - F\Lambda') + (\Lambda - \Lambda_0)H(\Lambda - \Lambda_0)' + B$. The three conditional posterior densities are obtained by removing the fixed parameters from the joint posterior distribution. The conditional posterior distribution of the factor loading vector is

$$\begin{aligned} p(\Lambda \mid F, \Psi, X) &\propto p(\Lambda, F, \Psi \mid X) p(X \mid F, \Lambda, \Psi) \\ &= p(\Lambda \mid \Psi) p(\Psi) p(F) p(X \mid F, \Lambda, \Psi) \\ &\propto \left|\Psi\right|^{-m/2} e^{-\frac{1}{2} tr \Psi^{-1} (\Lambda - \Lambda_0) H (\Lambda - \Lambda_0)'} \left|\Psi\right|^{-n/2} e^{-\frac{1}{2} tr \Psi^{-1} (X - F\Lambda')'(X - F\Lambda')} \\ &\propto e^{-\frac{1}{2} tr \Psi^{-1} [(\Lambda - \Lambda_0) H (\Lambda - \Lambda_0)' + (x_j - F\Lambda')'(x_j - F\Lambda')]}. \end{aligned} \tag{3.7}$$

After some algebra, this can be written as

$$p(\Lambda \mid F, \Psi, X) \propto p(\Lambda, F, \Psi \mid X) p(X \mid F, \Lambda, \Psi) \propto e^{-\frac{1}{2} tr \Psi^{-1} (\Lambda - \widetilde{\Lambda})(H + F'F)(\Lambda - \widetilde{\Lambda})'}, \tag{3.8}$$

where $\widetilde{\Lambda}$ is the mode of the conditional distribution $p(\Lambda \mid F, \Psi, X)$ and it is defined as follows

$$\widetilde{\Lambda} = (X'F + \Lambda_0 H)(H + F'F)^{-1}. \tag{3.9}$$

Then, the conditional posterior distribution of the factor loading matrix given factor scores, the disturbance covariance matrix, and data follows a normal distribution. The conditional posterior distribution of the disturbance covariance matrix is defined as follows

$$\begin{aligned} p(\Psi \mid F, \Lambda, X) &\propto p(\Psi)p(\Lambda \mid \Psi)p(F)p(X \mid F, \Lambda, \Psi) \\ &\propto |\Psi|^{-v/2} e^{-\frac{1}{2}tr\Psi^{-1}B} |\Psi|^{-m/2} e^{-\frac{1}{2}tr\Psi^{-1}(\Lambda-\Lambda_0)H(\Lambda-\Lambda_0)'} |\Psi|^{-n/2} \\ &\quad e^{-\frac{1}{2}tr\Psi^{-1}(X-F\Lambda')'(X-F\Lambda')} \\ &\propto |\Psi|^{-\frac{(n+m+v)}{2}} e^{-\frac{1}{2}tr\Psi^{-1}G}, \end{aligned} \tag{3.10}$$

where

$$G = (X - F\Lambda')'(X - F\Lambda') + (\Lambda - \Lambda_0)H(\Lambda - \Lambda_0)' + B.$$

Then, the conditional posterior distribution of the disturbance covariance matrix given factor scores, factor loadings and data is an inverted Wishart distribution. The mode of the conditional distribution $p(\Psi \mid F, \Lambda, X)$ is

$$\widetilde{\Psi} = \frac{G}{(n + m + v)}. \tag{3.11}$$

The conditional posterior distribution of the factor scores is

$$\begin{aligned} p(F \mid \Lambda, \Psi, X) &\propto p(\Psi)p(\Lambda \mid \Psi)p(F)p(X \mid F, \Lambda, \Psi) \\ &\propto e^{-\frac{1}{2}trF'F} |\Psi|^{-n/2} e^{-\frac{1}{2}tr\Psi^{-1}(X-F\Lambda')'(X-F\Lambda')} \\ &\propto e^{-\frac{1}{2}trF'F} e^{-\frac{1}{2}tr(X-F\Lambda')\Psi^{-1}(X-F\Lambda')'}, \end{aligned} \tag{3.12}$$

which, after some algebra, becomes

$$p(F \mid \Lambda, \Psi, X) \propto e^{-\frac{1}{2}tr(F-\widetilde{F})(I+\Lambda'\Psi^{-1}\Lambda)(F-\widetilde{F})'}, \tag{3.13}$$

where $\widetilde{F}$ is mode of the conditional distribution $p(F \mid \Lambda, \Psi, X)$ and it is defined as

$$\widetilde{F} = X\Psi^{-1}\Lambda(I + \Lambda'\Psi^{-1}\Lambda)^{-1}. \tag{3.14}$$

The conditional posterior distribution of the factor scores given factor loadings, the disturbance covariance matrix and data is normally distributed.

The steps of our approach are summarized as follows:

1. Assess the prior hyperparameters (Λ_0, ν, B, H). Λ_0 is obtained by the Sparse Root algorithm data adaptively. Assessment of the other priors is not influential on the results. We can assess them as: $H = \eta_0 I$ and $B = b_0 I$, for some preassigned scalar η_0 and b_0 to make v is as minimal as possible.

2. Calculate the maximum number of factors that can be extracted for a given dataset by $big\ factor = 2p + 1 - \sqrt{8p+1}$.

3. Apply the method of Press and Shigemasu to find the initial parameters for Gibbs sampling or ICM method.

4. Estimate the parameters by Gibbs sampling or ICM methods for the number of factors $m = 1, 2, .., big\ factor$ in the model.

5. Find the information criteria scores for each factor model.

6. Choose the model corresponding to the minimum information criteria.

3.2 Sparse Root Algorithm

The sparse root algorithm developed by Hartigan (1975) is an iterative clustering procedure which works on a factor model correlation (or covariance) matrix associated with a factor loading matrix. This technique produces the loading matrix and its simple pattern structure as the root of the model correlation matrix. It seeks roots of the model correlation matrix with many zeros. The zeros correspond to entries (or variables) that are not members of a given cluster, the cluster represented by a given column of the root matrix. The root matrix is determined iteratively by sweeping the model correlation matrix one column at a time. This approach is essentially a principal component type procedure that takes linear interdependence of the original variables into account. The columns of the root matrix are chosen to be eigenvectors of the model correlation matrix for which the ratio, eigenvalue/number of nonzero elements of the eigenvector, is a maximum. The model correlation matrix is modified in terms of a partial correlation matrix at each stage of the fitting process until the root, that is, the simple pattern structure of the loading matrix is reached. Thus, the end product of this procedure is a clustering model for the factor loading matrix, so that for each factor there is a set of associated cluster variables. These variables correspond to nonzero loadings on the factor. Such an approach puts zeros in the loading matrix directly from the data, rather than on any substantive grounds which are often biased based on the human "hindsight" of the researcher. Further, this approach discovers a simple factor pattern structure which permits interpretation of the final factors as clusters of variables. Moreover, it permits the researcher to test the postulated specific factor pattern structure based on prior knowledge. In the following, we give a very brief account of the Sparse Root algorithm and its steps following Hartigan (1975) using our notation.

Let R be the factor model correlation matrix, and let R_r be the restricted correlation matrix. We approximate R by $R_r = \widehat{\Lambda}\widehat{\Lambda}'$ where $\widehat{\Lambda}$ contains many zeros. The matrix $\widehat{\Lambda}$ is assessed by two properties.

i. the sum of squares $SS(\widehat{\Lambda}) = \sum_{i=1}^{p}\sum_{l=1}^{m}\Lambda_{i,l}^2$

ii. the number of zeros, $z(\widehat{\Lambda})$ = the number of times $\Lambda_{i,l} = 0$

To be able to apply the method in a stepwise manner during maximization, at each stage of the iterative process, it is necessary to require that the residual matrix be nonnegative definite given by

$$R_{res} = R - Rr \geq 0.$$

The main steps of the Sparse Root algorithm are as follows:

1. Set the column to be estimated, $l = 1 \quad 1 \leq l \leq m$. Initially, set $R_{res} = (r_{i,j})_{res} \equiv R = (r_{i,j}) \quad 1 \leq i,j \leq p$. Set $IP = p$, where p is the number of variables.

2. Let $\{\Lambda_j' = (\Lambda_1,\Lambda_2,\Lambda_3,....\Lambda_p) \quad 1 \leq j \leq p\}$ be the $(1 \times p)$ eigenvector (loadings) corresponding to the largest eigenvalue of the matrix R, where $SS(\widehat{\Lambda}_{i,1}) = \sum_{i=1}^{p}\Lambda_{i,1}^2$ is the first largest eigenvalue. Set

$$f(IP) \equiv f(p) = \frac{\sum_{i=1}^{p}\Lambda_i^2}{count\,(\Lambda_i \neq 0)}.$$

3. Choose the row IMIN to be the row of the matrix R which minimizes the squared correlations

$$r_i^2 = \frac{[\sum_{i=1}^{p}\Lambda_{i,1}r_{imin,i}]^2}{[\sum_{i=1}^{p}\Lambda_{i,1}^2]^2} \quad 1 \leq i \leq p$$

where the numerator is the square of the linear combination of the eigenvectors with the correlations $r_{i\,\min,i}$, and the denominator is the square of the sum of squares of eigenvectors. The minimum of $r_i^2 \quad 1 \leq i \leq p$, is chosen to be the row IMIN to be removed from the matrix R by replacing all correlations by the partial correlation with IMIN fixed and by setting all correlations involving IMIN equal to zero, which destroys R stagewise.

4. Compute the partial correlation matrix of R with IMIN "removed"; that is, change $r_{j,m}$ to $(r_{j,m} - r_{imin,j}r_{imin,m})/r_{imin,imin} \quad 1 \leq j, m \leq p$ and finally, set $r_{imin,j} = r_{j,imin} = 0 \quad 1 \leq j \leq p$. Set $IP = IP - 1 \equiv p - 1$, and if $IP \equiv p$ remains greater than zero, go to Step 2.

5. If $IP \equiv p = IMAX$ maximize $\{f(IP) \equiv f(p) \quad 1 \leq IP \leq p\}$. Set $\Lambda_{i,l} = \tilde{\Lambda}'_j$ $1 \leq i \leq p$, where $\tilde{\Lambda}'_j$ is now the eigenvector corresponding to $f(IP)$. Change $r_{i,j}$ to

$$r_{i,,j} - \Lambda_{i,l}\Lambda_{j,l'} \quad 1 \leq i, j \leq p.$$

Define $R = (r_{i,j}) \equiv R_{res} = r(i,j)_{res}$ increase l by 1 , and go to Step 2, unless $l = m$.

If there are not many zeros in the root factor loading matrix, then this might indicate that there is not sufficient clustering of the variables to create a simple interpretable factor pattern structure.

3.3 Estimation

3.3.1 The method of Press and Shigemasu

The marginal posterior distributions are obtained from joint posterior distribution by integrating out the nuisance parameters sequentially as shown in Press and Shigemasu (1989). Bayes estimates of the unknown parameters are obtained as follows:

- The Bayes estimates of the factor scores is given by

$$\widehat{F} = (I_n - XW^{-1}X)^{-1}XW^{-1}\Lambda_0 H \tag{3.15}$$

or

$$\widehat{F} = (I_n - X(XX' - W)^{-1}X')XW^{-1}\Lambda_0 H,$$

where

$$W = XX' + B + \Lambda_0 H \Lambda_0.$$

- The Bayes estimates of the factor loading matrix Λ conditional on F and X is given by

$$\widehat{\Lambda} = (X'\widehat{F} + \Lambda_0 H)(H + \widehat{F}'\widehat{F})^{-1}. \tag{3.16}$$

- The Bayes estimates of the disturbance matrix Ψ conditional on F, Λ and X is given by

$$\widehat{\Psi} = \frac{G}{n + m + v - 2p - 2}, \tag{3.17}$$

where

$$G = (X - \widehat{F}\widehat{\Lambda}')'(X - \widehat{F}\widehat{\Lambda}') + (\widehat{\Lambda} - \Lambda_0)H(\widehat{\Lambda} - \Lambda_0)' + B,$$

and

$$v = 2(p + 1) + 1.$$

Since we do not have prior knowledge about Ψ, we want to make v the degrees of freedom of Ψ, as minimal as possible. We will take $H = \eta_0 I$ and $B = b_0 I$, for some preassigned scalar η_0 and b_0 as in PS89. Since BFA is most sensitive to Λ_0, Λ_0 will be obtained data adaptively by the Sparse Root algorithm.

3.3.2 Gibbs Sampling

Gibbs sampling is one of the Markov Chain Monte Carlo (MCMC) algorithms. It was introduced by Geman and Geman (1984) in the context of image processing. Gelfand and Smith (1990) helped to demonstrate the value of the Gibbs algorithm in the Bayesian framework. Gibbs sampling strategies are claimed to be fast and sensitive, and avoid getting trapped in local optima. The advantage of this method is of its fast convergence to the joint posterior distribution.

To apply the Gibbs Sampling approach, the estimation is obtained by drawing a random sample from the posterior conditional distribution for each of the parameters which is conditional on the fixed value of all the other parameters and the data X. Let $p(\theta \mid X)$ be the posterior distribution of the parameters when $\theta = (\theta_1, \theta_2, \ldots, \theta_j)$ is the set of parameters and X is the data. We begin with the initial value $\theta^{(0)} = (\theta_1^{(0)}, \theta_2^{(0)}, \ldots, \theta_j^{(0)})$ and $\theta^{(i+1)} = (\theta_1^{(i+1)}, \theta_2^{(i+1)}, \ldots \theta_j^{(i+1)})$ which is defined in the i^{th} iteration:

$$\begin{aligned}
\theta_1^{(i+1)} &= \text{a random sample from } p(\theta_1 \mid \theta_2^{(i)}, \theta_3^{(i)}, \ldots \theta_j^{(i)}, X) \\
\theta_2^{(i+1)} &= \text{a random sample from } p(\theta_2 \mid \theta_1^{(i+1)}, \theta_3^{(i)} \ldots \theta_j^{(i)}, X) \\
\theta_3^{(i+1)} &= \text{a random sample from } p(\theta_3 \mid \theta_1^{(i+1)}, \theta_2^{(i+1)} \ldots \theta_j^{(i)}, X) \\
&\vdots \\
\theta_j^{(i+1)} &= \text{a random sample from } p(\theta_j \mid \theta_1^{(i+1)}, \theta_2^{(i+1)}, \theta_3^{(i+1)} \ldots \theta_{j-1}^{(i+1)}, X).
\end{aligned}$$

A random sample is drawn from the conditional posterior distribution at each step. We will have $\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(s)}, \theta^{(s+1)}, \ldots \theta^{(s+t)}$. The first s random samples are discarded since it is used for convergence. They are called "burn in" samples. The remaining t samples are kept. Posterior means and modal estimators of the parameters are $\theta = (\theta_1, \theta_2, \ldots, \theta_j)$ where

$$\theta_j = \frac{1}{t} \sum_{k=1}^{t} \theta_j^{(s+k)}. \tag{3.18}$$

Gibbs sampling requires an initial starting point for the parameters. The parameters obtained by the method of PS89 are used as initial values for Gibbs sampling. We start with the initial value for $F^{(0)}$ and $\Psi^{(0)}$ to estimate the parameters of the model from the posterior distribution by Gibbs Sampling.

$$\Lambda^{(i+1)} = \text{a random sample from } P(\Lambda \mid F^{(i)}, \Psi^{(i)}, X)$$
$$\Psi^{(i+1)} = \text{a random sample from } P(\Psi \mid F^{(i)}, \Lambda^{(i+1)}, X)$$
$$F^{(i+1)} = \text{a random sample from } P(F \mid \Lambda^{(i+1)}, \Psi^{(i+1)}, X).$$

After the first s random samples are discarded for convergence and the remaining t samples are kept and the means of the random samples are computed as

$$F = \frac{1}{t}\sum_{k=1}^{t} F^{(s+k)}, \quad \Lambda = \tfrac{1}{t}\textstyle\sum_{k=1}^{t} \Lambda^{(s+k)}, \quad \Psi = \frac{1}{t}\sum_{k=1}^{t} \Psi^{(s+k)}. \tag{3.19}$$

They are used as the posterior estimates of the parameters.

3.3.3 Iterated Conditional Modes (ICM)

Iterated Conditional Modes (ICM) introduced by Lindley and Smith (1972), is a deterministic optimization method that finds the joint posterior modal estimators of $p(\theta \mid X)$ when $\theta = (\theta_1, \theta_2, \ldots, \theta_j)$ is the set of the parameters and X is the data. We find the top of the hill of the $p(\theta \mid X)$ by ICM. This means that we converge to a mode or maximum by this method.

We begin with initial value $\widetilde{\theta}^{(0)} = (\widetilde{\theta}_1^{(0)}, \widetilde{\theta}_2^{(0)}, \ldots \widetilde{\theta}_j^{(0)})$ and $\widetilde{\theta}^{(i+1)} = (\widetilde{\theta}_1^{(i+1)}, \widetilde{\theta}_2^{(i+1)}, \ldots \widetilde{\theta}_j^{(i+1)})$ which is defined in i^{th} iteration, and proceed with

$$\begin{aligned}
\widetilde{\theta}_1^{(i+1)} &= \widetilde{\theta}_1(\widetilde{\theta}_2^{(i)}, \widetilde{\theta}_3^{(i)}, ...\widetilde{\theta}_j^{(i)}) \\
\widetilde{\theta}_2^{(i+1)} &= \widetilde{\theta}_2(\widetilde{\theta}_1^{(i+1)}, \widetilde{\theta}_3^{(i)}, ...\widetilde{\theta}_j^{(i)}) \\
&\vdots \\
\widetilde{\theta}_j^{(i+1)} &= \widetilde{\theta}_j(\widetilde{\theta}_1^{(i+1)}, \widetilde{\theta}_2^{(i+1)}, \widetilde{\theta}_3^{(i+1)}, ...\widetilde{\theta}_{j-1}^{(i+1)}),
\end{aligned}$$

and computing the maximum or mode is done at each step. The calculations are continued until convergence is reached. To apply ICM, we need to determine the functions $\widetilde{\theta}_j$ which maximize $p(\theta \mid X)$ with respect to $\widetilde{\theta}_j$, conditional on the fixed values of all the other elements of θ.

ICM requires an initial starting point for the parameters. The parameters are obtained from PS89 method are used to obtain initial values for ICM. We start with initial value for $F^{(0)}$ to estimate joint posterior modal estimator $(\widetilde{\Lambda}, \widetilde{\Psi}, \widetilde{F})$ by ICM.

$$\widetilde{\Lambda}^{(i+1)} = (X'\widetilde{F}^{(i)} + \Lambda_0 H)(H + \widetilde{F}'^{(i)}\widetilde{F}^{(i)})^{-1}$$

$$\widetilde{\Psi}^{(i+1)} = \frac{(X - \widetilde{F}^{(i)}\widetilde{\Lambda}'^{(i+1)})'(X - \widetilde{F}^{(i)}\widetilde{\Lambda}'^{(i+1)}) + (\widetilde{\Lambda}^{(i+1)} - \Lambda_0)H(\widetilde{\Lambda}^{(i+1)} - \Lambda_0)' + B}{(n + m + v)}$$

$$\widetilde{F}^{(i+1)} = X(\Psi^{(i+1)})^{-1}\widetilde{\Lambda}^{(i+1)}(I + \widetilde{\Lambda}'^{(i+1)}(\Psi^{(i+1)})^{-1}\widetilde{\Lambda}^{(i+1)})^{-1} \tag{3.20}$$

They are calculated until convergence is reached.

Chapter 4

Mixture of Factor Analyzers

4.1 Mixture Factor Model

Many cross-disciplinary researchers are faced with two endemic problems: unobserved heterogeneity and measurement error in the data. If the heterogeneity is not treated properly, analysis of the data can be seriously distorted and misleading results would be obtained. Heterogeneity in data sets may be caused in two situations. Heterogeneity arises by several populations which have different covariance structures associated in the first situation. The data is analyzed using regular multiple group covariance structure since each group is identified exactly with a single factor analysis model which causes a different covariance structure model for each group. (Jöreskog, 1971; Lee and Tsui, 1982; Muthén, 1989; Sörborn, 1974). In the second situation, heterogeneity refers to a non-normal distribution which is multi-modal and extremely skewed when the data is treated as a single group (Yung, 1997). If factor analysis is applied for all observations, the heterogeneity problem can be dealt with by using estimation techniques for a non-normal distribution (Kano et al., 1990). However, the above mentioned estimation techniques are not appropriate if the non-normal distribution is due to a sampling of observations from several distinct factor analysis models. The observations are partitioned into different factor analysis models and

the number of groups is the same as the number of factor analysis models in the first situation. There is only one group with possibly many factor analysis models in the second situation. Therefore, the observations need to be empirically classified into the factor analysis models at the same time as the parameters are estimated (Blâfield, 1980).

This dissertation focusses on mixture of factor analyzers (MFA) model from the perspective of a method for model-based density estimation to cluster the high-dimensional data and at the same time carry out factor analysis to reduce the curse of dimensionality. MFA models the covariance (or correlation) structure of high dimensional data using a small number of latent variables (factors), simultaneously in an expert data mining system. This approach results in a model which takes into account the unobserved heterogeneity which affects many statistical modeling procedures. Specifically, we are correcting for this and measurement error in the data concurrently by clustering the data and reducing the dimensionality. MFA model is a globally nonlinear latent variable model obtained by combining the standard sub-factor analysis models for the distributions with ideas from the analysis of mixture of distributions. Let X_i be a p-dimensional observed vector which comes from a mixture of M-factor analysis models. The marginal distribution of X given by

$$f(x) = f(x; \pi, \mu, \Sigma) = \sum_{k=1}^{M} \pi_k g_k(x; \mu_k, \Sigma_k) \tag{4.1}$$

with

$$g_k(x; \mu_k, \Sigma_k) \sim N_p(\mu_k, \Sigma_k = \Lambda_k \Lambda_k' + \Psi_k). \tag{4.2}$$

The k-factor model hold for each observation X_i with $\pi_k$is modeled as

$$X_i = \mu_k + \Lambda_k z_{ik} + \varepsilon_{ik}, \tag{4.3}$$

for $i = 1, 2, ..., n$, $k = 1, 2, ..., M$, where $\mu_k \epsilon R^p$ and $\Lambda_k \epsilon R^{p \times q}$ are unknown parameters, π_m is the mixing proportion; such that $\sum_{m=1}^{M} \pi_m = 1$, z_{ik} is a q_k dimensional matrix of latent or unobservable variables called *common factors*, $z_{ik} \sim N(0, I_{qk})$, ε_{ik} is the independent distribution vector for all i and k, $\varepsilon_{ik} \sim N(0, \Psi_k), \Psi_k = diag(\sigma_{k1}^2, \sigma_{k2}^2 ..., \sigma_{kp}^2)$ and z_{ik} and ε_{ik} are independent. The dimension of the common factors z_{ik} may be different for each subgroup as in model (4.3), but we assume that $q_1 = q_2 = ... = q_m = q$ in the model (4.3) (Ghahramani and Hinton, 1997; Fokouè, 2005; Yung, 1997; Zhou and Liu, 2008). That is the number of factors is the same for each subgroup. In this case, the model becomes

$$X_i = \mu_k + \Lambda_k z_i + \varepsilon_{ik} \quad i = 1, 2, ..., n, k = 1, 2, ..., M. \tag{4.4}$$

In addition, for all practical purposes, Ψ_k is taken to be the same for each subgroup $m = 1, 2, ...M$ (Ghahramani and Hinton, 1997; MacLachan and Peel, 2000; Fokouè, 2005; Cho and Zhang, 2002; MacLachlan et al., 2002) which reduces the number of parameter and for numerical reasons.

4.2 EM Algorithm for MFA Model

In a mixture of M factor analyzers indexed as $w_k, k = 1, 2, ...M$ we consider the following distribution

$$p(x) = \sum_{k=1}^{M} \int P(x|z, w_k) P(z|w_k) P(w_k) dz, \tag{4.5}$$

where $p(x|z, w_k)$ is a single factor model and is distributed as $N(\Lambda_k z + \mu_k, \Psi)$. The parameters $\mu_k, \Lambda_k, \pi_k, \Psi$ can be estimated using the EM algorithm similar to FA. The vector π parameterizes the adaptable mixing proportions $\pi_k = P(w_k)$. As in standard factor analysis, the factors are assumed to be $N(0, I)$, so $P(z|w_k) = P(z) = N(0, I)$. In this case, there is missing information in addition to the values of factors. That is,

we cannot know which model is responsible for generating each data point x_i. The indicator variables $w_k = 1$ indicates when the data point was generated by w_k. The expected log-likelihood function to be maximized is written as follows

$$E(\theta) = E\left[\ln \prod_{i=1}^{n}\prod_{k=1}^{M}\left\{(2\pi)^{-p/2}\left|\Psi\right|^{-1/2}\exp(-\frac{1}{2}(x_i-\mu_k-\Lambda_k z)'\Psi_k^{-1}(x_i-\mu_k-\Lambda_k z))\right\}^{w_j}\right]. \tag{4.6}$$

To jointly estimate of the mean vector μ_k and the factor loadings matrixΛ_k can be written as

$$\widetilde{z} = \begin{bmatrix} z \\ 1 \end{bmatrix}, \widetilde{\Lambda}_k = [\ \Lambda_k \quad \mu_k\ ].$$

Therefore, the expected likelihood function becomes

$$E(\theta) = E\left[\ln \prod_{i=1}^{n}\prod_{k=1}^{M}\left\{(2\pi)^{-p/2}\left|\Psi\right|^{-1/2}\exp(-\frac{1}{2}(x_i-\widetilde{\Lambda}_k\widetilde{z})'\Psi_k^{-1}(x_i-\widetilde{\Lambda}_k\widetilde{z}))\right\}^{w_j}\right]. \tag{4.7}$$

The E-step for general mixture model involves estimating the contribution of component w_k for each data point given by

$$\begin{aligned} h_k(x_i) &= E\{w_k|x_i\} = \frac{\pi_k g_k(x_i|w_k)}{\sum_{k=1}^{M}\pi_k g_k(x_i|w_k)} \\ &= \frac{\left|\Sigma_k\right|^{-1/2}\exp(-\frac{1}{2}(x_i-\mu_k)'\Sigma_k^{-1}(x_i-\mu_k))}{\sum_{k=1}^{M}\left|\Sigma_k\right|^{-1/2}\exp(-\frac{1}{2}(x_i-\mu_k)'\Sigma_k^{-1}(x_i-\mu_k))} \end{aligned} \tag{4.8}$$

which is the posterior probability of group membership. Each factor analyzer fits a Gaussian model to a portion of the data, weighted by the posterior probabilities, h_{ik}.

Therefore, the expected log-likelihood is then

$$\begin{aligned} E(\theta) &= c - \frac{n}{2}\log|\Psi| - \sum_{i=1}^{n}\sum_{k=1}^{M} h_{ik}x_i'\Psi^{-1}x_i - h_{ik}x_i'\Psi^{-1}x_i\widetilde{\Lambda}_j E(\widetilde{z}|x_i,w_k) \\ &+ \frac{1}{2}h_{ik}tr\left[\widetilde{\Lambda}_j'\Psi^{-1}\widetilde{\Lambda}_j E(\widetilde{z}\widetilde{z}'|x_i,w_k)\right], \end{aligned} \tag{4.9}$$

where

$$E(\widetilde{z}|x_i,w_k) = \begin{bmatrix} E(z|x_i,w_k) \\ 1 \end{bmatrix}, \text{ and } E(\widetilde{z}\widetilde{z}'|x_i,w_k) = \begin{bmatrix} E(zz'|x_i,w_k) & E(z|x_i,w_k) \\ E(z|x_i,w_k) & 1 \end{bmatrix}.$$

The posterior can be found by observing that the joint of the latent and observed variables for component k is also Gaussian distribution:

$$\begin{bmatrix} z \\ x \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ \mu_k \end{bmatrix}, \begin{bmatrix} I & \Lambda_k' \\ \Lambda_k & \Lambda_k\Lambda_k' + \Psi \end{bmatrix}\right).$$

The posterior can be determined from this joint distribution and is also Gaussian.

$$p_k(z|x_i,w_k) \sim N(\Lambda_k'(\Psi + \Lambda_k\Lambda_k')^{-1}(x_i - \mu_j), I - \Lambda_k'(\Psi + \Lambda_k\Lambda_k')^{-1}\Lambda_k).$$

We can obtain the expected value of factors as:

$$\begin{aligned} E\{z|x_i,w_k\} &= \beta_k(x_i - \mu_j), \\ E\{zz'|x_i,w_k\} &= I - \beta_k\Lambda_k + \beta_k(x_i - \mu_j)(x_i - \mu_j)'\beta_k', \end{aligned} \tag{4.10}$$

where $\beta_k = \Lambda_k'(\Psi + \Lambda_k\Lambda_k')^{-1}$. In the M-step, we are re-estimating the parameters Λ_k, μ_k, Ψ and Π_k. Maximizing (4.9) with respect to $\widetilde{\Lambda}_k$, we have

$$\begin{aligned}
\frac{\partial E(\theta)}{\partial \widetilde{\Lambda}_k} &= -\sum_{i=1}^{n} h_{ik}\Psi^{-1}x_i E(\widetilde{z}|x_i, w_k) + h_{ik}x_i'\Psi^{-1}\widetilde{\Lambda}_k^{new} E(\widetilde{z}\widetilde{z}'|x_i, w_k) = 0 \\
\widetilde{\Lambda}_k^{new} &= \begin{bmatrix} \Lambda_k^{new} & \mu_k^{new} \end{bmatrix} \\
&= \left(\sum_{i=1} h_{ik}x_i' E(\widetilde{z}|x_i, w_k)'\right)\left(\sum_{i=1} h_{ik}E(\widetilde{z}\widetilde{z}'|x_i, w_k)\right)^{-1}. \qquad (4.11)
\end{aligned}$$

Maximizing the equation (4.9) with respect to Ψ^{-1}, we obtain

$$\begin{aligned}
\frac{\partial E(\theta)}{\partial \Psi^{-1}} &= \frac{n}{2}\Psi^{new} - \sum_{i=1}^{n}\sum_{k=1}^{M}\frac{1}{2}h_{ik}x_i x_i' - h_{ik}\widetilde{\Lambda}_k^{new}E(\widetilde{z}|x_i, w_k)x_i' \\
&+ \frac{1}{2}h_{ik}\widetilde{\Lambda}_k^{new}E(\widetilde{z}\widetilde{z}'|x_i, w_k)\widetilde{\Lambda}_k^{new\prime} = 0. \qquad (4.12)
\end{aligned}$$

Solving this for Ψ^{new}, we have:

$$\Psi^{new} = \frac{1}{n}diag\left\{\sum_{i=1}^{n}\sum_{k=1}^{M} h_{ik}(x_i - \widetilde{\Lambda}_k^{new}E(\widetilde{z}|x_i, w_k))x_i'\right\}. \qquad (4.13)$$

The updates of the mixing proportions are then,

$$\pi_k^{new} = \frac{1}{n}\sum_{i=1}^{n} h_{ik}. \qquad (4.14)$$

4.3 Initialization Schemes

The EM algorithm can get trapped in one of many local maxima of the likelihood function without robust starting values since the log-likelihood parameter space is very rugged (Xu and Jordan, 1996; Vlassis and Likas, 2002). The initialization of the parameters of the log-likelihood function plays a very important role in the final solution. In this research, we introduce the intelligent initialization scheme

to be used in the EM algorithm, along with the random initialization scheme used by Ghahramani and Hinton (1997). We introduce K-means and genetic regularized Mahalobonis distance ($GARM$) initialization in the MFA model to initialize the EM algorithm. We assign each datapoint to a mixture cluster with these initialization tools and then apply the EM algorithm of the Standard factor model on each mixture to obtain the initial values of the EM algorithm in MFA model. After giving more details about the genetic algorithm(GA) and its steps, $GARM$ will be explained more detail in Chapter 6 separately.

4.3.1 K-Means initialization

In the literature, there are many ways to initialize the clustering methods. K-means is one of the simplest unsupervised learning algorithms that initializes clustering methods.K-Means in itself is used also as a clustering method. To initialize the EM algorithm in the mixture model, k clusters are found by the K-means algorithm. Then one can obtain the initial parameters of the model by applying the Standard factor model to each cluster. MacQueen (1967) introduced the K-means algorithm to assign each data point into the cluster with the closest mean. The algorithm is comprised of the following four steps:

1. Determine the initial k cluster centroids.

2. Determine the Euclidean distance of each datapoint to the centroids

$$e_i(k) = (x_1 - \widehat{\mu}_k)(x_1 - \widehat{\mu}_k)'. \tag{4.15}$$

3. Assign each datapoint to the closest cluster which by minimizing the Euclidean distance. $\widehat{y}_i = k$ such that $e_i(k) = \min_{k=1,...,\widehat{K}} e_i(k)$.

4. After all datapoints have been assigned to a cluster, recalculate and update the mean of the clusters. $\widehat{\mu}_k = \frac{\sum_{i=1}^{n} x_i I_k(\widehat{y}_i)}{\sum_{i=1}^{n} I_k(\widehat{y}_i)} \quad k = 1, ..., \widehat{K}$.

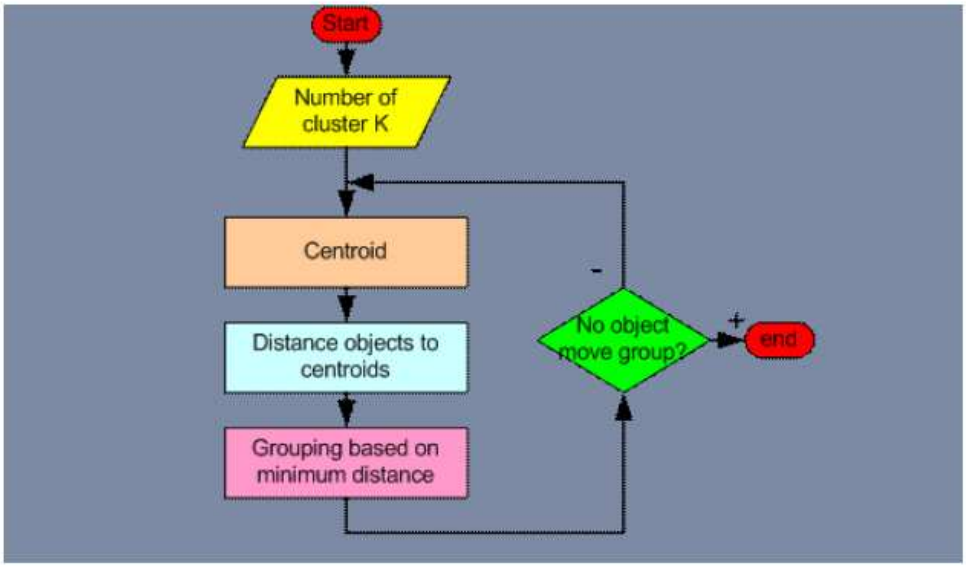


Figure 4.1: K-Means Flow Chart.

5. Repeat Steps 2 and 4 until $|E^{j+1} - E^{j}|$ meets some criterion, where

$$E^{(j+1)} = \sum_{i=1}^{n} \left[\sum_{k=1}^{\widehat{K}} I_k(\widehat{y}_i^{(j+1)}) e_i(k) \right]. \tag{4.16}$$

The flow chart of K-means is drawn in Figure 4.1 (Teknomo, 2007).

4.3.2 Hybridized Scheme

The main idea here is to define the centroids, one for each cluster. These centroids need to be placed intelligently because of different location causes different result. Bozdogan (1983) introduces a new the initialization scheme to choose the centroids of the cluster data-adaptively. In this dissertation, we used this initialization scheme is MFA model. A brief account of Bozdogan (1983) initialization scheme is as follows.

- Calculate the highest (x_n) and lowest (x_1) order statistics for given dataset. The initial centroids are defined using the midpoints for each cluster.
- $\mu_1 = \overline{x}_{11} = \frac{x_n + x_1}{2}$ is an initial centroid for $\widehat{k} = 1$.

- Centroids are $\mu_1 = \overline{x}_{21} = \frac{x_1 + \overline{x}_{11}}{2}$ and $\mu_2 = \overline{x}_{22} = \frac{\overline{x}_{11} - x_n}{2}$ for $\widehat{k} = 2$ clusters.
- For fitting $\widehat{k} = 3$ clusters, $\mu_1 = \overline{x}_{31} = \frac{x_1 + \overline{x}_{21}}{2}$ $\mu_2 = \overline{x}_{32} = \frac{\overline{x}_{21} + \overline{x}_{22}}{2}$ and $\mu_3 = \overline{x}_{33} = \frac{\overline{x}_{22} + x_n}{2}$.

This scheme is applied in a similar fashion for higher $\widehat{k}$.

As an example, we simulate the population of a multivariate normal data with the following parameters to show the working of the Bozdogan's hybridized initialization scheme with the clusters. Figure 4.2 shows the data with each population identified.

$$\mu_1 = \begin{bmatrix} 18 & 20 \end{bmatrix} \mu_2 = \begin{bmatrix} 10 & 12 \end{bmatrix}, \sigma_1 = \begin{bmatrix} 4 & 3 \\ 3 & 7 \end{bmatrix} \sigma_2 = \begin{bmatrix} 7 & 5 \\ 5 & 10 \end{bmatrix}.$$

The confusion matrix from the K-means algorithm is shown in Table 4.1. As can be seen in the Table 4.1, 13 observations from cluster one assigned to cluster two, and one observation from cluster two is assigned to cluster one. Hence, in this case the misclassification error is $13 + 1/300 = 4.67\%$.

Table 4.1: Confusion Matrix of K-means Algorithm

Actual/Predicted	1	2	
1	137	13	150
2	1	149	150
	138	162	300

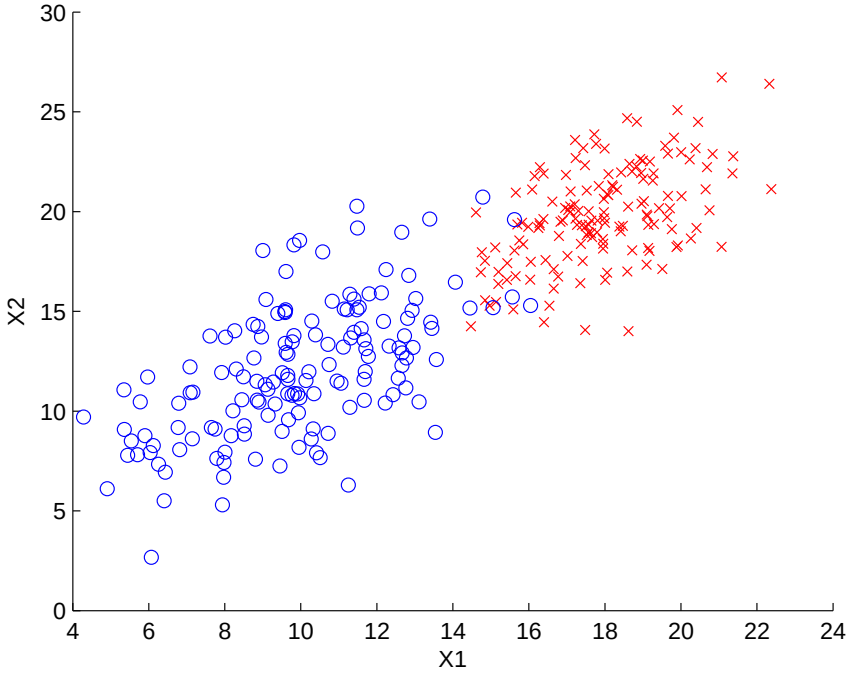


Figure 4.2: Scatter Plot of the Simulated Data.

Chapter 5

Information Criteria

5.1 Introduction and Purpose

It is well known, that the fundamental difficulty in statistical analysis is, estimating and determining the dimension of a *best fitting* model. This is a common problem when a statistical model contains many parameters. The main purpose of model selection is to find the best approximating model fits that the observed data. In recent years, in the literature, the necessity of introducing the concept of model selection or model evaluation has been recognized and the problem is posed how to find the *best approximating* model among a class of competing models with different numbers of parameters using a suitable model selection criterion.

Also, there is a great deal of interest in criteria based on the parsimony of parameters for choosing one model from a set of competing alternative models to describe a given data set. Parsimony can take into account a variety of attributes in the selected model. One such attribute is the measurement cost required to implement the model. A second attribute is the complexity of the selected model. Therefore, the best model is the one with the least complexity or equivalently the highest information gain. For example, in the factor model, parameter parsimony requires that we choose

the smallest number of factors such that the corresponding model fits the data.

In this Chapter, we detail a method to arbitrate among the results and help us choose the best model. In our case, the best number of factors in the SFA model,BFA model and the best number of clusters and the number of factors in MFA model. This is where information criteria come into the picture - the best model for the data is that which minimizes the information criterion (IC) function. The rational of introducing information criteria is that they provide an easy to use solutions to complex problems and avoid the difficulties in the usual hypothesis testing type procedures such as the likelihood ratio principle in such problems studied in this thesis. In what follows, we present the forms of the information criteria based on the work of Bozdogan (1987, 1988, 1990, 1994, 2000, 2004); Deniz and Bozdogan (2010)

5.2 Kullback-Liebler Distance

For a given dataset, the best model is one which balances a good fit to the data and the desired parsimony for the model. As model complexity increases, the goodness-of-fit must increase at least as much. Otherwise, the additional complexity is not worth the cost. Cost could refer to the actual cost of gathering additional data (or, the variables), but here we mostly refer to the cost of additional parameter and estimation uncertainty. Virtually all information criteria penalize a poorly-fitting model with negative twice the maximized log-likelihood, as an asymptotic estimate of the Kullback-Liebler (KL) Information. The fundamental basis for all information criteria is the KL information, first introduced by Kullback and Leibler (1951). The KL information or distance measures the difference between two probability distributions. Let us assume that θ^* is the true parameter vector of θ with its probability density function $f(X|\theta^*)$. Let $I(\theta^*|\theta)$ denotes the KL distance between the true model and fitted model. Then since the observations x_i for $i = 1, 2, ..., n$ are

independent, we have

$$\begin{aligned} I(\theta^*;\theta) &= E_{\theta^*}\left[\log\frac{f(x|\theta^*)}{f(x|\theta)}\right] = E\left[f(x|\theta^*) - f(x|\theta)\right] = \int \log\frac{f(x|\theta^*)}{f(x|\theta)} f(x|\theta^*)dx \\ &= \int f(x|\theta^*)\log f(x|\theta^*)dx - \int f(x|\theta^*)\log f(x|\theta)dx \\ &= H(\theta^*;\theta^*) - H(\theta^*;\theta), \end{aligned} \tag{5.1}$$

where E denotes the expectation operator with respect to the true distribution $f(x|\theta^*)$ of x. $H(\theta^*;\theta^*) = H(\theta^*)$ is the entropy which is a constant and can be dropped. $H(\theta^*;\theta)$ is the cross-entropy which determines the goodness of fit of $f(x|\theta)$ to $f(x|\theta^*)$. Therefore, we only have to estimate the second term, which is cross entropy or expected log-likelihood given by

$$-H(\theta^*;\theta) = -E[log f(X|\theta)] = -\sum_i^n E[log f_i(x_i|\theta)], \tag{5.2}$$

which can be estimated by

$$-\sum_i^n log f_i(x_i|\theta) = -logL(\theta|X). \tag{5.3}$$

In (5.3), $logL(\theta|X)$ is the log likelihood function of θ given the observations. In practice, we would estimate the parameter vector, typically using the MLE $\widehat{\theta}$ of θ, and so we use the maximized log likelihood function to approximate in (5.2) given by

$$-\sum_i^n log f_i(x_i|\widehat{\theta}) = -logL(\widehat{\theta}|X). \tag{5.4}$$

Thus, when there are competing models for a dataset, selecting the model with the highest maximized likelihood (or lowest negative maximized likelihood) should provide a model nearest to the true data generating process. All the information criteria use this approximation for the KL distance from the true model to penalize a poorly-fitting model. The difference then, is in the penalty for model complexity.

5.3 Akaike's Information Criterion AIC

Akaike (1973) developed the information-theoretic criterion AIC for the identification of an optimal a parsimonious model to choose for in data analysis from a class of competing alternative models. Let M_k for $k = 1, 2, ..., K$ be a set of competing model indexed by $k = 1, 2, .., K$. Then the criterion

$$AIC = -2logL(\widehat{\theta}|X) + 2k, \tag{5.5}$$

which is minimized to choose a model M_k over the set of models is a natural sample estimator of twice negentropy $2E[I(\theta^*; \theta_k)]$, or minus twice the expected log likelihood, $-2E[logf(x|\theta_k)]$ of the true distribution with respect to a model with parameters determined by the method of maximum likelihood Bozdogan (1987).

5.4 Schwarz's Bayesian Criterion SBC

Schwarz (1978) proposed another criterion using Bayesian framework to choose the best fitting model to the observed data. SBC is defined by

$$SBC = -2logL(\widehat{\theta}|X) + klog(n), \tag{5.6}$$

where k is the number of parameters, and $log(n)$ denotes the natural logarithm of the sample size n. SBC introplaces a heavier penalty term, then does AIC. Therefore, it should work well in large samples.

5.5 Consistent Akaike's information Criterion $CAIC$

Bozdogan (1987) developed CAIC to obtain stronger penalty term instead of the debatable magic number 2 in AIC, which has been questioned unfairly as being arbitrary. Bozdogan (1987) proposed to make AIC consistent by making the

multiplier of the free parameter in the penalty term depend on the sample size n. Therefore, $CAIC$ penalizes overly-complex models with the penalty term. It is defined by

$$CAIC = -2logL(\widehat{\theta}|X) + k[log(n) + 1] \tag{5.7}$$

The penalty of $CAIC$ is similar to that of SBC. But it has the added number of parameters term in addition to $klog(n)$. This gives us mush stringent penalty term, which penalizes overparametrized models more than AIC, and SBC.

5.6 Information Complexity ICOMP Criterion

The approach we take in this research is based on cost functions which measure the goodness of fit, or the performance, of a fitted model for a given dataset. The risk, that is, the expected cost of choosing the best fitting model, will be measured in terms of an entropic or information-based criterion which is based on a different characterization of good models by combining penalties with the lack-of-fit, lack-of-parsimony, and the profusion of complexity. Bozdogan (1988, 1990, 1994, 2004) developed information theoretic ideas of a measure of "overall" model complexity in statistical modeling to help provide new approaches relevant to statistical inference. The information complexity index ICOMP measures the fit between multivariate structural models and observed data as an example of the application of the covariance complexity measure. Van Emden (1971) provides a reasonable definition of informational complexity of a covariance matrix Σ, denoted by $C_0(\Sigma)$, under the multivariate normal distribution assumption. Bozdogan (1988) proposed to use the maximal amount of complexity of the covariance matrix Σ that is an upper bound $C_0(\Sigma)$ measure. A maximal information-theoretic measure of complexity of a covariance matrix of a multivariate normal distribution is then defined as

$$C_1(\Sigma) = \frac{s}{2}\log\left[\frac{tr(\Sigma)}{s}\right] - \frac{1}{2}\log|\Sigma|, \tag{5.8}$$

where $s = dim(\Sigma) = rank(\Sigma)$. Bozdogan (1988) further introduced $C_{1F}(\Sigma)$ by relating $C_0(\Sigma)$ to the Frobenious norm characterization of complexity $C_F(\Sigma)$ of Σ is defined by

$$\begin{aligned} C_{1F}(\Sigma) &= \frac{1}{4\overline{\lambda_a}^2}\sum_{j=1}^{s}(\lambda_j - \overline{\lambda}_a)^2 \\ &= \frac{s}{4}\left[\frac{\frac{tr(\Sigma)'\Sigma)}{s} - \frac{tr(\Sigma)}{s^2}}{\frac{tr(\Sigma)}{s^2}}\right], \end{aligned} \tag{5.9}$$

where λ_j denotes the eigenvalues of Σ,and $\overline{\lambda_a}$ denotes the arithmetic mean of the eigenvalues. Note that $C_{1F}(\cdot)$ is *scale invariant* and $C_{1F}(\cdot) \geq 0$ with $C_{1F}(\cdot) = 0$ only when $\lambda_j = \overline{\lambda}$ for $j = 1, ...p$. Also $C_{1F}(\cdot)$ measures the relative variation in the eigenvalues, rather than the absolute values of the eigenvalues. For a general model, in terms of a loss function,

$$LOSS = \text{Lack of fit} + \text{Lack of Parsimony} + \text{Profusion of Complexity},$$

$ICOMP$ is defined by

$$ICOMP = -2\log L(\widehat{\theta}) + 2C_1(Cov(\widehat{\theta})), \tag{5.10}$$

where $C_1(\cdot)$ measures the complexity of $Cov(\widehat{\theta}) = \Sigma(\widehat{\theta})$. That is, $\widehat{\theta} \sim N(\theta^*, \Sigma(\widehat{\theta}) = \widehat{\mathcal{F}}^{-1})$ where $\widehat{\mathcal{F}}^{-1}$ is the inverse of the estimated Fisher information matrix. We did not use this version of $ICOMP$ in the mixtures of factor analyzers model since the MFA model is more complex-overparametrized model which needs a much heavier penalization.

A very useful form of ICOMP can also be derived under the Bayesian framework by maximizing a posterior expected utility (PEU), as shown in Bozdogan and Haughton

(1998). $ICOMP_{PEU}$ enforces a stricter penalty and is defined as

$$ICOMP_{PEU} = -2\log(\widehat{\theta}|X) + k + 2C_1(\widehat{\mathcal{F}}^{-1}) \tag{5.11}$$

If we have a more complex-multivariate model, we would like to use stronger penalty to choose a parsimonious the true model. Consistent information complexity, $ICOMP_C$, was developed by Bozdogan (2010), and use in Deniz and Bozdogan (2010) is defined as

$$ICOMP_C = -2\log(\widehat{\theta}|X) + 2C_1(\widehat{\mathcal{F}}^{-1}) + k + 2k\log(n). \tag{5.12}$$

We provide a few highlights of the proof and justification from Bozdogan (2010).We consider a composite utility $U = U_1 \times U_2$. Let

$$U_1 = KL\left(f_{Post}\left(\theta \mid X\right), f_{Prior}\left(\theta \mid M_k\right)\right)$$

to be a *utility function* (Lindley, 1956; Poskitt, 1987), which relates to the the goodness-of-fit of the model, and define a second utility function given by

$$U_2 = \exp\left[-log(n)tr(\hat{\mathcal{F}}^{-1}\hat{\mathcal{R}}) - C_1(\hat{\mathcal{F}}^{-1})\right],$$

where $\hat{\mathcal{F}}$ and $\hat{\mathcal{R}}$ are the two forms of the Fisher information matrices. $\hat{\mathcal{F}}$ is the inner-product (or Hessian) form, and $\hat{\mathcal{R}}$ is the outer-product form. This second utility U_2 relates to the lack of parsimony and the profusion of complexity of the model. Therefore, $logU = logU_1 + logU_2$. For a given model M_k of dimension k, we can consider the KL distance between the posterior and prior densities given by

$$KL\left(f_{Post}\left(\theta \mid X\right), f_{Prior}\left(\theta \mid M_k\right)\right) = -\frac{k}{2}\log\left(2\pi\right) - \frac{k}{2} - \frac{1}{2}\log|\hat{\mathcal{F}}^{-1}| - \log f_{Prior}\left(\theta \mid M_k\right). \tag{5.13}$$

Now setting in (5.13), $logU_1 = KL$ and

$$logU_2 = -log(n)tr(\hat{\mathcal{F}}^{-1}\hat{\mathcal{R}}) - C_1(\hat{\mathcal{F}}^{-1}), \tag{5.14}$$

so $logU$ becomes

$$-\frac{k}{2}\log(2\pi)-\frac{k}{2}-\frac{1}{2}\log|\hat{\mathcal{F}}^{-1}|-\log f_{Prior}(\theta \mid M_k)-log(n)tr(\hat{\mathcal{F}}^{-1}\hat{\mathcal{R}})-C_1(\hat{\mathcal{F}}^{-1}). \quad (5.15)$$

The posterior expected utility can be approximated by

$$\log(PEU) \cong \log f(X \mid \hat{\theta}) + \frac{k}{2}\log(2\pi) + \frac{1}{2}\log|\hat{\mathcal{F}}^{-1}| + \log(U) + \log f_{Prior}\left(\hat{\theta} \mid M_k\right), \quad (5.16)$$

simplifying (5.16), we thus obtain a criterion to be maximized to choose a model

$$\log f(X \mid \hat{\theta}) - \frac{k}{2} - log(n)tr(\hat{\mathcal{F}}^{-1}\hat{\mathcal{R}}) - C_1(\hat{\mathcal{F}}^{-1}) + \log f(M_k). \quad (5.17)$$

Further, we note that if the model is correctly specified, $\hat{\mathcal{F}}$ and $\hat{\mathcal{R}}$ would be equal to one another. That is, if $\hat{\mathcal{F}} = \hat{\mathcal{R}}$, then $tr(\hat{\mathcal{F}}^{-1}\hat{\mathcal{R}}) = tr(I_k) = k$. In this case, we have the consistent $ICOMP$ given by

$$ICOMP(\hat{\mathcal{F}}^{-1})_C = -2\log L(\hat{\theta}|X) + k + 2klog(n) + 2C_1(\hat{\mathcal{F}}^{-1}). \quad (5.18)$$

5.7 Information Criteria for the Standard Factor Model

For a given standard k-factor model, information criteria are used to choose the number of factor in the Standard factor (SFA) model. Criteria are minimized for the best fitting model for a given dataset among k different factor models. The covariance structure of the SFA model is defined by

$$\Sigma_k = \Lambda_k\Lambda_k' + \Psi_k, \quad (5.19)$$

where $k = 1, 2, ...,$ big factor for a sample of n observations, the likelihood function is

$$\begin{aligned} L(\mu, \Lambda_k, \Psi_k) &= (2\pi)^{-np/2} \left|\Lambda_k \Lambda_k' + \Psi_k\right|^{-n/2} \\ &\quad \exp\left\{-\frac{1}{2} tr(\Lambda_k \Lambda_k' + \Psi_k)[\sum_{i=1}^{n}(x_i - \mu)(x_i - \mu)']\right\}. \end{aligned} \tag{5.20}$$

We maximize the likelihood function by

$$\begin{aligned} max L(\mu, \Lambda_k, \Psi_k) &= L(\widehat{\mu}, \widehat{\Lambda}_k, \widehat{\Psi}_k) = (2\pi)^{-np/2} \left|\widehat{\Lambda}_k \widehat{\Lambda}_k' + \widehat{\Psi}_k\right|^{-n/2} \\ &\quad \exp\left\{-\frac{1}{2} tr(\widehat{\Lambda}_k \widehat{\Lambda}_k' + \widehat{\Psi}_k) nS\right\}, \end{aligned} \tag{5.21}$$

where S is the sample covariance matrix defined as $S = \frac{X'X}{n}$ and that $A = nS = X'X$, sum of squares and cross product matrix.(Akaike, 1987). The log-likelihood function is

$$\log L(\widehat{\mu}, \widehat{\Lambda}_k, \widehat{\Sigma}_k) = -\frac{np}{2}\log(2\pi) - \frac{n}{2}\log\left|\widehat{\Sigma}_k\right| - \frac{1}{2} tr\widehat{\Sigma}_k nS. \tag{5.22}$$

The lack of fit term for the SFA model given by

$$-2\log L(\widehat{\mu}, \widehat{\Lambda}_k, \widehat{\Psi}_k) = n\left[\log(2\pi) + \log\left|\widehat{\Sigma}\right| + tr(\widehat{\Sigma}_k S)\right]. \tag{5.23}$$

The number of free parameters in the SFA model is computed by $s^* = kp + k - \frac{1}{2}k(k-1)$. Accordingly, the derived forms of AIC, $CAIC$, SBC, $ICOMP_C$ and $ICOMP_{PEU}$ to choose the number of factors in the SFA model in this dissertation are given as follows.

$$\begin{aligned} AIC &= n\left[plog(2\pi) + log|\widehat{\Sigma}_k| + tr(\widehat{\Sigma}_k^{-1} S)\right] + 2s^*, \\ CAIC &= n\left[plog(2\pi) + log|\widehat{\Sigma}_k| + tr(\widehat{\Sigma}_k^{-1} S)\right] + s^*[log(n) + 1], \\ SBC &= n\left[plog(2\pi) + log|\widehat{\Sigma}_k| + tr(\widehat{\Sigma}_k^{-1} S)\right] + s^* log(n). \end{aligned} \tag{5.24}$$

Similary the information complexity ($ICOMP$) criterion is given as follows.

$$ICOMP_C = n\left[plog(2\pi) + log|\widehat{\Sigma}_k| + tr(\widehat{\Sigma}_k^{-1}S)\right] + 2C_1(\widehat{\mathcal{F}}_k^{-1}) + s^* + 2s^*log(n), \quad (5.25)$$

where

$$\widehat{\mathcal{F}}_k^{-1}(\widehat{\pi}) = \begin{bmatrix} \widehat{\Sigma}_k & 0 \\ 0 & \frac{2}{n}D_p^+(\widehat{\Sigma}_k \otimes \widehat{\Sigma}_k)D_p^{+\prime} \end{bmatrix}, \quad (5.26)$$

and where

$$C_1(\Sigma) = \frac{r}{2}\log\left[\frac{tr(\widehat{\mathcal{F}}_k^{-1})}{r}\right] - \frac{1}{2}\log\left|\widehat{\mathcal{F}}_k^{-1}\right|. \quad (5.27)$$

In (5.27), $r = rank(\widehat{\mathcal{F}}_k^{-1})$ For more detail on the above, we refer the reader to Bozdogan (2010).

5.8 Information Criteria for the Bayesian Factor Model

As with the SFA model information criteria are used also to choose the number of factors in the Bayesian factor (BFA) model. Criteria are minimized to choose the best fitting BFA model for a given dataset. The covariance structure of the BFA model for the k^{th} factor is given by

$$\Sigma_k = \Lambda_k H \Lambda_k' + \Psi_k, \quad (5.28)$$

where $k = 1, 2...$big factor. When H, inter factor correlation matrix equals I_k, we obtain the orthogonal factor model. The number of free parameters $s^* = (kp + p) - 1/2k(k-1)$ in the model is less than the number of parameters in the covariance matrix.

Corresponding to the covariance structure in (5.28), the lack of fit term in the BFA model is given by

$$-2\log L(\widehat{\mu},\widehat{\Lambda}_k,\widehat{\Psi}_k) = n\left[p\log(2\pi)+\log|\widehat{\Sigma}_k|+tr(\widehat{\Sigma}_k^{-1}S)\right], \tag{5.29}$$

where S is the observed covariance matrix, $S=\frac{X'X}{n}$ and $\widehat{\Sigma}_k$ is the Bayes estimator of the covariance matrix obtained from fully Bayesian estimation results discussed in Section 3.3.1, 3.3.2, 3.3.3. Therefore, the derived forms of the information criteria at the posterior estimation level are given as follows.

- Akaike's information criterion

$$\begin{aligned} AIC &= n\left[plog(2\pi)+log(|\widehat{\Sigma}_k|)+tr(\widehat{\Sigma}_k^{-1}S)\right] \\ &\quad +2(\text{number of nonzero loadings}+1/2k(k+1)+p). \end{aligned}$$

- Consistent AIC of Bozdogan (1987)

$$\begin{aligned} CAIC &= n\left[plog(2\pi)+log(|\widehat{\Sigma}_k|)+tr(\widehat{\Sigma}_k^{-1}S)\right] \\ &\quad +(\log(n)+1)(\text{number of nonzero loadings}+1/2k(k+1)+p). \end{aligned}$$

- Finite sample version of Bozdogan (1988)

$$ICOMP = n\left[plog(2\pi)+log(|\widehat{\Sigma}_k|)+tr(\widehat{\Sigma}_k^{-1}S)\right]+2[(m+1)C_1(\widehat{\Psi}_k)+pC_1(\widehat{H}_k^{-1})].$$

5.9 Information Criteria for the Mixture Factor Model

In the mixture of factor analyzers (MFA) model, the major problem confronted by the researchers and the partitioners is the selection of the optimal number of clusters

present in a given dataset and at the same time to reduce the curse of dimensionality on the factors. This results in choosing also the number of factors to be extracted in the MFA model. Hence one must deal with both of these problems in a simulation fashion. This is not a trivial problem to deal with without using and developing information-theoretic model selection criteria. To use the usual likelihood ratio type criterion in the MFA model is an impossible problem to deal with since the problem here is not hypothesis testing problem. Furthermore, one does not know how to implement the likelihood ratio criterion in choosing the number of mixtures and at the same time to choose the number of factors in the MFA model. Therefore, in this dissertation, for the first time we develop and introduce information criteria to choose the number of mixtures and also the number of factors for a given dataset simultaneously. In other words, we learn the number of clusters resent and at the same time reduce the dimensionality of the dataset. This approach gives us a practical modeling approach in a complex MFA structure.

In the MFA model, the AIC criterion penalizes model complexity with four times the number of estimated parameters since we need a heavier penalty. From Bozdogan (1994), AIC thus becomes

$$AIC = -2logL(\widehat{\theta}|X) + 4s^*. \tag{5.30}$$

where $-2logL(\widehat{\theta}|X)$ is twice the log of the maximized likelihood if the MFA model and s^* is the number of free parameters in the MFA model. Note that for a mixture of m factor models indexed by $m = 1, 2, ..., M$, there is a different covariance structure in each mixture cluster. That is, $\Sigma_1 = \Lambda_1{\Lambda_1}' + \psi, ..., \Sigma_m = \Lambda_m{\Lambda_m}' + \psi$ for $m = 1, 2, ..., M$. The number of free parameters is then given by

$$s^* = mp + (m-1) + m(pk + p - 0.5k(k-1)), \tag{5.31}$$

where k is the number of factors, and m is the number of mixtures and p=number of variables, or the dimension of the data.

Similarly, SBC for the MFA model is

$$SBC = -2logL(\widehat{\theta}|X) + s^* log(n). \tag{5.32}$$

Consistent AIC ($CAIC$) is

$$CAIC = -2logL(\widehat{\theta}|X) + s^*[log(n) + 1]. \tag{5.33}$$

We provide other forms of $ICOMP$ criterion of Bozdogan (2010) from his furthering book as follows. These are modifications of the original criterion $ICOMP$ which guard the research to the presence of skewness and kurtosis in the data and against non-Gaussianity.

$$\begin{aligned}
ICOMP_C &= -2lnL(\widehat{\theta}) + 2C_{1F}(\widehat{\Sigma}) + s^* + 2s^* \log(n) \\
ICOMP_{CMISS} &= -2logL(\widehat{\theta}|X) + 2C_{1F}(\mathcal{F}_{MFA}^{-1}) + s^* + 2\log(n)\frac{ns^*}{n - s^* - 2} \\
ICOMP_{PEULN} &= -2logL(\widehat{\theta}|X) + \log(n)C_{1F}(\mathcal{F}_{MFA}^{-1}) + s^* \\
ICOMP_{PEUMISS} &= -2logL(\widehat{\theta}|X) + 2C_{1F}(\mathcal{F}_{MFA}^{-1}) + s^* + 2\frac{ns^*}{n - s^* - 2} \\
ICOMP_{PEULNMISS} &= -2logL(\widehat{\theta}|X) + \log(n)C_{1F}(\mathcal{F}_{MFA}^{-1}) + s^* + 2\frac{ns^*}{n - s^* - 2},
\end{aligned} \tag{5.34}$$

where C_{1F} is given in (5.9). For a mixture of m factor analyzers, we can define the estimated covariance matrix, $Cov(\widehat{\theta}) = \widehat{\mathcal{F}}_{MFA}^{-1}$ as a block diagonal matrix which combines the diagonal matrix of the mixing proportion and the estimated inverse

Fisher information matrix of each mixture is given by

$$\widehat{\mathcal{F}}_{MFA}^{-1} = \begin{bmatrix} \widehat{\mathcal{F}}^{-1}(\widehat{\pi}) & 0 & 0 & \dots & 0 \\ 0 & \widehat{\mathcal{F}}_1^{-1} & 0 & \dots & 0 \\ 0 & 0 & \widehat{\mathcal{F}}_2^{-1} & \dots & 0 \\ \vdots & 0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \widehat{\mathcal{F}}_m^{-1} \end{bmatrix}, \tag{5.35}$$

where

$$\widehat{\mathcal{F}}^{-1}(\widehat{\pi}) = \begin{bmatrix} \frac{1}{\widehat{\pi}_1} & 0 & \dots & 0 \\ 0 & \frac{1}{\widehat{\pi}_2} & \dots & 0 \\ \vdots & 0 & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{\widehat{\pi}_m} \end{bmatrix}, \text{ and}$$

$$\widehat{\mathcal{F}}_m^{-1}(\widehat{\pi}) = \begin{bmatrix} \widehat{\Sigma}_m & 0_{(p\times(p+1)/2)} \\ 0_{((p+1)/2\times p)} & \frac{2}{n_m} D_p^+ (\widehat{\Sigma}_m \otimes \widehat{\Sigma}_m) D_p^{+\prime} \end{bmatrix}.$$

The rational of showing these different forms is based on the fact that one needs different penalty functions depending upon the complexity of the datasets. We score all these criteria, but report only ones which give us parsimonious and reasonable solutions to the real data sets we utilized in this dissertation to achieve the Occan's Razor in the model fitting process.

5.9.1 Regularized Covariance Matrix

In cluster analysis, and MFA models often number of observations can be less than number of variables. That is $n < p$. In such a case, we have the ill-conditioned and non-positive definite covariance matrices. Therefore, it becomes difficult to estimate the covariance matrix Σ. The inverse of Σ may not exist and any estimator of the covariance matrix becomes unreliable. This lends itself to serious computational problems in the analysis, and model fitting process. To resolve such this problems, in this section we introduce robust estimators of Σ for the MFA model. We regularize

(shrink) $\widehat{\Sigma}$ in the hopes of achieving a robust estimator. Most regularize estimators take the from the form of the naive ridge regularization given by

$$\Sigma_{reg} = [\widehat{\Sigma} + \alpha I_p], \tag{5.36}$$

where α is called naive ridge parameter, $0 < \alpha < 1$, to be determined. This works to counteract the instability in $\widehat{\Sigma}$ by adjusting the eigenvalues of $\widehat{\Sigma}$. There are many different robust covariance estimators have been developed as a way to data adaptively improve ill-conditioned and/or singular covariance matrix estimates. Several of them use ridge regularization with a different alpha given in (5.36). In this dissertation, we use three different forms of smooth covariances. These are: the Maximum Likelihood/Empirical Bayes (MLE/EB), Stipulate Diagonal (SD) and Convex-Sum (CS), to regularize the covariance matrices in MFA model.

When α is taken $(p-1)/\left[ntr(\Sigma^{-1})\right]$ in the (5.36), then the MLE/EB regularized covariance matrix is defined by

$$\widehat{\Sigma}_{MLE/EB} = \widehat{\Sigma}_{MLE} + \frac{(p-1)}{ntr(\widehat{\Sigma}^{-1}_{MLE})} I_p. \tag{5.37}$$

When α is taken $p(p-1)[2ntr(\widehat{\Sigma})]^{-1}$ (5.36), then we have the Stipulate Diagonal Smooth Covariance defined by

$$\widehat{\Sigma}_{SD} = \widehat{\Sigma} + p(p-1)[2ntr(\widehat{\Sigma})]^{-1} I_p. \tag{5.38}$$

Finally, the Convex-Sum Estimator (CSE) Covariance is defined by

$$\widehat{\Sigma}_{CS} = \frac{n}{n+m}\widehat{\Sigma} + \left(1 - \frac{n}{n+m}\right)\left[\frac{tr(\widehat{\Sigma})}{p}\right] I_p, \tag{5.39}$$

where $m = \left[p(1 + \frac{tr(\widehat{\Sigma})^2}{tr(\widehat{\Sigma}^2)} - 2)\right] / \left[p - \frac{tr(\widehat{\Sigma})^2}{tr(\widehat{\Sigma}^2)}\right]$.

In our computations, we do not always replace MLE covariance estimator with the regularized covariance estimator unless we faced with ill-conditioned ($\kappa(\widehat{\Sigma}^{-1}) < 1e^{-10}$) or non-positive definite covariance matrix estimators. For instability of the covariance matrix, we use Thomaz stabilization (Thomaz, 2004) defined by

$$\widehat{\Sigma}_{Thomaz} = V \begin{bmatrix} \max(\lambda_1, \overline{\lambda}) & 0 & \cdots & 0 \\ 0 & \max(\lambda_2, \overline{\lambda}) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \max(\lambda_p, \overline{\lambda}) \end{bmatrix} V', \tag{5.40}$$

where V is the eigenvectors matrix of $\widehat{\Sigma}$.

Example: We illustrate the regularization of the covariance matrix using the smoothed covariance estimators and the stabilization on the wine dataset given in Appendix A for the MFA model. Table 5.1 through 5.6 show the model selection results using regularized covariance matrix with/without stabilization in the MFA model. As can be seen from the results, the number of mixtures and the number of factors chosen for this dataset are the same with/without stabilization method. Note that in Table 5.2 $ICOMP_{CMISS}$ score only with smoothed MLE/EB covariance matrix is the minimum at $\widehat{m} = 2$ mixtures and $\widehat{k} = 6$ factor model. Therefore, in our later analysis of the wine dataset we use MLE/EB covariance estimator.

Table 5.1: $ICOMP_{CMISS}$ scores using stabilization and smoothed MLE/EB Covariance matrix.

M \ K	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	27966	28202	28455	28740	29043	29365	29699	30035
2	31427	31808	34543	37832	52363	**-80302**	8353	16335
3	36580	46694	5292	24289	26224	2755	28067	28432
4	49022	26915	31176	34335	33045	32291	36241	36903

Table 5.2: $ICOMP_{CMISS}$ scores using only smoothed MLE/EB Covariance matrix.

M \ K	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	13146	13092	13228	13326	13609	13892	14212	14543
2	25110	27845	29477	33217	47553	**-85767**	3321	11310
3	28939	41221	5851	23685	26801	26946	27852	27808
4	47131	18781	29420	33197	38835	42868	47168	49673

Table 5.3: $ICOMP_{CMISS}$ scores using stabilization and Stipulate Diagonal Covariance matrix.

M \ K	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	27967	28204	28462	28739	9043	29366	29699	30037
2	31427	31807	33753	37832	52363	**-80296**	8350.1	163497
3	35228	43648	5695.1	25208	27758	29281	30608	30074
4	514287	30972	28628	32591	31395	312127	33359	32442

Table 5.4: $ICOMP_{CMISS}$ scores using only Stipulate Diagonal Covariance matrix.

M \ K	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	13147	13093	3229	13326	13610	13893	14212	14543
2	25102	27844	29092	33218	47731	**-85736**	3059.4	11335
3	38798	47176	283.83	23402	27443	23283	21935	28233
4	49801	31265	31964	34539	33729	38416	36075	36164

Table 5.5: $ICOMP_{CMISS}$ scores using stabilization and Convex-Sum Covariance matrix.

M \ K	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	27911	28147	28403	28682	28987	29309	29643	29978
2	31091	31459	33396	37941	52520	**-80773**	8479	16359
3	33414	47402	1279	22221	26369	25552	27761	26690
4	50249	22412	30156	32571	39051	31201	34821	39106

Table 5.6: $ICOMP_{CMISS}$ scores using only Convex-Sum Covariance matrix.

M \ K	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	22319	22540	22797	23070	23375	23697	24029	24364
2	32623	33939	37473	42566	57007	**-76171**	13027	20758
3	39339	48872	10433	35570	31071	32298	36337	37424
4	57913	39800	35753	50600	55595	50547	54476	60711

Chapter 6

Genetic Algorithm

6.1 Overview of Genetic Algorithm

In the 1950s and 1960s, computer scientists began research on an evolutionary system as a optimization tool. The idea was to evaluate a large population of potential solutions using the operators inspired by natural selection to obtain an optimal solution. The idea of evolutionary computing was introduced in the 1960s by I. Rechenberg in his work "*Evolution strategies*" (Evolutions strategies in original). His idea was then developed by other researchers. Genetic Algorithms (GAs) were further developed by John Holland and his students and colleagues in the 1960s. Holland presented the GA as an abstraction of biologic evaluation and gave a theoretical framework for adaptation under the GA in his 1975 book "*Adaption in Natural and Artificial Systems.*", see e.g., Holland (1975). His article in Scientific American (Holland, 1992) contributed further to GA's popularity.

The GA is a stochastic or probabilistic search algorithm that employs natural selection and genetic operators. A GA treats information as a series of codes on a string, where each string represents a different solution to a given problem. The GA works by moving from one population of chromosomes to a new population by using

concepts of natural selection embodied in with genetic operators, such as crossover, mutation and inversion. Genetic algorithms are less susceptible to getting stuck at local optima than gradient search methods. One advantage of the GA approach is that it is easy to incorporate arbitrary kinds of constraints and objectives as weighted components of the fitness function, making it easy to adapt the GA to the particular requirements of a very wide range of possible problems.

6.1.1 Basic Terminology

In the typical GA, the *chromosome* is a binary string, having two possible values: 0 and 1. Each position in the string is a *gene*. Each chromosome is a point in the search space of candidate solutions. A set of solutions (represented by chromosomes) of a generation is a *population*. The first generation of the GA process is usually generated as a set of wildly guessed or randomly generated solutions. Each iteration, which is called a *generation*, has P solutions in GA. More generations mean more computation time. However, not allowing the process to go through enough iterations can mean termination with a suboptimal result. The GA operating on a population of chromosomes from current population to generate a pair of new solutions, are called *offsprings*. The genetic algorithm requires a *Fitness function* that assigns a score to each chromosome based on its ability to solve the problem under consideration. At each iteration of the GA process, each chromosome in the current population is ranked according to their fitness score.

6.1.2 GA Operators

Genetic algorithms proceed with three types of operators: selection, crossover and mutation.

Selection: This operator selects chromosomes in the population to reproduce. In the original GA of Holland, the chance of a chromosome being selected was

that chromosome's fitness divided by the average fitness of the population. In the *tournament selection* method two chromosomes are randomly selected from the current population but the one with higher fitness value is added into the mating pool. This procedure is stopped, when the desired number of chromosomes is selected. A benefit of this method is that much computation time is saved due to not computing fitness values on the entire population. However, it is possible that the best solutions would never be evaluated for, since chromosomes are selected randomly with this method. Another selection operator is *Linear ranking selection*, in which all chromosomes in the current population are ranked according to their fitness value. The probability of selecting chromosome i for the replacement step is $p_i = \frac{2(n-j)}{n(n-1)}$ where n is the population size and j is the position of the chromosome i in he ranking (Alba and Dorronsoro, 2008).

There are still other selection methods, one example is *roulette wheel sampling*, which probabilistically selects chromosomes based on their fitness (Goldberg, 1989). Chromosomes are mapped one-to-one into the interval $[0, S]$, where S is the sum of the fitness values over all chromosomes in the current population. Each chromosome is assigned a slice of a circular roulette wheel, the size of the slice being proportional to the chromosome's fitness. To select the chromosome, a random number is generated in the interval $[0, S]$, and the chromosome whose slice spans the random number is selected (Chipperfield, 1997). Our selection operator is akin to using roulette wheel selection. We firstly compute the bin width as follows

$$b = \frac{2}{(P(P+1)}, \quad b \in [0, 1], \tag{6.1}$$

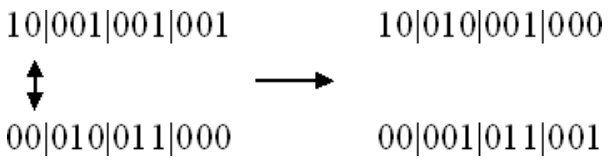
then bin limits (B^{low},B^{upp}) are computed for each chromosome. To select the chromosome, a random number is generated in the interval [B^{low},B^{upp}] and the chromosome whose slice spans the random number is selected.

Crossover: This operator randomly chooses a crossover point, and exchanges the estimated group label before and after that locus between two chromosomes to

create offsprings. The crossover probability is defined to be the probability that two chromosomes are chosen to crossover. The crossover probability is denoted by p_c, and typically it is in the (0.6, 0.9). If no crossover takes place, then the original chromosomes are duplicated. If the crossover probability is 1, then all the offsprings crossover. There are three types of crossover corresponding to different locations of the cross point: *single point*, *multiple point*, and *uniform*crossover.

Random single point crossover is used in this research.An integer is selected randomly between 2 and L (chromosome length), since we would like to switch labels for more than a single observation between chromosomes. For example, "|" shows a crossover point in the following.

For multipoint crossover, we select m crossover positions in the interval [2, L] randomly with no duplication. Then the values between crossover points are exchanged between two chromosomes to produce two new offsprings.



Every value is a potential crossover point for uniform crossover. A chromosome is generated randomly and its parity of the bits indicates which chromosome will supply the offspring with which bits. This generated chromosome is called mask. As an example, P1 and P2 are current chromosomes and O1 and O2 offsprings are generated from P1 and P2 according to the mask. The first offspring O1 is produced by taking the bit from P1 if corresponding mask bit is 1, or the bit from P2 if the corresponding mask bit is 0. The second offspring O2 is generated by swapping P1 and P2.

P1:	1 0 1 1 0 0 0 1 1 1
P2:	0 0 0 1 1 1 1 0 0 0
Mask:	0 0 1 1 0 0 1 1 0 0
O1:	0 0 1 1 1 1 0 1 0 0
O2:	1 0 0 1 0 0 1 0 1 1

Mutation: Mutation produces changes in gene sequences by randomly changing a value. Chromosomes are selected with a certain probability, then for each selected chromosome, the positions are chosen with the same probability of mutation ($\leq 10\%$) to mutate. The higher the mutation probability, the smaller is the danger of premature convergence. This operator has a big role in the GA because a population of chromosomes could quickly become homogenous and get stuck in a local optimum without mutation.

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	0	1	0	1	1	0	0	1	0	1	1	0	0	0	1

6.1.3 Steps of a Simple Genetic Algorithm

The outline of the GA procedures for model parameter estimation and model selection is summarized as follows Mitchell (1998):

1. Create an initial generation with a population of p chromosomes.
2. Rank each chromosome in the population according to the given fitness function.
3. Repeat the following steps until a new population has been created.
   (a) Perform crossover on selected chromosomes with the given method and crossover probability and create the new population.
   (b) Perform mutation on the new population with the given mutation probability.

4. Use the elitism rule if required. Elitism means that the best chromosome in the current population is guaranteed to be included in the new population.

5. Replace the current population with the new population.

6. Repeat step 2-5 until certain converge conditions are satisfied.

6.2 Genetic Algorithm for Regularized Mahalanobis Distance

We use an estimation technique to show how the mixture of factor analyzers (MFA) model can be fitted efficiently using an extension of the EM with an intelligent initialization algorithm proposed in this dissertation. We purpose to use the Genetic Algorithm for Regularized Mahalabonis Distance ($GARM$) to initialize the EM algorithm of mixture of factor analyzers. In clustering, as is well known, the Euclidean distance is used to compute the distance between the assigned cluster centers. When between-cluster variation is much larger than the within-cluster variation, any reasonable clustering method will be able to detect the clusters regardless of the cluster shape. Clustering algorithms with the Euclidian distance have an undesirable tendency to split large and elongated clusters Mao and Jain (1996). In fact, they found that many clusters have neither larger variation between clusters than within clusters nor the spherical shape. Because of this, they used the Mahalanobis distance given in (6.2) to fit hyperellipsoidal clusters. This measurement takes into account the covariance (or correlation) distance is defined by among the variables when computing statistical distances.

$$m_i(k) = (x_i - \hat{\mu}_k)' \hat{\Sigma}_k^{-1} (x_i - \hat{\mu}_k) \tag{6.2}$$

where $\hat{\mu}_k$ is the estimated mean vector of cluster k, and $\hat{\Sigma}_k^{-1}$ is the inverse of the covariance matrix of cluster k, and x_i is the observation vector. Mao and Jain (1996) proposed a *regularized Mahalanobis distance* in (6.3) to recover from the numerical

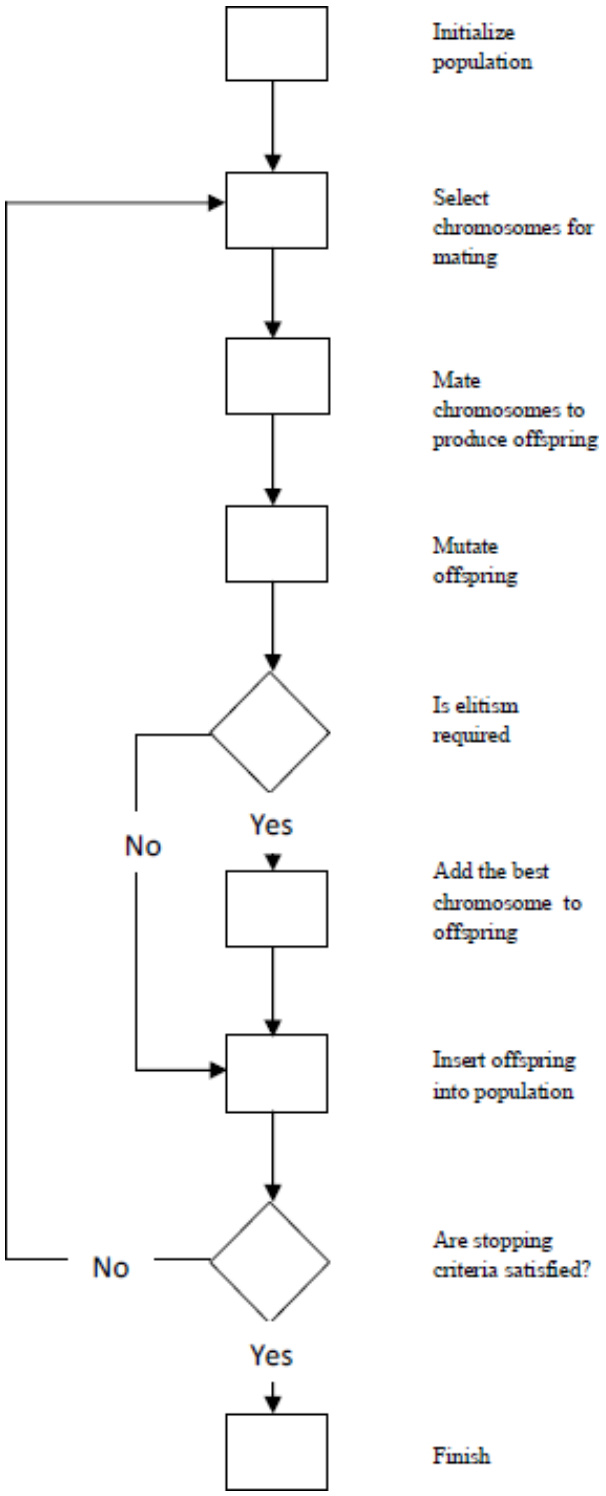


Figure 6.1: A flow chart of the genetic algorithm.

problem such as singularity of the sample covariance matrices of clusters, and producing unusually large or unusually small clusters.The regularized Mahalanobis distance is defined by

$$m_i\left(k\right) = \left(x_i - \hat{\mu}_k\right)' \hat{\Sigma}_k^* \left(x_i - \hat{\mu}_k\right), \tag{6.3}$$

where

$$\hat{\Sigma}_k^* = \left[\left(1-\lambda\right)\left(\hat{\Sigma}_k + \varepsilon I\right)^{-1} + \lambda I\right],$$

and where $0 < \lambda < 1$ is a regularization parameter along with ε. Their values give different shaped and oriented clusters. Therefore, the big issue is how to select λ and ε, especially λ has a big role in the stabilization the process. Song and Shaowei (1997) proposed a scaled Mahalanobis distance given in (6.4). The scale parameter c is constrained to be positive, and they suggest that $c = 1$ is typically sufficient.

$$m_i\left(k\right) = |\widehat{\Sigma_k}|^c \left(x_i - \hat{\mu}_k\right)\left(\widehat{\Sigma}_k\right)^{-1}\left(x_i - \hat{\mu}_k\right)'. \tag{6.4}$$

Recently, Howe (2009) used the complexity of the covariance matrix in calculating the regularized Mahalanobis distance idea that was introduced by Bozdogan. The advantages of this is that it prevents us to choose λ and ε subjectively.It permits also the use of the estimators to regularize the estimated covariance matrix. Therefore, we no longer have to choose a value of scale parameter c arbitrarily. In this manner, the complexity of the covariance structure is taken into account, i.e., both the determinant and the trace in the complexity measure. In this dissertation, we use this approach to regularize the Mahalanobis distance in MFA model. The complexity regularized Mahalanobis distance is given by

$$m_i\left(k\right) = C_1(\hat{\Sigma}_k^*)\left(x_i - \hat{\mu}_k\right)'\left(\hat{\Sigma}_k^*\right)^{-1}\left(x_i - \hat{\mu}_k\right), \tag{6.5}$$

where

$$C_1(\hat{\Sigma}_k^*) = \frac{rank(\hat{\Sigma}_k^*)}{2}\log\left(\frac{tr(\hat{\Sigma}_k^*)}{rank(\hat{\Sigma}_k^*)}\right) - \frac{1}{2}\log(|\hat{\Sigma}_k^*|).$$

Krishna and Murty (1999) applied their distance measure with the GA which is called GARM. When P_p is the p^{th} member of the current population, the fitness function used in GARM is defined as

$$f(P_p) = \frac{a(P_p)}{\sum_{p=1}^{P} a(P_p)}, \tag{6.6}$$

where

$$\begin{aligned} a(P_p) &= \frac{1}{1 + M(P_p) - \min_{1 \le p \le P} M(P_p)} \\ M(P_p) &= \sum_{i=1}^{n} \left[\sum_{k=1}^{\widehat{K}} m_i(k) \right]. \end{aligned}$$

Wicker (2006) extended GARM by implementing a biased mutation operator and a special operator called the *Mahalanobis operator* given in (6.7) below. He used the regularized Mahalanobis distance of Song and Shaowei (1997) in his results. For each selected chromosome in the new population, we uniformly select each elements to mutate with given the mutation probability. Looping through the selected datapoints, the regularized Mahalanobis distance is computed and stored. Mutation operator is defined as

$$M_i(k) = \frac{\max(m_i(k)) - m_i(k)}{\sum_{k=1}^{K} [\max(m_i(k)) - m_i(k)]}, \tag{6.7}$$

$M_i(k)$ represents the mutation chance for datapoint i to be included in cluster k. All datapoints on selected chromosomes are assigned to the closest group by the largest Mahalanobis operator. To prevent creating illegal chromosome where all groups are not represented, a singleton cluster for each missing cluster is created then p datapoints are pseudo randomly assigned into it (Krishna and Murty, 1999). By this method, we do not encounter the problem that a datapoint is assigned into a missing cluster then reassigned into a different missing cluster. We have to assign at

least p datapoints into singleton to able to invert the covariance matrix.

We use equation (6.6) as a fitness function to rank the solutions to obtain the best partitioning of the data set in GARM, then apply the crossover operator to obtain the new population. After selecting the chromosomes to mutate, each datapoint is assigned to the closest cluster using the Mahalabonis operator given in (6.7). The algorithm is stopped when it meets the specified GA inputs. Briefly, the steps of our algorithm are as follows:

1. Create an initial generation with a population.

2. Rank each chromosome in the population according to the fitness function given by

$$f(P_p) = \frac{a(P_p)}{\sum_{p=1}^{P} a(P_p)}.$$

3. Select the datapoints to mutate on the selected chromosomes with the given mutation probability. Then, mutate the selected datapoints into the cluster to which they have the highest probability of group membership according to the mutation operator

$$M_i(k) = \frac{\max(m_i(k)) - m_i(k)}{\sum_{k=1}^{K} [\max(m_i(k)) - m_i(k)]}, \tag{6.8}$$

   where $m_i(k)$ is defined in (6.5).

4. Perform single point crossover on selected chromosomes from the new population using the crossover probability.

5. Check for illegal chromosomes. We ensure each chromosome has at least p members in each cluster. If a cluster disappears, we assign p observations to that class.

6. Perform the Mahalanobis operation. In this operation, we choose chromosomes without replacement, then all datapoints on the selected chromosomes are mutated into the cluster to which they are most likely to belong according to the mutation operator given in (6.8).

7. Use elitism rule if desired.

8. Replace the current population with the new population.

9. Repeat steps 2-8 until the specified GA inputs are satisfied.

We use the simulated data shown in Figure 4.2 to illustrate the performance of GARM in determining the number of mixtures for the simulated dataset. The confusion matrix of GARM is shown in Table 6.1. Looking at Table 6.1, we see that a single observation from the first cluster is assigned to the second cluster and five observations from cluster two re assigned into the first cluster. The misclassification error in this case is 2%. In comparison to the hybridized K-means initialization example shown in Section 4.3.2 for the same simulated data set we had misclassification error which was 4.6% error rate indicating better performance of the GARM approach.

6.3 Genetic EM Algorithm

The Genetic algorithm (GA) has been used widely in machine learning applications, including classification and prediction. As we discussed before, the EM algorithm can

Table 6.1: Confusion Matrix of GARM Algorithm

Actual/Predicted	1	2	
1	149	1	150
2	5	145	150
	154	146	300

get trapped in one of the many local maxima. The EM algorithm can dramatically change the results obtained by poor choices of the initial values due to the ruggedness of the log-likelihood surface of the MFA model. If we have a poor initialization, the EM algorithm converges slowly.There are various approaches in the literature to estimate the parameters of the mixture of factor analyzers (MFA) model. Zhou and Liu (2008) used the EM and Newton Raphson algorithms to estimate the parameters of the extended MFA of Fokouè (2005). Depending on the starting values, the iterative EM algorithm can return different parameters estimates. MacLachan et al. (2003) fit the MFA model by using the Alternative Expectation-Conditional Maximization (AECM) algorithm of Meng and van Dyk (1997). The difference from the EM algorithm is that it has more of computational maximization steps in the M-step of the EM algorithm. The advantage of AECM algorithm is that, it has a good converge properties and the likelihood function is not decreased after each iteration regardless of the initial values. In addition to, Cho and Zhang (2002) implement an evolutionary optimization by distribution estimation with MFA, which it is abbreviated as EDA algorithm. EDA algorithm replaces crossover and mutation operators by candidate solutions from the probability distribution. The distribution needs to be estimated accurately to able to capture the structure of the given problem in this algorithm. Another algorithm called incremental MFA has introduced been by Salah and Alpaydin (2004). Their algorithm starts with a one-factor, one-component mixture model and proceeds by adding new factors or a new component at each iteration until some stopping criterion is satisfied. Vlassis and Likas (2002) proposed a greedy EM algorithm to learn the Gaussian mixtures due to the problems of the regular EM algorithm explained above. This method is similar to the incremental mixtures of factor analyzers (MFA) model. The algorithm starts with a single component and adds components sequentially until a maximum number k of components in terms of the likelihood of a test set is obtained. The final specialized GA using the expectation maximization algorithm is called GEM, introduced initially by Wicker (2006) to prevent the usual EM algorithm getting trapped into the local

maxima. Howe (2009) extended and used GEM algorithm with information criteria to find the number of mixtures of cluster in the Gaussian and Kernel mixture models.

In this dissertation we use the genetic EM algorithm (GEM) to search the parameter space of the mixture of factor analyzers (MFA) model more intelligently, regardless of the initial values. We combine the genetic algorithm (GA) with the EM algorithm for the MFA model discussed in Section 4.2 to obtain the parameters and optimize the information criteria to find the best fitting model for a given dataset. This way, we choose the best the number of factors and number of mixtures in the MFA model simultaneously. GARM is used to obtain the first population for GEM algorithm of MFA. The GEM uses a biased mutation operator like GARM but the mutation operator is now defined as

$$P_i(k) = \frac{\max(h_i(k)) - h_i(k)}{\sum_{k=1}^{K} [\max(h_i(k)) - h_i(k)]}, \tag{6.9}$$

where $h_i(k)$ (see (4.8)) is the posterior probability of group membership. $P_i(k)$ denotes the chance of the i^{th} observation belonging to the k^{th} mixture. All datapoints in a selected chromosome are assigned to the mixture in which they are most likely belong by this operator. We can summarize our algorithm as follows:

1. Obtain an initial partitions using GARM. Then we use the GARM results as our initial solutions in GEM to search the parameters space of the MFA model. That is the values of (Λ, Ψ)

2. Estimate the parameters of the MFA for each chromosome, then calculate the information criterion scores of each chromosome.

3. Rank each chromosome of the population according to their information criteria score.

4. For the selected chromosome, choose the datapoints to mutate using the mutation operator defined in (6.9). In this step, we repeat Step 2 for estimating

the parameters of the MFA model to compute the posterior probabilities for the selected datapoints. Then, we assign the selected datapoints to the mixture cluster where they are most likely to belong. This is to ensure the number of observations in each mixture to be more than p, the number of variables.

5. Perform crossover on selected chromosomes using single point crossover.

6. Perform the posterior operation in (6.9). In this operation, we select the chromosomes without replacement. Then, all datapoints in the selected chromosomes are mutated into the cluster to which they are most likely to belong according to again (6.9). This is to ensure the number of observation in each mixture cluster is more than p.Otherwise, we will have singular solutions. Update the mixing proportion.

7. Use elitism rule if desired.

8. Replace the current population with the new population.

9. Repeat steps 2-8 until the specified GA inputs are satisfied.

6.4 Two Stage Genetic EM Algorithm

As we discussed earlier, the EM algorithm can get trapped in a local maxima. Because of this, and we introduced the genetic algorithm (GEM) for the MFA model in the previous section 6.3. Another major issue in MFA model is the current assumption that covariance matrix of the random error term is assumed to be the same across the mixture of clusters, and that one extracts the same number of factors. In practice, this is not a viable assumption. Therefore, one may ask the question: "Why does the number of factors or the covariances have to be the same for each population?" To preserve the heterogeneity in the mixture clusters, in this dissertation, we propose a new method for MFA model to achieve flexibility in our assumptions in order to be able to obtain different number of factors across the mixture of clusters. In this

new method, we develop a Two Stage Genetic EM algorithm. In the first stage of Two Stage GEM, we discover the number mixture clusters by the GARM method, and then for each partition we obtain the best approximating number of factors by Kaiser criterion, and then we use the EM algorithm to obtain the parameter estimates of Λ and Ψ of MFA model. In the second stage of Two Stage GEM, we maximize the log likelihood function of the MFA model and using the information criteria we obtain the final number of factors and the covariance matrix of the random errors for each mixture cluster. To further stabilize the covariance matrix, we use both the stabilization and smoothing covariance matrix when we have ill-conditioning. The steps of implementing the Two stage GEM algorithm are follows:

1. Create a partitioning of initial clusters by GARM.

2. Choose the best number of factors that can be extracted for each mixture by Kaiser criterion, and estimate the parameters using the EM algorithm of the standard factor model for the best fitting factor model.

3. Calculate the information criterion scores for each chromosome by using the parameters of the best factor models then rank each chromosome of the population according to their information criterion.

4. Select the datapoints on selected chromosomes to mutate using the mutation operator, which is the same operator given in (6.9). To compute the posterior probability of group membership of the selected datapoints, repeat the step 2 to estimate the parameters of the best fitting factor models of mixtures. Then, assign the selected datapoints to the mixture where they are most likely to belong. This is to ensure the number of observations in each mixture to be more than p, the number of variables.

5. Perform crossover on selected chromosomes.

6. Perform the posterior operation. Select the chromosomes, then find the posterior probabilities using (6.9) for all datapoints in that chromosome. Of

course, we repeat step 2 for estimating the parameters of the best fitting factor models of the mixtures to compute the posterior probabilities of all the datapoints. Then, all datapoints in selected chromosomes are mutated into the cluster to which they are most likely to belong. This is to ensure the number of observations in each mixture to be more than p. Update the mixing proportion.

7. Use elitism rule, if desired.

8. Replace the current population with the new population.

9. Repeat steps 2-8 until the specified GA inputs are satisfied.

Hence, in this manner, we obtain different numbers of factors in each mixture cluster and different covariance matrix of of random error term. This approach further gives different covariance matrix structure across the mixture clusters, which preserves the heterogeneity in the data set.This we like since it gives us a more realistic assumption in the MFA model. This method is also much faster in its computational time in more complex problems.

Chapter 7

Numerical Results

In this chapter, we show all our simulated and real datasets for the standard factor (BFA) model; Bayesian factor (BFA) model; and the mixture of factor analyzers (MFA) model. All our computations are carried out using a newly developed MATLAB module for the SFA, BFA and MFA models. Our results are obtained running these modules on Newton High Performance Computing (HPC) system at the University of Tennessee in Knoxville (UTK). Newton is a cluster computing system designed for the use by researchers at UTK. The computational time and complexity of the results vary according to the datasets we used and their dimensionality. Most simulations took less than 30 minutes execusion time to run. Datasets such as Parkinson, and Breast cancer took about 23 hours using the genetic algorithm in fitting the MFA model.

7.1 Standard Factor Analysis (SFA)

Consider a simple data set which we generated from a Gaussian distribution with the true number of factors $k^* = 3$, sample size $n = 100$ and $p = 9$ variables with parameters given by

$$\Lambda = \begin{bmatrix} 0.9 & 0.9 & 0.9 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.9 & 0.9 & 0.9 & 0.9 & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.1 & 0.9 & 0.9 & 0.9 \end{bmatrix}',$$

and

$$\Psi = diag\,(0.1024, 0.1024, .0400, 0.0400, 0.1600, 0.1600, 0.1600, 0.1600, 0.1600, 0.1600).$$

The first three variables (x_1, x_2, x_3) are assigned to factor one, second four variables (x_4, x_5, x_6) are assigned to factor two, and the last three variables (x_7, x_8, x_9) are assigned to the factor three. The parameter estimates obtained by the EM algorithm recovers the true structure of the $k^* = 3$ factor model. The parameters are estimated parameters are given as follows.

$$\widehat{\Lambda} = \begin{bmatrix} 0.84 & 0.83 & 0.83 & 0.07 & 0.05 & 0.07 & 0.06 & 0.19 & 0.21 & 0.23 \\ 0.20 & 0.17 & 0.19 & 0.84 & 0.85 & 0.83 & 0.84 & 0.18 & 0.20 & 0.26 \\ -0.10 & -0.12 & -0.14 & 0.05 & 0.08 & 0.01 & 0.03 & 0.90 & 0.96 & 0.98 \end{bmatrix}'$$

$$\widehat{\Psi} = diag\,(0.1311, 0.1059, .0396, 0.0352, 0.0404, 0.1663, 0.1684, 0.1164, 0.0935, 0.1177).$$

This shows that the EM algorithm is a good estimation method in the standard factor model framework. As we mentioned earlier, the best number of factors to fit to a given dataset obtained by using the information criteria. In this dissertation, the performances of information criteria are compared according to how they choose the best fitting true model for a given dataset using the EM algorithm in the SFA model. For the same simulation protocol given above, we fit the SFA model for $k = 1, 2, ..., 6$ factors since the maximum number of factor is 6 for $p = 10$ based on the big factor formula which gives a upper bound in extracting the number of factors. The number of factors in the model is chosen by minimizing the information criteria. The model selection frequencies are shown in Table 7.1.

Table 7.1: Model Selection Frequencies for the Standard Factor Model.

K	AIC	ICOMP_C	CAIC	SBC
1	0	0	0	0
2	0	0	0	0
3*	**85**	**100**	**100**	**100**
4	13	0	0	0
5	2	0	0	0
6	0	0	0	0

Out of 100 simulations, all five criteria picked the true structure with a high frequency. $CAIC$, SBC and $ICOMP_C$ performed very well in picking the true number of factors, whereas AIC picked the true model 85%. It overestimates the true model 15%. This tendency of AIC is not surprising since AIC is not a consistent model selection criterion.

7.1.1 Real Data- Medical School Admission Data

The medical School Admission dataset analyzed here was collected by Bozdogan (1973) from the Emory University Medical School. In this dataset, there are $n = 263$ observations medical school applicants on $p = 24$ different psychological test scores. Before the formal of this dataset analysis, we obtain the scree plot given in Figure 7.1 and eigenvalues given in Table 7.2. Based on Kaiser (1960) criterion discussed on page 7 of the correlation matrix that are greater than one. In essence this is like saying that, a factor extracts at least as much as the equivalent of one original variable, we drop it. For the medical admission dataset, 79% of the total variance is explained by $\widehat{k} = 5$ factors based on Kaiser criterion. Now, we fit the SFA model for this dataset up to 17 factors using the EM algorithm information criteria. As can be seen in Table 7.3, $ICOMP$, $CAIC$, SBC and AIC pick the 4-factor, 6-factor, 7-factor and 12-factor models respectively. We select four factor model for this dataset based on $ICOMP$ score.The scree plot also supports the selected model since the elbow of

approximately is at the four model. Further, this is also supported by the Kaiser criterion. Note that AIC picks $\widehat{k} = 12$ factors which is high as compared to the other information criteria. 93% of the variation is explained by 12 factor as opposed to 75% of the variation is explained by only 4 factors. Although, a high percentage of total variation be explained with 12 factors; we do not need additional 8 factors to reduce the complexity of the model to explain further only 18% of the variation. In other words, from the point of the principal of parsimony, or the Occan's Razor selecting a large number of factors does not reduce the complexity of the model. Therefore, the smaller the number is better fit. We show the estimated $\widehat{\Lambda}$ and $\widehat{\Psi}$ for the best fitting $\widehat{k} = 4$ model.

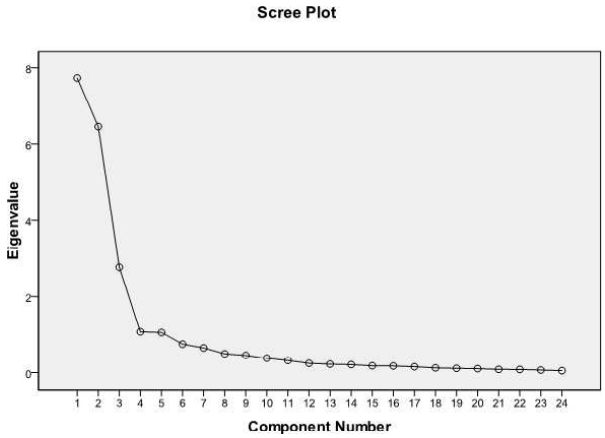


Figure 7.1: Scree plot for Medical School Admission Data.

Table 7.2: Eigenvalues of Medical School Admission Data.

	Initial Eigenvalues		
m	Total	% of Variance	Cumulative %
1	7.737	32.238	32.238
2	6.471	26.963	59.201
3	2.771	11.547	70.748
4	1.072	4.468	75.216
5	1.055	4.394	79.610
6	.750	3.125	82.736
7	.651	2.714	85.449
8	.499	2.079	87.528
9	.459	1.911	89.439
10	.383	1.595	91.034
11	.320	1.334	92.368
12	.247	1.031	93.399

Table 7.3: Model Selection for Medical School Admission Data.

m- factor model	AIC	ICOMP	CAIC	SBC
1	4375.5	4557.7	4397.5	4392.7
2	4134.5	4466.3	4167.0	4159.9
3	4023.9	4399.3	4066.4	4057.1
4	3988.7	**4387.1***	4040.8	4029.4
5	3974.2	4392.5	4035.5	4022.1
6	3963.2	4401.1	**4033.1***	4017.8
7	3955.5	4415.3	4033.7	**4016.6***
8	3950.4	4428.0	4036.4	4017.6
9	3946.7	4440.9	4040.0	4019.6
10	3944.0	4453.7	4044.1	4022.2
11	3942.0	4465.7	4048.5	4025.2
12	**3941.9***	4479.0	4054.3	4029.7
13	3942.2	4491.4	4060.2	4034.4
14	3942.7	4503.1	4065.7	4038.8
15	3943.1	4513.7	4070.6	4042.7
16	3944.2	4524.0	4075.9	4047.1
17	3945.6	4533.3	4080.9	4051.3

$$\widehat{\Lambda} = \begin{bmatrix} -0.99 & -0.95 & -0.61 & 0.60 \\ 1.43 & -5.89 & 1.69 & 0.16 \\ 0.70 & -7.43 & 2.92 & -0.33 \\ -1.47 & 3.17 & -3.29 & 1.17 \\ 2.22 & -5.76 & -2.30 & 5.99 \\ 3.21 & -2.83 & 4.87 & -4.50 \\ -4.31 & -1.59 & -1.11 & 1.87 \\ 1.17 & -5.99 & 4.01 & -2.14 \\ 5.30 & -5.49 & -1.18 & 2.99 \\ 3.90 & -5.97 & -1.62 & 6.01 \\ 7.54 & -4.08 & 0.81 & -0.75 \\ 7.44 & -3.45 & 1.24 & -1.11 \\ 1.80 & -5.10 & 4.10 & -2.45 \\ -1.26 & -6.42 & 3.21 & -4.72 \\ -1.62 & -7.77 & 2.18 & -0.71 \\ -4.54 & -6.28 & -2.57 & 0.68 \\ -1.91 & -3.06 & -4.59 & 5.74 \\ -0.63 & 1.36 & -2.20 & 7.59 \\ 1.02 & 2.67 & -5.79 & 5.94 \\ -3.70 & -2.18 & -1.16 & 4.45 \\ -2.36 & 4.04 & -4.94 & -4.14 \\ -1.38 & 3.01 & -0.50 & -7.37 \\ -0.43 & -0.23 & 2.43 & -8.29 \\ 1.11 & 5.11 & 1.90 & -3.22 \end{bmatrix} \quad diag\left(\widehat{\Psi}\right) = \begin{bmatrix} 66.83 \\ 16.23 \\ 11.51 \\ 21.15 \\ 21.42 \\ 19.15 \\ 62.53 \\ 18.06 \\ 12.41 \\ 7.83 \\ 5.48 \\ 14.90 \\ 36.73 \\ 15.76 \\ 19.83 \\ 42.01 \\ 18.37 \\ 19.04 \\ 9.16 \\ 49.19 \\ 16.74 \\ 13.00 \\ 11.22 \\ 35.48 \end{bmatrix}$$

7.2 Bayesian Factor Analysis (BFA)

We generate multivariate normal data using the model structure provided correspondence Bozdogan with the population parameters for the ture number of factors $k^* = 3$, sample size $n = 100$, and $p = 9$ variables given by

$$\Lambda' = \begin{bmatrix} 0.6 & 0.7 & 0.8 & 0.2 & 0 & 0 & 0.2 & 0 & 0 \\ 0.2 & 0 & 0 & 0.6 & 0.7 & 0.8 & 0 & 0.2 & 0 \\ 0 & 0.2 & 0 & 0 & 0.2 & 0 & 0.6 & 0.7 & 0.8 \end{bmatrix} \quad H = \begin{bmatrix} 1.0 & 0.08 & 0.12 \\ 0.08 & 1.0 & 0.24 \\ 0.12 & 0.24 & 1.0 \end{bmatrix}$$

$$\Psi = diag(0.581, 0.436, 0.360, 0.581, 0.403, 0.360, 0.571, 0.403, 0.360),$$

where Λ' is the transpose of factor loading matrix, H is the inter-factor correlation matrix, and Ψ is the unique variances matrix. As we mentioned earlier, the covariance matrix is $\Sigma = \Lambda H \Lambda' + \Psi$. To estimate the parameters of the BFA model, we use Gibbs Sampling and ICM methods. We use the Sparse Root algorithm to learn the prior factor pattern structure of Λ_0. The hyperparameters of the model are assessed as: $H_0 = 10I_m$, $B_0 = 0.2I_p$ and $v = 2(p+1)+1$.

Using PS89, Gibbs sampling, and ICM methods, we estimated the parameters for the true BFA model $k^* = 3$. In out simulation study and compare the three estimation methods. Note that the parameter estimates of Gibbs sampling approach are closer to the true parameter values as compared to PS89 and ICM methods. Since Gibbs sampling is using the conditional posterior estimates of the parameters rather than using modal values of the parameters which is used in ICM, Gibbs Sampling being better can be expected.

$$\widehat{\Lambda}'_{PS89} = \begin{bmatrix} 0.44 & 0.47 & 0.50 & 0.12 & -0.09 & -0.03 & 0.07 & -0.02 & 0.06 \\ 0.09 & -0.01 & 0.01 & 0.42 & 0.45 & 0.54 & 0.04 & 0.12 & 0.04 \\ -0.03 & 0.08 & -0.06 & -0.07 & 0.12 & -0.04 & 0.48 & 0.45 & 0.46 \end{bmatrix}$$

$$\widehat{\Lambda}'_{ICM} = \begin{bmatrix} 0.47 & 0.48 & 0.52 & 0.07 & -0.03 & -0.03 & 0.13 & 0.01 & -0.04 \\ 0.12 & 0.04 & -0.05 & 0.47 & 0.50 & 0.50 & -0.02 & 0.13 & 0.11 \\ 0.00 & 0.06 & -0.05 & -0.02 & 0.04 & -0.01 & 0.46 & 0.51 & 0.49 \end{bmatrix}$$

$$\widehat{\Lambda}'_{Gibbs} = \begin{bmatrix} 0.61 & 0.59 & 0.62 & 0.09 & 0.00 & 0.02 & 0.18 & 0.07 & 0.06 \\ 0.17 & 0.02 & 0.05 & 0.64 & 0.63 & 0.64 & 0.07 & 0.23 & 0.21 \\ 0.04 & 0.11 & 0.03 & 0.02 & 0.14 & 0.06 & 0.61 & 0.59 & 0.60 \end{bmatrix}$$

$$\widehat{\Psi}_{PS89} = \begin{bmatrix} 0.35 & -0.19 & -0.13 & -0.02 & 0.00 & -0.07 & -0.05 & -0.01 & 0.06 \\ & 0.32 & -0.11 & 0.01 & 0.01 & 0.02 & -0.03 & 0.06 & -0.03 \\ & & 0.25 & -0.01 & 0.01 & 0.03 & -0.01 & 0.00 & 0.02 \\ & & & 0.34 & -0.19 & -0.12 & 0.08 & -0.04 & -0.04 \\ & & & & 0.30 & -0.11 & -0.06 & 0.03 & 0.04 \\ & & & & & 0.27 & 0.09 & -0.03 & -0.06 \\ & & & & & & 0.43 & -0.22 & -0.21 \\ & & & & & & & 0.25 & -0.03 \\ & & & & & & & & 0.24 \end{bmatrix}$$

$$\widehat{\Psi}_{ICM} = \begin{bmatrix} 0.25 & -0.15 & -0.11 & 0.0143 & -0.04 & -0.02 & 0.02 & 0.00 & -0.04 \\ & 0.26 & -0.09 & 0.02 & 0.04 & 0.01 & -0.01 & 0.05 & 0.02 \\ & & 0.22 & 0.04 & -0.03 & -0.04 & -0.05 & -0.01 & 0.02 \\ & & & 0.27 & -0.13 & -0.15 & 0.03 & -0.02 & -0.02 \\ & & & & 0.2118 & -0.05 & 0.02 & 0.03 & -0.01 \\ & & & & & 0.23 & 0.01 & 0.00 & -0.01 \\ & & & & & & 0.31 & -0.11 & -0.21 \\ & & & & & & & 0.2395 & -0.10 \\ & & & & & & & & 0.31 \end{bmatrix}$$

$$\widehat{\Psi}_{Gibbs} = \begin{bmatrix} 0.28 & -0.09 & -0.04 & 0.09 & -0.01 & -0.01 & -0.04 & 0.00 & -0.02 \\ & 0.36 & 0.05 & 0.08 & -0.02 & -0.01 & 0.04 & 0.15 & 0.09 \\ & & 0.31 & 0.12 & -0.03 & 0.00 & 0.01 & 0.03 & 0.01 \\ & & & 0.29 & -0.09 & -0.06 & 0.02 & 0.02 & -0.06 \\ & & & & 0.29 & 0.05 & 0.11 & 0.12 & 0.12 \\ & & & & & 0.28 & 0.03 & 0.07 & 0.03 \\ & & & & & & 0.32 & -0.04 & -0.05 \\ & & & & & & & 0.32 & 0.09 \\ & & & & & & & & 0.32 \end{bmatrix}$$

Next, we fit $k = 1, 2, ..., 5$ factors since the maximum number of factors is 5 when $p = 9$. Again, the pattern structures are obtained using the Sparse Root algorithm corresponding to $k = 1, 2, ..., 5$ factors to initialize our prior factor loading matrix Λ_0 and the other prior hyperparameters are assessed by: $H_0 = 10I_m$, $B_0 = 0.2I_p$, and $v = 2(p + 1) + 1$. With this set up, we ran 100 simulations using the simulation structure given previously. The information criteria are scored for $k = 1, 2, ..., 5$. the true number of factors is selected using the minimum values of the information criteria. Finally, we obtained the model selection frequency for different sample sizes using the Gibbs Sampling, ICM and the PS89 methods. The results from there simulations are summarized in Tables 7.4 through 7.6. Looking at these tables, we see that as the number of factors increases, the percentage of hitting the true BFA

model also increases in all the methods. As can be seen in Table 7.4, PS89, which is a large sample approximation method, needs more samples to recover the true structure. All the criteria are minimized at the true factor model $k^* = 3$ with highest percentages. The performance of $ICOMP$ to choose the true model is better than AIC and $CAIC$.

We consistently hit the true model over 90% of the times using ICM and Gibbs sampling even if the sample size is small. The highest frequency is 60% to select the true model by the PS89 method and this is obtained when $n = 200$. But we obtain much higher performance when $n = 50$ using Gibbs sampling and ICM methods. Gibbs sampling is the best one to select the true model but there is a computational cost in using the Gibbs sampling method. It takes at least twice as much time to compute the estimators as compared to the ICM method. We have the same performance with $10,000$ ICM iteration, and $20,000$ Gibbs sampling when the sample size is over 100. Thus, we suggest to use Gibbs sampling for the small sample sizes, and the ICM method for the large sample sizes, since their performance is almost the same for large sample sizes. Moreover, we suggest to use $ICOMP$ to choose the best fitting number of factors in the BFA model.

7.2.1 Crime Data Set

We have analyzed this data using PS89, ICM and Gibbs sampling procedures to estimate the parameters in the BFA model with a prior loading structure that are

Table 7.4: Model Selection Frequencies in BFA Model for PS89 methods.

n=50						**n=100**						**n=200**					
	1	2	3*	4	5		1	2	3*	4	5		1	2	3*	4	5
AIC	0	30	**45**	18	7	AIC	1	31	**55**	12	1	AIC	0	31	**59**	10	0
CAIC	2	34	**45**	16	3	CAIC	1	38	**50**	11	0	CAIC	0	37	**58**	5	0
ICOMP	0	31	**50**	13	6	ICOMP	1	35	**56**	8	0	ICOMP	0	35	**61**	4	0

Table 7.5: Model Selection Frequencies in BFA Model for ICM.

n=50						**n=100**						**n=200**					
	1	2	3*	4	5		1	2	3*	4	5		1	2	3*	4	5
AIC	0	2	**67**	24	7	AIC	0	0	**91**	7	2	AIC	0	0	**91**	7	2
CAIC	0	4	**71**	20	5	CAIC	0	0	**96**	4	0	CAIC	0	0	**96**	4	0
ICOMP	0	2	**73**	20	5	ICOMP	0	0	**97**	3	0	ICOMP	0	0	**97**	3	0

Table 7.6: Model Selection Frequencies in BFA Model for Gibbs Sampling.

n=50						**n=100**						**n=200**					
	1	2	3*	4	5		1	2	3*	4	5		1	2	3*	4	5
AIC	0	0	**91**	8	1	AIC	0	0	**100**	7	2	AIC	0	0	**100**	0	0
CAIC	0	0	**93**	6	1	CAIC	0	0	**100**	4	0	CAIC	0	0	**100**	0	0
ICOMP	0	0	**90**	9	1	ICOMP	0	0	**100**	3	0	ICOMP	0	0	**100**	0	0

obtained data-adaptively to determine the number of factors using the information complexity ($ICOMP$). This dataset is collected from 16 US states on different types of crimes on $p = 7$ variables: x_1 =Murder, x_2 =Rape, x_3 =Robbery, x_4 =Assault, x_5 =Burglary, x_6=Larceny, x_7=Auto theft.

The scree plot and eigenvalues for this dataset are shown in Figure 7.2 and Table 7.7 to have an initial idea about the number of the factors present. Based on the scree plot, the number of factors in the BFA model is around 2, or 3. Based on the Kaiser criterion, the 2-factor model is selected. The two factor model explains the 68% of the variation, the 3-factor model explains 81% of the variation, according to the eigenvalues.

To apply the BFA model on this data, we obtained the prior factor loading matrix for models up to 4 factors by the Sparse Root algorithm given follows.

$$\Lambda_0 = \begin{bmatrix} 0 & 0.7 & 0 & -0.6 \\ 0 & 0.6 & 0.1 & 0.6 \\ 0 & 0.8 & 0.1 & -0.05 \\ 0 & 0.5 & -0.5 & -0.04 \\ 0 & 0.4 & -0.6 & 0.09 \\ 0.7 & 0.5 & 0.1 & 0.06 \end{bmatrix}.$$

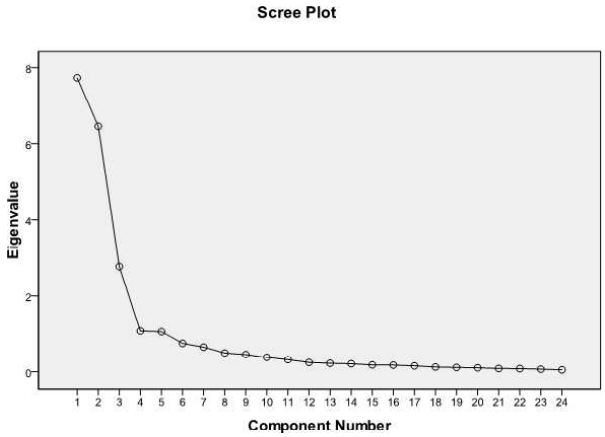


Figure 7.2: Scree plot for Crime Data.

Table 7.7: Eigenvalues of the Crime Data.

	Initial Eigenvalues		
m	Total	% of Variance	Cumulative %
1	3.452	49.308	49.308
2	1.333	19.038	68.346
3	.940	13.432	81.778
4	.627	8.957	90.735
5	.366	5.227	95.962
6	.170	2.429	98.391
7	.113	1.609	100.000

The inter-factor correlation matrix, H was assessed by $H = 10I_7$. The prior distribution of Ψ was assessed by $B = 0.2I_7$ and $v = 17$. As can be seen in Table 7.8, 7.10, 7.12, all of the information criteria are minimized at the true $k^* = 3$ factors model regardless of estimation methods we used. According to the parameter estimates of Λ for the 3-factor model which is the best fit model, we can combine *rape*, *robbery*, *assault* and *auto theft* under the first factor; *burglary* and *larceny* under the second factor, and *murder* is under the third factor by itself.

Table 7.8: Model Selection for the Crime Data Using PS89 Method.

m	AIC	CAIC	ICOMP
1	60034669.46	60034683.64	60034664.78
2	56111797.60	56111824.19	56111788.09
3	**29854611.96***	**29854650.95***	**29854596.32***
4	36971065.83	36971124.33	36971034.20

Table 7.9: Crime Data Results Using PS89 Method.

	Factor 1	Factor 2	Factor 3
Murder	-0.03	0.51	-0.69
Rape	0.09	0.70	-0.01
Robbery	-0.02	0.69	0.02
Assault	-0.01	0.80	-0.06
Burglary	0.70	0.42	0.02
Larceny	0.63	0.38	0.38
Auto thief	-0.07	0.54	0.47

Table 7.10: Model Selection for Crime Data Using Gibbs Sampling.

m	AIC	CAIC	ICOMP
1	99882961.70	99882975.88	99882954.14
2	90481534.09	90481560.68	90481514.29
3	**40863402.77***	**40863441.77***	**40863370.24***
4	51528807.14	51528865.64	51528754.20

Table 7.11: Crime Data Results Using Gibbs Sampling.

	Factor 1	Factor 2	Factor 3
Murder	-0.02	0.50	-0.68
Rape	0.07	0.69	-0.01
Robbery	-0.01	0.68	0.03
Assault	-0.02	0.79	-0.05
Burglary	0.69	0.41	0.02
Larceny	0.61	0.38	0.38
Auto thief	-0.03	0.52	0.43

Table 7.12: Model Selection for Crime Data Using ICM Method.

m	AIC	CAIC	ICOMP
1	113450832.83	113450847.01	113450828.26
2	102329359.91	102329386.50	102329350.41
3	**50830819.60***	**50830858.60***	**50830803.95***
4	61126531.98	61126590.47	61126500.30

Table 7.13: Crime data results using ICM method.

	Factor 1	Factor 2	Factor 3
Murder	-0.03	0.51	-0.68
Rape	0.09	0.70	-0.01
Robbery	-0.02	0.69	0.02
Assault	-0.00	0.80	-0.07
Burglary	0.69	0.42	0.02
Larceny	0.62	0.38	0.38
Auto thief	-0.08	0.54	0.48

7.3 EM Algorithm for the Mixture of Factor Analyzers with Random, GARM and K-means Initialization

In this section, we compare the EM algorithm used by Ghahramani and Hinton (1997).We use GARM and K-means initialization methods on simulated and real datasets. We compare the performances of information criteria to select the true MFA model in simulation study and the best approximating model using these initialization methods.

7.3.1 Estimation of the Parameters

We used simulation one (S1) structure given in Appendix A to generate the multivariate normal dataset with the number of variables $p = 10$ and $n = 200$ observations to estimate the parameters of the MFA model by using the EM algorithm. The simulated data is generated by combining two Standard factor (SFA) models. Each factor model is composed of three factors. The estimated parameters are given in Table 7.14 through Table 7.16. All of the parameter are estimated obtained by using the EM algorithm and all the three different initialization methods to choose the true MFA model with $m^* = 2$ mixtures and $k = 3^*$ factors. The estimates are closer to the true parameter values using the EM algorithm with the GARM and the hybertized K-means initialization schemes, rather than the random initialization scheme.

As we discussed earlier, it is really important how to initialize the EM algorithm so that it converges to a global optimum. Then, the estimates are closer to the real parameter values with better initialization techniques.

Table 7.14: Parameter Estimates Using the EM Algorithm for the MFA Model with Random Initialization.

m	$\widehat{\pi}_m$	$\widehat{\Lambda}_m$	$\widehat{\mu}_m$	$\text{diag}(\widehat{\Psi}_m)$
1	0.45	$\begin{bmatrix} -0.42 & 0.34 & -0.29 \\ -0.57 & -0.29 & 0.10 \\ -0.42 & 0.39 & -0.28 \\ -0.51 & -0.27 & 0.12 \\ -0.47 & 0.42 & -0.31 \\ -0.54 & -0.32 & 0.12 \\ -0.39 & 0.42 & -0.30 \\ -0.24 & -0.32 & -0.49 \\ -0.15 & -0.31 & -0.56 \\ -0.22 & -0.27 & -0.52 \end{bmatrix}$	$\begin{bmatrix} 17.16 \\ 17.09 \\ 17.13 \\ 17.08 \\ 17.19 \\ 17.08 \\ 17.15 \\ 17.10 \\ 17.08 \\ 17.10 \end{bmatrix}$	$\begin{bmatrix} 0.10 \\ 0.10 \\ 0.03 \\ 0.05 \\ 0.03 \\ 0.15 \\ 0.15 \\ 0.13 \\ 0.16 \\ 0.14 \end{bmatrix}$
2	0.50	$\begin{bmatrix} 0.60 & -0.49 & 0.00 \\ 0.60 & -0.48 & 0.05 \\ 0.59 & -0.49 & -0.03 \\ -0.33 & -0.56 & -0.37 \\ -0.29 & -0.61 & -0.38 \\ -0.33 & -0.56 & -0.35 \\ -0.32 & -0.54 & -0.38 \\ -0.11 & -0.51 & 0.61 \\ -0.17 & -0.54 & 0.63 \\ -0.12 & -0.54 & 0.58 \end{bmatrix}$	$\begin{bmatrix} 19.97 \\ 19.98 \\ 19.95 \\ 19.94 \\ 19.99 \\ 19.96 \\ 19.95 \\ 20.00 \\ 19.99 \\ 20.00 \end{bmatrix}$	$\begin{bmatrix} 0.10 \\ 0.10 \\ 0.03 \\ 0.05 \\ 0.03 \\ 0.15 \\ 0.15 \\ 0.13 \\ 0.16 \\ 0.14 \end{bmatrix}$

Table 7.15: Parameter Estimates Using the EM Algorithm for the MFA Model with GARM Initialization.

m	$\widehat{\pi}_m$	$\widehat{\Lambda}_m$	$\widehat{\mu}_m$	$\text{diag}(\widehat{\Psi}_m)$
1	0.49	$\begin{bmatrix} 0.58 & -0.09 & -0.24 \\ 0.34 & 0.58 & 0.17 \\ 0.58 & -0.10 & -0.30 \\ 0.29 & 0.54 & 0.15 \\ 0.64 & -0.11 & -0.32 \\ 0.29 & 0.58 & 0.19 \\ 0.57 & -0.16 & -0.31 \\ 0.45 & -0.06 & 0.50 \\ 0.42 & -0.17 & 0.54 \\ 0.46 & -0.11 & 0.47 \end{bmatrix}$	$\begin{bmatrix} 17.02 \\ 17.02 \\ 16.98 \\ 17.02 \\ 17.03 \\ 17.03 \\ 17.00 \\ 17.03 \\ 17.02 \\ 17.04 \end{bmatrix}$	$\begin{bmatrix} 0.10 \\ 0.10 \\ 0.03 \\ 0.05 \\ 0.03 \\ 0.15 \\ 0.15 \\ 0.13 \\ 0.16 \\ 0.14 \end{bmatrix}$
2	0.51	$\begin{bmatrix} 0.69 & -0.40 & -0.26 \\ 0.68 & -0.38 & -0.31 \\ 0.70 & -0.42 & -0.23 \\ -0.22 & -0.70 & 0.32 \\ -0.18 & -0.74 & 0.31 \\ -0.22 & -0.70 & 0.30 \\ -0.21 & -0.68 & 0.3 \\ -0.19 & -0.39 & -0.71 \\ -0.26 & -0.42 & -0.73 \\ -0.21 & -0.42 & -0.69 \end{bmatrix}$	$\begin{bmatrix} 20.03 \\ 20.03 \\ 20.01 \\ 20.02 \\ 20.04 \\ 20.04 \\ 20.03 \\ 20.04 \\ 20.03 \\ 20.04 \end{bmatrix}$	$\begin{bmatrix} 0.10 \\ 0.10 \\ 0.03 \\ 0.05 \\ 0.03 \\ 0.15 \\ 0.15 \\ 0.13 \\ 0.16 \\ 0.14 \end{bmatrix}$

Table 7.16: Parameter Estimates Using the EM Algorithm for the MFA Model with K-Means Initialization.

m	$\widehat{\pi}_m$	$\widehat{\Lambda}_m$	$\widehat{\mu}_m$	$\text{diag}(\widehat{\Psi}_m)$
1	0.4949	$\begin{bmatrix} 0.61 & 0.16 & 0.00 \\ -0.01 & 0.66 & 0.20 \\ 0.64 & 0.15 & -0.05 \\ -0.03 & 0.61 & 0.15 \\ 0.71 & 0.17 & -0.04 \\ -0.06 & 0.65 & 0.19 \\ 0.66 & 0.10 & -0.06 \\ 0.20 & 0.13 & 0.63 \\ 0.20 & 0.01 & 0.67 \\ 0.24 & 0.08 & 0.61 \end{bmatrix}$	$\begin{bmatrix} 17.02 \\ 17.02 \\ 16.98 \\ 17.02 \\ 17.03 \\ 17.03 \\ 17.00 \\ 17.03 \\ 17.02 \\ 17.04 \end{bmatrix}$	$\begin{bmatrix} 0.10 \\ 0.10 \\ 0.03 \\ 0.04 \\ 0.03 \\ 0.15 \\ 0.15 \\ 0.13 \\ 0.16 \\ 0.14 \end{bmatrix}$
2	0.5051	$\begin{bmatrix} 0.57 & 0.00 & -0.62 \\ 0.54 & 0.04 & -0.65 \\ 0.59 & -0.03 & -0.60 \\ -0.01 & -0.80 & -0.09 \\ 0.03 & -0.82 & -0.13 \\ -0.02 & -0.79 & -0.10 \\ -0.01 & -0.78 & -0.08 \\ -0.42 & -0.02 & -0.72 \\ -0.49 & -0.06 & -0.74 \\ -0.41 & -0.07 & -0.73 \end{bmatrix}$	$\begin{bmatrix} 20.03 \\ 20.03 \\ 20.01 \\ 20.02 \\ 20.04 \\ 20.04 \\ 20.03 \\ 20.04 \\ 20.03 \\ 20.04 \end{bmatrix}$	$\begin{bmatrix} 0.10 \\ 0.10 \\ 0.03 \\ 0.04 \\ 0.03 \\ 0.15 \\ 0.15 \\ 0.13 \\ 0.16 \\ 0.14 \end{bmatrix}$

7.3.2 Model Selection Using the EM Algorithm for the MFA Model

We generated MFA model which is obtained by combining two multivariate normal distributions in Appendix A with $p = 10$ variables and $n = 200$ observations to verify the model selection performance using the information criteria in the MFA model. We performed 100 replications of the simulation protocol in attempting to fit up to the maximum number of factor $K = 6$ for each mixture, and $m = 1, 2, ..., 4$ mixtures. We use the EM algorithm using all three initialization schemes. Our results are given in Tables 7.17 through 7.19. We use " * " to indicate the true model selection frequencies.

Looking at Table 7.17, the frequency of selecting the true model is over 70 out of 100 simulations with the random initialization scheme. SBC and $CAIC$ have better performance then $ICOMP_C$ and AIC. They both hit the true number of mixtures ($m^* = 2$) and the number of factors ($k^* = 3$) with the highest frequency. Although all of the criteria selected the true number of clusters with over 95%, the selection of number of factors changes. Looking at Table 7.18 and 7.19, we see that all the information criteria choose the true model with 100% using GARM and hybridized K-Means initialization of the EM algorithm for the MFA model. As a result, we suggest to use AIC, $CAIC$, $ICOMP_C$ or SBC to choose the number of factors and number of mixtures simultaneously in the MFA model.

7.3.3 Real Data Results Using the EM Algorithm for the MFA Model

In this section, we apply the EM algorithm in the MFA model is applied on real datasets. The performances of information criteria to select the best approximating model using the GARM, the hybridized K-means, and random initialization schemes. We compare our results from these analyzers in what follows.

Table 7.17: Model Selection Frequency Using EM Algorithm of MFA with Random Initialization.

AIC							**CAIC**						
m\k	**1**	**2**	**3***	**4**	**5**	**6**	**m\k**	**1**	**2**	**3***	**4**	**5**	**6**
1	0	0	0	0	0	0	**1**	0	0	0	0	0	0
2*	0	0	**77**	16	6	0	**2***	0	0	**81**	15	4	0
3	0	0	1	0	0	0	**3**	0	0	0	0	0	0
4	0	0	0	0	0	0	**4**	0	0	0	0	0	0
$ICOMP_C$							**SBC**						
m\k	**1**	**2**	**3***	**4**	**5**	**6**	**m\k**	**1**	**2**	**3***	**4**	**5**	**6**
1	0	0	0	5	0	0	**1**	0	0	0	0	0	0
2*	0	18	**70**	7	0	0	**2***	0	0	**80**	16	4	0
3	0	0	0	0	0	0	**3**	0	0	0	0	0	0
4	0	0	0	0	0	0	**4**	0	0	0	0	0	0

Table 7.18: Model Selection Frequency Using the EM Algorithm for MFA with K-Means Initialization.

AIC							**CAIC**						
m\k	**1**	**2**	**3***	**4**	**5**	**6**	**m\k**	**1**	**2**	**3***	**4**	**5**	**6**
1	0	0	0	0	0	0	**1**	0	0	0	0	0	0
2*	0	0	**100**	0	0	0	**2***	0	0	**100**	0	0	0
3	0	0	0	0	0	0	**3**	0	0	0	0	0	0
4	0	0	0	0	0	0	**4**	0	0	0	0	0	0
$ICOMP_C$							**SBC**						
m\k	**1**	**2**	**3***	**4**	**5**	**6**	**m\k**	**1**	**2**	**3***	**4**	**5**	**6**
1	0	0	0	0	0	0	**1**	0	0	0	0	0	0
2*	0	0	**100**	0	0	0	**2***	0	0	**100**	0	0	0
3	0	0	0	0	0	0	**3**	0	0	0	0	0	0
4	0	0	0	0	0	0	**4**	0	0	0	0	0	0

Table 7.19: Model Selection Frequency Using the EM Algorithm for MFA with GARM Initialization.

AIC							CAIC						
m\k	**1**	**2**	**3***	**4**	**5**	**6**	**m\k**	**1**	**2**	**3***	**4**	**5**	**6**
1	0	0	0	0	0	0	**1**	0	0	0	0	0	0
2*	0	0	**100**	0	0	0	**2***	0	0	**100**	0	0	0
3	0	0	0	0	0	0	**3**	0	0	0	0	0	0
4	0	0	0	0	0	0	**4**	0	0	0	0	0	0
$ICOMP_C$							**SBC**						
m\k	**1**	**2**	**3***	**4**	**5**	**6**	**m\k**	**1**	**2**	**3***	**4**	**5**	**6**
1	0	0	0	0	0	0	**1**	0	0	0	0	0	0
2*	0	0	**100**	0	0	0	**2***	0	0	**100**	0	0	0
3	0	0	0	0	0	0	**3**	0	0	0	0	0	0
4	0	0	0	0	0	0	**4**	0	0	0	0	0	0

College Data

Our first real dataset is the college data which is composed of $k = 2$ groups. There are $n = 123$ observations. The group one $n_1 = 34$ observation the most selective schools, group two has and $n_2 = 89$ the more selective schools. This dataset is given in Appendix A along with other datasets. We executed the EM algorithm initialized by hybridized K-Means, GARM, and random initialization for $k = 1, 2, ..., 5$ factors and $m = 1, ..., 4$ mixtures. Our results are shown in Table 7.20. Looking at Table 7.20, we see the random initialization seems to be not working well for this dataset. On the other hand, both GARM and hybridized K-Means are given us reasonable results using AIC, $CAIC$, SBC, and the $ICOMP$ criteria. We note that $\widehat{m} = 2$ mixtures and $\widehat{k} = 3$ factors seems to be the best approximating model with correct classification rate 93.31%.

Wine Data

This dataset is obtained from three different wines grown in the same region in Italy. $p = 13$ characteristic measurements were taken on $n_1 = 59$, $n_2 = 71$, $n_3 = 48$ cultivators. We executed the EM algorithm initialized by K-Means, GARM and

Table 7.20: College Data- MFA with EM Results.

	MFA Results					
IC	**Random Init.**		**GARM Init.**		**K-Means Init.**	
	MFA Model	**CCR**	**MFA Model**	**CCR**	**MFA Model**	**CCR**
AIC	$\widehat{m}=2,\widehat{k}=1$	94.31	$\widehat{m}=2,\widehat{k}=4$	94.31	$\widehat{m}=2,\widehat{k}=4$	94.31
CAIC	$\widehat{m}=2,\widehat{k}=1$	94.31	$\widehat{m}=2,\widehat{k}=4$	94.31	$\widehat{m}=2,\widehat{k}=3$	94.31
SBC	$\widehat{m}=2,\widehat{k}=1$	94.31	$\widehat{m}=2,\widehat{k}=4$	94.31	$\widehat{m}=2,\widehat{k}=3$	94.31
ICOMP$_{\text{CMISS}}$	$\widehat{m}=3,\widehat{k}=3$	80.49	$\widehat{m}=3,\widehat{k}=3$	78.86	$\widehat{m}=3,\widehat{k}=3$	75.61
ICOMP$_{\text{PEUMISS}}$	$\widehat{m}=2,\widehat{k}=6$	80.49	$\widehat{m}=3,\widehat{k}=3$	78.86	$\widehat{m}=3,\widehat{k}=3$	75.61

Table 7.21: Wine Data- MFA with EM Results.

	MFA Results					
IC	**Random Init.**		**GARM Init.**		**K-Means Init.**	
	MFA Model	**CCR**	**MFA Model**	**CCR**	**MFA Model**	**CCR**
AIC	$\widehat{m}=3,\widehat{k}=3$	97.75	$\widehat{m}=2,\widehat{k}=5$	58.43	$\widehat{m}=2,\widehat{k}=6$	60.11
CAIC	$\widehat{m}=3,\widehat{k}=1$	97.19	$\widehat{m}=2,\widehat{k}=5$	58.43	$\widehat{m}=2,\widehat{k}=3$	60.11
SBC	$\widehat{m}=3,\widehat{k}=3$	97.75	$\widehat{m}=2,\widehat{k}=5$	58.43	$\widehat{m}=2,\widehat{k}=3$	60.11
ICOMP$_{\text{CMISS}}$	$\widehat{m}=2,\widehat{k}=6$	60.11	$\widehat{m}=2,\widehat{k}=6$	58.43	$\widehat{m}=2,\widehat{k}=6$	60.11
ICOMP$_{\text{PEUMISS}}$	$\widehat{m}=2,\widehat{k}=6$	60.11	$\widehat{m}=2,\widehat{k}=6$	58.43	$\widehat{m}=2,\widehat{k}=6$	60.11

random initialization schemes for $m=1,...,4$ mixtures and $k=1,...,8$ factors in the MFA model. Our The results are summarized in Table 7.21. AIC, $CAIC$ and SBC pick $\widehat{m}=3$ mixtures with random initialization, $\widehat{m}=2$ mixtures with GARM and hybridized K-Means initialization. $ICOMP$ type criteria pick $\widehat{m}=2$ mixtures. As can be seen in the scatter plots given in Appendix A, this dataset is highly overlapped. Because of this, either $\widehat{m}=3$ mixtures and $\widehat{k}=3$ factors with correct classification rate 97.75%, or $\widehat{m}=2$ mixtures and $\widehat{k}=6$ factors with correct classification rate 60.11% seems to be the best approximating model.

Parkinson Data

Next, we apply the EM algorithm to the Parkinson dataset. This dataset was obtained from Little et al. (2007) in collaboration with the National Centre for Voice and Speech in Denver, Colorado. Each variable is a particular voice measure, and each observation

corresponds to one of $n = 195$ ($n_1 = 48$ parkinson and $n_2 = 147$ no parkinson disease) voice recordings from these individuals. There are $p = 22$ variables in this dataset. This is a very challenging dataset to analyze since the data is not normal in many dimensions and the groups are overlapped. This dataset and scatter plot matrix given in Appendix A. Little et al. (2007) categorized the variables for this dataset under 9 categories. These are:

- Average vocal fundamental frequency
- Maximum vocal fundamental frequency
- Minimum vocal fundamental frequency
- Measures of variation in fundamental frequency
- Measures of variation in amplitude
- Ratio of noise to tonal components in the voice
- Nonlinear dynamical complexity measures
- Signal fractal scaling exponent
- Nonlinear measures of fundamental frequency variation.

We fit the model $m = 1, ..., 4$ mixtures and $k = 1, ..., 15$ factors using the EM algorithm with random, GARM and hybridized K-means initializations. As can be

Table 7.22: Parkinson Data- MFA with EM Results.

	MFA Results					
IC	**Random Init.**		**GARM Init.**		**K-Means Init.**	
	MFA Model	**CCR**	**MFA Model**	**CCR**	**MFA Model**	**CCR**
AIC	$\widehat{m} = 3, \widehat{k} = 8$	35.90	$\widehat{m} = 2, \widehat{k} = 13$	61.03	$\widehat{m} = 2, \widehat{k} = 8$	56.41
CAIC	$\widehat{m} = 2, \widehat{k} = 5$	65.64	$\widehat{m} = 2, \widehat{k} = 7$	58.97	$\widehat{m} = 2, \widehat{k} = 8$	56.41
SBC	$\widehat{m} = 2, \widehat{k} = 6$	58.46	$\widehat{m} = 2, \widehat{k} = 7$	58.97	$\widehat{m} = 2, \widehat{k} = 8$	56.41
ICOMP$_{\mathbf{C}}$	$\widehat{m} = 1, \widehat{k} = 5$	-	$\widehat{m} = 2, \widehat{k} = 4$	67.18	$\widehat{m} = 2, \widehat{k} = 4$	68.21

seen in Table 7.22, all information criteria pick $\widehat{m} = 2$ mixtures with GARM and hybridized K-means initialization. The results obtained with random initialization are not consistent across the information criteria. $\widehat{m} = 2$ mixtures and $\widehat{k} = 4$ factors chosen by $ICOMP$ seems to be best approximanting model with highest correct classification rate of 68.21%. Therefore, we reduce the dimension to 4 factors from 22 original variables for this dataset using the MFA model. Note that our results further reduces the dimension of this data set on 4-factor model as compared to the categorization of the variables by Little et al. (2007) under 9 categorization.

Breast Cancer Data

The last dataset to which we apply the EM algorithm for the MFA model is the breast cancer dataset. This dataset is obtained from the University of Wisconsin Hospitals and used initially in Mangasarian and Wolberg (1990) paper. It is composed of $n = 569$ observations on 30 variables. The first group is called *malignant group* which has $n_1 = 212$ observations. The second group is called *benign group* has $n_2 = 357$ observations. For more details on this dataset, see the Appendix A. Table 7.23 shows the model selection results for this dataset by different information criteria using the EM algorithm with random, GARM and hybridized K-means initialization schemes. As shown in Table 7.23, AIC over estimates the number of mixtures and the number of factors. $CAIC$ and SBC pick the two mixtures model, but the number of factors appears to be still high. On the other hand, $ICOMP$ using GARM initialization picks $m = 2$ mixtures and $k = 9$ factors with correct classification error 90.51%. Therefore, we choose $\widehat{m} = 2$ and $\widehat{k} = 9$ MFA model as our best approximating model for this dataset.

7.4 Genetic EM (GEM) Algorithm

In this section, we apply our Genetic EM (GEM) algorithm for the MFA model on various simulated and real datasets to prevent us getting stuck in local maxima and

Table 7.23: Breast Cancer Data- MFA with EM Results.

	MFA Results					
IC	**Random Init.**		**GARM Init.**		**K-Means Init.**	
	MFA Model	**CCR**	**MFA Model**	**CCR**	**MFA Model**	**CCR**
AIC	$\widehat{m}=4,\widehat{k}=17$	31.63	$\widehat{m}=4,\widehat{k}=22$	37.43	$\widehat{m}=3,\widehat{k}=22$	93.32
CAIC	$\widehat{m}=2,\widehat{k}=16$	92.27	$\widehat{m}=2,\widehat{k}=21$	53.08	$\widehat{m}=2,\widehat{k}=18$	86.47
SBC	$\widehat{m}=2,\widehat{k}=16$	92.27	$\widehat{m}=3,\widehat{k}=22$	68.72	$\widehat{m}=2,\widehat{k}=20$	89.63
ICOMP$_{\text{CMISS}}$	$\widehat{m}=2,\widehat{k}=9$	92.44	$\widehat{m}=2,\widehat{k}=9$	90.51	$\widehat{m}=2,\widehat{k}=9$	88.40
ICOMP$_{\text{PEUMISS}}$	$\widehat{m}=2,\widehat{k}=9$	92.44	$\widehat{m}=2,\widehat{k}=9$	90.51	$\widehat{m}=2,\widehat{k}=9$	88.40
ICOMP$_{\text{PEULNMISS}}$	$\widehat{m}=1,\widehat{k}=15$	-	$\widehat{m}=2,\widehat{k}=9$	90.51	$\widehat{m}=2,\widehat{k}=9$	88.40

obtain better solutions. Moreover, we demonstrate the performances of information criteria to select the true model using the GEM algorithm in the MFA model.

7.4.1 Estimation of the Parameters

Here, we consider the S1 simulation structure, composed of $m^* = 2$ mixtures and $k^* = 3$ factors for each group to show how the parameters are estimated using GEM. Table 7.24 shows an example of the estimates obtained from using GEM in the MFA model for the true $m^* = 2$ mixtures and $k^* = 3$ factors. As can be seen, the estimates recover the true structure and are quite close to the true parameters values.

7.4.2 Model Selection Results Using the GEM Algorithm for the MFA Model

We used the simulation protocol given in the Appendix A with $n = 200$ observations. The information criteria are used as our fitness function in GEM algorithm to choose the best fitting model. We fit the MFA model for $m = 1, 2, ..., 4$ mixtures and $k = 1, 2, ..., 6$ factors and for this simulation data. After obtaining the initial partitions by the GARM method, information criteria are scored using GEM algorithm. Then, we select the best fitting model by minimizing the information criteria. With this structure, we executed 100 Monte-Carlo simulations to obtain the model selection

Table 7.24: Parameters Estimates by the GEM Algorithm.

m	$\widehat{\pi}_m$	$\widehat{\Lambda}_m$	$\widehat{\mu}_m$	$\text{diag}(\widehat{\Psi}_m)$
1	0.50	$\begin{bmatrix} 0.43 & 0.53 & 0.09 \\ 0.75 & -0.23 & 0.04 \\ 0.46 & 0.57 & 0.06 \\ 0.68 & -0.22 & 0.00 \\ 0.46 & 0.52 & 0.08 \\ 0.67 & -0.20 & 0.01 \\ 0.41 & 0.53 & 0.08 \\ 0.17 & 0.00 & 0.69 \\ 0.27 & 0.02 & 0.76 \\ 0.24 & 0.02 & 0.73 \end{bmatrix}$	$\begin{bmatrix} 16.95 \\ 17.13 \\ 16.97 \\ 17.11 \\ 16.98 \\ 17.22 \\ 17.00 \\ 16.98 \\ 16.93 \\ 16.99 \end{bmatrix}$	$\begin{bmatrix} 0.0714 \\ 0.0980 \\ 0.0410 \\ 0.0344 \\ 0.0536 \\ 0.1620 \\ 0.1596 \\ 0.1602 \\ 0.1326 \\ 0.1297 \end{bmatrix}$
2	0.50	$\begin{bmatrix} -0.09 & 0.76 & -0.19 \\ -0.08 & 0.74 & -0.12 \\ -0.08 & 0.75 & -0.12 \\ -0.77 & 0.06 & -0.07 \\ -0.70 & 0.06 & -0.05 \\ -0.71 & 0.06 & -0.05 \\ -0.72 & -0.01 & -0.03 \\ -0.20 & 0.20 & 0.80 \\ -0.25 & 0.21 & 0.81 \\ -0.21 & 0.30 & 0.88 \end{bmatrix}$	$\begin{bmatrix} 20.08 \\ 20.11 \\ 20.09 \\ 20.03 \\ 20.08 \\ 20.09 \\ 20.05 \\ 19.88 \\ 19.86 \\ 19.91 \end{bmatrix}$	$\begin{bmatrix} 0.0714 \\ 0.0980 \\ 0.0410 \\ 0.0344 \\ 0.0536 \\ 0.1620 \\ 0.1596 \\ 0.1602 \\ 0.1326 \\ 0.1297 \end{bmatrix}$

Table 7.25: Model Selection Frequency for GEM algorithm.

Information Criteria	# of hitting true model
AIC	100
CAIC	100
ICOMPC	100
SBC	100

frequency. Summary results are shown in Table 7.25. All four information criteria selected the true model with 100% frequency of choice. As can be seen from these results, the performances of information criteria are better using the GEM algorithm than the usual EM algorithm with random, hybridized K-Means or GARM initialization scheme to choose the true number of mixtures and the true number of factors in the MFA model.

7.4.3 Real Data Results Using the GEM Algorithm for the MFA Model

In this section, we executed the GEM algorithm to choose the model to the real datasets we considered before. GEM parameters used are as follows:

- Number of generations=30,
- Premature termination thresholds=40,
- Population size=20,
- Generation seeding=roulette,
- Crossover probability=0.75,
- Mutation probability=0.10
- Elitism=On.

We expect to have more stable results with this method since the genetic algorithm is able to search the entire solution landscape by preventing the EM algorithm from getting stuck in the local maxima.

College Data

In this data, colleges and universities are organized by how selective they can be for freshmen. Selectivity is determined by the test scores and high school class standing of applicants who enroll, plus the proportion of applicants who are accepted which are totally 9 different measurements. The data is obtained from $n_1 = 34$ the most selective schools and $n_2 = 89$ the more selective schools. Model selection results using the GEM algorithm are given in Table 7.26. AIC and $ICOMP$ type criteria pick the over estimated number of mixtures, but the number of factors selected by these criteria agree with $CAIC$ and SBC criteria. We note that the best approximating model for this data set is $\widehat{m} = 2$ mixtures and $\widehat{k} = 3$ factors with highest correct classification rate 94.31%. For the best approximating model ($\widehat{m} = 2$, $\widehat{k} = 3$), the variables are assigned to the factors as follows.
For most selective schools:

- *Factor 1*: Percentage of students who were in top 10% and 25% at high school class standing.
- *Factor 2*: ACT composite, 25th percentile and 75th percentile.
- *Factor 3*: Acceptance rate of applicants, SAT critical reading, 25th percentile and 75th percentile, SAT math, 25th percentile and 75th percentile.

For more selective schools:

- *Factor 1*: Percentage of students who were in top 10% and 25% at high school class standing.

Table 7.26: College Data- MFA with GEM Results.

IC	true # of mixtures	true # of factors	CCR
AIC	$\widehat{m} = 3$	$\widehat{k} = 3$	77.24
CAIC	$\widehat{m} = 2$	$\widehat{k} = 4$	94.31
SBC	$\widehat{m} = 2$	$\widehat{k} = 3$	94.31
$\textbf{ICOMP}_{\textbf{CMISS}}$	$\widehat{m} = 3$	$\widehat{k} = 3$	54.47
$\textbf{ICOMP}_{\textbf{PEUMISS}}$	$\widehat{m} = 3$	$\widehat{k} = 3$	43.90

- *Factor 2:* SAT critical reading, 25th percentile and 75th percentile, SAT math, 25th percentile and 75th percentile, ACT composite, 25th percentile and 75th percentile.
- *Factor 3:* Acceptance rate of applicants.

Wine Data

Note that this is the same wine dataset we studied before with $p = 13$ variables and $m = 3$ groups. Using the GEM algorithm, our results are shown in Table 7.29. Looking at the results, AIC picks the $\widehat{m} = 3$ mixtures with the highest correct classification rate 96.07%. $CAIC$ and SBC pick $\widehat{m} = 2$ mixtures with the highest correct classification rate 60.11%. Finally $ICOMP$ type criteria pick the $\widehat{m} = 2$ mixtures with the highest correct classification rate around 70%. Because of presence of overlap, either $\widehat{m} = 3$ mixtures and $\widehat{k} = 3$ factors, or $\widehat{m} = 2$ mixtures and $\widehat{k} = 6$ factors seem to be reasonable for this dataset. Considering the $(\widehat{m} = 3, \widehat{k} = 3)$ model, the variables are combined under the factors as follows.

- *Factor 1*: Alcohol, Total phenols, Flavonoids, Color intensity
- *Factor 2*: Nonflavanoid phenols, Hue , OD280/OD315 of diluted wines and Proline
- *Factor 3*: Malic acid, Ash, Alkalinity of ash, Magnesium and Proanthocyanins.

Table 7.27: Wine Data-MFA with GEM Results.

IC	true # of mixtures	true # of factors	CCR
AIC	$\widehat{m} = 3$	$\widehat{k} = 3$	96.07
CAIC	$\widehat{m} = 2$	$\widehat{k} = 2$	60.11
SBC	$\widehat{m} = 2$	$\widehat{k} = 2$	60.11
ICOMP$_{\text{CMISS}}$	$\widehat{m} = 2$	$\widehat{k} = 6$	70.22
ICOMP$_{\text{PEUMISS}}$	$\widehat{m} = 2$	$\widehat{k} = 6$	69.10

For the second mixture cluster, we can combine the variables as follows.

- *Factor 1*: Alcohol, Color intensity, Hue
- *Factor 2*: Ash, Alkalinity of ash, Magnesium, Flavonoids, Proanthocyanins and Proline
- *Factor 3*: Malic acid, Total phenols, Nonflavonoid phenols, OD280/OD315 of diluted wines.

For the third mixture cluster;

- *Factor 1*: Alcohol, Ash, Alkalinity of ash, Nonflavonoid phenols
- *Factor 2*: Magnesium, Color intensity
- *Factor 3*: Malic acid, Total phenols, Flavonoids, Proanthocyanins, Hue, Proline

Parkinson Data

Next, we come back to the Parkinson dataset of Little et al. (2007). This dataset is about parkinson disease (PD) has $n = 195$ observations ($n_1 = 48$ PD, $n_2 = 147$ non PD) individuals. We executed the GEM algorithm for $m = 1, ..., 4$ mixtures and $k = 1, ..., 15$ factors. Our results are summarized in Table 7.28. Looking at Table 7.28, AIC and $ICOMP$ criterion choose the over estimated number of mixtures for this dataset. $CAIC$ and SBC choose $\widehat{m} = 2$ mixtures, but CAIC picks $\widehat{k} = 11$

Table 7.28: Parkinson Data-MFA with GEM Results.

IC	true # of mixtures	true # of factors	CCR
AIC	$\widehat{m} = 4$	$\widehat{k} = 6$	42.56
CAIC	$\widehat{m} = 2$	$\widehat{k} = 11$	64.62
SBC	$\widehat{m} = 2$	$\widehat{k} = 6$	85.13
ICOMP$_{\text{CMISS}}$	$\widehat{m} = 3$	$\widehat{k} = 3$	42.56

Table 7.29: Breast Cancer Data-MFA with GEM Results.

IC	true # of mixtures	true # of factors	CCR
AIC	$\widehat{m} = 3$	$\widehat{k} = 22$	71.35
CAIC	$\widehat{m} = 3$	$\widehat{k} = 21$	80.32
SBC	$\widehat{m} = 3$	$\widehat{k} = 19$	86.99
ICOMP$_{\text{CMISS}}$	$\widehat{m} = 2$	$\widehat{k} = 9$	86.99
ICOMP$_{\text{PEUMISS}}$	$\widehat{m} = 2$	$\widehat{k} = 9$	86.99

factors which is pretty high for this dataset. The best approximating model is chosen by SBC at $\widehat{m} = 2$ mixtures and $\widehat{k} = 6$ factors with the highest correct classification rate 85.13%.

Breast Cancer Data

Finally, we revisit again the breast cancer dataset already analyzed using the EM algorithm. Recall that there are $n = 569$ observations, $p = 30$ variables, and two groups. The groups are composed of $n_1 = 212$ patients with malignant tumors and $n_2 = 357$ patients with benign tumors. Using the GEM algorithm, our results are summarized in Table 7.29. Looking at Table 7.29, we see that AIC, $CAIC$ and SBC pick the $\widehat{m} = 3$ mixtures with high number of factors. Hovewer, $ICOMP$ type criteria pick the $\widehat{m} = 2$ mixture model and $\widehat{k} = 9$ factors for this dataset. This solution seems to be the best approximating model with correct classification rate 86.99%.

7.5 Two-Stage GEM Algorithm

This section demonstrates the Two-Stage Genetic EM algorithm for the MFA model on simulated and real datasets. As we discuss earlier, we propose this algorithm to achieve flexibility in our assumptions in order to be able to obtain different number of factors across mixture of clusters. In the first stage, we discover the number mixture clusters, and then for each mixture we obtain the best approximating number of factors. In the second stage, we maximize the log likelihood function of the MFA model and using the information criteria we obtain the final number of factors and the covariance matrix of the random errors for each mixture cluster. In this dissertation, we used GARM algorithm to discover the mixtures and Kaiser criterion to obtain the best approximating number of factors for each mixture in the first stage of the algorithm. After giving an example in recovering the parameter estimates using this method, the performances of information criteria to select the true number of factors and true number of mixtures are compared using the Two-Stage GEM algorithm in MFA model. Finally, we do model selection on the real datasets used before using our Two-Stage GEM algorithm and information criteria as the fitness function.

7.5.1 Estimation of the Parameters

We generated the data from $m = 2$ mixtures and $k = 2$ factors MFA model given in Appendix A with $n = 200$ observations. With this structure, a simulation is executed to obtain an example of the estimates using the Two-Stage Genetic EM algorithm for true number of mixture ($m^* = 2$). Here, we cannot control the number of factors since in this case the number of factors are not considered to be the same, but varying in each mixture. Our analysis identifies three factors in both mixtures as can be seen in Table 7.30. The estimates recover the true structure and are close to the true parameter values. Note that, we achieve flexibility in our assumptions in MFA model in order to be able to obtain different Ψ across the mixture of clusters using the Two-Stage GEM algorithm.

Table 7.30: Estimated Parameters by Two-Stage Genetic EM Algorithm.

m	$\widehat{\pi}_m$	$\widehat{\Lambda}_m$	$\widehat{\mu}_m$	$\text{diag}(\widehat{\Psi}_m)$
1	0.50	$\begin{bmatrix} -0.60 & 0.21 & 0.05 \\ -0.32 & -0.57 & 0.09 \\ -0.54 & 0.21 & -0.07 \\ -0.34 & -0.57 & 0.09 \\ -0.59 & 0.22 & -0.09 \\ -0.32 & -0.53 & 0.07 \\ -0.56 & 0.18 & -0.05 \\ -0.09 & -0.21 & -0.57 \\ -0.15 & -0.17 & -0.55 \\ -0.14 & -0.23 & -0.52 \end{bmatrix}$	$\begin{bmatrix} 16.98 \\ 16.91 \\ 17.08 \\ 16.88 \\ 17.04 \\ 16.86 \\ 17.08 \\ 16.99 \\ 17.01 \\ 16.99 \end{bmatrix}$	$\begin{bmatrix} 0.09 \\ 0.09 \\ 0.04 \\ 0.04 \\ 0.04 \\ 0.17 \\ 0.18 \\ 0.15 \\ 0.11 \\ 0.10 \end{bmatrix}$
2	0.50	$\begin{bmatrix} 0.76 & 0.25 & 0.18 \\ 0.71 & 0.26 & 0.16 \\ 0.70 & 0.23 & 0.17 \\ 0.05 & 0.68 & -0.38 \\ 0.06 & 0.74 & -0.39 \\ 0.04 & 0.71 & -0.37 \\ 0.02 & 0.72 & -0.44 \\ -0.22 & 0.49 & 0.56 \\ -0.15 & 0.49 & 0.56 \\ -0.16 & 0.48 & 0.53 \end{bmatrix}$	$\begin{bmatrix} 19.98 \\ 19.93 \\ 20.04 \\ 19.88 \\ 19.88 \\ 19.85 \\ 19.92 \\ 19.98 \\ 20.00 \\ 19.98 \end{bmatrix}$	$\begin{bmatrix} 0.09 \\ 0.11 \\ 0.03 \\ 0.05 \\ 0.03 \\ 0.17 \\ 0.18 \\ 0.15 \\ 0.11 \\ 0.10 \end{bmatrix}$

7.5.2 Model Selection Using the Two-Stage GEM Algorithm for the MFA Model

We ran 100 Monte-Carlo simulations to obtain the hit ratios by information criteria using the Two-Stage GEM algorithm. We fit $m = 1, 2, ..., 4$ mixtures using simulation (S1) protocol with $n = 200$ observations. The model selection frequencies by four different information criteria out of 100 simulations are given in Table 7.31. Because we could obtain different number of factors in each mixture with this method, we showed the frequency of selecting the true number of mixture ($m^* = 2$) and true number of factors ($k^* = 3$ for each mixture) separately. As can be seen Table 7.31, all of the information criteria choose the true model with over 95%, the true number of mixture with over 98%. $CAIC$, SBC, and $ICOMP_C$ choose the true model with the highest frequencies in the MFA model. AIC has the lowest percentage in choosing both the true number of mixtures and the factors. It is not surprising to have such accurate results to select the true model, especially true number of mixtures by $CAIC$, $ICOMP_C$, and SBC using Two-Stage GEM algorithm. This is due to the fact that, we are not deciding the number of mixtures and number of factors within the same stage with this algorithm.

Table 7.31: Model Selection Frequency for Two-Stage EM algorithm of MFA.

Criteria	true # of mixture	true # of model
AIC	98	95
CAIC	100	99
ICOMP$_C$	100	99
SBC	99	99

7.6 Real Data Results Using the Two-Stage GEM Algorithm for the MFA Model

Here we apply the Two stage GEM algorithm on our real datasets. We choose the best MFA model by information criteria using this algorithm. In this section, we allow the number of factors to be different for each mixture cluster for the real dataset.

College Data

For the College dataset, we now use the Two stage GEM algorithm. Our results are summarized in Table 7.32. As can be seen in Table 7.32, AIC, $CAIC$ and SBC pick $\widehat{m} = 2$ mixtures with over 90% correct classification rate. $ICOMP$ type criteria pick the over estimated number of mixtures model. According to SBC, the best approximating model is chosen at $\widehat{m} = 2$ mixtures and $\widehat{k}_1 = 2$, $\widehat{k}_2 = 2$ factors with the correct classification rate 97.56%. Further, the correct classification rate appears to be higher with the Two Stage GEM algorithm as composed to the EM and the GEM algorithms for this dataset.

Wine Data

The Two-stage GEM algorithm results for the wine data set are given in Table 7.33. Recall that this dataset has $m = 3$ groups $p = 13$ variables. Looking at Table 7.33,

Table 7.32: College Data- MFA with Two Stage GEM algorithm.

IC	true # of mixtures	true # of factors	CCR
AIC	$\widehat{m} = 2$	$\widehat{k}_1 = 2, \widehat{k}_2 = 2$	97.56
CAIC	$\widehat{m} = 2$	$\widehat{k}_1 = 2, \widehat{k}_2 = 5$	92.68
SBC	$\widehat{m} = 2$	$\widehat{k}_1 = 2, \widehat{k}_2 = 2$	97.56
ICOMP$_{\mathbf{C}}$	$\widehat{m} = 3$	$\widehat{k}_1 = 1, \widehat{k}_2 = 2, \widehat{k}_3 = 2$	73.17
ICOMP$_{\mathbf{CMISS}}$	$\widehat{m} = 4$	$\widehat{k}_1 = 2, \widehat{k}_2 = 1, \widehat{k}_3 = 2, \widehat{k}_4 = 1$	49.59
ICOMP$_{\mathbf{PEU}}$	$\widehat{m} = 4$	$\widehat{k}_1 = 1, \widehat{k}_2 = 2, \widehat{k}_3 = 2, \widehat{k}_4 = 2$	59.35
ICOMP$_{\mathbf{PEUMISS}}$	$\widehat{m} = 4$	$\widehat{k}_1 = 2, \widehat{k}_2 = 2, \widehat{k}_3 = 1, \widehat{k}_4 = 1$	56.91

Table 7.33: Wine Data- MFA with Two Stage GEM algorithm.

IC	true # of mixtures	true # of factors	CCR
AIC	$\widehat{m}=2$	$\widehat{k}_1=4,\widehat{k}_2=4$	71.91
CAIC	$\widehat{m}=3$	$\widehat{k}_1=3,\widehat{k}_2=4,\widehat{k}_3=4$	74.16
SBC	$\widehat{m}=2$	$\widehat{k}_1=4,\widehat{k}_2=4$	72.47
ICOMP$_{\mathbf{C}}$	$\widehat{m}=3$	$\widehat{k}_1=4,\widehat{k}_2=4,\widehat{k}_3=4$	92.13
ICOMP$_{\mathbf{CMISS}}$	$\widehat{m}=3$	$\widehat{k}_1=2,\widehat{k}_2=4,\widehat{k}_3=4$	73.60
ICOMP$_{\mathbf{PEULN}}$	$\widehat{m}=3$	$\widehat{k}_1=5,\widehat{k}_2=4,\widehat{k}_3=5$	96.63
ICOMP$_{\mathbf{PEUMISS}}$	$\widehat{m}=3$	$\widehat{k}_1=3,\widehat{k}_2=4,\widehat{k}_3=3$	80.34
ICOMP$_{\mathbf{PEULNMISS}}$	$\widehat{m}=3$	$\widehat{k}_1=4,\widehat{k}_2=4,\widehat{k}_3=3$	83.15

we see that $CAIC$ and all the $ICOMP$ type criteria pick the $\widehat{m}=3$ mixtures model. AIC and SBC are not distinguished the $\widehat{m}=2$ mixtures since this dataset is highly overlapped. With $ICOMP_{PEULN}$, we choose $\widehat{m}=3$ mixtures and $\widehat{k}_1=5$, $\widehat{k}_1=4$, $\widehat{k}_1=5$ factors with the correct classification rate 96.63%.

Parkinson Data

Our penultimate example is the Parkinson dataset already evaluated by the EM and GEM algorithms. There are $p=22$ particular voice measurements and two groups (Parkinson disease and non-Parkinson disease) in this dataset. Looking at Table 7.34, AIC, $CAIC$ and SBC all agree on the same model. But, the number of mixtures is under estimated for this dataset based on AIC, $CAIC$ and SBC. The model selected by $ICOMP$ criteria is more reasonable. The best approximating model for this dataset seems to be $\widehat{m}=2$ mixtures and $\widehat{k}=3$ factors with correct classification rate 75.38%.

Breast Cancer Data

Finally, we have results from the Two-stage EM algorithm on the breast cancer dataset. This dataset is derived from benign and malignant tumors with $p=30$ variables. Our results are given in Table 7.35. Since this data set is clearly overlapped and non-normal in many dimensions, it is hard to distinguish the groups. None of

Table 7.34: Parkinson Data- MFA with Two Stage GEM algorithm.

IC	true # of mixtures	true # of factors	CCR
AIC	$\widehat{m}=1$	$\widehat{k}=4$	-
CAIC	$\widehat{m}=1$	$\widehat{k}=4$	-
SBC	$\widehat{m}=1$	$\widehat{k}=4$	-
$\mathbf{ICOMP_{CMISS}}$	$\widehat{m}=2$	$\widehat{k}_1=4,\widehat{k}_2=2$	69.23
$\mathbf{ICOMP_{PEUMISS}}$	$\widehat{m}=2$	$\widehat{k}_1=4,\widehat{k}_2=3$	75.38
$\mathbf{ICOMP_{PEULNMISS}}$	$\widehat{m}=2$	$\widehat{k}_1=4,\widehat{k}_2=3$	75.38

Table 7.35: Breast Cancer Data- MFA with Two Stage GEM algorithm.

IC	true # of mixtures	true # of factors	CCR
AIC	$\widehat{m}=1$	$\widehat{k}=6$	-
CAIC	$\widehat{m}=1$	$\widehat{k}=6$	-
SBC	$\widehat{m}=1$	$\widehat{k}=6$	-
$\mathbf{ICOMP_{C}}$	$\widehat{m}=1$	$\widehat{k}=6$	-
$\mathbf{ICOMP_{CMISS}}$	$\widehat{m}=3$	$\widehat{k}_1=6,\widehat{k}_2=5,\widehat{k}_3=5$	62.74
$\mathbf{ICOMP_{PEULNMISS}}$	$\widehat{m}=3$	$\widehat{k}_1=6,\widehat{k}_2=5,\widehat{k}_3=5$	62.74

the information criteria pick $m=2$ mixtures for this dataset using Two Stage GEM algorithm. But the dimension is reduced to 6 from 30 variables. According to AIC, $CAIC$, SBC, and $ICOMP_C$, the groups are homogenous in this dataset. Recall that $ICOMP_{CMISS}$ and $ICOMP_{PEULNMISS}$ have heavier penalty terms and penalize the MFA model more. The best approximating result based on $ICOMP_{CMISS}$ and $ICOMP_{PEULNMISS}$ is $\widehat{m}=3$ mixtures and $\widehat{k}_1=6,\widehat{k}_2=5,\widehat{k}_3=5$ factors with correct classification rate 62.74%.

Chapter 8

Conclusion

In this dissertation, we studied model selection problems in the Standard factor (SFA), Bayesian factor (BFA), and in Mixture of Factor Analyzers (MFA) models. We developed and introduced several information-theoretic model selection criteria commonly used in the literature in these areas of modern latent variable modeling to choose the number of factors in SFA and BFA models, and the number of mixture clusters and the number of factors in the MFA model simultaneously to resolve the problem of the reduction of the curse of dimensionality in a given dataset. In the thesis, we address the Heywood cases, or the improper solutions in the SFA model by introducing the BFA model. In the BFA model we learn the prior factor loading matrix using the Sparse Root algorithm data-adaptively along with Iterated Conditional Modes ICM), Gibbs sampling and the method of Press and Shigemasu (1989) to estimate the parameters of the BFA model using a fully Bayesian approach. In addition, we introduced and developed the information criteria in the BFA model to select the *best approximating model* among a set of candidate models at the posterior level. Although the natural choice is also the Bayes Factor (BF) in the BFA model, derivation of the BF is not a trivial exercise. Since BF has been already studied and compared along with the information criteria in the BFA model in a separate paper Turan and Bozdogan (2010), here we did not include the results using the BF.

In the MFA model, we addressed the high dependence of the solutions in choosing the number of mixtures and the number of factors upon the initial values, by introducing more intelligent Genetic Regularized Mahalanobis Distance (GARM) and hybridized K-means initialization to obtain better initial values for the EM algorithm as opposed to the random initialization scheme used by by Ghahramani and Hinton (1997) in the MFA model. Within the framework of the MFA model, we also addressed the numerical instability issues and manifestation of singular or ill-conditioned covariance estimators by introducing smoothed covariance estimators and stabilization of the eigenvalues of the mixture model covariance matrices. Further, we developed the genetic EM algorithm (GEM) and also Two-Stage Genetic EM algorithms to relax the spurious assumption in the number of factors across the mixture clusters be the same. In this manner, we are now able to choose different number of mixtures and at the same time different number of factors across each of the mixture clusters. To our knowledge, this was not possible before. To illustrate our results, in this thesis, we demonstrated our results across SFA, BFA, and the MFA models on both simulated and several real datasets with varying degrees of overlap and dimensionality.

What did we learn from this thesis? What kind of guidance can we give to the readers? These are legitimate questions to ask.

What we have learned from this thesis and our results is that, it is difficult to derive the usual likelihood ratio type of criteria especially in the MFA model since the likelihood function is changing in choosing the number of mixtures and at the same time the number of factors. Therefore, the development and the use of information criteria resolve intrinsically such problems in the conventional statistical procedures. Indeed, as illustrated in our numerical results, information criteria work quite well in both simulated and the real data examples. We learned that the computational time and the complexity of the results vary according to the datasets we used and their compactness, orientation, and overlap, as well as their dimensionality. Larger the dimensionality and existence of overlaps across the clusters, make the analysis

more complex. Therefore, we suggest high performance computing in the analysis of the MFA model.

As a guidance to the readers, in Figure 8.1 we outline the structure of the flow of this thesis in a learning tree diagram.

In conclusion, we believe that this research can be implemented and to solve complex data mining problems easily. For future work, we are planning to provide an easy to use user interface for the MFA model code (those are regular EM , EM with hybridized K-means initialization, EM with GARM initialization, GEM (with robust covariance estimators), and Two-Stage GEM (with robust covariance estimators). Therefore, these methods will be implemented in a future research. We plan to compare the performance in selecting the true model with the Bayesian mixture factor analyzers proposed by Ghahramani and Beal (2003). They argue that the number of components and the local dimensionality of each component can be obtained without overfitting with a Bayesian mixtures of factor analytic model. However, we have not seen any new and convincing work done in this direction. This is still an open problem.

Further, to relax the linearity in the latent variables, we shall introduce and study yet another novel learning approach using what is known as the kernel-based learning algorithms by embedding (or transforming) the data into a Reproducing Kernel Hilbert Space (RKHS), and searching for linear relations in such a space. The embedding is performed implicitly, by specifying the inner product between each pair of points of the data rather than by giving their coordinates explicitly. This approach has several advantages. The most important of them is that the inner product in the embedding space (or feature space) can often be computed much more easily than the coordinates of the points themselves. Thus, a non-linear model is built in two steps: use a fixed non-linear mapping to transform the data into a feature space, and then use a linear model to carry out the mixtures of factor analyzers (MFA) model in the feature space, without using explicit non-linear functions.

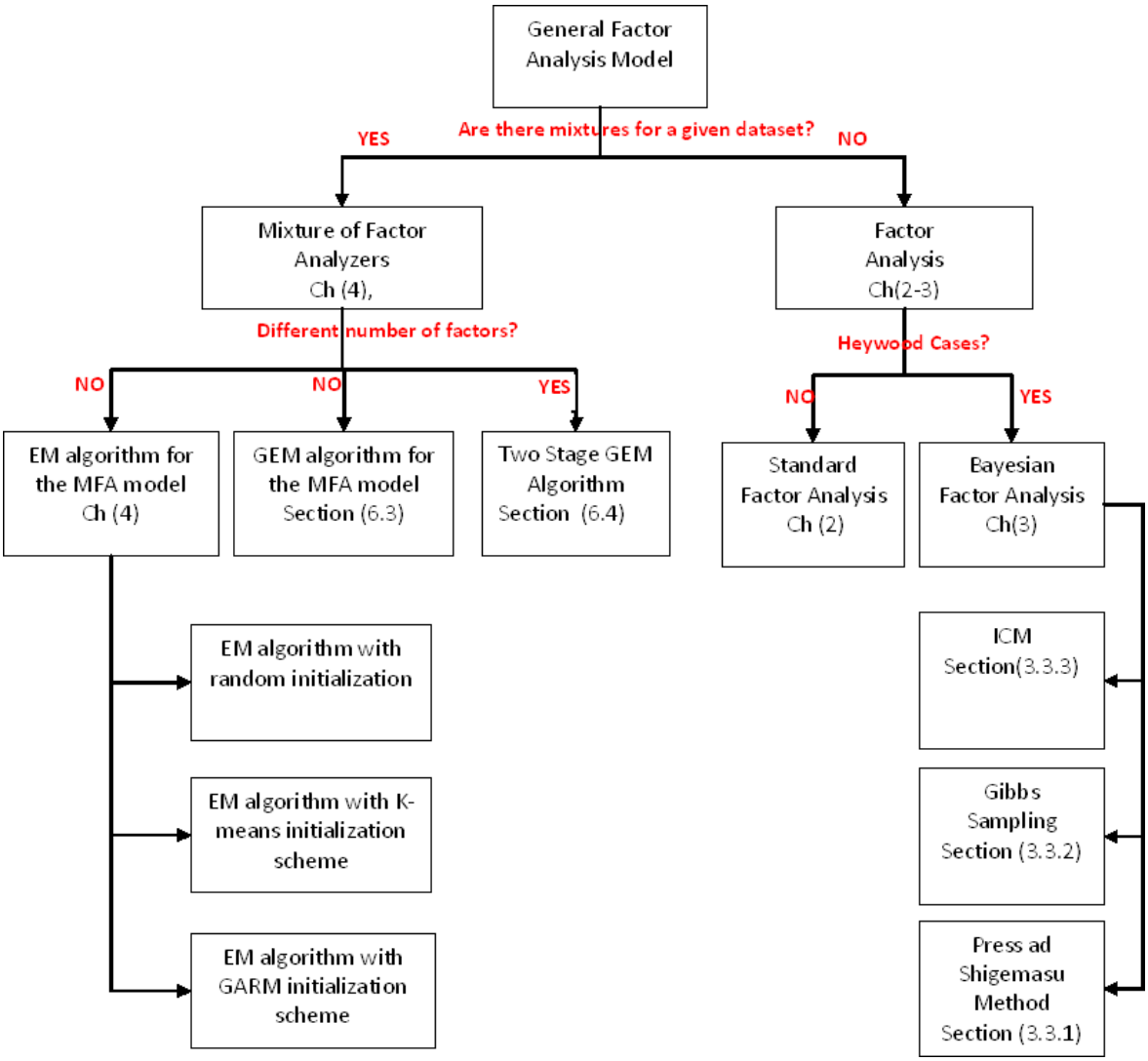


Figure 8.1: Summary of a Learning Tree of the Dissertation.

Bibliography

Bibliography

Aeberhard, S., C. D. and De vel, O. (1992). Comparison of classifiers in high dimensional settings. Technical Report 92-02 174, Dept. of Computer Science and Dept. of Mathematics and Statistics, James Cook University of North Queensland. 119

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. *Second International Symposium on Information Theory*, pages 267–281. 35

Akaike, H. (1987). Factor analysis and aic. *Psychometrika*, 52(3):317–332. 10, 40

Alba, E. and Dorronsoro, B. (2008). *Cellular Genetic Algorithm*. Springer. 52

Ansari, A. and Jedidi, K. (2000). Bayesian factor analysis for multilevel binary observations. *Psychometrika*, 65(4):475–496. 11

Arminger, G. and Muthen, B. (1998). A bayesian approach to nonlinear variable models using the gibbs sampling and the metropolis-hastings algorithm. *Psychometrika*, 63(3):271–300. 11

Blâfield, E. (1980). Clustering of observations from finite mixtures with structural information. *Jyvaskyla Studies in Computer Science,Economics and Statistics 2*. 23

Bozdogan, H. (1983). Determining the number of component clusters in the standardmultivariate normal mixture model using model-selection criteria. Technical Report UIC/DQM/A83-1 ARO Contract DAAG29-82-K-0155, Quantitative Methods Department,University of Illinois at Chicago. 29

Bozdogan, H. (1987). Model selection and akaike's information criterion (aic):the general theory and it's analytical extensions. *Psychometrika*, 52:345–370. 33, 35, 42

Bozdogan, H. (1988). Icomp: A new model-selection criteria. in bock, h., editor, classification and related methods of data analysis. *North-Holland*, 18. 33, 36, 37, 42

Bozdogan, H. (1990). On the information-based measure of covariance complexity and its application to the evaluation of multivariate linear models. *Communications in Statistics : Theory and Methods*, 19:221–278. 33, 36

Bozdogan, H. (1994). Choosing the number of clusters, subset selection of variables, and outlier detection in the standard mixture-model cluster analysis. In et al., E. D., editor, *New Approaches in Classification and Data Analysis*, pages 169–177. Springer-Verlag, New York. 33, 36, 43

Bozdogan, H. (2000). Akaike's information criterion and recent developments in information complexity. *Journal of Mathematical Psychology*, 44:62–91. 33

Bozdogan, H. (2004). *Statistical Data Mining Knowledge Discovery*, chapter Intelligent statistical data mining with information complexity and genetic algorithms, pages 15–56. ChapmanHall,CRC. 33, 36

Bozdogan, H. (2010). *Information Complexity in Multivariate Learning and High Dimensional Data Mining.* Chapman and Hall/CRC. Book in Preparation. 38, 41, 44

Bozdogan, H. and Haughton, D. (1998). Informational complexity criteria for regression models. *Computational Statistics and Data Analysis*, 28(19):51–76. 37

Bozdogan, H. and Ramirez, D. E. (1986). An expert model selection approach to determine the best pattern structure in factor analysis models. In Bozdogan H., G. A. K., editor, *Proceeding of the Advanced Symposium on Multivariate Modeling and Data Analysis*, pages 37–57. D Reidel. 9

Bozdogan, H. and Shigemasu, K. (1998). Bayesian factor analysis model and choosing the number of factors using a new informational complexity criterion. In Rizzi, A., V. M. and Bock, H.-H., editors, *Advances in Data Science and Classification*, pages 335–342. Springer. 10

Chipperfield, A. (1997). *Genetic Algorithm in Engineering Systems*, chapter Introduction to genetic algorithms, pages 1–45. The Institution of electrical Engineers. 52

Cho, D. and Zhang, B. (2002). Evoluationary optimization by distribution extimation with mixtures of factor analyzers. In *Proceedings of the 2002 Congress on Evolutionary Computation*, volume 2, pages 2029–2034. 24, 61

Deniz, E. and Bozdogan, H. (2010). Performance of information complexity criteria in structural equation models with applications. 33, 38

Fokouè, E. (2005). Mixtures of factor analyzers:an extension with covariates. *Journal of Multivariate Analysis*, 95:370–384. 24, 61

Gelfand, A. and Smith, A. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85:972–985. 19

Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions and the bayesian restoration of images. *IEEE Transactions of Pattern Analysis and Machine Intelligent*, 6:721–741. 19

Ghahramani, Z. and Hinton, E. G. (1997). The em algorithm for mixture of factor analyzers. *Technical Report CRG-TR*, 96-1. v, 2, 24, 28, 79, 103, 116

Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, New York. 52

Hartigan, J. (1975). *Clustering Algorithms*. John Wiley and Sons, New York. 15

Holland, J. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press. 50

Holland, J. (1992). Genetic algorithms. *Scientific American*, 31:66–72. 50

Howe, J. (2009). *A New Generation of Mixture-Model Cluster Analysis with Information Complexity and the Genetic EM Algorithm*. Phd thesis, University of Tennessee. 57, 62

Jöreskog, K. G. (1971). Simultaneous factor analysis in several populations. *Psychometrika*, 57:409–426. 22

Kaiser, H. F. (1960). The application of electronic computers to factor analysis. *Educational and Psychological Measurement*, 20:141–151. 68

Kano, Y., Berkane, M., and Bentler, P. M. (1990). Covariance structure analysis with heterogeneous kurtosis parameters. *Biometrika*, 77(3):575–585. 22

Krishna, K. and Murty, M. (1999). Genetic k-means algorithm. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 29(3):433–439,. 58

Kullback, A. and Leibler, R. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86. 33

Lee, S. (1981). A bayesian approach to confirmatory factor analysis. *Psyhchometrika*, 46:153–160. 10

Lee, S.-Y. and Tsui, K.-L. (1982). Covariance structure analysis in several population. *Psychometrika*, 47:297–308. 22

Lindley, D. (1956). On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005. 38

Lindley, D. V. and Smith, A. F. M. (1972). Bayes estimates for the linear model. *Journal of the Royal Statistical Society*, 34:1–41. 20

Little, M., McSharry, P., Roberts, S., Costello, D., and Moroz, I. (2007). Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. *BioMedical Engineering*. 86, 87, 88, 94, 122

Lopes, H. and West, M. (2004). Bayesian model assesment in factor analysis. *Statistica Sinica*, 14:41–67. 10

MacLachan, G. and Peel, D. (2000). Mixture of factor analyzers. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 599–606. 24

MacLachan, G., Peel, D., and Bean, R. (2003). Modelling high-dimensinal data by mixtures of factor analyzers. *Computational Statistics and Data Mining*, 41:379–388. 61

MacLachlan, G., Peel, D., and Bean, R. (2002). Modelling high-dimensinal data by mixtures of factor analyzers. *Computational Statistics and Data Mining*, 95:370–384. 24

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *In Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, page 281297. University of California, Berkeley. 28

Mangasarian, O. L. and Wolberg, W. H. (1990). Cancer diagnosis via linear programming. *SIAM News*, 23(5):1:18. 88, 125

Mao, J. and Jain, A. (1996). A self-organizing network for hyperellipsoidal clustering (hec). *IEEE Transactions on Neural Networks*, 7(1):16–29. 55

Martin, J. K. and McDonald, R. P. (1975). Bayesian estimation in unrestricted factor analysis. *Psyhchometrika*, 40(4):505–517. 9, 10

Meng, X. and van Dyk, D. (1997). Em algorithm-an old folk song sung to a fast new tune. *Journal of royal Statistics Soc. Series B*, 59:511–567. 61

Mitchell, M. (1998). *Introduction to Genetic Algorithms*. Massachusetts Institue of Technology. 54

Muthén, B. O. (1989). Latent variable modeling in heterogenous populations. *Psychometrika*, 54:557–585. 22

Newsom, J. T. (2005). *A Quick Primer on Exploratory Factor Analysis*. 7

Park, D.-H. (2009). *Data Adaptive Kernel Discriminant Analysis Using Information Complexity Criterion and Genetic Algorithm*. PhD thesis, The University of Tenneessee, Knoxville. 121

Poskitt, D. (1987). Precision, complexity and bayesian model determination. *Journal of the Royal Statistical Society, Series B (Methodological)*, 49(2):199–208. 38

Press, S. J.and Shigemasu, K. (1997). Bayesian inference in factor analysis-revised. Technical report, Department of Statistics, University of California Riverside. 10

Press, S. and Shigemasu, K. (1989). Bayeasian inference in factor analysis. In Gleser, L., Perlman, M., Press, S. J., and Sampson, A., editors, *Contributions to Probability and Statistics: Essay in Honor of Ingram Olkin*, pages 271–287, Verlag. Springer. 2, 10, 17, 102

Press, S.J., S. K. and Lee, S. (2008). Emprical bayes assesment of the hyperparameters in bayesian factor analysis. 10

Rowe, B. D. and Press, S. J. (1998). Gibbs sampling and hill climbing in bayesian factor analysis. Technical report, Department of Statistics, University of California Riverside. 2, 10

Rubin, D. B. and Thayer, D. T. (1982). Em algorithm for ml factor analysis. *Psychometrika*, 47:69–76. 5

Salah, A. and Alpaydin, E. (2004). Incremental mixtures of factor analysers. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR04)*, volume 1. 61

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, 6:461–464. 35

Song, W., F. M. W. S. and Shaowei, X. (1997). The hyperellipsoidal clustering using genetic algorithm. In *IEEE Int. Conf. on Intelligence Processing Systems*, pages 592–596. 57, 58

Sörborn, D. (1974). A general method for studying difference in factor means and factor structure between groups. *British Journal of Mathematical and Statistical Psychology*, 27:229–239. 22

Teknomo, K. (2007). *K-Means Clustering Tutorial*. 29

Thomaz, C. (2004). *Maximum Entropy Covariance Estimate for Statistical Pattern Recognition*. Phd thesis, University of London and for the Diploma of the Imperial College (D.I.C.). 47

Turan, E. and Bozdogan, H. (2010). Bayesian factor analysis using information complexity with gibbs sampling and iterated conditional modes (icm) algorithms. In *Multivariate High Dimensional Data Mining*. 102

Van Emden, M. (1971). *An Analysis of Complexity*. Mathematical Centre Tracts. 36

Vlassis, N. and Likas, A. (2002). A greedy em algorithm for gaussian mixture learning. *Neural Processing Letters*, 15(1):77–87. 27, 61

Wicker, J. (2006). *Applications of modern statistical methods to analysis of data in physical science*. Phd thesis, University of Tennessee. 58, 61

Xu, L. and Jordan, M. (1996). On convergence properties of the em algorithm for gaussian mixtures. *Neural Computation*, 8(1):129–151. 27

Yung, Y.-F. (1997). Finite mixtures in confirmatory factor analysis models. *Psychometrika*, 62:297–330. 22, 24

Zhou, X. and Liu, X. (2008). The em algorithm for the extended finite mixture of the factor analyzers model. *Computational Statistics and Data Mining*, 52:3939–3953. 24, 61

Appendix

Appendix A

Data Sets

A.1 Simulated Data-S1

The dataset is generated using the multivariate normal distribution with $p = 10$ variables $n = 100$ observations. There are $m = 2$ populations, and each population has $k = 3$ factors. The number of factor and Ψ are selected the same for each population to satisfy the assumption of the EM algorithm purposed by Ghahramani and Hinton (1997). The parameters of each population are given in the table A.1.

Table A.1: Simulation 1 - Data Generation Parameters of Mixture of Factor Analyzers.

M	π_M	Λ_M	μ_M	diag(Ψ_M)
1	0.5	$\begin{bmatrix} 0.7 & 0.1 & 0.1 \\ 0.1 & 0.7 & 0.1 \\ 0.7 & 0.1 & 0.1 \\ 0.1 & 0.7 & 0.1 \\ 0.7 & 0.1 & 0.1 \\ 0.1 & 0.7 & 0.1 \\ 0.7 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.7 \\ 0.1 & 0.1 & 0.7 \end{bmatrix}$	$\begin{bmatrix} 17 \\ 17 \\ 17 \\ 17 \\ 17 \\ 17 \\ 17 \\ 17 \\ 17 \\ 17 \end{bmatrix}$	$\begin{bmatrix} 0.1024 \\ 0.1024 \\ 0.0400 \\ 0.0400 \\ 0.0400 \\ 0.1600 \\ 0.1600 \\ 0.1600 \\ 0.1600 \\ 0.1600 \end{bmatrix}$
2	0.5	$\begin{bmatrix} 0.9 & 0.1 & 0.1 \\ 0.9 & 0.1 & 0.1 \\ 0.9 & 0.1 & 0.1 \\ 0.1 & 0.9 & 0.1 \\ 0.1 & 0.9 & 0.1 \\ 0.1 & 0.9 & 0.1 \\ 0.1 & 0.9 & 0.1 \\ 0.1 & 0.1 & 0.9 \\ 0.1 & 0.1 & 0.9 \\ 0.1 & 0.1 & 0.9 \end{bmatrix}$	$\begin{bmatrix} 20 \\ 20 \\ 20 \\ 20 \\ 20 \\ 20 \\ 20 \\ 20 \\ 20 \\ 20 \end{bmatrix}$	$\begin{bmatrix} 0.1024 \\ 0.1024 \\ 0.0400 \\ 0.0400 \\ 0.0400 \\ 0.1600 \\ 0.1600 \\ 0.1600 \\ 0.1600 \\ 0.1600 \end{bmatrix}$

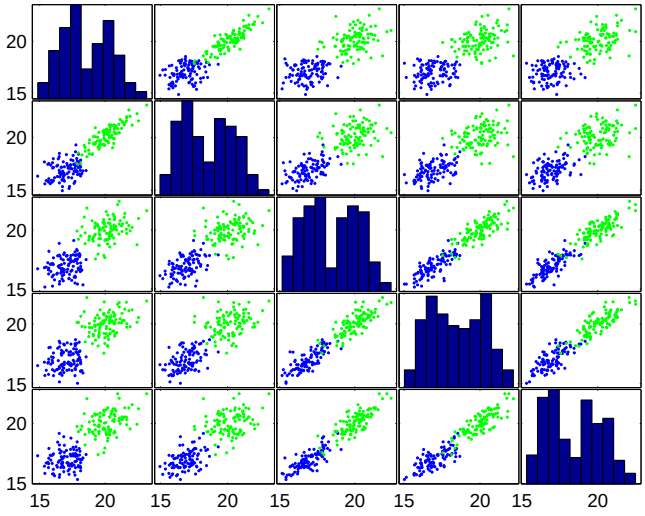


Figure A.1: Simulated Data- Grouped Scatter Plot for X1,..X5

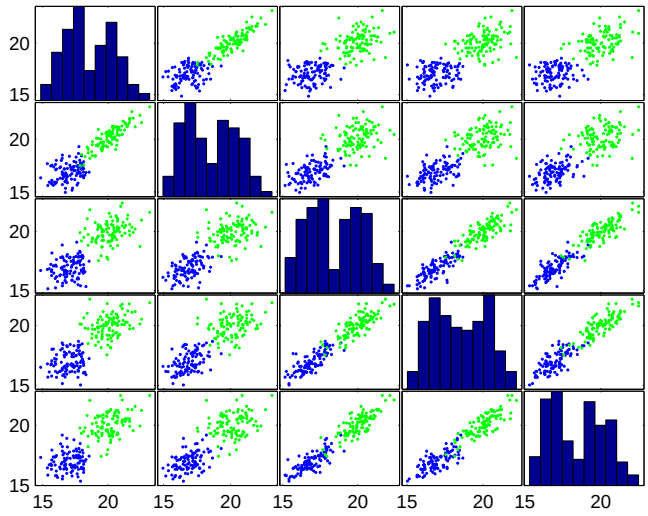


Figure A.2: Simulated Data-Grouped Scatter Plot for X6,..X10

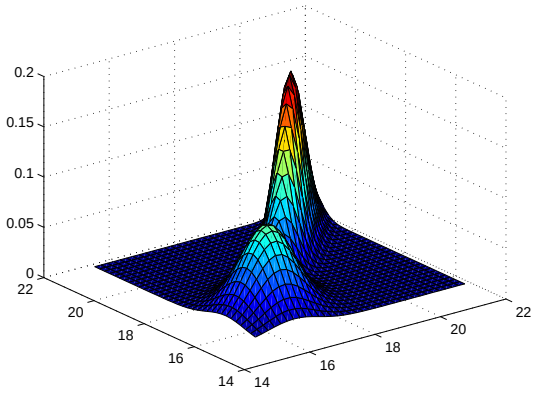


Figure A.3: Simulated Data- Surface Plot

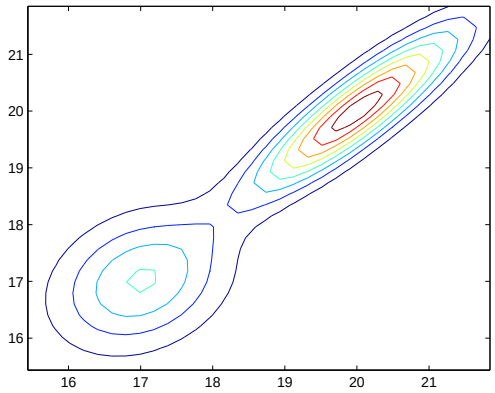


Figure A.4: Simulated Data- Contour Plot

A.2 Real Data

A.2.1 Wine Data

This data set of *Forina, M. et al* used in Aeberhard and De vel (1992).The data was used with many others for comparing various classifiers. This dataset includes a chemical analysis of $n = 178$ wines grown in the same region in Italy. It is derived from $m = 3$ different wines ($n_1 = 59$, $n_2 = 71$, $n_3 = 48$). The analysis determined the quantities of 13 constituents found in each of the three types of wines. Those variables are listed as follows:

x_1=Alcohol	x_8=Nonflavanoid phenols
x_2=Malic acid	x_9=Proanthocyanins
x_3=Ash	x_{10}=Color intensity
x_4=Alcalinity of ash	x_{11}=Hue
x_5=Magnesium	x_{12}=OD280/OD315 of diluted wines
x_6=Total phenols	x_{13}=Proline
x_7=Flavanoids	
Source http://archive.ics.uci.edu/ml/datasets/Wine	

As can be seen in Figure A.5 and A.6, this dataset substantial overlap of the three groups and has non-normal distribution in many dimensions.

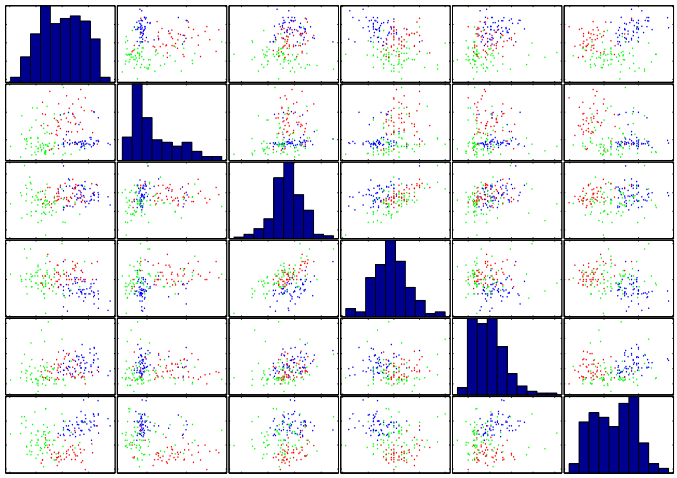


Figure A.5: Wine data - Grouped Scatterplot Matrix for x1 . . . x6.

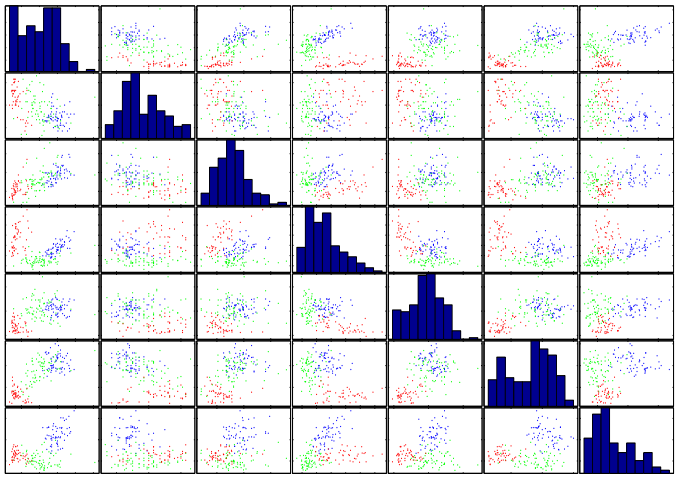


Figure A.6: Wine data - Grouped Scatterplot Matrix for x7 . . . x13.

A.2.2 College Data

This dataset is provided by U.S News and World Report (2008) about college selectivity. Originally there were 139 observations but we delete 16 observations with missing variables. Among 123 observations, $n_1 = 34$ colleges and universities are categorized as the most selective schools. The other $n_2 = 89$ colleges and universities are categorized as the more selective schools. Therefore, prior probability for first group is 27.64%, and second group is 72.36%. Park (2009) separates the groups in this dataset using Kernel Discriminant Analysis Using Information Complexity Criterion and Genetic Algorithm. In this data set, colleges and universities are organized by how picky they can be in choosing freshmen. Selectivity is determined by the test scores and high school class standing of applicants who enroll, plus the proportion of applicants who are accepted. Most of the $p = 9$ variables have very high correlations with each other. There are 9 variables listed as follows

x_1=Acceptance rate of applicants
x_2=SAT critical reading, 25th percentile
x_3=SAT critical reading, 75th percentil
x_4=SAT math, 25th percentile
x_5=SAT math, 75th percentile
x_6=ACT composite, 25th percentile
x_7=ACT composite, 75th percentile
x_8=Percentage of students who were in top 10% at high school class standing
x_9=Percentage of students who were in top 25% at high school class standing

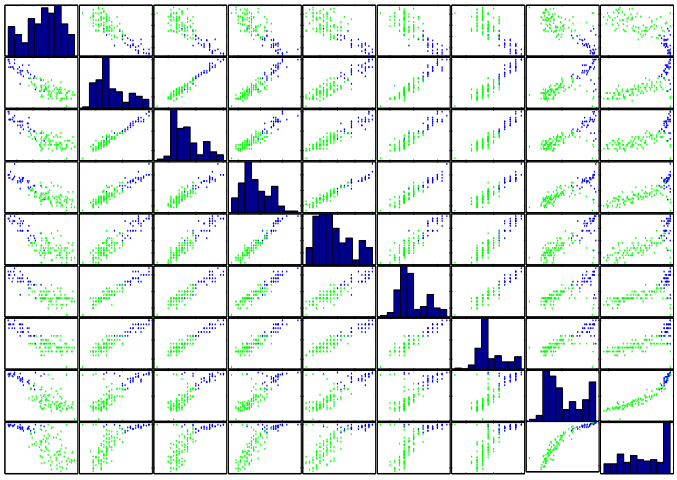


Figure A.7: College data - Grouped Scatterplot Matrix for x1 . . . x9

A.2.3 Parkinson Data

The dataset was created by Little et al. (2007) in collaboration with the National Center for Voice and Speech, Denver, Colorado. This dataset is composed of a range of biomedical voice measurements from 31 people, 23 with Parkinson's disease (PD). Each variable is a particular voice measure, and each observation corresponds one of $n = 195$ ($n_1 = 48$ PD $n_2 = 147$ non PD) voice recording from these individuals. Totally 22 variables in this dataset are listed as follows:

x_1=MDVP:Fo(Hz)	x_{12}=Shimmer:APQ5
x_2=MDVP:Fhi(Hz)	x_{13}=MDVP:APQ
x_3=MDVP:Flo(Hz)	x_{14}=Shimmer:DDA
x_4=MDVP:Jitter	x_{15}=NHR
x_5=MDVP:Jitter(Abs)	x_16=HNR
x_6=MDVP:RAP	x_{17}=RPDE
x_7=MDVP:PPQ	x_{18}=D2
x_8=Jitter:DDP	x_{19}=DFA
x_9=MDVP:Shimmer	x_{20}=spread1
x_10=MDVP:Shimmer(dB)	x_{21}=spread2
x_{11}=Shimmer:APQ3	x_{22}=PPE
Source:http://archive.ics.uci.edu/ml/datasets/Parkinsons	

Those variables are defined and categorized by creators;

- **Average vocal fundamental frequency**: MDVP:Fo(Hz),
- **Maximum vocal fundamental frequency**: MDVP:Fhi(Hz),
- **Minimum vocal fundamental frequency**: MDVP:Flo(Hz),
- **Measures of variation in fundamental frequency**: MDVP: Jitter(%), MDVP:Jitter(Abs), MDVP:RAP, MDVP:PPQ, Jitter:DDP,
- **Measures of variation in amplitude**: MDVP:Shimmer,MDVP:Shimmer(dB), Shimmer:APQ3, Shimmer: APQ5, MDVP:APQ, Shimmer:DDA,
- **Ratio of noise to tonal components in the voice**: NHR,HNR,
- **Nonlinear dynamical complexity measures**: RPDE, D2,
- **Signal fractal scaling exponent**: DFA,
- **Nonlinear measures of fundamental frequency variation**: Spread1, Spread2, PPE,

Figure A.8 through A.10 shows the group scatter plot of the dataset. It is very challenging data set since this dataset is not normal in many dimensions according to the shape of histogram and the groups are overlapped.

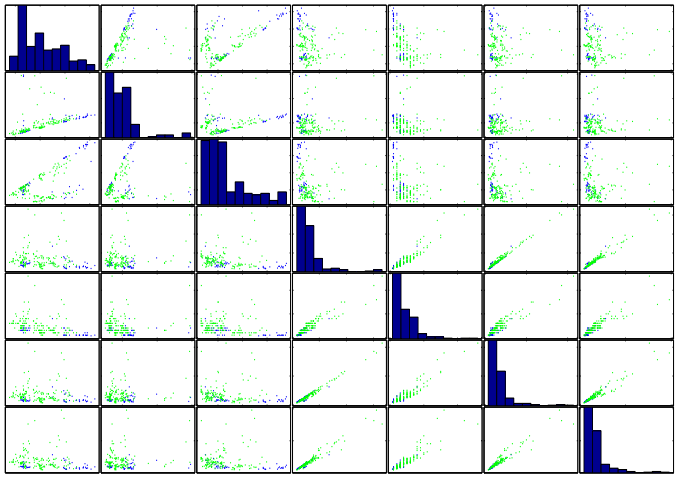


Figure A.8: Parkinson data - Grouped Scatterplot Matrix for x1 . . . x7.

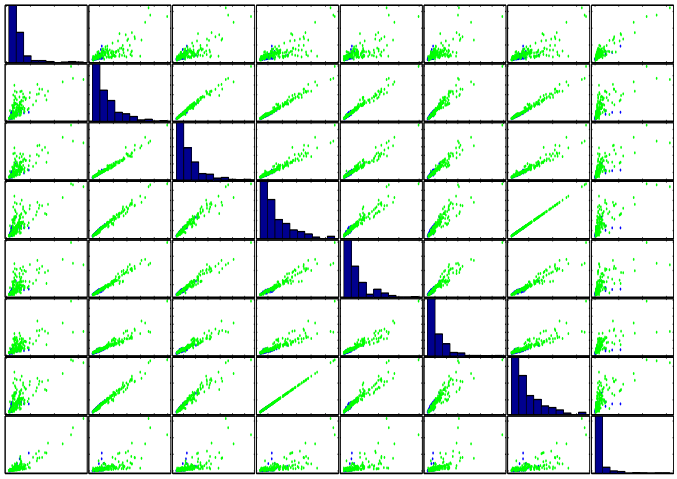


Figure A.9: Parkinson data - Grouped Scatterplot Matrix for x8 . . . x15.

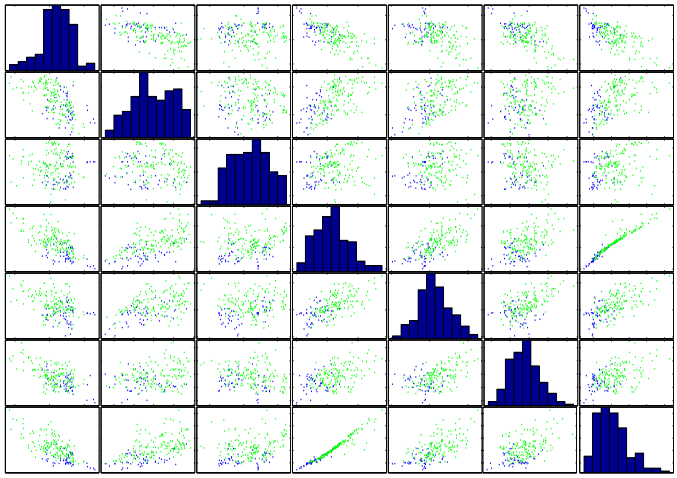


Figure A.10: Parkinson data - Grouped Scatterplot Matrix for x16 . . . x22

A.2.4 Breast Cancer Data

This breast cancer databases was obtained from the University of Wisconsin Hospitals and used first time in Mangasarian and Wolberg (1990) paper. Variables are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. This dataset is composed of $n = 569$ observations from 30 variables. First group is called *malignant group* has 212 observations. Second group is called *benign group* has 357 observations. Therefore, we have two groups and their prior probabilities are respectively 37.26% and 67.74%. The group scatter plot is given in figure A.11 through A.14. This dataset is not normal in many dimensions according to histogram. 10 variables and their calculation are listed as follows:

x_1=radius-mean of distances from center to points on the perimeter
x_2=texture- standard deviation of gray-scale values
x_3=perimeter
x_4=area
x_5=smoothness-local variation in radius lengths
x_6=compactness-$perimeter^2$ / area - 1.0
x_7=concavity -severity of concave portions of the contour
x_8=concave points- number of concave portions of the contour
x_9=symmetry
x_{10}=fractal dimension - "coastline approximation" -1
Source: http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29

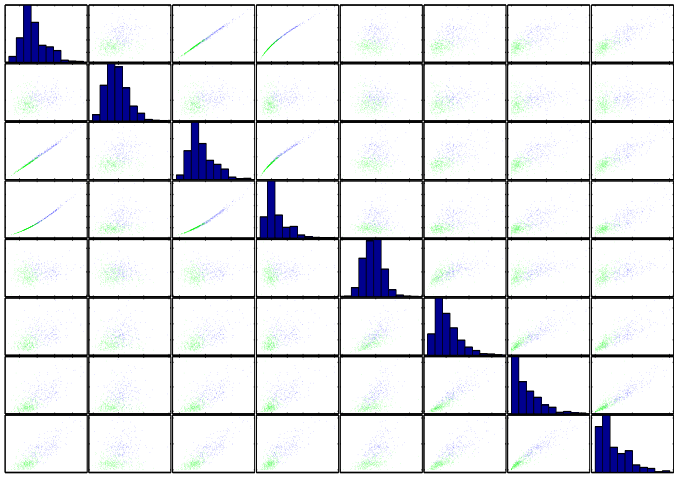


Figure A.11: Breast cancer data - Grouped Scatterplot Matrix for x1 . . . x8

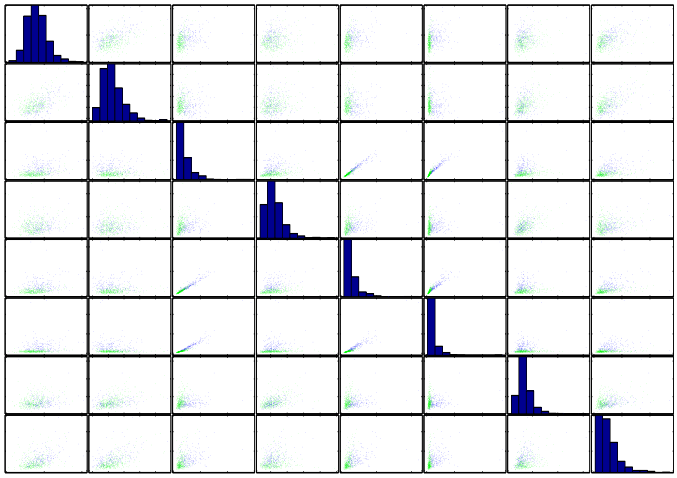


Figure A.12: Breast cancer data - Grouped Scatterplot Matrix for x9 . . . x16

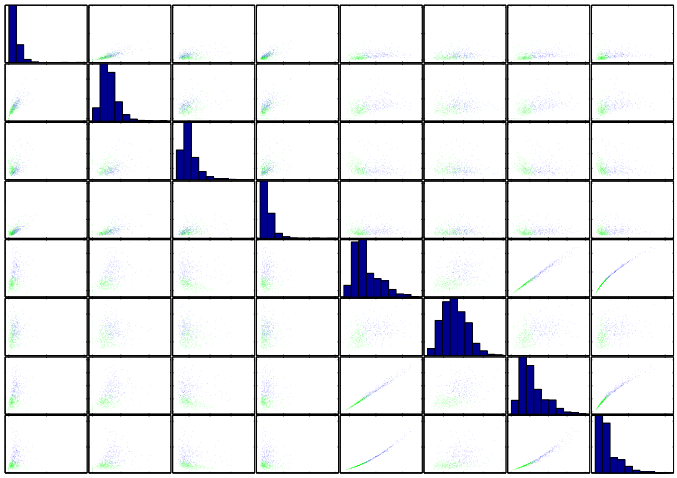


Figure A.13: Breast cancer data - Grouped Scatterplot Matrix for x17 . . . x24

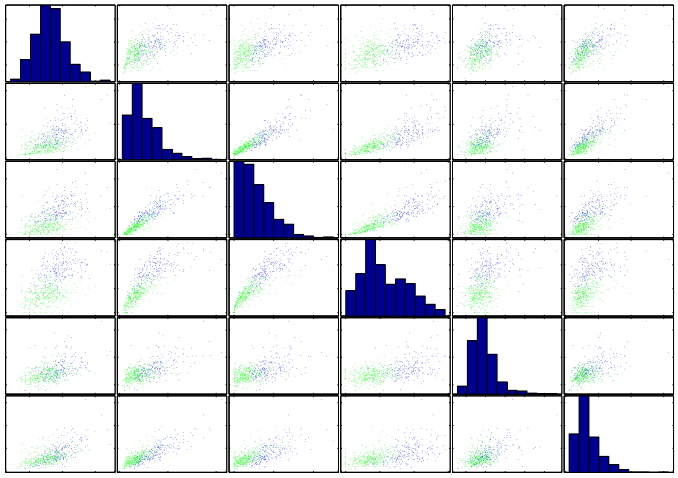


Figure A.14: Breast cancer data - Grouped Scatterplot Matrix for x25 . . . x30

Vita

Esra Turan was born in Turkey in 1982. In 1999 Esra matriculated to Osmangazi University, then transferred to Dokuz Eylul University, Turkey where she graduated as Valedictorian in 2004, having obtained her Bachelors degree in Statistics. While working full time as a Graduate Teaching Assistant at Yasar University in Izmir, Turkey, she completed her Masters degree in Statistics while attending Dokuz Eylul University in 2006. During that year, she achieved another Bachelors degree in Business Administration from Anatolian University, in Eskisehir, Turkey. In 2006, she was accepted to the PhD program in Econometrics at Dokuz Eylul University. After one semester, due to exemplary academic work she was awarded a scholarship from the Ministry of Education to obtain a PhD degree in the USA. After accepting, she joined the department of Statistics, Operations and Management Science at the University of Tennessee, Knoxville in 2007. During her college and graduate school, she has won several scholarships and numerous awards for her many academic achievements. In 2000, she won a scholarship from the Turkish Educational Charitable Foundation. This scholarship was continued during her college and graduate school in Turkey because of her continuing successes. Moreover, she won another scholarship from The Scientific and Technological Research Council of Turkey while attending graduate school in Turkey. Her successes continued in the USA and she was awarded a Graduate Teaching Assistant position at the University of Tennessee, Knoxville. In addition, she has won the Summer Graduate Research Assistantship Award from the University of

Tennessee, Knoxville in 2009. Also during 2009, she joined the competition for Young Researchers Data Mining Prize and her team was awarded the best solution prize.