Alma Mater Studiorum - Università di Bologna

Dottorato di Ricerca in
Scienze Statistiche

Ciclo 36

Settore concorsuale: 13/D1 - STATISTICA
Settore Scientifico Disciplinare: SECS-S/01 - STATISTICA

Modelling and Classification With Quantile-Based Distributions

Presentata da: Edoardo Redivo

Coordinatore Dottorato
Prof.ssa Monica Chiogna

Supervisore
Prof.ssa Cinzia Viroli

Esame finale anno 2023

Abstract

In this work, we explore and demonstrate the potential for modeling and classification using quantile-based distributions, which are random variables defined by their quantile function.

In the first part we formalize a least squares estimation framework for the class of linear quantile functions, leading to unbiased and asymptotically normal estimators. Among the distributions with a linear quantile function, we focus on the flattened generalized logistic distribution (*fgld*), which offers a wide range of distributional shapes. A novel naïve-Bayes classifier is proposed that utilizes the *fgld* estimated via least squares, and through simulations and applications, we demonstrate its competitiveness against state-of-the-art alternatives. The least squares estimator also enables asymptotic hypothesis tests that can serve as a variable selection method in this classification algorithm.

In the second part we consider the Bayesian estimation of quantile-based distributions. Despite being computationally expensive, modern computational tools now allow routine implementation. We introduce a factor model with independent latent variables, which are distributed according to the *fgld*. Similar to the independent factor analysis model, this approach accommodates flexible factor distributions while using fewer parameters. The model is presented within a Bayesian framework, an MCMC algorithm for its estimation is developed, and its effectiveness is illustrated with data coming from the European Social Survey.

The third part focuses on depth functions, which extend the concept of quantiles to multivariate data by imposing a center-outward ordering in the multivariate space. We investigate the recently introduced integrated rank-weighted (IRW) depth function, which is based on the distribution of random spherical projections of the multivariate data. This depth function proves to be computationally efficient and to increase its flexibility we propose different methods to explicitly model the projected univariate distributions. Its usefulness is shown in classification tasks: the maximum depth classifier based on the IRW depth is proven to be asymptotically optimal under certain conditions, and classifiers based on the IRW depth are shown to perform well in simulated and real data experiments.

Contents

Chapter 1

Introduction

Quantile-based distributions provide the quantile of a random variable using an analytical expression, which corresponds to the inverse of the cumulative distribution function. Utilizing quantile functions, as opposed to classical density functions, offers the advantage of constructing flexible distributional families using only a few parameters. Numerous such distributions have been proposed in the statistical literature, often by extending common distributions, such as the g-and-k distribution, which introduces skewness and kurtosis parameters to the normal distribution. However, a significant drawback of quantile-based distributions is the increased computational complexity associated with conventional estimation methods, such as maximum likelihood or posterior inference. Nevertheless, recent advances in Bayesian estimation have made use of increased computing power and improved sampling algorithms, partially addressing this challenge, though it comes with a computational burden for high-dimensional data.

The first contribution of this thesis is the development of a least squares method for estimating univariate quantile-based distributions, specifically those which are linear in their parameters. Our focus centered on the flattened generalized logistic distribution (*fgld*), which can be cast in the class of linear quantile functions, thus offering closed-form estimators. It is characterized by four parameters, making it highly versatile in capturing a wide range of data shapes, including skewed or flattened distributions. The resulting estimators are unbiased and asymptotically normal, enabling the derivation of a testing procedure. Furthermore, based on the theoretical insights regarding the *fgld* distribution,

we proposed a novel naïve-Bayes classifier that utilizes this quantile-based distribution instead of the conventional Gaussian density. In empirical studies, the naïve-Bayes with the *fgld* demonstrated good performance, showcasing its potential as a promising alternative to existing classifiers. Moreover, the application of the proposed testing procedure led to valuable by-products, such as strategies for variable importance and selection, offering practical insights for the data analysis process.

In the second part of this thesis, our focus shifted towards the development of a factor model with independent latent variables distributed according to univariate quantile-based distributions; in particular we again utilized the *fgld*. This model shares similarities with the independent factor analysis, wherein the components are independently distributed as mixtures of Gaussians. However, our proposed model proves to be more suitable for effectively describing data distributions with skewness or flattened central regions. Notably, the model can be easily estimated within a Bayesian framework, assuming weakly informative priors, making it a convenient and powerful approach. Throughout this line of research, we thoroughly investigated identifiability conditions and model selection strategies. The effectiveness of the model is illustrated on real-world data from the European Social Survey.

The extension of the quantile function and quantiles to multivariate scenarios has garnered significant attention, with numerous proposals and theoretical constructs, among which the concept of depth functions stands out. Depth functions impose a center-outward ordering of the multivariate space and various definitions have been studied extensively and compared according to their properties. In the third part of this thesis, we focused on the recently introduced integrated rank-weighted (IRW) depth function, part of the class of the integrated depth functions, which can be seen as the expectation of univariate depths along infinite uniformly distributed random directions. We extend the definition of the depth function, by considering the estimation of univariate cumulative distribution functions, needed for its empirical computation, via parametric or nonparametric models, with the quantile-based *fgld* being a valuable option. This depth definition offers both computational feasibility and flexibility, which in general do not go hand in hand for the most

famous depth functions. We show that its only missing property to qualify as a so-called statistical depth function, affine invariance, is gained when the data is sphered. We also highlight that the Mahalanobis depth can be seen as an integrated depth function when working with sphered data. Furthermore, we prove that the depth completely characterizes the probability distribution of the data and it gives rise to nested, though not necessarily convex, contours. Of particular significance is the application of depth functions to classification tasks. In this thesis, we prove that the maximum-depth classifier based on the IRW depth is asymptotically optimal under certain conditions, and we show its competitiveness against alternative algorithms in both simulated and real data experiments, where the IRW depth also proves useful as the basis for the more flexible DD-classifier.

These findings could hopefully contribute to the understanding and applicability of quantile-based distributions in statistical modeling and classification tasks, offering promising tools and insights that may pave the way for further advancements and future research in this field.

At the time of writing, I have presented the research findings from this research at three scientific meetings. The first contribution has been published as an article in the Statistics and Computing journal (Redivo, E., Viroli, C., & Farcomeni, A., 2023, *Quantile-distribution functions and their use for classification, with application to naïve Bayes classifiers*). The second part of the thesis is currently under submission. Regarding the third contribution, we have developed an R package named `dqclass`, available on GitHub, and a corresponding paper has been prepared and is also currently under submission.

Chapter 2

Quantile-distribution Functions and Their Use for Classification, with Application to Naïve Bayes Classifiers

2.1 Introduction

Quantile functions, defined as the generalised inverse of cumulative distribution functions, have nice properties that make them a valuable inferential tool. For instance, sums and convex linear combinations of quantile functions are still quantile functions. As a consequence, it is possible to construct arbitrary new quantile functions that have great flexibility and a small number of parameters (see, for instance, Karvanen (2006)). Thus, we can obtain distributions with a wide range of different shapes and also the exact or approximate form of many common distributions, including the normal, Students T and logistic distributions. See Gilchrist (2000) for a clear introduction to the use of quantile functions, their properties, and the main estimation methods.

Various flexible quantile functions have been proposed in the literature. The so-called *g-and-k* distribution (Haynes et al., 1997; Rayner and MacGillivray, 2002) is defined as a generalization of the Gaussian distribution with additional skewness and kurtosis parameters. Freimer et al. (1988) introduced the quantile-based representation of the generalized Lambda distribution. Sankaran

et al. (2016) proposed a new quantile function based on the sum of generalized Pareto and Weibull quantile functions.

Quantile functions that are linear in their parameters have desirable inferential properties, as will be shown in the following. Well-known examples are the flattened logistic distribution (Sharma and Chakrabarty, 2019) and the generalized flattened logistic distribution (Chakrabarty and Sharma, 2021).

Quantile functions can be estimated according to different strategies. Distributions that have analytical L-moments can be estimated by matching sample L-moments with their theoretical counterparts, in the same spirit as the method of moments (see, for instance, Chakrabarty and Sharma (2021)). Maximum likelihood estimation is possible as well; however, if the quantile function is not invertible - as is usually the case - then, for each observation of the data sample, say x, a numerical inversion needs to be carried out to find the correspondent percentile u, thus making the parameter estimation process numerically unstable and computationally expensive (Rayner and MacGillivray, 2002). An alternative illustrated in Gilchrist (2000) is based on the minimization of the $L1$ norm between the ordered statistics and their theoretical median, leading to a least absolute deviation method. Without explicit density functions Bayesian estimation cannot be applied; however Allingham et al. (2009); Drovandi and Pettitt (2011) developed an Approximate Bayesian Computation (ABC) strategy for the estimation of some classes of quantile functions.

In this work we show that the family of linear quantile functions can be efficiently estimated using least squares by exploiting the properties of the order statistics. We also develop the asymptotic distribution of a statistical test to check whether two estimated quantile functions have the same parameters. We also show how the procedure can be used for classification, by constructing a simple Naïve Bayes classifier based on quantile distributions, where the proposed testing procedure is used for variable selection and variable importance in a two-class problem. Empirical studies indicate that the proposed variable screening can help the classification task, and, in this perspective, it is alternative to variable weighting (see, for instance, Jiang et al. (2018) and Jiang et al. (2019)) or structure extensions by hidden variables Jiang et al. (2008). A completely different approach where quantile functions are used for classification is reported in Farcomeni et al. (2022a).

The rest of the paper is organised as following. In the next section we out-

line linear quantile functions and define our least squares estimator. Asymptotic results are given in Section 2.2.3, where we also derive the null distribution of relevant test statistics. In Section 2.3 we discuss how to use linear quantile functions for supervised classification and variable selection. Simulation studies are reported in Section 2.4 and the proposed strategy is illustrated on real data in Section 2.5.

2.2 Quantile-based distributions

Denote with $F(x; \boldsymbol{\theta})$ a distribution function that is right-continuous, depending on a vector of parameters $\boldsymbol{\theta}$ of length p. The quantile distribution function can be defined as in Parzen (1979):

$$F^{-1}(u; \boldsymbol{\theta}) = Q(u; \boldsymbol{\theta}) = \inf\{x : F(x; \boldsymbol{\theta}) \geq u\},$$

for $0 < u < 1$. As in Tukey (1965), we call

$$q(u; \boldsymbol{\theta}) = Q'(u; \boldsymbol{\theta}),$$

the quantile density function, which is related to the density function as:

$$f(x; \boldsymbol{\theta}) = \frac{1}{q(F(x; \boldsymbol{\theta}))}. \tag{2.1}$$

For certain probability distributions the quantile function can be derived in analytical form through the inversion of the cumulative distribution function. Some examples are reported in Table 2.1. Most probabilistic densities do not admit closed-form quantile functions though. One notable example is the Gaussian distribution. The contrary is also true: a quantile function can be defined without making reference to an explicit probability distribution function.

An interesting family of quantile functions is given by the ones that are linear in their parameters. Starting from the symmetric quantile function of the logistic distribution:

$$Q(u; \boldsymbol{\theta}) = \alpha + \beta[\log u - \log(1 - u)] \tag{2.2}$$

Sharma and Chakrabarty (2019) proposed the flattened version

$$Q(u;\boldsymbol{\theta}) = \alpha + \beta\left[\log\frac{u}{1-u} + \kappa u\right],$$

where the additional component indexed by the shape parameter κ regulates the flatness of the peak of the distribution. They derived classical and quantile-based properties of the distribution and compared its flexibility with respect to the logistic distribution in terms of fitting in empirical contexts.

More recently, Chakrabarty and Sharma (2021) proposed a generalization of the flattened logistic distribution (*fgld*):

$$Q(u;\boldsymbol{\theta}) = \alpha + \beta\left[(1-\delta)\log u - \delta\log(1-u) + \kappa u\right] \tag{2.3}$$

that proved to be very flexible and outperformed the existing strategies in terms of model fitting. Figure 2.1 and Figure 2.2 show the range of shapes this distribution can take.

2.2.1 Least squares estimation

In order to estimate the quantile function $Q(u,\boldsymbol{\theta})$, different strategies can be applied. L-moments matching (Chakrabarty and Sharma, 2021) requires the analytical form of L-moments for the quantile function, along the same lines

Probability distribution	Density function	Quantile function
Exponential	$\theta e^{-\theta x}$	$-\frac{\log(1-u)}{\theta}$
Extreme Value	$\frac{1}{\beta}e^{\frac{x}{\beta}}\exp\left[-e^{\frac{x}{\beta}}\right]$	$\beta\log\log(1-u)^{-1}$
Weibull	$\frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1}e^{-(x/\lambda)^k}$	$\lambda\{\log(1-u)^{-1}\}^{1/k}$
Logistic	$\frac{e^{-(x-\alpha)/\beta}}{\beta\left(1+e^{-(x-\alpha)/\beta}\right)^2}$	$\alpha+\beta\log\frac{u}{(1-u)}$
Double-Exponential	$\frac{e^{-\lvert x\rvert}}{2}$	$\log 2u, \quad u<0.5$ $-\log 2(1-u), \quad u>0.5$
Cauchy	$\frac{1}{\pi(1+x^2)}$	$\tan\pi(u-0.5)$
Pareto	$\frac{\alpha\mu^\alpha}{x^{\alpha+1}}$	$\mu\log(1-u)^{-\frac{1}{\alpha}}$

Table 2.1: Quantile functions of some probability distributions.

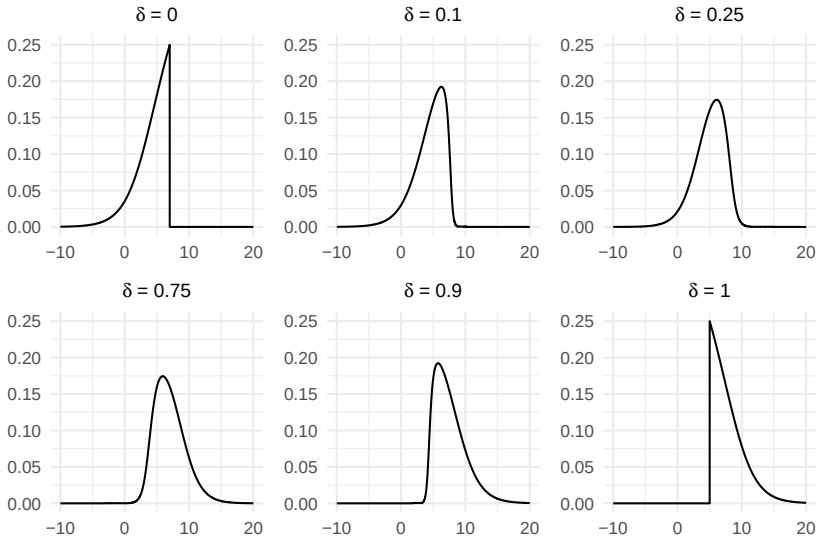

Figure 2.1: *fgld* with $\alpha = 5$, $\beta = 2$, $\kappa = 1$ and varying δ.

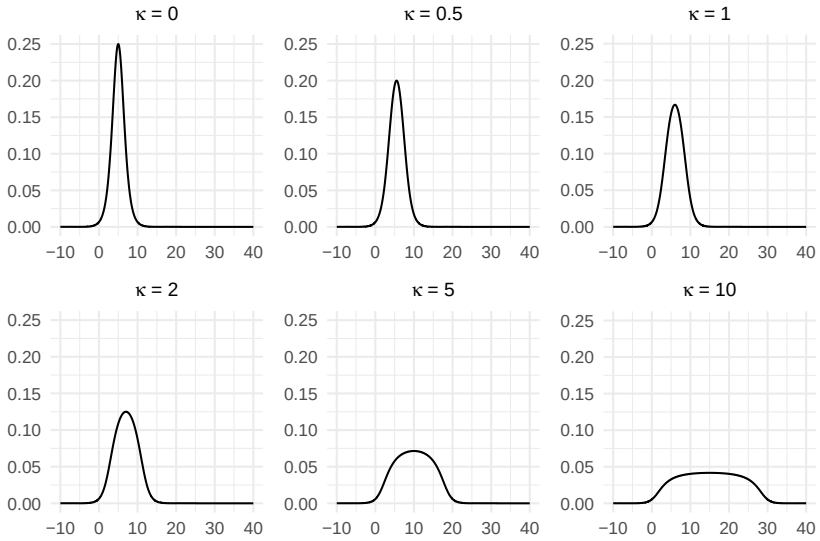

Figure 2.2: *fgld* with $\alpha = 5$, $\beta = 2$, $\delta = 0.5$ and varying κ.

of method of moments. Maximum likelihood is a possible alternative strategy but it requires the approximation of the percentiles for each observation and the inversion of the derivative of the quantile function, thus resulting in an computationally expensive method (Rayner and MacGillivray, 2002). In a Bayesian perspective, an Approximate Bayesian Computation (ABC) method has been developed (Allingham et al., 2009; Drovandi and Pettitt, 2011) for specific classes of quantile functions, but again at the price of computational burden.

In Gilchrist (2000) two estimation methods based on 'lack of fit criteria' are introduced, which are denoted as distributional least absolutes and distributional least squares. The first is based on the minimization of the $L1$ norm between the ordered statistics and their theoretical median. The second approach consists in minimizing the $L2$ norm between the expected and the observed ordered statistics. Gilchrist highlights that, if no analytical form for the expected order statistics is available, they need to be approximated by a Taylor series expansion. For this reason the author champions the approach of the L1 norm, which does not require such derivation. Here instead, we develop a framework under which the least squares approach can be effectively and efficiently used with a closed form solution, and we also derive some theoretical results.

In fact, there is a specific link between theoretical order statistics and quantile-based distributions (David and Nagaraja, 2004). More specifically, the expected value of an order statistic can expressed in terms of the quantile distribution as follows:

$$E[X_{(i)}] = \frac{1}{B(i, n-i+1)} \int_0^1 Q(u; \boldsymbol{\theta}) u^{i-1} (1-u)^{n-i} du. \qquad (2.4)$$

As stated in the following Lemma, if the quantile function is linear in its parameters, the expected value of the theoretical order statistics takes a similar linear form that simplifies the estimation method.

Lemma 1 *If a quantile distribution function is linear with respect to its parameters, then the expected order statistics of that distribution will also be linear with respect to those same parameters.*

The proof is shown in Appendix A. Take for instance the simple quantile function $Q(u; \boldsymbol{\theta}) = \theta_0 + \theta_1 u$, with $\theta_1 > 0$ and $\boldsymbol{\theta} = (\theta_0, \theta_1)$. Then by solving

the integral in (2.4) we easily get

$$E[X_{(i)}] = \theta_0 + \theta_1 \frac{i}{n+1} = \begin{bmatrix} 1 & \frac{i}{n+1} \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix} = \mathbf{b}_i^\top \boldsymbol{\theta}.$$

For a quantile function with a quadratic term in u, $Q(u; \boldsymbol{\theta}) = \theta_0 + \theta_1 u + \theta_2 u^2$, similarly we get

$$E[X_{(i)}] = \theta_0 + \frac{i}{n+1}\,\theta_1 + \frac{i\,(i+1)}{(n+2)(n+1)}\,\theta_2.$$

Thus, for any linear quantile function, the expected values of the order statistics can written as

$$E[X_{(i)}] = \mathbf{b}_i^\top \boldsymbol{\theta},$$

where $\mathbf{b}_i$ are p-dimensional vectors of known coefficients.

Now, given a sample of IID observations $(x_1, \ldots, x_n)$ from $X \sim F(\boldsymbol{\theta})$ denote with $x_{(i)}$ the observed i-th order statistics. We can minimize:

$$\phi(\boldsymbol{\theta}) = \sum_{i=1}^{n} \left(x_{(i)} - E[X_{(i)}]\right)^2 = \sum_{i=1}^{n} \left(x_{(i)} - \mathbf{b}_i^\top \boldsymbol{\theta}\right)^2 \tag{2.5}$$

with respect to $\boldsymbol{\theta}$.

The resulting least squares estimation method is very efficient, since it provides a closed-form solution for the parameters.

By defining $\mathbf{B}$ as the matrix of dimension $n \times p$ having as rows $\mathbf{b}_i$ and by $\mathbf{X}_{(\cdot)}$ the ordered random sample, the estimate of $\boldsymbol{\theta}$ is

$$\hat{\boldsymbol{\theta}} = (\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{B}^\top \mathbf{X}_{(\cdot)}. \tag{2.6}$$

Furthermore we have:

$$E[\hat{\boldsymbol{\theta}}] = (\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{B}^\top E[\mathbf{X}_{(\cdot)}] = (\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{B}^\top \mathbf{B}\,\boldsymbol{\theta} = \boldsymbol{\theta} \tag{2.7}$$

and

$$V[\hat{\boldsymbol{\theta}}] = (\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{B}^\top\, \boldsymbol{\Sigma}\, \mathbf{B} (\mathbf{B}^\top \mathbf{B})^{-1}$$

where $V[\mathbf{X}_{(\cdot)}] = \boldsymbol{\Sigma}$ is the covariance matrix of the order statistics. So the estimator $\hat{\boldsymbol{\theta}}$ is unbiased, but, given the correlation among order statistics, we can not invoke the BLUE property of the Gauss-Markov theorem.

2.2.2 An example: the flattened generalised logistic distribution

In this section, we derive the results needed for least squares parameter estimation of the flattened generalized logistic (*fgld*) quantile function defined in equation (2.3). To this aim it is convenient to re-parameterise the quantile function as follows:

$$\begin{cases} \alpha = \theta_0 \\ \beta\kappa = \theta_1 \\ \beta(1-\delta) = \theta_2 \\ \beta\delta = \theta_3 \end{cases} \qquad \begin{cases} \alpha = \theta_0 \\ \beta = \theta_2 + \theta_3 \\ \delta = \frac{\theta_3}{\theta_2+\theta_3} \\ \kappa = \frac{\theta_1}{\theta_2+\theta_3} \end{cases}$$

The quantile distribution function of the *fgld* becomes:

$$Q(u) = \theta_0 + \theta_1\, u + \theta_2\, \log u - \theta_3\, \log(1-u) \tag{2.8}$$

To estimate the parameters via least squares we need to derive the expected value of the order statistics.

Lemma 2 *The expected order statistic of the flattened generalised logistic distribution is equal to:*

$$E[X_{(i)}] = \theta_0 + \theta_1 \frac{i}{n+1} + \theta_2 \left(\psi(i) - \psi(n+1)\right) + \theta_3 \left(\psi(n+1) - \psi(n-i+1)\right) \tag{2.9}$$

where $\psi(\cdot)$ indicates the digamma function, which is defined as the derivative of the logarithm of the gamma function.

Therefore, in this case we get

$$\mathbf{b}_i = \left(1, \frac{i}{n+1}, \psi(i) - \psi(n+1), \psi(n+1) - \psi(n-i+1)\right).$$

For a proof see the Appendix A.

In order to compute the variance of the estimator we also need to derive the covariance matrix for the order statistics of the *fgld*.

Lemma 3 *The n-dimensional covariance matrix of the order statistics, $\mathbf{\Sigma}$, of the flattened generalised logistic distribution has diagonal variances given by*

$$\begin{aligned} V[X_{(r)}] = \; & \theta_1^2 \frac{r(n-r+1)}{(n+1)^2(n+2)} + \theta_1\theta_2 \frac{2(n-r+1)}{(n+1)^2} + \\ & + \theta_1\theta_3 \frac{2r}{(n+1)^2} + \theta_2^2 \left(\psi_1(r) - \psi_1(n+1)\right) + \\ & + \theta_2\theta_3 \, 2\psi_1(n+1) + \theta_3^2 \left(\psi_1(n-r+1) - \psi_1(n+1)\right) \end{aligned}$$

with $r = 1, \ldots, n$ and where $\psi_1(\cdot)$ indicates the trigamma function, which is the derivative of digamma function $\psi(\cdot)$.

The covariance between any two order statistics of the flattened generalised logistic distribution is equal to:

$$\begin{aligned} Cov[X_{(r)}, X_{(s)}] \; = \; & \theta_1^2 \left[\frac{r(n-s+1)}{(n+1)^2(n+2)}\right] + \theta_1\theta_2 \left[\frac{(n-s+1)(r+s)}{(n+1)^2 s}\right] + \\ & \theta_1\theta_3 \left[\frac{r(2n-r-s+2)}{(n+1)^2(n-r+1)}\right] + \theta_2^2 \left[\psi_1(s) - \psi_1(n+1)\right] + \\ & \theta_2\theta_3 \left[\left(\psi(n+1) - \psi(n-r+1)\right)\left(\psi(n+1) - \psi(s)\right) + \psi_1(n+1)\right] + \\ & \theta_3^2 \left[\psi_1(n-r+1) - \psi_1(n+1)\right] - \theta_2\theta_3\xi(n,r,s) \end{aligned}$$

where

$$\begin{aligned} \xi(n,r,s) = & \Gamma(s-r)\,\Gamma(n-s+1) \\ & \sum_{h=1}^{\infty} \frac{1}{h} \frac{\Gamma(h+r)}{\Gamma(n+h+1)} \left(\psi(n+h+1) - \psi(h+s)\right) \end{aligned}$$

for $r, s = 1, \ldots, n$.

A sketch of the proof in given in the Appendix A.

2.2.3 Asymptotic results

In this section we derive the asymptotic distribution of the estimator of the *fgld* defined in Equation 2.6. First notice that this estimator can be expressed as a linear combination of the order statistics:

$$\hat{\boldsymbol{\theta}} = \sum_{i=1}^{n} \boldsymbol{c}_{in} X_{(i)},$$

where the coefficients $\boldsymbol{c}_{in}$ are vectors of the same length p as $\hat{\boldsymbol{\theta}}$.

Lemma 4 *The coefficients $\boldsymbol{c}_{in}$ for the least squares estimator of the* fgld *are continuous and bounded.*

The proof is given in the Appendix A. Given this lemma we can derive the following theorem.

Theorem 1 *The least squares estimator for the parameters of the* fgld *linear quantile function has an asymptotically normal distribution:*

$$\hat{\boldsymbol{\theta}} \xrightarrow{d} N_p(\boldsymbol{\theta}, \boldsymbol{\Gamma}) \tag{2.10}$$

with $\boldsymbol{\Gamma} = (\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{B}^\top\boldsymbol{\Sigma}\mathbf{B}(\mathbf{B}^\top\mathbf{B})^{-1}$.

The proof of Theorem 1 is shown in the Appendix A.

Given the previous result, the null hypothesis that the sample comes from a quantile function with parameters $\boldsymbol{\theta}_0$ can be tested as stated in the following theorem.

Theorem 2 *The null hypothesis $H_0 : \boldsymbol{\theta} = \boldsymbol{\theta}_0$ can be checked through the test statistic*

$$(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0) \xrightarrow{d} N_p(\mathbf{0},\, \boldsymbol{\Gamma}),$$

where for fgld *quantile function the matrices* $\mathbf{B}$ *and* $\boldsymbol{\Sigma}$ *are known quantities derived in Lemma 1 and 3.*

As a simple consequence we can also test the hypothesis that two observed samples come from the same population $H_0 : \mathbf{B}\boldsymbol{\theta}_1 = \mathbf{B}\boldsymbol{\theta}_2$ which is equivalent to $H_0 : \boldsymbol{\theta}_1 = \boldsymbol{\theta}_2$.

Under the previous assumptions we get

$$(\hat{\boldsymbol{\theta}}_1 - \hat{\boldsymbol{\theta}}_2) \xrightarrow{d} N_p(\mathbf{0},\, 2\,\boldsymbol{\Gamma})$$

or alternatively

$$\frac{1}{2}(\hat{\boldsymbol{\theta}}_0 - \hat{\boldsymbol{\theta}}_1)^\top\boldsymbol{\Gamma}^{-1}(\hat{\boldsymbol{\theta}}_0 - \hat{\boldsymbol{\theta}}_1) \xrightarrow{d} \chi^2_p. \tag{2.11}$$

2.3 Application to supervised classification

Let Y be a categorical random variables taking values $y = \{1, \ldots, K\}$, where K denotes the total number of classes and let $\mathbf{X} = (X_1, \ldots, X_p)$ be a set of observed variables. One of the most used classification methods in the supervised setting is the so-called naïve Bayes classifier (John and Langley, 1995; Hand and Yu, 2001). Suppose you have a training data set in which both Y and $\mathbf{X}$ are known. According to the Bayesian rule, the posterior probability of belonging to a generic class k ($k = 1, \ldots, K$) is

$$Pr(Y = k \mid \mathbf{X} = \mathbf{x}) = \frac{\pi_k f(\mathbf{x} \mid Y = k)}{f(\mathbf{x})} = \frac{\pi_k f(\mathbf{x} \mid Y = k)}{\sum_{k'=1}^{K} \pi'_k f(\mathbf{x} \mid Y = k')}, \quad (2.12)$$

where π_k denotes the proportion of units that belong to class k in the training set.

The naïve Bayes classifier assumes conditional independence of the variables given the categorical response

$$f(\mathbf{x} \mid Y = k) = \prod_{j=1}^{p} f_j(x_j \mid Y = k),$$

thus each variable is treated separately.

The class conditional distributions $f(x_j \mid Y = k)$ are usually assumed to be Gaussian. An alternative has been proposed by John and Langley (2013), who suggested the use of kernel density estimation as a tool to allow for more flexible distributional shapes. A further common method is the discretization of all continuous variables, that is estimating the density function via a step function. For this method the main issue is to choose the breaks that define the categories; a recent heuristic proposal is that of Yang and Webb (2009), the so-called proportional discretization. This method achieves (approximately) a discretization with bins having both equal width and equal frequency, with the added advantage that the tuning parameter is derived automatically and based on the sample size (n): $\texttt{width} = \texttt{frequency} \approx \sqrt{n}$.

Quantile-based distributions can be applied in this setting with the goal of taking advantage of their flexible and parsimonious specifications and the fast and reliable estimation given by the least squares method.

The application of quantile-based distributions in the naïve Bayes algorithm involves the estimation of $K \times p$ univariate distributions, similarly to the

other methods. Each of the univariate samples is identified by a variable and a category of the response, and their quantile function can be estimated via least squares, provided we choose a linear quantile function. The output of the estimation phase is just a set of parameters: $\boldsymbol{\theta}_{jk}$, with $j = 1, \ldots, p$ and $k = 1, \ldots, K$. Given a single sample identified by a set of variables $\mathbf{x} = (x_1, \ldots, x_p)$, the class conditional distribution is evaluated as follows, for each variable j and categorical response k:

$$P(X_j = x_j \mid Y = k) = f_j(x_j; \boldsymbol{\theta}_{jk}) = \frac{1}{q_j(u_j; \boldsymbol{\theta}_{jk})},$$

where the density is evaluated based on the relationship shown in equation (2.1) and u_j is the inverse of $x_j = Q(u_j; \boldsymbol{\theta}_{jk})$ and needs to be computed numerically in the case of non-invertible quantile functions, such is the case of the *fgld*.

As a by-product of the least square fit, a simple distance measure between two quantile distributions can be derived. Imagine that $\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$ are the estimates of the parameters of two quantile functions. For instance, the quantile function of the classes 1 and 2 of the training sample. Then for each variable we can measure:

$$\|\mathbf{B}\hat{\boldsymbol{\theta}}_1 - \mathbf{B}\hat{\boldsymbol{\theta}}_2\|_2$$

where $\|\ldots\|_2$ denotes the Euclidean distance. The formula can also be interpreted as the Euclidean distance between two vectors containing the expected order statistics for the two distributions.

The formula can be applied seamlessly in the case of two response classes with equal number of observations. When the latter differs between the classes, n can be chosen for instance as the minimum class frequency; when the classes are more than two, the distance can be computed for each pair and the maximum pairwise distance can be retained, meaning that the variable can at least discriminate between those two classes.

This measure can serve to rank variables in terms of their importance, of course limited to their application in the naïve Bayes algorithm. This can be useful in interpreting and explaining the model, in a similar way to the use of variable importance measures derived from algorithms such as random forests.

Moreover, it can serve as the basis of a variable selection procedure as explained in Section 2.2.3 (Theorem 2). Imagine we have $K = 2$ classes, then a

variable is relevant for classification if the null $H_0 : \hat{\boldsymbol{\theta}}_1 = \hat{\boldsymbol{\theta}}_2$ is rejected, where $\hat{\boldsymbol{\theta}}_1$ and $\hat{\boldsymbol{\theta}}_2$ denote the parameters in the two class-populations.

2.4 Simulation study

In this section we present some empirical studies to evaluate the goodness-of-fit of the illustrated quantile functions in different scenarios, their classification performance in the naïve Bayes algorithm and the behaviour of the asymptotic test.

2.4.1 Empirical bias

In this first simulation we investigate the goodness-of-fit of three different quantile-based distributions: the simple quantile function with a linear term in u (*linear*), the quantile function with a quadratic term in u (*quad*) and the *fgld*. In order to measure the empirical bias and the variability of the estimators of $\boldsymbol{\theta}$ we compare the observed order statistics with their expectation according to the three models, by computing this empirical bias measure:

$$\sqrt{\frac{\sum_{i=1}^{n}(x_{(i)} - \hat{E}[X_{(i)}])^2}{n}} = \sqrt{\frac{\sum_{i=1}^{n}(x_{(i)} - \mathbf{B}\,\hat{\boldsymbol{\theta}})^2}{n}}.$$

We simulated $n = 100$ observations from four different distributions: a standard normal, a t distribution with 3 degrees of freedom, an exponential distribution with rate parameter equal to 0.5, and a $\log(|t_{\nu=3}|)$, that is the logarithm of the absolute value of a t distribution (again with 3 degrees of freedom). For each scenario we generated 100 replicates. Table 2.2 shows the mean of the empirical bias across the replicates for each scenario and model. In brackets the standard deviations offer an indication of the variability of the estimates.

Results show that the *fgld* is by far the most flexible model, it being able to fit well in all scenarios.

	linear	*quad*	*fgld*
norm	0.22 (0.05)	0.21 (0.05)	0.08 (0.02)
t	0.86 (0.54)	0.84 (0.53)	0.35 (0.38)
logabst	0.43 (0.13)	0.36 (0.12)	0.13 (0.06)
exp	1.01 (0.27)	0.72 (0.26)	0.23 (0.1)

Table 2.2: Average empirical bias over 100 replicates for 4 distributional scenarios (rows) and for 3 quantile-based distributions (columns). Standard deviations are reported in brackets.

2.4.2 Classification

We evaluated the performance of the quantile-based distributions in the naïve Bayes algorithm via a simulation study. We considered the *fgld* and the quantile function with a quadratic term in u (*quad*) described in Section 2.2.1. We generated p variables X_j ($j = 1, \ldots, p$) of sample size n, according to the four different distributions described in the previous subsection.
We fixed $K = 2$ classes, of equal size $n/2$. Denote X_{j0} the variable X_j when $Y = 0$ and X_{j1} when $Y = 1$. In order to separate the classes we shifted each variable according to the rule

$$X_{j1} = X_{j0} + 0.3\,(-1)^j \quad j = 1, \ldots, p$$

Alternatively, we have applied a scaling as

$$X_{j1} = 0.8\,X_{j0} \quad j = 1, \ldots, p$$

Shifting has been applied to all distributional settings, while scaling has been applied only to the $\log(|t_{\nu=3}|)$ distribution; thus creating five different scenarios: (i) shifted $N(0, 1)$, (ii) shifted $t_{\nu=3}$, (iii) shifted Exp($\lambda = 0.5$), (iv) shifted $\log(|t_{\nu=3}|)$ and (v) scaled $\log(|t_{\nu=3}|)$. For each scenario we let $p = \{10, 50, 100\}$, $n = \{100, 500, 1000\}$, and correlated or independent variables.

The five distributional scenarios, three variable set sizes, three sample sizes and two correlation structures lead to ninety settings. For each setting we repeated data generation and estimation 100 times. Misclassification rates were evaluated on test sets generated in same way as the training samples, and we report the average over the replicates.

We compared with other choices for the class-conditional distributions; namely the normal, the kernel (kde), with default Silverman's rule for the bandwidth, the discrete method (with proportional discretization (Yang and Webb, 2009)), the generalized extreme value distribution (*gev*) estimated via maximum likelihood by the R package `evd`.

Table 2.3 contains a summary of the computational times for this simulation. We can note that the time needed for the methods based on the least squares estimation of quantile functions is longer than for simpler methods such as the normal and the discrete, but it is manageable even for the larger data sets. Times are particularly affected by the increase in the number of independent variables (p).

Results for the classification are presented graphically in Figure 2.3 for each data generating distribution, where we collapse over the 18 settings evaluated for each case. We show scaled differences with respect to a reference method for each setting; we choose *fgld* as the reference. The scaled differences are computed as follows:

$$d_{jk} = \frac{e_{jk} - e_{j1}}{\bar{e}_j}$$

where $j = 1, \ldots, 18$ indicates the setting for fixed data generating distribution, $k = 1, \ldots, 5$ represents the method (with 1 being the reference method), and $\bar{e}_j$ being the average test error for that setting. From Figure 2.3 we can see that *fgld* is very competitive: as expected it performs worse than the normal when the data are indeed normal, but the discrepancy is minimal; it is the best method otherwise with the exception of the exponential data when only *gev* performs better.

2.4.3 Testing procedure

In order to evaluate the performance of the test we assess the distribution of the test statistic for the *fgld* and for the *quad* quantile functions under the null hypothesis $H_0 : \theta_1 = \theta_2$, and the power of the test when the null hypothesis is not true. The variance of the order statistics of the *quad* quantile function, needed for the variance of the least squares estimator, is reported in the Appendix A.

	method	p = 10	p = 50	p = 100
n = 100	discrete	0.06 (0.01)	0.33 (2.54)	1.42 (29.54)
	fgld	0.15 (0.02)	4.62 (61.03)	5.42 (61.08)
	gev	0.04 (0.00)	1.16 (31.51)	1.32 (31.51)
	kde	1.00 (29.54)	0.31 (0.02)	0.62 (0.04)
	normal	0.02 (0.00)	0.04 (0.01)	0.09 (0.01)
	quad	0.11 (0.01)	2.46 (43.17)	1.08 (0.06)
n = 500	discrete	0.06 (0.01)	0.26 (0.04)	0.51 (0.03)
	fgld	0.64 (0.03)	3.25 (0.14)	6.54 (0.16)
	gev	0.08 (0.01)	0.41 (0.08)	0.80 (0.12)
	kde	0.34 (0.02)	1.59 (0.10)	3.13 (0.12)
	normal	0.08 (0.01)	0.23 (0.03)	0.41 (0.03)
	quad	0.54 (0.04)	2.74 (0.18)	5.48 (0.29)
n = 1000	discrete	0.06 (0.01)	0.27 (0.02)	0.48 (0.07)
	fgld	1.31 (0.04)	6.46 (0.19)	11.94 (1.31)
	gev	0.15 (0.03)	0.72 (0.16)	1.33 (0.31)
	kde	0.70 (0.03)	3.19 (0.10)	5.85 (0.60)
	normal	0.16 (0.02)	0.46 (0.03)	0.76 (0.09)
	quad	1.11 (0.07)	5.48 (0.27)	10.12 (1.21)

Table 2.3: Computational average times in seconds for training the naïve Bayes classifier and applying its prediction on a test set over the 100 replications for the 5 distributional scenarios. In brackets standard deviations are reported.

Type I error

Under the null hypothesis the two samples come from the same distribution. In order to evaluate the convergence of the test statistic to its null distribution we compare empirical type I errors with the nominal significance level that has been chosen in advance.

A total of 200 sets of parameters have been randomly generated, and for each of them 1,000 two-group samples have been simulated. From each of these 1,000 data sets the test statistic can be computed and the empirical type I error corresponds to the proportion of test statistics above the critical value (the 95$^{\text{th}}$ quantile of the $\chi^2_{df=4}$ distribution for the *fgld* and the $\chi^2_{df=3}$ for the *quad*). This procedure has been repeated for different group sample sizes, with the same parameter sets, and the results are shown in Figure 2.4. As could be expected, the empirical type I error converges to the nominal one as the sample size increases in both cases.

ROC curves

To evaluate the power of the test we have simulated data sets of 1,000 variables, with half of those variables having a different distribution between the two balanced groups, and half having the same distribution. For each variable the p-value associated with the test statistic is computed.

This problem can be re-framed as a classification problem in which the response is whether or not the variable is useful (having a different or equal distribution across the two groups).

In the simulation we know whether the variable is useful or not, so we can evaluate it with the metrics of a classification model, such as a ROC curve. This is particularly suited to the test because the different thresholds (and subsequent classifications) can be interpreted as significance levels.

In Figure 2.5 we report the ROC curves for the *fgld* and *quad* that evaluate whether test statistics are able to identify correctly useful and not useful variables. In both cases we can see that as n increases the curves move more and more towards the top left corner. Even with low sample sizes there are cutoff points for which the test performs extremely well both in terms of sensitivity and of specificity.

	sample size	numerical variables	categorical variables	response classes
cleveland	297	6	7	2
credit	653	6	9	2
diabetes	768	8	0	2
glass	214	9	0	6
heart	270	6	7	2
ionosphere	351	32	2	2
letter	20000	16	0	26
sonar	208	60	0	2
thyroid	2751	6	21	2
vehicle	752	18	0	4
waveform	5000	40	0	3
wbcd	569	30	0	2

Table 2.4: Datasets from the UCI Machine Learning Repository used for comparing naïve Bayes methods, with some information regarding data size and type.

2.5 Real data examples

2.5.1 Benchmark datasets

We have compared the different methods for the naïve Bayes classifier used in Section 2.4.2 on some real datasets commonly used for benchmarking. The chosen datasets are all publicly available from the UCI Machine Learning Repository (Dua and Graff, 2019a). When available we used the preprocessed version from the R package `mlbench` (Leisch and Dimitriadou, 2021). In Table 2.4 some basic information of the datasets used is provided: we can note the general adaptability of the naïve Bayes classifier, being able to deal with both numerical and categorical variables at the same time and with multi-class response variables.

On these data we fitted the models that performed the best in the simulation study (Section 2.4.2), namely the *fgld*, the normal, the kde and the discrete. Results in terms of accuracy from 10-fold cross-validation are presented in Ta-

	fgld	normal	kde	discrete
cleveland	80.79	80.13	80.46	82.15
credit	80.36	73.94	76.28	84.36
diabetes	76.05	75.39	75.01	65.24
glass	57.58	45.84	54.55	53.23
heart	82.96	81.48	81.11	82.22
ionosphere	73.23	82.35	91.75	88.07
letter	65.39	64.28	70.48	51.57
sonar	70.18	67.63	75.49	74.90
thyroid	93.20	93.42	95.02	92.08
vehicle	59.32	44.92	57.15	62.50
waveform	80.28	80.00	79.86	75.24
wbcd	94.73	92.95	93.67	88.90

Table 2.5: Accuracy from different naïve Bayes methods (columns) applied on 12 benchmark datasets (rows). The results are obtained from 10-fold cross-validation.

ble 2.5. We can note that no method is uniformly superior to the others. In general, the additional flexibility given by the *fgld*, the kde and the discrete, with respect to the normal, proves advantageous. We can note the *fgld* performs comparatively well and there are multiple datasets where it achieves the maximum accuracy.

2.5.2 Variable selection

In this section we illustrate the proposed strategy for variable selection on a real dataset. We revisit data from Altman (1968), available in the R package `MixGHD` (Tortora et al., 2021), by adding noise variables. The original dataset contains information about $n = 66$ companies that have filed for bankruptcy. Our task is to predict the status of the firms (0 for 'bankruptcy' or 1 for 'financially sound'). The original predictors are two measurements related to the earnings of the firm. On top these two relevant variables we added 198 irrelevant variables sampled from a standard normal distribution, for a total $p = 200$. The goal is to check whether the variable selection procedure developed in Section 2.3 is able to identify the two real variables, and then to compare the accuracy of

various naïve Bayes classification algorithms in the complete dataset and with some other values of p.
To this aim we considered the naïve Bayes classifiers, with the previously used methods for estimating the distribution (normal, kde, discretization and *fgld*). We also compare these classifiers to other commonly used ones: k-nearest neighbors with k = 3, logistic regression and linear discriminant analysis.
First we computed the p-values associated with the test for each variable, and by using a procedure for controlling the false discovery rate (the Benjamini-Hochberg procedure), we correctly reject the null hypothesis only for the two original variables. Next, we re-ordered variables in ascending order by the obtained p-values and we compare the classifiers in datasets with an increasing number of variables, where variables with progressively higher p-values are included. Results are shown in Table 2.6 for values of $p = 2, 50, 100, 150, 200$. A visual representation of the naïve Bayes with the *fgld* is shown in Figure 2.6, where the first 7 variables in terms of p-value are visualised, separated by class, with a histogram and the density from the estimated *fgld*. It can be noted how the *fgld* can capture the skewness present in the first two original variables.

	p = 2	p = 50	p = 100	p = 150	p = 200
KNN k = 3	92.42	65.15	54.55	43.94	43.94
LDA	90.91	72.73	54.55	57.58	46.97
Logistic regression	95.45	56.06	53.03	53.03	53.03
naïve Bayes discrete	84.85	87.88	84.85	71.21	63.64
naïve Bayes *fgld*	95.45	87.88	78.79	69.70	60.61
naïve Bayes KDE	93.94	92.42	95.45	81.82	69.70
naïve Bayes normal	93.94	92.42	90.91	84.85	77.27

Table 2.6: Leave-one-out cross validation accuracy for different classification algorithms applied to the `bankruptcy` dataset with added noise variables. The columns are for different numbers of variables (p), being the ones with the lowest p-values for the *fgld* test.

We can note that the naïve Bayes with the *fgld* reaches its maximum with $p = 2$, that is with the original variables. This is the best accuracy obtained in a leave-one-out cross validation scheme, and the method is the best strategy together with logistic regression. As more and more noise variables are included the performance of all methods deteriorates, with the naïve Bayes

classifiers being pretty robust. This robustness, in particular of the normal and KDE naïve Bayes classifiers, has also been noted by the fact that it can happen that they retain or improve their accuracy even in presence of a moderate number of noisy variables, probably due random changes related to the small number of units n. However, the improvement given by the selection is sizable for all methods, and most of them benefit from the selection given by the *fgld* test, reaching very high accuracies when only the two original variables are included.

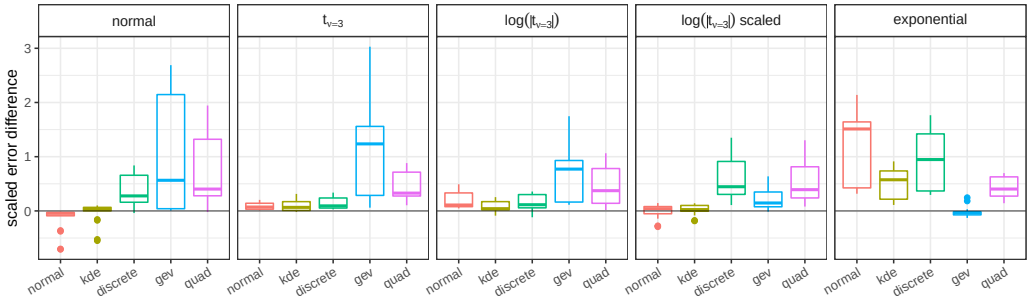

Figure 2.3: Results from a simulation study comparing different methods for the naïve Bayes classifier. Each panel represents a distributional scenario under which the data was simulated. Results are presented as scaled differences from the *fgld*, where a value higher than 0 means that for a setting (combination of sample size, number of variable and correlation structure) the method had a larger mean misclassification error than the *fgld*.

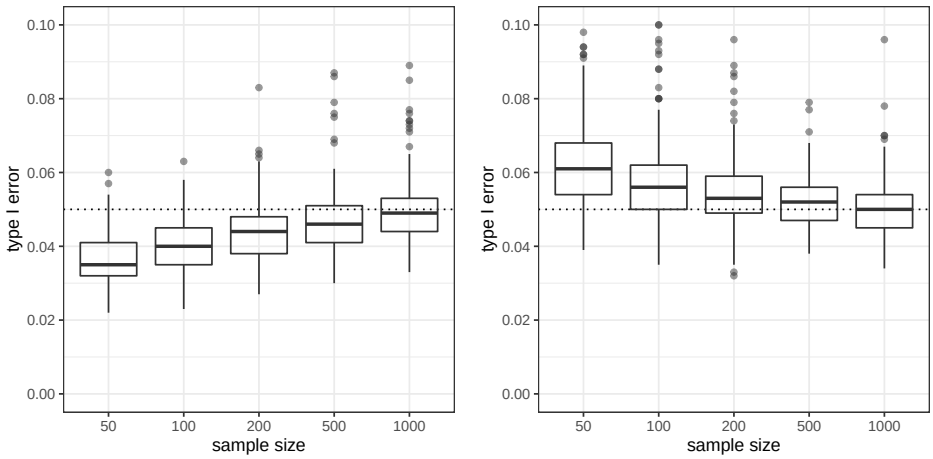

Figure 2.4: Distribution of empirical type I errors across 200 parameter sets for the *fgld* (left panel) and *quad* (right panel) for different group sample sizes. As the sample size increases, empirical type I errors get closer to their nominal 5% value. The left panel refers to the *fgld*, the right panel to the *quad*.

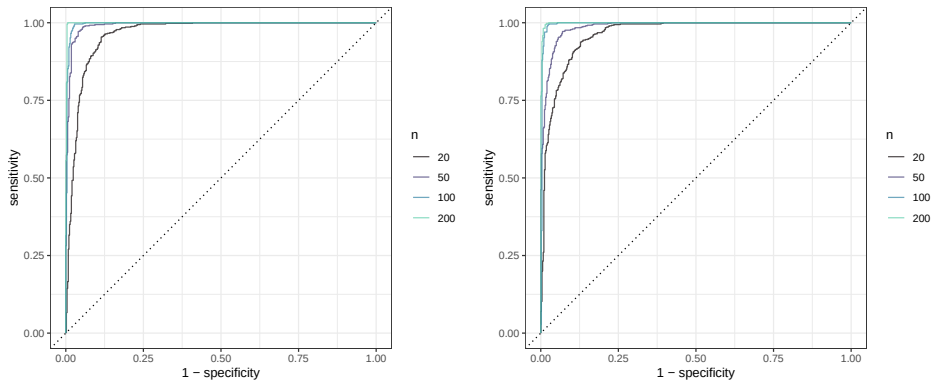

Figure 2.5: ROC curves based on the identification of whether a variable has the same distribution across two groups. Results are obtained by computing the hypothesis test across 1,000 variables, of which only half have the same distribution across the two groups.

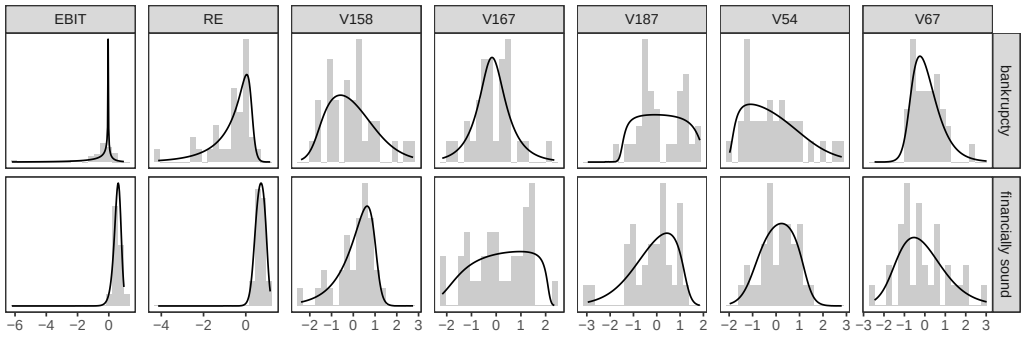

Figure 2.6: Naïve Bayes classifier with *fgld* applied to the `bankruptcy` dataset with added noise variables. The 7 variables with the lowest p-values are shown on the columns, while the rows identify the response class. The visualisation includes a histogram and the density function from the estimated *fgld*.

Chapter 3

Bayesian Estimation of a Quantile-based Factor Model

3.1 Introduction

Specifying a quantile function offers a valid approach to defining a continuous random variable as an alternative to the conventional probability density function (pdf) or cumulative distribution function (cdf). The resulting quantile-based distributions offer a means to define flexible distributions with only a few parameters. This is often achieved by specifying analytical expressions involving quantile functions for simple and common distributions. Adhering to certain rules ensures that the resulting expression remains a proper quantile function: a notable example of a property enjoyed by quantile functions is their closure with respect to convex linear combinations. This innovative approach for the construction of general quantile functions was introduced by Gilchrist (2000).

The main disadvantage of working with quantile-based distributions is that estimation procedures that are routinely employed, especially Bayesian inference and maximum likelihood procedures, become more computationally expensive. This is due to the fact that, for a quantile-based distribution the density function is generally not available analytically and can be only evaluated with the help of more computationally expensive numerical inversion methods, whereas most of statistics relies on the likelihood function, which entails the evaluation of several density functions. Other methods have been extensively

used in this context, particularly L-moments estimation (Hosking, 1990), and other quick and reliable procedures are available such as least squares (Redivo et al., 2023), as shown in the previous chapter.

Recently however, there has been renewed interest in the Bayesian estimation of quantile-based distributions, in particular thanks to a recent article by Perepolkin et al. (2023) that systematises such procedures and gives suggestions regarding the computational tools to adopt.

In the present work, we will begin by providing a brief review of the Bayesian estimation of the flattened generalized logistic distribution (*fgld*), which has been initially introduced in Chakrabarty and Sharma (2021). Next, we employ this quantile-based distribution to model the univariate distribution of the latent variables within a factor model, assuming that the factors are independent. This method is somewhat similar to the independent factor analysis (Attias, 1999; Montanari and Viroli, 2010b), where the probability density function for the latent variables is defined by mixtures of Gaussians.

To account for identifiability and estimation constraints, we use MCMC (Markov Chain Monte Carlo) methods for model estimation. Moreover, we explore the issue of model selection related to the number of factors using information criteria. Finally, we illustrate the use of the proposed model using data from the European Social Survey, which pertains to opinions on trust, ideals, and the functioning of institutions and democracy; we also compare the results with those obtained by using the classical factor analysis and the independent factor analysis models.

The remainder of the chapter is structured as follows. In Section 3.2 quantile-based distributions and their Bayesian estimation are introduced. In Section 3.3 we develop the independent factor analysis model, its estimation and discuss model selection. We conclude the chapter with the empirical illustration on the European Social Survey data (Section 3.4).

3.2 Bayesian estimation of quantile-based distributions

The quantile function for a continuous random variable X, having cumulative distribution function $F(x) = P(X \leq x)$, is defined as:

$$Q_X(u) = \inf\{x : F(x) \geq u\},$$

with $0 \leq u \leq 1$. If $F(x)$ is strictly increasing, then the quantile function can be defined simply as the inverse function of the cdf:

$$Q_X(u) = x = F^{-1}(u).$$

The derivative of the quantile function:

$$q_X(u) = \frac{\partial Q_X(u)}{\partial u},$$

is called quantile density function and provides the key link to the density function of X:

$$f(x) = \frac{1}{q_X(F(x))}. \tag{3.1}$$

Quantile-based distributions are defined via an analytical quantile function and in general we can assume that this will not be analytically invertible, thus not allowing for an expression for the cdf or for the pdf. The evaluation of the pdf of a point must then rely on a numerical inversion of the quantile function:

$$u = Q_X^{-1}(x),$$

which can then be plugged into the quantile density function to evaluate the density:

$$f(x) = f(Q_X(u)) = \frac{1}{q_X(u)}.$$

The numerical inversion needed for computing u is equivalent to finding the root, that is the zero, of the function

$$Q_X(u) - x = 0.$$

In Perepolkin et al. (2023) some possible choices for a root-finding algorithm are listed and we follow their suggestion of using the Brent braketing algorithm, which is available in R (R Core Team, 2023) as the function `uniroot`.

Interestingly, the authors also made available that same algorithm for its use in Stan (Stan Development Team, 2023) in the supplementary materials of that same article.

If we assume that our data $\mathbf{x} = (x_1, \ldots, x_n)$ is made of independent and identically distributed realizations from a random variable X, dependent on a parameter θ, the likelihood can be written as usual:

$$f(\mathbf{x} \mid \theta) = \prod_{i=1}^{n} f(x_i \mid \theta).$$

To evaluate the likelihood for a quantile-based distribution, we must first numerically derive the cdf at all sample points and then use Equation 3.1 for evaluating the density at each point:

$$\begin{aligned} &\mathbf{u} = (u_1, \ldots, u_n) \qquad u_i = Q^{-1}_{X|\theta}(x_i) \\ &f(\mathbf{x} \mid \theta) = \prod_{i=1}^{n} \frac{1}{q_{X|\theta}(u_i)}. \end{aligned} \tag{3.2}$$

To compute the formula of Bayesian inference that gives us the unnormalized posterior distribution of the parameter θ:

$$f(\theta \mid \mathbf{x}) \propto f(\mathbf{x} \mid \theta) f(\theta),$$

we are only missing the prior distribution for the parameter, $f(\theta)$. This can either have a density-based distribution, which is thus readily available for evaluation, or can in turn be defined via a quantile-based distribution. For the latter case, that will not be pursued here, refer to Nair et al. (2022) and Perepolkin et al. (2023).

Deriving a conjugate family from a quantile-based distribution seems like a tall order, given that few common distributions have an analytical density quantile function that can be recognized as part of the posterior distribution. Nevertheless, in Nair et al. (2022), formulas are given for deriving the quantile function of the posterior distribution and some approximations are proposed to estimate posterior mean and median.

The solution that will be taken here is instead that of relying on Markov Chain Monte Carlo (MCMC) methods to obtain samples from the posterior distribution. This procedure has been first explored in Haynes and Mengersen

(2005) for the g-and-k distribution. In particular they relied on the Metropolis-Hastings (MH) algorithm, which can be easily adapted to the case at hand by evaluating the likelihood using Equation 3.2. More specifically, the MH algorithm for obtaining samples from the posterior distribution defined with a quantile-based likelihood can be written as:

- Initialize the parameter(s) $\theta = \theta^{(0)}$.
- For each iteration $t = 1, \ldots, T$:
  - Sample θ^* from a proposal distribution $g(\theta \mid \theta^{(t-1)})$.
  - Evaluate the unnormalized posterior at the new value θ^*:

$$u_i^* = Q_{X|\theta^*}^{-1}(x_i) \qquad i = 1, \ldots, n$$
$$f(\theta^* \mid \mathbf{x}) \propto \prod_{i=1}^{n} \frac{1}{q_{X|\theta^*}(u_i^*)} f(\theta^*)$$

  - Take the proposal as the new value of the chain $\theta^{(t)} = \theta^*$ with probability r or stay at the previous iteration of the chain $\theta^{(t)} = \theta^{(t-1)}$ with probability $1 - r$, with r is defined as:

$$r = \min\left(\frac{f(\theta^* \mid \mathbf{x}) g(\theta^{(t-1)} \mid \theta^*)}{f(\theta^{(t)} \mid \mathbf{x}) g(\theta^* \mid \theta^{(t-1)})}, 1\right)$$

The computational bottleneck of this algorithm is given by the evaluation of the unnormalized posterior, due to the numerical inversion needed for computing u_i^*. This also implies that the computational complexity grows quickly with the number of observations. To save some computational cost, at each iteration, one should keep the previous value of the posterior density in memory. Moreover, as it is pointed out in Haynes and Mengersen (2005), if we have multiple parameters, $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$, it is recommendable to update them all at once, by having a multivariate proposal distribution, instead of updating them one-at-a-time, thus having to compute the posterior only once per iteration.

3.2.1 Bayesian estimation for the *fgld*

The flattened generalized logistic distribution (*fgld*) (Chakrabarty and Sharma, 2021) adds two parameters to the logistic distribution, one of the few common

distributions having an analytical expression for the quantile function. The first one is δ, allowing for skewness, and the second is κ, allowing for a flatter shape of the density. The quantile function is equal to:

$$Q(u) = \alpha + \beta\left[(1-\delta)\log(u) - \delta\log(1-u) + \kappa u\right]$$

with $\beta > 0$, $0 \leq \delta \leq 1$ and $\kappa \geq 0$. The resulting quantile density function is:

$$q(u) = \beta\left[\frac{1-\delta}{u} + \frac{\delta}{1-u} + \kappa\right],$$

and consequently the log likelihood of data ($\mathbf{x} = (x_1, \ldots, x_n)$) coming from the *fgld* can be written as:

$$\ell(\boldsymbol{\theta}) = f(\mathbf{x} \mid \boldsymbol{\theta}) = -n\log\beta - \sum_{i=1}^{n}\log\left[\frac{1-\delta}{u_i} + \frac{\delta}{1-u_i} + \kappa\right] \tag{3.3}$$

with $\boldsymbol{\theta} = (\alpha, \beta, \delta, \kappa)$ and $u_i = Q^{-1}_{X|\boldsymbol{\theta}}(x_i)$.

The Bayesian estimation of this distribution has been considered in Perepolkin et al. (2023), where it is chosen as the response distribution for a parametric quantile regression model. For our purposes we only consider its unconditional estimation, which is a stepping stone for the factor analysis model presented in Section 3.3.1. In Perepolkin et al. (2023), the algorithm used for the MCMC sampler is the robust adaptive Metropolis (Vihola, 2012), as implemented in the R package `fmcmc` (Yon and Marjoram, 2019), which we will also consider later.

From initial investigations, we found that adopting a simple MH algorithm of the type described in the previous section, with independent proposals for each parameter, leads to poor results. The variances of the proposal densities are the main tuning parameters of such an algorithm: their impact is relevant on the mixing of chains and finding values that work adequately for the data at hand can be quite challenging, thus the algorithm usually gives quite unsatisfactory results. For this reason, we turn to the more efficient adaptive MCMC algorithms, which can automatically tune the proposal variances: we will consider the adaptive Metropolis (AM) (Haario et al., 2001) and the robust adaptive Metropolis (RAM) (Vihola, 2012). Using a similar notation to Vihola (2012), we can frame both of these methods as special cases of the following proposal scheme for a general $p-$dimensional parameter $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}^{*} = \boldsymbol{\theta}^{(t-1)} + \mathbf{L}^{(t-1)}\mathbf{z}^{(t)},$$

where $\mathbf{z}^{(t)}$ is a sample from symmetric distribution (we will use a multivariate normal distribution with identity covariance matrix), $\boldsymbol{\theta}^{(t-1)}$ is the previous value in the chain and $\mathbf{L}^{(t-1)}$ is a non-singular matrix that determines the covariance among the parameters: $\text{var}(\boldsymbol{\theta}^*) = \mathbf{L}^{(t-1)}\mathbf{L}^{(t-1)\top}$. For instance, if $\mathbf{L}^{(t)} = \mathbf{I}_p$ for each t, then the proposal becomes that of a random walk Metropolis. In the adaptive proposals instead, the matrix $\mathbf{L}^{(t-1)}$ is based on the previous samples of the chain: the resulting algorithm loses the Markov chain property, while still maintaining ergodicity under fairly general conditions, allowing the estimation of integrals from the resulting samples. The AM scheme is based on an asymptotic result as the parameter dimension p increases, which tells us that the optimal proposal covariance matrix of a Gaussian proposal density, minimising the asymptotic variance of the MCMC algorithm under certain regularity conditions, is approximately equal to:

$$\text{var}(\boldsymbol{\theta}^*) = \frac{2.38^2}{p}\boldsymbol{\Sigma}, \tag{3.4}$$

where $\boldsymbol{\Sigma}$ is the posterior covariance of the parameters (Roberts et al., 1997). The AM algorithm substitutes $\boldsymbol{\Sigma}$ for $\mathbf{S}_{t-1}$, the covariance matrix among the samples up to iteration $t-1$:

$$\text{var}(\boldsymbol{\theta}^*) = \mathbf{L}^{(t-1)}\mathbf{L}^{(t-1)\top} = \frac{2.38^2}{p}\mathbf{S}_{t-1} + \epsilon\mathbf{I}_p, \tag{3.5}$$

where the additive term is there to ensure invertibility, with ϵ taken as a small value (i.e. 10^{-6}), and $\mathbf{L}^{(t-1)}$ can be derived via the Cholesky decomposition. The RAM scheme, on top of estimating the covariance of the target distribution, has also the goal of coercing the acceptance rate of the Metropolis algorithm, whose optimal mean value across the chain is approximately $r^* = 23.4\%$, a figure that comes from the same asymptotic result mentioned previously. The covariance of the RAM proposal is based on the following equation:

$$\mathbf{L}^{(t)}\mathbf{L}^{(t)\top} = \mathbf{L}^{(t-1)}\left(\mathbf{I}_p + \eta_t(r^{(t)} - r^*)\frac{\mathbf{z}^{(t)}\mathbf{z}^{(t)\top}}{\|\mathbf{z}^{(t)}\|_2^2}\right)\mathbf{L}^{(t-1)\top}.$$

Again, $\mathbf{L}^{(t)}$ can be found as a Cholesky factor of the right hand side, where η_t is a diminishing adaptation factor, which we fix as in Vihola (2012), $\eta_t = t^{-\frac{2}{3}}$, and $r^{(t)}$ is the acceptance probability at the t-th iteration.

To use the AM or the RAM proposal schemes for the *fgld* we need to transform its parameters to being unbounded:

$$\boldsymbol{\theta} = (\alpha, \log(\beta), \Phi^{-1}(\delta), \log(\kappa)),$$

where $\Phi^{-1}(\cdot)$ stands for the quantile function of a standard normal distribution. For simplicity the priors are also defined on this reparametrization:

$$\theta_1 \sim \text{Normal}(\mu = 0, \sigma^2 = 100), \quad \theta_2, \theta_4 \sim \text{Normal}(0, 4), \quad \theta_3 \sim \text{Normal}(0, 1),$$

which imply log-normal priors for the positive parameters β and κ and a uniform for δ, thanks to the probability integral transform[1]. Although the priors for θ_2 and θ_4 might seem restrictive, they imply a 99% quantile above 100 in the scale of the original parameters, thus being still weakly informative priors. The likelihood in Equation 3.3 will also be evaluated in terms of reparametrized parameters, while results will be shown with respect to their original version.

First, we compare the performance of the AM and RAM algorithms that we have just introduced for obtaining posterior samples for the *fgld* parameters. Both of them were implemented in R: for the AM we use the recursive formula of the covariance matrix presented in Haario et al. (2001), while to efficiently update the proposal covariance matrix of the RAM based on Equation 3.5, a function from the R package `ramcmc` (Helske, 2021) is used. On top of this code made specifically for estimating the *fgld* parameters, we also consider the implementation of the two Metropolis algorithms given by the R package `fmcmc`, which has a general purpose function for MCMC sampling to which one can provide a function for the target density. The two Metroplis algorithms both with two implementations were compared on a sample of size 100 from the *fgld* with parameters $\{\alpha = 3, \beta = 1, \delta = 0.7, \kappa = 1\}$; they were run 10 times with random starting points with 20,0000 samples of which the first 10,000 are discarded as burn-in.

Computational results are shown in Table 3.1: we note that AM has sometimes difficulty in reaching the stationary distribution within the fixed number of iterations, which results in the sometimes extremely low effective sample sizes (ESS). Even in the runs where the AM has its highest ESS, however, it is still below the RAM, which is more stable in the ESS across the different

[1]The implied prior is uniform as long as the distribution defined by the quantile function that transforms δ into θ_3, and the prior for θ_3 coincide.

	ESS mean	ESS range	Relative ESS/time	Mean acceptance rate
AM	63	$1 - 381$	0.2	1.1%
AM `fmcmc`	89	$1 - 295$	0.2	4.1%
RAM	385	$243 - 471$	1.0	24.1%
RAM `fmcmc`	299	$230 - 391$	0.5	25.6%

Table 3.1: Comparison of two adaptive Metropolis algorithms (AM and RAM) with two implementations (specialized R code and package `fmcmc`) in terms of effective sample size (ESS), computing time and mean acceptance rate.

chains. The acceptance rate of the RAM, as expected, is very close to its optimal target. We also note that the bespoke R implementation is faster for the both algorithms, and thus the most computationally efficient algorithm seems to be the RAM with a specialized R function.

To test the Bayesian estimation of the *fgld* with this latter approach, we carried out a small simulation study. We considered the following 5 sets of parameters, in which the four numbers refer to $(\alpha, \beta, \delta, \kappa)$:

- `logistic` = $(0, 1, 0.5, 10^{-5})$
- `exponential` = $(0, 1, 10^{-5}, 10^{-5})$
- `fgld1` = $(3, 1, 0.7, 1)$
- `fgld2` = $(0, 1, 0.5, 3)$
- `fgld3` = $(-2, 2, 0.99, 0.3)$.

The first two sets are named after the two common distributions that they reproduce (apart from the fact of κ not being exactly equal to 0 and δ to 1). The shape of the densities is shown in Figure 3.1.

From each distribution we have taken three samples of size $n = \{50, 100, 1000\}$ and we have run the RAM algorithm for 20,000 iterations, of which the first 10,000 serve as burn-in. The resulting posterior distributions for each parameter are displayed in Figure 3.2.

We can see how in general the sampler is able to recover the true values of the parameters, with most credible intervals covering them and becoming smaller as the sample size increases. We have inspected the chains visually and

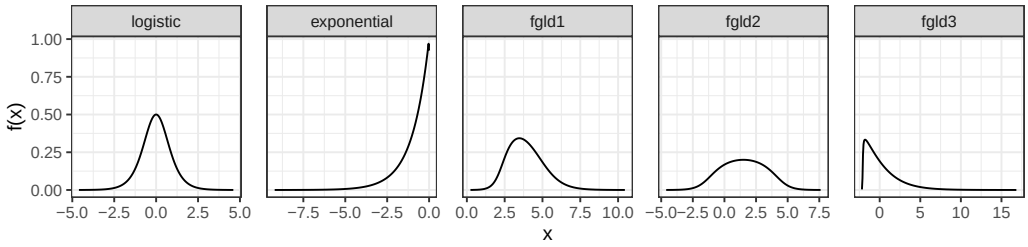

Figure 3.1: Probability density functions for the *fgld* with 5 sets of parameters which include the logistic and exponential distributions as special cases.

in general the convergence to the true values sets in quite fast. With exponential data however, there is sometimes correlation both within and between chains of the different parameters, which deviate temporarily from their stable distribution. This is the reason why the Geweke diagnostic test is significant (at the 1% level) for 3 parameters of the `exponential` with $n = 500$. For all other chains the test is not significant. The average effective sample size, excluding exponential data, where for the aforementioned problems is slightly below 200, is a bit above 400, consistent with the previous computational comparison (Table 3.1).

3.2.2 The standard *fgld*

Given the great versatility shown in Figure 3.1 by the *fgld* shows with different sets of parameters, we are interested in employing it as the distribution for the latent variables in a factor model. In order to achieve the identifiability of the model we need the distribution to be in so-called standard form, that is having expected value equal to 0 and variance equal to 1.

The expressions for both these moments have been derived in Chakrabarty and Sharma (2021):

$$E(X) = \alpha + \beta\left(2\delta - 1 + \frac{\kappa}{2}\right) \tag{3.6}$$

$$\text{var}(X) = \beta^2\left(1 - 4\delta(1-\delta) + \frac{\kappa}{2}\left(1 + \frac{\kappa}{6}\right) + \frac{\pi^2}{3}\delta(1-\delta)\right). \tag{3.7}$$

They are both non linear functions of the parameters and consequently equality constraints involving these expressions cannot be included as is in a convex optimization problem. Luckily instead, in the context of Bayesian Monte

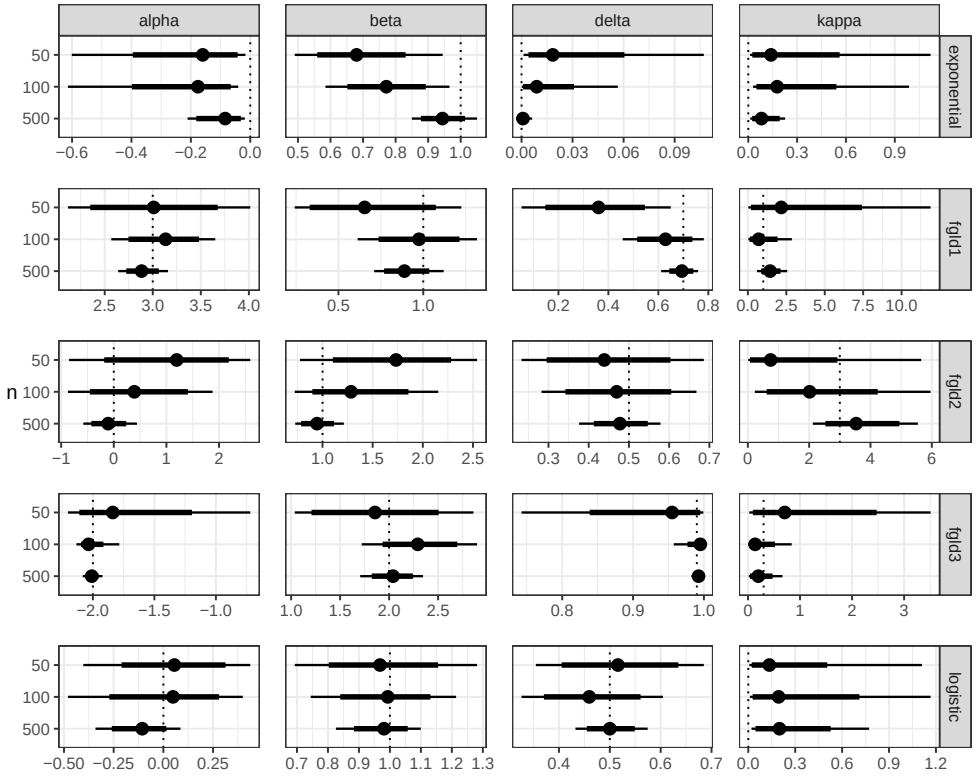

Figure 3.2: Posterior distributions for the parameters of the *fgld* (columns) for three parameter settings (rows); within each panel the results obtained from three samples of increasing size are shown. Results are based on 10,000 posterior samples obtained after 10,000 burn-in iterations. Two nested credible intervals of probabilities 80% and 95% are shown and the point indicates the posterior median (for this the R package `ggdist` was used, Kay (2023)). Vertical dotted lines signal the true value of the parameter.

Carlo methods, where the sampling procedure starts from a proposal value, these constraints can be easily implemented.

At each iteration of the MCMC algorithm, it is enough to propose new values for δ and κ and then compute the values for α and β that meet the moment constraints. The computation of α and β involves solving this simple system:

$$\begin{cases} \alpha + \beta\, h_1(\delta, \kappa) = 0 \\ \beta^2\, h_2(\delta, \kappa) = 1, \end{cases} \tag{3.8}$$

where functions h_1 and h_2 are the collection of the terms involving δ and κ from the expressions in Equation 3.6. Then, the posterior can be evaluated and the algorithm works as usual. In this way we are constraining the sampler to only work within the family of the standard *fgld*.

We have implemented a similar RAM algorithm to the one presented in the previous section, the only difference being that now proposals are only being made for two parameters (δ and κ), with the same specifications in terms of priors as before. The other two parameters α and β, are degenerate random variables, being a deterministic transformation of the other two.

To illustrate the functioning of the algorithm, we have carried out a small simulation study similarly to Section 3.2.1, where three sets of parameters are considered, which aim to exemplify the range of shapes that the standard *fgld* distribution can take. The parameters, in the order $(\alpha, \beta, \delta, \kappa)$, are the following:

- `fgld-std-1`: $(-0.0055, 1.1, 0.5, 0.01)$
- `fgld-std-2`: $(-0.82, 1.0, 0.9, 0.01)$
- `fgld-std-3`: $(-1.2, 0.66, 0.9, 2)$.

The values of α and β are reported approximately as they are a consequence of the other two, and do not in general end up being round numbers. The shape of the resulting density functions is shown in Figure 3.3. The resulting posterior distributions for parameters δ and κ are reported graphically in Figure 3.4. In general, we see that the behaviour of the algorithm is satisfactory. The posterior credible intervals cover the true parameter value in most cases and results significantly improve with the highest sample size. Moreover, there

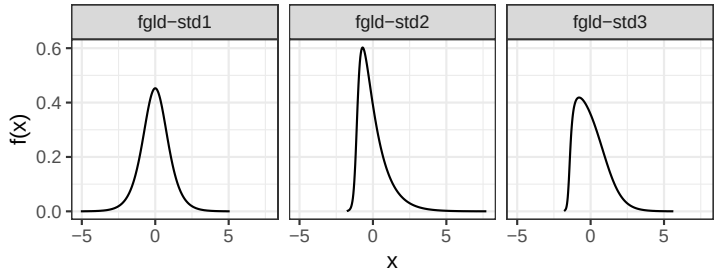

Figure 3.3: Probability density functions for the standard *fgld* under three sets of parameters resulting in symmetric, right-skewed and flattened right-skewed shapes.

does not seem to be problems of convergence, the chains have been visually inspected and they all pass the Geweke diagnostic test at a confidence level of 1%. Finally, the average effective sample size across all chains is around 1100, with little variation among them.

3.3 The independent factor analysis model

The classical factor analysis (FA) is a model for describing a multivariate random variable $\mathbf{x}$ of dimension p in terms of latent causes. It assumes that $\mathbf{x}$, whose realizations are observable, is the linear combination of a small number of latent random variables, called factors, that make up the vector $\mathbf{y}$ of dimension $k \ll p$, with the addition of a random noise component $\mathbf{u}$:

$$\mathbf{x} = \mathbf{\Lambda}\mathbf{y} + \mathbf{u}, \tag{3.9}$$

where the matrix $\mathbf{\Lambda}$ (of dimension $p \times k$), defining the linear combination, is called the matrix of factor loadings. The loadings are the main inferential quantity of interest, as they describe the relation between the observed variables in $\mathbf{x}$ and the latent variables in $\mathbf{y}$, which sometimes can provide an insightful description and summary of the observed data.
Without loss of generality we assume mean-centered data. Hence the distributional assumptions for the classical factor analysis model are the following:

$$\begin{aligned}
&\mathbf{u} \sim \text{Normal}(\mathbf{0}_p, \mathbf{\Psi}) \\
&\mathbf{y} \sim \text{Normal}(\mathbf{0}_k, \mathbf{I}_k) \\
&\mathbf{u} \perp \mathbf{y},
\end{aligned}$$

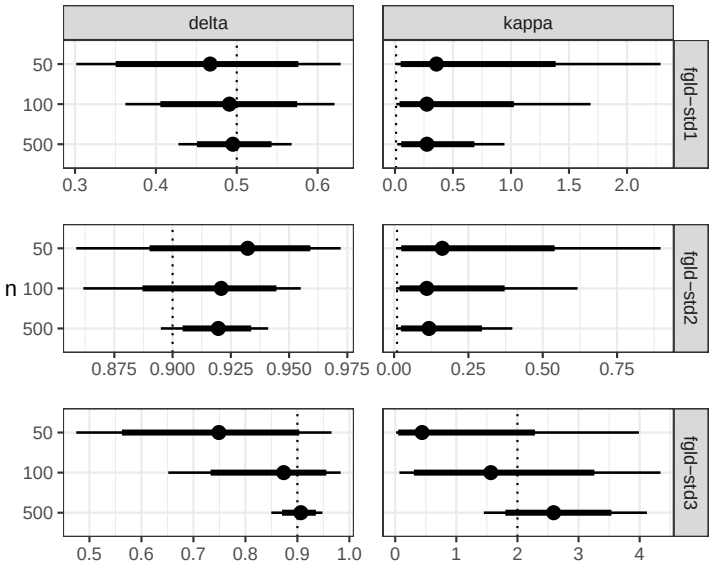

Figure 3.4: Posterior distributions for the parameters of the standard *fgld* (columns) for three parameter settings (rows); within each panel the results obtained from three samples of increasing size are shown. Results are based on 10,000 posterior samples obtained after 10,000 burn-in iterations. Two nested credible intervals of probabilities 80% and 95% are shown and the point indicates the posterior median (for this the R package `ggdist` was used, Kay (2023)). Vertical dotted lines signal the true value of the parameter.

where the two expected values are set to zero. The covariance of $\mathbf{u}$ is a diagonal matrix, while the covariance of $\mathbf{y}$ is set to the identity matrix. This latter assumption is made because the model with an arbitrary covariance matrix $\boldsymbol{\Omega}$ would be indistinguishable from one in which in which $\mathbf{y}^* = \boldsymbol{\Omega}^{-\frac{1}{2}}\mathbf{y}$ and $\boldsymbol{\Lambda}^* = \boldsymbol{\Lambda}\boldsymbol{\Omega}^{\frac{1}{2}}$, that is one having sphered latent variables (Montanari and Viroli, 2010b). This latest consideration points to the main lack of identifiability exhibited by this model, its invariance under orthogonal rotations, that is:

$$\mathbf{x} = \boldsymbol{\Lambda}\mathbf{y} + \mathbf{u} = \boldsymbol{\Lambda}\mathbf{G}^\top\mathbf{G}\mathbf{y} + \mathbf{u} = \boldsymbol{\Lambda}^*\mathbf{y}^* + \mathbf{u},$$

where $\mathbf{G}$ is orthogonal $(\mathbf{G}^\top\mathbf{G} = \mathbf{I})$, $\boldsymbol{\Lambda}^* = \boldsymbol{\Lambda}\mathbf{G}^\top$ and $\mathbf{y}^* = \mathbf{G}\mathbf{y} \sim \text{Normal}(\mathbf{0}_k, \mathbf{I}_k)$.

Besides being interpretable from the perspective of latent variables, the FA model can also be seen as a sparse or parsimonious covariance estimation method for normal data, in fact the model can simply be described as:

$$\mathbf{x} \sim \text{Normal}(\mathbf{0}_p, \boldsymbol{\Sigma})$$

with

$$\boldsymbol{\Sigma} = \boldsymbol{\Lambda}\boldsymbol{\Lambda}^\top + \boldsymbol{\Psi}. \tag{3.10}$$

This simplification can be derived either by integrating out the latent variable $\mathbf{y}$ or by considering the closure of normal distributions (in this case $\mathbf{y}$ and $\mathbf{u}$) with respect to sums.

A generalization of the normal factor model is the so-called independent factor analysis (IFA) model (Attias, 1999; Montanari and Viroli, 2010b). The main difference is that the latent variables $\mathbf{y}$ are allowed to take a more general non-Gaussian distribution, in particular they are considered to be a vector made up of mutually independent univariate Gaussian mixtures:

$$\begin{aligned} &\mathbf{y} = (y_1, \ldots, y_k) \qquad \text{with mutually independent entries} \\ &f(y_j) = \sum_{g_j=1}^{G_j} w_{jg_j}\mathcal{N}(\mu_{jg_j}, \sigma_{jg_j}) \qquad j = 1, \ldots, k, \end{aligned}$$

where g_j indicates the generic component of the mixture describing the j-th latent variable having a total number of G_j components. In contrast to normal distributions, where uncorrelatedness is equivalent to independence, in this setting the independence assumption among the factors needs to be made explicitly, which explains the name of the model.

In the IFA model we make the same assumptions about the distribution of the error term $\mathbf{u}$ and about the independence between $\mathbf{y}$ and $\mathbf{u}$ as in the FA model. Moreover, also the moment assumptions about $\mathbf{y}$ coincide. In principle the covariance matrix of $\mathbf{y}$ could be any diagonal matrix, but similarly to the FA case, the model still has an identifiability problem, which in this case is the invariance with respect to scale transformations:

$$\mathbf{x} = \mathbf{\Lambda}\mathbf{y} + \mathbf{u} = \mathbf{\Lambda}\mathbf{D}^{\frac{1}{2}}\mathbf{D}^{-\frac{1}{2}}\mathbf{y} + \mathbf{u} = \mathbf{\Lambda}^*\mathbf{y}^* + \mathbf{u}, \tag{3.11}$$

where $\mathbf{D}$ is diagonal. In Montanari and Viroli (2010b) the relationship between FA and IFA is explored more in depth, and also connections to Independent Component Analysis (ICA) are considered. In that same article, the IFA model is shown to be interpretable as a mixture of factor analysers: if we introduce a latent indicator variable $\mathbf{z}$ for the allocation of the elements of $\mathbf{y}$ in their mixture components, the distribution of $\mathbf{y}$ conditional on $\mathbf{z}$ becomes normal:

$$\mathbf{y} \mid \mathbf{z} \sim \text{Normal}(\boldsymbol{\mu}_z, \mathbf{V}_z)$$

and similarly to the FA case, the variable $\mathbf{y}$ can be integrated out obtaining:

$$\mathbf{x} \mid \mathbf{z} \sim \text{Normal}(\mathbf{\Lambda}\boldsymbol{\mu}_z, \mathbf{\Lambda}\mathbf{V}_z\mathbf{\Lambda}^\top + \mathbf{\Psi}).$$

This distribution can be seen as part of the complete likelihood from a mixture of factor analysers model, that is a normal mixture where the covariance of each component is modelled via the FA covariance structure shown in Equation 3.10. This framing of the model allows for the estimation via an EM algorithm similar to the one used for Gaussian mixture models: this algorithm is presented and explored in Montanari and Viroli (2010b). The IFA model is estimated from a Bayesian perspective using a Gibbs sampler in Viroli (2007). The results shown there will serve as a blueprint for the estimation of our novel IFA model, which will be described in the following section.

The IFA model, thanks to its mixture modelling approach, allows for the factor variables having arbitrarily flexible distributions, however, it comes with the added burden of selecting the number of components for each element of the latent variable. An alternative approach is using a distribution with parameters that affect moments higher than second, such as skewness and kurtosis, thus parametrically estimating non-Gaussian factors. This strategy obviates

the need of model selection within each factor, while in the process it loses the possibility of having multi-modal factors.

This issue has been explored in Montanari and Viroli (2010c), where the factors are modelled with the multivariate skew normal (MSN) distribution (Azzalini and Capitanio, 1999). In particular a property of the MSN distribution is utilized: any starting MSN distribution ($\mathbf{z}$) can be transformed, via an invertible affine function ($\mathbf{y} = \mathbf{A}\mathbf{z}$), to another MSN distribution with independent components that are all standard normal expect for one, which is univariate skew normal, in a sense absorbing all the skewness of the starting multivariate distribution. Using this canonical form of the distribution for the latent vector $\mathbf{y}$ greatly simplifies the estimation problem and moreover, it puts the model within the context of IFA. Once a solution with factors having the canonical form of the MSN is found, a rotation can be applied to the loadings to possibly simplify their interpretation; although the canonical form is lost, the rotated $\mathbf{y}$ remains within the MSN family.

The IFA model can be seen as a special case of a factor model where the joint vector of factors is modelled as a multivariate Gaussian mixture:

$$f(\mathbf{y}) = \sum_{g=1}^{G} w_g \mathcal{N}\left(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g\right).$$

This model, called heteroskedastic factor mixture analysis, has been introduced in Montanari and Viroli (2010a) and has recently been extended in a Bayesian context in Chandra et al. (2023), where it is called Lamb; here the number of components G can go to infinity, thanks to non-parametric priors on the weights, such as the Dirichlet Process. In the latter article, the model is motivated by high-dimensional clustering, and very interestingly it is shown to avoid some pitfalls, investigated in the same article, that affect model-based clustering. If the clustering is performed in the original high-dimensional space, common choices for the group covariance structure lead to asymptotic (as p increases) posteriors for the partition that assign either all observations to one component or each observation to a different component. The representation of the model when integrating out the latent factors becomes:

$$\mathbf{x} \sim \sum_{g=1}^{G} w_g \mathcal{N}\left(\boldsymbol{\Lambda}\boldsymbol{\mu}_g, \boldsymbol{\Lambda}\boldsymbol{\Sigma}_g\boldsymbol{\Lambda}^\top + \boldsymbol{\Psi}\right). \tag{3.12}$$

In IFA the total number of components for the mixture model of the latent factors is $G = \prod_{j=1}^{k} G_j$, that is any combination of the univariate mixture components defines a component in the multivariate mixture. Given the independence among the univariate mixture, it can only define diagonal matrices in the component variances $\mathbf{\Sigma}_g$, making it a special case of the model shown in Equation 3.12.

3.3.1 IFA model with standard *fgld* distributed factors

The main model equation takes the same form as IFA:

$$\mathbf{x} = \mathbf{\Lambda y} + \mathbf{u},$$

the only difference being the distribution used for the random vector of factors $\mathbf{y}$:

$$\begin{aligned} &\mathbf{y} = (y_1, \ldots, y_k) \qquad \text{with mutually independent entries} \\ &y_j \mid \boldsymbol{\theta}_j \sim \text{Standard fgld}(\boldsymbol{\theta}_j) \qquad j = 1, \ldots, k \end{aligned}$$

where $\boldsymbol{\theta}_j = (\alpha_j, \beta_j, \delta_j, \kappa_j)$, with the constraints presented in Section 3.2.2. With $\boldsymbol{\theta}$ we will denote the collection of all these parameters across the k latent variables. Moreover, for ease of notation, we will denote as

$$f(\mathbf{y} \mid \boldsymbol{\theta}) = \prod_{j=1}^{k} f(y_j \mid \boldsymbol{\theta}_j)$$

the joint density function of the latent variables, even though these density functions are not available in analytical form, each of them being modelled with a quantile-based distribution.

In Montanari and Viroli (2010c) a formal justification for using a non-Gaussian distribution for the factors is developed. It is noted that a factor model should be invariant to reversals in the direction of measurement, which translates to the factor distribution being closed with respect to a change of sign. This is the case for the *fgld*: if $X \sim \text{fgld}(\alpha, \beta, \delta, \kappa)$, then $-X \sim \text{fgld}(-\alpha - \kappa\beta,\ \beta,\ 1-\delta,\ \kappa)$, which can be derived by applying the so-called reflection-rule: $Q_{-X}(u) = -Q_X(1-u)$ (Gilchrist, 2000).

In order to estimate the model, we develop an MCMC algorithm, and for this we must first derive the unnormalized posterior distribution. Introducing

the latent variables $\mathbf{y}$ as part of the parameters, the likelihood of the model takes the following form:

$$\mathbf{x} \mid \mathbf{y}, \mathbf{\Lambda}, \mathbf{\Psi} \sim \text{Normal}(\mathbf{\Lambda}\mathbf{y}, \mathbf{\Psi}),$$

while the posterior of interest is the joint distribution of parameters and latent variables conditional on the observed data $\mathbf{x}$:

$$f(\mathbf{y}, \mathbf{\Lambda}, \mathbf{\Psi}, \boldsymbol{\theta} \mid \mathbf{x}) \propto f(\mathbf{x} \mid \mathbf{y}, \mathbf{\Lambda}, \mathbf{\Psi})\, f(\mathbf{y} \mid \boldsymbol{\theta})\, f(\boldsymbol{\theta})\, f(\mathbf{\Lambda})\, f(\mathbf{\Psi}).$$

The prior distributions for each $\boldsymbol{\theta}_j$ are set as follows:

$$\begin{aligned} \delta_j &\sim \text{Uniform}(0, 1) \\ \kappa_j &\sim \text{Log-normal}(\mu = 0, \sigma = 4). \end{aligned}$$

The matrix $\mathbf{\Psi}$ is diagonal with entries denoted as $(\Psi_1, \ldots, \Psi_p)$, for each of them the prior is set as:

$$\Psi_l \sim \text{Inverse-gamma}(a_0, b_0)$$

For matrix $\mathbf{\Lambda}$ we define a prior each of its rows, denoted as $\boldsymbol{\lambda}_l$

$$\boldsymbol{\lambda}_l \sim \text{Normal}(\mathbf{0}, \mathbf{I}_k).$$

This is a routinely employed prior for the factor loadings in Bayesian models, and, for instance, is the same one used in Ghosh and Dunson (2009).

For the MCMC algorithm we implement a Metropolis-within-Gibbs strategy (Robert and Casella, 2010), where the outer part of the algorithm is a Gibbs sampler, which means that each block of parameters is sampled from its full conditional distribution. However, in contrast to the IFA model with normal mixtures (Viroli, 2007), some full conditional distributions cannot be sampled from directly and for them we employ a Metropolis-Hastings algorithm.

3.3.2 Derivation of the full conditional distributions

Until now, for ease of notation, we have considered a single random variable $\mathbf{x}$ of dimension p. In this section instead, for completeness, we will consider the usual scenario where our data is made of n samples from $\mathbf{x}$, denoted as $\mathbf{x}_i = (x_{i1}, \ldots, x_{il}, \ldots, x_{ip})$, with $i = 1, \ldots, n$. The samples will be collectively

referred to as $\mathbf{X}$, a matrix of dimension $n \times p$ having $\mathbf{x}_i$ as the i-th row. The same notation is used for the latent vectors $\mathbf{y}_i = (y_{i1}, \ldots, y_{ij}, \ldots, y_{ik})$, jointly denoted as $\mathbf{Y}$, an $n \times k$ matrix having $\mathbf{y}_i$ as the i-th row. The likelihood becomes:

$$f(\mathbf{X} \mid \mathbf{Y}, \mathbf{\Lambda}, \mathbf{\Psi}) \propto \det(\mathbf{\Psi})^{-\frac{n}{2}} \exp\left(-\frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i - \mathbf{\Lambda}\mathbf{y}_i)^\top \mathbf{\Psi}^{-1}(\mathbf{x}_i - \mathbf{\Lambda}\mathbf{y}_i)\right) \tag{3.13}$$

The full conditional distribution of the generic element Ψ_l of matrix $\mathbf{\Psi}$ will be equal to:

$$f(\Psi_l \mid \cdot) \propto \Psi_l^{-\frac{n}{2}-a_0-1} \exp\left(-\frac{1}{\Psi_l}\frac{\sum_{i=1}^{n}(x_{il} - \boldsymbol{\lambda}_l^\top \mathbf{y}_i)^2}{2} - \frac{b_0}{\Psi_l}\right),$$

where $\boldsymbol{\lambda}_l$ denotes the l-th row of $\mathbf{\Lambda}$. This expression can be recognized as being proportional to an Inverse-gamma distribution:

$$\Psi_l \mid \cdot \sim \text{Inverse-gamma}\left(a_0 + \frac{n}{2}, b_0 + \frac{\sum_{i=1}^{n}(x_{il} - \boldsymbol{\lambda}_l^\top \mathbf{y}_i)^2}{2}\right).$$

The full conditional distribution for each latent variable $\mathbf{y}_i$ will be equal to:

$$f(\mathbf{y}_i \mid \cdot) \propto \exp\left(-\frac{1}{2}\mathbf{y}_i^\top \mathbf{\Lambda}^\top \mathbf{\Psi}^{-1}\mathbf{\Lambda} + \mathbf{y}_i^\top \mathbf{\Lambda}\mathbf{\Psi}^{-1}\mathbf{x}_i\right) \prod_{j=1}^{k} f(y_{ij} \mid \boldsymbol{\theta}_j).$$

We cannot sample directly from this expression, so we use a random walk Metropolis, where to improve the performance we employ the so-called Laplace approximation. The idea is to approximate the posterior covariance matrix in the asymptotically optimal proposal covariance (see Equation 3.4) with the Fisher information matrix evaluated at the maximum likelihood estimate (see Chopin and Ridgway (2017) for an extensive treatment and comparison with other algorithms). In the case at hand, wanting a sample from the full conditional posterior of $\mathbf{y}_i$, the density of $\mathbf{x}_i \mid \mathbf{y}_i$ plays the role of the likelihood, and thus we have that the Fisher information is equal to $\mathbf{\Lambda}^\top \mathbf{\Psi}^{-1}\mathbf{\Lambda}$, for what is effectively a linear model ($\mathbf{x}_i = \mathbf{\Lambda}\mathbf{y} + \mathbf{u}$). This expression is used as the proposal covariance matrix for elements of $\mathbf{y}_i$ and it is evaluated at the current values of the parameters, which does not require the further computation of the maximum likelihood estimate. This procedure leads to better performance both

in terms of time and mixing, than using a fixed scale random walk Metropolis updating the elements of $\mathbf{y}_i$ one-at-a-time.

A Metropolis-Hastings algorithm also needs to be employed to sample from the full conditional distribution of $\boldsymbol{\theta}_j$:

$$f(\boldsymbol{\theta}_j \mid \cdot) \propto \prod_{i=1}^{n} f(y_{ij} \mid \boldsymbol{\theta}_j) f(\boldsymbol{\theta}_j).$$

If we consider $(y_{1j}, \ldots, y_{nj})$ as a sample, this full conditional distribution simply defines the posterior distribution for a standard *fgld*: therefore we can use the RAM algorithm presented in Section 3.2.2 to sample from it.

The full conditional for each row of $\boldsymbol{\Lambda}$ is a multivariate normal distribution:

$$\boldsymbol{\lambda}_l \mid \cdot \sim \text{Normal}\left(\frac{1}{\Psi_l}\mathbf{S}_\lambda \mathbf{Y}^\top \mathbf{x}_{[l]}, \mathbf{S}_{\boldsymbol{\lambda}}\right)$$

with $\mathbf{S}_\lambda = \left(\mathbf{I}_k + \frac{1}{\Psi_l}\mathbf{Y}^\top\mathbf{Y}\right)^{-1}$, where $l = 1, \ldots, p$.

The MCMC algorithm also needs to take into account the identifiability issues that come with a latent factor model. Given that this is effectively an IFA model, the issue relates to an invariance of the model under scale transformations as shown in Equation 3.11.

A first solution we consider is applying a specific scale transformation at each iteration of the algorithm so that both the factors and the loadings stay at a particular solution. The transformation employed is the standardization of the latent factors. Denote by $\mathbf{D}$ the $k \times k$ diagonal matrix containing the variances of the latent factors, then after having derived a new sample for $\mathbf{Y}$ from its full conditional distribution, it is transformed to $\mathbf{Y}^* = \mathbf{Y}\mathbf{D}^{-\frac{1}{2}}$, and the inverse transformation is applied to the factor loadings $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^*\mathbf{D}^{-\frac{1}{2}}$.

The second solution is similar to that employed in Ghosh and Dunson (2009), where the authors use the technique of parameter expansion: the model from which sample are taken is overparameterized, that is it has some unidentifiable parameters. This, counter-intuitively, allows for more efficient sampling and from the unidentifiable working parameters, the inferential parameters of interest can still be recovered by taking the appropriate transformations that link the two sets of parameters. This transformation can be done even after the MCMC chain is derived, making it a so-called offline procedure. The Ghosh-Dunson model and its sampling algorithm are described in Appendix B. In our

case, the idea is to work not with a standard *fgld*, but with its centered version, which means only enforcing the first of Equations 3.8, thus allowing for the variance of the latent factors to be different from 1. The working model is thus:

$$\mathbf{x} = \mathbf{\Lambda}^* \mathbf{y}^* + \mathbf{u}$$

where $\mathbf{y}^* = \mathbf{\Phi}^{\frac{1}{2}} \mathbf{y}$, with $\mathbf{\Phi}^{\frac{1}{2}}$ being a diagonal matrix that contains the standard deviations of the latent factors, and $\mathbf{y}$ being the vector of standardized latent factors. Consequently, the inferential value for the matrix of factor loadings is: $\mathbf{\Lambda} = \mathbf{\Lambda}^* \mathbf{\Phi}^{\frac{1}{2}}$. In the case of the *fgld*, the standard deviation at each iteration can be computed based on Equation 3.6. Moreover, the parameters δ and κ do not change under a scale transformation of an *fgld* random vector, so their posterior chains can be used without transformations.

We compare the two algorithms given by the two identification strategies, which we refer to as scaling and parameter expansion, in a small simulation study similar to that of Section 3.2.1, whose results are shown in Table 3.2, based on 10 runs of the two algorithms on data generated with the following setup: $n = 100$, $p = 7$, $k = 2$,

$$\begin{aligned}
&\mathbf{\Lambda}^\top = \begin{bmatrix} -2 & 0.5 & 1.2 & 1 & -2.5 & 0 & 1 \\ 0.1 & 3 & 0.5 & 1.5 & -0.5 & -2 & 0 \end{bmatrix}, \\
&\text{diag}(\mathbf{\Psi}) = (0.15, 0.1, 0.02, 0.05, 0.01, 0.1, 0.01), \\
&\boldsymbol{\theta}_1 = (-0.68, 0.68, 0.50, 2.0), \\
&\boldsymbol{\theta}_2 = (-0.94, 0.99, 0.95, 0.1).
\end{aligned}$$

In this setting the scaling approach works better in terms of absolute bias (defined as the absolute value of the difference between the parameter value and its posterior mean). The parameter expansion algorithm works a bit better in terms of autocorrelation in the chains of $\mathbf{\Lambda}$ (testified by the ESS values), which is the main drawback of the first approach, but at the expense in the mixing of the $\boldsymbol{\theta}$ parameters, where there is an extra parameter to be sampled. On the whole the scaling approach seems thus to work better, and, given that we are mostly interested in interpreting the values of $\boldsymbol{\theta}$ and $\mathbf{\Lambda}$, we prefer this algorithm, which we will use in the following.

	Parameter	Absolute bias mean	Absolute bias range	ESS mean	ESS range
Parameter expansion	Λ	0.256	$0.001 - 2.327$	169	$6 - 1,239$
	Ψ	0.032	$0.005 - 0.061$	$2,533$	$718 - 5,326$
	θ	0.264	$0.005 - 0.877$	431	$80 - 765$
Scaling	Λ	0.141	$0.006 - 0.486$	72	$3 - 1,217$
	Ψ	0.033	$0.005 - 0.061$	$2,527$	$753 - 5,206$
	θ	0.189	$0.007 - 0.773$	747	$106 - 1,287$

Table 3.2: Comparison of two algorithms for the IFA with *fgld* model with two identification strategies (parameter expansion and scaling), in terms of absolute bias and effective sample size (ESS).

3.3.3 Using information criteria for choosing the number of factors

In this section we briefly consider the selection of the number of factors in the general factor analysis context. In the setting of a normal factor model estimated via maximum likelihood, we can employ a likelihood ratio test: we iteratively test the hypothesis that the covariance is adequately specified by the structure in Equation 3.10 with an increasing number of factors, and choose the first model for which the null hypothesis is not rejected. This model selection approach is employed in Montanari and Viroli (2010c) to show that a normal FA model cannot satisfactorily identify a model with a skew factor, as the hypothesis is generally rejected for all feasible values of k (the number of latent variables), which are bounded above by the well-known Lederman's condition.

In the Bayesian context instead, the parameter k can be included in the inference process directly, thus solving at the same time both estimation and model selection, at the cost of a greater computational complexity. This approach is taken in Lopes and West (2004) with the use of a reversible jump MCMC algorithm, while in Ghosh and Dunson (2009) they estimate the posterior probabilities for the number of factors: these can be written in terms of Bayes factors, which can in turn be approximated by using the so-called path sampling approach. In each of the two articles, some simulations are shown where the proposed approach works very well, however, among the model selection methods against which they compare, there are also information criteria based on maximum likelihood estimates. In particular, the BIC criterion

is shown to perform as well as the more complex methods based on the estimation of posterior probabilities for the models.

For this reason, in the present manuscript we propose the use of information criteria for the selection of the number of factors, also when estimating the factor model with Bayesian methods. The log-likelihood for the normal factor model can be written in compact form as:

$$\ell(\mathbf{\Sigma}) = -\frac{n}{2}\left(\log\left(2\pi \det\left(\mathbf{\Sigma}\right)\right) + \operatorname{trace}(\mathbf{\Sigma}^{-1}\mathbf{S})\right) \tag{3.14}$$

where

$$\mathbf{\Sigma} = \mathbf{\Lambda}\mathbf{\Lambda}^{\top} + \mathbf{\Psi}$$

and

$$\mathbf{S} = \frac{1}{n}\mathbf{X}^{\top}\mathbf{X},$$

assuming to be working with a centred data matrix $\mathbf{X}$. This is the normal likelihood where the latent factors have been integrated out and can be readily computed for a normal FA model. For the IFA with *fgld* factors the latent factors cannot be easily integrated out and thus the log-likelihood needs to be computed by taking the logarithm of the expression in Equation 3.13.

The BIC evaluates Equation 3.14 at the maximum likelihood estimate $\hat{\mathbf{\Sigma}}$ and is equal to:

$$\mathrm{BIC} = -2\ell(\hat{\mathbf{\Sigma}}) + \log\left(n\right) \cdot p_{\mathrm{FA}}$$

where

$$p_{\mathrm{FA}} = p(k+1) - \frac{k(k-1)}{2}, \tag{3.15}$$

is the effective number of parameters of the model. The use of this criterion for the FA model has been debated (see for instance Drton and Plummer (2017)), as some of the theoretical justifications for its use do not hold. Moreover, it is not a criterion specifically designed for Bayesian inference. For these reasons we also consider the use of two fully Bayesian information criteria, which are the Deviance Information Criterion (DIC) (Spiegelhalter et al., 2002) and the Widely Applicable Information Criterion (WAIC) (Watanabe, 2010). The DIC is defined as:

$$\mathrm{DIC} = -2\log p(x \mid \hat{\theta}_{\mathrm{Bayes}}) + 2p_{\mathrm{DIC}}$$

where the notation is taken from Gelman et al. (2013): $\log p(x \mid \hat{\theta}_{\mathrm{Bayes}})$ stands for the log-likelihood evaluated at the posterior mean of the parameters ($\hat{\theta}_{\mathrm{Bayes}}$),

and p_{DIC} is the effective number of parameters computed as:

$$p_{\text{DIC}} = 2 \operatorname{var}_{t=1}^{T} \left(\log p(x \mid \theta^{(t)}) \right),$$

that is two times the sample variance of the log-likelihood across all T posterior samples. With a similar notation, the WAIC is defined as:

$$\text{WAIC} = -2 \, \text{lppd} + 2 \, p_{\text{WAIC2}},$$

where the first part is the computed log pointwise posterior predictive density:

$$\text{lppd} = \sum_{i=1}^{n} \log \left(\frac{1}{T} \sum_{t=1}^{T} p(x_i \mid \theta^{(t)}) \right),$$

and the second part is again a correction for effective number of parameters:

$$p_{\text{WAIC2}} = \sum_{i=1}^{n} \operatorname{var}_{t=1}^{T} \left(\log p(x_i \mid \theta^{(t)}) \right).$$

We also consider the BICM criterion (Raftery et al., 2007), a posterior simulation-based version of the BIC, which is computed as:

$$\text{BICM} = 2\hat{\ell}_{\max} + \log(n) \cdot \hat{p}$$

with

$$\hat{\ell}_{\max} = \frac{1}{T} \sum_{t=1}^{T} \log p(x \mid \theta^{(t)}) + \operatorname{var}_{t=1}^{T} \left(\log p(x \mid \theta^{(t)}) \right)$$

and $\hat{p} = p_{\text{DIC}}$.

To test the efficacy of these information criteria in selecting the right number of factors in the normal factor model, we revisit two simulation studies carried out in Ghosh and Dunson (2009), the first one of which originally proposed in Lopes and West (2004). We use the Gibbs sampler algorithm for the Bayesian normal factor model introduced in Ghosh and Dunson (2009), which allows for a fast and efficient posterior sampling.

In the first simulation we have a one-factor ($k = 1$) model with the following settings: $n = 100$, $p = 7$,

$$\mathbf{\Lambda} = (0.995, 0.975, 0.949, 0.922, 0.894, 0.866, 0.837)^{\top}$$

and

$$\text{diag}(\boldsymbol{\Psi}) = (0.01, 0.05, 0.10, 0.15, 0.20, 0.25, 0.30).$$

The maximum number of identifiable factors is 3; we run the selection for 100 simulated datasets. We run for 25,000 iterations with the first 5,000 as burn-in. To check the compatibility of results with the original simulations we also consider the BIC computed from the maximum-likelihood estimate among the criteria. The results are shown in Table 3.3: we can see that all criteria work very well in this simple case.

	k = 1	k = 2	k = 3
BIC	100	0	0
BICM	100	0	0
DIC	98	2	0
WAIC	100	0	0

Table 3.3: Frequency of selected model according to different information criteria. For the BIC the maximum likelihood estimate was used, while for the others are based on Ghosh-Dunson Bayesian factor model.

We also applied the Bayesian information criteria to the IFA model with the standard *fgld*. The settings are the same, we just add the parameters for the distribution of the factor: $\boldsymbol{\theta} = (\alpha = -1.05, \beta = 0.81, \delta = 0.90, \kappa = 1.00)$. For this model, given its higher computational demand, we have run the selection for 10 simulated datasets and the burn-in was set at 12,500. The results are shown in Table 3.4: these are much more mixed than in the previous case, this might be due to the inclusion of the latent variables in the likelihood. However, the BICM still seems to perform satisfactorily.

	k = 1	k = 2	k = 3
BICM	8	2	0
DIC	6	3	1
WAIC	1	6	3

Table 3.4: Frequency of selected model according to different information criteria, computed from the posterior samples of the IFA model with standard *fgld*.

In the second simulation we have $n = 100$, $p = 7$, and $k = 3$, with the parameters being equal to:

$$\boldsymbol{\Lambda}^\top = \begin{bmatrix} 0.89 & 0.00 & 0.25 & 0.00 & 0.80 & 0.00 & 0.50 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.90 & 0.25 & 0.40 & 0.00 & 0.50 & 0.00 & 0.00 & -0.30 & -0.30 \\ 0.00 & 0.00 & 0.85 & 0.80 & 0.00 & 0.75 & 0.75 & 0.00 & 0.80 & 0.80 \end{bmatrix},$$

and

$$\text{diag}(\boldsymbol{\Psi}) = (0.2079, 0.19, 0.1525, 0.2, 0.36, 0.1875, 0.1875, 1, 0.27, 0.27).$$

The simulation was carried in the same way the previous one and the maximum number of factors considered is 4. The results are shown in Table 3.5: again all criteria perform very well, only the WAIC is slightly less accurate in selecting the correct number of factors. Again we also carried out the simulation for the

	k = 1	k = 2	k = 3	k = 4
BIC	0	0	100	0
BICM	0	0	100	0
DIC	0	0	100	0
WAIC	0	0	90	10

Table 3.5: Frequency of selected model according to different information criteria. For the BIC the maximum likelihood estimate was used, while for the others are based on Ghosh-Dunson Bayesian factor model.

IFA model with the standard *fgld*, the factor parameters were set as:

$$\begin{aligned} \boldsymbol{\theta}_1 &= (-1.05, 0.81, 0.90, 1.00) \\ \boldsymbol{\theta}_2 &= (-0.68, 0.68, 0.50, 2.00) \\ \boldsymbol{\theta}_3 &= (0.14, 0.94, 0.30, 0.50). \end{aligned}$$

The results, shown in Table 3.6, point to the fact that in this scenario the use of information criteria does not work properly in the identification of the true model. We suspect the issue might be due to the use of the complete likelihood, and to test this hypothesis we repeat the two simulations with the normal model, but evaluating the information criteria based on the complete likelihood of the model. Results, shown in Table 3.7, point to a deterioration of

	k = 1	k = 2	k = 3	k = 4
BICM	10	0	0	0
DIC	5	2	2	1
WAIC	0	0	6	4

Table 3.6: Frequency of selected model according to different information criteria, computed from the posterior samples of the IFA model with standard *fgld*.

the performance of the criteria, in particular in the second scenario. However, some useful insights can be derived by this empirical study: when working with the complete likelihood, the WAIC tends to favour more complex models, compared to the BICM, which is more conservative, resembling the typical trade-off between AIC and BIC in the frequentist setting.

	k = 1	k = 2	k = 3
BICM	99	1	0
DIC	98	2	0
WAIC	43	36	21

	k = 1	k = 2	k = 3	k = 4
BICM	77	20	3	0
DIC	58	26	16	0
WAIC	0	4	57	39

Table 3.7: Frequency of selected model according to different information criteria. For the BIC the maximum likelihood estimate was used, while for the others are based on Ghosh-Dunson Bayesian factor model.

3.4 Illustration with European Social Survey data

In this section we apply the IFA model with standard *fgld* factors that we have developed to data taken from the European Social Survey (ESS) (Norwegian Social Science Data Services, 2020). The ESS is a recurring survey about attitudes and behaviour that started in 2001 and is carried out every 2 years in most

European countries, Round 10 has 30 participating countries. The probability of inclusion is computed with the goal of obtaining a sample that can be used for estimating quantities about the national population of interest. The questionnaire contains items that are present in each round (the core section) and so-called rotating modules, which are groups of questions that focus more in depth on a particular topic and are sometimes repeated in later rounds. Moreover, in each iteration of the survey, a 21-item measure of human values and some test questions for the validation of some items are present.

In Round 10 of the ESS, there is rotating module called *"Europeans' understandings and evaluations of democracy"*, which explores the attitudes of citizens regarding the importance they assign to some characteristics related to democracy; all of these questions start with *'How important you think it is for democracy in general that …'* and have an 11-point measurement scale from 0 (*'Not at all important for democracy in general'*) to 10 (*'Extremely important for democracy in general'*). Questions about the same characteristics are then asked in relation to the country of the respondent, with the incipit *'To what extent you think each of the following statements applies in [country]'*, again on 11-point scale that goes from 0 (*'Does not apply at all'*) to 10 (*'Applies completely'*).

These items have been used to form some variables with the idea of applying the IFA model we have developed. Given the high correlations among groups of items and to increase the degree of continuity of the variables, variables have been constructed that are the mean of some original variables. There is some arbitrariness in the groupings, although, within each group, all variables have the same measurement scale and relate to the same topic to an extent that the constructed variable can be easily described.

All items from this module have been used with the exception of `stpldmi`, which is an alternative to the item `chpldmi`, the items `gptpelc` and `keydec` which were not correlated to other items relating to the same topic. Finally, two items (`implvdm` and `accalaw`) were excluded because they have a different measurement scale to the other items and are not enough to construct a continuous summary variable.

On top of the questions related to democracy, we have also retained a group of 8 questions part of the core module on political views, these are all questions about trust in institutions starting with *'How much you personally trust each of the institutions …'*, again on 0-10 scale from *'No trust at all'* to *'Complete trust'*.

The list of items that have been used in the analysis is shown in Table 3.8, which shows the questions, and the relation between original and constructed variables. For the questions related to democracy there is a second set of variables (from corresponding questions) that refer to the country of the respondent, the variable names are the same but with a 'c' at the end and the same naming distinction has been used for the constructed variables. The constructed variables can be briefly described as follows:

- `trst_pol`: trust in political entities and institutions.
- `trst_sys`: trust in systemic institutions.
- `demo_gov(c)`: democracy characteristics related to government interventions.
- `demo_fun(c)`: democracy characteristics related to the electoral and democratic process.
- `demo_pop(c)`: democracy characteristics related to the participation and representation of the population.

After the selection and construction of the analysis variables, the sample has been restricted to respondents from Italy. Moreover, observations with missing values in at least one of the original variables considered were excluded; this resulted in a sample size of 1081. A graphical representation of the resulting data is presented in Figure 3.5. From the kernel density estimates of the variables we can see how some of them are very skewed, such as `demo_gov` and `demo_fun` with most respondents assigning very high importance to the related characteristics.

We have then fit the IFA model with the standard *fgld* with k, the number of factors, ranging from 1 to 4. The algorithm was run each time for 25,000 iterations, with the first half serving as burn-in. There are no signs of problems with the convergence; traceplots for the models with $k = 1$ and $k = 2$ are shown in Appendix B. The information criteria do not offer an unequivocal guidance as to which value of k to choose, both the DIC and the BICM increase with k, while the WAIC does the opposite. We consider the two simplest models with $k = 1$ and $k = 2$, with the latter offering the most interesting results in terms of factor loadings and factor scores.

Question	Original variable	Constructed variable
Trust in country's parliament	trstprl	trst_pol
Trust in politicians	trstplt	
Trust in political parties	trstprt	
Trust in the European Parliament	trstep	
Trust in the United Nations	trstun	
Trust in the police	trstplc	trst_sys
Trust in legal system	trstlgl	
Trust in scientists	trstsci	
The government protects all citizens against poverty	gvctzpv	demo_gov
The government takes measures to reduce differences in income levels	grdfinc	
The rights of minority groups are protected	rghmgpr	
The courts treat everyone the same	cttresa	
National elections are free and fair	fairelc	demo_fun
Different political parties offer clear alternatives to one another	dfprtal	
The media are free to criticise the government	medcrgv	
The views of ordinary people prevail over the views of the political elite	viepol	demo_pop
The will of the people cannot be stopped	wpestop	
Government changes policies in response to what most people think	chpldmi	
Citizens have the final say on political issues by voting directly in referendums	votedir	

Table 3.8: Variables from the European Social Survey Round 10 that have been included for the analysis. On the left column are the questions, on the central column are the variable names in the original data set, and on the right column are the variables that have used for the factor analysis, that are the mean of the group of variables to their left.

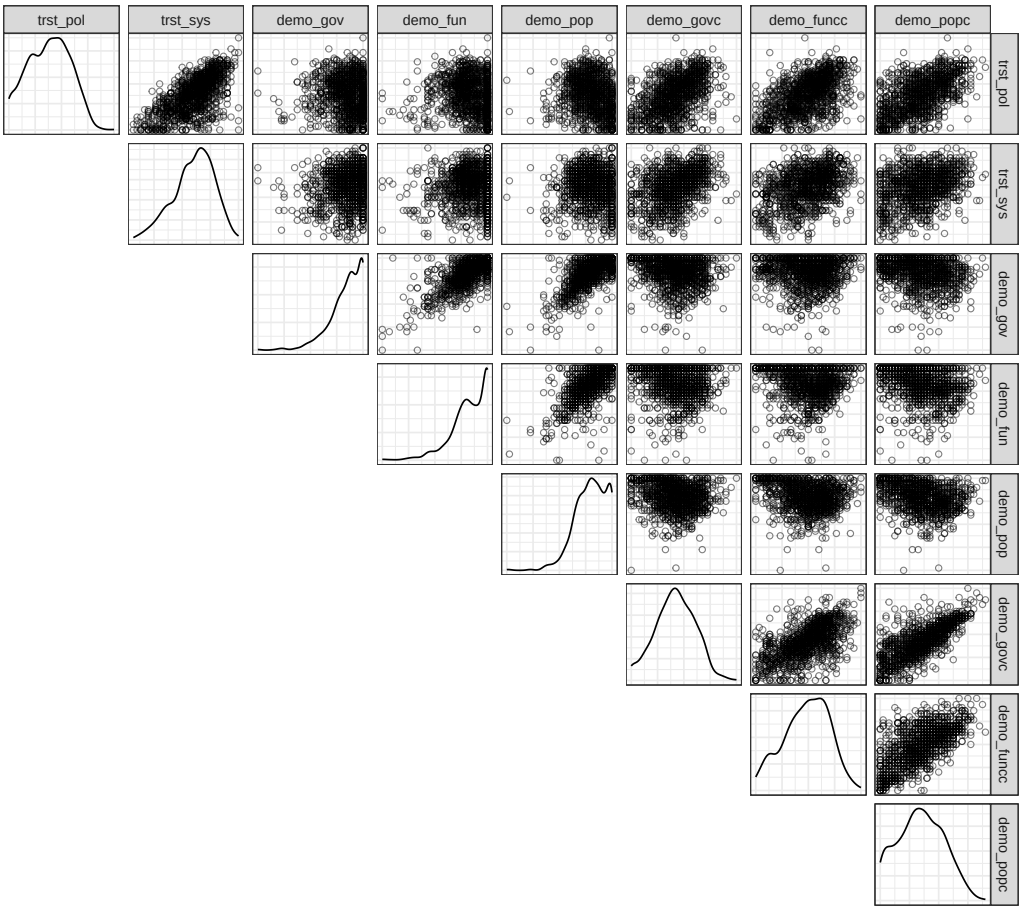

Figure 3.5: Scatter plot matrix showing all pairwise relationships between the 8 variables used for the factor analysis. On the diagonal are the kernel density estimates of each variable.

In the following we will compare results with two other models, the classical normally distributed factor analysis model fitted via maximum likelihood, and the IFA model with normal mixtures fitted again according to maximum likelihood with the EM algorithm (Montanari and Viroli, 2010b). For the latter we fit models with mixtures of three components for each factor ($G_j = 3, \quad j = 1, \ldots, k$), to ensure enough flexibility.

To make the results from the three models more easily comparable, we choose an arbitrary, but consistent, order and sign for the columns of $\mathbf{\Lambda}$: these are arranged in descending order with respect to the their squared sum, $\sum_{j=1}^{k} \lambda_{lj}^2$, and their sign is such that the sum of their elements is positive. These rules are already implemented in the built-in R function `factanal`, used for the maximum likelihood estimation of the FA model. This changes, applied to the fitted scores, do not affect any of the models as they are invariant to both permutations and sign changes to the latent factor variables.

The factor loadings from the models with $k = 2$ are presented graphically in Figures 3.6 and 3.7. The visualisation represents the loadings in the two-dimensional plane given by the two latent factors, in a similar way to a biplot, but without the factor scores. The plot allows for an immediate identification of which groups of variables are related the most to each factor. First of all, we can see how the loadings from the two IFA models (Figure 3.7) are very similar between them and also very similar to the ones given by the classical factor model with varimax rotation (Figure 3.6). This gives us some reassurance in robustness of the obtained solution to different model specifications. The two factors also turn out to be quite interpretable:

- The first one is related to the perceived characteristics of democracy in Italy and to the trust variables: respondents with high values for this factor will have generally high trust in politics and institutions and will also perceive that generally the features of democracy are present in Italy.

- The second factor is instead closely related to the ideal characteristics of democracy and tells us that in general people tend to give similar answers to the aspects related to the government (`demo_gov`), functioning of democracy (`demo_fun`) and popular representation (`demo_pop`).

Next we consider the factor scores, that is the fitted values for the latent variables $\mathbf{y}_i$. In the IFA with the standard *fgld* the factor scores are part of

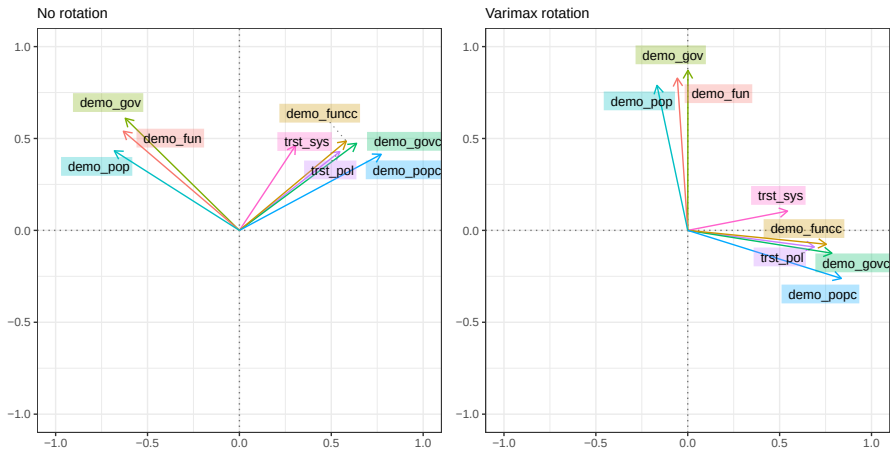

Figure 3.6: Factor loadings from the classical factor analysis with $k = 2$, with no rotation (left panel) and with varimax rotation (right panel).

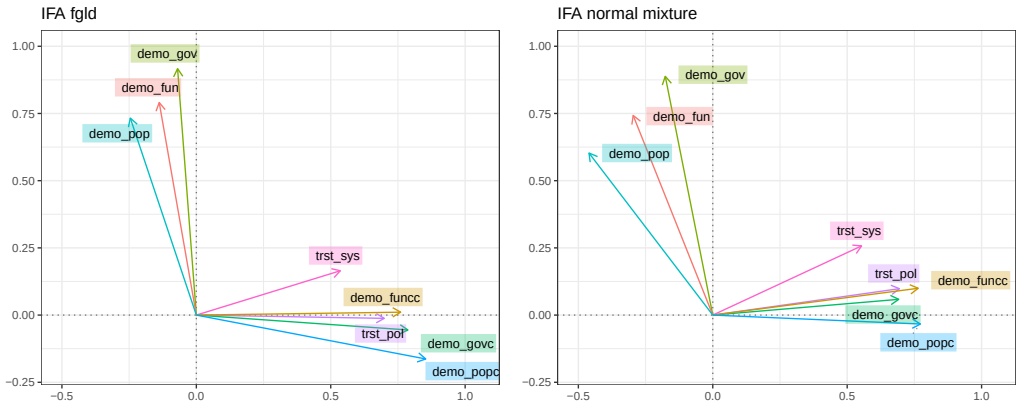

Figure 3.7: Factor loadings from the IFA with standard *fgld* (left panel) and from the IFA with normal mixtures (right panel).

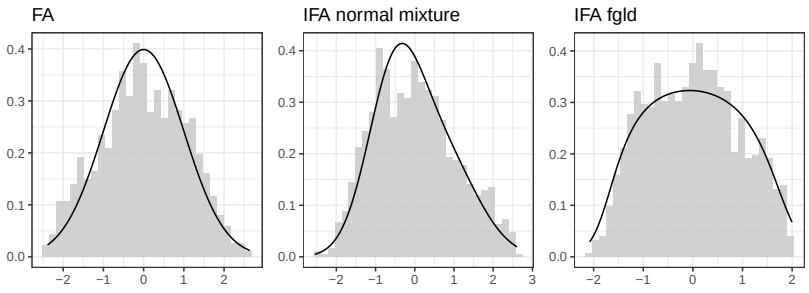

Figure 3.8: Factors scores with their fitted probability density estimate from the models with $k = 1$.

the model estimation and we can take their posterior mean as a summary for their value. For both the classical and the independent factor analysis models, the scores are not part of the estimation procedure and need to computed afterwards; for this we choose the Bartlett estimator:

$$\hat{y}_i = (\hat{\mathbf{\Lambda}}^\top \mathbf{\Psi}^{-1} \hat{\mathbf{\Lambda}})^{-1} \hat{\mathbf{\Lambda}}^\top \mathbf{\Psi}^{-1} \mathbf{x}_i \qquad i = 1, \ldots, n.$$

The distribution of the factor scores from the three models with $k = 1$ is shown in Figure 3.8. Overlaid on each histogram is the probability density that is predicted by the model: for the classical FA we have a standard normal distribution, for IFA with normal mixtures we have the 3-component Gaussian mixture with the estimated coefficients and for the IFA with standard *fgld* we plot the probability density at the posterior mean of the parameters. For this first factor, the scores from the IFA with normal mixtures is slightly skewed, while the solution with the *fgld* has a flatter shape that reflects the uniformity of the modelled scores around the center of the distribution. If look at the same picture from the models with $k = 2$, Figure 3.9, we see that the distribution of the scores for the second factor are skewed for all models. Of course, for the FA model the theoretical normal distribution does not match the fitted scores; for the IFA with normal mixture there is some correspondence between the histogram and the density function, but it is greater for the IFA with standard *fgld*, as it also has the advantage of deriving the scores in the modelling itself. In Table 3.9 we further compare the empirical distribution of factor scores for the 6 models we have just considered with computations for sample skewness and kurtosis. We highlight that in the second factor of the model with $k = 2$ the IFA with the standard *fgld* is able to achieve greater values in terms of

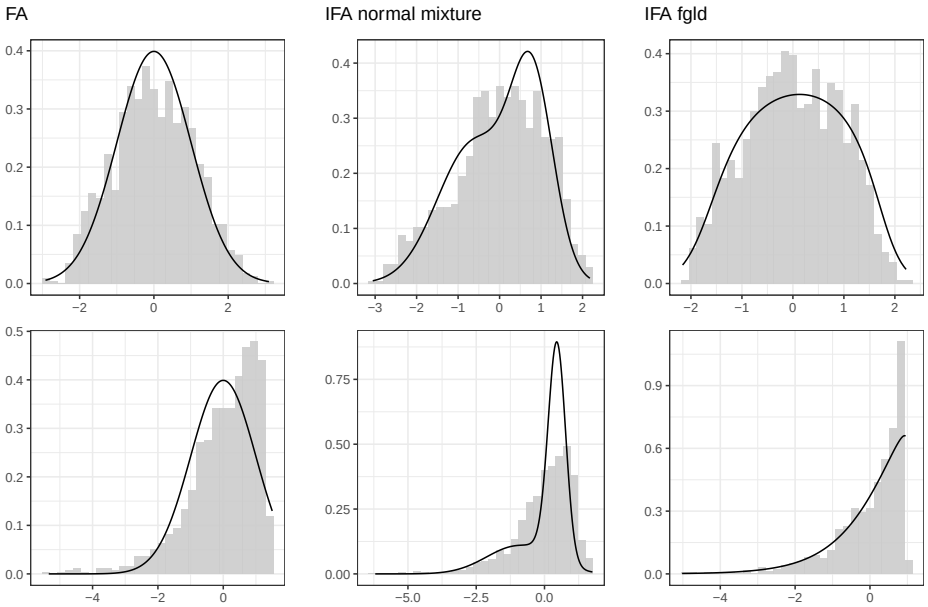

Figure 3.9: Factors scores with their fitted probability density estimate from the models with $k = 2$.

skewness and kurtosis with a fewer number of parameters than the IFA with the normal mixture. The number of effective parameters for the FA model is given in Equation 3.15. For the IFA with normal mixture this turns out to be $p(k+1) + (3\sum_{j=1}^{k} G_j - k)$[2], and for the IFA with the standard *fgld* is equal to $p(k+1) + 2k$. The latter model can thus, as in this case, achieve the flexibility of the normal mixture model with a smaller number of parameters.

[2]The formula refers to the estimation routine we have used, which does not constraint the expected value and variance of the latent variables. If that were the case the number of parameters would be equal to: $p(k+1) + 3(\sum_{j=1}^{k} G_j - k)$

	k = 1		
Method	# parameters	sample skewness	sample kurtosis
FA	16	-0.06	2.34
IFA normal mixture	24	0.30	2.41
IFA fgld	18	0.08	2.12

	k = 2				
Method	# parameters	sample skewness		sample kurtosis	
		1	2	1	2
FA	23	0.01	-1.32	2.50	5.48
IFA normal mixture	40	-0.40	-1.51	2.56	6.70
IFA fgld	28	-0.07	-1.81	2.19	7.30

Table 3.9: Comparison of the models with $k = 1$ and $k = 2$ in terms of number of parameters and shape of the factor scores.

Chapter 4

Multivariate Analysis and Classification with the Integrated Rank-Weighted Depth

4.1 Introduction

In multivariate analysis the identification of order statistics, quantiles and typical or atypical patterns is very challenging due to the lack of an order among observations, which is instead natural in the real line $\mathbb{R}^1$ (Kong and Mizera, 2012; Serfling, 2002). Since the early 1990s, considerable advancements have been made in developing more generalized statistics for assessing centrality and outlyingness of data in $\mathbb{R}^p$, with $p \geq 2$, and to identify central regions within a data cloud, comprising points with a specified degree of centrality. These advancements are grounded in the concept of statistical depth, which naturally arranges the sample points in a center-outward order. A depth function assigns a real number to each point of a multivariate dataset measuring the outlyingness of the point with respect to the barycenter. Thus, it provides a way to quantify how far an observation is from the center of the dataset, and is also used to identify order statistics via depth-induced contours.

Several popular depth functions have been introduced to measure the centrality of data points within a dataset. Notably, the halfspace depth (Tukey, 1975) finds the minimum probability of halfspaces containing the point, while the Mahalanobis depth (Liu and Singh, 1993) offers an alternative measure

based on the well-known Mahalanobis distance. In addition to these, various other depth functions, such as the simplicial, regression, and majority depth, have been proposed and applied across diverse domains, including classification, quality control in manufacturing, and exploratory statistical analysis. From a theoretical standpoint, Liu et al. (1999) and Serfling and Zuo (2000) provided the foundational general and constructive definition of depth functions, outlining crucial postulates like invariance, monotonicity, convexity, and continuity. Recently, Mosler and Mozharovskyi (2022) delved into the various notions of multivariate depth statistics, emphasizing both theoretical and practical aspects, such as invariance, uniqueness, robustness, and computational feasibility.

In this chapter we focus on a depth function called integrated rank-weighted (IRW) depth (Ramsay et al., 2019), which in turn is based on the integrated depth notion introduced by Cuevas and Fraiman (2009). For multivariate real data, integrated depths can be thought of as the expected value along infinite random directions of a univariate depth function computed on the projected data. We show that the IRW depth is affine invariant with sphered data, thus possessing all the properties of so-called statistical depth functions (Serfling and Zuo, 2000; Serfling, 2002); and we also show that the Mahalanobis depth is closely related to the concept of integrated depth function. In addition, we demonstrate that the IRW depth provides a complete characterization of the probability distribution of the data.

The strength of this depth definition lies in its generality, allowing for flexible model choices in the projected spaces. Among the models we consider for the univariate distributions we find the quantile-based *fgld* distribution to be a valuable option (Redivo et al., 2023; Chakrabarty and Sharma, 2021). We will also explore the depth regions and contours induced by the IRW depth function, highlighting its non-convex nature, which comes with both advantages and disadvantages. Nonetheless, this characteristic adds versatility and adaptability, making it particularly suitable for analyzing multivariate data with various shapes and distributions. Thanks to the flexibility of model choices and to its computational efficiency, the IRW depth function, serves as a promising tool for supervised classification tasks. In the same perspective of Ghosh and Chaudhuri (2005), we adopt maximum depth as a principle to measure the largest proximity to specific class distributions. The asymptotic optimality

of the classifier is demonstrated, under the same assumptions of the median classifier by Hall et al. (2009), which basically entail that the alternative populations may have arbitrary distributions and differ by locations shifts. The performance of the proposed method is evaluated through simulated experiments and real data application, and is shown to be very good, especially when compared to the same classification methods based on other depth notions.

The remainder of the chapter is structured as follows: in Section 4.2, we review the most important depth functions and we provide the definition of the IRW depth and its connections to other depth functions. We also derive some theoretical properties, starting with its population version and then introducing its sample counterpart with a demonstration of its asymptotic strong consistency as the sample size increases.

Moving on to Section 4.3, we demonstrate the practical application of the IRW depth function in supervised classification. We establish the asymptotic optimality of its maximum depth classifier, ensuring accurate classification with diminishing misclassification rates as the sample size, dimensionality, and the number of random projections considered tend towards infinity. The performance of the classifier is evaluated through empirical experiments in Section 4.4, encompassing both simulated and real data scenarios. The proofs of the theoretical results are found in Appendix C.

Furthermore, to facilitate the implementation of our approach, we provide a software package called `dqclass` in R, which is available in the reproducibility materials. This open-source package enables users to easily compute the IRW depth and the resulting classifiers in their own research or analyses.

4.2 Integrated rank-weighted depth

4.2.1 Depth functions

Let $\mathbf{X}$ be a multivariate random variable of dimension p with probability distribution F. A depth function measures how deep or central a point $\mathbf{x} \in \mathbb{R}^p$ is with respect to a data cloud of points sampled from $\mathbf{X}$ or with respect to the theoretical distribution of $\mathbf{X}$ itself. It can be formalized as a function $D(\mathbf{x}, F) : \mathbb{R}^p \times \mathcal{F} \to \mathbb{R}$, where usually the codomain is restricted to the interval $[0, 1]$ so that the most central point(s) have a depth equal to 1. In the following

theoretical definitions of some of the most popular statistical depths are given:

- Halfspace or Tukey's depth (Tukey, 1975; Donoho and Gasko, 1992).
  For a multivariate random vector $\mathbf{X}$ with distribution F the halfspace depth of a point $\mathbf{x} \in \mathbb{R}^p$ is given by the minimum probability of a halfspace that contains that same point:

$$\mathrm{HD}(\mathbf{x}, F) = \inf\{P(H) : H \text{ is a closed halfspace}, \mathbf{x} \in H\}.$$

- Mahalanobis depth (Mahalanobis, 1936).
  It is inversely proportional to the Mahalanobis distance:

$$\mathrm{MD}(\mathbf{x}, F) = \left[1 + (\mathbf{x} - \boldsymbol{\mu}_F)\boldsymbol{\Sigma}_F^{-1}(\mathbf{x} - \boldsymbol{\mu}_F)\right]^{-1},$$

  where $\boldsymbol{\mu}_F$ and $\boldsymbol{\Sigma}_F$ are the mean vector and dispersion matrix of $\mathbf{X}$, which, if needed, can be estimated from sample data, giving rise to the sample version of the depth function.

- Simplicial depth (Liu, 1990; Serfling and Zuo, 2000).
  It is defined as a probability that the point $\mathbf{x} \in \mathbb{R}^p$ at which we want to compute the depth, belongs to a random simplex in $\mathbb{R}^p$. The latter can be defined as the convex hull of a set of $p + 1$ random points, the definition is thus:

$$\mathrm{MS}(\mathbf{x}, F) = P(\mathbf{x} \in \mathrm{conv}(\{\mathbf{X}_1, \ldots, \mathbf{X}_{p+1}\}),$$

  where $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_p$ are independent copies of $\mathbf{X}$ and $\mathrm{conv}(\{\cdot\})$ indicates the convex hull.

- Projection depth (Liu, 1992; Serfling and Zuo, 2000).
  The projection depth value of a given point $\mathbf{x} \in \mathbb{R}^p$ with respect to the distribution F can be defined as:

$$\mathrm{PD}(\mathbf{x}, F) = \frac{1}{1 + \mathrm{O}(\mathbf{x}, F)},$$

  where $\mathrm{O}(\mathbf{x}, F)$ is an outlyingness measure:

$$\mathrm{O}(\mathbf{x}, F) = \sup_{\mathbf{s} \in \mathcal{S}^{p-1}} \left|Q(\mathbf{s}, \mathbf{x}, F)\right|,$$

  where $\mathcal{S}^{p-1}$ is the unit sphere in $\mathbb{R}^p$:

$$\mathcal{S}^{p-1} = \{\mathbf{s} \in \mathbb{R}^p : \mathbf{s}^\top \mathbf{s} = 1\}.$$

The function $Q(\mathbf{s}, \mathbf{x}, F) = \frac{\mathbf{s}^\top \mathbf{x} - \mu(F_\mathbf{s})}{\sigma(F_\mathbf{s})}$ represents the normalized projection of $\mathbf{x}$ onto the unit vector $\mathbf{s}$. $F_\mathbf{s}$ is the distribution of $\mathbf{s}^T\mathbf{X}$. The projection depth and its associated estimators depend on the choice of $\mu(F_\mathbf{s})$ and $\sigma(F_\mathbf{s})$. A commonly used robust choice is given by the median and median absolute deviation, respectively. The depth is thus defined as the worst case outlyingness of $\mathbf{x}$ in any one-dimensional projection.

For a comprehensive listing of other important depth functions see Mosler and Mozharovskyi (2022).

In their seminal work, Serfling and Zuo (2000) introduced a comprehensive framework for depth functions and outlined four essential properties that such functions should possess. Specifically, they denote as a statistical depth function a non-negative and bounded function, satisfying the following key properties:

(i) Affine invariance: The depth function remains invariant under changes in the coordinate system or scale of the underlying measurements.

(ii) Maximality at the center: The deepest central point within the dataset attains the highest depth value.

(iii) Monotonicity: As a point moves away from the deepest central point, the depth function monotonically decreases.

(iv) Asymptotic behavior: The depth function approaches zero as a point moves towards infinity.

The halfspace depth meets all these desirable properties. In contrast, the Mahalanobis depth function qualifies as a proper depth function only when the underlying distribution F is symmetric, with affine equivariant first and second moments (Serfling and Zuo, 2000).

Another crucial aspect of a depth function lies in its capacity to effectively identify and characterize the underlying distribution F based on the depth scores for all possible values of $\mathbf{X}$. Various depth functions exhibit varying degrees of power in this task, making the selection of an appropriate depth function critical for accurately characterizing and understanding the data distribution. For example, while the Mahalanobis depth can only identify the first two moments of F, while the so-called zonoid depth (Koshevoy and Mosler,

1997) can fully determine F. Special attention has been devoted to the characterization property of the Tukey depth function (Struyf and Rousseeuw, 1999; Koshevoy, 2002; Nagy, 2021; Kong and Zuo, 2010): the halfspace depth can identify a finite discrete distribution uniquely, but it may not do so for infinite discrete or continuous distributions.

In addition to these features, a depth function evaluated for a sample should be consistent to its population counterpart, as the sample size increases, and should be computationally efficient, *i.e.*, it should be possible to compute the depth values of data points efficiently even for large p.

Given that our goal in this manuscript is that using depth functions as a tool for classification, we want to work with a depth that is both flexible, thus able to accommodate a wide range of shapes for the class distributions, and at the same time computationally efficient and feasible for a wide range of data sizes, in particular as the number of variables increases. For instance, the Mahalanobis depth is very easy and cheap to compute, but not very flexible, being limited to represent elliptical contours, similarly to assuming a Gaussian density. Depth notions based on geometrical notions instead, such as simplicial and halfspace depths, are very flexible but quite expensive to compute especially as the number of dimensions increases. Our choice has been that of focusing on the class of integrated depth functions, which will be introduced in the next section and will be shown throughout to possess both of these qualities. They are still closely related to the more common depths that we have just listed, but deal with the multivariate nature of problem with the popular and fruitful approach of resorting to projections.

4.2.2 Integrated depth functions

The notion of integrated depth functions has been introduced by Cuevas and Fraiman (2009) with the stated goal of being able to deal with infinite dimensional data, such as that encountered in functional analysis. Similarly to the projection depth introduced in the previous section it is based on infinite random univariate projections. Given a univariate depth function denoted as D_1, and a probability measure Q, the general definition of the integrated dual depth (IDD) is:

$$D_{ID}(\mathbf{x}, F) = \int D_1(f(\mathbf{x}), F_{f(\mathbf{x})})\, dQ(f),$$

where $\mathbf{x}$ can belong to a Banach space and f is a function belonging to its dual space. In the present work, we only focus on real-valued data, so that $\mathbf{x} \in \mathbb{R}^p$ and we fix Q to be the uniform distribution on the unit sphere, the so-called Haar measure, then the definition becomes:

$$D_{ID}(\mathbf{x}, F) = \int_{\mathcal{S}^{p-1}} D_1(\mathbf{s}^\top \mathbf{x}, F_{\mathbf{s}^\top \mathbf{X}}) d\mathbf{s} = \mathbb{E}_{\mathbf{S}} \left[D_1(\mathbf{s}^\top \mathbf{x}, F_{\mathbf{s}^\top \mathbf{X}}) \right].$$

From this expression we see that the IDD can be seen as the expected value along the infinite random directions belonging to unit sphere ($\mathbf{s} \in \mathcal{S}^{p-1}$) of a univariate depth evaluated at the projected target point according to the projected probability distribution.

In Cuevas and Fraiman (2009), the authors focus only on the IDD defined on the univariate version of the simplicial depth: a simplex in one dimension is a closed segment and the probability that a point x belongs to a random segment $[X_1, X_2]$, with $X_1, X_2 \overset{\text{iid}}{\sim} F$ is given by:

$$SD_1(x, F) = P(x \in [X_1, X_2]) = P(X_1 \leq x \leq X_2) = F(x)\,(1 - F(x-)),$$

where $F(x-)$ is a shorthand notation for $\lim_{x \to x-} F(x)$. Another integrated depth function is introduced in Ramsay et al. (2019), where the authors choose as starting point the univariate halfspace depth. The univariate equivalent of a halfspace is a ray, and there are only two rays at the point x whose probability we need to consider, resulting in the following definition:

$$HD_1(x, F) = \min\{F(x), 1 - F(x-)\}. \tag{4.1}$$

The integrated depth derived from HD_1 is called integrated rank-weighted (IRW) depth, and given that it is the depth that we will focus for the remainder of this work we denote it simply as D:

$$D(\mathbf{x}, F) = 2 \int_{\mathcal{S}^{p-1}} HD_1(\mathbf{s}^\top \mathbf{x}, F_{\mathbf{s}^\top \mathbf{X}}) d\mathbf{s},$$

where the factor 2 steps in so that the maximum of the univariate depth is equal to1. In Ramsay et al. (2019), the authors also add at the denominator a term for the volume of the p-dimensional sphere, so that also the integrated depth has a maximum of 1; however, in the present work we choose, without loss of generality, to avoid the term, which is only a multiplication constant

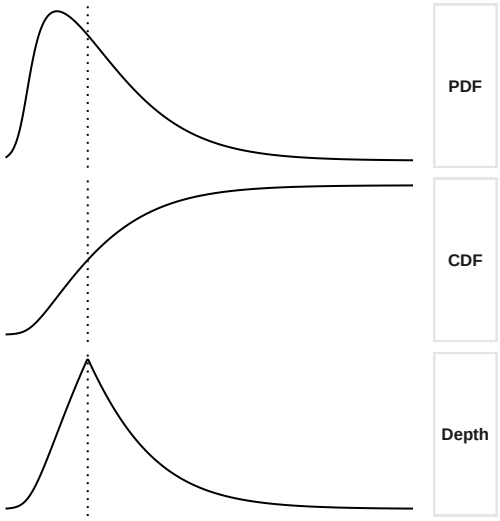

Figure 4.1: Probability density function (PDF), cumulative distribution function (CDF) and univariate halfspace depth function of a generic skew random variable. The dashed line denotes the median.

not needed in the actual computation, thus simplifying the notation. The factor 2 instead, simplifies if we express $\min\{x, y\} = \frac{x+y-|x-y|}{2}$, resulting in this equivalent expression:

$$D(\mathbf{x}, F) = \int_{\mathcal{S}^{p-1}} (1 - F(x) + F(x-) - |1 - F(x) - F(x-)|)\, d\mathbf{s},$$

where to simplify notation $x \equiv \mathbf{s}^\top \mathbf{x}$ and $F \equiv F_{s^\top \mathbf{x}}$. Furthermore, we assume from here on that the cumulative distribution of the projection is continuous along each direction, which results in the following working definition for the IRW depth:

$$D(\mathbf{x}, F) = \mathbb{E}_{\mathbf{S}} \left[1 - 2|F_{\mathbf{S}^\top \mathbf{x}}(\mathbf{S}^\top \mathbf{x}) - 0.5|\right]. \tag{4.2}$$

The univariate depth underlying this definition, $D_1(x, F) = 1 - 2|F(x) - 0.5|$, is represented in Figure 4.1 along with the density (PDF) and the cumulative distribution functions (CDF). It can be seen as a simple transformation of the CDF made in such a way that it reaches its maximum value at the median and it decreases linearly as a function of the CDF as the probability moves away from 0.5, reaching 0 at the extremes of the domain of the random variable.

The Mahalanobis distance, on which the depth by the same name is based, is connected to the idea of integrating the results from random uniform directions as the following Lemma shows.

Lemma 5 *Let* $\mathbf{X}$ *be a multivariate random variable of dimension* p *with center* $\boldsymbol{\mu}$ *and finite precision matrix* $\boldsymbol{\Sigma}^{-1} = \mathbf{W}^\top \mathbf{W}$*, and let* $\tilde{\mathbf{X}} = \mathbf{W}\mathbf{X}$ *be a sphering transformation having center at* $\tilde{\boldsymbol{\mu}} = \mathbf{W}\boldsymbol{\mu}$ *and identity covariance matrix. Let* $\mathbf{x}$ *be a generic point of* $\mathbb{R}^p$*, and* $\tilde{\mathbf{x}}$ *its counterpart after sphering. Then, the expectation of all the Euclidean distances of the projected point* $\mathbf{s}^\top \tilde{\mathbf{x}}$ *to* $\mathbf{s}^\top \tilde{\boldsymbol{\mu}}$ *over the uniformly distributed directions* $\mathbf{s} \in \mathcal{S}^{p-1}$*, coincides with the Mahalanobis distance between* $\mathbf{x}$ *to* $\boldsymbol{\mu}$*, divided by* p*.*

The proof can be found in the Supplementary Material. This is mostly a theoretical result as using this result for computing the Mahalanobis distance would still require the knowledge or computation of the covariance matrix in order to derive the sphering matrix. In the following Corollary the Mahalanobis depth is restated based on the expression for the distance we have just found:

Corollary 1 *The Mahalanobis depth can be also evaluated based on the expected value along infinite random projections as:*

$$MD(\mathbf{x}) = \left[1 + p\,\mathbb{E}_{\mathbf{S}}\left[\left(\mathbf{S}^\top \tilde{\mathbf{x}} - \mathbf{S}^\top \tilde{\boldsymbol{\mu}}\right)^2\right]\right]^{-1},$$

Next, we consider the four properties introduced in Serfling and Zuo (2000), that define a so-called statistical depth function, in relation to the IRW depth. In the paper where the depth has been first introduced (Ramsay et al., 2019), three of these are proven and it is noted that the depth does not possess affine invariance. In the following theorem we show that also the latter is achieved with sphered data.

Theorem 3 *Given* $\mathbf{x} \in \mathbf{X}$ *with finite precision matrix* $\boldsymbol{\Sigma}^{-1} = \mathbf{W}^\top \mathbf{W}$ *and let* $\tilde{\mathbf{X}} = \mathbf{W}\mathbf{X}$ *be the sphering transformation with identity covariance matrix. The depth function on sphered data*

$$D(\mathbf{x}, F) = \mathbb{E}_{\mathbf{S}}\left[1 - 2|F_{\mathbf{S}^\top \tilde{\mathbf{X}}}(\mathbf{S}^\top \tilde{\mathbf{x}}) - 0.5|\right], \tag{4.3}$$

is bounded, non-negative and satisfies the following properties:

(i) The depth is affine invariant, that is $D(\mathbf{A}\mathbf{x} + \mathbf{b}, F_{\mathbf{A}\mathbf{X}+\mathbf{b}}) = D(\mathbf{x}, F_{\mathbf{X}})$ *for any random vector* $\mathbf{X}$ *in* $\mathbb{R}^p$*, any non-singular* $p \times p$ *matrix* $\mathbf{A}$ *and any* p*-vector* $\mathbf{b}$*;*

(ii) $D(\boldsymbol{\mu}, F) = \sup_{\mathbf{x}\in\mathbb{R}^p} D(\mathbf{x}, F)$ *holds for any* $F \in \mathcal{F}$ *having center at* $\boldsymbol{\mu}$*;*

(iii) for any $F \in \mathcal{F}$ *having center at* $\boldsymbol{\mu}$*,* $D(\mathbf{x}, F) \leq D(\mathbf{x}', F)$ *holds with* $\mathbf{x}' = \mu + \alpha(\mathbf{x} - \mu)$ *and* $\alpha \in [0, 1]$*;*

(iv) $D(\mathbf{x}, F) \to 0$ *as* $\|\mathbf{x}\| \to \infty$*.*

The proof of point (i) is provided in the Supplementary Material.

In the following theorem, we demonstrate that the IRW distribution depth function is unique with respect to F, and as a result, it uniquely characterizes the random variable.

Theorem 4 *Let* $\mathbf{X}$ *be a continuous random variable. The integrated rank-weighted depth function defined in (4.2) completely characterizes the underlying distribution of* $\mathbf{X}$*.*

The proof is given in the Supplementary Material.

4.2.3 Sample version and computation

In Ramsay et al. (2019) the authors provide an exact formula for the computation of the IRW depth given some data points: it is based on a weighted sum of ranks and from it the depth function takes its name.
The intuition for the formula can be built in the bidimensional case: the relative ranking between the projection of a target point and that of any point in the data only changes when projection directions cross the bisector line between the two points. With directions parallel to the bisector, the two projected points coincide, and thus the bisector divides the space in two sections where the relative ranking between the two points is constant. This reasoning can be repeated by considering the direction parallel to the bisector between the target point and any point in the data: all of these directions will create circular sectors where the ranking of the projected points is constant. Then, the empirical distribution function is substituted for the CDF in the most general formula for the univariate halfspace depth HD_1 (Equation 4.1): its value only depends on the rankings and will also be constant within each sector. The depth can finally be computed as a weighted average of HD_1 within each sector, with the weights given by the angles of the sectors. This formula can

also be extended to higher dimensions and in general to any depth based on the projection of univariate CDFs. For a general dimension p and sample size n however, the complexity needed for computing the depth is dominated by a term of the order $O(n^{p-1})$, which makes the computation expensive in higher dimensions.

Another way of computing the depth based on a sample, and the one we employ, is approximating the expected value of Equation 4.2 with the sample mean, randomly sampling directions on the unit sphere. This computation method, which is a Monte Carlo approximation of the integral, has been proposed in Cuevas and Fraiman (2009) and it has the great advantage that its computation complexity is only linear in the dimensionality of the space: given B random directions on which the computation is based the order is $O(Bnp)$, as mentioned in (Ramsay et al., 2019).

Let $\mathbf{X}_n$ be a sample of size n from $\mathbf{X}$. Then the sample version of the IRW depth for a generic point $\mathbf{x}$, given B randomly sampled spherical directions (whose generic element is denoted as $\mathbf{s}_b$) is given by:

$$D_B(\mathbf{x}_i, \hat{F}_n) = \frac{\sum_{b=1}^{B} \left[1 - 2|\hat{F}_{\mathbf{s}_b^\top \mathbf{X}_n}(\mathbf{s}_b^\top \mathbf{x}) - 0.5|\right]}{B}. \tag{4.4}$$

An illustration of the computation of the IWR depth from a bivariate sample is given in Figure 4.2. Few equally spaced directions, for illustrative purposes, are shown in the left panel, the red target point in the data cloud is where each of the univariate depth functions on the right panel is evaluated. The mean of the values will gives us the approximation to the IRW depth.

Two choices need to made when working with the sample version of this depth: the estimator for the univariate CDF along each direction and the number of random directions.

In Cuevas and Fraiman (2009) and Ramsay et al. (2019) the authors only consider the substitution of the unknown population CDF for its empirical counterpart. However, we feel that the flexibility that the depth allows in choosing how to model the distribution function is a strength of the definition since it allows the method to handle a wide range of data types, making it suitable for diverse applications. In the following examples and applications we consider three models: the Gaussian distribution; the flattened generalized logistic distribution (*fgld*) (Chakrabarty and Sharma, 2021), which is a flexible

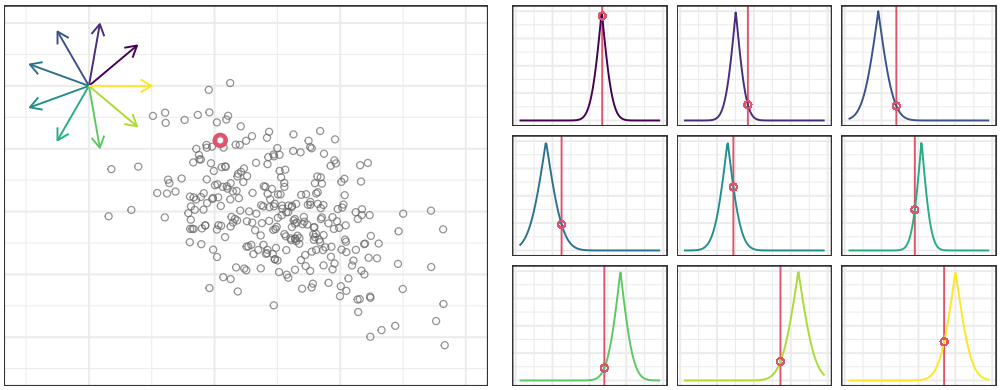

Figure 4.2: Illustration for the approximation of the IRW depth: on the left panel there is a data cloud, nine color-coded directions and in red we have the target point; on the right panel we have the univariate depth functions along the nine directions, evaluated at the point of interest.

quantile-based distribution with four parameters, allowing for skewness and flatness in the shape of the density and finally we also consider the kernel density estimation (KDE) as a nonparametric approach to distribution fitting. The *fgld* is estimated via least squares as in Redivo et al. (2023). For the KDE we use a normal kernel and the default bandwidth selection method employed in the `density` function in R. For estimating the CDF, the KDE method is translated to:

$$\hat{F}(x) = \frac{1}{n} \sum_{i=1}^{n} \Phi \left(\frac{x - x_i}{s} \right),$$

where Φ is the Gaussian CDF and s is the bandwidth. The other choice, the number of directions B, is strictly related to computational efficiency. On the one hand, the depth computation is easily scalable for multivariate data as it tackles the issue of dimensionality by projecting multivariate data into univariate data, effectively overcoming any limitations imposed by the dimensionality of variables. On the other hand, the computational efficiency is influenced by the choice of the number of projections denoted by B. While a large value of B is necessary to ensure asymptotic results, like its consistency to the theoretical definition of the depth (see Theorem 5), it can slow down the estimation process, especially when the sample size is also large. Therefore, careful consideration is needed to strike a balance between accuracy and computational

time.

In the following theorem we consider the strong consistency of $D_B(\mathbf{x}, \hat{F}_n)$ as an estimator of the theoretical population depth. A similar result is stated in Ramsay et al. (2019), for which we give in this work detailed proof in the Supplementary Material.

Theorem 5 *As $n \to \infty$ and $B \to \infty$ the sample depth converges almost surely to the population depth function:*

$$D_B(\mathbf{x}, \hat{F}_n) \xrightarrow{a.s.} D(\mathbf{x}, F) \tag{4.5}$$

Empirical consistency

To check how well the approximation works for a finite sample and a finite number of random directions, we carried out a small empirical simulation, whose results are shown in Figure 4.3. To look at the convergence of the computed value of the depth at a certain target point, we can see how this value changes as we add more and more random directions. To also account for the variability in the procedure, we can get multiple such sequences by using different sets of random directions. In each panel of the figure we show 100 such *random paths*, which are cumulative means up to a certain number of directions B.

The data is simulated from a multivariate normal distribution with center in the origin. A covariance matrix of dimension $p = 50$ has been sampled from a Wishart distribution, and the covariance matrices for the other dimensions ($p = 10$ and $p = 2$) have been taken as the top-left submatrices of it. Within each dimensionality p, the samples of different sizes ($n = \{50, 100, 250\}$) form nested sets of observations, that is as n increases new observations are added, so that results are more comparable, the difference being the new information coming from the additional units. Also, the target point is fixed for each dimension p, and it is found as a point having a fixed Mahalanobis distance (the quantile of level 0.68 of the chi-squared distribution $\chi^2_{\text{d.f.}=p}$ times p), so that that the depth can have similar values across the dimensions.

Figure 4.3 shows that, at least for this simple Gaussian data, the value of the computed depth is quite stable even for low values of n and does not need an appreciably larger number of directions when p is large. The convergence as the number of directions increases seems quite fast in all settings and for all

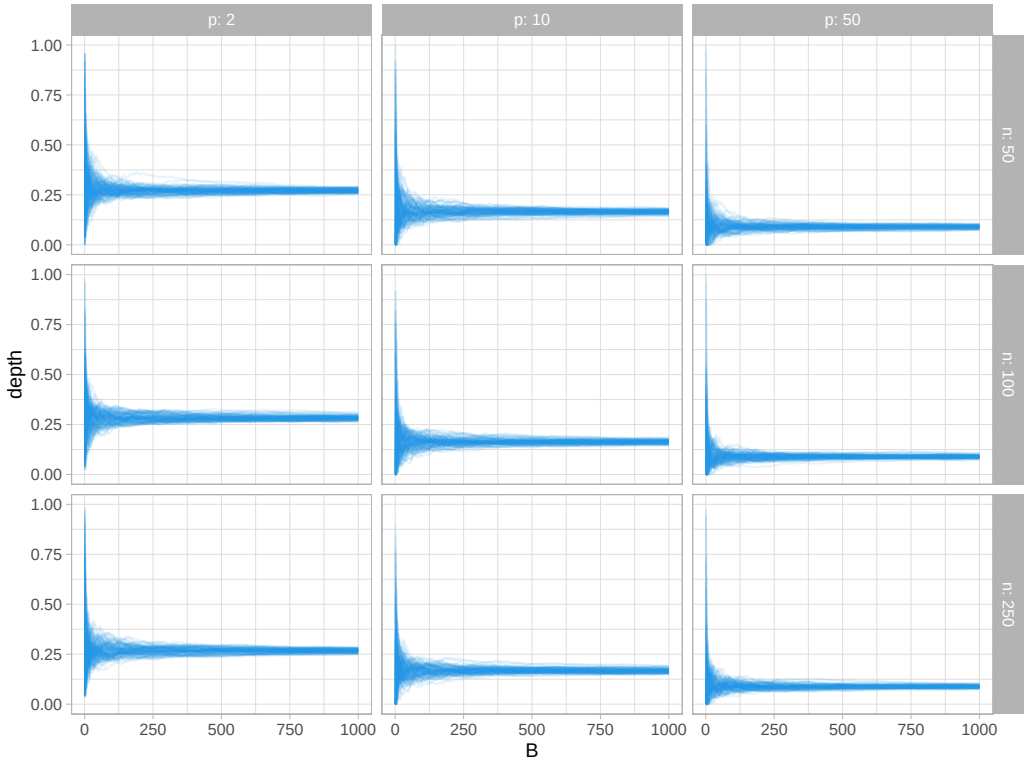

Figure 4.3: Each panel represents 100 random paths for the sample version of the IRW depth approximated with an increasing number of random directions (B in the x-axis). The columns show different numbers of variables (p) and the rows different sample sizes (n).

of them an empirical convergence seems to have been reached with $B = 500$, which is the value that we will also use in Section 4.3.

Remark 1 *Direction importance.*

One might wonder whether it is worthwhile to consider the fact that different directions could contribute differently to determining the depth of a point. The issue can be framed as asking whether it makes sense to weigh the directions by transforming the definition in Equation 4.4 into a weighted average using coefficients w_b, to be determined according to some criterion. Intuitively, a direction is informative when it is able to concentrate the units more around the barycenter, this is mainly in the perspective of classification (see Section 4.3), where we want to separate multiple groups. However, if data are sphered, the variability along each direction is constant, since the marginal distributions along each direction have unit variance. More importantly, for each direction, the sum of the depths of all points is also constant. In particular, given $d_{ib} = 1 - 2|\hat{F}_{\mathbf{s}_b^\top \mathbf{X}_n}(\mathbf{s}_b^\top x_i) - 0.5|$, we have

$$\sum_{i=1}^{n} d_{ib} \cong \frac{n}{2}, \tag{4.6}$$

for every b, and this sum converges to $\frac{n}{2}$ as the sample size increases. Therefore, surprisingly, for the purpose of determining the depth, the directions all have the same importance. In order to prove Equation 4.6, without loss of generality, consider an even n for which the median is at position $n/2$. Then result is exact if $\hat{F}_n$ is the empirical distribution function, since in this case:

$$\sum_{i=1}^{n} d_{ib} = n - 2\sum_{i=1}^{n}\left(\frac{i}{n}\mathbb{1}_{[i>n/2]} - \frac{i}{n}\mathbb{1}_{[i<n/2]}\right).$$

The approximation comes from the fact that $\hat{F}_n(x) \xrightarrow{a.s.} F(x) \quad \forall x$.

Remark 2 *Mahalanobis distance preservation.*

Lemma 5 is very important from an empirical point of view, since it guarantees that the expected value of the distances between univariate projections on sphered data is proportional to the Mahalanobis distance on the original multivariate data. This ensures coherence between the Mahalanobis depth and the

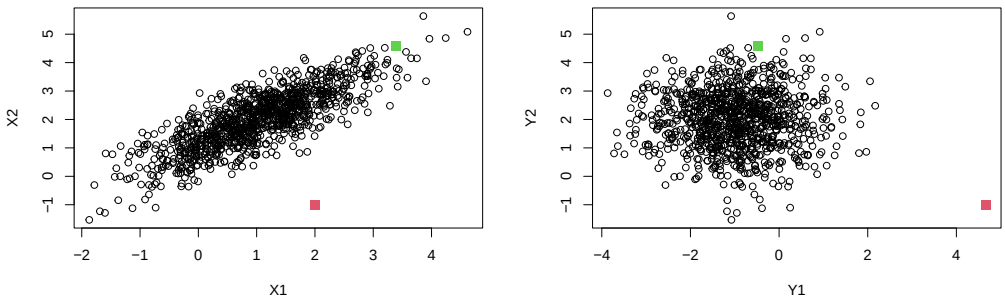

Figure 4.4: First panel: data drawn from a multivariate Gaussian. Second panel: sphered data.

expected value across the projections of Euclidean distances on sphered data. Take, for example, the point cloud represented in the first panel of Figure 4.4. The red point with original coordinates {2,-1} is clearly further away from the data cloud than the green point with coordinates {3.5,4.5}. Their Mahalanobis distances are 6.4 and 2.6, respectively. Table 4.1 shows the depths obtained on sphered and non-sphered data.

	$\mathbf{x}_a = \{2, -1\}$	$\mathbf{x}_b = \{3.5, 4.5\}$
without sphering	0.193	0.085
with sphering	0.082	0.206

Table 4.1: Computed depths for the two points on raw data and on sphered data.

From this example, it is clear that on raw data the green point $\mathbf{x}_a$ is projected far from the barycenter of the data in a limited number of directions, while $\mathbf{x}_a$ is projected far from the barycenter for most directions. But sphering makes angles and distances constant along the different projections, and on sphered data the depth indicated that $\mathbf{x}_b$ is deeper than $\mathbf{x}_a$. This is also evident from the second panel of Figure 4.4.

Remark 3 *Prediction.*

The IRW depth has a theoretical definition based on a population probability distribution F, moreover, its sample version can be based on a model (parametric or non-parametric) for $F_{\mathbf{s}^\top \mathbf{X}_n}$. The fact that the depth defines a model

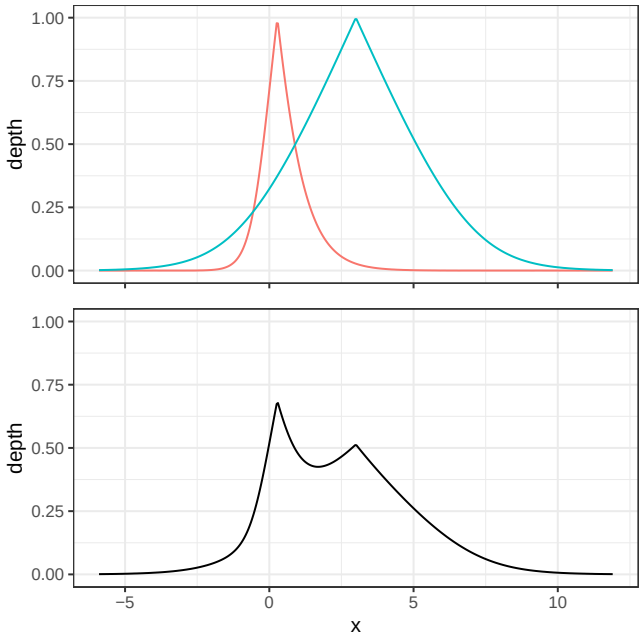

Figure 4.5: Two univariate halfspace depth functions on the top panel and their mean in the bottom panel.

for the data is a great advantage in terms of prediction. Unlike other depth functions that are defined only for empirical distributions, it is possible to estimate the depth for new out-of-sample values once the distributions along each direction have been estimated. The ability to provide predictions makes this depth function a good candidate tool for supervised classification purposes, as it will be shown in the next sections.

4.2.4 Depth regions and contours

The set $DR(p, F) = \{\mathbf{x} : D(\mathbf{x}, F) \geq p\}$ is the $p-$th depth region for $0 \leq p \leq 1$. The corresponding contour is defined as $DC(\mathbf{x}, F) = \{\mathbf{x} : D(\mathbf{x}, F) = p\}$.

Theorem 6 *The depth regions associated to the IRW depth function are affine equivariant (on sphered data), nested but not necessarily convex.*

A formal proof is given in the Supplementary Material. Intuition for the proof arises from the fact that the quasi-concavity of the depth function is a necessary and sufficient condition for the convexity of depth regions (Zuo, 2003). The

IRW depth function is quasi-concave along each spherical direction, as shown in the top panel of Figure 4.5. However, the sum of quasi-concave functions is not necessarily quasi-concave, and an example is given in the bottom panel of Figure 4.5, where two univariate depths are summed. Therefore, the expected value, which is based on summing comes from summing infinite such univariate depth functions is not necessarily quasi-concave.

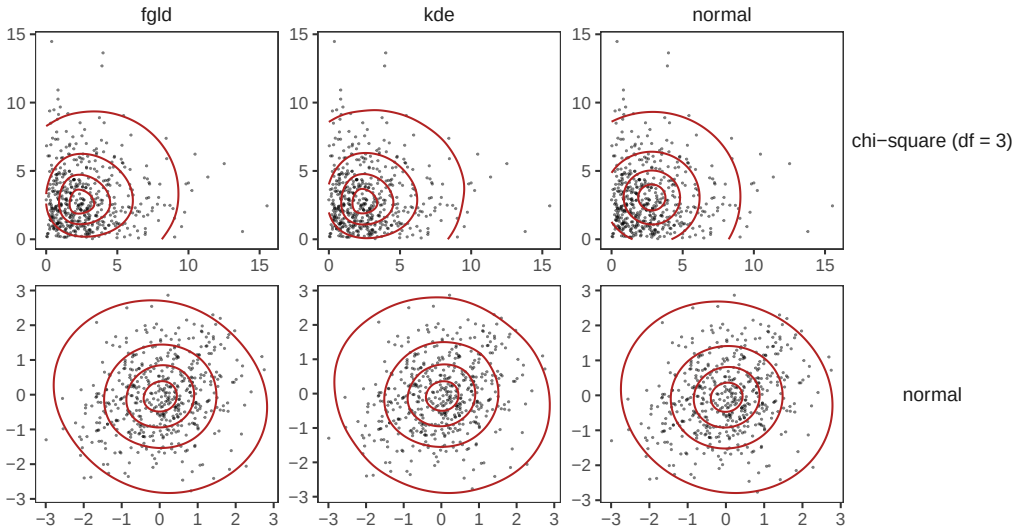

Figure 4.6: IRW depth contours based on three different CDF estimators: normal, *fgld* and KDE (on the columns), for data drawn from a standard Gaussian and a chi-squared distribution with 3 degrees of freedom (on the rows).

Convexity is an interesting characteristic for a depth function: it assures that, for any two points in the space, if their depths are above a certain level, then any point lying on the line segment connecting these two points will also have a depth above that level. Advantages of having convex depth regions include their clear geometric interpretation, which aids in visualizing and summarizing the dataset's structure, as well as their robustness against outliers and contamination in the data. On the other hand, non-convex depth regions offer particular advantages in scenarios that require accurate representation of complex and intricate data structures, such as datasets with intricate patterns or clusters. They are also beneficial when capturing specific characteristics or outliers that may be overlooked by convex regions. Additionally, non-convex depth regions enhance discrimination in classification tasks by effectively dis-

tinguishing between different classes or groups, accounting for within-class variability and capturing specific boundaries.

Figure 4.6 shows the depth contours obtained by applying the IRW depth function with different estimators for the cumulative distribution functions: a Gaussian distribution, the quantile-based *fgld*, and the KDE method. Data are generated from a standard Gaussian and from a chi-squared distribution with 3 degrees of freedom.

4.3 Supervised Classification

Thanks to previous theorems establishing the consistency of the sample estimator and highlighting its unique characterization property, the IRW depth can be employed to determine whether a new observation belongs to a specific population among various populations. As a result, it proves to be a valuable tool within the framework of supervised classification.

In supervised classification, the parameters of the class distributions are estimated using an observable training set. Within each class we estimate univariate CDFs on a set of random projections, which then allows to measure the depth of a point with respect to each of the classes. The allocation of a new observation from the test set to the appropriate class is determined by the maximum depth criterion among the K populations (Ghosh and Chaudhuri, 2005). Given a new statistical unit $\mathbf{y}$, the predicted class is the one with respect to which the unit has the maximum depth. The classifier can be expressed as:

$$\arg\max_{k \in K} D(\mathbf{y}, F^{(k)}).$$

where $D(\mathbf{y}, F^{(k)})$ represents the depth of $\mathbf{y}$ with respect to the population k, whose distribution is denoted as $F^{(k)}$. The prior probabilities of the populations are assumed to be equal. The use of the IRW depth in the previous classifier offers the advantage of a larger flexibility. In contrast to traditional convex depth functions, the non-convex nature of the IRW depth function provides more versatility in capturing complex decision boundaries and intricate data structures. This flexibility allows for a more accurate and nuanced classification, particularly in datasets with intricate patterns or clusters. Moreover, the definition allows for multiple choices in the modelling of the univariate projections, that can be driven by the data types and shapes. Finally, the IRW is

more computationally efficient than other flexible depths, such as the halfspace or simplicial depths.

4.3.1 Asymptotic optimality

Ghosh and Chaudhuri (2005) established the asymptotic properties of the maximum depth classifier based on the sample version of some depth functions, under certain conditions. They specifically demonstrated that when the population distribution is elliptic, with the density function strictly decreasing in every direction from its center of symmetry, the risk of maximum depth classifier based on some specific depths (including halfspace, simplicial, and projection depths) converges to the optimal Bayes risk as the sample sizes of the classes increase.

In this work we extend these findings to provide an optimality result for arbitrary population distributions, assuming that class differences arise due to location shifts. This assumption is commonly used as the basis for the asymptotic optimality of other classifiers based on median differences or quantile distances (Hall et al., 2009; Hennig and Viroli, 2016; Farcomeni et al., 2022b).

Consider the sample maximum depth classifier based on the IRW depth:

$$\arg\max_{k\in K} \sum_{b=1}^{B} \left[1 - 2|\hat{F}^{(k)}_{\mathbf{s}_b^\top \mathbf{X}}(\mathbf{s}_b^\top \mathbf{y}) - 0.5|\right] = \arg\min_{k\in K} \sum_{b=1}^{B} |\hat{F}^{(k)}_{\mathbf{s}_b^\top \mathbf{X}}(\mathbf{s}_b^\top \mathbf{y}) - 0.5|.$$

If $\hat{F}^{(k)}_{\mathbf{s}_b^\top \mathbf{X}}$ is a proper strictly monotonically increasing function, the sample classifier can be equivalently rewritten in terms of L1 distance of the projected point with respect to the medians of the classes:

$$\arg\min_{k\in K} \sum_{b=1}^{B} \left|\mathbf{s}_b^\top \mathbf{y} - Me^{(k)}(\mathbf{s}_b^\top \mathbf{X})\right|, \tag{4.7}$$

where $Me^{(k)}(\mathbf{s}_b^\top \mathbf{X})$ is the median of the projected data $\mathbf{s}_b^\top \mathbf{X}^{(k)}$ belonging to the class k. Therefore, from this standpoint, the maximum depth classifier, utilizing IRW depth functions, can be viewed as an extension of the median-based classifier (Hall et al., 2009). More precisely, unlike the median-based classifier, which operates on marginal distributions, the maximum depth classifier considers the distributions derived from projections on arbitrary directions and it

coincides with the median classifier when $B = p$ and $\mathbf{s}_1, \ldots, \mathbf{s}_p$ are the canonical directions. This distinction allows for a more flexible approach provided that each $\hat{F}^{(k)}_{\mathbf{s}_b^\top \mathbf{X}}(\mathbf{s}_b^\top \mathbf{y})$ gives a good fit of the true $F^{(k)}_{\mathbf{s}_b^\top \mathbf{X}}(\mathbf{s}_b^\top \mathbf{y})$.

For $K = 2$ populations, the classification rule criterion for the sample IRW depth-based classifier can be written as

$$d(\mathbf{y}, \mathbf{s}_1, \ldots, \mathbf{s}_B) = \sum_{b=1}^{B} \left\{ |\hat{F}^{(2)}_{\mathbf{s}_b^\top \mathbf{X}}(\mathbf{s}_b^\top \mathbf{y}) - 0.5| - |\hat{F}^{(1)}_{\mathbf{s}_b^\top \mathbf{X}}(\mathbf{s}_b^\top \mathbf{y}) - 0.5| \right\}, \qquad (4.8)$$

where $\mathbf{X}$ are fully observed units in the training set and $\mathbf{y}$ is a new unit to be classified. The classifier assigns $\mathbf{y}$ to the first population when $d(\mathbf{y}, \mathbf{s}_1, \ldots, \mathbf{s}_B)$ is positive and vice versa. Its extension to $K > 2$ classes requires contrasting each class against the remaining $K - 1$ classes.

In the next theorem, we prove that under certain assumptions, the misclassification rate of the sample IRW depth-based classifier converges to zero when the number of projections grows to infinity along with the sample size and the variable dimension. The theorem has the same structure of the optimality result provided in Hall et al. (2009), but it is based on milder assumptions.

Our theorem is developed for any $K = 2$ classes, but its extension to $K > 2$ is straightforward even if notationally heavy.

Theorem 7 *Consider $n = \max(n_1, n_2)$, with n_1 and n_2 denoting the sample sizes of the two groups in the training set and a set of B directions sampled from a unit p-sphere, having (at least) the same order of n. Assume*

(i) The p variables $X_1^{(k)}, X_2^{(k)}, \ldots, X_p^{(k)}$ have each the same distribution as $W_1 + \mu_1^{(k)}, W_2 + \mu_2^{(k)}, \ldots, W_p + \mu_p^{(k)}$, respectively. Moreover, $Me(W_j) = 0 \; \forall j$ and $\sup_{j \geq 1} Var(W_j) = A_2 < +\infty$. Define $Z^{(k)} \equiv \mathbf{s}^\top \mathbf{X}^{(k)}$.

(ii) The first moments of the projections are uniformly bounded in a strong sense. This implies that $\forall c > 0$ and $\forall \mathbf{s}$, $\exists \mathbf{v}$ with $|\mathbf{s}^\top \mathbf{v}| > c$ such that

$$\inf_{b \geq 1} \inf_{|\mathbf{s}_b^\top \mathbf{v}| > c} \mathrm{E}\,|\mathbf{s}_b^\top \mathbf{W} + \mathbf{s}_b^\top \mathbf{v}| - \mathrm{E}\,|\mathbf{s}_b^\top \mathbf{W}| > 0.$$

(iii) For some $\epsilon > 0$, the proportion of values $b \in \{1, 2, \ldots, B\}$ for which

$$|\mathbf{s}_b^\top \boldsymbol{\mu}^{(2)} - \mathbf{s}_b^\top \boldsymbol{\mu}^{(1)}| > \epsilon$$

multiplied by $n^{1/2}$, say $n^{1/2}\sharp\mathcal{K}_\epsilon$, is of larger order than B, which means $B\left(n^{1/2}\sharp\mathcal{K}_\epsilon\right)^{-1}$ goes to zero as n and B increase.

Under the previous assumptions, the IRW depth-based maximum depth classifier in (4.8) makes the correct choice asymptotically, as $p \to \infty$, and both B, n_1 and n_2 diverge with p.

Observe that the first assumption implies that the two populations differ up to location-shifts $\mu_j^{(k)}$ from median centered distributions with finite variances. No assumption about the population distributions is made, differently from Ghosh and Chaudhuri (2005), which requires them to be elliptic and strictly decreasing in every direction from their center of symmetry.

Condition (ii) concerns uniform continuity and boundedness along every direction $\mathbf{s}_b$. The last assumption guarantees the consistency of the classifier, allowing a small number of nonzero signals as the number of directions B and the sample size n increase.

4.4 Empirical Analysis

To evaluate the performance of supervised classification with the IRW depth, we conduct experiments on both simulated and real datasets. In our comparative analysis, we assess the classification accuracy of our approach against several popular methods: these include depth-based classifiers based on other depth definitions including the Mahalanobis, projection and halfspace depths, and other general classifiers: linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), k-nearest neighbors (k-NN) and support vector machines (SVM). For the computation of the other depth functions and for fitting the DD-classifier introduced in Section 4.4.2 we use the R package `ddalpha` (Pokotylo et al., 2019).

4.4.1 Simulated data

In this section, we examine the effectiveness of the IRW depth-based maximum depth classifier on simulated datasets. We consider various scenarios, including datasets with varying degrees of complexity and different feature spaces. By comparing our approach with established methods, we aim to demonstrate

the strengths of the IRW depth and its potential for achieving competitive performance in supervised classification tasks.

We generate simulated datasets with different sample sizes $n = \{50, 100, 250\}$ and feature dimensions $p = \{2, 10, 50\}$. For each combination, we consider both independent and correlated data to evaluate the performance of the IRW-depth classifier under different levels of dependency among variables.

To explore the robustness of our approach to different data distributions, we generate data from three different scenarios: normal, Student's t-distribution with 3 degrees of freedom, and skewed data obtained through the transformation $\log(|t_{\nu=3}|)$. These distributions allow us to assess the performance of the IRW depth maximum depth classifier on datasets with varying levels of symmetry and tail behavior.

For each combination of sample size, feature dimension, data dependency, and data distribution, we generate 100 simulated datasets each made of $K = 2$ classes. Each dataset consists of training and test sets of the same sample size n. The training set is used to estimate the class parameters, while the test set is used for evaluating the classification performance. In our experiments, we employ $B = 500$ random directions to compute the IRW depth for each observation. This allows us to capture the directional information and determine the allocation of the test observations to the appropriate class.

We compare the IRW depth-based maximum depth classifier with the Gaussian, the *fgld* and the KDE as CDF estimators (MIWR); maximum depth classifiers based on different definitions of depths: Mahalanobis depth (MM), projection depth (MP), simplicial depth (MS) and halfspace depth (MH), and finally LDA and QDA. It is worth noting that QDA and the maximum depth classifier based on the Mahalanobis distance (MM) are very similar: the Mahalanobis distance is part of the group-specific multivariate normal density which is the model implicitly assumed in QDA, the only differences being the priors (estimated as the class frequencies of the training set) that are included in the QDA classification rule based on the posterior probability, and the term involving the determinant of the group covariance matrix, which is part of the normal density but not of the Mahalanobis distance.

To summarize and compare results among the different classifiers and across the simulation settings, we present the relative performance of each classifier with respect to the misclassification rate of the IRW-based classifier with the

KDE, which is taken as reference. More specifically, the results are given by computing the following scaled error difference:

$$d_{jk} = \frac{e_{jk} - e_{j1}}{\bar{e}_j},$$

where e_{jk} stands for the misclassification rate for method k (with 1 being the reference method) in the j-th setting, and $\bar{e}_j$ being the average misclassification rate for the j-th setting. Results are presented in Figures 4.7, 4.8 and 4.9, which refer respectively to Gaussian, Student's t and skewed data. Results from correlated and uncorrelated independent variables are pooled together, so each boxplot is made of 200 points (100 replicates times the 2 correlation structures). Some methods for some settings are missing: QDA and MM cannot be performed when $n_g \geq p$, because of the inversion of the group-specific covariance matrix based on n_g points, while the simplicial depth, and consequently its associated classifier (MS), becomes too computationally expensive when $p \geq 10$.

The results of our experiments demonstrate the effectiveness of the classification methods based on the IRW depth. For Gaussian data, these methods generally outperform other approaches, including LDA and QDA, which also demonstrate competitive performance. It is also worth noting that LDA exhibits higher variability in the misclassification rates across different runs. When the data has heavier tails, such as in the case of the t-distribution, LDA emerges as the best-performing method, particularly as the feature dimension (p) increases. However, the methods based on the IRW depth consistently rank second in terms of performance. In the case of skewed data, once again, the methods based on the IRW depths exhibit superior performance, with the KDE and the *fgld* models showing slightly better results compared to the Gaussian distribution. In this scenario, both LDA and QDA perform poorly, especially for larger feature dimensions ($p = 10$ and $p = 50$), while no significant differences among the classifiers are observed for $p = 2$.

The classifiers based on other depths, except for the halfspace depth, show comparable performance to the IRW depths for $p = 2$, but their performance deteriorates as p increases. While the halfspace depth method shows reasonable performance for small feature dimensions (p), its effectiveness quickly diminishes as p increases. We do not observe significant effects of the sample size (n) on the classification performance.

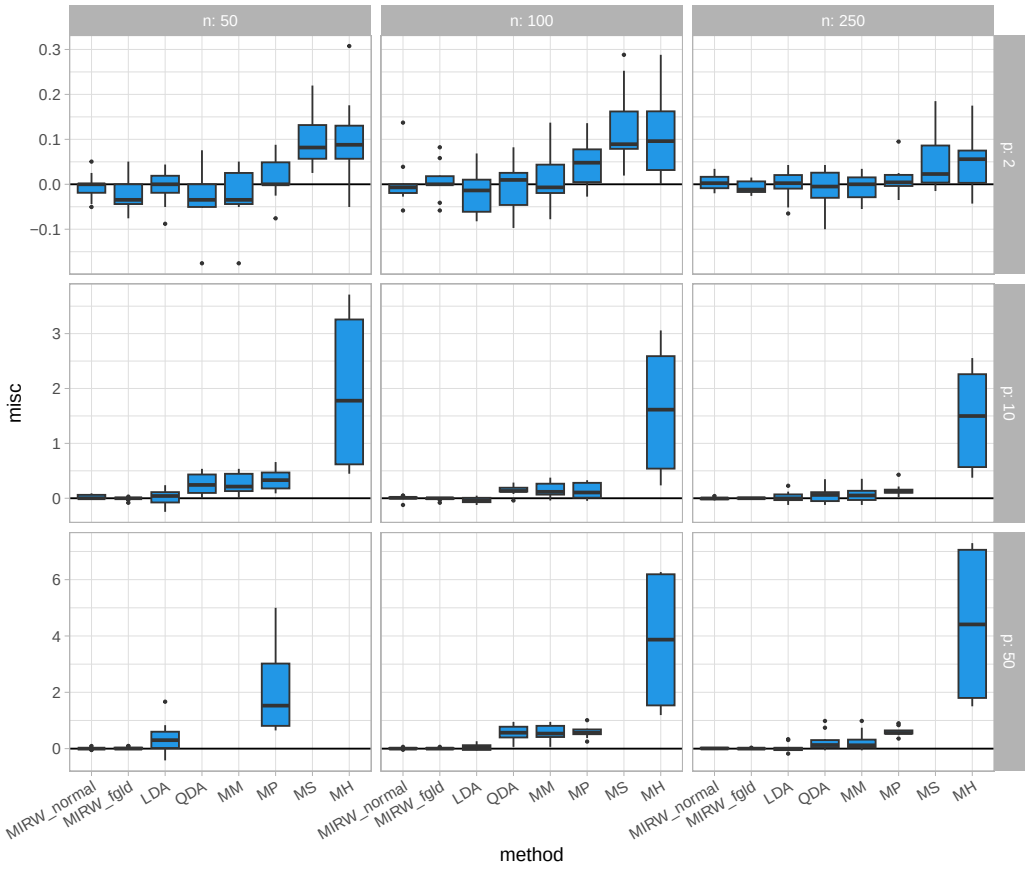

Figure 4.7: Relative performance of the classifiers with respect to the maximum depth classifier based IRW depth with KDE, which is taken as baseline. Gaussian data.

Overall, our findings highlight a very good performance of the IRW depth-based methods, which generally outperform maximum depth classifiers based on other depths and other common approaches including LDA and QDA.

4.4.2 Real data application

In this section we apply various classification methods to real datasets, commonly used as benchmarks when comparing different algorithms. We compare the following classifiers: LDA, QDA, k-nearest neighbours (KNN), support vector machines (SVM), maximum depth classifiers based on the Mahalanobis depth (MM), on the halfspace depth (MH), and on the IRW depth with the three chosen distributions (MIRW_normal, MIRW_flgd, and MIRW_kde),

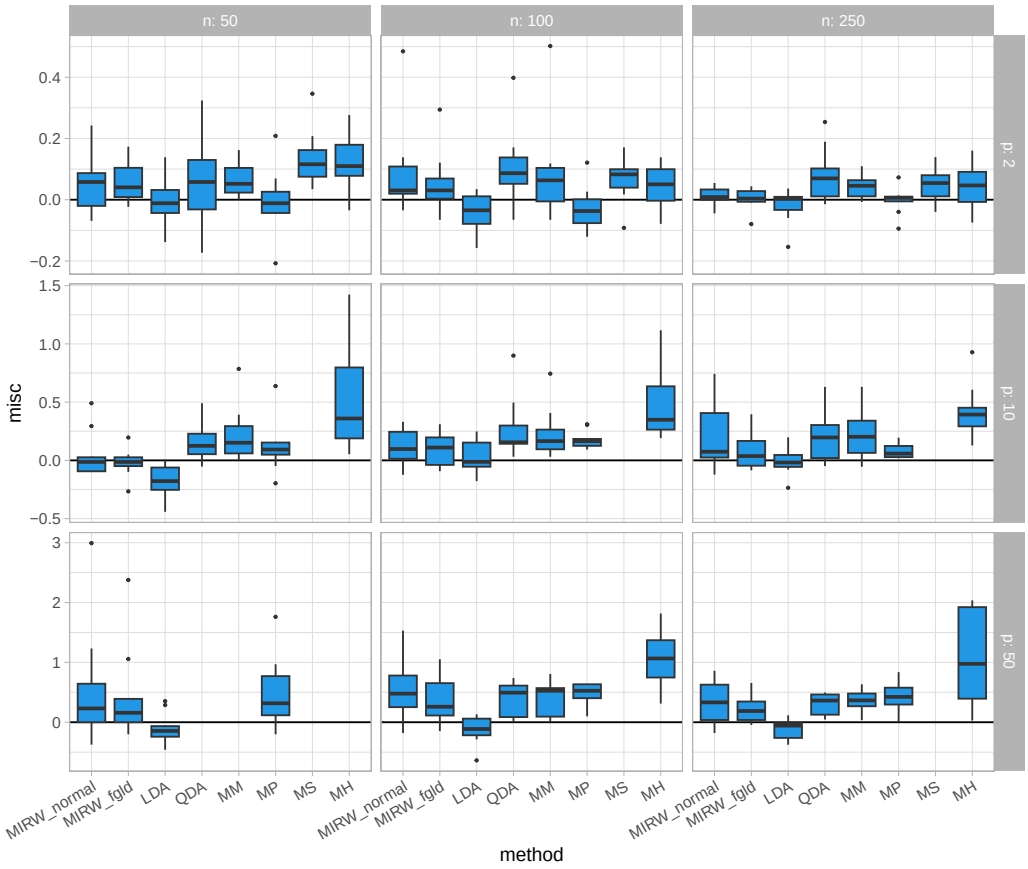

Figure 4.8: Relative performance of the classifiers with respect to the maximum depth classifier based IRW depth with KDE, which is taken as baseline. t-distributed data.

and DD-classifiers, which we introduce next, based on the same depths.

For $K = 2$ classes, the DD-plot represents the depth values of the data points with respect to the two underlying distributions, and thus transforms the samples from any dimension to a simple two-dimensional scatter-plot. On this so-called depth space, where the coordinates are the depths with respect to a class category, the idea behind the DD-classifier is that of looking for a non-linear curve, a polynomial, that best separates the two classes. In the DD-plot the classification boundary chosen by the maximum depth classifier is instead simply the bisector line between the two axes. In Figure 4.10 the translation from the original data space to the depth space given by the DD-plot is illustrated, with the addition of classification boundaries of the maximum depth and DD-classifiers. The DD-classifier was first proposed by Li et al. (2012)

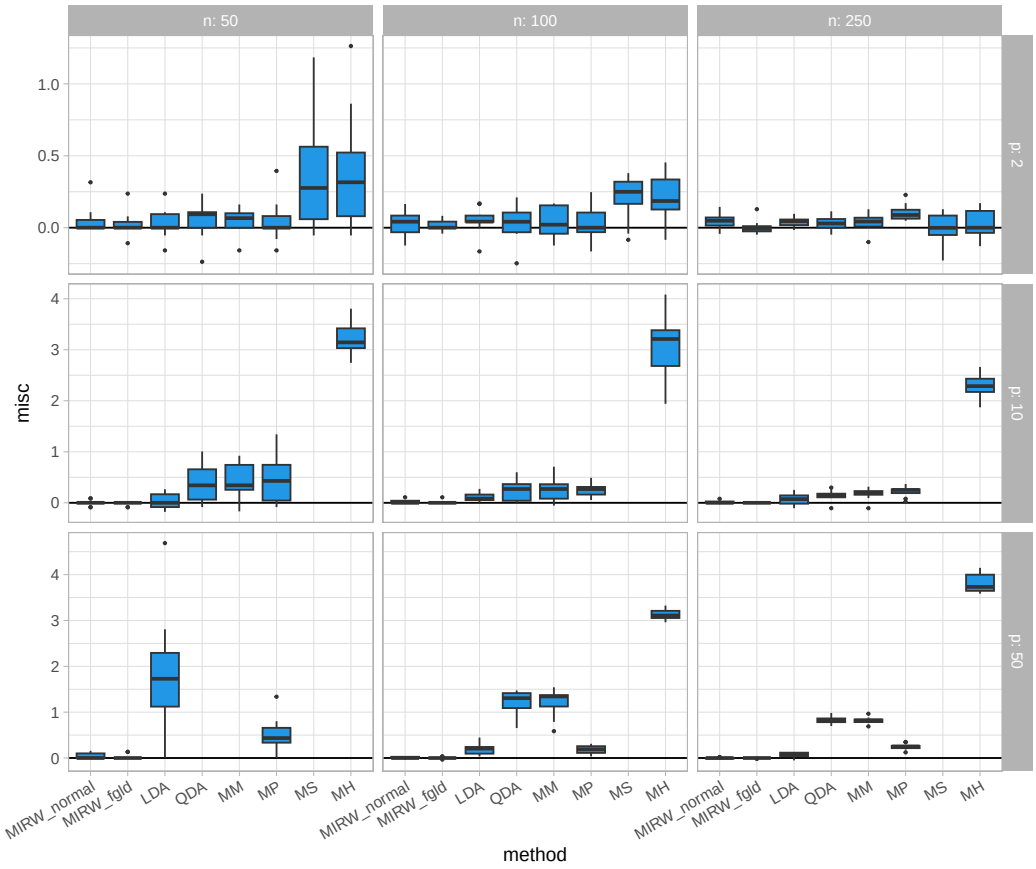

Figure 4.9: Relative performance of the classifiers with respect to the maximum depth classifier based IRW depth with KDE, which is taken as baseline. Skewed data obtained via the transformation $\log|t_{\nu=3}|$.

and has been extended in Lange et al. (2014). The DD-classifier has shown to lead to better separation and increased performance with respect to maximum depth classifiers, in particular, it is a great improvement when homoskedasticity among the groups is too restrictive of an assumption. The method can be easily extended to $K > 2$, by applying a majority vote on the results coming from training the classifier on each pair of response classes.

We applied the methods listed earlier to six datasets, whose sample sizes, number of variables, and of classes are indicated in Table 4.2. Three of these (Biomed, Blood and Image3) have also been used in Li et al. (2012).

The bankruptcy dataset (Bank), available in the R package `MixGHD`, contains the ratio of retained earnings (RE) to total assets and the ratio of earnings

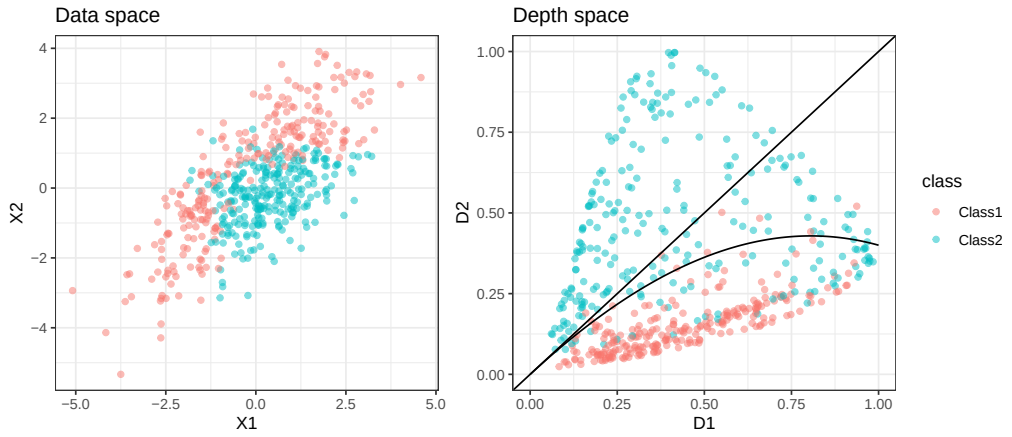

Figure 4.10: Graphical illustration of depth-based classification. In the left panel the scatter-plot of simulated data from two classes is shown. The right depicts the DD-plot with the maximum depth classifier (bisector line) and DD-classifier (polynomial separator).

	Bank	Biomed	Blood	Image3	Image4	Ionosphere	WBCD
n	66	209	748	990	1320	351	569
p	2	4	3	8	7	32	30
K	2	2	2	3	4	2	2

Table 4.2: Dataset information: sample size (n), number of variables (p) and number of classes (K).

before interests and taxes (EBIT) to total assets of 66 American firms recorded in the form of ratios and the response variable is whether the firms filed for bankruptcy. The biomedical data (Biomed), available at `http://lib.stat.cmu.edu/datasets/`, consists of four different blood measurements for 134 normal individuals and 75 carriers of rare genetic disorders. The blood transfusion dataset (Blood) contains information on 748 blood donors randomly selected from the donor database of the Blood Transfusion Service Center in Hsin-Chu City, Taiwan. This dataset is available at the UCI Machine Learning Repository (Dua and Graff, 2019b). The two groups of donors depend on whether or not the donor donated blood in March 2007. The three variables are months since the last donation, total number of donations, and months since the first donation. Two datasets are taken from the Image Segmentation,

	Bank	Biomed	Blood	Image3	Image4	Ionosphere	WBCD
LDA	0.101	0.175	0.295	0.132	0.100	0.195	0.067
QDA	0.052	0.141	0.289	0.086	0.080	0.111	**0.047**
KNN	**0.046**	0.142	0.290	0.166	0.104	0.184	0.084
SVM	0.051	0.175	0.309	**0.072**	0.073	0.115	0.056
MM	0.162	0.263	0.325	0.174	0.167	0.289	0.086
MH	0.131	0.208	0.303	0.376	0.326	0.361	0.101
MIRW_flgd	0.057	0.290	0.309	0.375	0.361	0.313	0.110
MIRW_normal	0.094	0.282	0.342	0.377	0.371	0.394	0.117
MIRW_kde	0.058	0.255	0.309	0.370	0.351	0.347	0.109
PM	0.066	0.132	0.278	0.089	0.071	0.104	0.061
PH	0.134	0.216	0.286	0.259	0.218	0.317	0.101
PIRW_flgd	0.061	0.136	0.274	0.091	**0.070**	0.089	0.054
PIRW_normal	0.062	**0.126**	0.281	0.091	0.072	**0.088**	0.054
PIRW_kde	0.064	0.135	**0.269**	0.091	0.071	0.089	0.055

Table 4.3: Mean classification error rates from 100 random training-test splits.

again part of the UCI repository. The data contains pixel information for different materials: cement, window, brickface (Image3, with $K = 3$), and with the addition of a fourth class, sky, for Image4. The Ionosphere dataset, also available from the UCI repository, concerns the classification of radar returns from the ionosphere. The targets were free electrons in the ionosphere: 'good' radar returns are those showing evidence of some type of structure in the ionosphere, while 'bad' returns are those that do not; their signals pass through the ionosphere. Finally, the WBCD dataset, from the UCI repository, concerns the diagnosis of breast cancer into malignant or benign based on 30 features computed from a digitized image of a fine needle aspirate of a breast mass.

Table 4.3 shows the results from applying each method to the different datasets in terms of mean misclassification rates from 100 random training-test splits. The IRW depth-based classifiers consistently exhibit competitive performance, frequently surpassing popular classifiers and other depth-based methods. On the WBCD dataset with a large feature dimension of 30, QDA demonstrates superior performance. Additionally, the k-NN and SVM classifiers also exhibit competitive performance, emerging as the best methods in two cases.

In almost all cases depth-based classifiers benefit in terms of misclassification

rates when using the polynomial separator of the DD-classifier, instead of the simple maximum depth assignment. Among DD-classifiers, the one based on the halfspace depth (PH) is the one that performs the worst, and is also the one that shows the least improvement from its maximum depth counterpart (MH). The DD-classifier with the Mahalanobis depth (PM) instead is quite competitive, probably due to its computational stability and simplicity and the presence of datasets with quite a large ratio between observations and variables. However, the DD-classifiers based the IRW depth (PIRW) perform generally slightly better, and reach multiple times the best error rates, with no clear winner among the three distributional approaches.

These findings suggest that the IRW depth coupled with the DD-classifier approach effectively handles diverse real-world datasets with varying complexities and feature spaces. Furthermore, this method exhibit superior performance compared to traditional classifiers in specific scenarios, showcasing their potential as valuable tools for data analysis and classification tasks.

Chapter 5

Conclusions

In this thesis, we have explored the use of quantile-based distributions and the closely related concept of depth functions for statistical modelling and classification tasks.

In the first part, Chapter 2, we started by investigating the family of linear quantile functions, focusing on the flattened generalized logistic distribution (*fgld*). Through a least squares estimation procedure, we derived unbiased and asymptotically normal estimators, enabling the development of a reliable testing procedure. As by-products, strategies for variable importance and variable selection have been obtained by the simple application of the testing procedure developed in the first part of the work. The *fgld* quantile distribution demonstrated great flexibility in capturing a wide range of data shapes, making it a valuable tool for statistical analysis and classification. The proposed novel naïve Bayes classifier based on the quantile distribution showed promising performance in empirical studies, paving the way for further exploration of its potential in various applications. A challenging extension for future work is to develop an inferential framework for multivariate quantile functions, in the spirit of Farcomeni et al. (2022), with potentially different applications and statistical purposes. One could also consider an extension to quantile regression, where we speculate that the evaluation of the impact of changes in explanatory variables on marginal distributions of an outcome could be straightforward within the family of linear quantile functions (Firpo et al. 2009).

In the second part of this thesis, Chapter 3, we have integrated the use

of quantile-based distributions in the multivariate model of independent factor analysis. This can serve as a multivariate parametric model with a built-in dimension reduction strategy, where the focus is usually on exploring and describing the relationships among the observed variables and possibly finding an interpretable summary of them.
We developed the model in a Bayesian framework, and we showed that the quantile-based *fgld* can be estimated via an MCMC algorithm. In its unrestricted form, results greatly improve by using adaptive Metropolis Hastings algorithms and we found the robust adaptive Metropolis proposal scheme to be a good choice. When fixing the first two moments of the distribution, obtaining the standard *fgld*, the same algorithm improves its performance thanks to the reduced number of parameters. The estimation is of course more computationally expensive than with density-based distributions. This is particularly true as the sample size increases, as the numerical inversion needed to evaluate the likelihood is the computational bottleneck of the process and is also the source of some increased numerical instability in the MCMC algorithms with respect to density-based likelihoods. However, thanks to the great computing power available in modern PCs, it now possible to routinely use quantile-based distributions in the Bayesian context.
The issue of the selection of the number of factors is confronted and for the Bayesian normal factor model we have shown that information criteria that are especially designed for this inferential framework, such as the DIC, WAIC and BICM, work well in simulated experiments. The results are not so satisfactory for the IFA model that we introduce and this is probably due to the fact that we can only rely on a complete likelihood, which has also shown to work poorly for model selection in the classical normal model. Further investigation into this aspect is needed. An alternative approach for model selection could be using a shrinkage prior on a very large loadings matrix to automatically select the number of factors as in the so-called sparse Bayesian infinite factor model (Bhattacharya and Dunson, 2011).

The IFA model that we introduce is able to directly adapt to a wide range of factor distributions and this has been shown in an illustration with data coming from the European Social Survey. The results have been compared closely with those coming from the classical normal factor model and with the original IFA model, which uses normal mixtures to fit the factor distributions. We

show that our model is able to capture flexible distributions, which for the first two factors result in a flatter shape and very high skewness respectively. It is able to achieve this with fewer parameters than the IFA with normal mixtures.

In the third part of the thesis, Chapter 4, we have focused on and extended the recently introduced integrated rank-weighted (IRW) depth function. This depth function satisfies the essential properties of statistical depths for sphered data; it is computationally feasible even in high dimensions, given an approximation based on random projections and it is flexible thanks to the various models one can choose for the univariate distributions resulting from the projections, among which the *fgld* proves a valuable choice. Moreover, thanks to its characterization property, it can be utilized for prediction out-of-sample, making it applicable to supervised classification problems.

Through simulated experiments and real data applications, we have evaluated the performance of the IRW depth in classification and demonstrate its effectiveness in particular compared to other depth-based classifiers that use other depth notions. Our experiments and the theoretical asymptotic optimality the maximum depth classifier based on the IRW depth, highlight the strength of classifiers based on it in handling datasets with varying complexities and feature spaces, offering superior performance in many scenarios.

Future research directions encompass exploring various model choices in the projected spaces to handle mixed-type data effectively. Additionally, a compelling area of interest lies in investigating the potential usefulness of this depth function in the context of unsupervised classification. Unsupervised learning tasks, which involve clustering and pattern discovery without labeled data, present a challenging yet crucial domain for exploring the depth function's utility in identifying underlying structures and patterns within data. Finally, in empirical data analysis, the presence of outliers can significantly impact the robustness and reliability of statistical methods. Thus, improving the understanding of how the methodological framework copes with and accommodates outliers will provide insights into its applicability in real-world scenarios.

Bibliography

Allingham, D., R. A. R. King, and K. L. Mengersen (2009, June). Bayesian estimation of quantile distributions. *Statist. Comput. 19*(2), 189–201. Publisher: Springer US.

Altman, E. I. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The Journal of Finance 23*(4), 589–609.

Attias, H. (1999, May). Independent Factor Analysis. *Neural Comput. 11*(4), 803–851. Publisher: MIT Press.

Azzalini, A. and A. Capitanio (1999, September). Statistical Applications of the Multivariate Skew Normal Distribution. *Journal of the Royal Statistical Society Series B: Statistical Methodology 61*(3), 579–602.

Bhattacharya, A. and D. B. Dunson (2011, June). Sparse Bayesian infinite factor models. *Biometrika 98*(2), 291–306.

Bradley, R. C. (2005). Basic properties of strong mixing conditions: A survey and some open questions. *Probability Surveys 2*, 107–144.

Chakrabarty, T. K. and D. Sharma (2021). A generalization of the quantile-based flattened logistic distribution. *Annals of Data Science 8*(3), 603–627.

Chandra, N. K., A. Canale, and D. B. Dunson (2023). Escaping The Curse of Dimensionality in Bayesian Model-Based Clustering. *Journal of Machine Learning Research 24*(144), 1–42.

Chopin, N. and J. Ridgway (2017, February). Leave Pima Indians Alone: Binary Regression as a Benchmark for Bayesian Computation. *Statistical Science 32*(1), 64–87.

Cramér, H. and H. Wold (1936). Some Theorems on Distribution Functions. *Journal of the London Mathematical Society s1-11*(4), 290–294.

Cuevas, A. and R. Fraiman (2009). On depth measures and dual statistics. A methodology for dealing with general data. *Journal of Multivariate Analysis 100*(4), 753–766.

David, H. A. and H. N. Nagaraja (2004, March). *Order Statistics*. John Wiley & Sons.

Donoho, D. L. and M. Gasko (1992). Breakdown Properties of Location Estimates Based on Halfspace Depth and Projected Outlyingness. *The Annals of Statistics 20*(4), 1803–1827.

Drovandi, C. C. and A. N. Pettitt (2011). Likelihood-free bayesian estimation of multivariate quantile distributions. *Computational Statistics & Data Analysis 55*(9), 2541–2556.

Drton, M. and M. Plummer (2017, March). A Bayesian Information Criterion for Singular Models. *Journal of the Royal Statistical Society Series B: Statistical Methodology 79*(2), 323–380.

Dua, D. and C. Graff (2019a). UCI machine learning repository.

Dua, D. and C. Graff (2019b). UCI Machine Learning Repository.

Farcomeni, A., M. Geraci, and C. Viroli (2022a). Directional quantile classifiers. *Journal of Computational and Graphical Statistics 31*, 907–916.

Farcomeni, A., M. Geraci, and C. Viroli (2022b). Directional Quantile Classifiers. *Journal of Computational and Graphical Statistics 31*(3), 907–916.

Freimer, M., G. Kollia, G. S. Mudholkar, and C. T. Lin (1988, January). a study of the generalized tukey lambda family. *Comm. Statist. Theory Methods 17*(10), 3547–3567. Publisher: Taylor & Francis.

Gelman, A., J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin (2013, November). *Bayesian Data Analysis* (0 ed.). Chapman and Hall/CRC.

Ghosh, A. K. and P. Chaudhuri (2005). On Maximum Depth and Related Classifiers. *Scandinavian Journal of Statistics 32*(2), 327–350.

Ghosh, J. and D. B. Dunson (2009, January). Default Prior Distributions and Efficient Posterior Computation in Bayesian Factor Analysis. *Journal of Computational and Graphical Statistics 18*(2), 306–320.

Gilchrist, W. (2000). *Statistical Modelling with Quantile Functions*. Andover, England, UK: Taylor & Francis.

Haario, H., E. Saksman, and J. Tamminen (2001). An Adaptive Metropolis Algorithm. *Bernoulli* 7(2), 223.

Hall, P., D. M. Titterington, and J.-H. Xue (2009). Median-Based Classifiers for High-Dimensional Data. *Journal of the American Statistical Association 104*(488), 1597–1608.

Hand, D. and K. Yu (2001). Idiot's Bayes - Not so Stupid After All? *International Statistical Review 69*, 385–398.

Haynes, M. and K. Mengersen (2005, March). Bayesian estimation of g-and-k distributions using MCMC. *Computational Statistics 20*(1), 7–30.

Haynes, M. A., H. L. MacGillivray, and K. L. Mengersen (1997). Robustness of ranking and selection rules using generalised g-and-k distributions. *Journal of Statistical Planning and Inference 65*(1), 45–66.

Helske, J. (2021). *ramcmc: Robust Adaptive Metropolis Algorithm*.

Hennig, C. and C. Viroli (2016). Quantile-based classifiers. *Biometrika 103*(2), 435–446.

Hosking, J. R. M. (1990, September). L-Moments: Analysis and Estimation of Distributions Using Linear Combinations of Order Statistics. *Journal of the Royal Statistical Society: Series B (Methodological) 52*(1), 105–124.

Jiang, L., H. Zhang, and Z. Cai (2008, December). A Novel Bayes Model: Hidden Naive Bayes. *IEEE Trans. Knowl. Data Eng. 21*(10), 1361–1371.

Jiang, L., L. Zhang, C. Li, and J. Wu (2018, May). A Correlation-Based Feature Weighting Filter for Naive Bayes. *IEEE Trans. Knowl. Data Eng. 31*(2), 201–213.

Jiang, L., L. Zhang, L. Yu, and D. Wang (2019, April). Class-specific attribute weighted naive Bayes. *Pattern Recognit. 88*, 321–330.

John, G. and P. Langley (1995). Estimating Continuous Distributions in Bayesian Classifiers. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pp. 338–345.

John, G. H. and P. Langley (2013). Estimating continuous distributions in bayesian classifiers. *CoRR abs/1302.4964*.

Karvanen, J. (2006, November). Estimation of quantile mixtures via L-moments and trimmed L-moments. *Comput. Statist. Data Anal. 51*(2), 947–959.

Kay, M. (2023). *ggdist: Visualizations of Distributions and Uncertainty*.

Kong, L. and I. Mizera (2012). Quantile Tomography: Using Quantiles with Multivariate Data. *Statistica Sinica 22*(4), 1589–1610.

Kong, L. and Y. Zuo (2010). Smooth depth contours characterize the underlying distribution. *Journal of Multivariate Analysis 101*(9), 2222–2226.

Koshevoy, G. and K. Mosler (1997). Zonoid trimming for multivariate distributions. *The Annals of Statistics 25*(5), 1998–2017.

Koshevoy, G. A. (2002). The Tukey Depth Characterizes the Atomic Measure. *Journal of Multivariate Analysis 83*(2), 360–364.

Lange, T., K. Mosler, and P. Mozharovskyi (2014). Fast nonparametric classification based on data depth. *Statistical Papers 55*(1), 49–69.

Leisch, F. and E. Dimitriadou (2021). *mlbench: Machine Learning Benchmark Problems*. R package version 2.1-3.

Li, J., J. A. Cuesta-Albertos, and R. Y. Liu (2012). DD-Classifier: Nonparametric Classification Procedure Based on DD-Plot. *Journal of the American Statistical Association 107*(498), 737–753.

Liu, R. Y. (1990). On a Notion of Data Depth Based on Random Simplices. *The Annals of Statistics 18*(1), 405 – 414.

Liu, R. Y. (1992). Data depth and multivariate rank tests. *L1-statistical analysis and related methods*, 279–294.

Liu, R. Y., J. M. Parelius, and K. Singh (1999). Multivariate analysis by data depth: Descriptive statistics, graphics and inference. *The Annals of Statistics 27*(3), 783–858.

Liu, R. Y. and K. Singh (1993). A Quality Index Based on Data Depth and Multivariate Rank Tests. *Journal of the American Statistical Association 88*(421), 252–260.

Lopes, H. F. and M. West (2004). Bayesian Model Assessment in Factor Analysis. *Statistica Sinica 14*(1), 41–67. Publisher: Institute of Statistical Science, Academia Sinica.

Mahalanobis, P. C. (1936). On the Generalized Distance in Statistics. *Proceedings of the National Institute of Science of India 2*, 49–55.

Montanari, A. and C. Viroli (2010a, December). Heteroscedastic factor mixture analysis. *Statistical Modelling 10*(4), 441–460.

Montanari, A. and C. Viroli (2010b, August). The independent factor analysis approach to latent variable modelling. *Statistics 44*(4), 397–416.

Montanari, A. and C. Viroli (2010c, March). A skew-normal factor model for the analysis of student satisfaction towards university courses. *Journal of Applied Statistics 37*(3), 473–487.

Mosler, K. and P. Mozharovskyi (2022). Choosing Among Notions of Multivariate Depth Statistics. *Statistical Science 37*(3), 348–368.

Nagy, S. (2021). Halfspace depth does not characterize probability distributions. *Statistical Papers 62*(3), 1135–1139.

Nair, N. U., P. G. Sankaran, and M. Dileepkumar (2022, July). Bayesian inference in quantile functions. *Communications in Statistics - Theory and Methods 51*(14), 4877–4889.

Norwegian Social Science Data Services (2020). ESS Round 10: European Social Survey Round 10 Data.

Parzen, E. (1979, March). Nonparametric Statistical Data Modeling. *J. Am. Stat. Assoc. 74*(365), 105–121.

Perepolkin, D., B. Goodrich, and U. Sahlin (2023, November). The tenets of quantile-based inference in Bayesian models. *Computational Statistics & Data Analysis 187*, 107795.

Petersen, K. B. and M. S. Pedersen (2012, nov). The matrix cookbook. Version 20121115.

Pokotylo, O., P. Mozharovskyi, and R. Dyckerhoff (2019). Depth and Depth-Based Classification with R Package ddalpha. *Journal of Statistical Software 91*, 1–46.

R Core Team (2023). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

Raftery, A., M. Newton, J. Satagopan, and P. Krivitsky (2007, January). Estimating the integrated likelihood via posterior simulation using the harmonic mean identity. *Bayesian statistics 8*.

Ramsay, K., S. Durocher, and A. Leblanc (2019). Integrated rank-weighted depth. *Journal of Multivariate Analysis 173*, 51–69.

Rayner, G. D. and H. L. MacGillivray (2002, January). Numerical maximum likelihood estimation for the g-and-k and generalized g-and-h distributions. *Statist. Comput. 12*(1), 57–75. Publisher: Kluwer Academic Publishers.

Redivo, E., C. Viroli, and A. Farcomeni (2023, April). Quantile-distribution functions and their use for classification, with application to naïve Bayes classifiers. *Statistics and Computing 33*(2), 55.

Robert, C. and G. Casella (2010). *Introducing Monte Carlo Methods with R*. New York, NY: Springer New York.

Roberts, G. O., A. Gelman, and W. R. Gilks (1997). Weak Convergence and Optimal Scaling of Random Walk Metropolis Algorithms. *The Annals of Applied Probability* 7(1), 110–120.

Sankaran, P. G., N. U. Nair, and N. N. Midhu (2016). A New Quantile Function with Applications to Reliability Analysis. *Communications in Statistics - Simulation and Computation 45*(2), 566–582. Publisher: Taylor & Francis.

Serfling, R. (2002). Quantile functions for multivariate analysis: approaches and applications. *Statistica Neerlandica 56*(2), 214–232.

Serfling, R. and Y. Zuo (2000). General notions of statistical depth function. *The Annals of Statistics 28*(2), 461–482.

Sharma, D. and T. K. Chakrabarty (2019, July). The quantile-based flattened logistic distribution: Some properties and applications. *Comm. Statist. Theory Methods 48*(14), 3643–3662.

Spiegelhalter, D. J., N. G. Best, B. P. Carlin, and A. Van Der Linde (2002, October). Bayesian Measures of Model Complexity and Fit. *Journal of the Royal Statistical Society Series B: Statistical Methodology 64*(4), 583–639.

Stam, A. J. (1982). Limit theorems for uniform distributions on spheres in high-dimensional Euclidean spaces. *Journal of Applied Probabability 19*(1), 221–228.

Stan Development Team (2023). Stan Modeling Language Users Guide and Reference Manual, 2.23.

Struyf, A. J. and P. J. Rousseeuw (1999). Halfspace Depth and Regression Depth Characterize the Empirical Distribution. *Journal of Multivariate Analysis 69*(1), 135–153.

Tortora, C., R. P. Browne, A. ElSherbiny, B. C. Franczak, and P. D. McNicholas (2021). Model-based clustering, classification, and discriminant analysis using the generalized hyperbolic distribution: MixGHD R package. *Journal of Statistical Software 98*(3), 1–24.

Tukey, J. W. (1965, January). Which part of the sample contains the Information? *Proc. Natl. Acad. Sci. U.S.A. 53*(1), 127.

Tukey, J. W. (1975). Mathematics and the Picturing of Data. *Proceedings of the International Congress of Mathematicians, Vancouver, 1975 2*, 523–531.

Vihola, M. (2012). Robust adaptive Metropolis algorithm with coerced acceptance rate. *Statistics and Computing 22*(5), 997–1008.

Viroli, C. (2007, September). Fitting the independent factor analysis model using the MCMC algorithm. *Journal of Statistical Computation and Simulation* 77(9), 725–737.

Watanabe, S. (2010). Asymptotic Equivalence of Bayes Cross Validation and Widely Applicable Information Criterion in Singular Learning Theory. *Journal of Machine Learning Research 11*(116), 3571–3594.

Yang, Y. and G. I. Webb (2009, January). Discretization for naive-Bayes learning: managing discretization bias and variance. *Machine Learning* 74(1), 39–74.

Yon, G. and P. Marjoram (2019, July). fmcmc: A friendly MCMC framework. *Journal of Open Source Software 4*(39), 1427.

Zuo, Y. (2003). Projection-Based Depth Functions and Associated Medians. *The Annals of Statistics 31*(5), 1460–1490.

Appendix A

Appendix of Chapter 2

Proof of Lemma 1

We assume that the quantile distribution function is linear with respect to parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$.

$$Q(u) = \theta_1 \, h_1(u) + \cdots + \theta_p \, h_p(u).$$

The expected value of the i-th order statistic can be written as follows, where $g(u)$ is the density of a Beta distribution with parameters equal to i and $n-i+1$.

$$\begin{aligned}
E(X_{(i)}) &= \int_0^1 Q(u)\, g(u)\, du = \\
&= \int_0^1 \left[\theta_1 \, h_1(u) + \cdots + \theta_p \, h_p(u)\right] g(u)\; du = \\
&= \int_0^1 \left[\theta_1 \, h_1(u) g(u) + \cdots + \theta_p \, h_p(u) g(u)\right]\; du = \\
&= \theta_1 \left[\int_0^1 h_1(u) g(u)\, du\right] + \cdots + \theta_p \left[\int_0^1 h_p(u) g(u)\, du\right] = \\
&= \theta_1 \, b_{1i} + \cdots + \theta_p \, b_{pi}
\end{aligned}$$

This shows that the expected value of a generic order statistic is linear with respect to those same parameters. Alternatively we can think of the proof in terms of maps, the quantile distribution function

$$Q : \boldsymbol{\theta} \to Q(\boldsymbol{\theta})$$

is a linear map by hypothesis with the following two defining properties:

$$Q(\boldsymbol{\theta}_1 + \boldsymbol{\theta}_2) = Q(\boldsymbol{\theta}_1) + Q(\boldsymbol{\theta}_2)$$

$$Q(\alpha\, \boldsymbol{\theta}_1) = \alpha\, Q(\boldsymbol{\theta}_1)$$

the expected value

$$E : Q \to E(X_{(i)})$$

is also a linear map (a definite integral is a linear map from the space of all real-valued integrable functions to $\mathbb{R}$). The composition of linear maps is linear, so $E \circ Q$ is linear.

Proof of Lemma 2

To obtained the expected value of the i-th order statistic of a sample of size n we need to solve the following integral:

$$E[X_{(i:n)}] = \frac{1}{B(i,\, n-i+1)} \int_0^1 \left[\theta_0 + \theta_1\, u + \theta_2\, \log u - \theta_3\, \log(1-u)\right] u^{i-1}(1-u)^{n-i}\, du$$

The first two additive terms are easily solvable by recognizing the beta function:

$$\int_0^1 u^{i-1}(1-u)^{n-i}\, du = B(i, n-i+1)$$

$$\int_0^1 u^{i}(1-u)^{n-i}\, du = B(i+1, n-i+1)$$

For solving the third term we can use the following rule, in which a and b are two positive real numbers.

$$\int_0^1 \log x\, x^{a-1}\,(1-x)^{b-1}\, dx = \int_0^1 \frac{\partial}{\partial\, a} x^{a-1}\,(1-x)^{b-1}\, dx =$$
$$= \frac{\partial\, B(a,\, b)}{\partial\, a} = \frac{\partial}{\partial\, a}\, \frac{\Gamma(a)\,\Gamma(b)}{\Gamma(a+b)} =$$
$$= \frac{\Gamma'(a)\Gamma(b)\Gamma(a+b) - \Gamma(a)\Gamma(b)\Gamma'(a+b)}{\Gamma(a+b)^2} =$$
$$= \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}\, \left[\frac{\Gamma'(a)\Gamma(a+b)}{\Gamma(a)\Gamma(a+b)} - \frac{\Gamma'(a+b)}{\Gamma(a+b)}\right] = B(a,\, b)\,(\psi(a) - \psi(a+b))$$

In a similar way it can be shown that:

$$\int_0^1 \log(1-x)\, x^{a-1}\,(1-x)^{b-1}\, dx = B(a,b)(\psi(b) - \psi(a+b))$$

Thus the third and fourth term are equal respectively to:

$$\int_0^1 \log(u) u^{i-1}(1-u)^{n-i}\, du = B(i,\, n-i+1)\,(\psi(i) - \psi(n+1))$$

$$\int_0^1 \log(1-u) u^{i-1}(1-u)^{n-i}\, du = B(n-i+1,\, i)\,(\psi(n-i+1) - \psi(n+1))$$

By adding together the terms multiplied by their respective parameters and simplifying the beta functions the final result is obtained.

Proof of Lemma 3

The covariance between the r-th and s-th order statistics is given by the following integral (David and Nagaraja, 2004):

$$Cov[X_{(r)}, X_{(s)}] = \frac{n!}{(r-1)!(s-r-1)!(n-s)!} \int_0^1 \int_0^v (Q(u) - E[X_{(r)}])(Q(v) - E[X_{(s)}]) \\ u^{r-1}(v-u)^{s-r-1}(1-v)^{n-s}\, du\, dv$$

Denoting the product of factorials before the double integral as $C_{n,r,s}$, the expected values of the order statistics as μ_r and μ_s, and carrying out the product of the first two terms in the integral, the formula can be rewritten as:

$$Cov[X_{(r)}, X_{(s)}] = C_{n,r,s} \int_0^1 \int_0^v Q_u\, Q_v\, u^{r-1}(v-u)^{s-r-1}(1-v)^{n-s}\, du\, dv - \mu_r \mu_s$$

Given that the quantile function for the *fgld* has 4 terms. the product $Q_u\, Q_v$ will have 16 terms, so the integral can be split into 16 parts that can be tackled one at the time. For instance, the solution of one of these 16 terms, up to the multiplicative constant $-\theta_2\, \theta_3$, is shown below:

$$
\begin{aligned}
&C_{n,r,s} \int_0^1 \int_0^v \log(u)\, \log(1-v)\, u^{r-1}(v-u)^{s-r-1}(1-v)^{n-s}\, du\, dv = \\
&= C_{n,r,s} \int_0^1 \log(1-v)(1-v)^{n-s} \int_0^v \log(u)\, u^{r-1}(v-u)^{s-r-1}\, du\, dv = \\
&= C_{n,r,s} \int_0^1 \log(1-v)(1-v)^{n-s} v^{s-1} \int_0^1 \log(vt)\, t^{r-1}(1-t)^{s-r-1}\, dt\, dv = \\
&= C_{n,r,s} \int_0^1 \log(1-v)(1-v)^{n-s} v^{s-1}\, B(r, s-r)\, [\log(v) + \psi(r) - \psi(s)]\, dv = \\
&= C_{n,r,s} B(r, s-r) \int_0^1 \log(1-v)(1-v)^{n-s} v^{s-1}\, [\log(v) + \psi(r) - \psi(s)]\, dv = \\
&= [\psi(n-s+1) - \psi(n+1)]\, [\psi(r) - \psi(n+1)] - \psi_1(n+1)
\end{aligned}
$$

The only integral that, to our understanding, has no easy expression through the identification of special functions is the following (up to the constant $-\theta_2\, \theta_3$), whose solution involves a series:

$$
\begin{aligned}
&C_{n,r,s} \int_0^1 \int_0^v \log(1-u)\, \log(v)\, u^{r-1}(v-u)^{s-r-1}(1-v)^{n-s}\, du\, dv = \\
&= C_{n,r,s} \int_0^1 \log(v)\, (1-v)^{n-s}\, v^{s-1} \int_0^1 \log(1-vt)\, t^{r-1}(1-t)^{s-r-1}\, dt\, dv = \\
&= C_{n,r,s} \int_0^1 \log(v)\, (1-v)^{n-s}\, v^{s-1} \int_0^1 \sum_{h=1}^{\infty} \frac{-(vt)^h}{h}\, t^{r-1}(1-t)^{s-r-1}\, dt\, dv = \\
&= -\, C_{n,r,s} \sum_{h=1}^{\infty} \frac{B(h+r, s-r)}{h} \int_0^1 \log(v)\, (1-v)^{n-s}\, v^{h+s-1} dv = \\
&= -\, C_{n,r,s} \sum_{h=1}^{\infty} \frac{B(h+r, s-r)}{h} \frac{\partial}{\partial h} \int_0^1 (1-v)^{n-s}\, v^{h+s-1} dv = \\
&= -\, C_{n,r,s} \sum_{h=1}^{\infty} \frac{B(h+r, s-r)}{h} B(n-s+1, h+s)(\psi(h+s) - \psi(n+h+1)) = \\
&= \frac{\Gamma(n+1)}{\Gamma(r)} \sum_{h=1}^{\infty} \frac{1}{h} \frac{\Gamma(h+r)}{\Gamma(n+h+1)} (\psi(n+h+1) - \psi(h+s))
\end{aligned}
$$

After solving the 16 integrals and getting the 16 terms from the product $\mu_r\, \mu_s$, terms with the same parameters can be collected: all of the terms involv-

ing θ_0 cancel out in the difference and the 6 combinations that are left make up the terms shown in the resulting expression.

Proof of Lemma 4

The least squares estimator for the *fgld* distribution is given by equation 2.6. The coefficients $\boldsymbol{c}_{in}$ that form the linear combination of order statistics are defined as follows:

$$\hat{\boldsymbol{\theta}} = (\mathbf{B}^\top \mathbf{B})^{-1}\mathbf{B}^\top \mathbf{x}_{(\cdot)} = \begin{bmatrix} \boldsymbol{c}_{1n} & \boldsymbol{c}_{2n} & \cdots & \boldsymbol{c}_{1nn} \end{bmatrix} \begin{bmatrix} x_{(1)} \\ x_{(2)} \\ \vdots \\ x_{(n)} \end{bmatrix},$$

that is they constitute the columns of the $p \times n$ matrix $(\mathbf{B}^\top \mathbf{B})^{-1}\mathbf{B}^\top$. To prove that they are bounded it is enough to prove that each of the elements in the matrix $(\mathbf{B}^\top \mathbf{B})^{-1}\mathbf{B}^\top$ is bounded.
We start by expanding matrix $\mathbf{B}$:

$$\mathbf{B} = \begin{bmatrix} 1 & \frac{1}{n+1} & \psi(1) - \psi(n+1) & \psi(n+1) - \psi(n) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \frac{i}{n+1} & \psi(i) - \psi(n+1) & \psi(n+1) - \psi(n-i+1) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \frac{n}{n+1} & \psi(n) - \psi(n+1) & \psi(n+1) - \psi(1) \end{bmatrix}$$

The product $\mathbf{B}^\top \mathbf{B}$ can be analytically defined up to the 4 entries that involve the summations involving the digamma functions. For them we can only define an asymptotic order, which we will denote as k. In the following it will be shown that for any $k > 1$ the boundedness of the coefficients is preserved:

$$\mathbf{B}^\top \mathbf{B} = \begin{bmatrix} n & \frac{n}{2} & -n & n \\ \frac{n}{2} & \frac{n(1+2n)}{6(1+n)} & \frac{-3n-n^2}{4(n+1)} & \frac{3n^2+n}{4(n+1)} \\ -n & \frac{-3n-n^2}{4(n+1)} & \mathcal{O}(n^k) & \mathcal{O}(n^k) \\ n & \frac{3n^2+n}{4(n+1)} & \mathcal{O}(n^k) & \mathcal{O}(n^k) \end{bmatrix}$$

Next we need to compute the inverse of $\mathbf{B}^\top \mathbf{B}$. To this aim we will use the formula for a block diagonal matrix in order to reframe the problem in terms

of the inversion 2×2 matrices (Petersen and Pedersen, 2012). First we identify four 2×2 blocks in $\mathbf{B}^\top\mathbf{B}$:

$$\mathbf{B}^\top\mathbf{B} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix},$$

then the inverse is defined as:

$$(\mathbf{B}^\top\mathbf{B})^{-1} = \begin{bmatrix} \mathbf{C}_1^{-1} & -\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\mathbf{C}_2^{-1} \\ -\mathbf{C}_2^{-1}\mathbf{A}_{21}\mathbf{A}_{11}^{-1} & \mathbf{C}_2^{-1} \end{bmatrix},$$

where

$$\begin{aligned} \mathbf{C}_1 &= \mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21} \\ \mathbf{C}_2 &= \mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}. \end{aligned}$$

In the following we derive the submatrices and their combinations needed for the inverse, we will assume that the determinants written in big O notation are not zero, so that the inverse can be computed.

$$\mathbf{A}_{22}^{-1} = \det(\mathbf{A}_{22})^{-1} \begin{bmatrix} \mathcal{O}(n^k) & \mathcal{O}(n^k) \\ \mathcal{O}(n^k) & \mathcal{O}(n^k) \end{bmatrix} = \mathcal{O}(n^{-2k}) \begin{bmatrix} \mathcal{O}(n^k) & \mathcal{O}(n^k) \\ \mathcal{O}(n^k) & \mathcal{O}(n^k) \end{bmatrix} = \begin{bmatrix} \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) \\ \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) \end{bmatrix}$$

$$\mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12} = \begin{bmatrix} \frac{7n^2+n}{4n+4} & -\frac{n(n+7)}{4(n+1)} \\ -\frac{n(n+7)}{4(n+1)} & \frac{7n^2+n}{4n+4} \end{bmatrix} = \begin{bmatrix} \mathcal{O}(n) & \mathcal{O}(n) \\ \mathcal{O}(n) & \mathcal{O}(n) \end{bmatrix}$$

$$\mathbf{C}_2 = \begin{bmatrix} \mathcal{O}(n^k) & \mathcal{O}(n^k) \\ \mathcal{O}(n^k) & \mathcal{O}(n^k) \end{bmatrix} - \begin{bmatrix} \mathcal{O}(n) & \mathcal{O}(n) \\ \mathcal{O}(n) & \mathcal{O}(n) \end{bmatrix} = \begin{bmatrix} \mathcal{O}(n^k) & \mathcal{O}(n^k) \\ \mathcal{O}(n^k) & \mathcal{O}(n^k) \end{bmatrix}$$

$$\mathbf{C}_2^{-1} = \begin{bmatrix} \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) \\ \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) \end{bmatrix}$$

$$\begin{aligned} \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21} &= \begin{bmatrix} \mathcal{O}(n) & \mathcal{O}(n) \\ \mathcal{O}(n) & \mathcal{O}(n) \end{bmatrix} \begin{bmatrix} \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) \\ \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) \end{bmatrix} \begin{bmatrix} \mathcal{O}(n) & \mathcal{O}(n) \\ \mathcal{O}(n) & \mathcal{O}(n) \end{bmatrix} = \\ &= \begin{bmatrix} \mathcal{O}(n^{-1}) & \mathcal{O}(n^{-1}) \\ \mathcal{O}(n^{-1}) & \mathcal{O}(n^{-1}) \end{bmatrix} \end{aligned}$$

$$\mathbf{C}_1 = \begin{bmatrix} \mathcal{O}(n) & \mathcal{O}(n) \\ \mathcal{O}(n) & \mathcal{O}(n) \end{bmatrix} - \begin{bmatrix} \mathcal{O}(n^{-1}) & \mathcal{O}(n^{-1}) \\ \mathcal{O}(n^{-1}) & \mathcal{O}(n^{-1}) \end{bmatrix} = \begin{bmatrix} \mathcal{O}(n) & \mathcal{O}(n) \\ \mathcal{O}(n) & \mathcal{O}(n) \end{bmatrix}$$

$$\mathbf{C}_1^{-1} = \begin{bmatrix} \mathcal{O}(n^{-1}) & \mathcal{O}(n^{-1}) \\ \mathcal{O}(n^{-1}) & \mathcal{O}(n^{-1}) \end{bmatrix}$$

$$\mathbf{A}_{11}^{-1}\mathbf{A}_{12} = \left[\mathbf{A}_{21}\mathbf{A}_{11}^{-1}\right]^\top = \begin{bmatrix} -\frac{5}{2} & -\frac{1}{2} \\ 3 & 3 \end{bmatrix}$$

$$(\mathbf{B}^\top\mathbf{B})^{-1} = \begin{bmatrix} \mathcal{O}(n^{-1}) & \mathcal{O}(n^{-1}) & \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) \\ \mathcal{O}(n^{-1}) & \mathcal{O}(n^{-1}) & \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) \\ \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) \\ \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) & \mathcal{O}(n^{-k}) \end{bmatrix}$$

The final step is to multiply the inverse we have just derived by the transpose of $\mathbf{B}$, which we will write in asymptotic notation:

$$\mathbf{B}^\top = \begin{bmatrix} \mathcal{O}(1) & \cdots & \mathcal{O}(1) \\ \mathcal{O}(1) & \cdots & \mathcal{O}(1) \\ \mathcal{O}(k) & \cdots & \mathcal{O}(k) \\ \mathcal{O}(k) & \cdots & \mathcal{O}(k) \end{bmatrix}$$

The final matrix $(\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{B}^\top$ will contain terms of order 1 ($\mathcal{O}(1)$), that is bounded, or below (from n^{-1} to n^{-k}), so all the entries of the coefficients $\boldsymbol{c}_{in}$ are bounded.

Moreover, to prove that the functions that produce the coefficients $\boldsymbol{c}_{in}$ are continuous it is enough to note that although no analytical form for the functions is available, they are the result of products and sums of the continuous functions that define the columns of $\mathbf{B}$, so they will also be continuous.

Proof of Theorem 1

The theorem is based on the application of an asymptotic result regarding the linear combinations of order statistics (David and Nagaraja, 2004, Theorem 11.4). The linear combination is denoted as:

$$L_n = \frac{1}{n}\sum_{i=1}^{n} J\left(\frac{i}{n}\right) X_{(i)},$$

where the coefficients are $\boldsymbol{c}_{in} = \frac{1}{n} J\left(\frac{i}{n}\right)$. In our case L_n is the vector of the least squares estimator $\hat{\boldsymbol{\theta}}$. The conditions for the asymptotic normality of L_n are that the variance of the distribution X is finite, which is true for the *fgld*, and that the functions $J(u)$ that define coefficients $\boldsymbol{c}_{in}$ are bounded and continuous, which is shown in Lemma 4. The expected value and variance of the limiting normal distribution are given by the ones of the linear combination. In our case these are equal respectively to the theoretical value of the parameters and the variance of the least squares estimator, for which–in the case of the *fgld*–we have an exact result, thanks to Lemmas 2 and 3.

Variance of the order statistics for the *quad* quantile function

$$\begin{aligned} Cov[X_{(r)},\, X_{(s)}] = \theta_1^2 \; & \frac{r(n-s+1)}{(n+1)^2(n+2)} + \\ & + \theta_1\, \theta_2 \; \frac{2r(n-s+1)(r+s+2)}{(n+1)^2(n+2)(n+3)} \\ & + \theta_2^2 \; \frac{2r(r+1)(n-s+1)(n(2s+3)+5s+6)}{(n+1)^2(n+2)^2(n+3)(n+4)} \end{aligned}$$

Appendix B

Appendix of Chapter 3

Ghosh-Dunson Bayesian factor model

In Ghosh and Dunson (2009), the authors introduce a fast and efficient algorithm for Bayesian factor analysis. They use a parameter expansion approach, which involves taking posterior samples from an overparametrised working model, from which the parameters of interest of the inferential model can be recovered through a transformation.

To solve the identifiability problem of the factor model, the matrix of factor loadings $\mathbf{\Lambda}$ is constrained to be lower triangular matrix. In the following, we use the same notation as in the main text. It is worth noting that we denote the l-th row of matrix $\mathbf{\Lambda}^*$, which is also lower triangular, as $\boldsymbol{\lambda}_l^*$. This means that when $l = 1, \ldots, k$ the dimension of $\boldsymbol{\lambda}_l^*$ will range from 1 to k. Nonetheless, to simplify notation, we denote them all in the same way; the distributions and parameters referred to them should be understood as having the corresponding dimension.

The working model is the following:

$$\mathbf{x}_i = \mathbf{\Lambda}^* \mathbf{y}_i^* + \mathbf{u}_i \qquad i = 1, \ldots, n$$

$$\mathbf{u}_i \sim \text{Normal}(\mathbf{0}, \mathbf{\Psi}) \qquad \mathbf{\Psi} = \text{diag}(\Psi_1, \ldots, \Psi_p)$$

$$\mathbf{y}_i^* \sim \text{Normal}(\mathbf{0}, \mathbf{\Phi}) \qquad \mathbf{\Phi} = \text{diag}(\Phi_1, \ldots, \Phi_k)$$

which, integrating out the latent variables $\mathbf{y}_i$, becomes:

$$\mathbf{x}_i \sim \text{Normal}(\mathbf{0}, \mathbf{\Lambda}^* \mathbf{\Phi} \mathbf{\Lambda}^{*\top} + \mathbf{\Psi}).$$

The transformations that turn the working parameters into their inferential counterparts are the following:

$$\begin{aligned} \mathbf{\Lambda} &= \mathbf{\Lambda}^* \mathbf{\Phi}^{\frac{1}{2}} \text{diag}(\mathbf{s}) \\ \mathbf{y}_i &= \text{diag}(\mathbf{s}) \mathbf{\Phi}^{-\frac{1}{2}} \mathbf{y}_i^*, \end{aligned}$$

where $\mathbf{s}$ is the vector containing the signs of the main diagonal of $\mathbf{\Lambda}^*$:

$$\mathbf{s} = \text{sgn}(\Lambda_{11}^*, \ldots, \Lambda_{kk}^*).$$

The premultiplication by $\text{diag}(\mathbf{s})$ in the previous formulas forces the main diagonal of $\mathbf{\Lambda}$ to be positive.

The priors are set as follows:

- $\boldsymbol{\lambda}_l^* \sim \text{Normal}(\boldsymbol{\mu}_\lambda, \mathbf{\Sigma}_\lambda) \quad l = 1, \ldots, p$
- $\Psi_l \sim \text{Inverse gamma}(a_\Psi, b_\Psi) \quad l = 1, \ldots, p$
- $\Phi_j \sim \text{Inverse gamma}(a_\Phi, b_\Phi) \quad j = 1, \ldots, k$

And the full conditionals, become:

- $\boldsymbol{\lambda}_l^* \mid \cdot \sim \text{Normal}\left(\mathbf{S}_\lambda \left(\mathbf{\Sigma}_\lambda^{-1} \boldsymbol{\mu}_\lambda + \frac{1}{\Psi_l} \mathbf{Y}_l^{*\top} \mathbf{x}_{[l]}\right), \mathbf{S}_\lambda\right)$
  with $\mathbf{S}_\lambda = \left(\mathbf{\Sigma}_\lambda^{-1} + \frac{1}{\Psi_l} \mathbf{Y}_l^{*\top} \mathbf{Y}_l^*\right)^{-1}$ and $l = 1, \ldots, p$.

  $\mathbf{Y}_l^*$ denotes the first $\min(l, k)$ columns of matrix $\mathbf{Y}^*$, whose i-th row is $\mathbf{y}_i^*$, while $\mathbf{x}_{[l]}$ denotes the l-th column of $\mathbf{X}$.

- $\mathbf{y}_i^* \mid \cdot \sim \text{Normal}\left(\mathbf{S}_y \mathbf{\Lambda}^{*\top} \mathbf{\Psi}^{-1} \mathbf{x}_i, \mathbf{S}_y\right)$
  with $\mathbf{S}_y = \left(\mathbf{\Phi}^{-1} + \mathbf{\Lambda}^{*\top} \mathbf{\Psi}^{-1} \mathbf{\Lambda}^*\right)^{-1}$
- $\Psi_l \mid \cdot \sim \text{Inverse-gamma}\left(a_\Psi + \frac{n}{2}, b_\Psi + \frac{\sum_{i=1}^n (x_{il} - \boldsymbol{\lambda}_l^{*\top} \mathbf{y}_i^*)^{*2}}{2}\right)$

- $\Phi_j \mid \cdot \sim$ Inverse gamma $\left(a_\Phi + \frac{n}{2}, b_\Phi + \frac{1}{2} \sum_{i=1}^{n} y_{ij}^{*2} \right)$

The Gibbs sampler works by sampling directing from the full conditional posteriors of the working parameters. However, at each iteration the transformation formulas can be applied to get posterior samples from the inferential parameters.

MCMC chains from the illustration with European Social Survey data

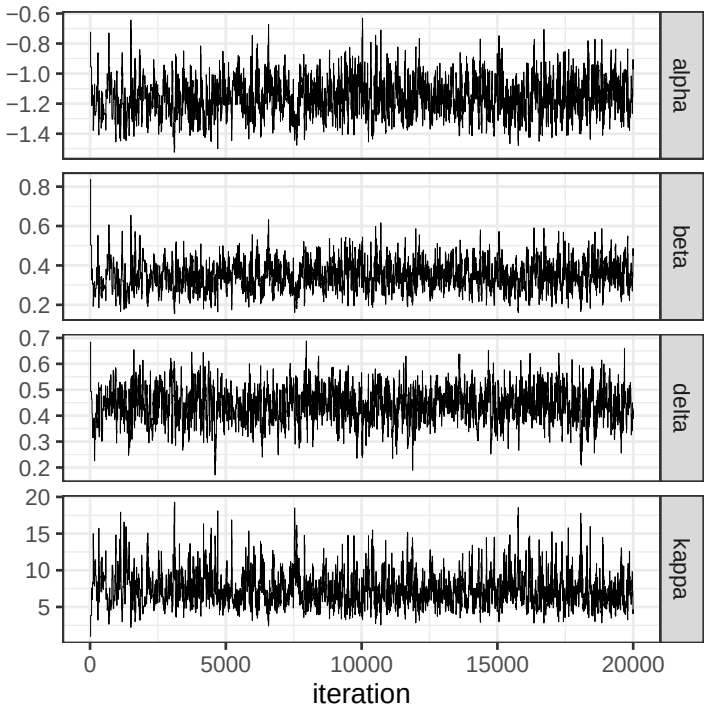

Figure B.1: Traceplots for $\boldsymbol{\theta} = (\alpha, \beta, \delta, \kappa)$ for the IFA model with *fgld* with $k = 1$ for the ESS data.

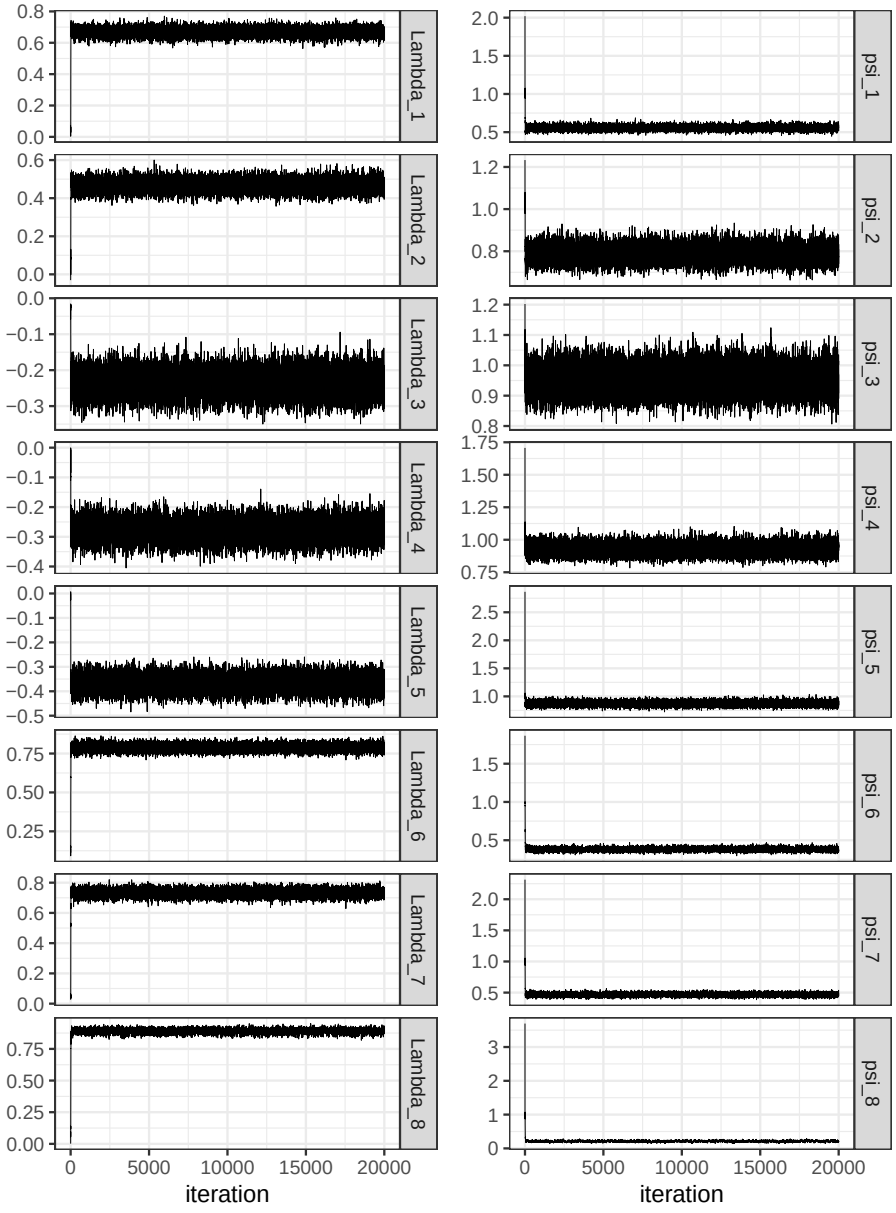

Figure B.2: Traceplots for $\mathbf{\Lambda}$ (left) and $\mathbf{\Psi}$ (right) for the IFA model with *fgld* with $k = 1$ for the ESS data.

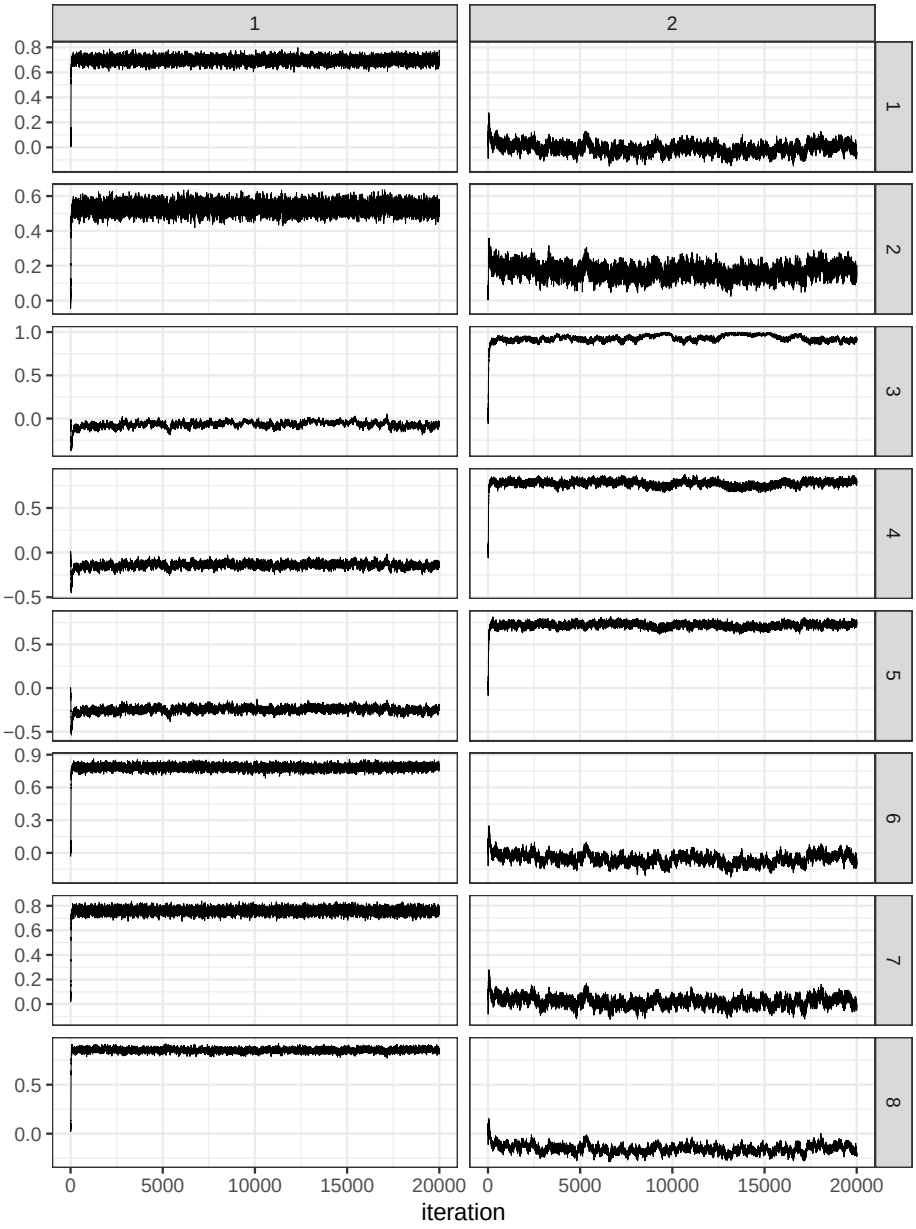

Figure B.3: Traceplots for $\mathbf{\Lambda}$ for the IFA model with *fgld* with $k = 2$ for the ESS data. The positions of the traceplots correspond to the positions of the parameters in the 8×2 matrix.

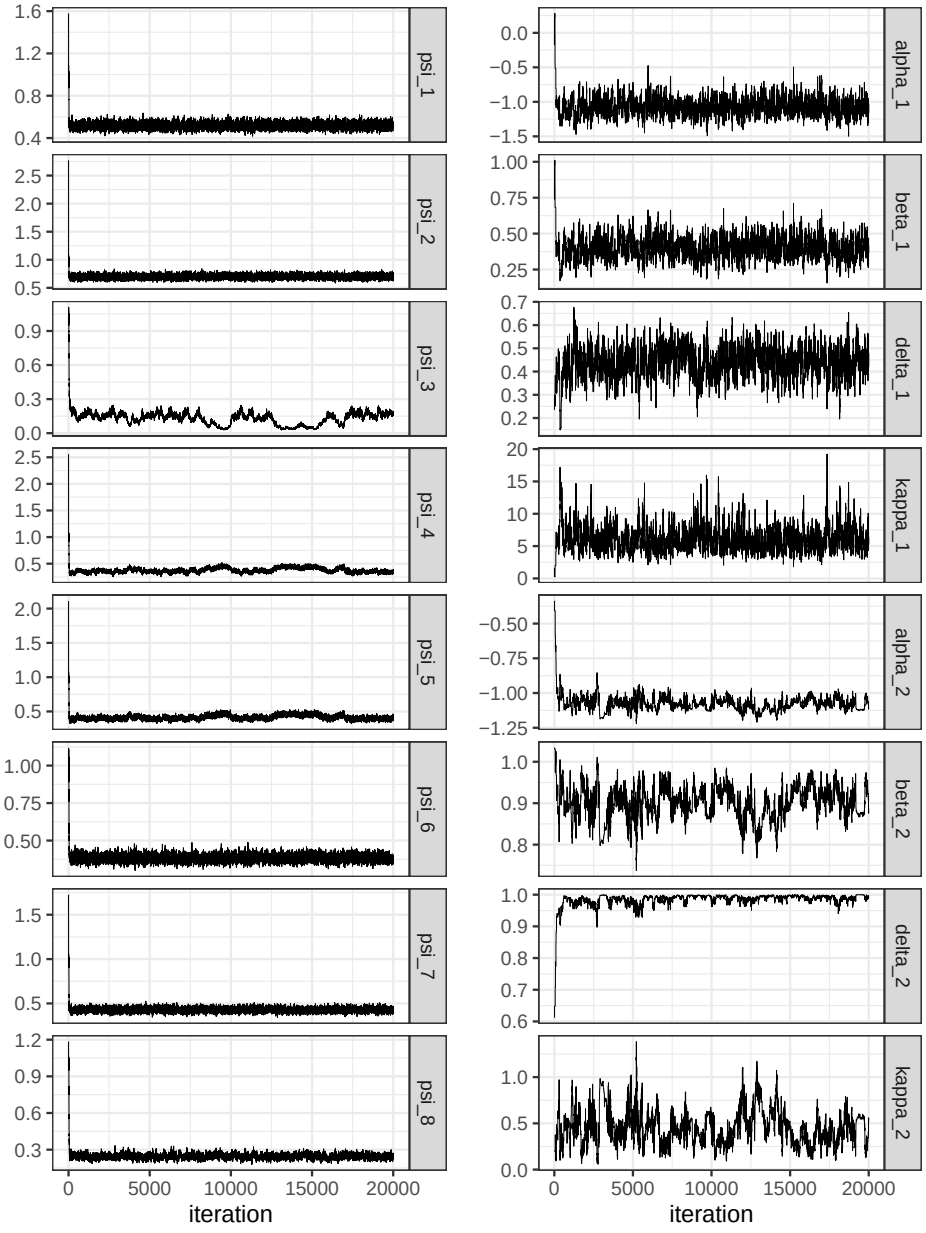

Figure B.4: Traceplots for $\boldsymbol{\Psi}$ (left) and $\boldsymbol{\theta}$ (right) for the IFA model with *fgld* with $k = 2$ for the ESS data.

Appendix C

Appendix of Chapter 4

Proof of Theorems

Lemma 1

To prove the Lemma, start with $\mathbb{E}_{\mathbf{S}}\left[(\mathbf{S}^\top \tilde{\mathbf{x}} - \mathbf{S}^\top \tilde{\boldsymbol{\mu}})^2\right]$:

$$\begin{aligned}\mathbb{E}_{\mathbf{S}}\left[(\mathbf{S}^\top \tilde{\mathbf{x}} - \mathbf{S}^\top \tilde{\boldsymbol{\mu}})^2\right] &= \mathbb{E}_{\mathbf{S}}\left[(\mathbf{S}^\top \tilde{\mathbf{x}} - \mathbf{S}^\top \tilde{\boldsymbol{\mu}})^\top (\mathbf{S}^\top \tilde{\mathbf{x}} - \mathbf{S}^\top \tilde{\boldsymbol{\mu}})\right] = \\ &= (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}})^\top \mathbb{E}_{\mathbf{S}}[\mathbf{S}\mathbf{S}^\top] (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}})\end{aligned}$$

Now consider that the covariance matrix of $\mathbf{S}$, which is a uniform vector on the sphere, is $\frac{\mathbf{I}_p}{p}$. Then:

$$\begin{aligned}\mathbb{E}_{\mathbf{S}}\left[(\mathbf{S}^\top \tilde{\mathbf{x}} - \mathbf{S}^\top \tilde{\boldsymbol{\mu}})^2\right] &= \frac{(\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}})^\top (\tilde{\mathbf{x}} - \tilde{\boldsymbol{\mu}})}{p} = \\ &= \frac{(\mathbf{W}\mathbf{x} - \mathbf{W}\boldsymbol{\mu})^\top (\mathbf{W}\mathbf{x} - \mathbf{W}\boldsymbol{\mu})}{p} = \\ &= \frac{(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})}{p}.\end{aligned}$$

Theorem 1

The depth function is clearly bounded and non-negative since it is the expectation of normalized cumulative distribution functions. The affine invariance property implies that the method is independent of the coordinate system used. Since the usual multivariate cumulative probability function is not

affine invariant we can obtain a definition of depth function uniquely defined on whitened random variables. More specifically, property (i) is true when $F_{\mathbf{S}^\top\tilde{\mathbf{Y}}}(\mathbf{S}^\top\tilde{\mathbf{y}}) = F_{\mathbf{S}^\top\tilde{\mathbf{X}}}(\mathbf{S}^\top\tilde{\mathbf{x}})$ for any direction $\mathbf{S}$, and constants $\mathbf{A}$, $\mathbf{b}$, where $\tilde{\mathbf{X}} = \mathbf{W}_x\mathbf{X}$, with $\mathbf{W}_x$ being a whitening matrix for $\mathbf{X}$,
$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{b}$, $\tilde{\mathbf{Y}} = \mathbf{W}_y\mathbf{Y}$ and $\mathbf{W}_y$ is a whitening matrix for $\mathbf{Y}$. Notice that

$$\begin{aligned} F_{\mathbf{S}^\top\tilde{\mathbf{Y}}}(\mathbf{S}^\top\tilde{\mathbf{y}}) &= P(\mathbf{S}^\top\tilde{\mathbf{Y}} \leq \mathbf{S}^\top\tilde{\mathbf{y}}) = \\ &= P(\mathbf{S}^\top(\mathbf{W}_y\mathbf{A}\mathbf{X} + \mathbf{b}) \leq \mathbf{S}^\top(\mathbf{W}_y\mathbf{A}\mathbf{x} + \mathbf{b})) = \\ &= P(\mathbf{S}^\top\mathbf{W}_y\mathbf{A}\mathbf{X} \leq \mathbf{S}^\top\mathbf{W}_y\mathbf{A}\mathbf{x}). \end{aligned}$$

Therefore we get that $P(\mathbf{S}^\top\mathbf{W}_y\mathbf{A}\mathbf{X} \leq \mathbf{S}^\top\mathbf{W}_y\mathbf{A}\mathbf{x})$ coincides with $P(\mathbf{S}^\top\mathbf{W}_x\mathbf{X} \leq \mathbf{S}^\top\mathbf{W}_x\mathbf{x})$ when $\mathbf{W}_y$ is the whitening matrix $\mathbf{W}_y = \mathbf{W}_x\mathbf{A}^{-1}$. c

Theorem 2

Consider the function $\Psi_{\mathbf{S}}(\mathbf{x}, F) = 1 - 2|F_{\mathbf{S}^\top\mathbf{X}}(\mathbf{S}^\top\mathbf{x}) - 0.5|$, which measures the depth of the point $\mathbf{x}$ with respect to the distribution of $\mathbf{X}$ in the direction of $\mathbf{S}$.

We first show that $\mathbf{S}$, $\Psi_{\mathbf{S}}(\mathbf{x}, F)$ uniquely characterizes the distribution of $\mathbf{X}$ in all directions $\mathbf{S}$. To do this, we use the Cramer-Wold theorem (Cramér and Wold, 1936), which states that a probability measure on $\mathbb{R}^p$ is uniquely determined by the marginal distributions along its one-dimensional infinite projections. This result clearly holds for any invertible linear transformation of marginal distributions, that are simply scaled and translated by a constant. However, $\Psi_{\mathbf{S}}(\mathbf{x}, F)$, as a triangular transformation, is only piece-wise invertible. Therefore, we need to modify it to make it invertible for any $\mathbf{S}$ and $\mathbf{x}$. We observe that if $\mathbf{S} \in \mathbb{S}^{p-1}$, then $-\mathbf{S} \in \mathbb{S}^{p-1}$ as well. Using this fact, we can rewrite the IRW depth function as

$$\Psi_{\mathbf{S}}(\mathbf{x}, F) = 1 - 2\left[F_{\mathbf{S}^\top\mathbf{X}}(\mathbf{S}^\top\mathbf{x})\mathbb{1}_{\left[\mathbf{S}^\top\mathbf{x}\geq 0.5\right]} + F_{-\mathbf{S}^\top\mathbf{X}}(-\mathbf{S}^\top\mathbf{x})\mathbb{1}_{\left[\mathbf{S}^\top\mathbf{x}<0.5\right]} - 0.5\right],$$

which is invertible for given $\mathbf{S}$ and $\mathbf{x}$.

Now, we claim that if $\Psi_{\mathbf{S}}(\mathbf{x}, F)$ fully characterize the multivariate distribution of $\mathbf{X}$ in all directions $\mathbf{S}$, then the expected value $\mathbb{E}_{\mathbf{S}}[\Psi_{\mathbf{S}}(\mathbf{x}, F)]$ also characterizes the distribution of $\mathbf{X}$. This can be proven by the following argument. Fully characterization of the IRW depth means that if $\mathbf{X}_1$ and $\mathbf{X}_2$ are random

variables in $\mathbb{R}^p$ with different distributions $F^{(1)}(\mathbf{x}) \neq F^{(2)}(\mathbf{x})$, then

$$Pr\left\{\mathbb{E}_{\mathbf{S}}[\Psi_{\mathbf{S}}(\mathbf{x}, F^{(1)})] \neq \mathbb{E}_{\mathbf{S}}[\Psi_{\mathbf{S}}(\mathbf{x}, F^{(2)})]\right\} = 1$$

for (at least) one value of $\mathbf{x} \in \mathbb{R}^p$. Or, alternatively,

$$Pr\left\{\left|\mathbb{E}_{\mathbf{S}}[\Psi_{\mathbf{S}}(\mathbf{x}, F^{(1)})] - \mathbb{E}_{\mathbf{S}}[\Psi_{\mathbf{S}}(\mathbf{x}, F^{(2)})]\right| = 0\right\} = 0. \tag{C.1}$$

This is true since it exist (at least) a point $\mathbf{x} \in \mathbb{R}^p$, such that $F^{(1)}(\mathbf{x}) \neq F^{(2)}(\mathbf{x})$ and, as a consequence, $F^{(1)}_{\mathbf{S}^\top\mathbf{X}}(\mathbf{S}^\top\mathbf{x}) \neq F^{(2)}_{\mathbf{S}^\top\mathbf{X}}(\mathbf{S}^\top\mathbf{x})$ due to the characterization assumption along all directions. By the strong law of large numbers and by the fact that $\Psi_{\mathbf{S}}(\mathbf{x}, F) \in [0, 1]$

$$\begin{aligned} & \left|\mathbb{E}_{\mathbf{S}}[\Psi_{\mathbf{S}}(\mathbf{x}, F^{(1)})] - \mathbb{E}_{\mathbf{S}}[\Psi_{\mathbf{S}}(\mathbf{x}, F^{(2)})]\right| \\ = & \lim_{B\to\infty} \frac{\sum_{b=1}^{B} \left|\Psi_{\mathbf{S}_b}(\mathbf{x}, F^{(1)}) - \Psi_{\mathbf{S}_b}(\mathbf{x}, F^{(2)})\right|}{B} = \mathbb{E}_{\mathbf{S}} \left|\Psi_{\mathbf{S}}(\mathbf{x}, F^{(1)}) - \Psi_{\mathbf{S}}(\mathbf{x}, F^{(2)})\right| \end{aligned}$$

Then observe that, since $\Psi_{\mathbf{S}}(\mathbf{x}, F) \in [0, 1]$ the expected value of the absolute difference is zero only when the $\Psi_{\mathbf{S}}(\mathbf{x}, F^{(1)}) = \Psi_{\mathbf{S}}(\mathbf{x}, F^{(2)})$ for all $\mathbf{S}$, which is not true since it fully characterizes the two different random variables, hence (C.1) is satisfied.

Theorem 3

We want to show that as $n \to \infty$ and $B \to \infty$

$$D_B(\mathbf{x}, \hat{F}_n) \xrightarrow{a.s.} D(\mathbf{x}, F).$$

If we assume that B and n diverge independently, it is enough to show that:

$$\lim_{n\to\infty} \lim_{B\to\infty} D_B(\mathbf{x}, \hat{F}_n) = D(\mathbf{x}, F).$$

By the strong law of large numbers we first observe that

$$\frac{\sum_{b=1}^{B} \left[1 - 2|\hat{F}_{\mathbf{s}_b^\top\mathbf{X}_n}(\mathbf{s}_b^\top\mathbf{x}) - 0.5|\right]}{B} \xrightarrow{a.s.} \mathbb{E}_{\mathbf{s}} \left[1 - 2|\hat{F}_{\mathbf{s}^\top\mathbf{X}_n}(\mathbf{s}^\top\mathbf{x}) - 0.5|\right]$$

that is

$$D_B(\mathbf{x}, \hat{F}_n) \xrightarrow{a.s.} D(\mathbf{x}, \hat{F}_n)$$

as $B \to \infty$. Now, let $D_{\mathbf{s}}(\mathbf{x}, \hat{F}_n) = 1 - 2|\hat{F}_{\mathbf{s}^\top \mathbf{X}_n}(\mathbf{s}^\top \mathbf{x}) - 0.5|$ and $D_{\mathbf{s}}(\mathbf{x}, F) = 1 - 2|F_{\mathbf{s}^\top \mathbf{X}}(\mathbf{s}^\top \mathbf{x}) - 0.5|$. Notice that $D_{\mathbf{s}}(\mathbf{x}, \hat{F}) = O_p(1)$, and provided the estimator $\hat{F}_n$ is consistent for F (as is the empirical distribution function thanks to the Glivenko-Cantelli theorem), then $D_{\mathbf{s}}(\mathbf{x}, \hat{F}_n) \xrightarrow{a.s.} D_{\mathbf{s}}(\mathbf{x}, F)$. Thus by the dominated convergence theorem

$$\mathbb{E}[D_{\mathbf{s}}(\mathbf{x}, \hat{F}_n)] = D(\mathbf{x}, \hat{F}_n) \xrightarrow[n \to \infty]{} \mathbb{E}[D_{\mathbf{s}}(\mathbf{x}, F)] = D(\mathbf{x}, F).$$

Theorem 4

The first two properties easily come from Theorem 1. Zuo (2003) established that a necessary and sufficient condition for the convexity of depth regions is that the associated depth function is quasi-concave. A functional $T(x) = F(x)$ is quasi-concave if $T(\lambda x_1 + (1 - \lambda)x_2) \geq \min\{T(x_1), T(x_2)\}$ for any $0 \leq \lambda \leq 1$ and two points x_1, x_2 in $\mathbb{R}$. The univariate depth transformation along each direction is quasi-concave, but sums and the expectation of quasi-concave functions are not necessarily quasi-concave. Hence, we can conclude that contours and regions of the IRW depth function are not convex.

Theorem 5

The structure of the proof is an adaptation of the optimality theorem provided in Farcomeni et al. (2022b) to the depth classifier. First observe that by equation (6) in the paper, the classifier for $\mathbf{Y}$ with unknown label is equivalent to

$$\sum_{b=1}^{B} \left\{ |\mathbf{s}_b^\top \mathbf{Y} - Me^{(2)}(\mathbf{s}_b^\top \mathbf{X})| - |\mathbf{s}_b^\top \mathbf{Y} - Me^{(1)}(\mathbf{s}_b^\top \mathbf{X})| \right\}.$$

Let $\boldsymbol{\mu}_y$ denote the vector of marginal medians of $\mathbf{Y}$, and put $\boldsymbol{\mu}_y^{(k)} = \boldsymbol{\mu}^{(k)} - \boldsymbol{\mu}_y$ for $k = 1, 2$ and write $\mathbf{V} = \mathbf{Y} - \boldsymbol{\mu}_\mathbf{y}$. By the triangular inequality

$$\begin{aligned} & |\mathbf{s}_b^\top \mathbf{Y} - Me^{(2)}(\mathbf{s}_b^\top \mathbf{X})| - |\mathbf{s}_b^\top \mathbf{Y} - Me^{(1)}(\mathbf{s}_b^\top \mathbf{X})| \\ = \; & |\mathbf{s}_b^\top \mathbf{V} - \mathbf{s}_b^\top \boldsymbol{\mu}_y^{(2)}| - |\mathbf{s}_b^\top \mathbf{V} - \mathbf{s}_b^\top \boldsymbol{\mu}_y^{(1)}| \\ + \; & \tau_2 |Me^{(2)}(\mathbf{s}_b^\top \mathbf{X}) - \mathbf{s}_b^\top \boldsymbol{\mu}^{(2)}| + \tau_1 |Me^{(1)}(\mathbf{s}_b^\top \mathbf{X}) - \mathbf{s}_b^\top \boldsymbol{\mu}^{(1)}|, \end{aligned}$$

where τ_1 and τ_2 satisfy $|\tau_k| \leq 1$, $k = 1, 2$. Then, we define a new random variable $T_1 \equiv T_2 + \tau_1 R_1 + \tau_2 R_2$ where $T_2 = \sum_{b=1}^{B} |\mathbf{s}_b^\top \mathbf{V} - \mathbf{s}_b^\top \boldsymbol{\mu}_y^{(2)}| - |\mathbf{s}_b^\top \mathbf{V} -$

$\mathbf{s}_b^\top \boldsymbol{\mu}_y^{(1)}|$, $R_1 = \sum_{b=1}^{B} |Me^{(1)}(\mathbf{s}_b^\top \mathbf{X}) - \mathbf{s}_b^\top \boldsymbol{\mu}^{(1)}|$ and $R_2 = \sum_{s=1}^{S} |Me^{(2)}(\mathbf{s}_b^\top \mathbf{X}) - \mathbf{s}_b^\top \boldsymbol{\mu}^{(2)}|$ and we want to prove that if $\mathbf{Y}$ belong to the first population $P^{(1)}(T_1 > 0) \to 1$, and viceversa.

Since the empirical medians converge to the population ones we have,

$$\begin{aligned} P^{(1)}(T_1 > c_1 - 2c_2 B n^{-1/2}) &\geq P^{(1)}(T_2 > c_1) - P(R_1 > c_2 B n^{-1/2}) - P(R_2 > c_2 B n^{-1/2}) \\ &\geq P^{(1)}(T_2 > c_1) - 2\sum_{b=1}^{B} e^{-2n_1 \delta_b^{(1)}} - 2\sum_{b=1}^{B} e^{-2n_2 \delta_b^{(2)}} \end{aligned}$$

for any $c_1, c_2 > 0$, where

$$\delta_b^{(k)} = \left[\min \left\{ F^{(k)} \left(\mathbf{s}_b^\top \boldsymbol{\mu}^{(k)} + \frac{c_2 B}{n^{1/2}} \right) - 0.5, 0.5 - F^{(k)} \left(\mathbf{s}_b^\top \boldsymbol{\mu}^{(k)} - \frac{c_2 B}{n^{1/2}} \right) \right\} \right]^2 .$$

Now define

$$d_b = \mathrm{E} \left\{ |\mathbf{s}_b^\top (\mathbf{V} - \boldsymbol{\mu}_y^{(2)})| - |\mathbf{s}_b^\top (\mathbf{V} - \boldsymbol{\mu}_y^{(1)})| \right\}.$$

Given $\epsilon > 0$, let $\mathcal{K}_\epsilon$ denote the set of indices $b \in \{1, 2, \ldots, B\}$ such that

$$|\mathbf{s}_b^\top \boldsymbol{\mu}_2 - \mathbf{s}_b^\top \boldsymbol{\mu}_1| > \epsilon.$$

Suppose $\mathbf{Y}$ belongs to the first population, i.e. it has distribution $F^{(1)}$. Under this assumption we have

$$d_b = \mathrm{E}_1 |\mathbf{s}_b^\top (\mathbf{Z} + \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)| - \mathrm{E}_1 |\mathbf{s}_b^\top \mathbf{Z}|,$$

where E_1 is the expectation under $F^{(1)}$. Therefore, by assumption (ii) and provided $c \geq \epsilon$, we have

$$\sum_{b \in \mathcal{K}_\epsilon} d_b \geq a(c)(\sharp \mathcal{K}_c)$$

where $a(c) > 0$, with $a(c) = \mathrm{E}_1 |\mathbf{s}_b^\top (\mathbf{Z} + \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)| - \mathrm{E}_1 |\mathbf{s}_b^\top \mathbf{Z}|$. Then, for $\mathrm{E}_1(T_2) = \sum_{b=1}^{B} d_b$ and $\epsilon \to 0$, and $\forall c$, we have

$$\mathrm{E}_1(T_2) \geq a(c)(\sharp \mathcal{K}_c), \tag{C.2}$$

where $\sharp A$ denotes the cardinality of the set A. By the Chebychev inequality and provided that $c_1 < \frac{1}{2} \mathrm{E}_1(T_2)$, under assumption (i) we get

$$\begin{aligned} P^{(1)}(T_2 > c_1) &\geq 1 - P^{(1)}(|T_2 - \mathrm{E}_1(T_2)| > c_1) \geq 1 - c_1^{-2} \mathrm{E}_1 \{T_2 - \mathrm{E}_1(T_2)\}^2 \\ &\geq 1 - c_1^{-2} \operatorname{var}_1(T2) \geq 1 - A_2 c_1^{-2} B, \end{aligned}$$

where var_1 denotes the variance under $P^{(1)}$. At this point it is possible to show var_1 is bounded, and differently from the asymptotic result in Hall et al. (2009), here we do not require the projections obey to a ψ-mixing condition (Bradley, 2005). More specifically

$$\begin{aligned}
\mathrm{var}_1(T_2) &= \mathrm{var}_1\left\{\sum_{b=1}^{B}\left(|\mathbf{s}_b^\top(\mathbf{V}-\boldsymbol{\mu}_y^{(2)})|-|\mathbf{s}_b^\top(\mathbf{V}-\boldsymbol{\mu}_y^{(1)})|\right)\right\} \\
&\leq \mathrm{var}_1\left\{\sum_{b=1}^{B}\left(\mathbf{s}_b^\top(\mathbf{V}-\boldsymbol{\mu}_y^{(2)})-\mathbf{s}_b^\top(\mathbf{V}-\boldsymbol{\mu}_y^{(1)})\right)\right\} \\
&= \mathrm{var}_1\left\{\sum_{b=1}^{B}\left(\mathbf{s}_b^\top(\mathbf{W}+\boldsymbol{\mu}^{(1)}-\boldsymbol{\mu}^{(2)})-\mathbf{s}_b^\top\mathbf{W}\right)\right\} \\
&\leq \sum_{b=1}^{B} A_2\mathbf{s}_b^\top\mathbf{s}_b + 2\sum_{b=1}^{B-1}\sum_{b'=b+1}^{B} A_2\mathbf{s}_b^\top\mathbf{s}_{b'}. \qquad \text{(C.3)}
\end{aligned}$$

Now, we use the property that a uniform random variable on the sphere, $\mathbf{U}\in R^p$, converges to a standard Gaussian as $p\to\infty$ (Stam, 1982). Therefore, for $B\to\infty$, by the strong law of large numbers we have that the second term of (C.3) become negligible as p increases since it converges to the covariance of two independent standard Gaussians.

Finally, it remains to prove that $c_1 < \frac{1}{2}\,\mathrm{E}_1(T_2)$. To this aim, consider $c_1 = \frac{c_3 B}{n^{1/2}}$, where c_3 is a positive constant. By (C.2), the latter holds if $c_3 B n^{-1/2} < \frac{1}{2}a(c)\mathcal{K}_c$. But this is true because it implies that

$$B\left(n^{1/2}\sharp\mathcal{K}_c\right)^{-1} < \frac{1}{2}a(c)c_3^{-1},$$

where the term on the left goes to zero according to assumption (iii) while $a(c)>0$, thus $c_3^{-1}>0$. For $c_1=\frac{c_3B}{n^{1/2}}$, we have

$$P^{(1)}(T_1 > c_3Bn^{-1/2} - 2c_2Bn^{-1/2}) \geq 1 - A_2\frac{n}{c_3^2B} - 2\sum_{b=1}^{B} e^{-2n_1\delta_b^{(1)}} - 2\sum_{b=1}^{B} e^{-2n_2\delta_b^{(2)}}.$$

To complete the proof, we need to choose c_3 and c_2 such as

$$P^{(1)}(T_1>0)\geq 1-\epsilon.$$

If B has (at least) the same order of n, we have that $B \geq A_1 n$ for a constant $A_1 > 0$. Therefore, we fix ϵ and choose c_3 such that $\frac{A_2}{c_3^2 A_1} \leq \epsilon$. It follows that

$$\frac{A_2 B}{c_1^2} = A_2 \frac{n}{c_3^2 B} \leq \frac{A_2}{c_3^2 A_1} \leq \epsilon.$$

Then we choose c_2 such that $c_3 > 2c_2$ and observe that $2\sum_{b=1}^{B} e^{-2n_1 \delta_b^{(1)}} + 2\sum_{b=1}^{B} e^{-2n_2 \delta_b^{(2)}} \to 0$ for $n, B \to \infty$. Since this is true for each $\epsilon > 0$, then $P^{(1)}(T_1 > 0) \to 1$, and similarly $P^{(2)}(T_1 < 0) \to 1$.