Technische Universität München

Department of Mathematics

Master's Thesis

# Illustration of variable selection methods for clustering based on Gaussian mixture and vine copula mixture models using Alzheimer Data

Zichun Li

Supervisor: Prof. Claudia Czado, PhD

Advisor: Prof. Claudia Czado, PhD
Özge Sahin

Submission Date: 12.08.2022

I assure the single handed composition of this thesis is only supported by declared resources.

Garching,

# Acknowledgements

This thesis would not have been possible without the support of many people. Many thanks to Prof.Claudia Czado and Özge Sahin, who read my numerous revisions and helped me sort out some of the confusion. Throughout the process of writing this thesis, they provided invaluable feedback on my analysis and framework, often even responding to emails late at night and early in the morning.

And finally, thanks to my parents, and numerous friends who endured this long process with me, always offering support and love.

# Contents

# 1 Introduction

Model- and distance-based clustering approaches are commonly used for clustering multivariate data. Usually, the data contains many variables. However, in many cases, considering all variables increases the complexity of the model. In addition, the data may contain irrelevant or redundant variables that do not provide much benefit or may even cause interference in detecting the hidden structure of the data. Therefore, variable selection in clustering may not only simplify the model but also increase its accuracy. In this paper, we discussed several different approaches of variable selection for model-and distance-based clustering and apply them to the real data set to compare results. This dataset is obtained from the Alzheimer's Disease Neuroimaging Initiative, an organization that unites researchers with research data to determine the progression of Alzheimer's disease and is available for free download under https://ADNI1.loni.usc.edu/data-samples/access-data/. The ADNI dataset consists of 10 variables and three different disease states, namely **Cognitively normal**, **Mild cognitive impairment** and **Dementia**.

In the second section, we will give a theoretical background including the mathematical definitions, theory, and approaches for clustering used throughout the thesis. Section 2 consists of seven subsections. In Sections 2.1-2.3, we mainly introduce the mixture model. Besides, we also discuss the expectation-maximization algorithm, which is the basis for model-based clustering approaches. Section 2.4 briefly introduces the distance- and model-based clustering approaches and their relationship. In Section 2.5, we review the definitions and theory of vine copula. Section 2.6 provides two different performance measures for clustering to evaluate our approach later. Section 2.7 describes several different variable selection approaches that we will use later.

For the third section, we will provide the data description and carry out the exploratory data analysis. Later, we will show the clustering results of the different approaches for the ADNI dataset. In Section 4, we use the different variable selection approaches for Gaussian mixture models and perform the clustering to compare the results. In Section 5, we use the vine copula mixture model introduced by Sahin and Czado [2022] to find the structure of the ADNI dataset and compare the results with the clustering approaches based on the classical Gaussian mixture model. In addition to this, we also make preliminary variable selections for the vine copula mixture model. Section 6 will be the summary of our thesis.

# 2 Theoretical Background

## 2.1 Distributions

**Univariate normal distribution** A univariate Gaussian distribution, also known as the univariate normal distribution, is a continuous probability distribution that has a symmetrical bell-shaped curve. It consists of two parameters, mean $\mu$ and variance $\sigma^2$.

**Definition 1** (Univariate normal distribution)**.** *The probability density function of univariate normal distribution is defined as follow:*

$$\phi(x;\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}}\exp(-\frac{1}{2\sigma^2}(x-\mu)^2), \tag{1}$$

*where $\mu$ and $\sigma^2$ are scalars representing the mean and variance.*

According to the Equation (1), its density reaches its maximum value when $x$ equals the mean $\mu$ and gets smaller the further away from the mean (see a left plot of Figure 1), as the exponential function is monotonic. The univariate Gaussian distribution is denoted as $\mathcal{N}(\mu,\sigma^2)$. Therefore, if the random variable X is Gaussian distributed with mean $\mu$ and standard deviation $\sigma$, we can express it as $X \sim \mathcal{N}(\mu,\sigma^2)$.

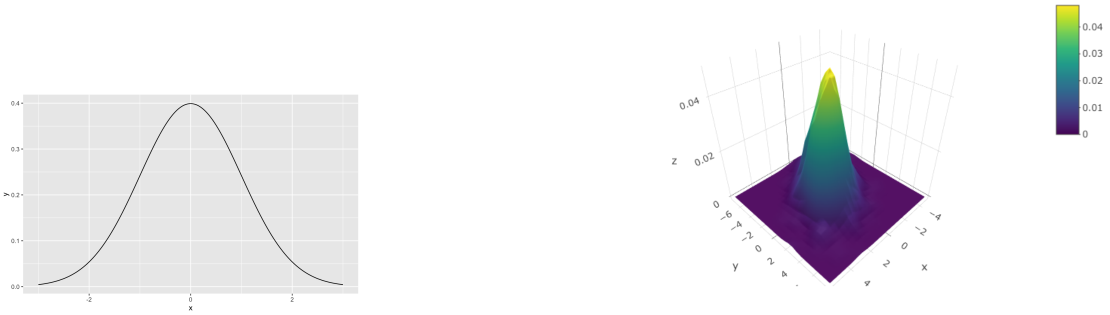


Figure 1: The left graph is univariate Gaussian density for a variable $X \sim \mathcal{N}(0,1)$ and the right one is multivariate Gaussian density over two variables $(X,Y) \sim \mathcal{N}_2(\mathbf{0},\Sigma)$, where $\Sigma = (3,2;2,5)$.

**Multivariate normal distribution** A univariate Gaussian defines the distribution of a random variable. However, in many real-world problems, we have more than one random variable. To be able to handle this multivariate situation, the multivariate normal distribution is a generalization of a univariate normal distribution to more than one variable.

**Definition 2** (Multivariate normal distribution)**.** *If a random vector $\boldsymbol{X} = (X_1,\ldots,X_D)^T$ follows a multivariate normal distribution with mean vector $\boldsymbol{\mu} \in \mathbb{R}^D$ ; covariance matrix $\Sigma \in \mathbb{R}^{D\times D}$, which we assume it is a non-singular matrix, so that the inverse of covariance matrix $\Sigma$ exists, then the density function of $\boldsymbol{X}$ at its realization $\boldsymbol{x} \in \mathbb{R}^D$ has the following form:*

$$\phi_D(\boldsymbol{x};\boldsymbol{\mu},\Sigma) = \frac{1}{2\pi^{D/2}|\Sigma|^{1/2}}\exp(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})), \tag{2}$$

*where $|\Sigma|$ is a determinant of the covariance matrix $\Sigma$ and $\Sigma^{-1}$ is the inverse of $\Sigma$. We will use abbreviation $\boldsymbol{X} \sim \mathcal{N}_D(\boldsymbol{\mu}, \Sigma)$. The right plot of Figure 1 is the example of a bivariate normal density.*

Compared to the univariate Gaussian density, for the density of multivariate Gaussian distribution, $(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})$ is also a quadratic function of vector $\boldsymbol{x}$. If the covariance matrix $\Sigma$ is positive definite, then the inverse of the covariance matrix is also positive definite. Then, by the definition of positive definiteness, if any vector $\boldsymbol{x} \neq \boldsymbol{\mu}$, we have

$$(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu}) > 0.$$

This implies,

$$-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu}) < 0.$$

Since the quadratic form $f(x)$ is a paraboloid and its level set is $f(x) = c$ with fixed c, it is an ellipsoid along the direction of the eigenvectors of the matrix $\Sigma$. Because of the negative coefficient of the quadratic function, like the univariate Gaussian density, the parabola points downwards and the density will decrease as we move away from $\boldsymbol{x} = \boldsymbol{\mu}$, as seen in the right panel of Figure 1.

**Other Marginal distribution** Before introducing the theory of copula, we will introduce some distribution density functions except for the previously mentioned Gaussian distribution defined in Czado [2019].

**Definition 3** (Log-normal distribution)**.** *A random variable $X$ is log-normal distributed if its logarithm is normal distributed. The probability density function of variable $X \in \mathbb{R}$ is defined by the mean $\mu$ and standard deviation $\sigma$, such that,*

$$f(x;\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}}\exp(-\frac{1}{2\sigma^2}(\ln x - \mu)^2), \quad x > 0 \tag{3}$$

*where $\sigma$ is also called the shape parameter which affects the shape of the log-normal distribution and $\mu$ is the parameter that changes the location of the graph. Figure* **??***shows the shape of log-normal under the different values of mean and variance.*

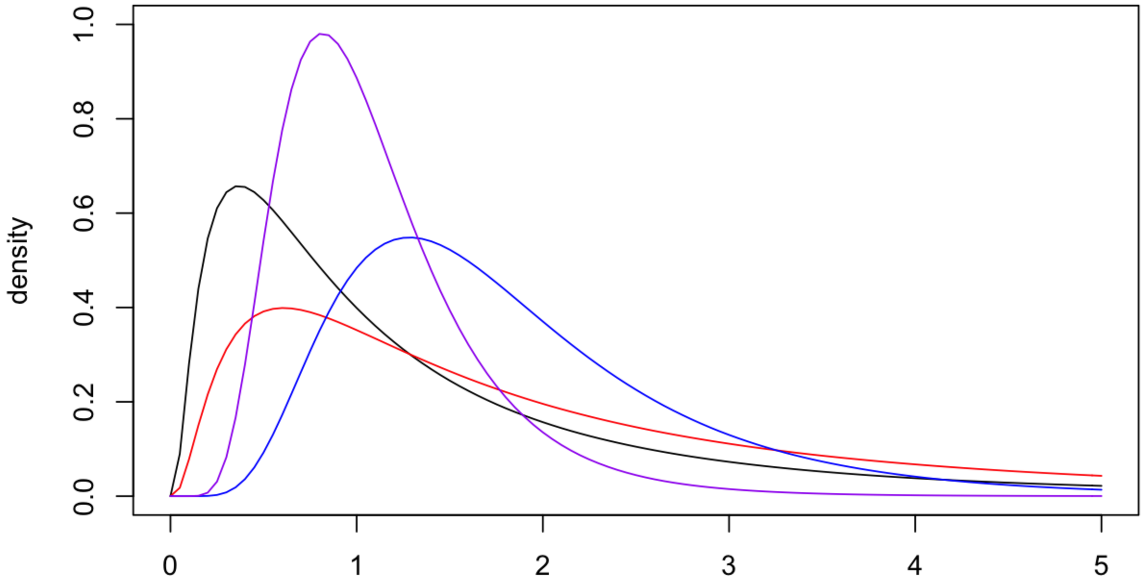


Figure 2: Density of log-normal distribution, where black is $lnorm(0, 1)$; red is $lnorm(0.5, 1)$; blue is $lnorm(0.5, 0.5)$; purple is $lnorm(0, 0.45)$. The parameter value are given inside the parenthesis $(\mu/\sigma)$.

Before further study, we first introduced a special function called the Gamma function, which is a generalization of the factorial function.

**Definition 4** (Gamma function)**.** *The **Gamma function** $\Gamma$ is defined as*

$$\Gamma(k) = \int_0^\infty x^{k-1}e^{-x}dx \quad k > 0 \tag{4}$$

**Definition 5** (chi-square distribution)**.** *If $X_1, \ldots, X_\nu$ are independent standard normal random variables, then the sum of squares $Y = \sum_{i=1}^{\nu} X_i^2$ will follows chi-square distribution with $\nu$ degrees of freedom and the probability density function at $x$ is,*

$$f(x; \nu) = \frac{1}{2^{\nu/2}\Gamma(\nu/2)} x^{\nu/2-1}e^{-x/2} \tag{5}$$

**Definition 6** (Gamma distribution)**.** *Suppose the random variable $X$ follows the Gamma distribution with shape parameter $\alpha \in (0, \infty)$ and rate parameter $\beta \in (0, \infty)$, then the probability density function at $x$ is given by,*

$$f(x; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1}e^{-\beta x} \tag{6}$$

*and its mean and variance are,*

$$\mu = E(X) = \frac{\alpha}{\beta} \quad \sigma^2 = V(X) = \frac{\alpha}{\beta^2} \tag{7}$$

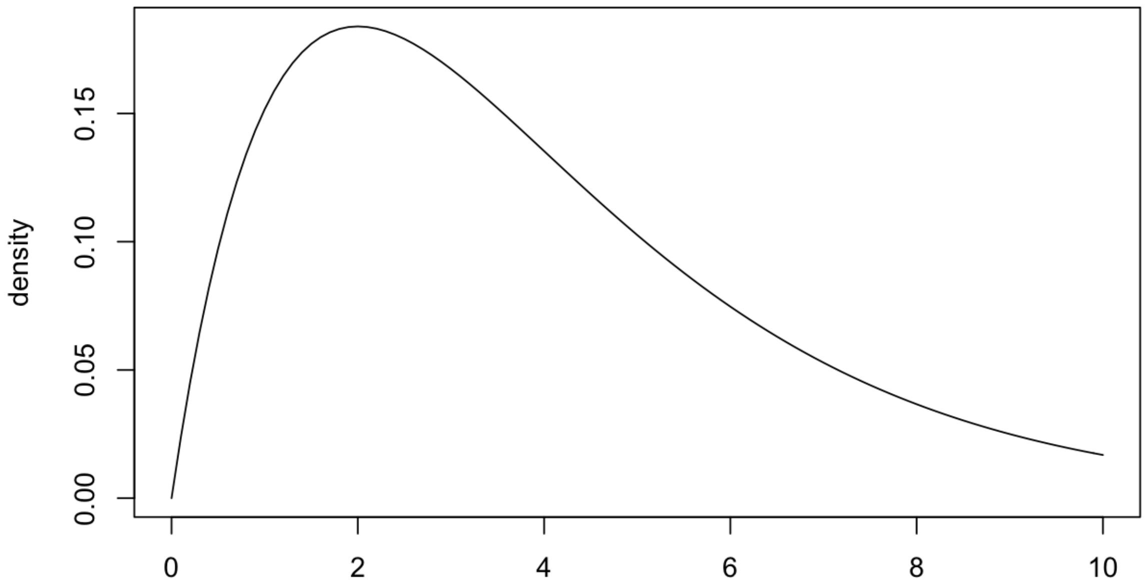


Figure 3: Density of Gamma distribution $G(2,2)$, where the parameter value are given inside the parenthesis $(\mu/s)$ $(\alpha;\beta)$.

*Besides, there is a special case of Gamma-distribution, that is the chi-square distribution where $\alpha = \nu/2$ and $\beta = \frac{1}{2}$; $\nu$ is the degree of freedom. Figure 3 shows the density of Gamma distribution with $\alpha = 2$ and $\beta = 0.5$*

**Definition 7** (Student's t distribution)**.** *Suppose $Y$ and $Z$ are independent random variable, $Y$ follows the chi-square distribution with $\nu$ degree of freedom and $Z$ follows normal distribution such that $Z \sim \mathcal{N}(0,1)$, then we could say variable $X$ is student t distribution with $\nu$ degree of freedom if it satisfies:*

$$X = \frac{Z}{\sqrt{Y/n}}$$

*and its density of at $x$ is given by :*

$$f(x;\nu) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\Gamma(\frac{\nu}{2})}(1+\frac{x^2}{\nu})^{-\frac{\nu+1}{2}} \tag{8}$$

*where $\Gamma$ is the Gamma function. The shape of the probability density function of $X$ is similar to a normal distribution with an $\mu = 0$ and $\sigma = 1$, but lower and wider. As the degree of freedom $\nu$ increases, it becomes closer to a normal distribution with an expected value of 0 and a variance of 1, see in Figure 4.*

**Definition 8** (logistic distribution)**.** *If a random variable $X \in R$ follows a logistic distribution with the mean, also called location parameter, $\mu \in \mathbb{R}$, and the scale parameter*

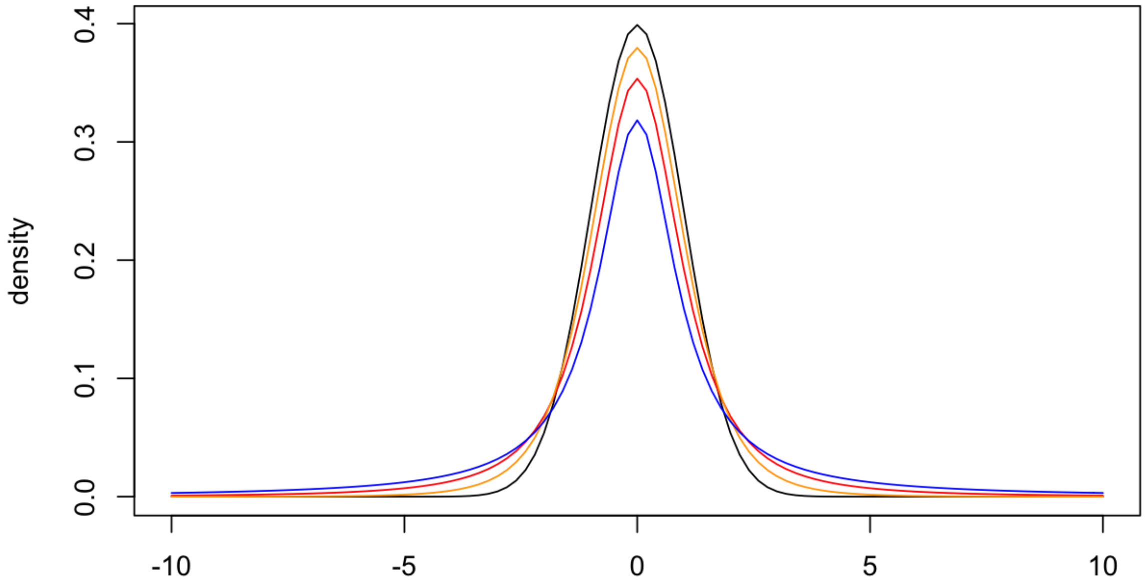


Figure 4: Density of Student's t distribution and the density of normal distribution $\mathcal{N}(0, 1)$, where black is normal distribution; red is $t(2)$; orange is $t(5)$; blue is $t(1)$.The parameter value is given inside the parenthesis ($\nu$).

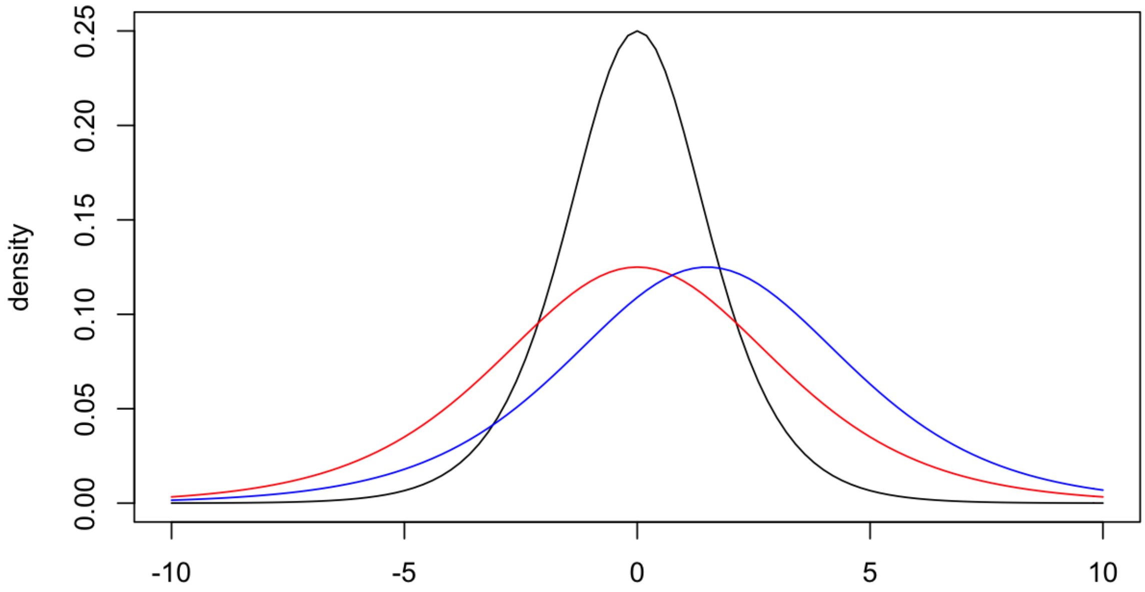


Figure 5: Density of logistic distribution, where black is $logis(0,1)$; red is $logis(0,2)$; blue is $logis(1.5,2)$.The parameters value are given inside the parenthesis $(\mu/s)$.

*$s \in \mathbb{R}^+$, then its probability density function at $x$ is given by,*

$$f(x;\mu,s) = \frac{e^{-(x-\mu)/s}}{s(1+e^{-(x-\mu)/s})^2} \tag{9}$$

*Figure 5 shows the density of logistic distribution under the different values of mean and scale parameters.*

**Definition 9** (log-logistic distribution)**.** *For the random variable $X$, if its logarithm has the logistic distribution, such that $Y = \ln(X)$ follows logistic distribution, then $X$ is log-logistic distributed. The probability density of $X$ at $x$ is given by,*

$$f(x;\alpha,\beta) = \frac{(\beta/\alpha)(x/\alpha)^{\beta-1}}{(1+(x/\alpha)^\beta)^2} \tag{10}$$

*where $\alpha > 0$ is the scale parameter and $\beta > 0$ is the shape parameter. Figure 6vshows the density of log-logistic distribution under different parameters.*

**Definition 10** (skew normal distribution)**.** *For a random variable $X$, if it is skew-normal distributed, then the probability density of $X$ at $x$ is*

$$f(x;\xi,\omega,\alpha) = \frac{2}{\omega}\phi(\frac{x-\xi}{\omega})\Phi(\alpha(\frac{x-\xi}{\omega})) \tag{11}$$

*where $\xi \in \mathbb{R}$ is location parameter, $\omega \in \mathbb{R}^+$ is scale parameter and $\alpha \in \mathbb{R}$ is shape parameter. $\phi$ and $\Phi$ are the density and distribution functions of the univariate standard*

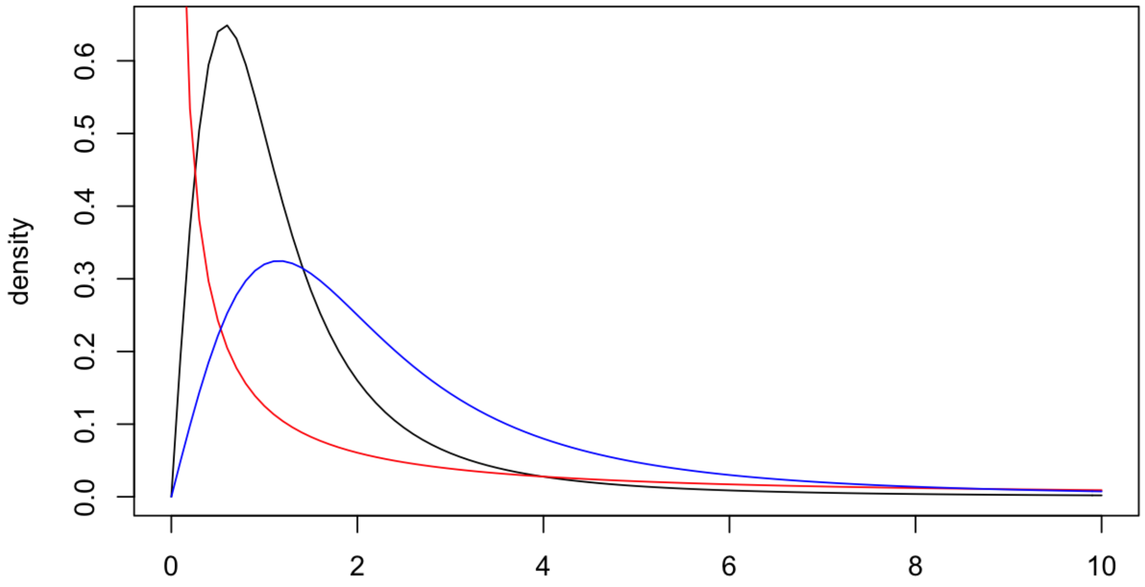


Figure 6: Density of log-logistic distribution, where black is $llogis(1, 2)$; red is $llogis(1, 0.5)$; blue is $llogis(0.5, 2)$.The parameters value are given inside the parenthesis $(\alpha/\beta)$.

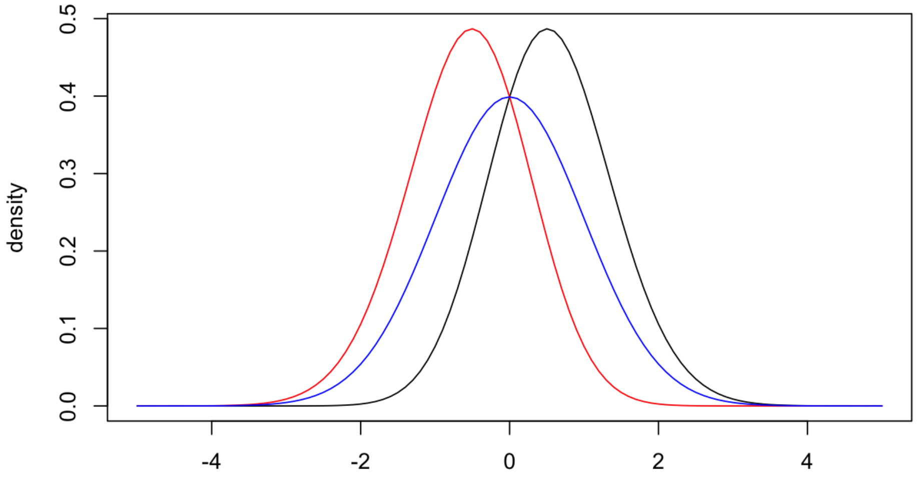


Figure 7: Density of skew-normal distribution, where black is $SN(0,1,1)$; red is $SN(0,1,-1)$; blue is $SN(0,1,0)$.The parameters value are given inside the parenthesis $(\xi,\omega,\alpha)$.

*normal distribution. If $\alpha > 0$, we say distribution is right-skewed and if $\alpha < 0$ then it is left-skewed. For $\alpha = 0$, it means the normal distribution is recovered. The example of the skew-normal distribution can be seen in Figure 7.*

**Definition 11** (skew Student's t distribution)**.**

$$f(x;\xi,\omega,\nu,\alpha) = 2t(x;\xi,\omega,\nu)T(\zeta;\nu+1) \tag{12}$$

*where*

$$\zeta = \alpha(\frac{x-\xi}{\omega})(\frac{\nu+1}{\nu+(\frac{x-\xi}{\omega})^2})^{1/2}$$

*and $t$ and $T$ are the density and distribution function of the univariate standard student-t distribution, such that*

$$t(x;\xi,\omega,\nu) = \frac{\Gamma(\frac{\nu+1}{2})(1+(\frac{x-\xi}{\omega})^2\frac{1}{\nu})^{-\nu/2-1/2}}{\omega(\pi\nu)^{1/2}\Gamma(\nu/2)}$$
$$T(x;\nu) = \int_{-\infty}^{x} t(u;0,1,\nu)du$$

*$\xi \in \mathbb{R}$ is the location parameter, $\omega \in \mathbb{R}^+$ is the scale parameter, $\nu \in \mathbb{R}^+$ describes the shape and $\alpha$ is the skewness parameter. If $\alpha = 0$, it means the standard student's t distribution is recovered. Besides, if $\alpha = 0$ and $\nu \to \infty$, then it will be the standard normal distribution. The Figure 8 shows the comparison of the density of standard normal distribution and density of skew student's t distribution under different $\nu$.*

We also introduce the definition of empirical distribution function which will be used in the later section.

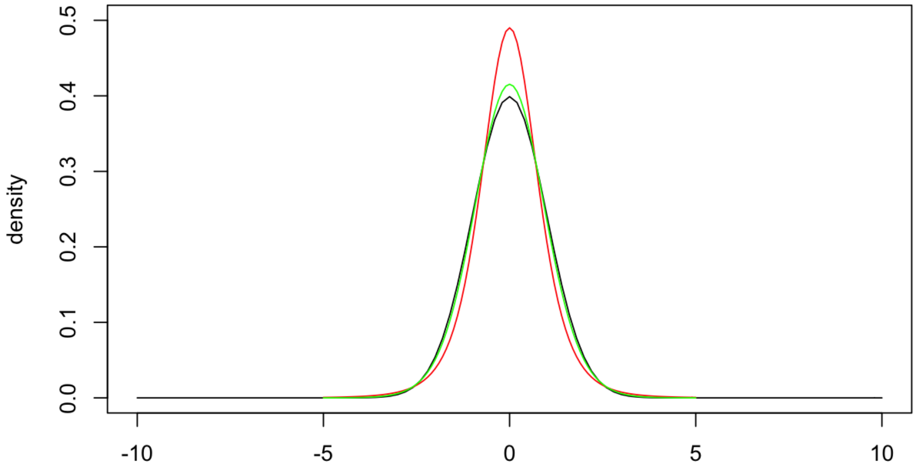


Figure 8: Density of skew Student's t distribution and standard normal distribution, where black is $N(0,1)$; red is $ST(0,1,5,1)$; green is $ST(0,1,20,1)$.The parameters value are given inside the parenthesis $(\xi, \omega, \nu, \alpha)$.

**Definition 12** (Empirical distribution)**.** *Empirical distribution function is the distribution function which is an estimate of the cumulative distribution function that associated with the points in the sample. Suppose there are n sample, $x_1, \ldots, x_n$, which are i.i.d points from the distribution function $F$, then the empirical distribution of the sample is given by,*

$$\hat{F}(x) = \frac{1}{n+1} \sum_{i=1}^{n} 1_{\{x_i \leq x\}}$$

Converting data from a given continuous distribution of random variables to a random variable with a standard uniform distribution, we have to use the **probability integral transform**, which is defined in Definition 13.

**Definition 13** (Probability integral transform)**.** *Suppose $X$ is a random variable that follows the continuous $F$ distribution, then for an observed value of variable $X$, $u := F(x)$ is called the probability integral transform (PIT) at $x$.*

According to the definition of **probability integral transform**, we can prove that for U:=F(x), it will follow the uniform distribution. Since for $u \in [0,1]$

$$P(U \leq u) = P(F(X) \leq u) = P(X \leq F^{-1}(u)) = F(F^{-1}(u)) = u.$$

Therefore, for the vector $\boldsymbol{X} = (X_1, \ldots, X_d)$ follows the multivariate distribution $F$ and $F_j$, for $for\ j = 1, \ldots, d$, are their corresponding marginal distribution functions, we can create a vector $U$ such that $\boldsymbol{U} := (U_1, \ldots, U_d) = (F_1(X_1), \ldots, F_d(X_d))$ and $U_j$ for $j = 1, \ldots, d$ are uniform distributed. Then, if $x$ is the observed sample value of $X$, we can estimate u-scale value of $x$, denoted as $u$.

## 2.2 Gaussian mixture model

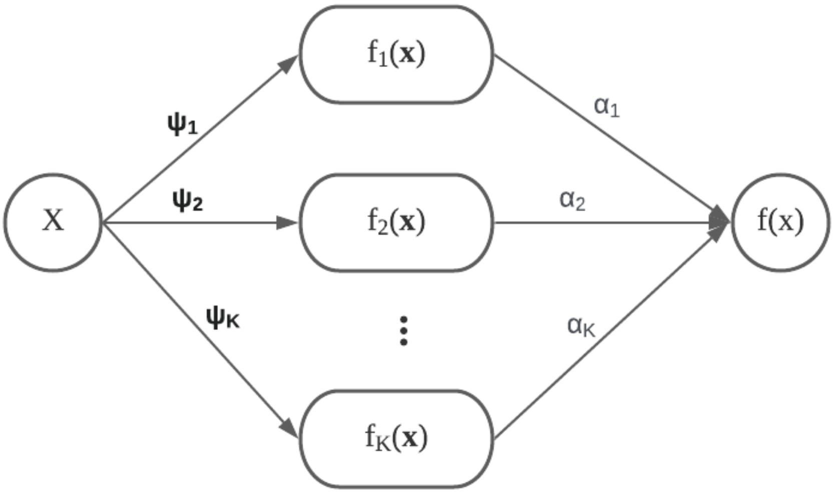


Figure 9: Structure of mixture model: $f_k(\boldsymbol{x})$ is the $k^{\text{th}}$ component density of mixture model and $f(\boldsymbol{x})$ is the density of mixture model; $\boldsymbol{\psi}_k$ is the parameters of the $k^{\text{th}}$ component's distribution; $\alpha_k$ is the mixing proportion of the $k^{th}$ component.

A mixture model can be used to model population data that are known or suspected to contain several separate subpopulations. The most commonly used one is the Gaussian mixture distribution (univariate and multivariate), which is currently the basis of a very popular statistical model for clustering.

**Definition 14** (Mixture model)**.** *A mixture model can always be expressed as the convex combination, or weighted average of several simple component distributions. The density of mixture model consisting of $K$ components for a random vector $\boldsymbol{X} \in \mathbb{R}^D$ at its realization $\boldsymbol{x} \in \mathbb{R}^D$ has the following form:*

$$\begin{aligned} f(\boldsymbol{x};\boldsymbol{\eta}) &= \sum_{k=1}^{K} \alpha_k f_k(\boldsymbol{x};\boldsymbol{\psi}_k); \\ \sum_{k=1}^{K} \alpha_k &= 1, \quad 0 \le \alpha_k \le 1. \end{aligned} \tag{13}$$

*Here, $\alpha_k$ is called the mixing coefficient for component $k$. The term $f_k(\boldsymbol{x};\boldsymbol{\psi}_k)$ denotes the probability density function of $k^{th}$ component. Further, $\boldsymbol{\psi}_k$ denote the parameters of the $k^{th}$ mixture component, and $\boldsymbol{\eta}$ collects all unknown parameters of distribution,i.e., $\boldsymbol{\eta} = (\boldsymbol{\eta}_1,\ldots,\boldsymbol{\eta}_K)^T$,$\boldsymbol{\eta}_k = (\boldsymbol{\psi}_k,\alpha_k)^T$,for $k = 1,\ldots,K$. Figure 9 illustrates the structure of the mixture model.*

### 2.2.1 Univariate Gaussian mixture model

**Definition 15** (Univariate Gaussian mixture model)**.** *Suppose that $x$ represents a random sample of a random variable $X$ from the mixture of $K$ univariate Gaussian distribution,* *then we have the following form for the density function:*

$$f(x;\boldsymbol{\eta}) = \sum_{k=1}^{K} \alpha_k \phi(x;\boldsymbol{\psi}_k), \tag{14}$$

*where* $\boldsymbol{\eta} = (\boldsymbol{\eta}_1, \ldots, \boldsymbol{\eta}_K)^T, \boldsymbol{\eta}_k = (\mu_k, \sigma_k, \alpha_k)^T, \boldsymbol{\psi}_k = (\mu_k, \sigma_k)^T$, $\mu_k \in \mathbb{R}$ *and* $\sigma_k \in \mathbb{R}$ *are the mean and standard deviation of the* $k^{th}$ *Gaussian distribution,* $\alpha_k$ *is the mixing proportion of mixture model,* $\alpha_k \geq 0$, $\sum_{k=1}^{K} \alpha_k = 1$, *for* $k = 1, \ldots, K$. *Besides,* $\phi(x;\boldsymbol{\psi})$ *is the density of univariate Gaussian distribution for the data* $x$ *with parameters* $\boldsymbol{\psi} = (\mu, \sigma^2)$.

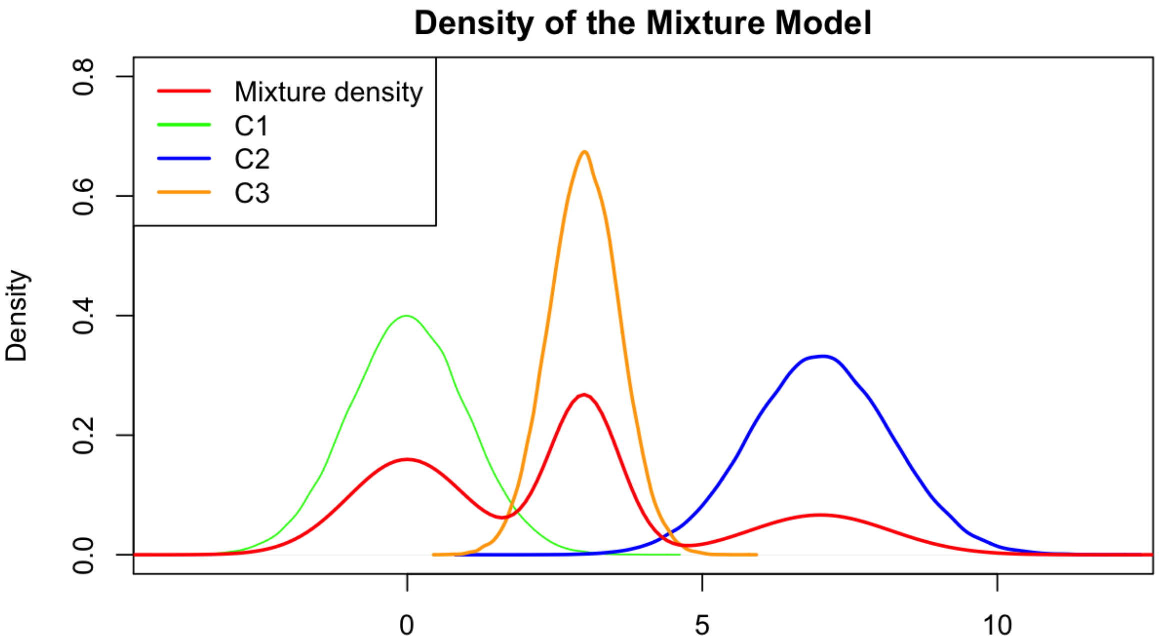


Figure 10: An example of a univariate mixture of Gaussian model, where the green line shows component $C_1 \sim \mathcal{N}(0,1)$, the blue line is $C_2 \sim \mathcal{N}(7,1.2)$ and the yellow line $C_3 \sim \mathcal{N}(3,0.6)$. Further, the mixing proportion is given as $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3)^T = (0.4, 0.2, 0.4)^T$ The mixture density is expressed as the red line.

**Example 1.** *Let's consider a simple example of a univariate Gaussian mixture model with 3 components, as shown in Figure 10. The first component's* $C_1$ *distribution is given by* $\mathcal{N}(0,1)$; *the distribution of second component* $C_2$ *is given by* $\mathcal{N}(7,1.2)$ *and of third component* $C_3$ *is given by* $\mathcal{N}(3,0.6)$. *If the probabilities of choosing components* $C_1$,$C_2$,$C_3$ *are 0.4,0.2,0.4 respectively, then the probability density function(PDF) over* $x$ *is*

$$\begin{aligned} f(x;\boldsymbol{\eta}) &= 0.4 \cdot \mathcal{N}(0,1) + 0.2 \cdot \mathcal{N}(7,1.2) + 0.4 \cdot \mathcal{N}(3,0.6) \\ &= \frac{0.4}{(2\pi)^{1/2}} \exp(-\frac{1}{2}x^2) + \frac{0.2}{(2.4\pi)^{1/2}} \exp(-\frac{1}{2.4}(x-7)^2) + \frac{0.4}{(1.2\pi)^{1/2}} \exp(-\frac{1}{1.2}(x-3)^2) \end{aligned} \tag{15}$$

where $\boldsymbol{\eta} = (0, 1, 0.4, 7, 1.2, 0.2, 3, 0.6, 0.4)^T$

### 2.2.2 Multivariate Gaussian mixture model

**Definition 16** (Multivariate Gaussian mixture model)**.** *A multivariate Gaussian mixture model is a weighted combination of multivariate Gaussian distribution. Suppose we have i.i.d observations* $\mathcal{D} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_N)^T \in \mathbb{R}^{N \times D}$*, where* $\boldsymbol{x}_i^T = (x_{i1}, \dots, x_{iD})^T \in \mathbb{R}^D$ *for* $i = 1, \dots, N$*. The density of the multivariate Gaussian mixture model with* $K$ *components at* $\boldsymbol{x}_i$ *can be written as:*

$$f(\boldsymbol{x}_i; \boldsymbol{\eta}) = \sum_{k=1}^{K} \alpha_k \phi_D(\boldsymbol{x}_i; \boldsymbol{\psi}_k) \quad \textit{for } i = 1, \dots, N, \tag{16}$$

*where* $\boldsymbol{\eta} = (\boldsymbol{\eta}_1, \dots, \boldsymbol{\eta}_K)^T$ *with* $\boldsymbol{\eta}_k = (\boldsymbol{\mu}_k, \Sigma_k, \alpha_k)^T$*,* $\boldsymbol{\psi}_k = (\boldsymbol{\mu}_k, \Sigma_k)^T$*,* $\boldsymbol{\mu}_k \in \mathbb{R}^D$ *is the mean vector,* $\Sigma_k \in \mathbb{R}^{D \times D}$ *is the covariance matrices;* $\alpha_k$ *is the mixing proportion,* $\alpha_k \geq 0$*,* $\sum_{k=1}^{K} \alpha_k = 1$*, for* $k = 1, \dots, K$*.*

Then, the loglikelihood of the Gaussian mixture model based on $\mathcal{D}$ can be expressed as,

$$L(\boldsymbol{\eta}; \mathcal{D}) = \sum_{i=1}^{N} \ln f(\boldsymbol{x}_i; \boldsymbol{\eta}) = \sum_{i=1}^{N} \ln(\sum_{k=1}^{K} \alpha_k \phi_D(\boldsymbol{x}_i; \boldsymbol{\psi}_k)). \tag{17}$$

The mixture model always contains a binary latent variable that corresponds to the membership for mixture components. Here we introduce a latent variable and build a latent model framework for multivariate Gaussian mixture model. Let $Z_k$ be a binary variable such that

$$p(Z_k = 1) = \alpha_k \qquad \text{for } k = 1, \dots, K$$

and assume that $Z_{ik}$, $i = 1, \dots, N$ are i.i.d copies of $Z_k$ with realization $z_{ik}$.

We now use the binary latent variables $Z_i$=$\{Z_{ik}, k = 1, \dots, K\}$,for $i = 1, \dots, N$,in which a particular $Z_{ik}$ is equal to 1 and other elements are equal to 0, to identify the component to which a D-dimension random vector $\boldsymbol{X}_i$ belongs, $i = 1, \dots, N$, i.e.,

$$\boldsymbol{X}_i | Z_{ik} = 1 \sim \mathcal{N}_D(\boldsymbol{\mu}_k, \Sigma_k)$$

with density $\phi_D(\cdot, \boldsymbol{\mu}_k, \Sigma_k)$ and $\sum_{k=1}^{K} Z_{ik} = 1$. Then, we define the latent conditional success probability, also called **responsibility** of $k^{\text{th}}$ component for $\boldsymbol{x}_i$, $\gamma_k(\boldsymbol{x}_i)$ such that,

$$\gamma_k(\boldsymbol{x}_i) = p(Z_{ik} = 1 | \boldsymbol{X}_i = \boldsymbol{x}_i), \qquad \text{for } i = 1, \dots, N.$$

By Bayes theorem, we have,

$$\begin{aligned} \gamma_k(\boldsymbol{x}_i; \boldsymbol{\eta}) &= \frac{p(\boldsymbol{X}_i = \boldsymbol{x}_i | Z_{ik} = 1) p(Z_{ik} = 1)}{p(\boldsymbol{X}_i = \boldsymbol{x}_i)} \\ &= \frac{\alpha_k \phi_D(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \Sigma_k)}{\sum_{j=1}^{K} p(\boldsymbol{X}_i = \boldsymbol{x}_i | Z_{ij} = 1) p(Z_{ij} = 1)} \\ &= \frac{\alpha_k \phi_D(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \Sigma_k)}{\sum_{j=1}^{K} \alpha_j \phi_D(\boldsymbol{x}_i; \boldsymbol{\mu}_j, \Sigma_j)}. \end{aligned} \tag{18}$$

**Example 2.** *To better understand the responsibility of a new data point, we use the example of one-dimension observations that we illustrate in Example 1 to show the computation process. Suppose there is a new observed data x=2.5, we can estimate the following terms,*

$$
\begin{aligned}
p(Z_1 = 1) &= 0.4 \\
p(Z_2 = 1) &= 0.2 \\
p(Z_3 = 1) &= 0.4
\end{aligned}
$$

*Since for first component $C_1$, the distribution is $\mathcal{N}(0, 1)$, given data $x$ in the first component, the probability that $x = 2.5$ is,*

$$p(x = 2.5|Z_1 = 1) \sim \mathcal{N}(0, 1) \approx 0.018$$

*For the second component $C_2 \sim \mathcal{N}(7, 1.2)$, the probability that $X = 2.5$ is*

$$p(x = 2.5|Z_2 = 1) \sim \mathcal{N}(7, 1.2) \approx 0.000$$

*For the third component $C_3 \sim \mathcal{N}(3, 0.6)$, the probability that $X = 2.5$ is*

$$p(x = 2.5|Z_3 = 1) \sim \mathcal{N}(3, 0.6) \approx 0.470$$

*Therefore,*

$$
\begin{aligned}
\gamma_1(x = 2.5) &= p(Z_1 = 1|x = 2.5) \\
&= \frac{p(Z_1 = 1)p(x|Z_1 = 1)}{p(Z_1 = 1)p(x|Z_1 = 1) + p(Z_2 = 1)p(x|Z_2 = 1) + p(Z_3 = 1)p(x|Z_3 = 1)} \\
&= 0.037
\end{aligned}
$$

*Similarly,*

$$
\begin{aligned}
\gamma_2(x = 2.5) = p(Z_2 = 1|x = 2.5) = 0, \\
\gamma_3(x = 2.5) = p(Z_3 = 1|x = 2.5) = 0.963.
\end{aligned}
$$

## 2.3 Expectation-Maximization algorithm for GMM

We cannot use the maximum likelihood method to estimate the parameters of the mixture model since the mixture coefficients are unknown, therefore we need iterative approaches. The main problem of fitting a Gaussian mixture model can be explained by the following example.

**Example 3.** *Suppose a data set $\mathcal{D} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_N)^T \in \mathbb{R}^{N\times D}$,$\boldsymbol{x}_i \in \mathbb{R}^D$, containing two subgroups, each of which is generated by a Gaussian distribution and we denote as $\mathcal{N}_D(\boldsymbol{\mu}_1, \Sigma_1)$ and $\mathcal{N}_D(\boldsymbol{\mu}_2, \Sigma_2)$ respectively.*

In the example, if we want to know which observations belong to which kind of sub-distributions, we consider two cases as follows:

If we know the parameters of two distributions and we want to know to which group each data belongs, we can estimate the responsibility $\gamma_k(\boldsymbol{x}_i)$ and assign the data to the group that they are more likely belong to.

If we know the label of the data, i.e., know the data belongs to $\mathcal{N}_D(\boldsymbol{\mu}_1, \Sigma_1)$ or $\mathcal{N}_D(\boldsymbol{\mu}_2, \Sigma_2)$, and we want to know the mean and variance matrix of two Gaussian distributions, then we can get the parameters of each distribution by estimating them over two groups of the data.

However, if we aim to find out the parameters space (mean and variance in univariate case; mean and covariance matrix in multivariate case) of the two Gaussian distributions, and to which distribution or group each data point belongs, that is both are unknown, then it will lead to a chicken-egg problem because only by knowing one can we get the other.

Considering the above two cases, there is an iterative approach with two steps. First, given the parameter of Gaussian distributions and we estimate the label of data; second, use the label that we derived in step 1 to update the parameters of Gaussian distribution.

Therefore, Dempster et al. [1977] proposed the Expectation Maximization (EM) algorithm to solve the Gaussian mixture model problem. It has two major steps, one is **Expectation step**, the other is **Maximization step**. The **expectation step** is to calculate the probability $\gamma_k(\boldsymbol{x}_i)$ by using the current value of parameters $\boldsymbol{\eta}^0 = (\boldsymbol{\mu}^0, \Sigma^0, \boldsymbol{\alpha}^0)^T$, which can be done using the Equation (18). Since we do not know the complete data loglikelihood, we will use $\gamma_k$ to find the expectation of complete data likelihood evaluated at any estimated $\boldsymbol{\eta}$. Suppose the set of all possible binary latent variables is $\mathcal{Z} = \{\boldsymbol{Z}_1, \dots, \boldsymbol{Z}_K\}$ and $\boldsymbol{Z}_i = (Z_{i1}, \dots, Z_{iK})$, where $Z_{ik} \in \{0, 1\}$. In **Maximization step**, we will estimate new parameters $\boldsymbol{\eta}^* = (\boldsymbol{\mu}_k^*, \Sigma_k^*, \alpha_k^*)_{k=1,\dots,K}$ by maximizing the expectation of complete data log-likelihood.

First, in the E-step, the $\boldsymbol{\eta}^{\mathbf{0}}$ is used to find the posterior distribution of binary latent variables $p(\boldsymbol{Z}|\boldsymbol{x}, \boldsymbol{\eta}^{\mathbf{0}})$. Then, in the M-step, we find the expectation of complete data log-likelihood with respect to the posterior evaluated at $\boldsymbol{\eta}^{\mathbf{0}}$ as $\mathcal{Q}(\boldsymbol{\eta}, \boldsymbol{\eta}^0)$ and it equals to,

$$\begin{aligned} \mathcal{Q}(\boldsymbol{\eta}, \boldsymbol{\eta}^{\mathbf{0}}) &= \mathbb{E}_{\mathcal{Z}|\mathcal{D}, \boldsymbol{\eta}^{\mathbf{0}}}[\ln p(\mathcal{D}, \mathcal{Z}|\boldsymbol{\eta})] \\ &= \sum_{i=1}^{N} \sum_{Z} p(\boldsymbol{Z}_i|\boldsymbol{x}_i, \boldsymbol{\eta}^{\mathbf{0}}) \ln p(\boldsymbol{x}_i, \boldsymbol{Z}_i|\boldsymbol{\eta}). \end{aligned} \tag{19}$$

Since the likelihood can be expressed as

$$p(\boldsymbol{x_i}, \boldsymbol{Z}_i|\boldsymbol{\eta}) = \prod_{k=1}^{K} \alpha_k^{Z_{ik}} \phi_D(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \Sigma_k)^{Z_{ik}}. \tag{20}$$

Then, the log-likelihood is,

$$\ln p(\boldsymbol{x}_i, \boldsymbol{Z}_i|\boldsymbol{\eta}) = \sum_{k=1}^{K} Z_{ik}[\ln \alpha_k + \ln \phi_D(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \Sigma_k)], \tag{21}$$

and $p(Z_{ik} = 1|\boldsymbol{x}_i, \boldsymbol{\eta}^{\mathbf{0}}) = \gamma_k(\boldsymbol{x}_i)$. Thus, combining Equation (19) and Equation (21),the expectation of complete data loglikelihood $\mathcal{Q}(\boldsymbol{\eta}, \boldsymbol{\eta}^{\mathbf{0}})$ can be expressed as,

$$\mathcal{Q}(\boldsymbol{\eta}, \boldsymbol{\eta}^{\mathbf{0}}) = \sum_{i=1}^{N} \sum_{k=1}^{K} \gamma_k(\boldsymbol{x}_i)[\ln \alpha_k + \ln \phi_D(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \Sigma_k)]. \tag{22}$$

Now, in **maximization step**, the parameter $\boldsymbol{\eta}^*$ can be estimated by maximizing the $\mathcal{Q}$, such that:

$$\boldsymbol{\eta}^* = \arg\max_{\boldsymbol{\eta}} \mathcal{Q}(\boldsymbol{\eta}, \boldsymbol{\eta}^{\mathbf{0}}).$$

Since we need $\sum_{k=1}^{K} \alpha_k = 1$, we introduce a Lagrange multiplier to enforce it, i.e., we consider the optimization of $\mathcal{Q}(\boldsymbol{\eta}, \boldsymbol{\eta}^{\mathbf{0}})$ over $\lambda$,

$$\mathcal{Q}(\boldsymbol{\eta}, \boldsymbol{\eta}^{\mathbf{0}}) = \sum_{i=1}^{N} \sum_{k=1}^{K} \gamma_k(\boldsymbol{x}_i)[\ln \alpha_k + \ln \phi_D(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \Sigma_k)] - \lambda(\sum_{k=1}^{K} \alpha_k - 1). \tag{23}$$

Taking the derivative of $\mathcal{Q}$, with respect to $\alpha_k$ and setting it to zero, we get

$$\begin{aligned} \alpha_k &= \frac{\sum_{i=1}^{N} \gamma_k(\boldsymbol{x}_i)}{N} \\ &= \frac{N_k}{N}, \end{aligned} \tag{24}$$

where we define $N_k := \sum_{i=1}^{N} \gamma_k(\boldsymbol{x}_i)$ is the the total responsibility of the $k^{th}$ mixture component for the observations. Similarly, we take the derivative with respect to $\boldsymbol{\mu}$ and $\Sigma$ and set it to zero, we obtain

$$\begin{aligned} \boldsymbol{\mu}_k^* &= \frac{\sum_{i=1}^{N} \gamma_k(\boldsymbol{x}_i)\boldsymbol{x_i}}{\sum_{i=1}^{N} \gamma(\boldsymbol{x}_i)} \\ &= \frac{\sum_{i=1}^{N} \gamma_k(\boldsymbol{x}_i)\boldsymbol{x}_i}{N_k}; \end{aligned} \tag{25}$$

$$\begin{aligned} \Sigma_k^* &= \frac{\sum_{i=1}^{N} \gamma_k(\boldsymbol{x}_i)(\boldsymbol{x}_i - \boldsymbol{\mu}_k)(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T)}{\sum_{i=1}^{N} \gamma_k(\boldsymbol{x}_i)} \\ &= \frac{\sum_{i=1}^{N} \gamma_k(\boldsymbol{x}_i)(\boldsymbol{x}_i - \boldsymbol{\mu}_k)(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^T)}{N_k}. \end{aligned} \tag{26}$$

Then, given the complete data $(\mathcal{D}, \mathcal{Z})$, we can perform an iteration of the EM algorithm. Although the log-likelihood increase during the iterations of the EM, the EM does not necessarily converge to the global maximum likelihood solution. It is possible for the EM algorithm to converge to a local maximum of the log-likelihood. We could run the EM multiple times and use different initial parameters $\boldsymbol{\eta}$ to reduce the risk of the model ending with a bad local optimum.

## 2.4 Clustering

Clustering is an unsupervised learning technique to divide a set of observations into several groups such that the observations in the same groups are more similar. It is important since it can discover hidden group structures among unlabelled data. The clustering is done by using different criteria such as the distance, density of data points, graph, or various statistical distributions.

### 2.4.1 Model-based clustering approaches

Model-based clustering is a statistical approach to clustering, which assumes that observation is generated from a finite mixture of component models (see in Fraley and Raftery [2002]). The observation within each subgroup follows a multivariate distribution, such as multivariate Gaussian distribution. Then, the subgroup an observation should be assigned to will depend on the parameters(mean and variance matrix for the GMM) of that subgroup's distribution. Besides, the parameter estimates of a subgroup will depend on which observations are assigned to it. We can use the EM algorithm of GMM for clustering since we need to estimate the parameters of each subgroup's distribution and identify each observation's group. Furthermore, the model-based clustering method can automatically identify the optimal number of components or clusters based on information criteria.

In practice, we always perform hard-clustering, that is each observation can only be assigned to one subgroup. After estimating parameters, we will use **maximum a posterior (MAP) rule** to assign the observation. To be more specific, for a observation $\boldsymbol{x}_i$, we would assign it to the subgroup with the highest posterior probability(or called responsibility $\gamma_k(\boldsymbol{x}_i)$ ). Sometimes, if we consider soft-clustering, where observations can be assigned to more than one subgroup, then we will use a score to indicate the degree of association between the observation and the subgroup, and this score for observation $\boldsymbol{x}_i$ corresponding to each cluster $k$ is the posterior probability(or called responsibility) $\gamma_k(\boldsymbol{x}_i)$. In this paper, we will only consider the hard-clustering case.

Suppose there are N observations for clustering, i.e., $\mathcal{D} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)^T \in \mathbb{R}^{N\times D}$,$\boldsymbol{x}_i \in \mathbb{R}^D$, containing $K$ subgroups. The observation in each subgroup $k$ follows the distribution $\mathcal{N}(\boldsymbol{\mu}_k, \Sigma_k)$, then we do the following steps to find the optimal partitions of GMM.

**Step 1** "Guess" the centers $\boldsymbol{\mu}_k^0$ and covariance $\Sigma_k^0$ of K clusters, that is initialize the parameters with $\boldsymbol{\eta}_k^0 = (\boldsymbol{\mu}_k^0, \Sigma_k^0, \alpha_k^0)$, for each cluster $k = 1, \ldots, K$. Then, perform the iteration of EM-algorithm to update the $\boldsymbol{\eta}$ until the loglikelihood converges. The new parameter is denoted as $\boldsymbol{\eta}^* = (\boldsymbol{\mu}_k^*, \Sigma_k^*, \alpha_k^*)$.

**Step 2** For each observation $\boldsymbol{x}_i$, we will allocate the observation by MAP rule. The posterior probability of belonging to each group $k$ is calculated according to Bayes' theorem,

$$\begin{aligned}\gamma_k(\boldsymbol{x}_i) &= p(Z_{ik} = 1|\boldsymbol{X}_i = \boldsymbol{x}_i)\\ &= \frac{p(\boldsymbol{X}_i = \boldsymbol{x}_i|Z_{ik} = 1)p(Z_{ik} = 1)}{p(\boldsymbol{X}_i = \boldsymbol{x}_i)}\\ &= \frac{\alpha_k^*\phi_D(\boldsymbol{x}_i; \boldsymbol{\mu}_k^*, \Sigma_k^*)}{\sum_{j=1}^K \alpha_j\phi_D(\boldsymbol{x}_i; \boldsymbol{\mu}_j^*, \Sigma_j^*)}\end{aligned}$$

and the observation $\boldsymbol{x}_i$ is allocated to the subgroup $\hat{k}$ with the highest posterior probability, that is,

$$\hat{k} = MAP(\gamma_k(\boldsymbol{x}_i)) = \underset{k}{\operatorname{argmax}}\ \{\gamma_1(\boldsymbol{x}_i), \ldots, \gamma_K(\boldsymbol{x}_i)\}.$$

Consider the Example 2, suppose we have known the parameters of the model by EM iterations and we want to infer, given a new data point $x = 2.5$, which component it might belong to. In Example 2, we have estimated the posterior probability of observation $x$ for each group, given by $\gamma_1(x = 2.5)$, $\gamma_2(x = 2.5)$, $\gamma_3(x = 2.5)$, such that

$$\gamma_1(x = 2.5) = p(Z_1 = 1|x = 2.5) = 0.037,$$

$$\gamma_2(x = 2.5) = p(Z_2 = 1|x = 2.5) = 0,$$

$$\gamma_3(x = 2.5) = p(Z_3 = 1|x = 2.5) = 0.963.$$

Since $\gamma_3(x = 2.5) > \gamma_1(x = 2.5) > \gamma_2(x = 2.5)$, we can conclude the data $x$ is more likely to belong to third component.

### 2.4.2 Distance-based clustering method

In distance-based clustering, the observations are clustered by using the distance metric, which is the criteria to determine the similarity between continuous observations. The distance metric can be used to cluster observations by using the distance between each pair of observations or the distance between observations and the center of the subgroup. In contrast to model-based clustering, distance-based clustering has the drawback in that it requires the number of clusters to be known in advance. One of the most common distance-based clustering approaches is kmeans approach.

Considering the problem of finding clusters in a set of observations, we first use an algorithm called **standard kmeans**. To partition the observations into different groups, we first assume the number of clusters $K$ is given. Since each cluster is composed of a set of observations, we can assume that the inter-point distances of these observations are smaller than the distances of the observations outside the cluster. Therefore, we are trying to find a partition $\mathcal{C} = \{C_1, \dots, C_K\}$ that the sum of distance between the data in the group and the corresponding empirical mean $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K)^T$ of the group space( or the clustering) are minimized, where $\boldsymbol{\mu}_k = (\mu_{k1}, \dots, \mu_{kD})$ is the vector of the mean of each variable in the group $k$, where $\mu_{kd} = \frac{1}{n_k}\sum_{i\in\mathcal{C}_k} x_{id}$ and $n_k$ is the number of points in cluster $k$.

Let $\mathcal{D} \in \mathbb{R}^{N\times D}$ be a data matrix containing N observations and each observation has D variables. Suppose $\boldsymbol{x}_i \in \mathbb{R}^D$ and $\boldsymbol{y}_d \in \mathbb{R}^N$ are the $i^{th}$ row and $d^{th}$ column of $\mathcal{D}$. So, $\mathcal{D} = (\boldsymbol{y}_1, \dots, \boldsymbol{y}_D)^T = (\boldsymbol{x}_1^T, \dots, \boldsymbol{x}_N^T) = (x_{id})_{N\times D}$. Then, it could be formulated as an optimization problem that minimize the sum of dissimilarity of each data point to its assigned center, $J(\mathcal{C}, \boldsymbol{\mu})$, that is,

$$\min_{(\mathcal{C},\boldsymbol{\mu})} J(\mathcal{C}, \boldsymbol{\mu}) = \min_{(\mathcal{C},\boldsymbol{\mu})} \sum_{i=1}^{N} \sum_{k=1}^{K} \sum_{\boldsymbol{i}\in C_k} d(\boldsymbol{x_i}, \boldsymbol{\mu}_k), \tag{27}$$

where $d(\cdot, \cdot)$ is called the **dissimilarity function** that satisfy $d(\boldsymbol{m}, \boldsymbol{m}) = 0$,$d(\boldsymbol{m}, \boldsymbol{n}) \geq 0$ and $d(\boldsymbol{m}, \boldsymbol{n}) = d(\boldsymbol{n}, \boldsymbol{m})$, where $\boldsymbol{m}, \boldsymbol{n}$ are random vectors. Here, we choose the square of Euclidean distance as the dissimilarity function, which is $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 = \sum_{q=1}^{p}(x_{iq} - x_{jq})$, for $\boldsymbol{x}_i \in \mathbb{R}^p$.

To solve the problem of choosing the number of clusters $K$ in kmeans, Tibshirani et al. [2001] come up with a method called **Gap statistic**. We define variance quantity $W_K$ as,

$$W_K := \sum_{k=1}^{K} \frac{1}{2n_k} \sum_{i,j\in C_k} d(\boldsymbol{x}_i, \boldsymbol{x}_j), \tag{28}$$

where $K$ is the number of clusters and $n_k$ is the number of points in cluster $k$. The idea of gap statistic is the comparison of the expectation of $\ln(W_K)$ with an appropriate null reference distribution of the dataset and $\ln(W_K)$ of original dataset. The **Gap function** is defined as follow,

$$\text{Gap}(K) := E[\ln(W_K^*)] - \ln(W_K), \tag{29}$$

where $E_n[\ln(W_K^*)]$ is estimated by the empirical mean of B copies $\ln(W_K^*)$ which are generated with a Monte Carlo sample of the reference distribution so we have

$$E[\ln(\hat{W_K^*})] = \frac{1}{B}\sum_{b=1}^{B}\ln(W_{Kb}^*), \tag{30}$$

where $\ln(W_{Kb}^*)$ is derived by clustering the b reference dataset. The simulation error $s_K$ of Monte Carlo simulation can be determined by the standard deviation of $sd(K)$ and it given by

$$s_K = sd(K)\sqrt{1+\frac{1}{B}}, \tag{31}$$

where

$$sd(K) = [\frac{1}{B}\sum_{b=1}^{B}(\ln(W_{Kb}^*) - E[\ln(\hat{W_K^*})])^2]^{\frac{1}{2}}, \tag{32}$$

Hence, after computing each number of clusters $K$, the optimal number of clusters $\hat{K}$ is given by the smallest $K$ that satisfies,

$$\text{Gap}(K) \geq \text{Gap}(K+1) - s_{K+1}. \tag{33}$$

### 2.4.3 Relationship between EM and kmeans

The Gaussian mixture model is very similar to the kmeans clustering algorithm as we will show that the kmeans algorithm is also an EM type algorithm that assigns data points to clusters. To have a better understanding of the relationship between the Gaussian mixture model and kmeans, we introduce a binary variable $Z$ with its realization $z_{ik} \in \{0,1\}$, where $k = 1,\dots,K$, to represent the assigned result. If the data point $\boldsymbol{x_i}$ is belong to the cluster $k$, that is $z_{ik} = 1$ if $\boldsymbol{x_i}$ belongs to cluster $k$ ,$z_{ik} = 0$, otherwise. Then we could rewrite the equation (27) as:

$$\begin{aligned}\min_{(\mathcal{C},\boldsymbol{\mu})} J(\mathcal{C},\boldsymbol{\mu}) &= \min_{(\mathcal{C},\boldsymbol{\mu})}\sum_{i=1}^{N}\sum_{k=1}^{K} z_{ik} d(\boldsymbol{x}_i,\boldsymbol{\mu}_k) \\ &= \min_{(\mathcal{C},\boldsymbol{\mu}}\sum_{i=1}^{N}\sum_{k=1}^{K} z_{ik}\|\boldsymbol{x}_i - \boldsymbol{\mu}_k\|^2.\end{aligned} \tag{34}$$

The kmeans algorithm is an iterative procedure to find out the mean $\boldsymbol{\mu}_k$ and clustering result $z_{ik}$ that minimizes the objective function $J$. It mainly contains two steps. After initializing the value of the mean $\boldsymbol{\mu}_k$ for each group $k$, we firstly minimize the objective function $J$ with respect to $z_{ik}$ by given $\boldsymbol{\mu}_k$. In the second step, we minimize the objective function $J$ with respect to $\boldsymbol{\mu}_k$ by given $z_{ik}$. Then, we repeat these two steps until it converges. During the first step, we assigned the data $\boldsymbol{x}_i$ for $i = 1,\dots,N$ to the closest cluster. The assigned result $z_{ik}$ for data $\boldsymbol{x}_i$ can be estimated by,

$$z_{ik} = \begin{cases} 1 & \text{if} \quad k = argmin_j \|\boldsymbol{x}_i - \boldsymbol{\mu}_j\|^2 \\ 0 & \text{otherwise.} \end{cases} \tag{35}$$

For the second step, fix the cluster result $z_{ik}$ of each data $\boldsymbol{x}_i$, we update the mean $\boldsymbol{\mu}_k$ for $k = 1, \dots, K$ by deriving the objective function $J$. Since $J$ is a quadratic function respect to $\boldsymbol{\mu}_k$, we set the derivative to zero:

$$2\sum_{i=1}^{N} z_{ik}(\boldsymbol{x}_i - \boldsymbol{\mu}_k) = 0 \tag{36}$$

Thus,

$$\boldsymbol{\mu}_k^{new} = \frac{\sum_{i=1}^{N} z_{ik}^{old}\boldsymbol{x}_i}{\sum_{i=1}^{N} z_{ik}^{old}}. \tag{37}$$

Equation (37) is quite similar to the updating Equation (25) of the mean during the EM for the Gaussian mixture model. We can think of kmeans as a special case of GMM, where the mean in GMM is the cluster center and the covariance is set to the identity matrix . As described by MacKay et al. [2003], kmeans assigns data points "hard" to cluster centers, that is each data point is assigned to a specific unique cluster, while GMM makes "soft" assignments through the responsibility measure and does not guarantee assignment to a unique cluster point. The following is a derivation of kmeans as considered a special case of EM for Gaussian mixtures

Suppose a GMM with $K$ components, in which the covariance matrix is $\Sigma_k = \sigma^2\mathbb{I} \in \mathbb{R}^{D\times D}$, $\sigma^2$ is variance and is the same for all $K$ components, the mean of component $k$ is $\boldsymbol{\mu}_k$ for $k = 1, \dots, K$. $\mathbb{I}$ is the identity matrix. Therefore, the density of GMM for component k is that,

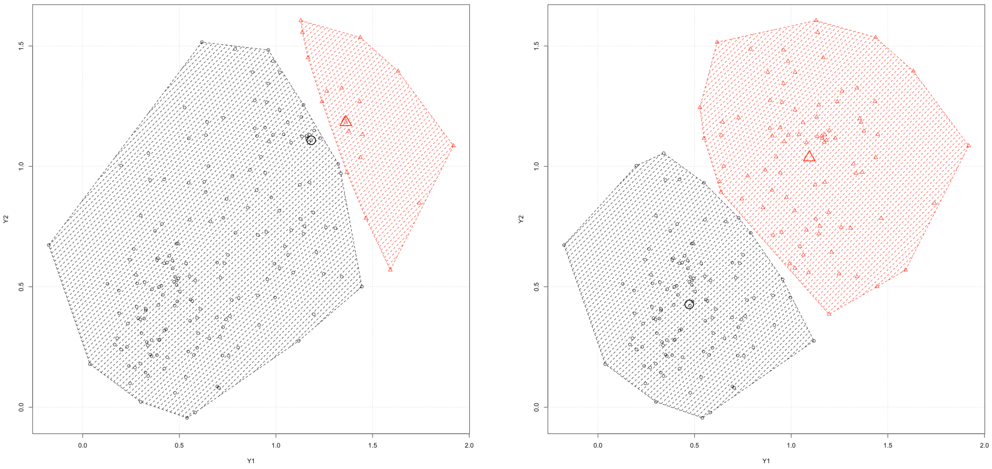
$$f(\boldsymbol{x}_i; \boldsymbol{\mu}_k) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp(-\frac{1}{2\sigma^2}(\boldsymbol{x}_i - \boldsymbol{\mu}_k)^2). \tag{38}$$

Considering the EM algorithm of GMM we discussed before and from the formula of responsibility, as seen in Equation (39), for fixed variance $\sigma^2$, the responsibility of a given data $\boldsymbol{x}_i$ is

$$\gamma_k(\boldsymbol{x}_i) = \frac{\alpha_k exp\{-\|\boldsymbol{x}_i - \boldsymbol{\mu}_k\|^2/2\sigma^2\}}{\sum_{j=1}^{K} \alpha_j exp\{-\|\boldsymbol{x}_i - \boldsymbol{\mu}_j\|^2/2\sigma^2\}}. \tag{39}$$

When $\sigma^2 \to 0$, the term in the denominator for which $\|\boldsymbol{x}_i - \boldsymbol{\mu}_j\|^2$ is smallest will go to zero most slowly, thus the responsibilities $\gamma_k(\boldsymbol{x}_i)$ for data $\boldsymbol{x}_i$ will go to zero except for term $j$, for which the responsibility $\gamma_j(\boldsymbol{x}_i)$ will go to 1. Then $\gamma_k(\boldsymbol{x}_i) \to z_{ik}$, which we defined in Equation (35). Therefore, each data will assigned to only one cluster and we will obtain the hard assignment of data points. Since $\gamma_k(\boldsymbol{x}_i) \to z_{ik}$, the update formula of $\boldsymbol{\mu}_k$ in EM for GMM, which given in Equation (25), is equivalent to the kmeans result in Equation (37). Besides, the re-estimation of mixing coefficient is no longer active due to the hard assignment we discussed above. Thus, the expected complete-data log likelihood will reduce to,

$$\mathbb{E}[\ln p(\mathcal{D}, Z|\boldsymbol{\alpha}, \boldsymbol{\mu}, \Sigma)] \to -\frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{K} z_{ik}\|\boldsymbol{x}_i - \boldsymbol{\mu}_k\|^2 + const. \tag{40}$$



(a) Iteration=1

(b) Iteration=7

Figure 11: Example of two sets of centers in the iterative kmeans process.

Hence, it is obvious that maximizing expected complete-data log-likelihood is equivalent to minimizing the dissimilarity function $J$ in kmeans approaches, defined by Equation (27).

**Example 4.** *We generate the 400 observations from bivariate normal distribution with $Y_1 \sim \mathcal{N}(0.45, 0.2), Y_2 \sim \mathcal{N}(1, 0.3)$. We clustered them using the kmeans algorithm, which consists of 7 iterations. Figure 11 shows the change of centers of two groups in iteration =1 and iteration =7 by using the R package **tryCatchLog**( Altfeld [2021]).*

## 2.5 Copulas

Now we are going to the concepts of the copula. By using the copula, we can handle the multivariate data by modeling the margins separately. To measure the ordinal association between two measured quantities, we will introduce a measure called Kendall's $\tau$ coefficient.

**Definition 17** (Kendall's $\tau$)**.** *The Kendall's $\tau$ between the continuous random variables $X_1$ and $X_2$ is defined as the probability of concordance minus the probability of discordance of two random variables $X_1$ and $X_2$.*

$$\tau(X_1, X_2) = P((X_{11}X_{21})(X_{12}X_{22}) > 0)P((X_{11}X_{21})(X_{12}X_{22}) < 0).$$

*where $(X_{11}, X_{12})$ and $(X_{21}, X_{22})$ are i.i.d distributed copies of $(X_1, X_2)$*

**Definition 18** (Copula and Copula density)**.** *A d-dimensional copula $C$ is a multivariate distribution function $C : [0, 1]^d \to [0, 1]$ with $d$ uniformly distributed random variables,*

$$C(u_1, \dots, u_d) = P(U_1 \leq u_1, \dots, U_d \leq u_d).$$

*The corresponding density, denoted as $c$, can be estimated by partial differentiation, for all $u \in [0, 1]^d$*

$$c(u_1, \dots, u_d) := \frac{\partial^d}{\partial u_1 \dots \partial u_d} C(u_1, \dots, u_d).$$

According to Sklar (1959), there is a fundamental representation theorem for multivariate distributions.

**Theorem 1** (Sklar's theorem)**.** *For a d-dimensional random vector $\boldsymbol{X}$ with joint distribution function $F$ and its marginal distribution functions $F_i$, for $i = 1, \ldots, d$, then we cam express the joint distribution function as*

$$F(x_1, \ldots, x_d) = C(F_1(x_1), \ldots, F_d(x_d)). \tag{41}$$

*and the associated density function is*

$$f(x_1, \ldots, x_d) = c(F_1(x_1), \ldots, F_d(x_d))f_1(x_1) \ldots f_d(x_d). \tag{42}$$

*for some d-dimension copula $C$ with copula density $c$. It also hold for inverse: the copula corresponding to a multivariate distribution function $F$ with the marginal distribution $F_i$, for $i = 1, \ldots, d$ can be expressed as*

$$C(u_1, \ldots, u_d) = F(F_1^{-1}(u_1), \ldots, F_d^{-1}(u_d)). \tag{43}$$

*with the copula density*

$$c(u_1, \ldots, u_d) = \frac{f(F_1^{-1}(u_1), \ldots, F_d^{-1}(u_d))}{f_1(F_1^{-1}(u_1)) \ldots f_d(F_d^{-1}(u_d))}. \tag{44}$$

**Theorem 2** (Kendall's $\tau$ expressed in terms of copula)**.** *Suppose $(X_1, X_2)$ are random variables, then the Kendall's tau of these two variables can be expressed as,*

$$\tau = 4 \int_{[0,1]^2} C(u_1, u_2) dC(u_1, u_2) - 1. \tag{45}$$

### 2.5.1 Bivariate Copula Formulas

Now, we are going to introduce the bivariate copula, which will be applied in our dataset analysis later.

**Copulas from elliptical distributions** We now discuss the copulas derived from elliptical distributions.

**Example 5** (Bivariate Gaussian copula)**.** *The bivariate Gaussian copula can be expressed as*

$$C(u_1, u_2; \rho)\Phi_2(\Phi^{-1}(u_1), \Phi^{-1}(u_2); \rho). \tag{46}$$

*where $\Phi(\cdot)$ is the distribution function of a standard normal $\mathcal{N}(0,1)$; $\Phi(\cdot, \cdot; \rho)$ is the bivariate normal distribution functionwith zero means and unit covariance and correlation $\rho$. Besides, its density is given by,*

$$c(u_1, u_2; \rho) = \frac{1}{\phi(x_1)\phi(x_2)} \frac{1}{\sqrt{1-\rho^2}} exp\{-\frac{\rho^2(x_1^2 + x_2^2) - 2\rho x_1 x_2}{2(1-\rho^2)}\}. \tag{47}$$

**Example 6** (Bivariate Student t copula)**.** *Suppose $t(\cdot,\cdot;\nu,\rho)$ is the density of the bivariate student's t distribution with $\nu$ degrees of freedom, zero mean and the correlation $\rho$, and $T_\nu$ is the univariate student's t distribution function with $\nu$ degree of freedom, zero mean and density $t_\nu$. Then the bivariate student's t copula can be expressed as,*

$$\begin{aligned} C(u_1,u_2;\nu,\rho) &= \int_0^{u_1}\int_0^{u_2} \frac{t(T_\nu^{-1}(v_1),T_\nu^{-1}(v_2);\nu,\rho)}{t_\nu(T_\nu^{-1}(v_1))t_\nu(T_\nu^{-1}(v_2))}dv_1dv_2 \\ &= \int_{-\infty}^{T_\nu^{-1}(u_1)}\int_{-\infty}^{T_\nu^{-1}(u_2)} t(x_1,x_2;\nu,\rho)dx_1dx_2. \end{aligned} \tag{48}$$

*The density is given by*

$$c(u_1,u_2;\nu,\rho) = \frac{t(T_\nu^{-1}(v_1),T_\nu^{-1}(v_2);\nu,\rho)}{t_\nu(T_\nu^{-1}(v_1))t_\nu(T_\nu^{-1}(v_2))}. \tag{49}$$

### Archimedean copula

**Definition 19** (Bivariate Archimedean copula)**.** *Suppose $\Omega$ is the set of all continuous, strictly monotone decreasing, and convex functions $\phi$ such that $[0,1]\to[0,\infty]$ with $\phi(1)=0$. For $\phi\in\Omega$, then*

$$C(u_1,u_2) = \phi^{[-1]}(\phi(u_1)+\phi(u_2)). \tag{50}$$

*is called a bivariate Archimedean copula with generator $\phi$, where $\phi^{[-1]}$ is the pseudo-inverse of $\phi$, which is defined as $\phi^{[-1]}$: $[0,\infty]\to[0,1]$*

$$\phi^{[-1]}(t) := \begin{cases} \phi^{-1}(t), & 0\le t\le\phi(0) \\ 0, & \phi(0)\le t\le\infty. \end{cases} \tag{51}$$

Then the density of the continuous Archimedean copula is given by

$$c(u_1,u_2) = \frac{\partial^2 C(u_1,u_2)}{\partial u_1\partial u_2} = \frac{\phi''(C(u_1,u_2))\phi'(u_1)\phi'(u_2)}{[\phi'(C(u_1,u_2))]^3}. \tag{52}$$

The following are some examples of Archimedean copula

**Example 7** (Clayton copula)**.** *The bivariate Clayton copula can be expressed as*

$$C(u_1,u_2) = (u_1^{-\delta}+u_2^{-\delta}-1)^{-\frac{1}{\delta}}, \tag{53}$$

*where $0<\delta<\infty$ is the dependency parameter.*

**Example 8** (Bivariate Gumbel copula)**.** *The bivariate Gumbel copula can be expressed as*

$$C(u_1,u_2) = exp[-\{(-\ln u_1)^\delta+(-\ln u_2)^\delta\}^{\frac{1}{\delta}}], \tag{54}$$

*where $\delta\ge1$ is the dependency parameter. If $\delta=1$ corresponds to independence.*

**Example 9** (Bivariate Frank copula)**.** *The bivariate Frank copula can be expressed as*

$$C(u_1,u_2) = -\frac{1}{\delta}\ln(\frac{1}{1-e^{-\delta}}[(1-e^{-\delta})-(1-e^{-\delta u_1})(1-e^{-\delta u_2})]), \tag{55}$$

*where $\delta\in[-\infty,+\infty]\backslash0$ is the dependency parameter.*

**Example 10** (Bivariate Joe copula)**.** *The bivariate Joe copula can be expressed as*

$$C(u_1,u_2) = 1-((1-u_1)^\delta+(1-u_2)^\delta-(1-u_1)^\delta(1-u_2)^\delta)^{\frac{1}{\delta}}, \tag{56}$$

*where $\delta\ge1$ is the dependency parameter. If $\delta=1$, the independence copula arises.*

### 2.5.2 Vine copula

**Definition 20** (Graph, node and edge). *A graph is a pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of sets such that $E \subseteq x, y : x, y \in \mathcal{V}$, where $E$ are the edges of the graph $\mathcal{G}$, while $\mathcal{V}$ are nodes. We also denote $d(v)$ as the the number of neighbors of a node $v \in \mathcal{V}$, which is also called the degree of $v$.*

**Definition 21** (Path, cycle). *For a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the path is a sequence of edges $(e_1, e_2, \ldots, e_{n-1})$ for which there is a sequence of nodes $(v_1, v_2, \ldots, v_n)$ such that $e_i = (v_i, v_{i+1})$ for $i = 1, 2, \ldots, n-1$. A cycle is a path $v_1 = v_n$.*

**Definition 22** (Tree). *For a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, if any two nodes of $\mathcal{G}$ are connected by a unique path in $\mathcal{G}$, then we call it a tree.*

**Definition 23** (R-vine tree sequence). *A regular vine tree sequence on $d$ element is the sequence of the tree $\mathcal{T} = (T_1, \ldots, T_{d-1})$ that satisfied*

- *Each tree $T_j = (\mathcal{V}_j, \mathcal{E}_j)$ is connected*
- *$T_1$ is a tree that has node set $\mathcal{V}_1 = 1, \ldots, d$ and edge set $\mathcal{E}_1$.*
- *For $j \geq 2$, $T_j = (\mathcal{V}_j, \mathcal{E}_j)$ is a tree with node set $\mathcal{V}_j = E_{j-1}$.*
- *For all $j = 2, \ldots, d-1$,if $(a, b) \in \mathcal{E}_j$, then it must satisfy $|a \cap b| = 1$.*

**Definition 24** (C-vine tree sequence, D-vine tree sequence). *For a regular vine tree sequence $\mathcal{T} = (T_1, \ldots, T_{d-1})$ is, where $T_i = (\mathcal{V}_i, \mathcal{E}_\rangle)$*

- *D-vine tree sequence if it satisfies $|\{e \in \mathcal{E}_i | v \in e\}| = d - i$ for each node $v \in \mathcal{V}_i$.*

*C-vine tree sequence if for each Tree $T_i$ there is one node $v \in \mathcal{V}_i$ such that —$|\{e \in \mathcal{E}_i | v \in e\}| = d - i$. Such a node is called the root node of tree $T_i$ .*

**Definition 25** (Pair copula). *The copula $C_e$ corresponding to edge $e$ can be denoted as $C_{Ce,aCe,b;De}$ and its density is expressed as $c_{Ce,aCe,b;De}$. $C_e$ can be called as a pair copula.*

**Definition 26** (R-vine distribution). *For the d-dimensional random vector $X = (X_1, \ldots, X_d)$ with joint distribution $F$, it has a regular vine distribution, if we can find a triplet $(\mathcal{F}, \mathcal{V}, \mathcal{B})$ satisfies:*

- *Marginal distributions $\mathcal{F} = (F_1, \ldots, F_d)$ is a vector of continuous invertible marginal distribution functions, which shows the marginal distribution functions of the random variable $X_i$, for $i = 1, \ldots, d$.*
- *Regular vine tree sequence: $\mathcal{T} = (T_1, \ldots, T_{d-1})$ is an R-vine tree sequence on $d$ elements.*
- *Bivariate copulas: $\mathcal{B} = \{C_e | e \in \mathcal{E}_i; i = 1, \ldots, d-1\}$ is a set of copulas, where $\mathcal{E}_i$ is the edge set of tree $T_i$ and $C_e$ is symmetric bivariate copula.*
- *$C_e$ is the copula associated with the conditional distribution of $X_{Ce,a}$ and $X_{Ce,b}$ given $X_{De} = x_{De}$. And it does not dependent on the specific value of $x_{De}$*

### 2.5.3 Vine copula mixture model

In the subsequent clustering analysis, we will use the model-based clustering algorithm based on the vine copula proposed by Sahin and Czado [2022], namely **VCMM**.

The algorithm of **VCMM** consists of roughly the following steps. More details can be seen in Sahin and Czado [2022].

- First, we assign observations into K components using the fast clustering algorithm. Most of our subsequent analysis will be discussed using kmeans as the initial clustering method. In addition to this, we will also use GMM as an initial clustering method for comparison.
- Second, we select an initial VCMM model. The marginal distributions of each variable are determined by model selection criteria and truncate a vine tree structure at tree level one for the initial selection of vine copula models.
- Third, we estimate the parameters via the ECM algorithm allowing for vine tree structures. More detail on the ECM algorithm is provided in Sahin and Czado [2022]. With the ECM algorithm, we constantly update the parameters of our model (incl. mixture weight, pair copula parameters, and marginal parameters), iterating until the stopping condition is met.
- Fourth, temporary clustering assignment: we use the updated posterior probabilities to partition the observations into K clusters and make a temporary clustering assignment.
- Final model selection and clustering assignment based on the full vine specification: we estimate all possible vine tree levels and their parameters, with a temporary clustering assignment obtained in the previous step. Finally, we cluster the observations to the component by posterior probabilities estimated in the final model.

## 2.6 Performance measures for clustering

### 2.6.1 Adjusted Rand Index(ARI)

To compare the clustering performance of different approaches, we will use the **Adjusted Rand Index(ARI)**, which is put forward by Hubert and Arabie [1985]. Before talking about **ARI**, we will first introduce the **Rand Index(RI)**. The **Rand Index(RI)** is based on the idea of comparing two clustering results. The result of measuring two different partitions is to calculate the items that are correctly clustered. **Rand Index(RI)** does not count individual elements, but pairs of identical elements. We can therefore define **Rand Index(RI)** as follows:

**Definition 27** (Rand Index). *For N observations in the set $\mathcal{S} = \{s_1, \dots, s_N\}$, suppose there are two distinct partitions $\mathcal{U} = \{u_1, \dots, u_G\}$, $\mathcal{V} = \{v_1, \dots, v_K\}$, which satisfied $\cup_{i=1}^{G} u_i = \mathcal{S} = \cup_{j=1}^{K} v_j$ and $u_i \cap u_{i'} = \emptyset = v_j \cap v_{j'}$ for all $i \neq i'$, $j \neq j'$. We define the following*

- *a, the number of pairs of elements in $\mathcal{S}$ that are in the **same** subset in U and in the **same** subset in V*

- *$b$, the number of pairs of elements in $\mathcal{S}$ that are in the **different** subset in $U$ and in the **different** subset in $V$*
- *$c$, the number of pairs of elements in $\mathcal{S}$ that are in the **same** subset in $U$ and in the **different** subset in $V$*
- *$d$, the number of pairs of elements in $\mathcal{S}$ that are in the **different** subset in $U$ and in the **same** subset in $V$*

*Then, the **Rand Index(RI)** is given by*

$$\boldsymbol{RI} = \frac{a+b}{a+b+c+d}. \tag{57}$$

*Here **RI** ranges from 0(no pair classified in the same way under both clusterings) to 1(similar clustering). The term $a$ and $b$ can be seen as agreements and $b$, and $c$ as disagreements.*

However, a major problem of **RI** is that the expected value of **Rand Index** of two random partition does not take a constant value. Therefore, to solve this problem, the **Adjusted Rand Index** was introduced, where the generalized hypergeometric distribution is used as the model of randomness. With the consideration of generalized hypergeometric model, **Adjusted Rand Index** is given as the following form,

$$\mathbf{ARI} = \frac{\text{RI} - \mathbb{E}(\text{RI})}{max(\text{RI}) - \mathbb{E}(\text{RI})}.$$

where $max(\text{RI})$ is the maximum possible Rand Index.

**Definition 28** (Adjusted Rand Index(ARI))**.** *According to Yeung and Ruzzo [2001], we can rewrite the formula of **Adjusted Rand Index**. For $N$ observations in the set $\mathcal{S} = \{s_1, \dots, s_N\}$, suppose there are two distinct partitions $\mathcal{U} = \{u_1, \dots, u_G\}$, $\mathcal{V} = \{v_1, \dots, v_K\}$, which satisfied $\cup_{i=1}^{G} u_i = \mathcal{S} = \cup_{j=1}^{K} v_j$ and $u_i \cap u_{i'} = \emptyset = v_j \cap v_{j'}$ for all $i \neq i'$, $j \neq j'$. If $\mathcal{U}$ is the real partition and $\mathcal{V}$ is the clustering result, then we could estimate ARI to compare the two partitions and the formula is as follows:*

$$\boldsymbol{ARI} = \frac{\sum_g^G \sum_k^K \binom{N_{gk}}{2} - [\sum_g^G \binom{N_{g.}}{2} \sum_k^K \binom{N_{.k}}{2}]/\binom{N}{2}}{\frac{1}{2}[\sum_g^G \binom{N_{g.}}{2} + \sum_k^K \binom{N_{.k}}{2}] - [\sum_g^G \binom{N_{g.}}{2} \sum_k^K \binom{N_{.k}}{2}]/\binom{N}{2}}, \tag{58}$$

*where $G$ and $K$ are the number of clusters in two different partitions of the dataset, $\mathcal{U}, \mathcal{V}$; $N_{gk}$ is the number of observations that are in the cluster $g$ of partition $\mathcal{U}$ and cluster $k$ of partition $\mathcal{V}$. $N_{g.} = \sum_{k=1}^{K} N_{gk}$ is the number of observations when lie in the cluster $g$ of partition $\mathcal{U}$ and $N_{.k} = \sum_{K=1}^{G} N_{gk}$ is the number of observation which fall in cluster $K$ of partition $\mathcal{V}$. ARI lies between 0 and 1. The ARI is equal to 1 when two partitions are exactly the same and are close to 0 for a random partition. Later, we will use ARI and assess the performance of the estimated partition with real labels of dataset.*

**Example 11.** *Suppose we have two different partitions and each partition contains 3 clusters, see Table 1*

| True class/ Assigned cluster | $v_1$ | $v_2$ | $v_3$ | Sum |
|---|---|---|---|---|
| $u_1$ | 3 | 1 | 0 | 4 |
| $u_2$ | 1 | 2 | 1 | 4 |
| $u_3$ | 0 | 2 | 4 | 6 |
| Sum | 4 | 5 | 5 | 14 |

Table 1: Example for computing the **ARI**

From the Table 1, we can estimate each term of Equation (58) as follow, since $\binom{14}{2} = 91$ and

$$\sum_g^G \sum_k^K \binom{N_{gk}}{2} = \binom{3}{2} + \binom{1}{2} + \binom{0}{2} + \binom{1}{2} + \binom{2}{2} + \binom{1}{2} + \binom{0}{2} + \binom{2}{2} + \binom{4}{2}$$
$$= 3 + 1 + 1 + 6 = 11$$

The number of pairs of observations in the same class in $\mathcal{U}$:

$$\sum_g^G \binom{N_{g.}}{2} = \binom{4}{2} + \binom{4}{2} + \binom{6}{2} = 6 + 6 + 15$$
$$= 27$$

The number of pairs of observations in the same class in $\mathcal{V}$:

$$\sum_k^K \binom{N_{.k}}{2} = \binom{4}{2} + \binom{5}{2} + \binom{5}{2} = 6 + 10 + 10 = 26$$

And

$$ARI = \frac{11 - (27 \times 26)/91}{\frac{1}{2}(27 + 26) - (27 \times 26)/91} = 0.175$$

Therefore, the agreement between the true classification and the clustering result is only 0.175.

### 2.6.2 Maximum-Match-Measure(MMM)

To explain another performance measure, we first give the following definitions.

**Definition 29** (Bipartite Graph). *A bipartite graph $\mathcal{G} = (V, E)$ is a graph whose vertex set $V$ can be partitioned into to nonempty subsets $A$ and $B$(i.e., $A \cup B = V$ and $A \cap B = \emptyset$) such that each edge of $\mathcal{G}$ has one endpoint in $A$ and one end point in $B$. The partition $V = A \cup B$ is called bipartition of $\mathcal{G}$.*

**Definition 30** (Matching). *Given a bipartite graph $\mathcal{G} = (V, E)$ with bipartition $(A, B)$, a subset of edges $M$ is called a matching in $\mathcal{G}$ if no two edges in $M$ share a common end point in $B$.*

**Definition 31** (Perfect Matching). *A matching $M$ of graph $\mathcal{G}$ is perfect if every vertex is connected to exactly one edge.*

**Definition 32** (Maximum Weight Matching)**.** *Given a bipartite graph $\mathcal{G} = (V, E)$ with bipartition $(A, B)$ and weight function $w : \mathbb{E} \to \mathbb{R}$ find a matching of maximum weight where the weight of matching $M$ is given by $w(M) = \sum_{e \in M} w(e)$*

**Example 12.** *Suppose we have 5 observations in the set $\mathcal{S} = (x_1, x_2, \ldots, , x_5)$ and the true clustering of them are $C_1^{true} = \{x_1, x_2, x_3\}$, $C_2^{true} = \{x_4, x_5\}$. The assigned clustering contains 3 components that given by, $C_1^{ass} = \{x_1\}$, $C_2^{ass} = \{x_2, x_3\}$ and $C_1^{ass} = \{x_4, x_5\}$, then we can compute the weight of each pairs as follow by counting the number of points they have in common, i.e.,*

$$w(C_1^{true}, C_1^{ass}) = 1, w(C_1^{true}, C_2^{ass}) = 2, w(C_1^{true}, C_3^{ass}) = 0$$

$$w(C_2^{true}, C_1^{ass}) = 0, w(C_2^{true}, C_2^{ass}) = 0, w(C_2^{true}, C_3^{ass}) = 2$$

*Therefore, by the definition of maximum weight of perfect matching, we will assign the $C_1^{true} \leftarrow C_2^{ass}$ and $C_2^{true} \leftarrow C_3^{ass}$.*

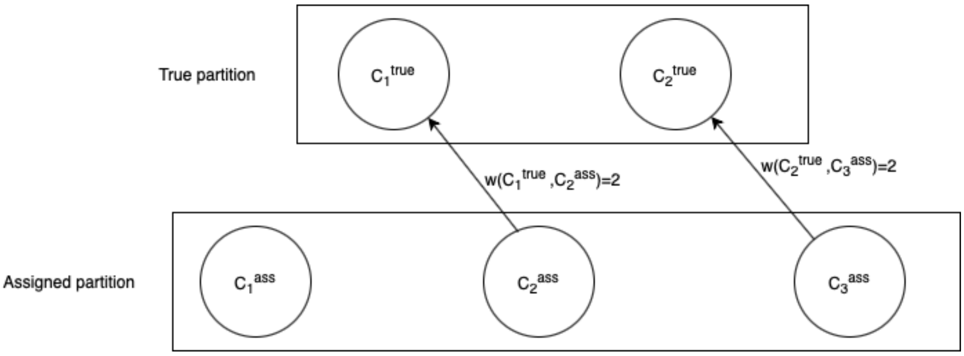


Figure 12: The bipartite of Example 12.

The **Maximum Matching Measure(MMM)**, described in Wagner and Wagner [2007], is motivated by the above idea, i.e., finding a feasible pair of the true partitions and clusters that make the maximum weight matching, which is matching that the sum of the weights of its edges is maximized. Besides, we also require this match should be perfect, that is only one cluster can be assigned to a known partition.

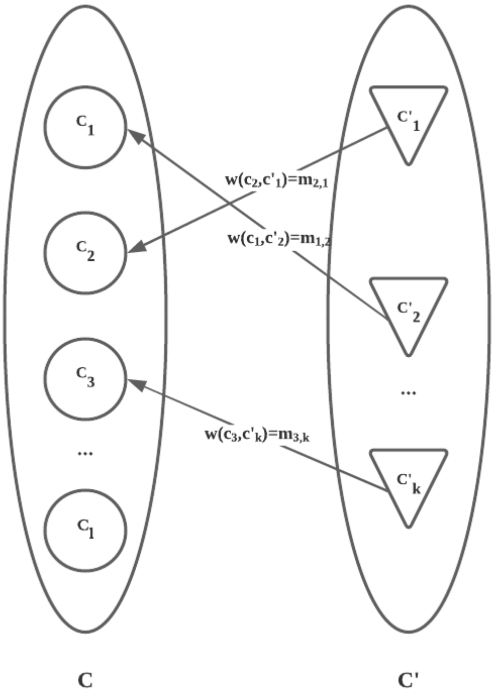


Figure 13: The bipartite of Maximum-Match-Measure(MMM), where $\boldsymbol{C}$ is the true partitions that contains $l$ components and $\boldsymbol{C'}$ is the clustering results with k clusters; the weight $w(C_i, C'_j)$ equal to the number of common points denoted by $m_{i,j}$.

Formally, we can treat it as a complete weighted bipartite graph $\mathcal{G} = (V, E)$, as shown in Figure 13. Suppose each edge $e$ in the graph $\mathcal{G}(V, E)$ connects two vertices, where each partition and cluster is a node, such that V= $C \cup C'$. The edge $e = (C_i, C'_j)$ represent the cluster $C'_j$ is assigned to the true partition $C_i$ with the weight $w(C_i, C'_j)$ equal to the number of common elements $m_{i,j}$. A perfect matching M in $\mathcal{G}$ is a subset of E, such that the edges $e$ in M do not have a common vertex. The perfect maximum weighted match M is the subset of edge set E that satisfy $w(M') = \sum_{e \in M'} w(e) \leq w(M) = \sum_{e \in M} w(e)$, where M' is any perfect match in $\mathcal{G}$ created by random assignment of vertices, $w(e)$ is the number of common point that in $C_i$ and $C'_j$. Hence, the maximum match measure is trying to find the hard assignment that maximize the sum of the number of common elements in true partition and assigned partition. Then, the maximum match of the clustering could be defined as :

$$match(C, C') = \arg\max_{M} \frac{w(M)}{n},$$

where $w(M) = \sum_{e \in M} w(e)$. The maximum matching measure is calculated as follows

$$MMM(C, C') = \frac{1}{n} \sum_{i}^{min\{G,K\}} w(C_i, C'_{i'}) = \frac{1}{n} \sum_{i}^{min\{G,K\}} m_{i,i'}, \tag{59}$$

where $n$ is the total number of the observations, $m_{i,i'}$ is the number of common points in the partition $C_i$ and the cluster $C'_{i'}$ where cluster $C'_{i'}$ is assigned to true partition $C_i$, $G$ is the number of clusters and $K$ is the number of class we have. When $K = K$, the maximum-match-measure is equal to $classificationaccuracy = \sum_i^K m_{i,i'}$. If $G \neq K$, This measure completely ignores the $|K - G|$ rest cluster in "higher cardinality" clustering.

**Example 13.** *Suppose there are two sets, one is the partition of dataset $\mathcal{U} = \{u_1, u_2, u_3\}$ and the other is the clustering result $\mathcal{V} = \{v_1, v_2, v_3, v_4\}$. The relationship between these two partitions is shown in Table 2*

| True class/Assigned cluster | $v_1$ | $v_2$ | $v_3$ | $v_4$ |
|---|---|---|---|---|
| $u_1$ | 3 | 1 | 0 | 1 |
| $u_2$ | 1 | 2 | 1 | 0 |
| $u_3$ | 0 | 2 | 4 | 1 |

Table 2: Example of MMM

Then, by the definition of maximum-match-measure, we will first assign $v_3 \to u_3$ since 4 is the largest number of common points. Then, we delete the $v_3's$ column and $u_3's$ row. The common points between $v_1$ and $u_2$ now becomes the largest so we assign $v_1$ to $u_1$. Same as before, we delete first column and first row of the common points and the rest is just $v_2's$ column and $u_2's$ row. Finally, we assign $v_2$ to $u_2$. That is $v_1 \to u_1$, $v_2 \to u_2$, $v_3 \to u_3$. Hence, the MMM can be estimated by Equation (59), that is $\frac{3+2+4}{3+1+0+1+1+2+1+0+0+2+4+1} = 0.56$.

## 2.7 Variables selection for clustering

Generally, we can use all variables for clustering. However, in many cases, considering all variables will increase the complexity of the model and also may cause inaccurate clustering. Variables can be broadly classified into three types: relevant variable, irrelevant variable, and redundant variable. **Relevant variable** is the variable whose information is useful for clustering. **Irrelevant variable** is variables that do not convey any beneficial information. And **Redundant variable** is the variable that information of it is already contained in another relevant variable for clustering. Some irrelevant and redundant variables correspond to noise and their distribution is completely independent of the group structure. Therefore, variable selection will keep relevant variables for clustering and removes redundant and irrelevant variables.

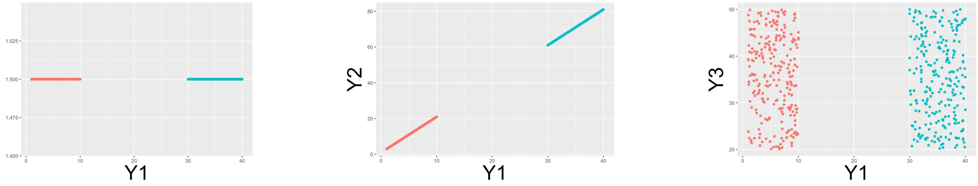


Figure 14: The left graph is the relevant variable $\mathrm{Y}_1$, the middle graph shows the redundant variable $\mathrm{Y}_2$ and the right graph shows the irrelevant variable $\mathrm{Y}_3$ for clustering, where red and blue color represents the different labels of observations.

**Example 14.** *For binary clustering, suppose there is a dataset with each data is $\mathcal{D} = (\boldsymbol{y}_1, \boldsymbol{y}_2, \boldsymbol{y}_3) \in \mathbb{R}^{N\times 3}$, which is a "N" i.i.d observations with variables vector $\mathcal{Y} = (Y_1, Y_2, Y_3)$.*

*As Figure 14 shows, $Y_1$ is relevant since it is able to discriminate two classes, see in the left plot . However given $Y_1$, $Y_2$ is redundant as $Y_2$ is perfectly correlated with $Y_1$, see in the middle plot. $Y_3$ is irrelevant since it cannot separate the two classes, see in the right plot. Therefore, removal of $Y_2$ and $Y_3$ will not negatively impact the clustering performance in the dataset.*

The approaches for variable selection in clustering can be broadly distinguished according to the type of statistical approach used. Three major approach are the **Bayesian approach**, **penalization approach** and **model selection-approach**. In general, most of the variable selection approaches have some degree of overlap. Here, we will focus on the **penalization approaches** and **model-selection approaches**.

### 2.7.1 Penalization approach

In this part the penalization term is introduced on the model parameters. We aim to maximize the penalized log likelihood under a Gaussian mixture model and remove the variables those parameter estimates are shrunken to 0. The general form of penalized loglikelihood is given by

$$L_{\mathcal{Q}}(\boldsymbol{\eta};\mathcal{D}) = \sum_{i=1}^{N} \ln(\sum_{k=1}^{K} \alpha_k \phi_D(\boldsymbol{x}_i;\boldsymbol{\psi}_k)) - \mathcal{Q}_\lambda(\Theta). \tag{60}$$

where the penalization term $\mathcal{Q}_\lambda(\Theta)$ is a function of the Gaussian densities parameter $\Theta$ and $\lambda$, a generic penalty parameter. The various methods are differentiated by the form of function $\mathcal{Q}_\lambda(\cdot)$. For example, Pan and Shen [2007] suggest a $L_1$ penalty function $\lambda \sum_1^K \sum_{d=1}^D \|\mu_{kd}\|$; Wang and Zhu [2008] replaces the $L_1$ norm to $L_\infty$ norm such that $\lambda \sum_{d=1}^D max_k\{\|\mu_{1d}\|,\dots,|\mu_{Kd}\|\}$.

Now, we are going to introduce a method that differs from the general form of penalized loglikelihood Equation (60). For clustering, we always rewrite the objection function $J$, in Equation (27), by using the **Within-Cluster Sum of Square(WCSS)** to minimize the dissimilarity of the data in the same cluster. Suppose $\boldsymbol{x_i} \in \mathbb{R}^D$ and $\mathcal{D} = (\boldsymbol{y_1},\dots,\boldsymbol{y_d}) = (\boldsymbol{x_1^T},\dots,\boldsymbol{x_N^T}) = (x_{id})_{N\times D}$. Then, **Within-Cluster Sum of Square(WCSS)** can be expressed as

$$WCSS = \sum_{k=1}^{K} \sum_{i\in C_k} \sum_{d=1}^{D} (x_{id} - \mu_{kd})^2, \tag{61}$$

where $\mu_{kd} = \frac{1}{n_k}\sum_{i\in\mathcal{C}_k} x_{id}$ is the mean of the variable $Y_d$ for all observations that belong to cluster $k$, where $n_k$ is the number of observations in cluster $k$, $\mathcal{C}_k$ is the set that contains the index of observations that belongs to cluster $k$.

However, in many practical examples, it is more convenient to consider **Between-Cluster Sum of Squares (BCSS)**, which is defined by

$$BCSS = \sum_{d=1}^{D} [\sum_{i=1}^{N} (x_{id} - \mu_d)^2 - \sum_{k=1}^{K} \sum_{i\in C_k} (x_{id} - \mu_{kd})^2], \tag{62}$$

where $\mu_d = \frac{1}{N}\sum_{i=1}^{N} x_{id}$ is the mean of variable $Y_d$ for all observations in the dataset. The second term of function is within cluster sum of square. Hence, minimizing distortion measure $J$ or minimizing within cluster sum of square (WCSS) is equivalent to maximizing Between-Cluster Sum of Squares (BCSS).

Considering the selection of variables, Witten and Tibshirani [2010] suggested sparse kmeans procedure, which generalized the BCSS to the optimization problem. Suppose $\boldsymbol{x_i} \in \mathbb{R}^D$ and $\boldsymbol{y_d} \in \mathbb{R}^N$ are the $i^{th}$ row and $d^{th}$ column of $\mathcal{D}$. So, the observation data matrix $\mathcal{D} = (\boldsymbol{y_1}, \ldots, \boldsymbol{y_D}) = (\boldsymbol{x_1^T}, \ldots, \boldsymbol{x_N^T}) = (x_{nd})_{N\times D}$. Let $\mathcal{C} = \{C_1, \ldots, C_K\}$ be a partition of the observations into $K$ disjoint subgroups, and $\boldsymbol{w} = (w_1, ..., w_d, ..., w_D)^T$ be a vector of weights for each variable $Y_d$, for $d = 1, \ldots, D$. Then we need to solve the following optimization problem to select the variables and cluster our data.

$$\begin{array}{ll} \underset{\mathcal{C},\boldsymbol{w},\boldsymbol{\mu}}{\operatorname{argmax}} & \{\sum_{d=1}^{D} w_d[\sum_{i=1}^{N}(x_{id}-\mu_d)^2 - \sum_{k=1}^{K}\sum_{i\in C_k}(x_{id}-\mu_{kd})^2]\} \\ \text{subject to} & \|\boldsymbol{w}\|^2 \leq 1,\ |\boldsymbol{w}| \leq s, \\ \text{with} & w_d \geq 0\ \forall d. \end{array} \tag{63}$$

where $x_{id}$ is observation $i$ on variable $Y_d$, $\mu_{kd}$ is the mean of variable $d$ in group $k$, $\mu_d$ is the sample mean of variable $Y_d$ and $s$ is a tuning parameter. This approach optimizes the weighted between-cluster sum of squares subject to constraints on the weight of variables. The $L_1$ penalty of $\boldsymbol{w}$ results in sparsity for small values of tuning parameters and some of weights will equal to zero. The constraint of the $L_2$ penalty on $\boldsymbol{w}$ will guarantee the weights of variables in the interval [0,1]. If $w_d = 0$, the variable $Y_d$ can be removed from the clustering process since it does not affect clustering. Furthermore, the penalization kmeans approach needs the number of clusters $K$ to be known in advance.

To solve the optimization problem, we first fix $\boldsymbol{w}$, and optimize the objective function with respect to $\mathcal{C}$. After that, we fix $\mathcal{C}$ with respect to $\boldsymbol{w}$. The first step can be solved by applying the standard kmeans approach. For the second step, we rewrite the convex problem given in Equation (63) as

$$\begin{array}{ll} \max\limits_{\boldsymbol{w}} & \{\boldsymbol{w}^T\boldsymbol{a}\} \\ \text{subject to} & \|\boldsymbol{w}\|^2 \leq 1,\ |\boldsymbol{w}| \leq s, \\ \text{with} & w_d \geq 0\ \forall d. \\ \text{where} & \boldsymbol{a} = (a_1, \ldots, a_D)^T, \text{with } a_d = \sum_{n=1}^{N}(x_{nd}-\mu_d)^2 - \sum_{k=1}^{K}\sum_{n\in C_k}(x_{nd}-\mu_{kd})^2 \quad \forall d. \end{array} \tag{64}$$

To solve the convex problem, we review some results from convex optimization Boyd et al. [2004]. For a minimization problem,

$$\begin{array}{ll} \min\limits_{\boldsymbol{x}} & f_0(\boldsymbol{x}) \\ \text{subject to} & f_i(\boldsymbol{x}) \leq 0,\ i = 1, \ldots, m \\ & h_i(\boldsymbol{x}) = 0,\ i = 1, \ldots, p \end{array} \tag{65}$$

with variable vector $\boldsymbol{x} \in \mathbb{R}^n$, then the Lagrangian function is defined as:

$$L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{v}) = f_0(\boldsymbol{x}) + \sum_{i=1}^{m}\lambda_i f_i(\boldsymbol{x}) + \sum_{i=1}^{p} v_i h_i(\boldsymbol{x}), \tag{66}$$

where $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^T$, $\boldsymbol{v} = (v_1, \ldots, v_p)^T$ and the Lagrange dual function is:

$$g(\boldsymbol{\lambda}, \boldsymbol{v}) = \min_{\boldsymbol{x}} L(\boldsymbol{x}, \lambda, \boldsymbol{v}). \tag{67}$$

The corresponding dual problem is:

$$\begin{aligned} &\max_{\lambda, \boldsymbol{v}} && g(\lambda, \boldsymbol{v}) \\ &\text{subject to} && \boldsymbol{\lambda} \geq 0 \end{aligned} \tag{68}$$

For primal and dual problems, they satisfy the weak duality such that:

$$f_0^* \geq g^* \tag{69}$$

When Slater's condition Slater [2014] is satisfied, then $f_0^* = g^*$ holds, where $f_0^*$ and $g^*$ are primal and dual optimal values.

Suppose $\hat{\boldsymbol{x}}$ is a primal optimal and $(\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{v}})$ is dual optimal point then

$$\begin{aligned} f_0(\boldsymbol{x}^*) &= g(\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{v}}) \\ &= \inf_{\boldsymbol{x}} \left(f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \hat{\lambda}_i f_i(\boldsymbol{x}) + \sum_{i=1}^{p} \hat{v}_i h_i(\boldsymbol{x})\right) \\ &\leq f_0(\hat{\boldsymbol{x}}) + \sum_{i=1}^{m} \hat{\lambda}_i f_i(\hat{\boldsymbol{x}}) + \sum_{i=1}^{p} \hat{v}_i h_i(\hat{\boldsymbol{x}}) \\ &\leq f_0(\hat{\boldsymbol{x}}) \end{aligned} \tag{70}$$

The second line is derived by the definition of dual problem. The last inequality follows from the assumption that $\hat{\lambda}_i \geq 0, f_i(\hat{\boldsymbol{x}}) \leq 0$ and $h_i(\hat{\boldsymbol{x}}) = 0$ for $i = 1, \ldots, m$. And it is obvious that the last two inequalities hold with equality. Therefore, we can conclude that

$$\sum_{i=1}^{m} \hat{\lambda}_i f_i(\hat{\boldsymbol{x}}) = 0, \tag{71}$$

and since each term is non-positive, we have

$$\lambda_i f_i(\hat{\boldsymbol{x}}) = 0 \text{ for } i = 1, \ldots, m. \tag{72}$$

Therefore, for a general problem in Equation (65), if $f_i$ are convex and $h_i$ are affine, $\hat{\boldsymbol{x}}$,$\hat{\lambda}$,$\hat{v}$ are any point that satisfy the Karush-Kuhn-Tucker conditions(KKT conditions):

$$\begin{aligned} & f_i(\hat{\boldsymbol{x}}) \leq 0, i = 1, \ldots, m \\ & h_i(\hat{\boldsymbol{x}}) = 0, i = 1, \ldots, p \\ & \hat{\lambda}_i \geq 0, i = 1 \ldots, m \\ & \hat{\lambda}_i f_i(\hat{\boldsymbol{x}}) = 0, i = 1 \ldots, m \\ & \nabla f_0(\hat{\boldsymbol{x}}) + \sum_{i=1}^{m} \hat{\lambda}_i \nabla f_i(\hat{\boldsymbol{x}}) + \sum_{i=1}^{p} \hat{v}_i \nabla h_i(\hat{\boldsymbol{x}}) = 0 \end{aligned} \tag{73}$$

Then $\widetilde{\boldsymbol{x}}$ and $(\widetilde{\lambda}, \widetilde{v})$ are primal and dual optimal.

**Proposition 1.** *By the water filling example in the book Boyd et al. [2004] which is derived if those Karush-Kuhn-Tucker conditions are satisfied, the convex problem, given in Equation (64), has the solution*

$$\boldsymbol{w} = \frac{S(\boldsymbol{a}, \Delta)}{\|S(\boldsymbol{a}, \Delta)\|_2}, \tag{74}$$

*where $\Delta = 0$ if $|\boldsymbol{w}| \leq s$; otherwise, $\Delta > 0$ is chosen so that $|\boldsymbol{w}| = s$. $S$ is* ***soft-thresholding function*** *with $S(\boldsymbol{a}, \Delta) = max(\boldsymbol{a} - \Delta, 0)$.*

To illustrate the range of tuning parameters we used above, we consider the following proposition below.

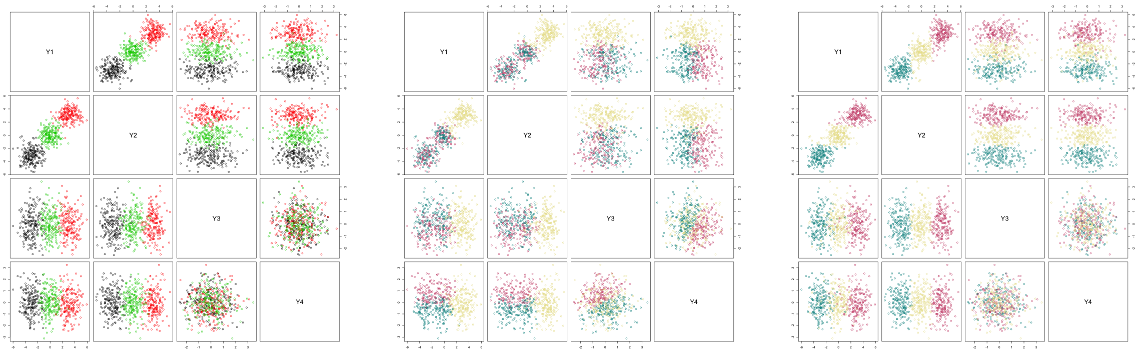
**Proposition 2.** *For the vector $\boldsymbol{w} \in \mathbb{R}^D$, $L_q$ penalty of $\boldsymbol{w}$ must satisfies the following inequalities,*

$$D^{\frac{1}{2}-\frac{1}{q}}\|\boldsymbol{w}\|_q \leq \|\boldsymbol{w}\|_2 \leq \|\boldsymbol{w}\|_q.$$

Therefore, for the $s$ in the convex problem in Equation (64), it must be in the range $[1, \sqrt{D}]$. If $s > \sqrt{D}$, the $L_1$ constrain will be inactive and it will lead to the solution $\boldsymbol{w}^* = \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|^2}$, which none of them will equal to zero. If $0 < s \leq 1$, the $L_2$ constraint will be inactive and the solution $\boldsymbol{w}^*$ will only have one non-zero element. For example, if $\boldsymbol{w}$ is two-dimension vector and $0 < s \leq 1$, then the optimal solution of optimization problem in Equation (64) will be $\boldsymbol{w} = (0, s)$ or $\boldsymbol{w} = (s, 0)$. Therefore,the tuning parameter must satisfies $1 < s < \sqrt{D}$. According to the optimization problem above, the sparse kmeans clustering maximized the objective function can be summarized by the following steps:

```
Algorithm 1 Variable selection using a sparse kmeans approach
Input: observations: $\mathcal{D} = (\boldsymbol{x_1}, \dots, \boldsymbol{x_N}) \in \mathbb{R}^{N\times D}$, where $\boldsymbol{x_i} = (x_{i1}, \dots, x_{iD})^T$; K: number
    of clusters; $\boldsymbol{w}$:weight vector of variables; $\mu_{kd}$ is the mean of the variable $Y_d$ for all
    observations that belong to cluster $k$.
Output: optimal variable set; clusters $\mathcal{C}$
1: Initialize $\boldsymbol{w}$ as $w_1 = w_2 = \cdots = w_D = \frac{1}{\sqrt{D}}$.
2: Keeping $\boldsymbol{w}$ fixed, optimize the following formula with respect to $\mathcal{C} = \{C_1, \dots, C_K\}$ :

       $$\min_{\mathcal{C}}(\sum_{k=1}^{K}\sum_{i\in C_k}\sum_{d=1}^{D} w_d(x_{id} - \mu_{kd})^2)$$

   by the standard kmeans algorithm.
3: Keeping $\mathcal{C} = \{C_1, \dots, C_K\}$ fixed, optimize the convex problem in Equation (64) with
   respect to weight $\boldsymbol{w}$. The weight $\boldsymbol{w}$ can be updated by the Equation (74).
4: Iterate 2) and 3) until the change of weight $\boldsymbol{w}$ converge , that is

       $$\frac{\sum_{d=1}^{D}|w_d^{new} - w_d^{old}|}{\sum_{d=1}^{D}|w_d^{old}|} < 10^{-4}.$$

   where $w_d^{new}$ and $w_d^{old}$ are the weight of variable $Y_d$ estimated in the past and current
   iteration respectively.
```

**Example 15.** *We generate 600 observations from a 4-dimensional multivariate normal distribution with covariance matrix equal to identity matrix. For each observation, it contains four variables, denoted as $Y_1$, $Y_2$, $Y_3$, $Y_4$. There are three classes of the data and the label of observations are defined by the first two variables $Y_1$, $Y_2$. The mean vector of each cluster are* $(-3,-3,0,0),(0,0,0,0),(3,3,0,0)$.



(a) initial data with true class

(b) kmeans

(c) sparse kmeans with s=1.1

Figure 15: Comparison of clustering partition of kmeans and sparse kmeans.

In Example 15, if we set $s = 1.1$, the algorithm will choose only first two variables, $\mathrm{Y}_1, \mathrm{Y}_2$, for clustering and the weight $\boldsymbol{w} = (0.650, 0.760, 0.00, 0.00)$. If we choose $s = 2$, then the result will change to choosing 4 variables to cluster the dataset and the weight vector of 4 variables are $\boldsymbol{w} = (0.70, 0.713, 0.002, 0.004)$. We can see that even though the results contain four variables, $\mathrm{Y}_3$, $\mathrm{Y}_4$ are given very low weights. Figure 15 shows the difference between the result of standard kmeans with 4 variables,$\mathrm{Y}_1, \mathrm{Y}_2, \mathrm{Y}_3\mathrm{Y}_4$, and the result of the sparse kmeans with only two clustering variables, $\mathrm{Y}_1$ and $\mathrm{Y}_2$. From the pairs plot figure of initial observations, seen in Figure 15(a), we can find that the observations are distinguished by the variable $\mathrm{Y}_1$ and $\mathrm{Y}_2$. However, since the standard kmeans cannot select irrelevant variables, the clustering result is negatively influenced by some variables. For sparse kmeans, it only chooses the relevant variables and therefore, it gets good partitions.

To consider the selection of the value of tuning parameters $s$, we cannot directly choose the $s$ that maximizes the objective function. Since if $s$ increases, the value of the objective function will increase. Therefore, for choosing the tuning parameter $s$, Witten and Tibshirani [2010] use a permutation approach that is similar to the gap statistic Tibshirani et al. [2001] for choosing the number of clusters in standard kmeans, see in Section 2.4.2. We randomly and independently permute the original dataset, and then the variables in the permuted dataset are uncorrelated, even though they are previously highly correlated.

The algorithm for selecting tuning parameters contains the following steps:

- Obtained permuted the datasets $\mathcal{D}_1, \mathcal{D}_2 \ldots, \mathcal{D}_B \in \mathbb{R}^{N\times D}$ by randomly and independently permuting the observations within the variables.
- Determine the objective function of the sparse kmeans approach by the sparse kmeans algorithm with tuning parameter $s$ and the original data $\mathcal{D}$, given by

$O(s) = \sum_{d=1}^{D} w_d[\sum_{i=1}^{N}(x_{id} - \mu_d)^2 - \sum_{k=1}^{K}\sum_{i\in C_k}(x_{id} - \mu_{kd})^2]$, and the objective functions obtained by performing sparse kmeans with tuning parameters $s$ on each dataset $\mathcal{D}_1, \mathcal{D}_1 \ldots, \mathcal{D}_B$, that is $O_b(s)$, for $b = 1, \ldots, B$.

- Calculate $Gap(s) = \ln(O(s)) - \frac{1}{B}\sum_{b=1}^{B}\ln(O_b(s))$.
- Choose $\hat{s}$ that make the value of $Gap(s)$ be largest.

If the group structure in the original dataset is supported by several correlated variables, then permutation will destroy this support. Here, the *Gap* function is a measure of strength between the clustering result on the real dataset and the permuted dataset that has no meaningful clustering structure. This approach of parameter selection allows the optimal parameter $s$ to be chosen as the value where the gap is greatest. However, this approach is not the best algorithm to choose the value of $s$.

There can be some problems with using permutations of the original dataset to eliminate clustering structures. If the values of a variable itself play an important role in clustering, they will still be strongly clustered in the permuted dataset. Besides, if there are a small number of variables, this may still lead to meaningful clustering, so permutation may not eliminate all clustering structures. In Example 15, by executing this algorithm, we obtain $s = 1.4$, which will select all four variables for clustering, although the weights of the last two variables are very close to zero.

### 2.7.2 Information criteria-based approaches

**Variable selection based on BIC** Information criteria are often used for model selection in clustering. One of the most popular criterion is the **Bayesian Information Criterion(BIC)**. Consider the model family $\mathcal{F} = \{(K, m) \in \{2, \ldots, K_{max}\} \times \mathcal{M}\}$, where $\mathcal{M}$ is the set of 28 different Gaussian mixture models that available in **MIXMOD** software (Langrognet et al. [2020]) and $K$ is the total number of clusters. Now, we are trying to find the model that maximize the posterior probability $p((K, m)|\mathcal{D})$ and by Bayes theorem, we can update prior probabilities of some model $(K, M)$, given by $(p(K, m))$, to posterior probabilities after observing $\mathcal{D}$ by accounting for the probabilities of observing $\mathcal{D}$ given the model, written as $f(\mathcal{D}|K, m)$, and $f(\mathcal{D})$ is the marginal density of the observations

$$p((K, m)|\mathcal{D}) = \frac{f(\mathcal{D}|K, m)p((K, m))}{f(\mathcal{D})}. \tag{75}$$

Then, the best model we select is $(\hat{K}, \hat{m}) = \underset{(K,m)\in\mathcal{F}}{\text{argmax}} f(\mathcal{D}|K, m)$, where the integrated likelihood is defined as

$$f(\mathcal{D}|K, m) = \int f(\mathcal{D}|K, m, \boldsymbol{\eta})\pi(\boldsymbol{\eta}|K, m)\, d\boldsymbol{\eta}, \tag{76}$$

where $\boldsymbol{\eta}$ is the parameter vector of the model $(K, m)$ and $\pi(\boldsymbol{\eta}|K, m)$ is its prior distribution. Since the integrated likelihood is difficult to calculate, we will use BIC to choose the optimal $(K, m)$, which is the approximation of $-2\ln f(\mathcal{D}|K, m)$.

**Definition 33** (Bayesian Information Criterion(BIC))**.** *For N observations,* ***Bayesian Information Criterion(BIC)*** *can be defined as:*

$$BIC_{clust}(\mathcal{D}|K, m) = -2\ln f(\mathcal{D}|K, m, \hat{\boldsymbol{\eta}}) + q\ln(N), \tag{77}$$

*and the model we select is*

$$(\hat{K}, \hat{m}) = \underset{(K,m)\in\mathcal{F}}{argmaxBIC_{clust}}(\mathcal{D}|K, m), \tag{78}$$

*where q is the number of parameters in model (K,m) and the vector of all parameters of model computed by the EM algorithm is expressed as $\widehat{\boldsymbol{\eta}}$, $f(\mathcal{D}|K, m, \widehat{\boldsymbol{\eta}})$ is the maximum likelihood under this model. Then, the best model that makes the BIC minimum is selected.*

As Fop et al. [2018] mentioned, there is a model selection approach proposed Maugis et al. [2009] based on the Bayesian information criterion (BIC). For a dataset $\mathcal{D} = (\boldsymbol{y_1}, \ldots, \boldsymbol{y_D})^T$ with variables $Y_1, \ldots, Y_D$ we consider to separate the variables set $\mathcal{Y}$ into the following three subgroups: $\mathcal{Y}^S$, $y^P$,$\mathcal{Y}^{NS}$, where $\mathcal{Y}^S$ is the subset of variables that already selected and will be considered in the clustering model, $y^P$ is the candidate variable that has not yet been decided whether to add to the model, and $\mathcal{Y}^{NS}$ is the set of remaining variables. Then, by comparing the BIC of the models with and without the $y^P$, we can decide whether to add the variable $y^P$ to the set of variables for clustering and obtain the clustering model which has the lowest BIC.

More specifically, by comparing the following two models, it is possible to decide whether to add or remove the variable $y^P$ from the set of clustering variables.

**Model A** We consider model A, denoted as $M_A$, that now contains the clustering variable set $\mathcal{Y}^S$ and the variable set $\mathcal{Y}^{NS}$ is supposed to be independent of the clustering but dependent or independent of the clustering variables $\mathcal{Y}^S$ and $y^P$. For variable $y^P$, it can be explained by the subset of the current clustering variable set $\mathcal{Y}^S$ and has no additional information for clustering. Hence, $y^P$ is independent of the clustering but depends on the relevant variables through regression equations. Then, the integrated likelihood of Model A is

$$\begin{aligned} f_A(\mathcal{D}|K, m) &= f_A(\mathcal{Y}^{NS}, y^P, \mathcal{Y}^S|K, m) \\ &= \sum_{\boldsymbol{z}} f_A(\mathcal{Y}^{NS}, y^P, \mathcal{Y}^S|\boldsymbol{z}, K, m) f_A(\boldsymbol{z}|K, m) \\ &= f_A(\mathcal{Y}^{NS}|y^P, \mathcal{Y}^S) f_A(y^P|\mathcal{Y}^S) \sum_{\boldsymbol{z}} f_A(\mathcal{Y}^S|\boldsymbol{z}, K, m) f_A(\boldsymbol{z}|K, m) \\ &= f_A(\mathcal{Y}^{NS}|y^P, \mathcal{Y}^S) f_{\text{reg}}(y^P|\mathcal{Y}^R \subseteq \mathcal{Y}^S) f_{\text{clust}}(\mathcal{Y}^S|K, m), \end{aligned}$$

where $\mathcal{Y}^R$ is the subset of current clustering variables set $\mathcal{Y}^S$, which have linear relationship with $y^P$ and is allowed to be empty set. $\boldsymbol{\eta}$ is the vector of all unknown parameters of model $M_A$; $\boldsymbol{z} = (\boldsymbol{z_1}, \ldots, \boldsymbol{z_n})$ consists out of binary vectors $\boldsymbol{z_i} = (z_{i1}, \ldots, z_{iK})^T$ such that $z_{ik} = \{0, 1\}$ , which indicate the partition of observations. $f_{\text{reg}}(y^P|\mathcal{Y}^R \subseteq \mathcal{Y}^S)$ is the multidimensional regression integrated likelihood and $f_{\text{clust}}(\mathcal{Y}^S|K, m)$ is the mixture integrated likelihood.

**Model B** We suppose model B, denoted as $M_B$, not only contains the current clustering variable set $\mathcal{Y}^S$ but also contains the variable $y^P$, which will useful for clustering. Then, the integrated likelihood of model B is

$$\begin{aligned} f_B(\mathcal{D}|K,m) &= f_B(\mathcal{Y}^{NS}, y^P, \mathcal{Y}^S|K,m) \\ &= \sum_{\boldsymbol{z}} f_B(\mathcal{Y}^{NS}, y^P, \mathcal{Y}^S|\boldsymbol{z}, \boldsymbol{\eta}) f_B(\boldsymbol{z}|K,m) \\ &= f_B(\mathcal{Y}^{NS}|y^P, \mathcal{Y}^S) \sum_{\boldsymbol{z}} f_B(y^P, \mathcal{Y}^S|\boldsymbol{z}, K, m) f_B(\boldsymbol{z}|K,m) \\ &= f_B(\mathcal{Y}^{NS}|y^P, \mathcal{Y}^S) f_{\text{clust}}(y^P, \mathcal{Y}^S|K,m). \end{aligned}$$

To choose one of two models based on the observed dataset $\mathcal{D}$, we can use the **Bayes factor**, which is a likelihood ratio of the marginal likelihood of two hypotheses, to make the model comparison.

**Definition 34.** *Suppose we have two models $M_1$ and $M_2$ with parameter vector $\boldsymbol{\Theta_1}$ and $\boldsymbol{\Theta_2}$ respectively, Then the Bayes factor $\mathbb{B}_{M_1M_2}$ of model $M_1$ and $M_2$ is defined as the ratio quantifying the relative probability of observed data under each of two models, such that*

$$\begin{aligned} \mathbb{B}_{M_1M_2} &= \frac{f(D|M_1)}{f(D|M_2)} \\ &= \frac{\int f(\boldsymbol{\Theta_1}|M_1) f(D|\boldsymbol{\Theta_1}, M_1) d\boldsymbol{\Theta_1}}{\int f(\boldsymbol{\Theta_2}|M_2) f(D|\boldsymbol{\Theta_2}, M_2) d\boldsymbol{\Theta_2}}. \end{aligned} \tag{79}$$

If the two model have same prior probability, i.e.,$f(\boldsymbol{\Theta_1}|M_1) = f(\boldsymbol{\Theta_2}|M_2)$, the Bayes factor is equal to the ratio of posterior probability of $M_1$ and $M_2$. If $\mathbb{B}_{M_1M_2} > 1$, it means $M_1$ is more strongly supported by the observation, compared to model $M_2$; if $\mathbb{B}_{M_1M_2} < 1$, there is no enough evidence to support the model $M_1$ and thus we will choose the $M_2$.

Since $f_A(\mathcal{Y}^{NS}|y^P, \mathcal{Y}^S) = f_B(\mathcal{Y}^{NS}|y^P, \mathcal{Y}^S)$ by the definition, then according to Bayes factor $\mathbb{B}_{M_AM_B}$ for model $M_A$ against model $M_B$, we can choose the model with high strength evidence, i.e., the Bayes factor is

$$\begin{aligned} \mathbb{B}_{M_AM_B} &= \frac{f_A(\mathcal{D}|K,m)}{f_B(\mathcal{D}|K,m)} \\ &= \frac{f_{\text{reg}}(y^P|\mathcal{Y}^R \subseteq \mathcal{Y}^S) f_{\text{clust}}(\mathcal{Y}^S|K,m)}{f_{\text{clust}}(y^P, \mathcal{Y}^S|K,m)}. \end{aligned} \tag{80}$$

As we discussed before, the integrated likelihood is hard to compute, we prefer using the approximation of $2 \ln \mathbb{B}_{M_AM_B}$ and the model selection problem is reduced to a comparison of the model's BIC, that is

$$BIC_{\text{diff}}(y^P) = BIC_{\text{clust}}(\mathcal{Y}^S, y^P|K,m) - [BIC_{\text{clust}}(\mathcal{Y}^S|K,m) + BIC_{\text{reg}}(y^P|\mathcal{Y}^R \subseteq \mathcal{Y}^S)]. \tag{81}$$

The decision to add variable $y^P$ to the current variable set $\mathcal{Y}^S$ for the model does now depend on the difference between the BIC of the model $M$ and $M_B$. Hence, the variable $y^P$ is added to the model if the Equation (81) is less than 0; otherwise, remove it.

$BIC_{\text{clust}}(\mathcal{Y}^S, y^P|K,m)$ is the BIC of $M_B$, which is GMM model contains the $\mathcal{Y}^S$ and new variable $y^P$. $BIC_{\text{clust}}(\mathcal{Y}^S|K,m)$ is the BIC of GMM model contains only the current variable set $\mathcal{Y}^S$ and $BIC_{\text{reg}}(y^P|\mathcal{Y}^R \subseteq \mathcal{Y}^S)$ is BIC of linear regression model of $y^P$, given $\mathcal{Y}^R$. Therefore, the combination of last two terms is the BIC of the model $M_A$.

Then, by searching through all variables, we can finally get the optimal subset of variables and do the clustering by the selected variables set and EM algorithm. For the searching step, we have two options, that is forward and backward search. To consider the effects of all variables simultaneously, we will use the backward search for our later study. Therefore, the following algorithm shows the variable selection with backward direction search and it consists of the exclusion and inclusion steps. These two steps are based on the BIC approximation of the Bayes factor, which we derived above.

**Algorithm 2** Variable selection based on BIC criterion with backward direction search

```
Input: Original variable set $\mathcal{Y}$; dataset $\mathcal{D} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_D) \in \mathbb{R}^{N \times D}$; the index of the variable set $\mathcal{S} = \{1, \ldots, D\}$; the index
    of the excluded/included variables $i_E/i_I$; number of clusters $K$
Output: Optimal variable set;
initialize $\mathcal{S} = \{1, \ldots, D\}$, $i_E = \emptyset$, $i_I = \emptyset$;
Exclusion Step: $\forall i \in \mathcal{S}$, compute $\text{BIC}_{\text{diff}}(y_i)$ in Equation (81) and

    $$i_E = \underset{i \in \mathcal{S}}{\text{argmax}} \text{BIC}_{\text{diff}}(y_i)$$

if $\text{BIC}_{\text{diff}}(y_{i_E}) \geq 0$ then
    $\mathcal{S} = \mathcal{S} \backslash \{i_E\}$
    if $i_E = i_I$ then
        stop
    else
        go to inclusion step
    end if
else
    if $i_I = \emptyset$ then
        stop
    else
        go to inclusion step
    end if
end if
Inclusion Step: $\forall i \in \mathcal{S}^c$, where $\mathcal{S}^c$ is the complementary set of $\mathcal{S}$, compute $\text{BIC}_{\text{diff}}(y_i)$ and

    $$i_I = \underset{i \in \mathcal{S}^c}{\text{argmin}} BIC_{\text{diff}}(y_i)$$

if $\text{BIC}_{\text{diff}}(y_{i_I}) < 0$ then
    if $i_E = i_I$ then
        stop
    else
        $\mathcal{S} = \mathcal{S} \cup i_I$ and go to the exclusion step
    end if
else
    go to exclusion step
end if
```

**Variable selection based on the maximum integrated likelihood of the complete data(MICL) criterion** Assume $\mathcal{D} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_D) = (\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_N^T) = (x_{id})_{N \times D}$ is a data matrix in $\mathbb{R}^{N \times D}$. We consider a binary indicator vector $\boldsymbol{z} = (\boldsymbol{z_1}, \ldots, \boldsymbol{z_N})^T$, $\boldsymbol{z_i} = (z_{i1}, \ldots, z_{iK})$ indicates the class label of the observation $\boldsymbol{x}_i$, i.e., $z_{ik} = 1$ if $\boldsymbol{x}_i$ belongs to component k and $z_{ik} = 0$ otherwise. Then, according to Marbac and Sedki [2019], **integrated likelihood of the complete data** $(\mathcal{D}, Z)$**(ICL)**(observed data and latent variable) could be preferred to the **integrated likelihood**(only observed data) since it doesn't require heavy parameter estimation, where the **ICL** is defined as

$$f(\mathcal{D},\boldsymbol{z}|\boldsymbol{m}) = \int f(\mathcal{D},\boldsymbol{z}|\boldsymbol{m},\boldsymbol{\eta})\pi(\boldsymbol{\eta}|\boldsymbol{m})d\boldsymbol{\eta}. \tag{82}$$

And

$$f(\mathcal{D},\boldsymbol{z}|\boldsymbol{m},\boldsymbol{\eta}) = \prod_{i=1}^{N} f(\boldsymbol{x_i},\boldsymbol{z_i}|\boldsymbol{m},\boldsymbol{\eta});$$

$$f(\boldsymbol{x_i},\boldsymbol{z_i}|\boldsymbol{m},\boldsymbol{\eta}) = \prod_{k=1}^{K}[\alpha_k\phi_D(\boldsymbol{x_i};\boldsymbol{\psi_k})]^{z_{ik}},$$

where a model $\boldsymbol{m} = (k,\omega)$ is defined by a number of components $k$ and the binary vector $\boldsymbol{\omega} = (\omega_1,\ldots,\omega_D)$ such that $\omega_d \in \{0,1\}$, which encode whether the variables are relevant, i.e., if the variable $Y_d$ is relevant, then $\omega_d = 1$, otherwise $\omega_d = 0$. $\boldsymbol{\eta} = (\boldsymbol{\mu},\Sigma,\boldsymbol{\alpha})$ are all parameters to be estimated and $\pi(\boldsymbol{\eta}|\boldsymbol{m})$ is the prior distribution of $\boldsymbol{\eta}$, $\boldsymbol{\mu} = (\mu_{kd}; k = 1,\ldots,K; d = 1,\ldots,D)$ are the means of all variables split by clusters, $\Sigma_k \in \mathbb{R}^{d\times d}$ is the covariance matrices in cluster $k$, $\boldsymbol{\alpha} = (\alpha_1,\ldots,\alpha_K)$ is the vector of the mixing proportion.

After assuming that relevant variables are independent within clusters and irrelevant variables are independent of relevant variables and placing conjugate priors, the integrated completed data likelihood Equation (82) can be reduced to

$$f(\mathcal{D},\boldsymbol{z}|\boldsymbol{m}) = f(\boldsymbol{z}|k)\prod_{d=1}^{D} f(\boldsymbol{x_{\cdot,d}}|k,\omega_d,\boldsymbol{z}), \tag{83}$$

where $\boldsymbol{x_{\cdot,d}} = \{x_{id}; i = 1,\ldots,n\}$

But if we directly use the **ICL** criteria: $ICL(\boldsymbol{m}) = \ln f(\mathcal{D},\hat{\boldsymbol{z}}|\boldsymbol{m})$ and EM algorithm, to carry out the variable selection, it would be difficult to compute. Here $\hat{\boldsymbol{z}}$ is the clustering result estimated by MAP rule (discussed in Section 2.4.1) at maximum likelihood estimate of model's parameter $\hat{\boldsymbol{\eta}} = (\hat{\boldsymbol{\mu}},\hat{\Sigma},\hat{\boldsymbol{\alpha}})$, i.e.,

$$\hat{z}_{ik} = \begin{cases} 1 & if\ k = \underset{k=1,\ldots,K}{\operatorname{argmax}}\ \hat{\alpha}_k\prod_{d=1}^{D}\phi_D(x_{id};\hat{\mu}_{kd},\hat{\Sigma}_{kd}) \\ 0 & \text{otherwise.} \end{cases}$$

This approach has a heavy computational workload due to the estimation of the partition of $\hat{\boldsymbol{z}}$. Therefore, the approach carried out by Marbac and Sedki [2019] is based on a new criterion called **MICL**. MICL criterion is based on the **maximum value of integrated complete-data likelihood** among all the possible clustering result $\boldsymbol{z}$. MICL is expressed as,

$$\text{MICL}(\boldsymbol{m}) = \ln f(\mathcal{D},\hat{\boldsymbol{z}}|\boldsymbol{m}), with\ \hat{\boldsymbol{z}} = \underset{z}{\operatorname{argmax}}\ln f(\mathcal{D},\boldsymbol{z}|\boldsymbol{m}). \tag{84}$$

Compared with the ICL criterion, it does not require the maximum likelihood estimate, making it less computationally difficult. And we select the model that maximizes the MICL among all the possible model sets. In particular, for specific cluster $k$, the algorithm iterates in two steps, one is **partition step**, that is optimize the partition $\boldsymbol{z}$ by given the observation $\mathcal{D}$ and the model $\boldsymbol{m} = (k,\omega)$. Then, the next step is a **maximization step** that maximizes over $\boldsymbol{m}$, given the observation $\mathcal{D}$ and the partition $\boldsymbol{z}$ that we calculated at previous step. Therefore, we could select the optimal model $\hat{\boldsymbol{m}}_k$ for number of cluster $K = k$ by

$$\hat{\boldsymbol{m}}_k = \underset{\boldsymbol{m}\in\mathcal{M}_k}{\operatorname{argmax}}\text{MICL}(\boldsymbol{m}),$$

with $\mathcal{M}_k = \{(k, \omega) : \omega \in \{0, 1\}^d\}$. Finally, to determine the optimal number of clusters $\hat{k}$, we running the algorithm from $k = 1$ to $k = K_{\max}$ and get the best model, which has the largest MICL. The algorithm is shown below.

**Algorithm 3** Variable selection based on MICL criterion

```
Input: Dataset $\mathcal{D} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_D) \in \mathbb{R}^{N\times D}$; Maximum number of clusters $K_{max}$;
Output: Optimal model $\hat{\boldsymbol{m}} = (k, \boldsymbol{\omega})$ containing the indicator of variable (i.e., if the
    variable $Y_d$ is relevant, then $\omega_d = 1$, otherwise $\omega_d = 0$); $\boldsymbol{\omega}$ and optimal number of
    clusters $k$; Clustering result of observations $\boldsymbol{z}$;
1: For $k = 1, \ldots, K_{\max}$, we iterate the following steps until the integrated completed-data
   likelihood converge and get the optimal model, denoted by $\hat{\boldsymbol{m}}_k$
     a Optimize $\boldsymbol{z}^{[t]}$ such that
         $\ln f(\boldsymbol{D}, \boldsymbol{z}^{[t]}|\boldsymbol{m}_k^{[t]}) \geq \ln f(\boldsymbol{D}, \boldsymbol{z}^{[t-1]}|\boldsymbol{m}_k^{[t]})$.
     b Optimize model $\boldsymbol{m}_k^{[t+1]}$ that satisfy $\boldsymbol{m}_k^{[t+1]} = \underset{\boldsymbol{m}\in\mathcal{M}_k}{\operatorname{argmax}} \ln f(\boldsymbol{D}, \boldsymbol{z}^{[t]}|\boldsymbol{m})$, such that
         $\boldsymbol{m}_k^{[t+1]} = (k, \boldsymbol{\omega}^{[t+1]})$ where $\omega_d^{[t+1]} = \underset{\omega_d\in\{0,1\}}{\operatorname{argmax}} f(x_{\cdot d}|k, \omega_d, \boldsymbol{z}^{[t]})$.
2: For all possible number of clusters $k = 1, \ldots, K_{\max}$, select the model with maximum
   MICL such that,
         $\hat{\boldsymbol{m}} = \underset{k=1,\ldots,K_{\max}}{\operatorname{argmax}} \text{MICL}(\hat{\boldsymbol{m}}_k) = \underset{k=1,\ldots,K_{\max}}{\operatorname{argmax}} \ln f(\mathcal{D}, \boldsymbol{z}|\hat{\boldsymbol{m}}_k)$.
```

### 2.7.3 Hybrid filter-wrapper approaches

**Hybrid filter-wrapper approach based on within-group variance** We discuss now a hybrid filter-wrapper algorithm that proposed by Andrews and McNicholas [2014]. The idea of this approach is based on finding variables that not only minimize the within-group variance but also maximize the between-group variance. For the observation data $\mathcal{D} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_D) = (\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_N^T) = (x_{id})_{N\times D} \in \mathbb{R}^{N\times D}$, within-group variance of the variable $Y_d$ is given by:

$$\mathcal{W}_d = \frac{\sum_{k=1}^{K}\sum_{i=1}^{N} z_{ik}(x_{id} - \mu_d)^2}{N}, \tag{85}$$

where $\mu_d = \frac{1}{N}\sum_{i=1}^{N} x_{id}$ is the mean of variable $Y_d$ for all observations in the dataset; binary indicator vector $\boldsymbol{z} = (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N)^T$, $\boldsymbol{z}_i = (z_{i1}, \ldots, z_{iK})$ indicates the class label of the observation $x_i$, i.e., $z_{ik} = 1$ if $x_i$ belongs to component k and $z_{ik} = 0$ otherwise.

Then, the variance of between-group for variable $Y_d$ can be expressed by the variance of variable $Y_d$, $\sigma_d^2$, as $\sigma_d^2 - \mathcal{W}_d$. As the variance between groups needs to be taken into account, to simplify the problem, we standardized the data so that the variance of each variable is the same with variance equal to 1 and mean equal to 0, then minimizing the within-group variance is equivalent to maximizing the between-group variance.

Suppose we just select a variable from the remaining variable set by using a simple threshold process. For example, by sorting $\mathcal{W} = \{\mathcal{W}_1, \dots, \mathcal{W}_D\}$ in ascending order and selecting those variables $Y_d$ for which $\mathcal{W}_d$ is less than some value $w$, where all correlations between the variable $Y_d$ and the variable $Y_r$ in the selected set of variables are also less than some value $c$. It seems to be a viable approach. However, our need is to first consider the within-group variance and then the correlations between every two variables. Therefore, we will use a sliding correlation threshold to express the relationship between the within-group variance and the correlation so that is more forgiving for small values of $\mathcal{W}_d$ and more strict for larger values.

The algorithm of variable selection based on within-group variance is given as follows, we first calculate the within-group variance of each variable by Equation (85), $\mathcal{W}_d$, $\forall d = 1, \dots, D$, and sort them by ascending order. Then, we choose the first one to the selected variable set $\mathcal{H}$ and for the later variable, $Y_d$, we will add variable $Y_d$ to the variable set for clustering if

$$|\rho_{dr}| \leq 1 - \mathcal{W}_d, \ \forall Y_r \in \mathcal{H}, \tag{86}$$

where $\rho_{rd} = \frac{\sum_i(\boldsymbol{y}_{id}-\mu_d)(\boldsymbol{y}_{ir}-\mu_r)}{\sqrt{\sum_i(\boldsymbol{y}_{id}-\mu_d)^2 \sum_i(\boldsymbol{y}_{ir}-\mu_r)^2}}$ is the empirical correlation between the variable $Y_r$ in the selected variable set and variable $Y_d$ that proposed to be added or removed. Hence, the algorithm can be expressed as,

**Algorithm 4** A hybrid filter-wrapper approach based on within-group variance

```
Input: Original variable set; Dataset $\mathcal{D} = (\boldsymbol{y}_1, \dots, \boldsymbol{y}_D) \in \mathbb{R}^{N \times D}$
Output: Optimal variable set
Calculate within-group variance $\mathcal{W}_d$ by Formula (85);
Sort $\mathcal{W}_d$ in ascending group;
Initial selected variable set is $\mathcal{H} = \{Y_s\}$, where $Y_s$ is the variable that $\mathcal{W}_s$ is the minimal.
for $d \leq D$ do
    Compute correlation between $Y_d$ and $Y_r$, denote as $\rho_{dr}$, $\forall Y_r \in \mathcal{H}$ and $d \neq r$;
    if $|\rho_{dr}| \leq 1 - \mathcal{W}_d, \forall Y_r \in \mathcal{H}$ then
        Variable $Y_d$ is selected to variable set such that
            $\mathcal{H} = \mathcal{H} \cup Y_d$
    end if
    $d = d + 1$;
end for
```

However, Andrews and McNicholas [2014] suggests that the linear relationship in Equation (86) may be too strong. For example, if a variable with within-group variation $\mathcal{W} = 0.3$ and the correlation with one of the variables in the selected variable set is 0.71, it would be removed from the clustering variable set. Therefore, we also need to consider the relationship of order greater than one, see Table 3. Since we use different

| | |
|---|---|
| Linear | $\lvert\rho_{dr}\rvert \leq 1 - \mathcal{W}_d$ |
| Quadratic | $\lvert\rho_{dr}\rvert \leq 1 - \mathcal{W}_d^2$ |
| Cubic | $\lvert\rho_{dr}\rvert \leq 1 - \mathcal{W}_d^3$ |
| Quartic | $\lvert\rho_{dr}\rvert \leq 1 - \mathcal{W}_d^4$ |
| Quintic | $\lvert\rho_{dr}\rvert \leq 1 - \mathcal{W}_d^5$ |

Table 3: Relationship between variance and correlation

variance-correlation relationships in the Table 3 as criteria, we will get results for up to five different variable choice sets. To determine the final clustering variables set, we need an approach to select one of these subsets for clustering. We will therefore present an approach based on estimating the **uncertainty** of the clustering results.

The uncertainty of the clustering result is the sum of uncertainty of each observations. The uncertainty of each observations can be estimated by the $N \times K$ fuzzy matrix containing the posterior probability $\gamma_k(\boldsymbol{x}_i)$. $\gamma_k(\boldsymbol{x}_i)$ evaluates the strength of evidence that observation $\boldsymbol{x}_i$ belongs to cluster $k$. If an observation $\boldsymbol{x}_i$ is clustered perfectly, all values in the row $i$ will be almost equal to 0 except for only one of $\gamma_k(\boldsymbol{x}_i)$ which will be close to 1. Thus, for this observation $\boldsymbol{x}_i$, the uncertainty is equal to the sum of all $\gamma_k(\boldsymbol{x}_i)$ without the highest one $\max_k\{\gamma_k(\boldsymbol{x}_i)\}$. For the uncertainty of clustering result, we can estimate it by the sum of all $\gamma_k(\boldsymbol{x}_i)$ for $i = 1, \ldots, N, k = 1, \ldots, K$, without the $\max_k\{\gamma_k(\boldsymbol{x}_i)\}$ for $i = 1, \ldots, N =$, that is $\sum_{i=1}^{N}\sum_{k=1}^{K}\gamma_k(\boldsymbol{x}_i) - \sum_{i=1}^{N}\max_k\{\gamma_k(\boldsymbol{x}_i)\}$, can be rewrite as

$$\text{Uncertainty} = N - \sum_{i=1}^{N}\max_k\{\gamma_k(\boldsymbol{x}_i)\}. \tag{87}$$

We will select variance-correlation and its corresponding subset of variables such that uncertainty is minimized so that we can obtain the strongest group structure of variables.

**Hybrid filter-wrapper approach based on lasso-like procedure** Celeux et al. [2019] proposed a hybrid approach that performs variable selection through penalization. This will reduce the dimension of data and reduce computational difficulty when we use a BIC-based variable selection approach. It contains two steps.

First of all, according to Zhou et al. [2009], use the lasso-like procedure to rank the variables. For the data matrix $\mathcal{D} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_D) = (\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_N^T) = (x_{id})_{N\times D} \in \mathbb{R}^{N\times D}$ with K clusters, we need to maximize the criterion called **lasso-loglikelihood**:

$$\sum_{i=1}^{N}\ln[\sum_{k=1}^{K}\alpha_k\phi_D(\bar{\boldsymbol{x}}_i|\boldsymbol{\mu}_k, \Sigma_k)] - \lambda_1\sum_{k=1}^{K}\|\boldsymbol{\mu}_k\|_1 - \lambda_2\left\|\Sigma_k^{-1}\right\|_1, \tag{88}$$

where

$$\|\boldsymbol{\mu}_k\|_1 = \sum_{d=1}^{D}|\mu_{kd}|, \ \left\|\Sigma_k^{-1}\right\|_1 = \sum_{d_1,d_2=1,d_1\neq d_2}\left|(\Sigma_k^{-1})_{d_1,d_2}\right|_1$$

and $\bar{\boldsymbol{x}}_i$ is normalized data such that $\bar{\boldsymbol{x}}_i = (x_{i1} - \mu_1, \ldots, x_{iD} - \mu_D)$ with $\mu_d = \frac{1}{N}\sum_{i=1}^{N}x_{id}$, for $d = 1, \ldots, D$; $\boldsymbol{\mu}_k \in \mathbb{R}^D$ are the means vector of each cluster $k$; $\Sigma_k \in \mathbb{R}^{D\times D}$ is the covariance matrices of the cluster k; $\alpha_k$ is the mixing proportion, $\alpha_k \geq 0$, $\sum_{k=1}^{K}\alpha_k = 1$, for $k = 1, \ldots, K$. $\lambda_1, \lambda_2$ are the penalization parameters.

Then, by the EM algorithm, the estimated parameter $\hat{\boldsymbol{\eta}}$ with regularization parameters $\lambda_1$ and $\lambda_2$ can be computed, which is of the form

$$\hat{\boldsymbol{\eta}}(\lambda_1,\lambda_2) = (\hat{\boldsymbol{\alpha}}(\lambda_1,\lambda_2), \hat{\boldsymbol{\mu}}_1(\lambda_1,\lambda_2), \ldots, \hat{\boldsymbol{\mu}}_K(\lambda_1,\lambda_2), \hat{\Sigma}_1(\lambda_1,\lambda_2), \ldots, \hat{\Sigma}_K(\lambda_1,\lambda_2)).$$

Considering the selection of relevant variables, we need to define a score that measures the relevance of variables for clustering. Celeux et al. [2019] mention that if the $\hat{\mu}_{kd}(\lambda_1,\lambda_2) = 0$ for all $k = 1,\ldots,K$, the variable $Y_d$ is independent for clustering and will be removed from variable set. Therefore, by choosing specified number of cluster K and the different combination of two non-negative regularization parameters $(\lambda_1,\lambda_2)$ from two sets $\mathcal{L}_{\lambda_1}$ and $\mathcal{L}_{\lambda_2}$, the clustering score for each variables $Y_d$, for $d = 1,\ldots,D$ with number of cluster K is defined by:

$$\mathcal{O}_K(Y_d) = \sum_{(\lambda_1,\lambda_2)\in\mathcal{L}_{\lambda_1}\times\mathcal{L}_{\lambda_2}} \mathcal{B}_{(K,\lambda_1,\lambda_2)}(Y_d), \tag{89}$$

where

$$\mathcal{B}_{(K,\lambda_1,\lambda_2)}(Y_d) = \begin{cases} 0 & if\ \hat{\mu}_{1d}(\lambda_1,\lambda_2) = \cdots = \hat{\mu}_{Kd}(\lambda_1,\lambda_2) = 0 \\ 1 & \text{otherwise.} \end{cases}$$

$\mathcal{O}_K(Y_d)$ is the sum of scores among all possible regularization parameters. A larger value of $\mathcal{O}_K(Y_d)$ indicates that the variable is more relevant to the clustering and we prefer to select it. And we sort them in descending order, such that,

$$\mathcal{I}_K = (Y_1,\ldots,Y_D) \quad \text{with } \mathcal{O}_k(Y_1) > \cdots > \mathcal{O}_k(Y_d) \cdots > \mathcal{O}_k(Y_D).$$

Secondly, to perform the variable selection, we use variable selection relies on BIC. By Equation (81), we scanned the variables by the order we got at the first step, denoted as $\mathcal{I}_K$, and put the variables to the clustering variable set until $c$ consecutive variables have nonnegative $BIC_{\text{diff}}$, where $c$ is a fixed positive number. For variables that are independent of clustering, we search through the reversed order of $\mathcal{I}_K$ and move the variable to the irrelevant variable set until $c$ consecutive variables are judged to be dependent on clustering. Then, the remaining variables are redundant variables for clustering.

**Algorithm 5** A hybrid approach based on the lasso-like procedure

```
Input: Original variable set $\mathcal{Y} = (Y_1, \dots, Y_D)$; Number of clusters K; Possible regularization parameters set $\mathcal{L}_{\lambda_1}$,$\mathcal{L}_{\lambda_2}$;
Output: Optimal variable set;
For all combination of regularization parameters $(\lambda_1, \lambda_2) \in \mathcal{L}_{\lambda_1} \times \mathcal{L}_{\lambda_2}$, estimate the mixture parameter $\boldsymbol{\eta}$ by using EM algorithm for the problem of maximizing the lasso-loglikelihood function $g$ in Equation (88) and iterate until the lasso-loglikelihood converge , where
    $\boldsymbol{\eta}(\lambda_1, \lambda_2) = (\boldsymbol{\alpha}(\lambda_1, \lambda_2), \boldsymbol{\mu}_1(\lambda_1, \lambda_2), \dots, \boldsymbol{\mu}_K(\lambda_1, \lambda_2), \Sigma_1(\lambda_1, \lambda_2), \dots, \Sigma_K(\lambda_1, \lambda_2)).$
Calculate the score of each variables $Y_d$, denoted as $\mathcal{O}_K(Y_d)$,for $d = 1, \dots, D$, over K clusters, such that
    $\mathcal{O}_K(Y_d) = \sum_{(\lambda_1,\lambda_2) \in \mathcal{L}_{\lambda_1} \times \mathcal{L}_{\lambda_2}} \mathcal{B}_{(K,\lambda_1,\lambda_2)}(Y_d)$
    and $\mathcal{B}_{(K,\lambda_1,\lambda_2)}(Y_d)$ is a binary variable that indicate the relevance of variable $Y_d$ with regard to clustering. If $\mu_{kd}(\lambda_1, \lambda_2) = 0, \forall k = 1, \dots, K$, then $\mathcal{B}_{(K,\lambda_1,\lambda_2)}(Y_d) = 0$; otherwise.
According to the score of each variable $Y_d$, $\mathcal{O}_K(Y_d)$, we rank the variables by decreasing values $\mathcal{O}_K(Y_d)$, denoted as $\mathcal{I}_K$, such that
    $\mathcal{I}_K = (Y_1, \dots, Y_D) \quad \text{with } \mathcal{O}_k(Y_1) > \dots > \mathcal{O}_k(Y_d) \dots > \mathcal{O}_k(Y_D).$
Scan the variable set according to the order $\mathcal{I}_K$ and use BIC based variable selection we discussed in Section 2.7.2. Variable $Y_d$ is added if
    $BIC_{\text{diff}}(Y_d) = BIC_{\text{clust}}(Y_d, \mathcal{Y}^S | K, m) - [BIC_{\text{clust}}(\mathcal{Y}^S | K, m) + BIC_{\text{reg}}(Y_d | \mathcal{Y}^R \subseteq \mathcal{Y}^S)]$
    is negative. Here, $\mathcal{Y}^S$ is the variable set for clustering and $\mathcal{Y}^R$ is the subset of $\mathcal{Y}^S$ that linearly explain the variable $Y_d$.
```

# 3 Alzheimer's Disease Neuroimaging Initiative(ADNI) data analysis

## 3.1 Data description

Alzheimer's disease will cause a sustained decline in thinking, behavior, and social skills, disrupting a person's ability to care for themselves. Early symptoms of this disease may be forgetting recent events or conversations. As the disease progresses, people with Alzheimer's disease will experience severe memory impairment and lose the ability to perform everyday activities.

Many patients are unable to work full-time or part-time because of Alzheimer's disease, and their own lives, employment, families, and even lives are affected, not only by the financial burden but also by the many more complex social issues that arise. This shows that Alzheimer's is a disease for governments to worry about, and the challenges it poses to families, healthcare, the economy, society, and regulations are becoming increasingly profound and extensive. Rapid prevention and early detection programs are therefore needed, as many cases of dementia can be effectively prevented through early detection.

This dataset is obtained from the Alzheimer's Disease Neuroimaging Initiative(ADNI) under https://ADNI1.loni.usc.edu/data- samples/access-data/. ADNI began in 2004 under the leadership of Dr. Michael W. Weiner. The study now has three phases. At each stage, the patients are tested for various indicators and their disease status. In our research, we just consider the participants that join the first phase of ADNI, which was launched in October 2004 for a 6-year duration. In the dataset, it provides the measurement data from neuropsychological tests, MRI and disease states of each patients including **Cognitively Normal(CN)**, **Mild Cognitive Impairment(MCI)** and **Dementia (DEM)**. **Dementia (DEM)** represents Alzheimer's disease and **Mild Cognitive Impairment (MCI)** is a transitional stage between **CN** ageing and the onset of **DEM**. If the condition is not detected in time, there is a very high risk that the patient's condition will progress to **DEM**. Not all people diagnosed with **MCI** show progressive decline in cognitive ability, many of them remain at **MCI** levels and a significant proportion revert to **cognitive normal (CN)** status with treatment. Early detection of **MCI** is therefore essential. There are many models for predicting Alzheimer's disease. In contrast, our research in this thesis focuses on identifying the constituent structures of potential patients, helping us to identify the subtypes of patients for earlier treatment.

- **Neuropsychological test**: contains Alzheimer's Disease Assessment Scale(ADAS) test and Alzheimer Cognitive Composite(MPACC) tests that evaluate a person's cognition and analyze the dysfunctional level of Alzheimer's disease.
- **MRI measure**: is an approach applied for visualization of the physiological process of the brain.
- **Disease status**: Cognitively normal, Mild Cognitive impairment and Alzheimer's disease.

There are 203 **CN** ,129 **DEM**, 312 **MCI** in dataset, see in Table 4. In our analysis of the dataset ADNI, we just consider neuropsychological test variables (incl.**S13**, **mPd**, **mPt**) and the MRI variables, which contains **Ven**, **Hip**,**WhB**,**Ent** since in the other

| Disease status | Number of patients |
|---|---|
| CN | 203 |
| MCI | 312 |
| DEM | 129 |
| Total | 644 |

Table 4: Several patients in each disease status.

variables too many missing values and some are discrete, **Fui**, **MTp**, **ICV**. Table 5 gives a short description of each variables in our data set obtained from ADNI group [2010].

| Attribute | Number of unique values | Description | Domain |
|---|---|---|---|
| **Cognitive test measurement** | | | |
| **S13** | 114 | The Cognitive Subscale (13 items) Alzheimer's Disease Assessment Scale | [1,50] |
| **mPd** | 608 | Modified Preclinical Alzheimer Cognitive Composite with Digit test Composite | [-20.646,5.176] |
| **mPt** | 605 | Modified Preclinical Alzheimer Cognitive Composite with Trails test | [-19.46,5.455] |
| **MRI measurements** | | | |
| **Ven** | 626 | Ventricles volume($mm^3$) | [5834,145115] |
| **Hip** | 590 | Hippocampus volume($mm^3$) | [3091,10769] |
| **WhB** | 627 | Whole Brain volume($mm^3$) | [669364,1364690] |
| **Ent** | 566 | Entorhinal volume($mm^3$) | [1467,5731] |
| **Fui** | 610 | Fusiform Gyrus volume($mm^3$) | [9610,24788] |
| **MTp** | 613 | Middle Temporal Gyrus volume($mm^3$) | [9375,28103] |
| **ICV** | 624 | intracranial volume($mm^3$) | [1116280,2110290] |
| **Patient specific measurement** | | | |
| **DX** | 3 | disease status | CN,MCI,DEM |
| **PTGEND** | 2 | Gender | male, female |
| **PTRACT** | 2 | Race | white, not-white |
| **PTMARY** | 2 | Marital status | married, not-married |

Table 5: Description of variables studied in **ADNI dataset**.

## 3.2 Exploratory Data Analysis(EDA)

Figure 16 and Figure 17 show the pairwise scatter plots of variables in the dataset ADNI. From Figure 17, it is clear that there are some differences between **CN** and the other two disease status. For **MCI** and **DEM**, their distributions on the variables **mPd** and **mPt** also differed. Then we fit the empirical cumulative distribution function of each variable to obtain u-data.

Figure 18 shows the normalized contour plot of **ADNI** dataset. The variables **mPd** and **mPt** have very strong dependence since **mPd** and **mPt** use a similar test system. But

**mPd** measures working memory by asking the patient to repeat back a sequence of digits of increasing length, until they are not able to. The **mPt** test determines performance of processing speed with a smaller score indicating more severe impairment. Besides, **S13** is dependent on **mPt** with an empirical Kendall's $\tau$ equal to -0.71. **WhB** is highly correlated with **ICV**, **MTp** and **Hip**. Moreover, **Ent** and **Hip** also have high correlation coefficient. Besides, we observe there are non- Gaussian dependence between some pairs of variables since the contours are non-elliptical. For example the counter plot of **S13** and mPd as well as **mPt** shows they are non-Gaussian dependent.

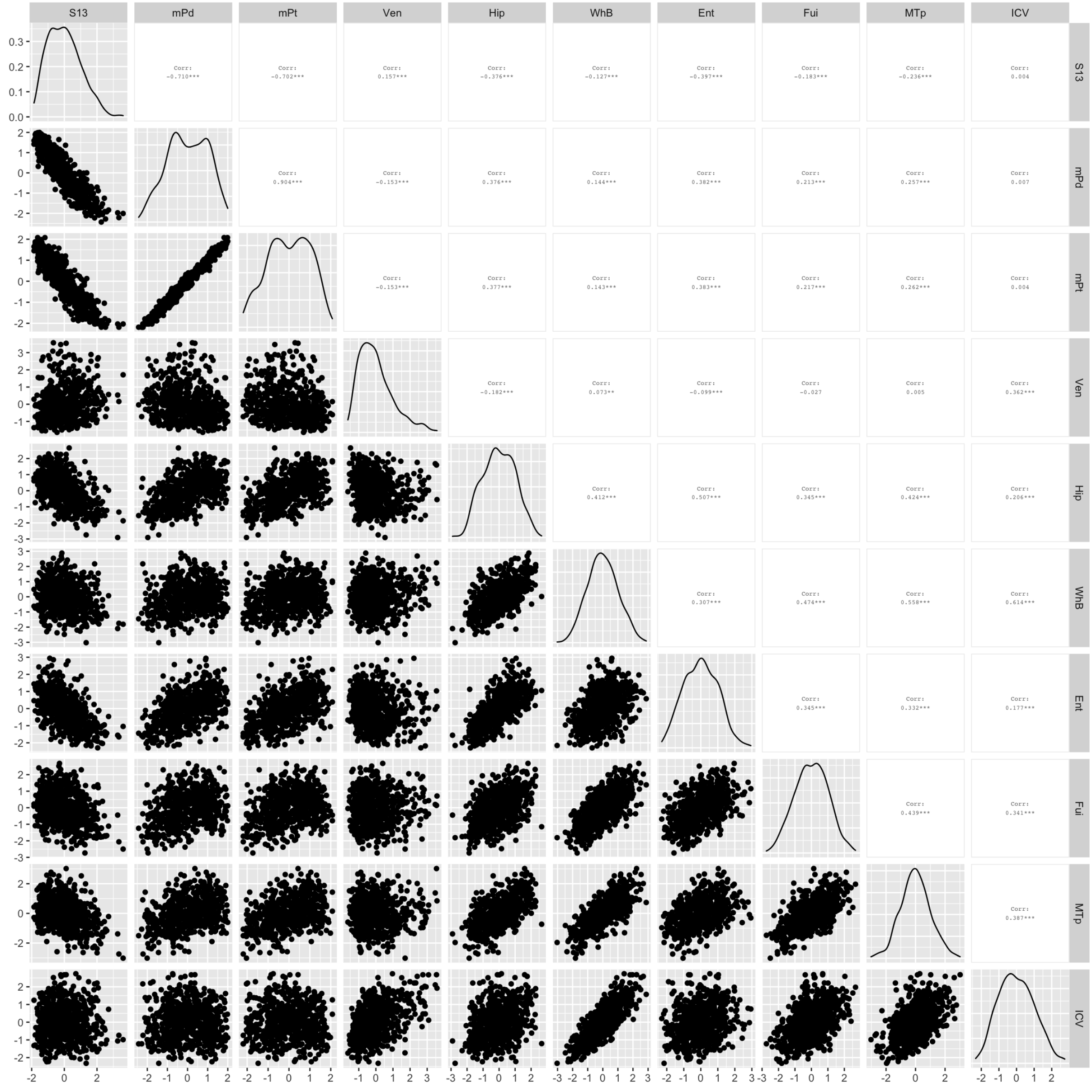


Figure 16: Pairwise scatter plots of **ADNI1**.

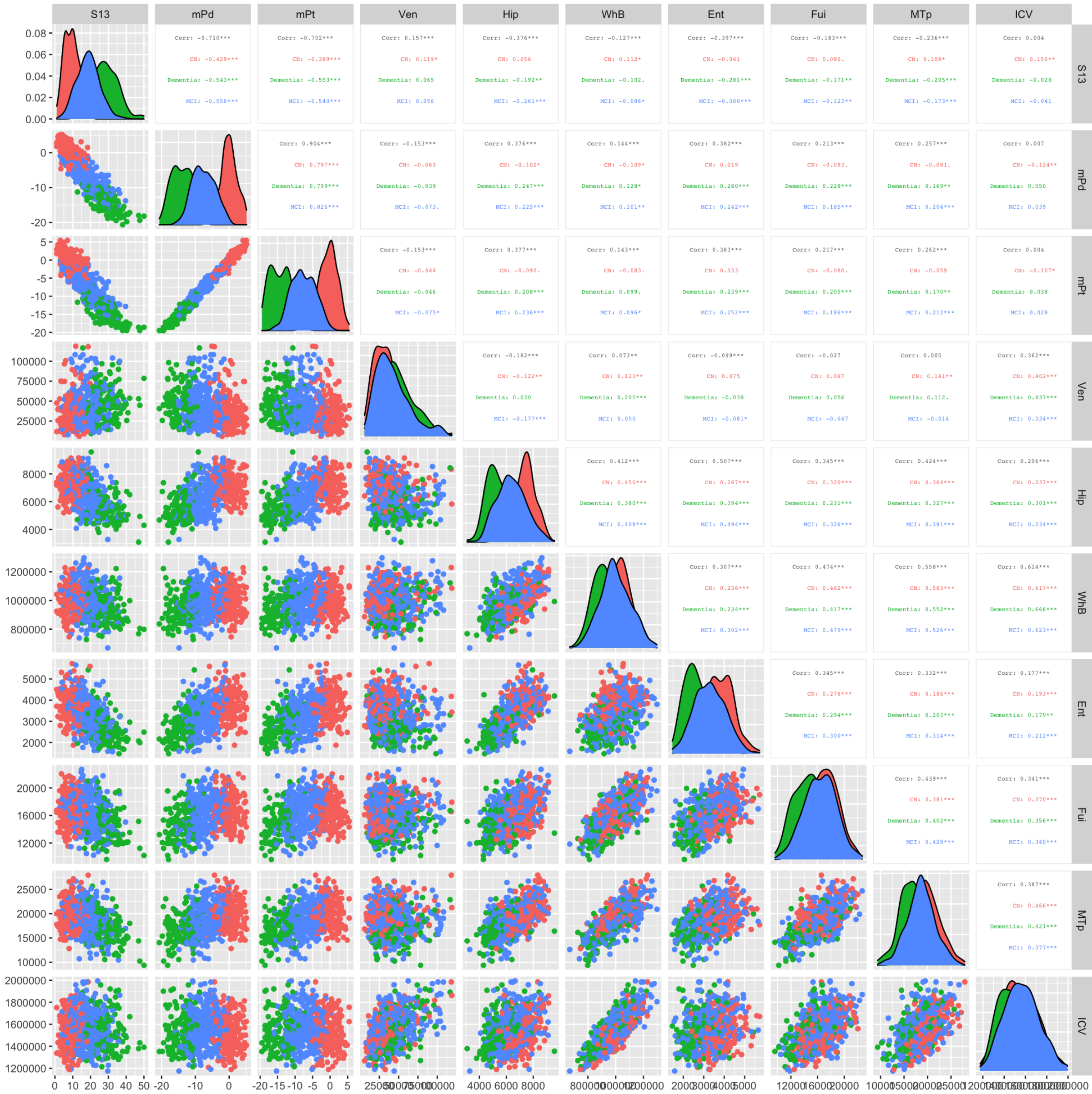


Figure 17: Pairwise scatter plots of the **ADNI** dataset, where red points represent **CN** patients, blue is for the **MCI** patients and green is for the **DEM** patients; diagonal is the marginal density function of each variable stratified; upper triangular contains the pairwise Kendall's $\tau$ also stratified by disease status.

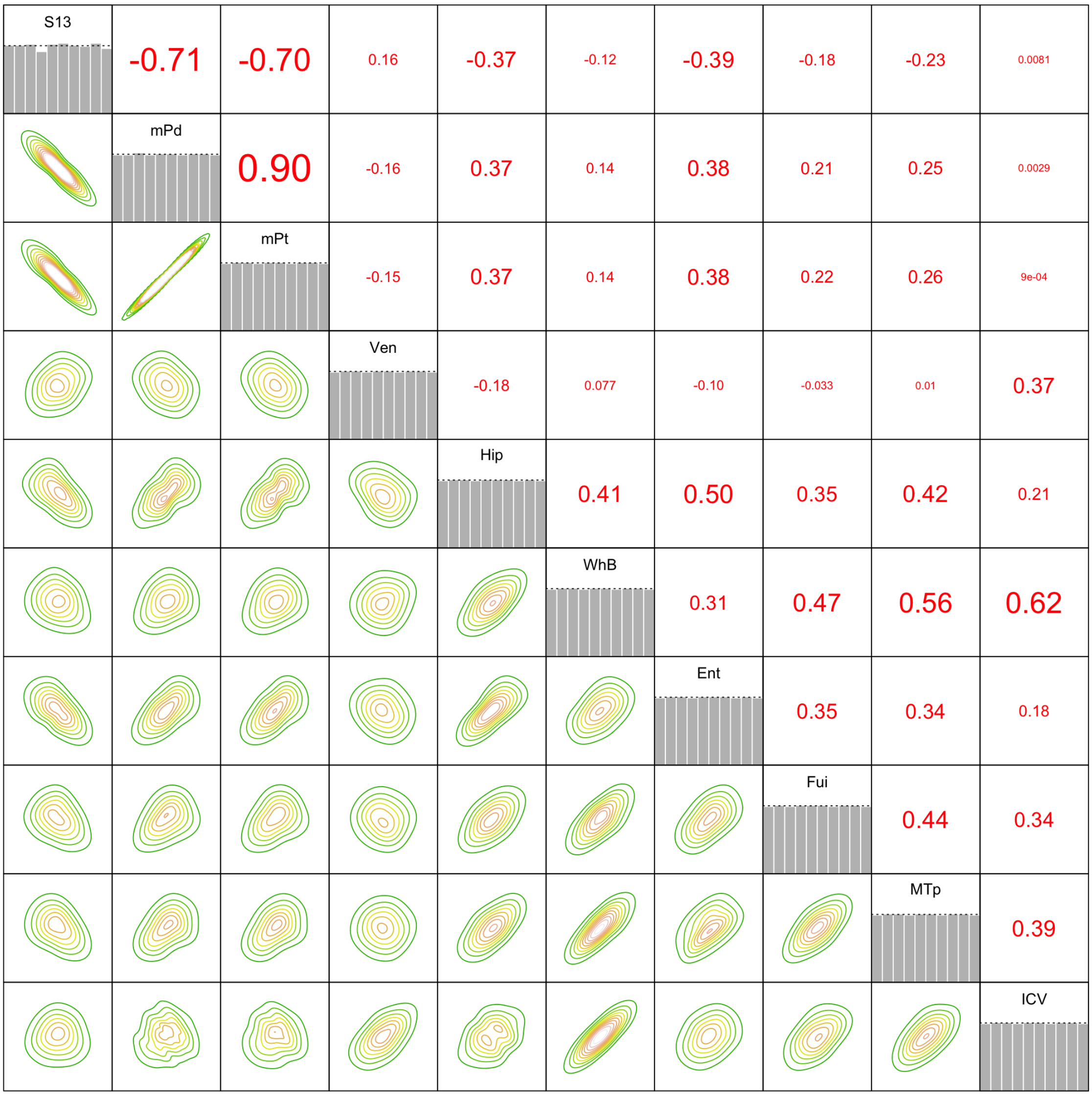
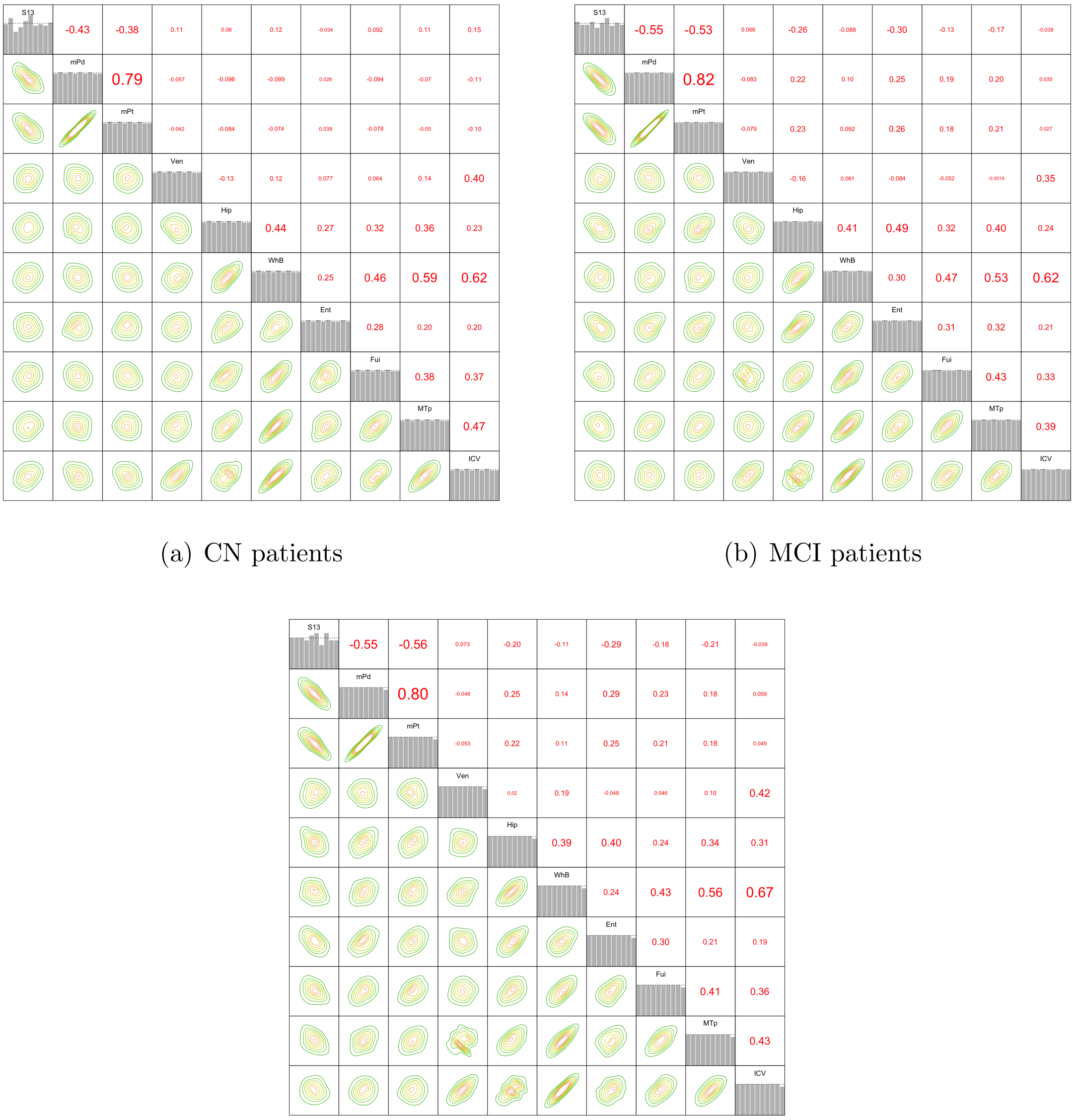


Figure 18: Normalized contour plot of **ADNI** (lower triangular: pairwise normalized contours; diagonal: histogram of empirical copula margins; upper triangular: pairwise empirical Kendall's $\tau$ ).

### Normalized contour plot of all variables separated by disease status

We also fit the empirical cumulative distribution function of each variable in each disease status to obtain u-data and plot the normalized contour plot for each class, including cognitively normal (**CN**), mild cognitive impairment(**MCI**) and Alzheimer's disease(**DEM**), see in Figure 19. There are still many contours plots that are non-elliptical, which means there are non-Gaussian dependence structures. Furthermore, pairwise dependency patterns between the same pair of variables are usually similar for **CN**, **MCI**, and **DEM** patients but may differ in strength. As we can see, for each disease status, it exists non Gaussian dependence between **S13** and **mPt**. Comparing three different disease status, it has higher strength for **DEM** patients than others. We use the R package **VineCopula**(Nagler et al. [2021]) for the analysis. After fitting a bivariate copula for (**S13**, **mPt**) in each disease status, we have the following results.

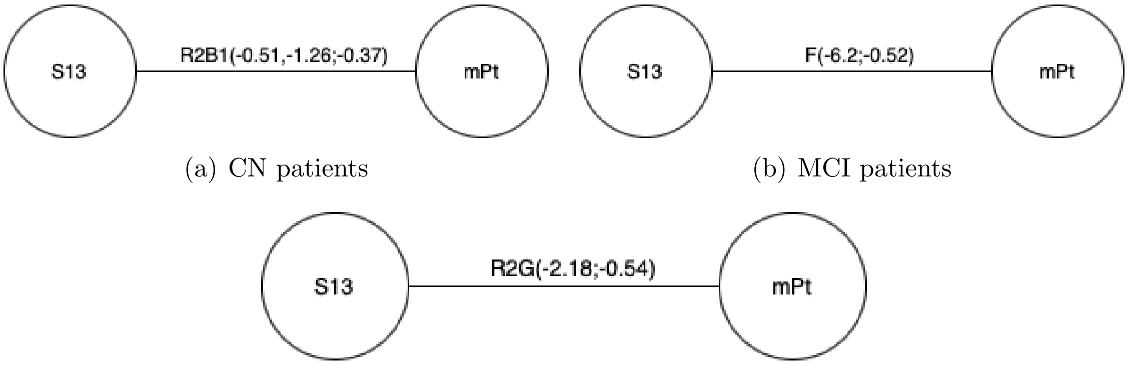


(a) CN patients

(b) MCI patients

(c) DEM patients

Figure 19: Normalized empirical contour plot for each disease status (lower triangular: pairwise normalized contours; diagonal: histogram of empirical copula margins; upper triangular: pairwise empirical Kendall's $\tau$).



(a) CN patients

(b) MCI patients

(c) DEM patients

Figure 20: Bivariate copula for (**S13**, **mPt**) for each disease status where R2B1: Rotated BB1 270 degrees; F:F; R2G:Rotated Gumbel 270 degrees. The true parameter value and corresponding Kendall's $\tau$ of the pair copula are given inside the parenthesis (parameter/Kendall's $\tau$).

### Empirical pairwise Kendall's $\tau$ separated by disease status

The Table 6 shows the empirical pairwise Kendall's $\tau$ separated by disease status. We can see that the empirical Kendall's $\tau$ coefficient between **S13** and **Hip** is different for **CN** and others two disease status. The Kendall's $\tau$ is much smaller than the values for **MCI** and **DEM**. Similarly, **mPt** and **Ent** has lower correlation in the group **CN**. Similarly for **mPt** and **ICV**. For **Hip** and **WhB**, the Kendall's $\tau$ of **MCI** and **DEM** are higher than that of **CN**. Besides, for group **MCI**, the **Ven** and **MTp**, **mPd** and **Fui** are hardly dependent.

| | status | S13 | mPd | mPt | Ven | Hip | WhB | Ent | Fui | MTp | ICV |
|---|---|---|---|---|---|---|---|---|---|---|---|
| S13 | CN | 1 | -0.43 | -0.38 | 0.11 | 0.06 | 0.12 | -0.03 | 0.09 | 0.11 | 0.15 |
| | MCI | 1 | -0.55 | -0.53 | 0.07 | -0.26 | -0.09 | -0.30 | -0.13 | -0.17 | -0.04 |
| | DEM | 1 | -0.55 | -0.56 | 0.07 | -0.20 | -0.11 | -0.29 | -0.18 | -0.21 | -0.04 |
| mPd | CN | | 1 | 0.79 | -0.06 | -0.09 | -0.10 | 0.03 | -0.01 | -0.07 | -0.11 |
| | MCI | | 1 | 0.82 | -0.09 | 0.22 | 0.10 | 0.25 | 0.19 | 0.20 | 0.04 |
| | DEM | | 1 | 0.80 | -0.046 | 0.25 | 0.14 | 0.29 | 0.23 | 0.18 | 0.059 |
| mPt | CN | | | 1 | -0.04 | -0.09 | -0.07 | 0.04 | -0.08 | -0.05 | -0.10 |
| | MCI | | | 1 | -0.08 | 0.23 | 0.09 | 0.26 | 0.18 | 0.21 | 0.03 |
| | DEM | | | 1 | -0.05 | 0.22 | 0.11 | 0.25 | 0.21 | 0.18 | 0.05 |
| Ven | CN | | | | 1 | -0.13 | 0.12 | 0.08 | 0.06 | 0.14 | 0.40 |
| | MCI | | | | 1 | -0.16 | 0.06 | -0.08 | -0.05 | 0.00 | 0.35 |
| | DEM | | | | 1 | 0.02 | 0.19 | -0.05 | 0.05 | 0.10 | 0.42 |
| Hip | CN | | | | | 1 | 0.44 | 0.27 | 0.32 | 0.36 | 0.23 |
| | MCI | | | | | 1 | 0.41 | 0.49 | 0.32 | 0.40 | 0.24 |
| | DEM | | | | | 1 | 0.39 | 0.40 | 0.24 | 0.34 | 0.31 |
| WhB | CN | | | | | | 1 | 0.25 | 0.46 | 0.59 | 0.62 |
| | MCI | | | | | | 1 | 0.30 | 0.47 | 0.53 | 0.62 |
| | DEM | | | | | | 1 | 0.24 | 0.43 | 0.56 | 0.67 |
| Ent | CN | | | | | | | 1 | 0.28 | 0.20 | 0.20 |
| | MCI | | | | | | | 1 | 0.31 | 0.32 | 0.21 |
| | DEM | | | | | | | 1 | 0.30 | 0.21 | 0.19 |
| Fui | CN | | | | | | | | 1 | 0.38 | 0.37 |
| | MCI | | | | | | | | 1 | 0.43 | 0.33 |
| | DEM | | | | | | | | 1 | 0.41 | 0.36 |
| MTp | CN | | | | | | | | | 1 | 0.47 |
| | MCI | | | | | | | | | 1 | 0.39 |
| | DEM | | | | | | | | | 1 | 0.43 |
| ICV | CN | | | | | | | | | | 1 |
| | MCI | | | | | | | | | | 1 |
| | DEM | | | | | | | | | | 1 |

Table 6: Empirical pairwise Kendall's $\tau$ separated by disease status.

### Boxplot of all continuous variables separated by disease status

According to the boxplot of **ADNI** in Figure 21, we can conclude that the variables **S13**, **mPd** and **mPt** behave differently for each disease status group. The values of **mPd**, **mPt** and **Ent** will get smaller as the disease progress, while **S13** presented an opposite trend, i.e increased as the condition worsens. **Ent** and **Hip** also differ slightly in the status of disease. For variable **Ven**, the patients who are **CN** has lower value than whose in other two status and values for **MCI** are similar to **DEM**. The same is true for variable **Fui**, i.e., the value for **DEM** is distinct but for other status, they are almost same.

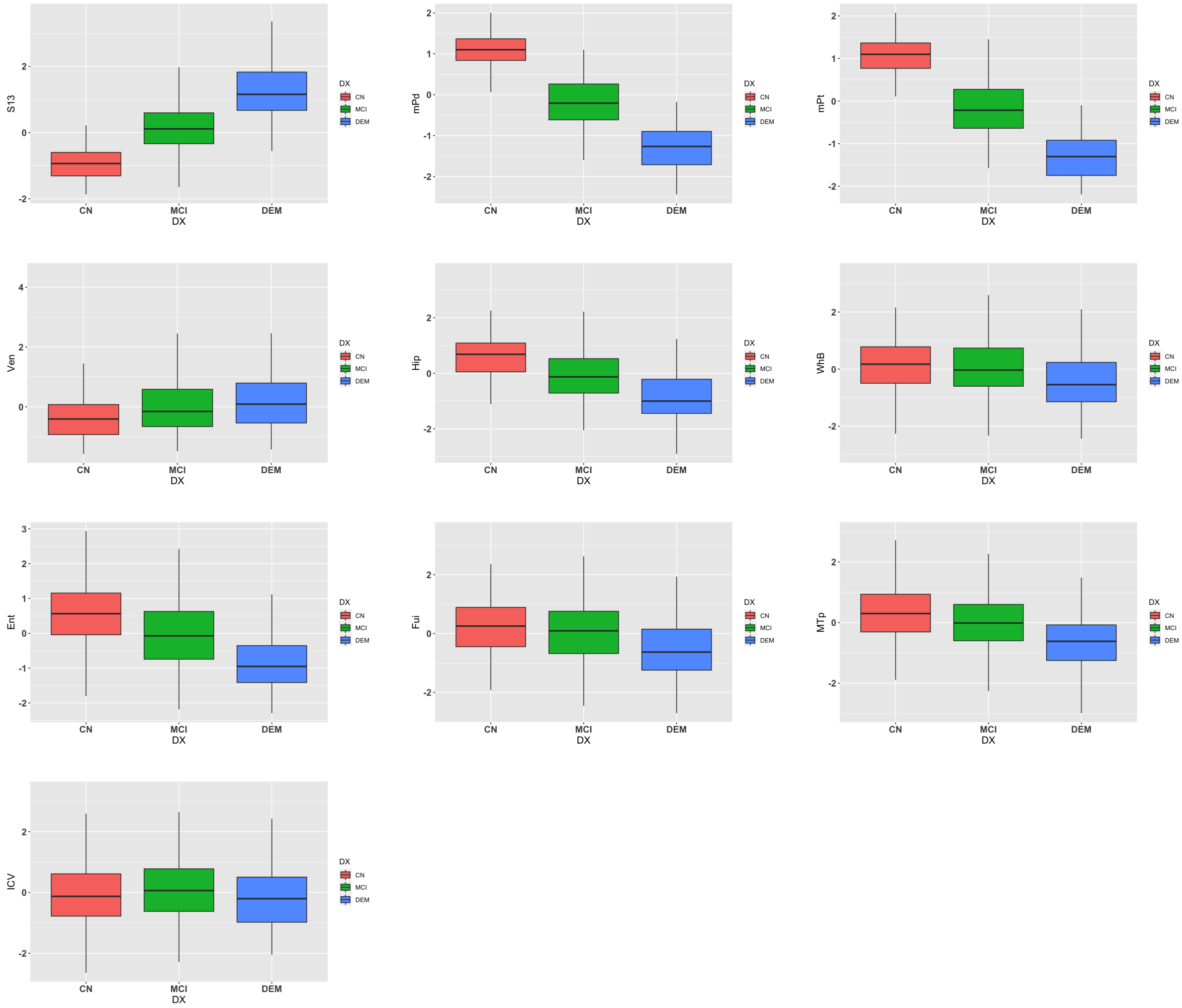


Figure 21: Boxplots of continuous variables **S13**, **mPd**, **mPt**, **Ven**, **Hip**, **WhB**, **Ent**, **Fui**, **MTp** and **ICV** separated by disease status(DX).

### Histograms of all variables separated by disease status

The histograms of continuous variables separated by three disease status are given in Figure 22. As we mentioned in the pairwise scatter plot in Figure 17, the first three variables, **S13**, **mPd** and **mPt** maybe useful for clustering since there exists the clear difference between **CN**, **MCI** and **DEM**, especially for **CN** and **DEM**.

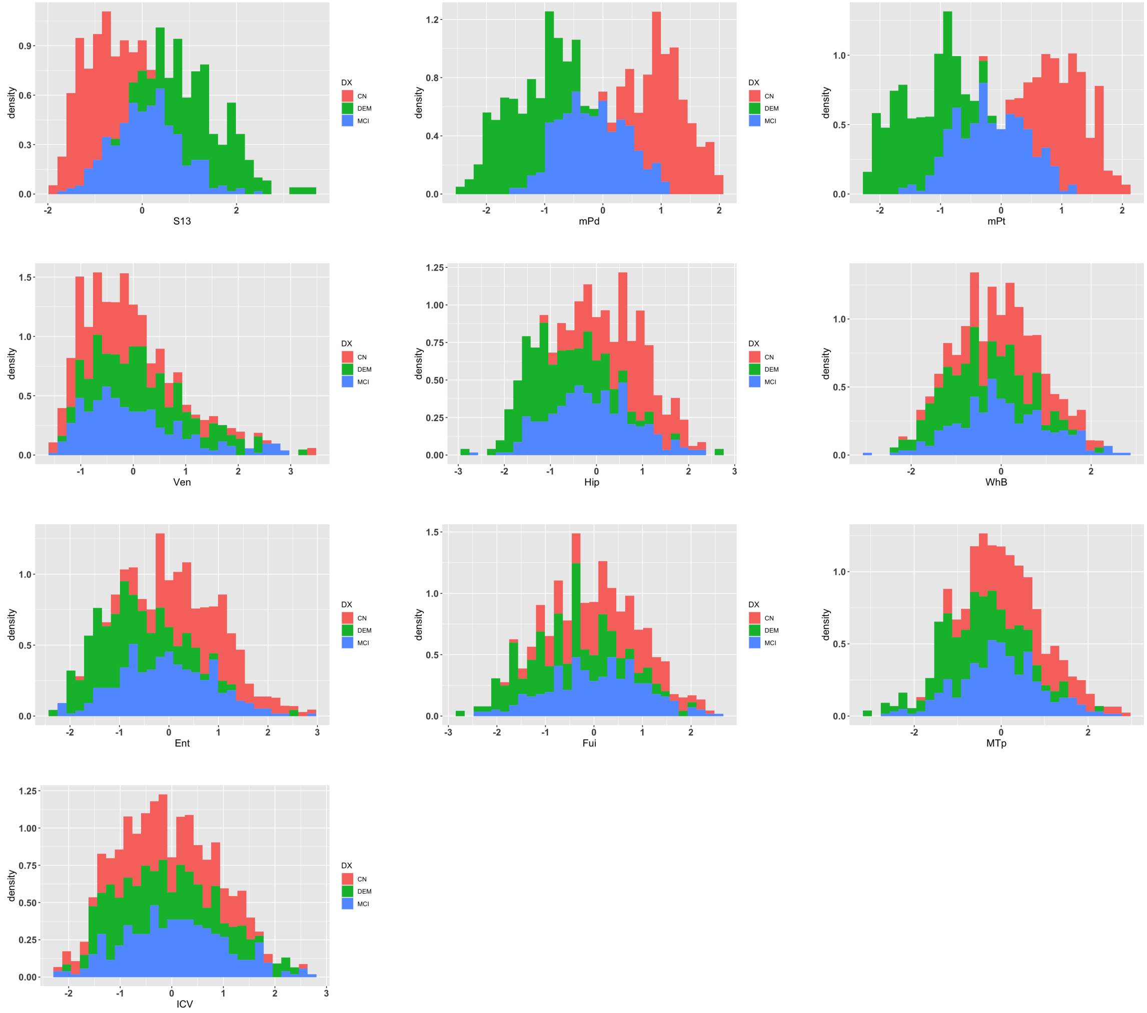


Figure 22: Histogram of continuous variables **S13**, **mPd**, **mPt**, **Ven**, **Hip**, **WhB**, **Ent**, **Fui**, **MTp** and **ICV** separated by disease status, where red color represent **CN** patients, green **DEM** patients, Blue **MCI** patients.

### Boxplot of all variables separated by gender

The dataset also contains variables for gender, and to better understand the dataset, we plot the boxplot of **ADNI** classified by gender. As seen in Table 7, there are 97 females and 107 males that are diagnosed with **CN**. For **MCI**, 115 females and 198 males are in the group, and for DEM, it has 129 patients with 61 females and 68 males. From Figure 23, there is some difference between females and males in the variables. The average volume of male's **Ven**, **Hip**, **Ent**, **Fui**, **MTP** and **ICV** are higher than the female. The value of the first three variables differed for the three disease status in both males and females. However,for variable **ICV**, the value of female who is **CN** is close to the **MCI**, which may mislead the clustering results.

| Disease status | female | male |
|---|---|---|
| CN | 96 | 107 |
| MCI | 115 | 197 |
| DEM | 61 | 68 |

Table 7: Number of patients in each disease status and gender

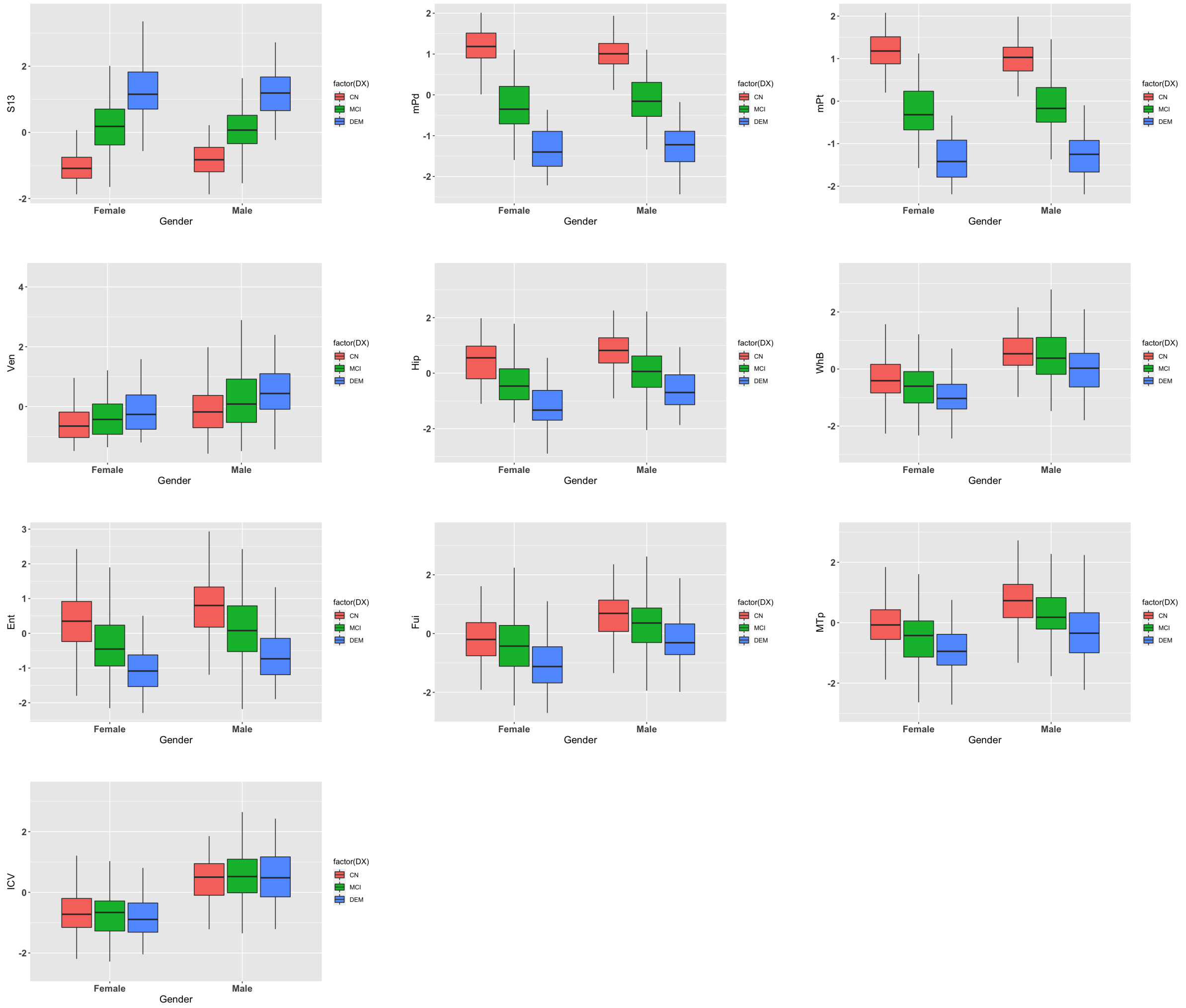


Figure 23: Boxplot of *ADNI1* by gender.

### Boxplot of all variables separated by marital status

In addition, the dataset contains information on the marital status of patients. The status contains married, divorced, never married, and widowed. To simplify these status, we divide them into two groups one is married and another is non-married. The Table 8 shows the number of patients with each disease status and marital status. There is not much difference between married and non-married patients. For married person, the values of **Ven** for **MCI** and **DEM** are almost the same. Therefore, these variables may mislead the performance of results for a married person.

| Disease status | married | not married |
|---|---|---|
| CN | 140 | 62 |
| MCI | 244 | 68 |
| DEM | 106 | 23 |

Table 8: Number of patients in each disease status and marital

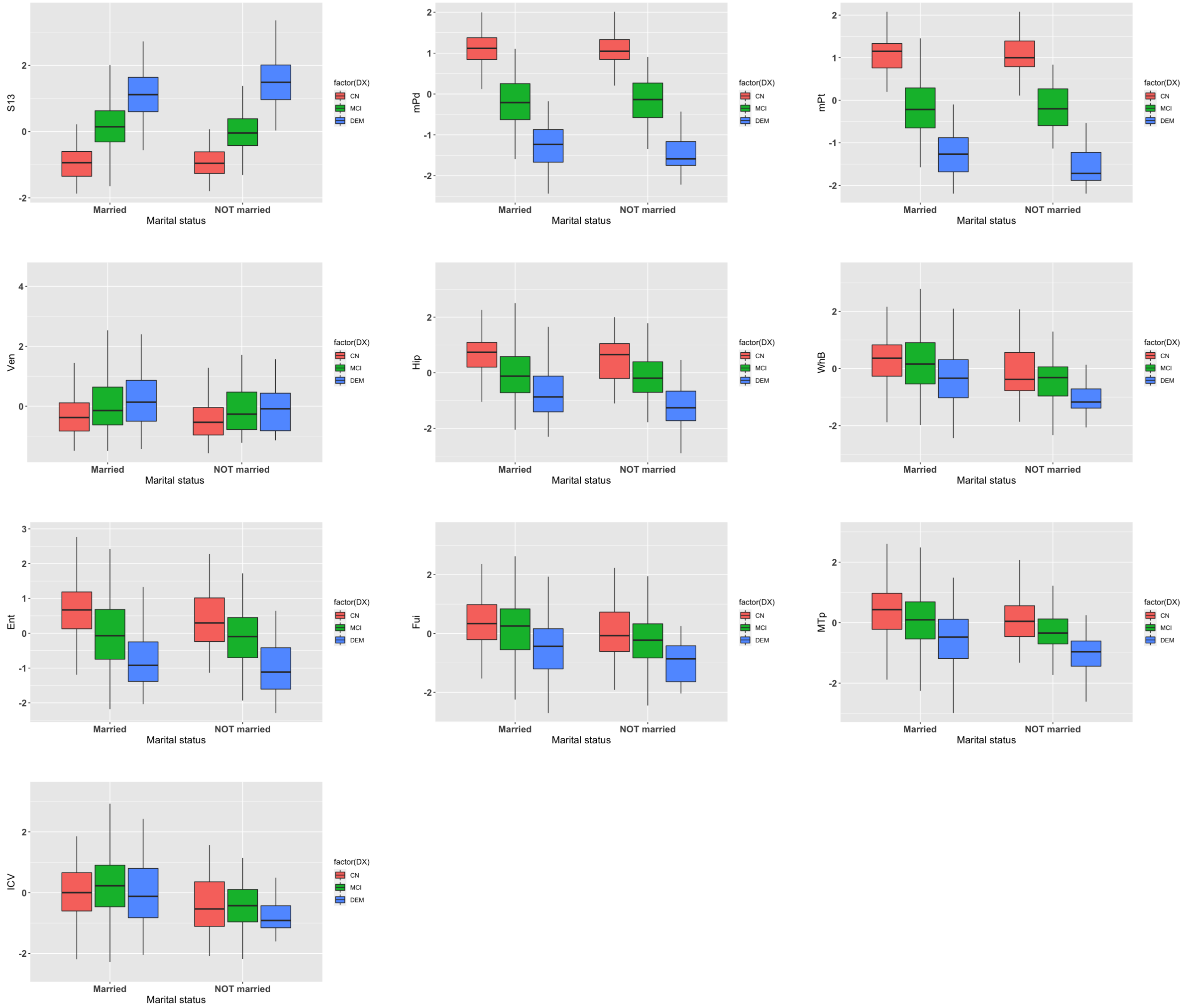


Figure 24: Boxplot of ADNI dataset by marital status.

# 4 Performance of the Gaussian variable selection approaches for clustering in the ADNI data

In the subsequent clustering analysis, we will apply the variable selection approaches discussed in the Section 2.7 and compare the results of the distance-based and different Gaussian model-based approaches. The R packages we use for clustering tasks are: **mclust**(Scrucca et al. [2016]), **sparcl**(Witten and Tibshirani [2018]), **Clustvarsel** (Scrucca and Raftery [2018]), **VarselLCM**(Marbac and Sedki [2017]), **Vscc**(Andrews and McNicholas [2013]) and **SelvarMix**(Sedki et al. [2017]). The R package **mclust** is the package for classic Gaussian mixture model clustering without performing variable selection and the other 5 packages are used for the variable selection. Table 9 shows the type of variable selection approach that each R package used. All information criterion-based and hybrid-based approaches except **VarselLCM** perform Gaussian mixture model clustering via R package **mclust** after variable selection. For the approach **VSCC** we need to check the uncertainty of each possible model to select variables, as shown in Equation (87). However, because it only makes sense to perform variable selection without "harm", we can calculate the uncertainty from the original dataset and its solution can be considered as part of the variable selection process. In other words, under **VSCC**, if the uncertainty of the full data set is minimal, we will select the full data set rather than a reduced set. In addition, we have scaled data with mean 0 and variance 1 in advance due to some assumptions and requirements of variable selection approaches.

| Approach | R package | Detail | Section |
|---|---|---|---|
| Penalization | **Sparcl** | Penalization k-means approach | 2.7.1 |
| Information criterion | **Clustvarsel** | Model selection based on BIC | 2.7.2 |
| | **VarSelLCM** | Model selection based on MICL | 2.7.2 |
| Hybrid | **VSCC** | Hybrid filter-wrapper approach | 2.7.3 |
| | **SelvarMix** | Hybrid approach based on lasso penalization | 2.7.3 |

Table 9: Variable selection approaches and their corresponding R package.

## 4.1 Performance using complete ADNI dataset $\mathbf{ADNI}_{10}$

According to the data description, we have 10 variables in the dataset and we denoted it as $\mathbf{ADNI}_{10}$. To find out the relevant variables for clustering, we applied 5 variable selection approaches explained in Section 2.7 and compare them with **GMM**. Since the number of clusters is unknown, we estimate it after fitting 2 to 10 clusters and compute the Adjusted Rand Index(**ARI**) as well as Maximum-Match-Measure(**MMM**). In Table 10, it is clear that the optimal clusters that most of the approaches selected are either 3 or 4. The **Sparcl** has highest **ARI** and **MMM** when the number of components is 3, while **VSCC** is second largest value of **ARI** for the cluster K=4. The **ARI** of the original approach for the Gaussian mixture model,**GMM**, reached its maximum at K=2 compared to the other variable selection approach.

| ARI | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| number of clusters | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| GMM | **0.31** | 0.29 | 0.21 | 0.19 | 0.15 | 0.17 | 0.18 | 0.13 | 0.11 |
| Sparcl | 0.28 | **0.43** | 0.22 | 0.25 | 0.24 | 0.21 | 0.20 | 0.17 | 0.15 |
| Clustvarsel | 0.08 | **0.29** | 0.24 | 0.14 | 0.13 | 0.19 | 0.16 | 0.16 | 0.11 |
| VarSelLCM | 0.28 | **0.34** | 0.23 | 0.24 | 0.23 | 0.21 | 0.20 | 0.18 | 0.17 |
| VSCC | 0.10 | 0.29 | **0.38** | 0.17 | 0.22 | 0.25 | 0.24 | 0.21 | 0.11 |
| SelvarMix | 0.16 | 0.24 | **0.26** | 0.25 | 0.23 | 0.21 | 0.20 | 0.22 | 0.23 |
| MMM | | | | | | | | | |
| number of clusters | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| GMM | **0.66** | 0.60 | 0.51 | 0.39 | 0.38 | 0.39 | 0.42 | 0.30 | 0.28 |
| Sparcl | 0.59 | **0.77** | 0.52 | 0.49 | 0.44 | 0.42 | 0.37 | 0.31 | 0.30 |
| Clustvarsel | 0.41 | **0.61** | 0.56 | 0.42 | 0.36 | 0.54 | 0.41 | 0.39 | 0.36 |
| VarSelLCM | 0.60 | **0.71** | 0.52 | 0.47 | 0.40 | 0.38 | 0.37 | 0.32 | 0.29 |
| VSCC | 0.44 | 0.60 | **0.61** | 0.46 | 0.53 | 0.42 | 0.43 | 0.40 | 0.28 |
| SelvarMix | 0.46 | 0.56 | **0.60** | 0.57 | 0.54 | 0.47 | 0.47 | 0.50 | 0.50 |

Table 10: **ARI** and **MMM** of approaches using **ADNI**$_{10}$ data under different numbers of K (The best model selected by each approach is bolded).

**Analysis for assuming K=2 to 10 clusters:** By the **mclust** package, we performed the GMM approach by setting K=2 to 10 and considering all variables to get the optimal clusters that are equal to 3 with **ARI**=0.29. Then, we applied 5 variable selection approaches to the **ADNI**$_{10}$ dataset with all 10 variables and get the results. As for the penalization kmeans approach **Sparcl**, it requires the number of clusters in advance. Therefore, we will use the number of clusters K=3, which is estimated by the GMM approach in **mclust** package. For the remaining four approaches of variable selection, we derived relevant variables and corresponding clustering results by iterating over the number of clusters from K = 2 to 10, see Figure 25.

As we can see in Figure 25, except for **SelvarMix**, the other approaches all choose 3 as the optimal number of clusters and it is the same as the number of true components in our dataset. According to **ARI** and **MMM**, the performance of **Sparcl** is great but it selects all the variables. For the approach **Clustvarsel**, we choose the backward direction rather than the forward direction since for forward selection, each addition of a new variable may render one or more of the already included variables non-significant. And it chooses only 5 variables with **ARI** =0.294 and **MMM**=0.61. The results are almost the same as **GMM** and **VSCC**, but it chooses fewer variables which means some variables may be irrelevant for clustering. **VarselLCM** also chooses all the variables for clustering with the number of clusters equal to 9, which is unsatisfactory. For these 6 approaches, they all consider **Ven**, **Hip**, **WhB** and **ICV** for clustering. For **Hip** is reasonable because we have observed from the boxplot in Figure 21 that different disease status are represented differently in this variable. However, in boxplot, **Ven**, **WhB** and **ICV** do not distinguish diseases well, so including these variables may mislead the results.

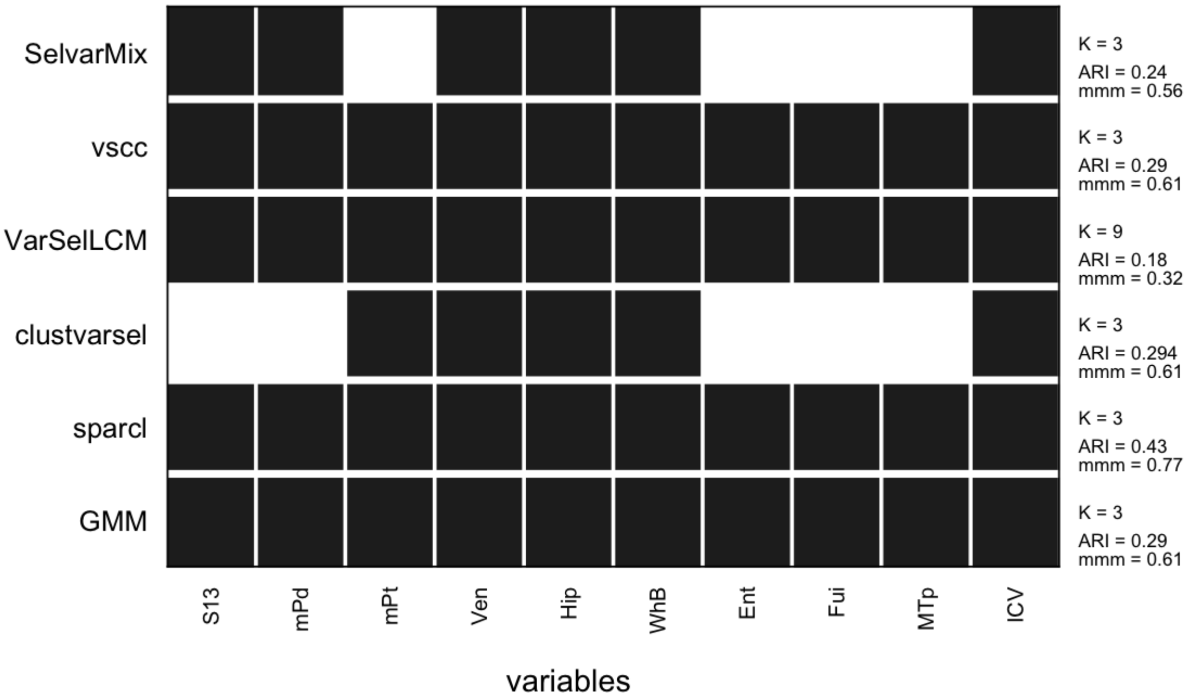


Figure 25: Selected variables(black) chosen according to variable selection approaches indicated in the row using $\mathbf{ADNI}_{10}$ dataset allowing for **K=2 to 10** clusters.

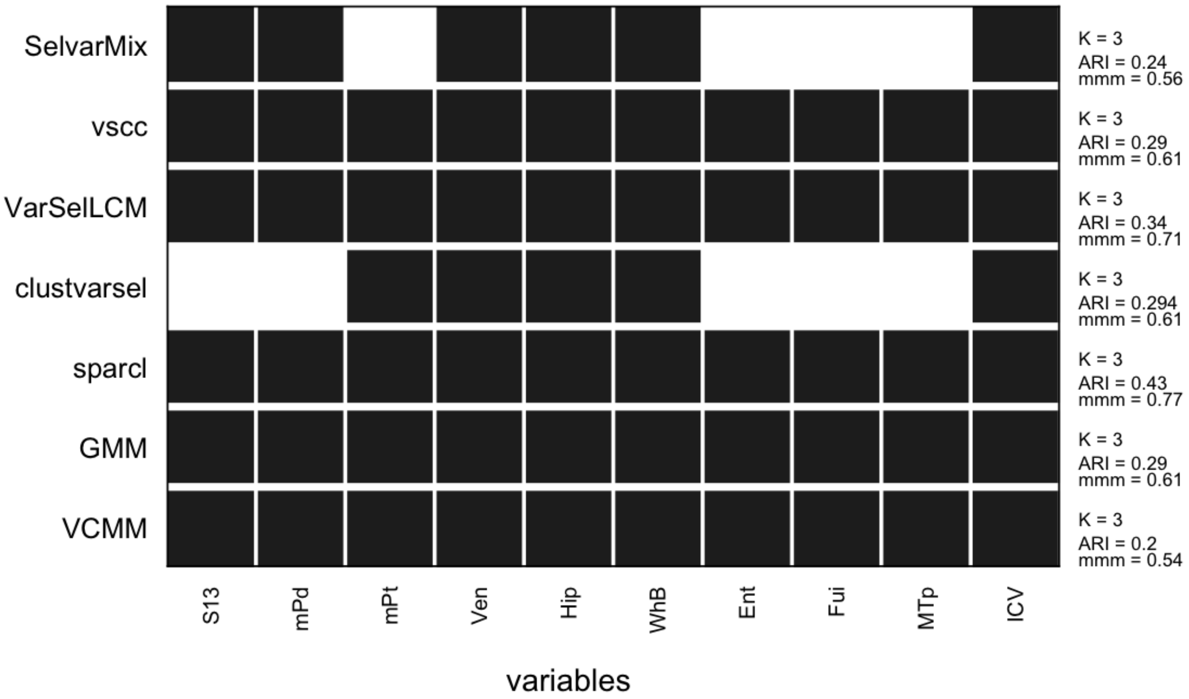


Figure 26: Selected variables(black) chosen according to variable selection approaches indicated in the row using $\mathbf{ADNI}_{10}$ dataset assuming **K=3** clusters.

**Performance assuming K=3:** Now, we specify K=3 and cluster the observations again to see the effect of the number of clusters for **VarSelLCM** see in Figure 26. As we said at the beginning of our introduction to **VSCC** in Section 2.7.3, we consider the model with full variables as an option for model selection, that is, if the uncertainty in the complete dataset is minimal, we select the complete dataset. Since the complete dataset has the least certainty in this clustering process, we select all variables and cluster them with **mclust**, which leads to the same results for **GMM** and **VSCC**. However, if we do not consider the full data model even though it has lowest uncertainty during the selection process, i.e., if we must exclude variables, we will get a model with only three variables, that is **mPd**, **Ven** and **Hip** with **ARI**= 0.22 and MMM=0.54. Besides, the result of **VarSelLCM** is improved, which becomes the second-best result. Compare this with the original **GMM** approach, which both selects all variables for clustering, but the results are different.

Therefore, to compare these two models estimated by **GMM** and **VarSelLCM**, we used the **Hungarian algorithm** and found the maximum weight matching to assign each cluster to the true partitions. The mixture probability of **GMM** and **VarSelLCM** can be easily estimated and is given in Table 11.

| | **DEM** | **CN** | **MCI** |
|---|---|---|---|
| GMM | | | |
| $\alpha$ | 0.10 | 0.41 | 0.49 |
| VarSelLCM | | | |
| $\alpha$ | 0.28 | 0.42 | 0.30 |

Table 11: Mixture probability of two approaches.

According to the mixture probability, we found that the probability of the cluster assigned to **DEM** under the model estimated by **GMM** is much smaller than the others, which implies that **DEM** is not a strong component under this model. However, the model estimated by **VarSelLCM** can clearly identify the other two disease status,**MCI** and **DEM**. Figure 27 shows the mean of each variable estimated by **GMM** and **VarSelLCM** as well as the true partitions. As we can see, the cluster $C_3\_GMM$ and $C_2\_VarSelLCM$ is similar to the disease status **CN**. But for $C_1\_GMM$, it differs from the **DEM**, while for the other two clusters estimated by **VarSelLCM**, i.e., $C_1\_VarSelLCM$ and $C_3\_VarSelLCM$, are both closer to the true disease status. The most important reason for the different result is that **VarSelLCM** assumes the independence of the variables used for clustering, so it only considers the variable's variance and not the covariance. Thus, it does not model the dependence we indicated in pairsplot in Figure 17. For **GMM** estimated by **mclust**, it considers covariance of variables as well as a richer family of models. The result of **VarSelLCM** is better than **GMM** may because **GMM** considers larger covariance matrices and might overfit.

Table 12 shows the partition results for all six approaches. From the table we found that almost all approaches identify **CN** patients. However, most of them do not perfectly distinguish between **MCI** and **DEM**, as they are all clustered in the same cluster. Compared to **clustervarsel** and **selvarmix**, **Sparcl** and **VarSelLCM** used all the variables but are able to better distinguish between the two diseases, i.e., **MCI** and **DEM**. For

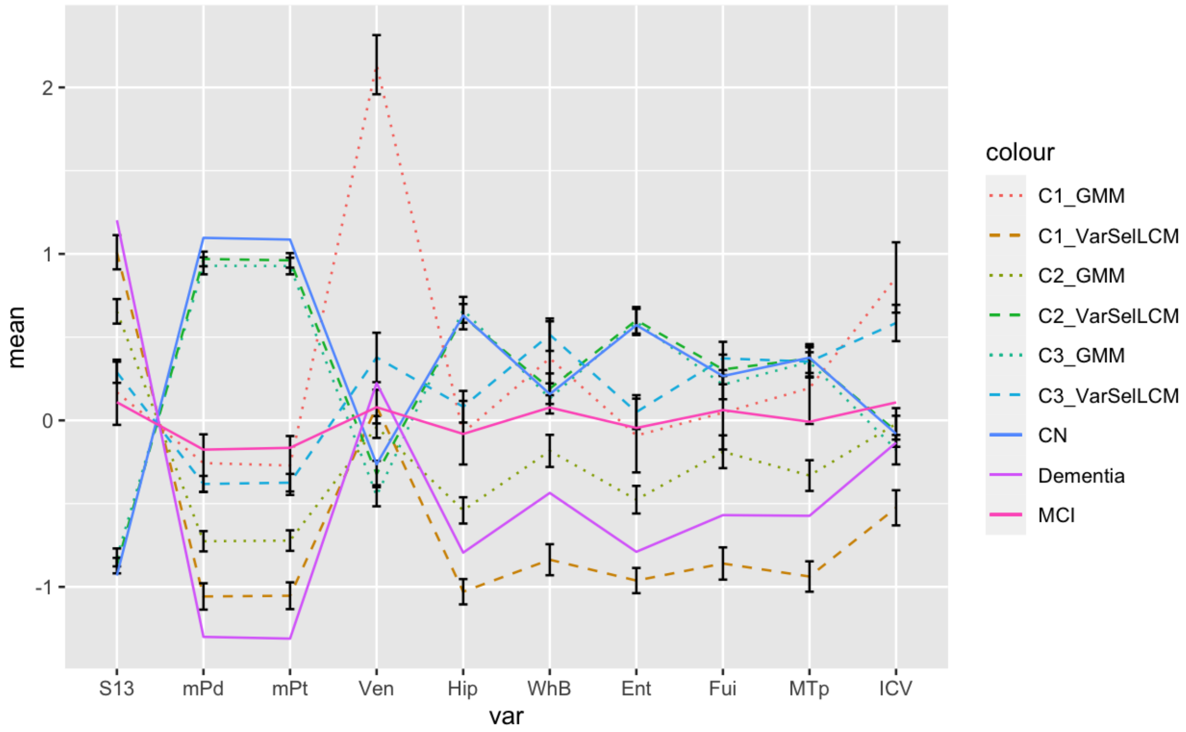


Figure 27: Comparison of the cluster estimated by **GMM** and **VarSelLCM** and true class mean for each clustering variable for **K=3** (the dotted line represents the mean of each variable in each cluster estimated by **GMM**, dashed line represents the mean of each variable in each cluster estimated by **VarSelLCM** and the solid line represents the mean of each variable in each disease status).

**Sparcl**, not only 98% of **CN** is clustered into the same components, but 81% of **DEM** is in component $C_2$, while 62% of **MCI** was in the remaining components, which performs better than the other approaches.

Sparcl

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | **199** | 0 | 4 |
| DEM | 0 | **105** | 24 |
| MCI | 69 | 49 | **194** |

GMM

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | 14 | 4 | **185** |
| DEM | **12** | 116 | 1 |
| MCI | 36 | **197** | 79 |

Clustvarsel

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | 14 | 2 | **187** |
| DEM | **13** | 116 | 0 |
| MCI | 38 | **193** | 81 |

VarselLCM

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | 1 | **197** | 5 |
| DEM | **100** | 0 | 29 |
| MCI | 79 | 75 | **158** |

VSCC

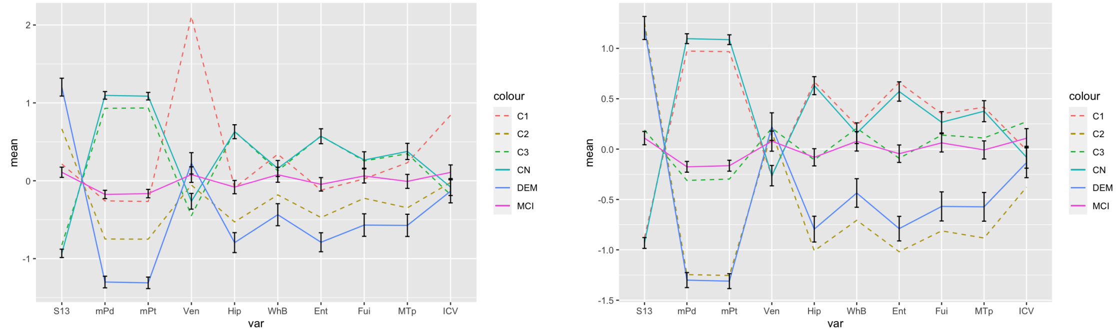
| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | 14 | 4 | **185** |
| DEM | **12** | 116 | 1 |
| MCI | 36 | **197** | 79 |

selvarMix

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | **189** | 1 | 13 |
| DEM | 4 | 115 | **10** |
| MCI | 116 | **163** | 33 |

Table 12: Partition results for the complete dataset $\mathbf{ADNI}_{10}$ for the 6 approaches and the number of cluster is chosen to **K=3** clusters.

To better compare the clustering results, we plot the mean of each variable in the true class and clustering group. Figure 28 represents a comparison of the estimated results with true classes. For **Clustvarsel** in Figure 28(a), the mean of each variable in $C_3$ is close to the mean of group **CN**, but the mean of each variable in $C_1$ is greater than the means in **DEM** as well as and the mean of $C_2$ is also lower than the true **MCI**'s means. In Figure 28(b), three lines are more consistent with the mean lines of true disease status.



(a) Clustvarsel when K=3

(b) Sparcl when K=3

Figure 28: Comparison of the cluster and true class means for each clustering variable for **K=3** (the dotted line represents the mean of each variable in each cluster estimated by the approaches and the solid line represents the mean of each variable in each disease status).

Although the set of relevant variables we derived from **Sparcl** for clustering is all 10 variables, we can compare the weights of the individual variables in the model, see Table 13. The weights of variables mean the importance of this variable for the clustering dataset. According to the Table 13, the weight of **mPd** is greatest and **mPt**, **S13**, **Hip** as well as **Ent** are also higher than others. This is reasonable since the boxplots in Figure 21 demonstrate that **S13**, **mPd** and **mPt** play important roles in distinguishing between the three disease states. Besides, the weights of variables **Ven** and **ICV** are much lower than the others. As we mentioned before, Figure 21 and 22 show that there is not much difference between the three disease status in variable **ICV**. The strictness of the $L_1$ penalty and the selection of the value of the parameter $s$ caused the weights to be not zero but close to zero, only 0.034 and 0.039 respectively. Here, $s$ is calculated by the Algorithm 2.7.1 which equals 2.59. If we change the value of $s$, the weight of some variables may converge to 0 or equal to zero. Because of the lower weights of these two variables, we can consider them as irrelevant variables that hardly matter for the model and cluster the observations again. Given the remaining variables **S13**, **mPd**,**mPt**,**Hip**,**WhB**,**Ent**,**Fui** and **MTp**, we perform the clustering and the model we got is almost the same with the model considering 10 variables, where ARI=0.43 and MMM=0.77, which also indicates the influence of **Ven** and **ICV** can be ignored. The Table 13 also shows the new weight of variables after removing two low-weight variables. In the table, we can find that **S13**,**mPd** and **mPt** still have high weight and **WhB** is the least important for clustering.

| The weights of variables estimated by **Sparcl** | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | S13 | mPd | mPt | Ven | Hip | WhB | Ent | Fui | MTp | ICV |
| Weight | 0.469 | 0.526 | 0.525 | **0.034** | 0.286 | 0.104 | 0.285 | 0.142 | 0.173 | **0.039** |
| The weights of variables estimated by **Sparcl** after removing **Ven** and **ICV** | | | | | | | | | | |
| Weight | 0.473 | 0.527 | 0.526 | 0 | 0.285 | 0.100 | 0.286 | 0.142 | 0.169 | 0 |

Table 13: The weights of variables estimated by **Sparcl**.

Considering the gender of patients we also estimate the **ARI** and **MMM** with respect to gender, where Table 14 shows the **ARI** and **MMM** results for females and males as well as for a different marital status. According to Table 14, we find by most of the approaches, the performance of these approaches for females is better than that of the male while the performance of married patients is worse than the non-married patients. This may be due to the relatively smaller proportion of female **MCI** patients and of non-married **MCI** in our dataset and the fact that we know that these approaches do not perform very well for **MCI** discrimination. Comparing the result of **Clustvarsel** and the result of **VSCC** and **GMM**, the **ARI** of male in **GMM** and **VSCC** is a bit higher than that in the **Clustvarsel** but of female is lower than the **Clustvarsel**. Similarly, the **MMM** of male in **VaselLCM** is higher than **Clustvarsel** but for female, it is lower. The reason may because the model estimated by **GMM** and **VSCC** contain more variables, such that **Fui**,**MTp**. According to the boxplots of variables classified by gender shown in Figure 23, there is no significant difference between the two disease status **CN** and **MCI** for females in these variables, only for males. For female patients, containing these variables may mislead the clustering result.

| | ARI | | MMM | | ARI | | MMM | |
|---|---|---|---|---|---|---|---|---|
| | male | female | male | female | married | non-married | married | non-married |
| GMM | 0.23 | 0.38 | 0.58 | 0.64 | 0.26 | 0.36 | 0.59 | 0.67 |
| Sparcl | 0.43 | 0.47 | 0.78 | 0.76 | 0.40 | 0.53 | 0.76 | 0.82 |
| Clustvarsel | 0.21 | 0.39 | 0.57 | 0.66 | 0.27 | 0.37 | 0.60 | 0.67 |
| VarSelLCM | 0.36 | 0.37 | 0.74 | 0.65 | 0.32 | 0.40 | 0.70 | 0.73 |
| VSCC | 0.23 | 0.38 | 0.58 | 0.64 | 0.26 | 0.36 | 0.59 | 0.67 |
| SelvarMix | 0.18 | 0.34 | 0.51 | 0.63 | 0.21 | 0.35 | 0.53 | 0.66 |

Table 14: The **ARI** and **MMM** of different approaches for complete dataset $\textbf{ADNI}_{10}$ classified by gender and marital status when **K=3**.

**Performance assuming K=4 clusters:** According to Table 10, some approaches perform better when K=4. We also set K=4 to see the clustering result, shown in Figure 30. **SelvarMix** and **VSCC** have a better result when K=4. For **SelvarMix**, it discards **WhB** but the **ARI** is improved, which indicates that **WhB** might not be helpful in the clustering. The boxplot in Figure 21 shows that there is no significant difference between the three disease status. Compared with the result of K=3, **VSCC** chooses only two variables but gets the highest **ARI**, which is quite reasonable. As we mentioned before, the value of three disease status in these two variables **mPd** and **mPt** represent differently and it is useful for clustering. However, **Clustvarsel** also choose two variables but the results were not good. The reason is that, according to the pairsplot and boxplot in Figure 17 and 21 **Ven** did not help much for clustering compared to **mPt**. Furthermore, we found that **Sparcl** is very sensitive to the number of different clusters.

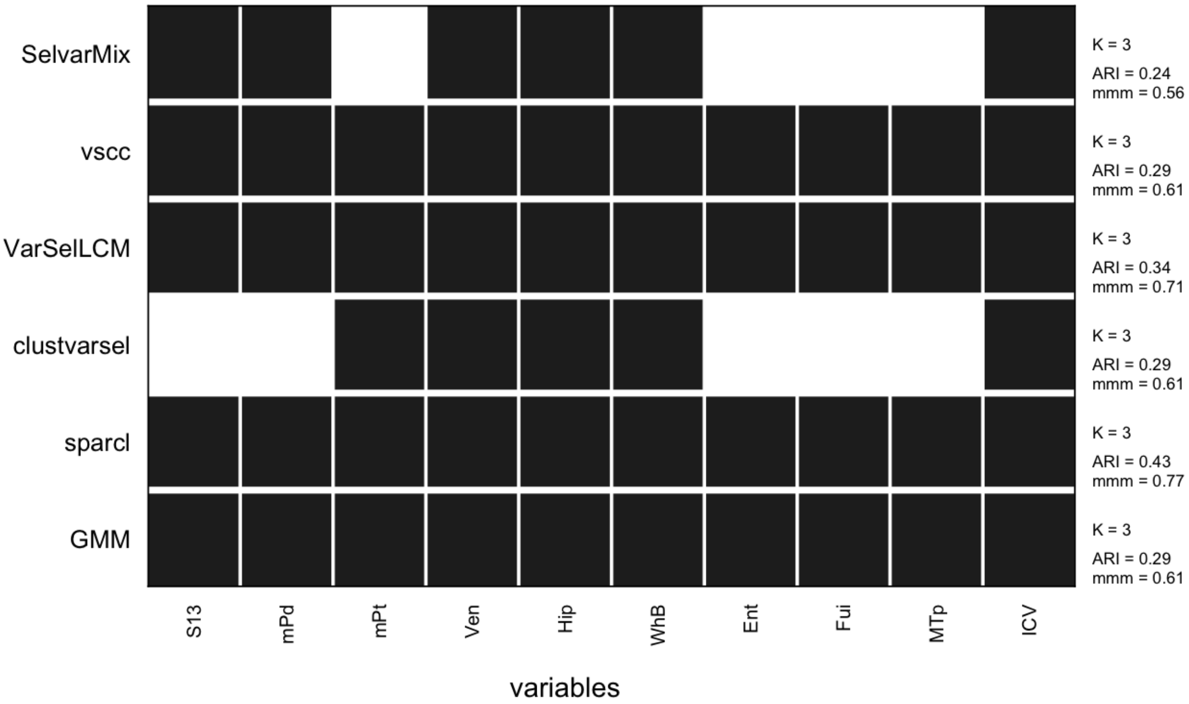


Figure 29: Selected variables(black) chosen according to variable selection approaches indicated in the row using $\mathbf{ADNI}_{10}$ dataset assuming **K=3** clusters.

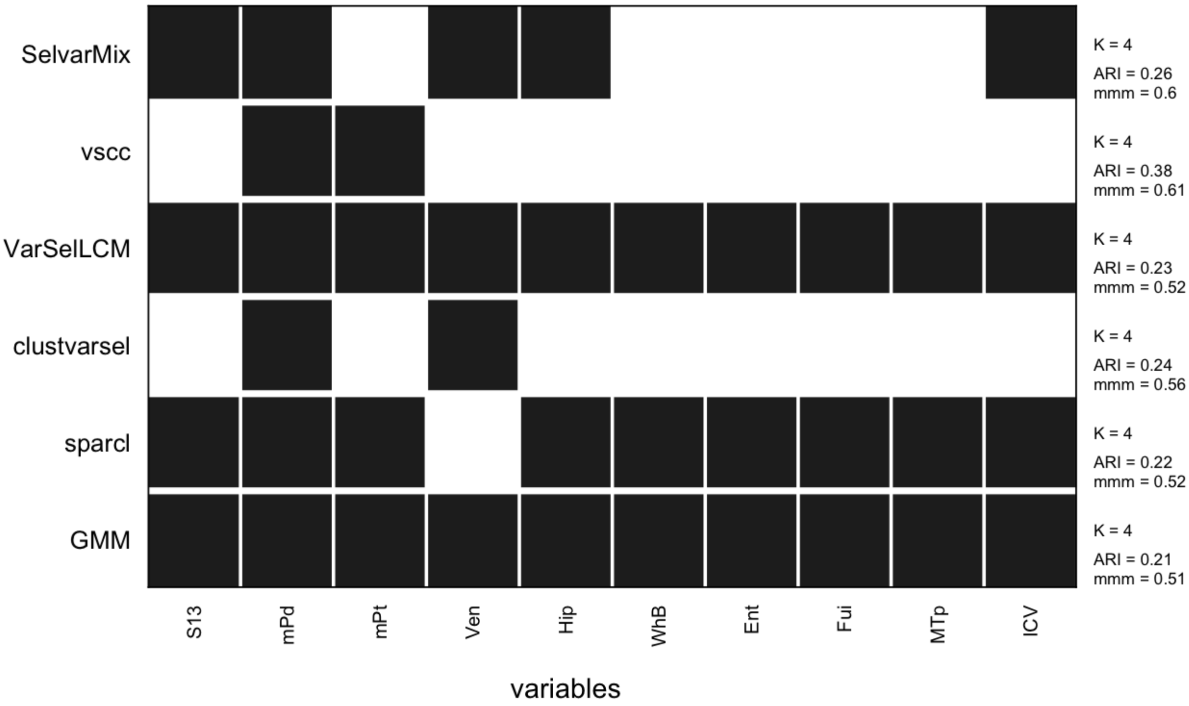


Figure 30: Selected variables(black) chosen according to variable selection approaches indicated in the row using $\mathbf{ADNI}_{10}$ dataset assuming **K=4** clusters.

Table 15 shows the partition of 6 approaches after specifying K=4. Comparing the partition when K=3 in Table 12, we can find that for most of the approaches except **GMM**, the majority **MCI** and **DEM** are no longer clustered in the same component. The reason why the **MMM** are still not satisfactory may be because the true number of components is 3 and for each real disease state, they are clustered into two clusters instead of concentrating in just one. As we can see in **Sparcl**, only 52% of **CN** are clustered to the one component $C_4$ but if we specify K=3, as shown in Table 12, then 98% of **CN** will be distinguished into the same component. The same is true for **VSCC**, the percentage of **CN** that clustered into same cluster decreased from 91% to 77% and of **MCI** decreased from 63% to 53% while for **DEM**, the percentage increased significantly from 9% to 54%.

Sparcl

| | $C_1$ | $C_2$ | $C_3$ | $C_4$ |
|---|---|---|---|---|
| CN | 95 | 2 | 0 | **106** |
| DEM | 1 | 42 | **86** | 0 |
| MCI | 49 | **142** | 66 | 55 |

GMM

| | $C_1$ | $C_2$ | $C_3$ | $C_4$ |
|---|---|---|---|---|
| CN | 13 | 49 | 2 | **139** |
| DEM | **11** | 5 | 113 | 0 |
| MCI | 35 | 60 | **178** | 39 |

Clustvarsel

| | $C_1$ | $C_2$ | $C_3$ | $C_4$ |
|---|---|---|---|---|
| CN | 21 | 3 | 1 | **178** |
| DEM | 22 | 37 | **70** | 0 |
| MCI | 58 | **112** | 82 | 60 |

VarselLCM

| | $C_1$ | $C_2$ | $C_3$ | $C_4$ |
|---|---|---|---|---|
| CN | 0 | **107** | 93 | 3 |
| DEM | **77** | 0 | 0 | 52 |
| MCI | 66 | 44 | 50 | **152** |

VSCC

| | $C_1$ | $C_2$ | $C_3$ | $C_4$ |
|---|---|---|---|---|
| CN | 45 | 0 | 1 | **157** |
| DEM | 1 | **70** | 58 | 0 |
| MCI | 124 | 6 | **166** | 16 |

SelvarMix

| | $C_1$ | $C_2$ | $C_3$ | $C_4$ |
|---|---|---|---|---|
| CN | **191** | 0 | 1 | 11 |
| DEM | 2 | **41** | 79 | 7 |
| MCI | 104 | 17 | **160** | 31 |

Table 15: Partition results for the complete dataset **ADNI**$_{10}$ for the 6 approaches and the number of clusters is chosen to **K=4**.

## 4.2 Analysis of the complete dataset ADNI$_{10}$ based on in-sample and out-of-sample

To perform out-of-sample analyses, we randomly sample 80% of our data as the training data, denoted as **ADNI$_{10}$_80%**, and 20% as the testing data, denoted as **ADNI$_{10}$_20%**. It contains 515 training data and 129 testing data. First, we use the 6 approaches above to train the dataset with cluster number K=3 and get the results. The results is shown in Figure 31. **SelvarMix** selects fewest variables. **VSCC**, **GMM** and **Clustvarsel** perform similarly, but **Clustvarsel** chooses fewer variables, making it more concise. Besides, according to the situation we mentioned before, that is the weight of **Sparcl** may very close to zero, we check the weight of variables that estimate by **Sparcl**, and we found that the weight of **Ven** and **ICV** are still quite small, both of them less than 0.03, see Table 16. Therefore, we remove them and cluster the observations again. The result have similar **ARI** and **MMM** in training dataset, result denoted as **Sparcl***.

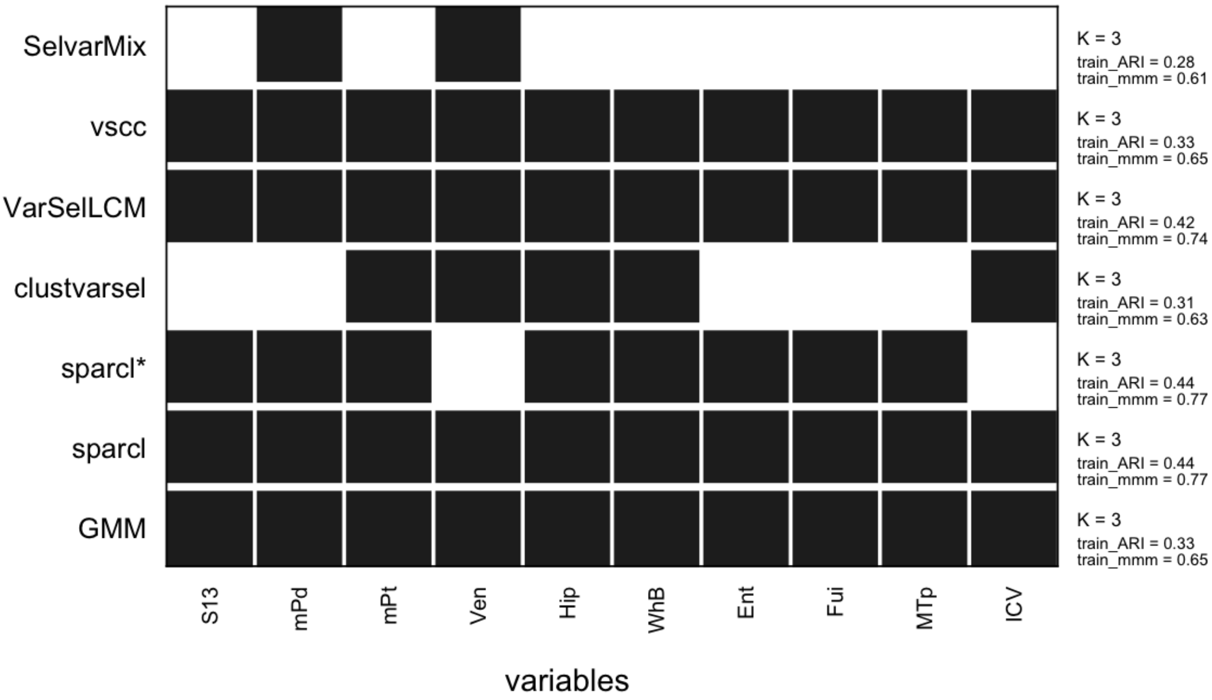


Figure 31: Selected variables(black) are chosen according to variable selection approaches indicated in the row using **ADNI$_{10}$_80%** dataset as training data assuming **K=3** clusters.

| | S13 | mPd | mPt | Ven | Hip | WhB | Ent | Fui | MTp | ICV |
|---|---|---|---|---|---|---|---|---|---|---|
| First time | 0.47 | 0.53 | 0.54 | 0.03 | 0.28 | 0.1 | 0.27 | 0.13 | 0.16 | 0.03 |

Table 16: Weight of each variable estimated by **Sparcl** using the training dataset **ADNI$_{10}$_80%** when K=3.

In the training process, we estimate the Gaussian mixture model for clustering and get the mean matrix and covariance matrix of each cluster $C_k$, where $K = 1, 2, 3$. Then

| approaches | ARI | MMM |
|---|---|---|
| SelvarMix | 0.25 | 0.54 |
| VSCC | 0.27 | 0.55 |
| VarSelLCM | 0.41 | 0.74 |
| Clustvarsel | 0.28 | 0.57 |
| Sparcl | **0.44** | **0.76** |
| Sparcl* | **0.46** | **0.78** |
| GMM | 0.27 | 0.55 |

Table 17: **ARI** and **MMM** of different approaches for test dataset, where Sparcl* is the clustering result after removing the variables(**Ven** and **ICV**) whose weight is close to zero.

Sparcl

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | **41** | 0 | 0 |
| DEM | 0 | **26** | 7 |
| MCI | 17 | 6 | **32** |

Clustvarsel

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | 4 | **37** | 0 |
| DEM | 5 | 0 | **28** |
| MCI | **9** | 19 | 27 |

VarselLCM

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | **41** | 0 | 0 |
| DEM | 0 | 7 | **26** |
| MCI | 16 | **29** | 10 |

Table 18: Partition results for the 3 approaches under test dataset and the number of clusters is chosen to **K=3**.

by using these parameters, we can calculate the loglikelihood of each cluster for each observation $\boldsymbol{x}_i$ in the testing dataset. Since larger loglikelihood $L_{ik}$ for observation $\boldsymbol{x}_i$ in cluster $g$ means that the observation $\boldsymbol{x}_i$ tend to be grouped into that cluster $g$. Therefore, we assigned the observation $\boldsymbol{x}_i$ in the testing dataset to the cluster that has the highest loglikelihood. Furthermore, to evaluate the performance of the clustering, we will estimate the **ARI** and **MMM** for each approach, which is shown in Table 17 as follow.

As Table 17 shows, the performance of the **Sparcl** is the best with highest **ARI**=0.44 and **MMM**=0.76. After removing the variables whose weight is close to zero, the **ARI** and **MMM** improved, which is shown in the result of **Sparcl***. **VSCC** and the **GMM** have the same results. **Clustvarsel** works well since it choose the fewer variables than **GMM** but the result of **ARI** is good and **MMM** is also higher. From the Table 18, **Clustvarsel** can separate **CN** well but still not work for identifying **DEM** and **MCI** since the majority of these two disease status are clustered in the same component. However, **Sparcl** can distinguish **DEM** and **MCI** better. Therefore, the penalization- and distance based approach is better suited to our dataset than the classic **GMM**. Besides, comparison of the result from **VSCC**, **GMM** and **VarselLCM**, the clustering approach based on MICL-criterion performs better than the BIC-based model selection approach.

## 4.3 Variable selection performance for a 5-dimension variable subset ADNI$_5$

Based on our analysis and the results of the variable selection for each approach, we found that these approaches select some variables that are highly correlated with the clustering, including **mPd**, **mPt** and **Hip** but also some variables that are not relevant with the clustering, such as **Ven**, making our clustering results not satisfactory and only distinguishing between **CN** and **Non-CN**. Therefore, we reduce the variables to 5 variables as follows: **S13**, **mPd**, **mPt Hip** and **Ent** and denoted the new dataset as **ADNI**$_5$, since they seem to group the patients marginally by observing the boxplots in Figure 21. As before, since **Sparcl** needs to know the size of K in advance, we will choose the number of clusters calculated by **mclust**, that is **GMM** in figure.

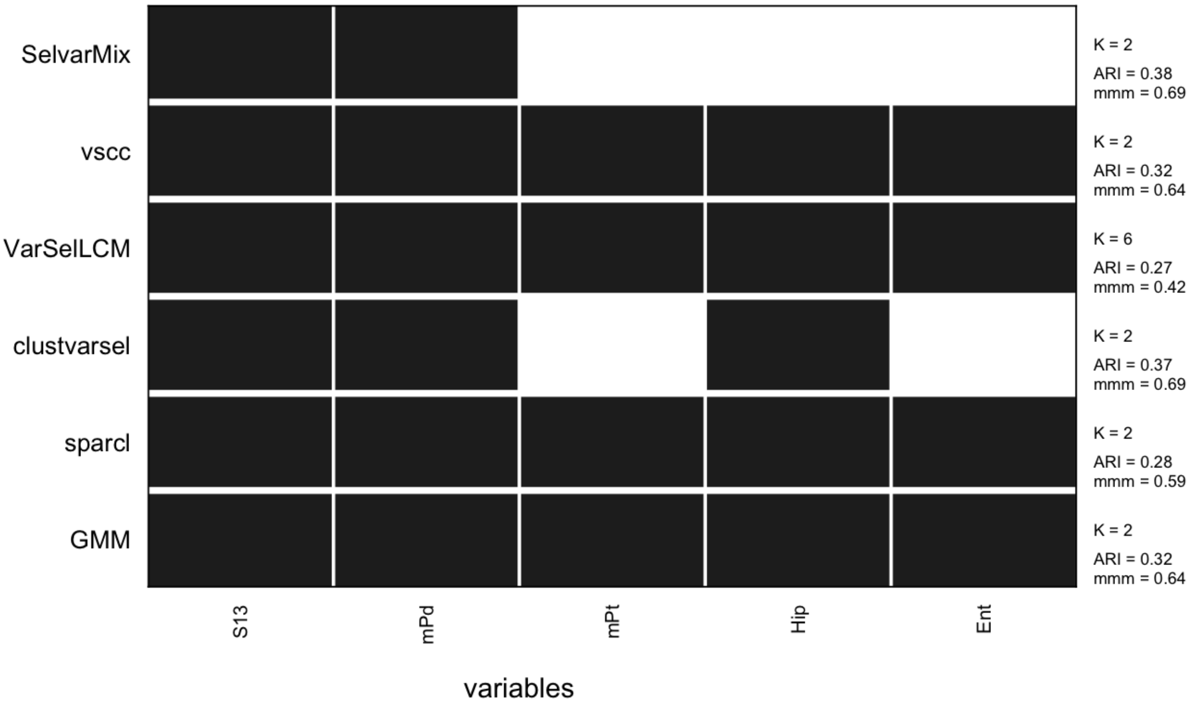


Figure 32: Selected variables(black) chosen according to variable selection approaches indicated in the row using **ADNI**$_5$ dataset allowing for **K=2 to 10** clusters.

As Figure 32 shown, results of **ARI** and **MMM** are significantly improved for all approaches except **Sparcl**. The number of clusters chosen by the approaches is almost K= 2. As before, **VarselLcm** tends to choose a larger number of clusters therefore the value of **MMM** is very low even though it chooses all variables. The performance of **clustvarsel** and **SelvarMix** are great since it is parsimonious, which just choose the fewest variables but with high **ARI** and **MMM**. Comparing **Clustvarsel** with **SelvarMix**, both of which use the same approach of clustering as **GMM** after performing variable selection, they perform better than **GMM** which clusters with all the variables. It indicates that there are still redundant or irrelevant variables that affect the clustering result among these five variables. Furthermore, as we mentioned before **S13** and **mPd** are relevant variables to distinguish the three disease status, which is why it is selected by all these 6 approaches.

For most approaches, however, they choose 2 as the optimal number of clusters rather than 3, which is the true number of components. Therefore, to discover which disease status the two clusters correspond to, we can see the partition result estimated by these 6 approaches, shown in Table 21. As we have seen, for both **CN** and **MCI** patients, they can be clustered into one component completely. But for **MCI**, it is not exactly gathered at the same cluster and therefore cannot be distinguished very well. The Figure 33 shows the comparison of the mean of each variable in the group estimated by **SelvarMix** with true classes. As our analysis of the table, the line for $C_2$ is very close to **CN**, but the line for $C_1$ is between **MCI** and **DEM**, indicating an unclear distinction between these two disease status.

Sparcl

| | $C_1$ | $C_2$ |
|---|---|---|
| CN | **203** | 0 |
| DEM | 0 | 129 |
| MCI | 136 | **176** |

GMM

| | $C_1$ | $C_2$ |
|---|---|---|
| CN | **200** | 3 |
| DEM | 0 | 129 |
| MCI | 102 | **210** |

Clustvarsel

| | $C_1$ | $C_2$ |
|---|---|---|
| CN | 13 | 190 |
| DEM | **128** | 1 |
| MCI | 58 | **256** |

VarselLCM

| | $C_1$ | $C_2$ | $C_3$ | $C_4$ | $C_5$ | $C_6$ |
|---|---|---|---|---|---|---|
| CN | 15 | **104** | 0 | 84 | 0 | 0 |
| DEM | 0 | 0 | 55 | 0 | **62** | 12 |
| MCI | **104** | 3 | 71 | 39 | 5 | 90 |

VSCC

| | $C_1$ | $C_2$ |
|---|---|---|
| CN | **200** | 3 |
| DEM | 0 | 129 |
| MCI | 102 | **210** |

SelvarMix

| | $C_1$ | $C_2$ |
|---|---|---|
| CN | 10 | **193** |
| DEM | 129 | 40 |
| MCI | **254** | 58 |

Table 19: Partition results for **ADNI**$_5$ dataset for the 6 approaches and the number of clusters is allowing for **K=2 to 10**.

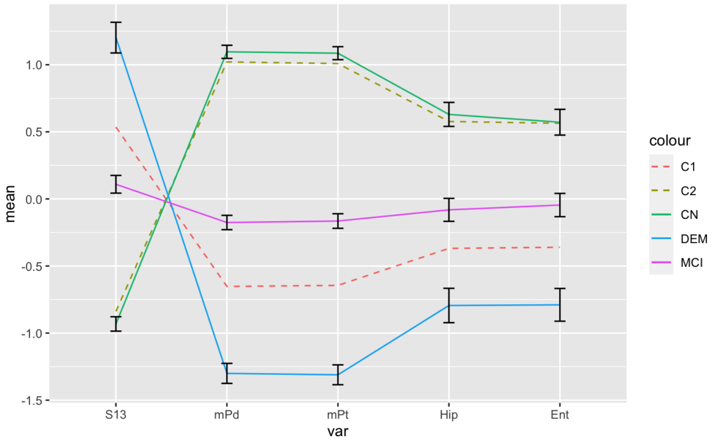


Figure 33: Comparison of the cluster estimated by **SelvarMix** and true class means for dataset **ADNI**$_5$ for K=2; where dotted line represents the mean of each variable in each cluster estimated by the approach and solid line represents the mean of each variable in each disease status.

To compare the results under same number of clusters, we fix the number of clusters K=3 and perform the clustering. Here is the variable selection and clustering performance results when we specify K=3 on the **ADNI**$_5$ dataset, i.e., the dataset that only contains 5 variables (**S13**, **mPd**, **mPt**,**Hip** and **Ent**), see in Figure 34. After setting K=3, the variables used for clustering are unchanged except for **VSCC**. It chooses only two variables instead of all five, but the results, including **ARI** and **MMM**, are significantly improved. Although both **SelvarMix** and **VSCC** select two variables, **SelvarMix** does not perform as well as **VSCC**, which indicates that **S13** is not as effective as **Hip** in clustering. Apart from that, **Clustvarsel** selects one more variable **S13**, but the result is still lower than **VSCC**, which also suggests that **S13** may be a redundant variable if **mPd** and **Hip** have already been selected. Furthermore, when we do clustering under K=3 for the dataset **ADNI**$_{10}$, if we disregard the whole dataset as an option for variable selection by **VSCC**, we got a model with three variables **mPd**,**Ven** and **Hip**, whose **ARI** and **MMM** are 0.22 and 0.54 respectively. Comparing this model to the two-variable model here(incl. **mPd** and **Hip**), we found that the result of the model without variable **Ven** have improved dramatically. For penalization- and distance-based clustering, we check the weights of each variable calculated by **Sparcl**, see in Table 20, and it shows that the first three variables, i.e., **S13**, **mPd** and **mPt**, have higher weights than the last two, i.e., **Hip** and **Ven**, indicating that the first three variables are more helpful for clustering the observations in the **Sparcl** approach, which is somewhat different from the conclusions obtained from other approaches. Besides, since the true number of component is 3, the **MMM** of **Clustvarsel** and **Sparcl** are better when we fix the number of cluster as 3 instead of 2. Comparing with the result on whole dataset under K=3 or **GMM**'s result, we can conclude that the existence of relevant or redundant variables in the **ADNI** dataset causes the model to perform poorly when more variables are present.

| S13 | mPd | mPt | Hip | Ven |
|---|---|---|---|---|
| 0.519 | 0.580 | 0.582 | 0.170 | 0.160 |

Table 20: Weight of each variables estimated by **Sparcl** when **K=3**.

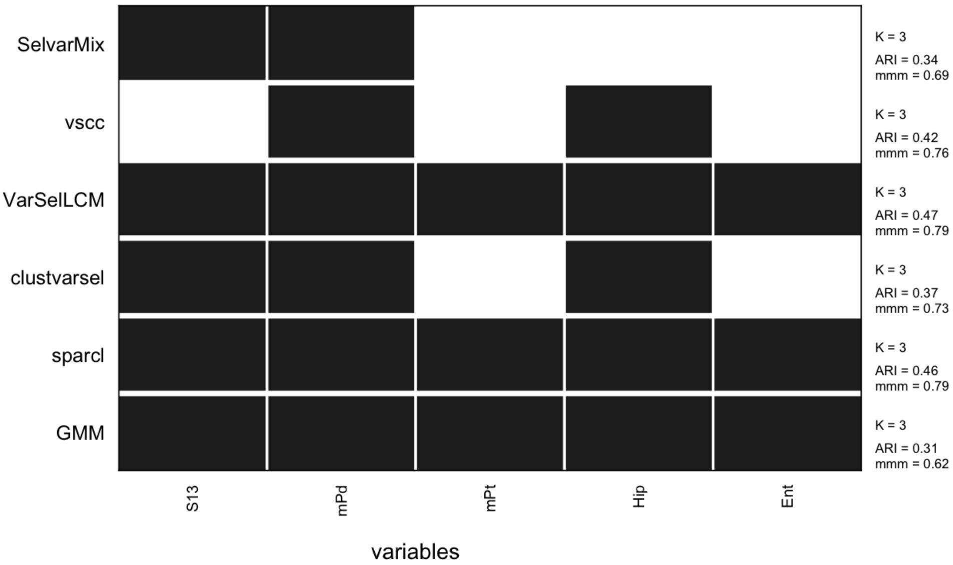


Figure 34: Selected variables(black) chosen according to variable selection approaches indicated in the row using $\mathbf{ADNI}_5$ dataset assuming **K=3** clusters.

Table 21 shows the partition results of 6 approaches. It is clear that for the **Sparcl**, **Clustvarsel**, **VarselLCM** and **VSCC**, the majority of **MCI** and **DEM** are no longer clustered into the same clusters which means it can distinguished the **MCI** better than before. Besides, according to the table, **VSCC** clusters better than **Sparcl** for **MCI** but performs worse for **DEM**.

Figure 35 shows the difference in means of each variable in each cluster and true disease status after reducing variables. Figure 35(a) shows the result of the original Gaussian mixture model with 5 variables. Except for **CN**, the other two clusters $C_1$ and $C_2$ are somewhat different from both **MCI** and **DEM**. Thus it also shows that the interference of redundant variables causes **MCI** and **DEM** to be poorly discriminated. But for distance and weight-based clustering approaches **Sparcl**, shown in Figure 35(b), we can observe that the mean of cluster estimated by **Sparcl** approach is almost the same as the mean of true disease status although it also uses all the variables. Besides, both **Clustvarsel** and **VSCC** use fewer variables, but the curves do not differ much from the true disease status, especially for **CN**, see Figure 35(c) and Figure 35(d). However, for **DEM** and **MCI**, the means of variables **mPd**, **mPt** and **Hip** in group estimated by **Clustvarsel** and **Vscc** are lower than the means in **MCI** and **DEM**.

Sparcl

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | **198** | 0 | 5 |
| DEM | 0 | **103** | 26 |
| MCI | 68 | 38 | **206** |

GMM

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | 5 | 4 | **194** |
| DEM | **26** | 103 | 0 |
| MCI | 89 | **179** | 44 |

Clustvarsel

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | 2 | 0 | **201** |
| DEM | 62 | **65** | 2 |
| MCI | **209** | 70 | 33 |

VarselLCM

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | 3 | **200** | 0 |
| DEM | 28 | 0 | **101** |
| MCI | **209** | 70 | 33 |

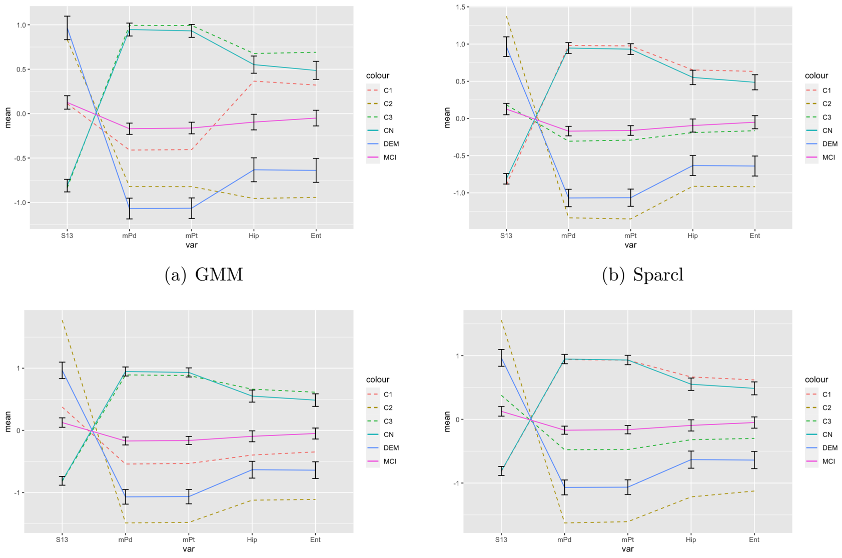
VSCC

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | **199** | 0 | 4 |
| DEM | 0 | **72** | 57 |
| MCI | 83 | 8 | **221** |

selvarMix

| | $C_1$ | $C_2$ | $C_3$ |
|---|---|---|---|
| CN | **201** | 0 | 2 |
| DEM | 0 | **48** | 81 |
| MCI | 101 | 13 | **198** |

Table 21: Partition results for the dataset $\mathbf{ADNI}_5$ for the 6 approaches and the number of clusters is chosen to **K=3**.



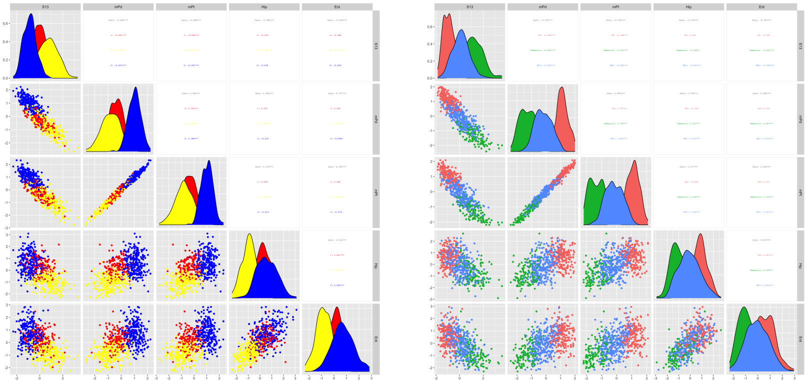
(a) GMM

(b) Sparcl

(c) Clustvarsel

(d) VSCC

Figure 35: Comparison of the cluster and true class means for 5 variables( incl. **S13**, **mPd**, **mPt**,**Hip** and **Ent**) for K=3(dotted line represent the mean of each variable in each clusters estimated by the approach and solid line represent the mean of each variable in each disease status).

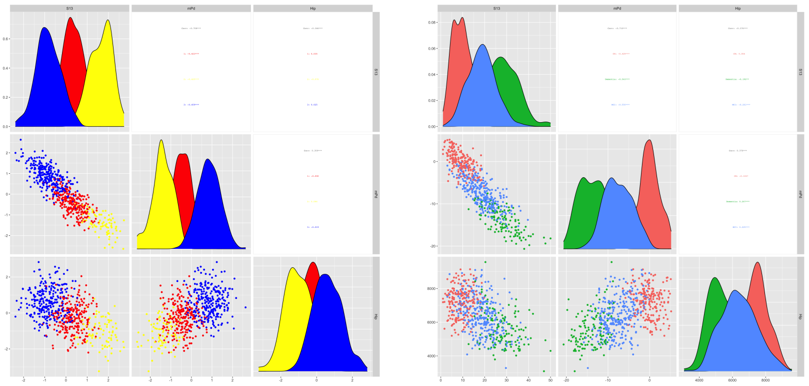


(a) Pairsplot of dataset sampled from the model estimated by **GMM** with 5 variables(**S13**,**mPd**,**mPt**,**Hip** and **Ent**).

(b) Pairsplot of original dataset with 5 variables(**S13**,**mPd**,**mPt**,**Hip** and **Ent**).

Figure 36: Comparison of simulated data sampled from the model estimated by **GMM** and original dataset.

Figure 36(a) shows pairsplot of dataset sampled from the model estimated by **GMM** with 5 variables(**S13**,**mPd**,**mPt**,**Hip** and **Ent**). Comparing with the pairsplot of original dataset, the distributions are not similar in variable **mPd**, **mPt**,**Hip** and **Ent**. However, for the **clustvarsel**, see in Figure 37(a), the distributions of simulated data are more similar. Hence, the **ARI** and **MMM** of the clustering result estimated by **clustvarsel** are higher than **GMM**.

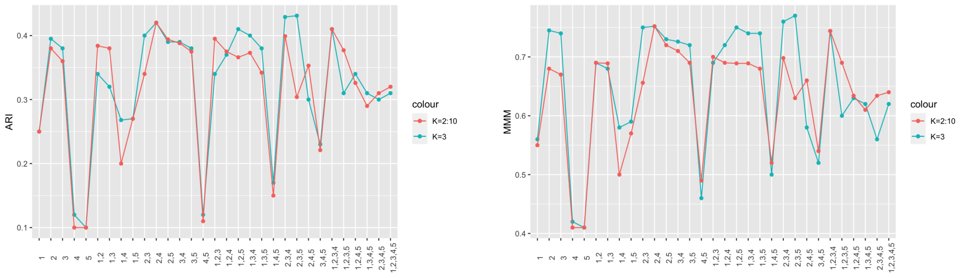


(a) Pairsplot of dataset sampled from the model estimated by **clustvarsel** with 5 variables(**S13**,**mPd** and **Ent**).

(b) Pairsplot of original dataset with 3 variables(**S13**,**mPd**,**Hip** and **Ent**.

Figure 37: Comparison of simulated data sampled from the model estimated by **clustvarsel** and original dataset.

**Optimal variable subset** Furthermore, to confirm our conclusions,i.e., **S13**, **mPd** and **Hip** are both relevant for clustering, we considering the 31 different combinations of these 5 variables and fit GMM again by **mclust** allowing the number of cluster K from 2 to 10, we found that the best result of **ARI** is from the model that contains only **mPd**, **Hip** and its **ARI** is equal to 0.42, with the number of clusters K=3, see the red line in Figure 38(a) and Figure 38(b). After specifying K=3, we fit GMM using these 5 variables again,see the blue line in Figure 38(a) and Figure 38(b). The model with **S13**, **mPd** and **Hip** performs well. Besides, the model containing **mPd**,**mPt** and **Ent** with cluster K=3 has best performance, that is **ARI** =0.43, **MMM**=0.77. According to the weights estimated by **sparcl**, the most relevant variables are **S13**,**mPd** and **mPt**. We can compare this result with the one estimated by **GMM**, the model with these three variables also has satisfactory **ARI** and **MMM**, i.e. **ARI**= 0.34, **MMM**=0.69 .



(a) ARI of different variables combination, where 1: **S13**; 2: **mPd**; 3:**mPt**; 4:**Hip**; 5:**Ent**.

(b) MMM of different variables combination, where 1: **S13**; 2: **mPd**; 3:**mPt**; 4:**Hip**; 5:**Ent**.

Figure 38: ARI and MMM of different variables combination where red represents the best model chosen allowing for **K=1 to 10** clusters and the blue represents the model with the number of cluster **K=3** .

## 4.4 Conclusion

- **Sparcl**, the penalization- and distance-based approach, is the most effective among all approaches for identifying **DEM** and **MCI**. But it needs to know the number of clusters in advance. It always chooses all the variables into consideration, which may because the setting of $s$ in the $L_1$ penalty affects the weight of variables, leading to situations where variables with weights close to, but not equal to, zero occur. For example, when we use the $s = 2.59$ estimated by the algorithm in Section 2.7.1, we will discard none of the variables in the dataset. But if we set $s = 1.2$, the weights of variables will change so that only variables **mPd** and **mPt** will be selected. We may therefore need to confirm whether the weights of the variables are small or choose a better and more appropriate $s$ to adjust the weights of the variables and the number of variables selected. For kmeans-based and GMM approaches, we were expecting an even better performance from GMM. In general, according to Section 9 of Bishop and Nasrabadi [2006], GMM clustering should perform better since it is somewhat more flexible and with a covariance matrix so that we can make the boundaries elliptical, as opposed to circular boundaries with k-means. However, according to our result, the penalization-and kmeans-based approach performs more prominently than the traditional GMM-based approach. The reason may be that it identifies the relevant and irrelevant variables and gives them different weights so that the clustering results are not disturbed too much by the irrelevant variables. Besides, some data are probably non-Gaussian.

- **Clustvarsel**, the approach based on information criterion BIC, has a satisfactory outcome, i.e., it always has better results than **GMM**. Its performance is stable, always removing some variables that are not relevant to the clustering and retaining those that contribute to clustering. Although it does not perform as well as **Sparcl**, which cannot distinguish between patients with **DEM** and **MCI**, it can cluster **CN** perfectly with a small number of variables.

- For **VarselLCM**, the approach based on information criterion MICL, it does not discard any variables in our dataset and an optimal number of clusters is always large. As we mentioned in the Section 2.7.2, this approach assumes that the variables are independent when it estimates the integrated complete-data likelihood. Therefore, this approach does not perform very well for variable selection and choosing the optimal clusters. However, we found that after determining the number of clusters, its clustering results were better than those of the classic GMM models and it can cluster the **MCI** and **DEM** patients into two different components even though both approaches used all variables. This phenomenon may be explained by the fact that for classic **GMM**, it considers a richer family of models and covariance matrix than the **VarselLCM** approach, which may lead to over-fitting. However, the performance of the classic **GMM** is more stable when the exact number of clusters is not known, and the number of clusters obtained by **GMM** will be closer to the true number of clusters, which also implies that the classic **GMM** may be more flexible.

- **VSCC**, the hybrid based on the within-group variance approach, will sometimes selects all variables for clustering though, and the results are the same as those of a traditional GMM. But in many cases, **VSCC** is still able to select the most relevant variables for clustering, making the model more concise. However, it is sensitive to the number of clusters and its results may change when the number of clusters changes significantly.

- **SelvarMix**, the hybrid based on the lasso penalization approach, performs not bad and also removes some variables but its results sometimes tend to be lower than the classic GMM approach that uses all variables for clustering.
- For the analysis of gender and marital status, we found that the performance of these approaches for a female is better than the male while the performance of a married person is worse than the non-married person.

# 5 Analysis of the vine copula mixture model(VCMM) for clustering

As we can see that there are some problems in the clustering with the Gaussian mixture model and the Gaussian mixture model does not cluster the data perfectly, since the **MCI** and **DEM** are always clustered into the same component. We have observed some non-Gaussian dependence between the pair of variables in the Figure 18. Therefore, we will now continue to use the vine copula mixture model to see if the performance can be improved.

First of all, we will fit the data with the vine copula mixture model. For this purpose, we use the univariate Gaussian normal, log-normal, log, logistic, gamma, t-fix, skew-normal and skew-t distributions as possible candidate distributions for the marginal distribution for each variable. Besides, we allow the Gaussian, t, Clayton, Gambel, Frank, Joe, BB1, BB6, and BB8 families of covariates and their rotations as candidates for the family of the pair copula.

## 5.1 Analysis of the complete dataset $\textbf{ADNI}_{10}$

From the analysis of the Gaussian mixture model with the dataset containing 10 variables, $\textbf{ADNI}_{10}$, we find that the clustering model with K=2,3 and 4 fit the dataset better than the others. Therefore, we will use K=2, 3, and 4 to see the performance of the vine copula mixture model on the $\textbf{ADNI}_{10}$ dataset. First, we set the number of clusters K=3, then we fit $\textbf{ADNI}_{10}$ dataset with kmeans as the initial clustering approach. Figure 39 shows the estimated marginal distributions of three clusters. The first tree level of the estimated vine copula model for $\textbf{ADNI}_{10}$ data with three clusters is shown in Figure 40.

Then, we can evaluate the performance of this model by comparing the **ARI** and **MMM** with each patient's disease status. According to the partition result of **VCMM** in Table 22, we can associate the final clusters with the 3 disease status to estimate the classification rate. For **MMM** calculation, we assigned the clusters to the corresponding disease status, that is $C_1 \rightarrow$ **CN**, $C_2 \rightarrow$ **MCI** and $C_3 \rightarrow$ **DEM**. Besides, in Table 23, we can find most of **CN**, **DEM** are clustered to the $C_1$ and $C_2$ but for **MCI** it is split over three clusters and it is hard to distinguish. Besides, as shown in Table 23, most of the **DEM** and **MCI** clustered in the same clusters, and the number of people identified as DEM according to the **Hungarian algorithm** is very low, is only 16. The **ARI** for this 10 variable model is lower, which equals 0.20. It is much worse than the result of the model based on the Gaussian mixture model in Table 10. We, therefore, wonder whether our clustering model has been affected by the existence of redundant and irrelevant variables.

| | 1 | 2 | 3 |
|---|---|---|---|
| CN | **155** | 5 | 43 |
| DEM | 3 | 110 | **16** |
| MCI | 75 | **150** | 87 |
| ARI | 0.20 | | |
| MMM | 0.50 | | |

Table 22: Partition of **VCMM** for **ADNI**$_{10}$ by using kmeans for the initial partition.

| | 1 | 2 | 3 |
|---|---|---|---|
| CN | **0.764** | 0.024 | 0.212 |
| DEM | 0.023 | **0.853** | 0.124 |
| MCI | 0.240 | **0.481** | 0.279 |

Table 23: Percentage of observations in a disease status assigned to the clusters for **ADNI**$_{10}$ by using kmeans for the initial partition.

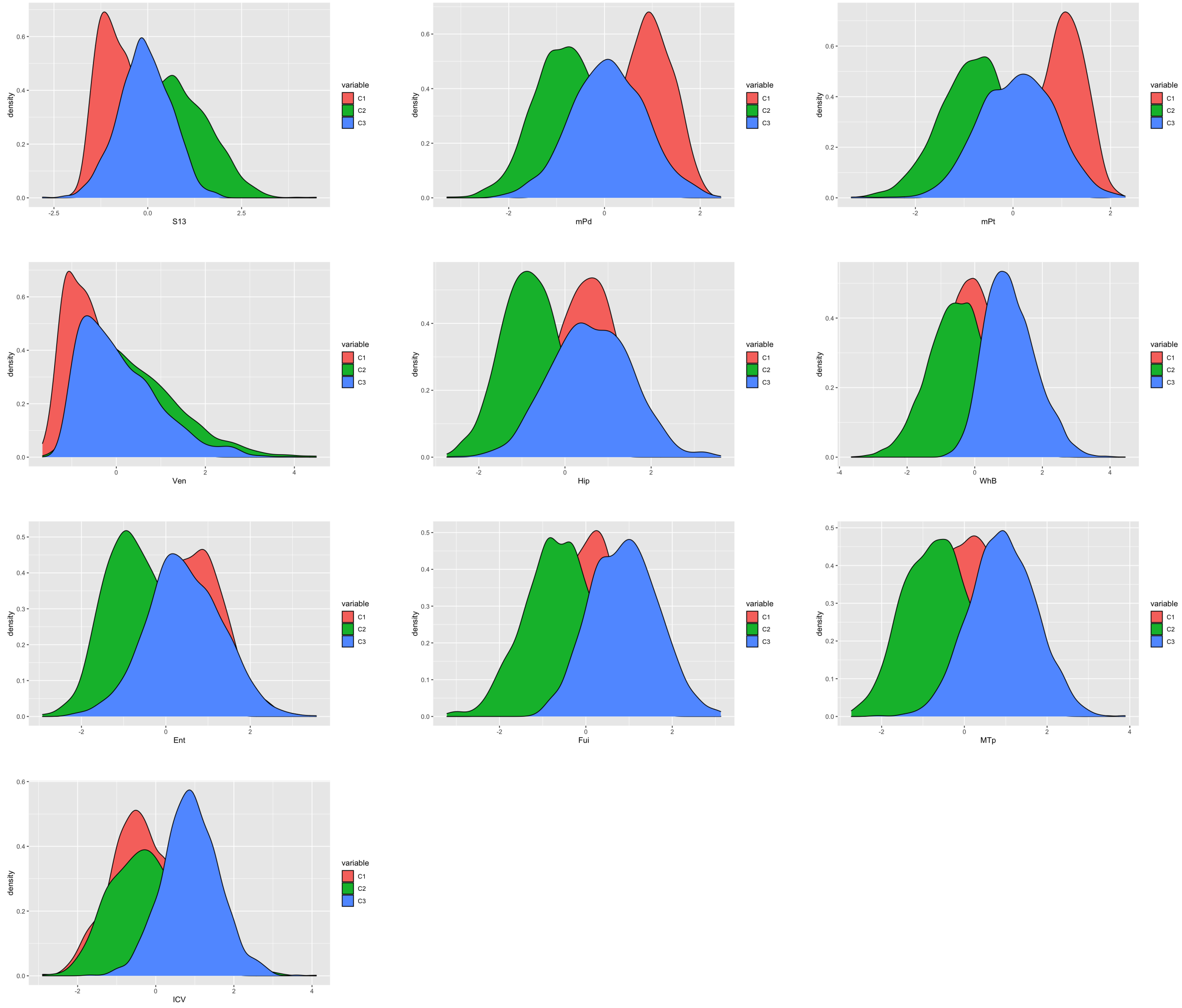
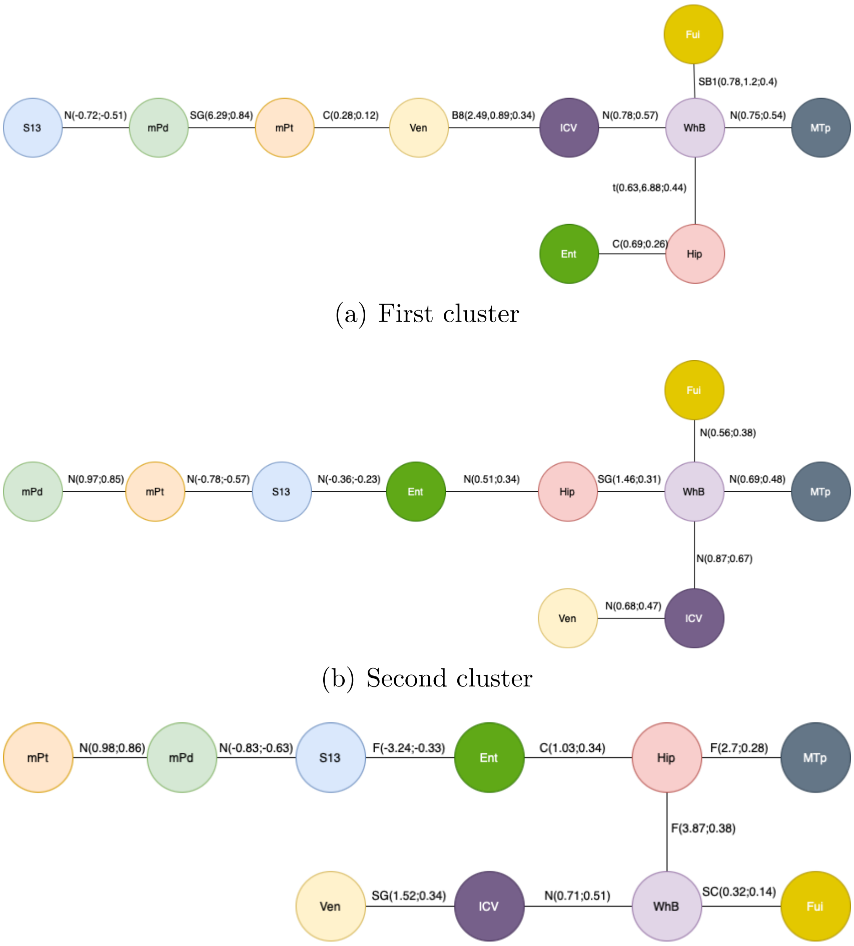


Figure 39: Estimated marginal distributions of three clusters for **ADNI**$_{10}$ using kmeans for the initial partition.



(a) First cluster

(b) Second cluster

(c) Third cluster

Figure 40: The first tree level of the estimated vine copula model for the ADNI dataset with three clusters by using kmeans for the initial partition. A letter at an edge refers to its bivariate copula family, where N: Gaussian; t:t; C: Clayton; SB1:Survival BB1; SC: Survival Clayton; SG: Survival Gumbel; BB1:BB1; F: Frank; BB8:B8. The true parameter value and corresponding Kendall's $\tau$ of the pair copula are given inside the parenthesis (parameter/Kendall's $\tau$).

To consider the initial clustering effect, we also try the Gaussian mixture model clustering as the initial clustering approach and get the following Table 24. The clustering results are also not as good as those previously obtained by the approaches based on GMM. As the table shows, for both approaches, the optimal number of components is K=2. The **ARI** and **MMM** of the models estimated using GMM as an initial clustering approach are higher in most cases.

Tables 25 and 26 show the partition result by using kmeans and GMM as initial clustering approach respectively. From the tables, it is obvious that the two components that the model only separates out are **CN** and **non-CN**. In the kmeans approach, the **MCI** is spread over two clustering points but in GMM approach, it is concentrated on the same clusters as **DEM**.

| | ARI | MMM |
|---|---|---|
| kmeans as initial clustering | | |
| K=2 | 0.22 | 0.56 |
| K=3 | 0.20 | 0.50 |
| K=4 | 0.16 | 0.46 |
| GMM as initial clustering | | |
| K=2 | 0.31 | 0.65 |
| K=3 | 0.25 | 0.59 |
| K=4 | 0.15 | 0.42 |

Table 24: Clustering results of the **VCMM** for **ADNI**$_{10}$ using different initial clustering approaches.

| | 1 | 2 |
|---|---|---|
| CN | **198** | 5 |
| DEM | 13 | 116 |
| MCI | 143 | **169** |
| ARI | 0.22 | |
| MMM | 0.56 | |

Table 25: Partition of **VCMM** for **ADNI**$_{10}$ by using kmeans for the initial partition when **K=2**.

| | 1 | 2 |
|---|---|---|
| CN | **184** | 19 |
| DEM | 0 | 129 |
| MCI | 74 | **238** |
| ARI | 0.31 | |
| MMM | 0.65 | |

Table 26: Partition of **VCMM** for **ADNI**$_{10}$ by using GMM for the initial partition when **K=2**.

## 5.2 Performance of VCMM on a subset of variables ADNI$_5$

Considering the effect of possible redundant or irrelevant variables and according to the pairsplot in Figure 17 and boxplots in Figure 21, we first choose the 5 variables(**S13**, **mPd**, **mPt**,**Hip** and **Ent**) to do the clustering by vine copula model and as we mentioned before, the dataset is denoted as **ADNI**$_5$. As before, we try different initial clustering approaches and compare the performance of these two approaches. Unlike the result for **ADNI**$_{10}$, the results obtained using kmeans as an initial clustering approach are generally better than those of GMM. In particular, we get the best **ARI** and **MMM** when K = 3.

| | ARI | MMM |
|---|---|---|
| kmeans as initial clustering | | |
| K=2 | 0.31 | 0.63 |
| K=3 | 0.40 | 0.75 |
| K=4 | 0.36 | 0.64 |
| GMM as initial clustering | | |
| K=2 | 0.38 | 0.69 |
| K=3 | 0.35 | 0.65 |
| K=4 | 0.34 | 0.64 |

Table 27: Clustering results of **VCMM** using different initial clustering approaches for **ADNI**$_5$

Figure 41 shows the first level of the vine tree structure of **ADNI** data for each cluster using kmeans as the initial clustering approach when K=3. The estimated vine tree structure is different for **MCI** and the other two disease status since it is a D-vine structure. As we can see the partition of **ADNI** in Table 28, the **CN** is perfectly identified and clustered to $C_2$. Unlike the result in the 10-variables model, the cluster with the highest percentage of **DEM** and **DEM** are also no longer the same, indicating its performance is improved after reducing the variables. Besides, the **ARI** and **MMM** is also outperformed most of previous models in Gaussian mixture model except for **Sparcl**, **vscc** and **VarselLCM**.

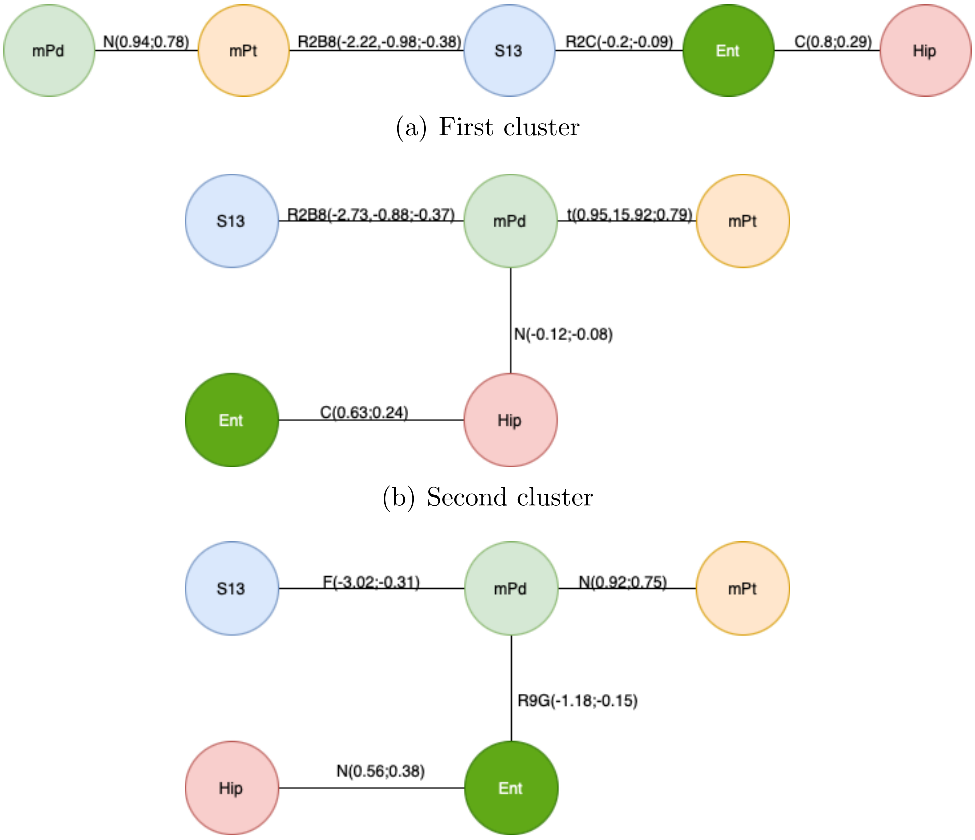
| | 1 | 2 | 3 |
|---|---|---|---|
| CN | 0 | **199** | 4 |
| DEM | **100** | 0 | 29 |
| MCI | 56 | 71 | **185** |
| ARI | 0.40 | | |
| MMM | 0.75 | | |

Table 28: Partition of **VCMM** for the dataset **ADNI**$_5$ by using kmeans for the initial partition when **K=3**.

| | 1 | 2 | 3 |
|---|---|---|---|
| CN | 0 | **0.98** | 0.02 |
| DEM | **0.78** | 0 | 0.22 |
| MCI | 0.17 | 0.23 | **0.60** |

Table 29: Percentage of observations in a disease status assigned to the clusters for dataset **ADNI**$_5$ by using kmeans for the initial partition when **K=3**.

Besides, we can simulate data from the model we estimated by using the 5 variables and we only use kmeans as the initial clustering approach since it performs better, see Figure 42(a), and compare the simulated data to our observations. Figure 43(a) shows a contour plot of the dataset sampled from the model with 5 variables. If the datasets



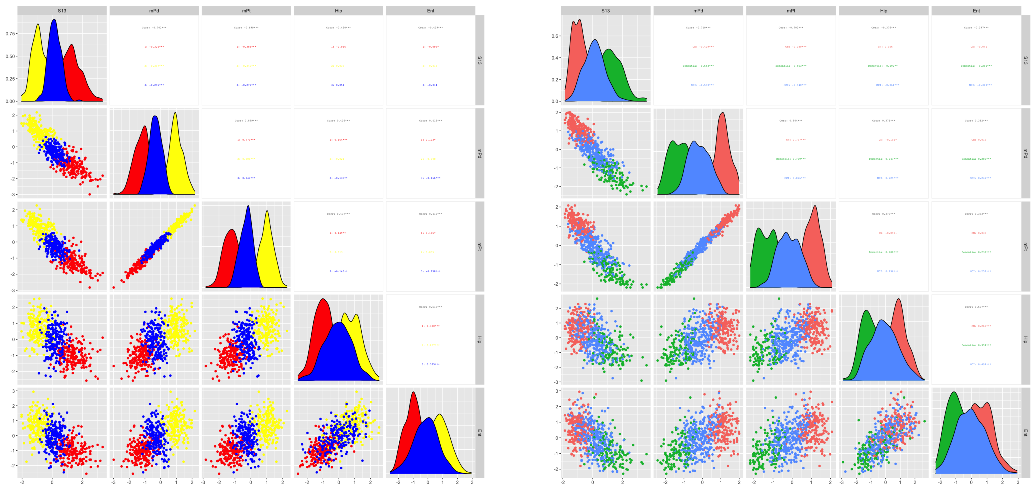
(a) First cluster

(b) Second cluster

(c) Third cluster

Figure 41: First level of vine tree structure of **ADNI**$_5$ data with three clusters **K=3** by using kmeans for the initial partition. A letter at an edge refers to its bivariate copula family, where N: Gaussian; C: Clayton; t:t; F: F; R2B8:Rotated BB8 270 degrees; R2C: Rotated Clayton 270 degrees; R9G: Rotated Gumbel 90 degrees. The true parameter value and corresponding Kendall's $\tau$ of the pair copula are given inside the parenthesis (parameter/Kendall's $\tau$).

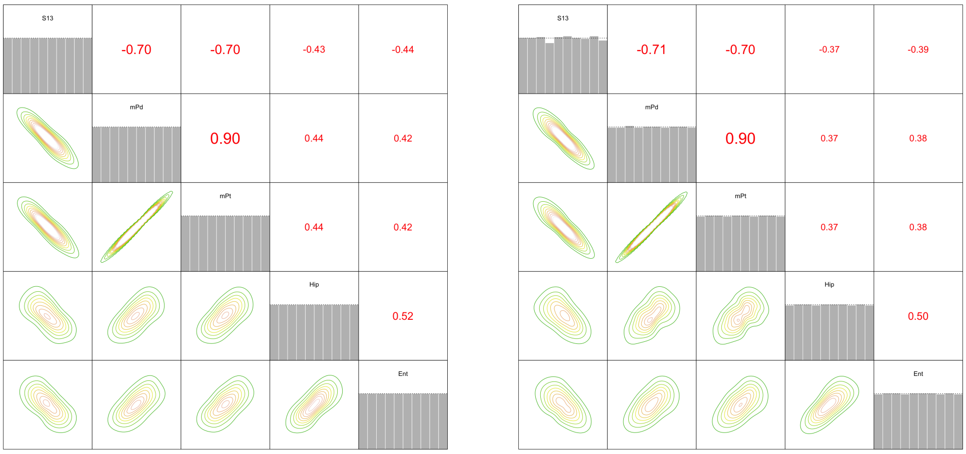
are sampled from the same distribution, then they are likely to look similar. Compare with the Figure 17, we found that the distributions of these variables are similar in some variables, especially for the variable **Ent**. For comparison of contour plots, some are similar but we can still see that there are still some differences between the original and simulated data from this model.



(a) Pairsplot of dataset sampled from the model with 5 variables(**S13**,**mPd**,**mPt**,**Hip** and **Ent**).

(b) Pairsplot of original dataset with 5 variables(**S13**,**mPd**,**mPt**,**Hip** and **Ent**).

Figure 42: Comparison of dataset sampled from the model estimated by **VCMM** with 5 variables and original dataset.



(a) Contour plot of dataset sampled from the model with 5 variables(**S13**,**mPd**,**mPt**,**Hip** and **Ent**).

(b) Contour plot of original dataset with 5 variables(**S13**,**mPd**,**mPt**,**Hip** and **Ent**).

Figure 43: Comparison of contour plot for dataset sampled from the model estimated by **VCMM** with 5 variables and for original dataset.

Then, we can compare the result of the 10-variable model with a 5-variable model to see if the performance improved for any other value of K. By associating with the true disease status, we obtain the **ARI** and **MMM** of the model as well as the loglikelihood and BIC, which shown in Table 30. It is clear that **ARI** and **MMM** of the 5-variables model are much higher than the model estimated by the complete dataset. When we reduced the variables, the result improved considerably, which proved our assumption that our data contained some irrelevant and redundant variables for clustering and considering all variables will lead to over-fitting.

| **ADNI$_{10}$** | | | | |
|---|---|---|---|---|
| | ARI | MMM | loglik | BIC |
| K=2 | 0.22 | 0.56 | -5347 | 11625 |
| K=3 | 0.20 | 0.54 | -5285 | 11960 |
| K=4 | 0.16 | 0.46 | -5260 | 12377 |
| **ADNI$_5$** | | | | |
| | ARI | MMM | loglik | BIC |
| K=2 | 0.31 | 0.63 | -2448 | 5208 |
| K=3 | 0.40 | 0.75 | -2420 | 5275 |
| K=4 | 0.36 | 0.62 | -2404 | 5404 |

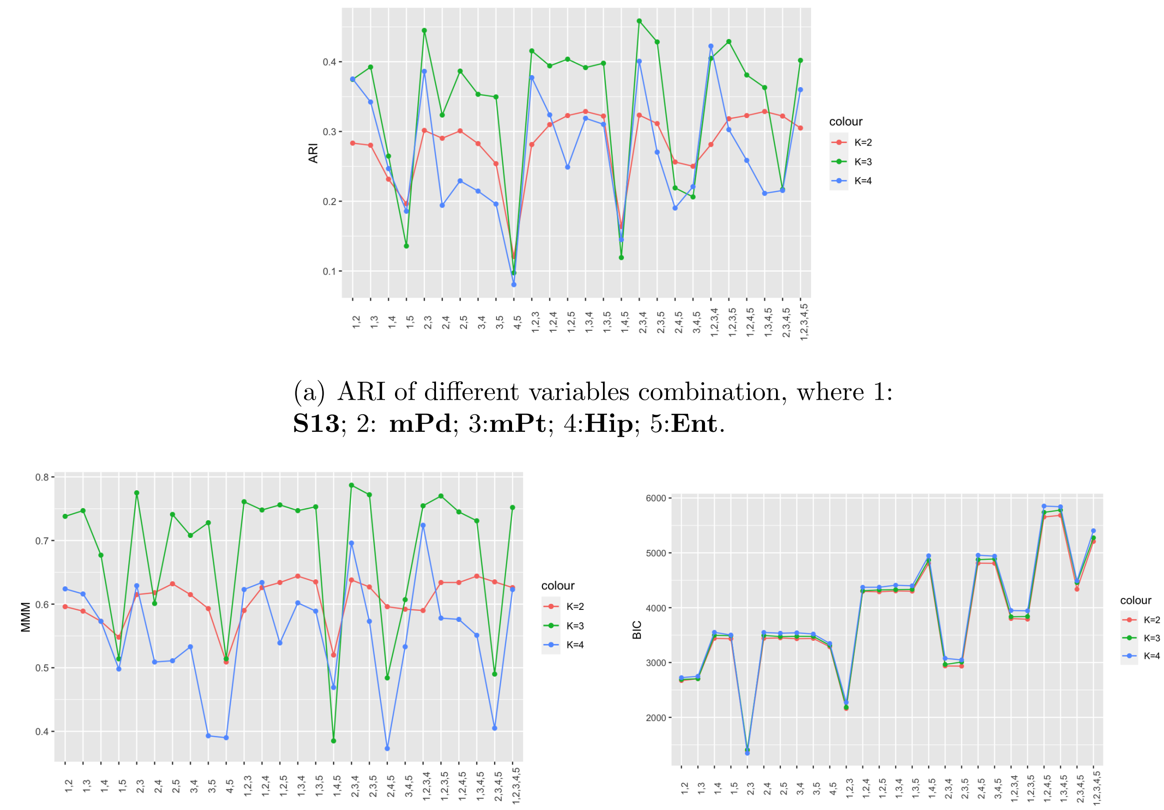
Table 30: Clustering results of a different number of variables allowing several clusters **K=2,3,4** by using kmeans for the initial partition.

### 5.2.1 Exploring variable selection for VCMM

To find out which variables are relevant to clustering, we first fit the observations in **ADNI$_5$** for K=2, K=3, and K=4 by using the different combinations of these 5 variables and use kmeans as initial clustering since it performs better.

To see the performance of each model, we estimated the **ARI** and **MMM** for each models and make comparison, see in Figure 44(a) and Figure 44(b). We can see in the Figure 44(a) and 44(b) that both of them illustrates that the model with K=3 tends to perform better than those with K=2 and K=4 in most cases. And for K=2, the value of **MMM** are in the range between 0.55 and 0.65, which does not fluctuate much with the change of clustering variables. The model containing variable **mPd**,**mPt** and **Hip** has highest **ARI**=0.46 and **MMM**=0.79, which is different from the result estimated by **GMM**, i.e., the model estimated by **GMM** reaches the maximum **ARI**=0.43 and **MMM**=0.77 when it contains **mPd**, **mPt** and **Ent**. Apart from that, clustering result with only two variables, **mPd** and **mPt**, is also the second-highest among all the combinations. The results of the models including **mPd** are satisfactory, which also indicates that **mPd** plays an important role in clustering. This is reasonable since in the pairsplot in Figure 17 and boxplot in Figure 21 the values of different disease status and distributions are distinct in variable **mPd**. In addition, since the most three relevant variables estimated by **sparcl** are **S13**, **mPd** and **mPt**, we also applied **VCMM** to these three variables and the **ARI**=0.42, and **MMM**=0.76.

Figure 44(c) indicates the BIC for each model estimated with each combination of variables. It can be seen that the model with only two variables(**mPd**,**mPt**) has lowest



(a) ARI of different variables combination, where 1: **S13**; 2: **mPd**; 3:**mPt**; 4:**Hip**; 5:**Ent**.

(b) MMM of different variables combination, where 1: **S13**; 2: **mPd**; 3:**mPt**; 4:**Hip**; 5:**Ent**.

(c) BIC of different variables combination, where 1: **S13**; 2: **mPd**; 3:**mPt**; 4:**Hip**; 5:**Ent**.

Figure 44: ARI, MMM, and BIC of different variables combination.

BIC. However, the best variable combination by **ARI** in Figure 44(a) is different, that is the model with **mPd**,**mPt** and **Hip**. The main reason for this may be the greater number of variables included in the model. Comparing the models with the same number of variables, we also found that the smaller the BIC of the model, the higher the **ARI**.

## 5.3 Backward search for variable selection for VCMM

Now, we go back and look at the complete dataset **ADNI**$_{10}$ which has 10 variables. To find variables that are relevant for clustering so that our clusters are not affected by irrelevant and redundant variables, we need a way to filter out the available variables. Here we decide to use the backward-forward search for variable selection, as shown in the Algorithm 6. For the backward search, we start with the model containing all variable. Then, in each following step, we remove the clustering variables that would improve the BIC of the existing model. We iterate until removing any variables in the selected variable set will not have lower BIC. Finally, we can derive the most relevant variables for clustering.

**Algorithm 6** Backward selection

```
Input: Original variable set; Dataset $\mathcal{D} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_D) \in \mathbb{R}^{N \times D}$
Output: Optimal variable set $S$
$S \leftarrow V_{\text{all}}$, where $V_{all}$ is all the variables in dataset$\mathcal{D}$;
For backward search:
while S changes do
    $V_{\text{worst}} \leftarrow \text{argmin}_{V \in S} BIC(S \backslash V)$
    if BIC($S \backslash V$) < BIC($S$) then
        $S \leftarrow S \backslash V_{\text{worst}}$
    end if
end while
```

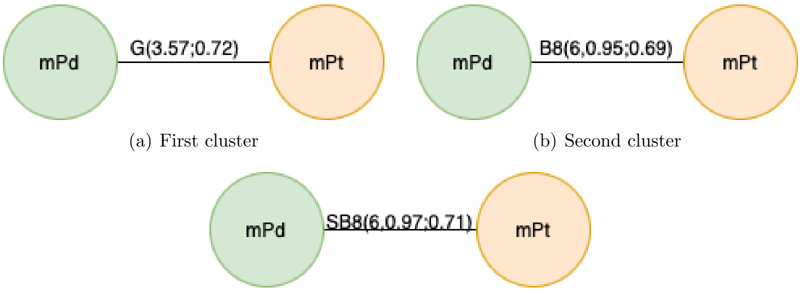
Therefore, to verify the relevant variables for clustering, we applied the backward-forward search to select the variables based on the BIC criterion. From Table 31, the model with **mPd** and **mPt** are chosen due to the lowest BIC and we denoted the dataset containing these two variables as **ADNI**$_2$. Besides, we can find that the first variable to be removed is **ICV** and in the box plot in Figure 21, there is little difference in the performance of this variable between disease status. Other than that it removes **WhB**, **Ven**, **MTp** and **Fui**. When there are five variables left, they are exactly the ones we selected earlier based on the boxplots in Figure 21, which proves our previous suspicions, i.e., several of the excluded variables do not contribute much to the clustering. Table 32 shows the partition of the model contains only **mPd** and **mPt**. According to the partition, it is easy to distinguish the **CN** and **DEM** even though there is still some confusion about the **MCI**. The Figure 47 shows the first level of the vine tree structure for **ADNI** data with clusters K=3.

| | 1 | 2 | 3 |
|---|---|---|---|
| CN | 28 | **175** | 0 |
| DEM | 14 | 0 | **115** |
| MCI | **209** | 27 | 76 |
| ARI | 0.44 | | |
| MMM | 0.77 | | |

Table 32: Partition result for the dataset **ADNI**$_2$ containing **mPd** and **mPt** as well as the **ARI** and **MMM**.

| | iter0 | iter 1 | iter 2 | iter 3 | iter 4 | iter 5 | iter 6 | iter 7 | iter 8 |
|---|---|---|---|---|---|---|---|---|---|
| S13 | √ | √ | √ | √ | √ | √ | √ | √ | |
| mPd | √ | √ | √ | √ | √ | √ | √ | √ | √ |
| mPt | √ | √ | √ | √ | √ | √ | √ | √ | √ |
| Ven | √ | √ | √ | | | | | | |
| Hip | √ | √ | √ | √ | √ | √ | | | |
| WhB | √ | √ | | | | | | | |
| Ent | √ | √ | √ | √ | √ | √ | √ | | |
| Fui | √ | √ | √ | √ | √ | | | | |
| MTp | √ | √ | √ | √ | | | | | |
| ICV | √ | | | | | | | | |

Table 31: Process of the backward search for variable selection based on BIC.



(a) First cluster

(b) Second cluster

(c) Third cluster

Figure 45: Vine tree structure of $\mathbf{ADNI}_{10}$ data with three clusters. A letter at an edge refers to its bivariate copula family, where BB8:BB8; C: Clayton. The true parameter value and corresponding Kendall's $\tau$ of the pair copula are given inside the parenthesis (parameter/Kendall's $\tau$).

We also simulate data from the best fit model we estimate above. Figure 46 shows a contour plot of the dataset sampled from the model with 2 variables. Compare with the Figure 18, the contour plot is similar to those of the original observations.

Figure 47(a), Figure 47(b) and Figure 47(c) show the **ARI**, **MMM** and **BIC** of the best BIC models in each iteration. As we can see, most of the time the model performs better with fewer variables, except when the second variable(**WhB**) is discarded. The results of the model improved a lot when the first variable was removed, indicating that **ICV** hurt the clustering results.

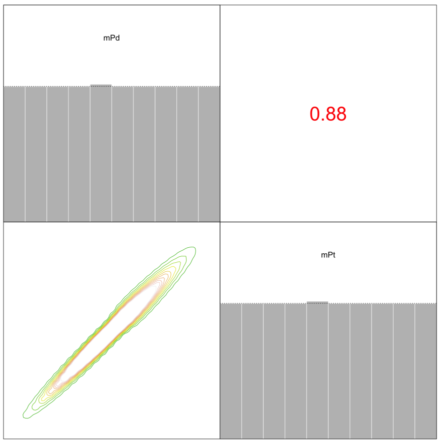
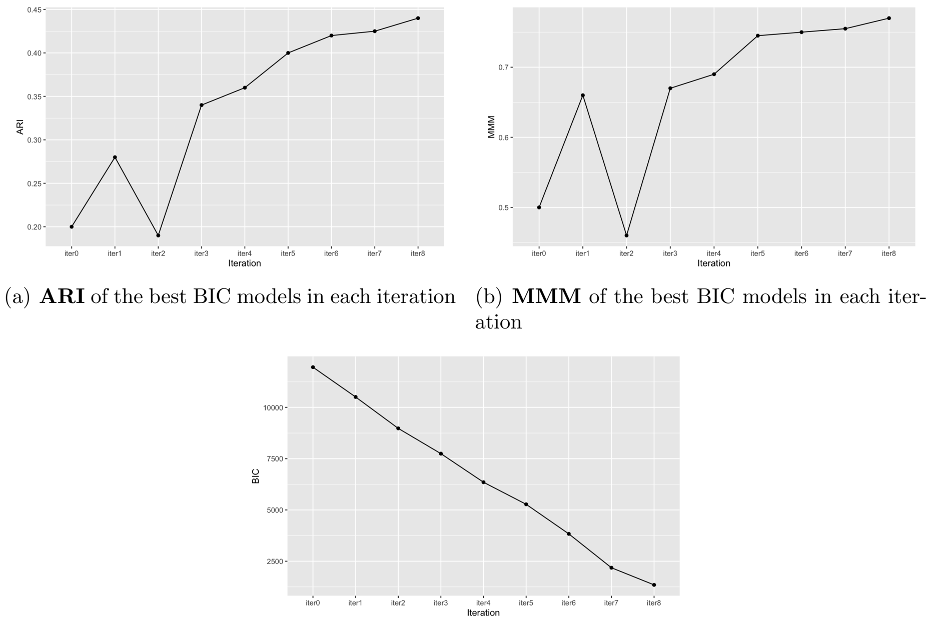


Figure 46: Contour plot of dataset sampled from the model with 2 variables(**mPd** and **mPt**).



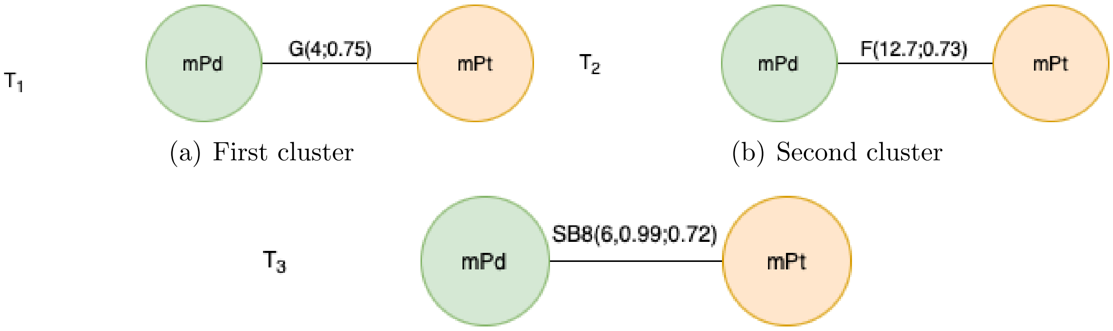
(a) **ARI** of the best BIC models in each iteration

(b) **MMM** of the best BIC models in each iteration

(c) **BIC** of the best BIC models in each iteration

## 5.4 Analysis of complete dataset ADNI$_{10}$ based on in-sample and out-of-sample

In the previous chapters, we do the out-of-sample analysis to evaluate the different Gaussian mixture model clustering approaches. We want to do the same here to compare the result of GMM and **VCMM**. We choose 80% of the dataset **ADNI**$_{10}$ and fit them with the vine copula mixture model under K=3 over all variables and by the backward forward search introduced in the previous section to find out the best model with BIC. The following are the result using the same observations as the example shown in the previous Section 4.2. The model we estimated is the **VCMM** with only **mPd** and **mPt**.



(a) First cluster

(b) Second cluster

(c) Third cluster

Figure 47: Vine tree structure of **ADNI**$_{10}$_80% as training data with three clusters. A letter at an edge refers to its bivariate copula family, where G: Gumbel; F: F; SB8:Survival BB8. The true parameter value and corresponding Kendall's  of the pair copula are given inside the parenthesis (parameter/Kendall's $\tau$).

Then, by using the model we trained before, we test the model using the observations in the test set. We compute the loglikelihood of each observation for each cluster $C_g$ for $K = 1, 2, 3$ and assign them to the cluster that has a higher loglikelihood. Then the result is shown as follows, see Table 33. We can estimate the performance of the result by using **MMM** and **ARI**. According to the table, we found that the model can cluster the patients who are the **CN** and **DEM** but for those who are **MCI**, unlike in the Section 5.3, it was distributed in two components rather than clustered in the one component. To see the reason, we compare the vine structure of the model estimated by the training dataset with the structure of the tree obtained in Section 5.3, we found for the first and third clusters the structure of the tree is similar, but for the second cluster its structure has changed from BB8 to F and the second cluster corresponds to the disease status **MCI**. It means that the tree structure of the second cluster obtained from the training dataset does not match the original data very well.

Compared to the model obtained by GMM, we find that this model gives better results than most of the previous approaches because it chooses the fewest number of variables but obtains better results. Although in the previous chapter, **Sparcl** obtained a model with higher **MMM** and **ARI**, i.e., **MMM**=0.76, ARI=0.44, it contains more variables. In contrast, **VCMM** is more parsimonious, i.e., **VCMM** is the approach that uses the fewest variables but its results are third only to **Sparcl** and **VarSelLcm**. Table

34 shows the partition of **clustvarsel**, which is the approach based on BIC and backward search for GMM. GMM cannot identifies the disease status **MCI** well. Comparing with the results in Section 4.2, **VCMM** does fit our dataset better than most Gaussian model-based clustering approaches. This may be due to its flexibility, taking into account the non-Gaussian relationship.

| | 1 | 2 | 3 |
|---|---|---|---|
| CN | **40** | 1 | 0 |
| DEM | 6 | 2 | **25** |
| MCI | 11 | **20** | 24 |
| ARI | 0.32 | | |
| MMM | 0.65 | | |

Table 33: Partition result of test dataset by **VCMM** as well as **ARI** and **MMM** result using variable **mPd** and **mPt**

| | 1 | 2 | 3 |
|---|---|---|---|
| CN | 4 | **37** | 0 |
| DEM | 5 | 0 | **28** |
| MCI | **9** | 19 | 27 |
| ARI | 0.28 | | |
| MMM | 0.57 | | |

Table 34: Partition of test dataset by **GMM** and backward selection as well as **ARI** and **MMM** result using variable **mPt**, **Ven**, **Hip**,**WhB** and **ICV**.

We also consider the influence on gender and marital status. Table 35 represents the **ARI** and **MMM** results for different gender and marital status. Same as the Gaussian mixture model-based clustering approaches, see in Table 14, **VCMM** performs better for females than males and worse for a married person than non-married. In addition, the difference in **ARI** and **MMM** between males and females decreases when some irrelevant variables are removed.

| | ARI | | MMM | | ARI | | MMM | |
|---|---|---|---|---|---|---|---|---|
| | male | female | male | female | married | non-married | married | non-married |
| VCMM | 0.17 | 0.30 | 0.45 | 0.57 | 0.18 | 0.27 | 0.48 | 0.58 |
| VCMM_5ft | 0.39 | 0.43 | 0.75 | 0.76 | 0.37 | 0.50 | 0.73 | 0.80 |
| VCMM_sel | 0.41 | 0.50 | 0.76 | 0.79 | 0.42 | 0.52 | 0.76 | 0.80 |

Table 35: Estimated **ARI** and **MMM** of models estimated by **VCMM** for stratified dataset by gender and marital status when **K=3**, respectively, where **VCMM_5ft** is the model using **ADNI**$_5$ dataset and **VCMM_sel** is the model estimated by backward-forward approach (including **mPd** and **mPt**).

## 5.5 Comparison of VCMM based with Gaussian model-based clustering results

We applied the variable selection approaches for Gaussian model-based and kmeans clustering. Besides, we also perform GMM clustering without variable selection in Section 4 as well as the VCMM approach in Section 5 to the ADNI dataset. The approach **sparcl** is a penalization kmeans-based method and we consider a rather simple BIC criteria (Ramsey et al. [2008]) given by,

$$\text{BIC} = WCSS + K\ln(n)(d+1)$$

where WCSS denotes the within cluster sums of squares calculate on all variables, K is the number of cluster and d is number of variables. Possible improvement on these rather naive criteria could be in the form of more accurate estimation of the degrees of freedom in the BIC criteria Hofmeyr [2020]. In our comparison, we set K=3 for all models studied.

**Results using ADNI$_{10}$:** Table 36 shows the performance of each approach using ADNI$_{10}$. The penalization-based clustering approach **sparcl** has the highest **ARI** and **MMM** among the results of GMM approaches. Since the **sparcl** contains all variables and some of them have weights close to zero, we removed these variables and got the new model denoted as **sparcl***, which has the almost the same **ARI** and **MMM**. The Gaussian mixture model-based approach without variable selection, **GMM**, performs better than **VCMM**. Although **VarselLCM** consider all variables, it has better results than **GMM**. Besides, **clustvarsel** chose fewer variables but the result is almost the same as the **vscc**. However, if we applied the stepwise algorithm for **VCMM** to select variables described in Section 5.3, we will have the best **ARI** and **MMM**, see **VCMM_sel**.

| Approach | SelvarMix | vscc | clustvarsel | GMM | VarselLCM | sparcl | sparcl* | VCMM | VCMM_sel |
|---|---|---|---|---|---|---|---|---|---|
| ARI | 0.24 | 0.29 | 0.294 | 0.29 | 0.34 | 0.43 | 0.43 | 0.2 | 0.44 |
| MMM | 0.56 | 0.61 | 0.611 | 0.61 | 0.71 | 0.77 | 0.77 | 0.54 | 0.77 |
| BIC | 8048 | 11605 | 7173 | 11605 | 7827 | 4165 | 2896 | 11960 | 1346 |
| Number of free parameters | 84 | 198 | 63 | 198 | 63 | 40 | 32 | 213 | 20 |
| Variables | | | | | | | | | |
| S13 | √ | √ | | √ | √ | √ | √ | √ | |
| mPd | √ | √ | | √ | √ | √ | √ | √ | √ |
| mPt | | √ | √ | √ | √ | √ | √ | √ | √ |
| Ven | √ | √ | √ | √ | √ | √ | √ | √ | |
| Hip | √ | √ | √ | √ | √ | √ | √ | √ | |
| WhB | √ | √ | √ | √ | √ | √ | | √ | |
| Ent | | √ | | √ | √ | √ | √ | √ | |
| Fui | | √ | | √ | √ | √ | √ | √ | |
| MTp | | √ | | √ | √ | √ | √ | √ | |
| ICV | √ | √ | √ | √ | √ | √ | | √ | |
| Number of variables selected | 6 | 10 | 5 | 10 | 10 | 10 | 8 | 10 | 2 |

Table 36: The clustering results and their performance of each approach using ADNI$_{10}$ with all observations.

**Results using ADNI$_5$: VarselLCM** has highest **ARI** and **MMM**. **Sparcl** also has the better performance than most approaches. Compared to the results of ADNI$_{10}$, we found an improvement in the **ARI** and **MMM** of the models estimated by these approaches, especially for **vscc**, see in Table 37. It chose the fewest variables and the **ARI** and **MMM** are the fourth highest. **Clustvarsel** selects one more variable but the result is lower than **vscc**, which indicates that **S13** may be a redundant variable if **mPd** and **Hip** have been selected.

| Approach | SelvarMix | vscc | clustvarsel | GMM | VarselLCM | sparcl | VCMM | VCMM_sel |
|---|---|---|---|---|---|---|---|---|
| ARI | 0.34 | 0.42 | 0.37 | 0.31 | 0.47 | 0.46 | 0.4 | 0.44 |
| MMM | 0.69 | 0.76 | 0.73 | 0.62 | 0.792 | 0.79 | 0.75 | 0.77 |
| BIC | 3407 | 2657 | 4264 | 5144 | 3432 | 1328 | 5275 | 1346 |
| Number of parameters | 18 | 18 | 27 | 63 | 33 | 20 | 68 | 20 |
| Variables | | | | | | | | |
| S13 | √ | | √ | √ | √ | √ | √ | |
| mPd | √ | √ | √ | √ | √ | √ | √ | √ |
| mPt | | | | √ | √ | √ | √ | √ |
| Hip | | √ | √ | √ | √ | √ | √ | |
| Ent | | | | √ | √ | √ | √ | |
| Number of variables selected | 2 | 2 | 3 | 5 | 5 | 5 | 5 | 2 |

Table 37: The clustering results and their performance of each approach using ADNI$_5$ with all observations.

**Results using ADNI$_2$:** Since we used the stepwise algorithm to do the variable selection for **VCMM** and we got the model containing variable **mPd** and **mPt**, then we also applied **GMM** and variable selection approaches for GMM to the dataset **ADNI$_2$** containing these 2 variables and compare their results, see in Table 38. The penalization approach **sparcl** still has the best performance. **VCMM** performs better than **GMM**, which implies that these two variables may have non-Gaussian dependence. For **VSCC** and **clustvarsel**, they only consider one variable **mPt** but the results are quite good.

| Approach | SelvarMix | vscc | clustvarsel | GMM | VarselLCM | sparcl | VCMM |
|---|---|---|---|---|---|---|---|
| ARI | 0.41 | 0.4 | 0.4 | 0.41 | 0.39 | 0.45 | 0.44 |
| MMM | 0.75 | 0.75 | 0.75 | 0.75 | 0.742 | 0.78 | 0.77 |
| BIC | 1272 | 1636 | 1636 | 1272 | 1239 | 696 | 1346 |
| Number of parameters | 18 | 9 | 9 | 18 | 15 | 8 | 20 |
| Variables | | | | | | | |
| mPd | √ | | | √ | √ | √ | √ |
| mPt | √ | √ | √ | √ | √ | √ | √ |
| Number of variables selected | 2 | 1 | 1 | 2 | 2 | 2 | 2 |

Table 38: The clustering results and their performance of each approach using ADNI$_2$ with all observations.

Table 39-Table 41 show the selected variables for the models estimated by different approaches and models' performance using 80% ADNI$_{10}$, ADNI$_5$ and ADNI$_2$ data as the train dataset. **VCMM** with all variables has the worst result but after reducing variables, the performance of **VCMM** improved considerably, which is better than most of the models estimated by the variable selection approaches for the Gaussian mixture model except **VarSelLCM**. This indicates that the data may have been over-fitted. Besides, we also found the performance of **sparcl** is similar in three different dataset. The reason that the results of **sparcl** are stable for ADNI$_{10}$_80%, ADNI$_5$_80% and ADNI$_2$_80% is that the weights estimated by **sparcl** of relevant variables **S13**, **mPd** and **mPt** are always much higher than of the other irrelevant variables. As before, we removed the variables with almost zero weight in ADNI$_{10}$_80% and we will have better test results **ARI**=0.46 and **MMM**=0.78.

| Approach | SelvarMix | vscc | clustvarsel | GMM | VarselLCM | sparcl | sparcl* | VCMM | VCMM_sel |
|---|---|---|---|---|---|---|---|---|---|
| training ARI | 0.28 | 0.33 | 0.31 | 0.33 | 0.42 | 0.44 | 0.44 | 0.26 | 0.38 |
| training MMM | 0.61 | 0.65 | 0.63 | 0.65 | 0.74 | 0.77 | 0.77 | 0.54 | 0.73 |
| test ARI | 0.25 | 0.27 | 0.28 | 0.27 | 0.41 | 0.44 | 0.46 | 0.2 | 0.32 |
| test MMM | 0.54 | 0.55 | 0.57 | 0.55 | 0.74 | 0.76 | 0.78 | 0.43 | 0.65 |
| BIC of the training model | 2819 | 9423 | 5761 | 9423 | 6308 | 3415 | 2402 | 9730 | 1140 |
| Number of free parameters | 18 | 198 | 63 | 198 | 63 | 40 | 32 | 208 | 20 |
| Variables | | | | | | | | | |
| S13 | | √ | | √ | √ | √ | √ | √ | |
| mPd | √ | √ | | √ | √ | √ | √ | √ | √ |
| mPt | | √ | √ | √ | √ | √ | √ | √ | √ |
| Ven | √ | √ | √ | √ | √ | √ | √ | √ | |
| Hip | | √ | √ | √ | √ | √ | √ | √ | |
| WhB | | √ | √ | √ | √ | √ | √ | √ | |
| Ent | | √ | | √ | √ | √ | √ | √ | |
| Fui | | √ | | √ | √ | √ | √ | √ | |
| MTp | | √ | | √ | √ | √ | √ | √ | |
| ICV | | √ | √ | √ | √ | √ | | √ | |
| Number of variables selected | 2 | 10 | 5 | 10 | 10 | 10 | 8 | 10 | 2 |

Table 39: The clustering results and their performance of each approach using using ADNI$_{10}$_80% for training and ADNI$_{10}$_20% for testing with all observations.

| Approach | SelvarMix | vscc | clustvarsel | GMM | VarselLCM | sparcl | VCMM | VCMM_sel |
|---|---|---|---|---|---|---|---|---|
| training ARI | 0.35 | 0.35 | 0.35 | 0.33 | 0.5 | 0.51 | 0.36 | 0.38 |
| training MMM | 0.7 | 0.7 | 0.7 | 0.66 | 0.8 | 0.8 | 0.72 | 0.73 |
| test ARI | 0.28 | 0.28 | 0.28 | 0.27 | 0.43 | 0.44 | 0.3 | 0.32 |
| test MMM | 0.59 | 0.59 | 0.59 | 0.57 | 0.77 | 0.77 | 0.61 | 0.65 |
| BIC of the training model | 2145 | 2145 | 2145 | 4163 | 2663 | 1094 | 4323 | 1140 |
| Number of free parameters | 18 | 18 | 18 | 63 | 33 | 20 | 65 | 20 |
| Variables | | | | | | | | |
| S13 | √ | √ | √ | √ | √ | √ | √ | |
| mPd | √ | √ | √ | √ | √ | √ | √ | √ |
| mPt | | | | √ | √ | √ | √ | √ |
| Hip | | | | √ | √ | √ | √ | |
| Ent | | | | √ | √ | √ | √ | |
| Number of variables selected | 2 | 2 | 2 | 5 | 5 | 5 | 5 | 2 |

Table 40: The clustering results and their performance of each approach using ADNI$_5$_80% for training and ADNI$_5$_20% for testing with all observations.

| Approach | SelvarMix | vscc | clustvarsel | GMM | VarselLCM | sparcl | VCMM |
|---|---|---|---|---|---|---|---|
| training ARI | 0.36 | 0.35 | 0.38 | 0.36 | 0.42 | 0.45 | 0.38 |
| training MMM | 0.708 | 0.706 | 0.72 | 0.708 | 0.76 | 0.78 | 0.73 |
| test ARI | 0.3 | 0.29 | 0.31 | 0.3 | 0.4 | 0.45 | 0.32 |
| test MMM | 0.6 | 0.59 | 0.62 | 0.6 | 0.75 | 0.78 | 0.65 |
| BIC of the training model | 1016 | 1496 | 1444 | 1016 | 1036 | 558 | 1140 |
| Number of free parameters | 18 | 9 | 9 | 18 | 15 | 8 | 20 |
| Variables | | | | | | | |
| mPd | √ | | √ | √ | √ | √ | √ |
| mPt | √ | √ | | √ | √ | √ | √ |
| Number of variables selected | 2 | 1 | 1 | 2 | 2 | 2 | 2 |

Table 41: The clustering results and their performance of each approach using ADNI$_2$_80% for training and ADNI$_2$_20% for testing with all observations.

Table 42 and Table 43 show the ranks of the performance of each clustering approach for ADNI$_{10}$, ADNI$_5$ and ADNI$_2$ and corresponding train-test dataset. **Sparcl** always has lowest BIC and its performance is great. But as we said before, the BIC estimation for **sparcl** is rather simple and is a bit different from the other Gaussian mixture models. For the other variable selection approaches for GMM, the average rank of BIC in the three dataset of **SelvarMix** is a little lower. Although the models estimated **VarselLCM** contains all the variables, it also outperforms most approaches. Besides, **VCMM** always has highest BIC but it performs better than **GMM** regarding to **ARI** and **MMM** in ADNI$_5$, ADNI$_2$ and their corresponding train-test dataset. The models estimated by **Clustvarsel** have fewer variables and the performance of **Clustvarsel** is always better than **GMM**. **SelvarMix** also select few variables using ADNI$_{10}$, but its results are worse than those of **GMM**. Sometimes, **VSCC** will select all variables, which is the same result as **GMM**. However, it is also often able to select the relevant variables to make the clustering results better than **GMM**.

| | Approach | SelvarMix | vscc | clustvarsel | GMM | VarselLCM | sparcl | VCMM | VCMM_sel |
|---|---|---|---|---|---|---|---|---|---|
| ARI/MMM | ADNI$_{10}$ | 7 | 5 | 4 | 5 | 3 | 2 | 8 | 1 |
| | ADNI$_5$ | 7 | 4 | 6 | 8 | 1 | 2 | 5 | 3 |
| | ADNI$_2$ | 4 | 6 | 6 | 4 | 8 | 1 | 2 | 2 |
| BIC | ADNI$_{10}$ | 5 | 6 | 3 | 6 | 4 | 2 | 8 | 1 |
| | ADNI$_5$ | 4 | 3 | 6 | 7 | 5 | 1 | 8 | 2 |
| | ADNI$_2$ | 3 | 7 | 7 | 3 | 2 | 1 | 5 | 5 |
| Number of selected variables (low variable is better) | ADNI$_{10}$ | 3 | 4 | 2 | 4 | 4 | 4 | 4 | 1 |
| | ADNI$_5$ | 1 | 1 | 4 | 5 | 5 | 5 | 5 | 1 |
| | ADNI$_2$ | 3 | 1 | 1 | 3 | 3 | 3 | 3 | 3 |

Table 42: The rank of the performance of each clustering approach for ADNI$_{10}$, ADNI$_5$ and ADNI$_2$ with all observations.

| | Approach | SelvarMix | vscc | clustvarsel | GMM | VarselLCM | sparcl | VCMM | VCMM_sel |
|---|---|---|---|---|---|---|---|---|---|
| Training ARI/MMM | $ADNI_{10}$_80% | 7 | 4 | 6 | 4 | 2 | 1 | 8 | 3 |
| | $ADNI_5$_80% | 5 | 5 | 5 | 8 | 2 | 1 | 4 | 3 |
| | $ADNI_2$_80% | 6 | 8 | 5 | 6 | 2 | 1 | 3 | 3 |
| Test ARI/MMM | $ADNI_{10}$_80% | 7 | 5 | 4 | 5 | 2 | 1 | 8 | 3 |
| | $ADNI_5$_80% | 5 | 5 | 5 | 8 | 2 | 1 | 4 | 3 |
| | $ADNI_2$_80% | 6 | 8 | 5 | 6 | 2 | 1 | 3 | 3 |
| BIC of the training model | $ADNI_{10}$_80% | 3 | 6 | 4 | 6 | 5 | 2 | 8 | 1 |
| | $ADNI_5$_80% | 7 | 3 | 3 | 3 | 6 | 1 | 8 | 2 |
| | $ADNI_2$_80% | 2 | 7 | 7 | 2 | 4 | 1 | 5 | 5 |
| Number of selected variables (low variable is better) | $ADNI_{10}$_80% | 1 | 4 | 3 | 4 | 4 | 4 | 4 | 1 |
| | $ADNI_5$_80% | 1 | 1 | 1 | 5 | 5 | 5 | 5 | 1 |
| | $ADNI_2$_80% | 3 | 1 | 1 | 3 | 3 | 3 | 3 | 3 |

Table 43: The rank of the performance of each clustering approach using $ADNI_{10}$_80%, $ADNI_5$_80% and $ADNI_2$_80% for training and $ADNI_{10}$_20%, $ADNI_5$_20% and $ADNI_2$_20% for testing with all observations.

We also counted the number of non-Gaussian and Gaussian pair copula in the models estimated by **VCMM** using $ADNI_{10}$, $ADNI_5$, $ADNI_2$ and their corresponding train-test dataset, see Table 44. The table shows that the number of non-Gaussian pair copula is more than the number of Gaussian pair copula, and according to the previous tables, **VCMM** outperforms some GMM methods using $ADNI_5$, $ADNI_2$, $ADNI_5$_80% and $ADNI_2$_80%, which indicates the need for non-Gaussian models.

| | Number of Gaussian pair copulas | Number of non-Gaussian pair copulas |
|---|---|---|
| First cluster | | |
| $ADNI_{10}$ | 8 | 37 |
| $ADNI_5$ | 4 | 6 |
| $ADNI_2$ | 0 | 1 |
| $ADNI_{10}$_80% | 11 | 34 |
| $ADNI_5$_80% | 1 | 9 |
| $ADNI_2$_80% | 0 | 1 |
| Second cluster | | |
| $ADNI_{10}$ | 9 | 36 |
| $ADNI_5$ | 2 | 8 |
| $ADNI_2$ | 0 | 1 |
| $ADNI_{10}$_80% | 10 | 35 |
| $ADNI_5$_80% | 0 | 10 |
| $ADNI_2$_80% | 0 | 1 |
| Third cluster | | |
| $ADNI_{10}$ | 7 | 38 |
| $ADNI_5$ | 2 | 8 |
| $ADNI_2$ | 0 | 1 |
| $ADNI_{10}$_80% | 6 | 39 |
| $ADNI_5$_80% | 2 | 8 |
| $ADNI_2$_80% | 0 | 1 |

Table 44: Number of different types of copula pairs.

In the models estimated by **sparcl**, different variables have different weights, with

| | Dataset | GMM | VCMM |
|---|---|---|---|
| ARI | ADNI$_{10}$ | 0.29 | 0.2 |
| | ADNI$_5$ | 0.31 | 0.40 |
| | ADNI$_3$ | 0.34 | 0.42 |
| | ADNI$_2$ | 0.41 | 0.44 |
| MMM | ADNI$_{10}$ | 0.61 | 0.54 |
| | ADNI$_5$ | 0.62 | 0.75 |
| | ADNI$_3$ | 0.69 | 0.76 |
| | ADNI$_2$ | 0.75 | 0.77 |
| BIC | ADNI$_{10}$ | 11605 | 11960 |
| | ADNI$_5$ | 5144 | 5275 |
| | ADNI$_3$ | 2125 | 2182 |
| | ADNI$_2$ | 1272 | 1346 |
| Using 80% of dataset for training and 20% for testing | | | |
| training ARI | ADNI$_{10}$_80% | 0.33 | 0.26 |
| | ADNI$_5$_80% | 0.33 | 0.36 |
| | ADNI$_3$_80% | 0.38 | 0.41 |
| | ADNI$_2$_80% | 0.36 | 0.38 |
| training MMM | ADNI$_{10}$_80% | 0.65 | 0.54 |
| | ADNI$_5$_80% | 0.66 | 0.72 |
| | ADNI$_3$_80% | 0.74 | 0.76 |
| | ADNI$_2$_80% | 0.71 | 0.73 |
| test ARI | ADNI$_{10}$_80% | 0.27 | 0.20 |
| | ADNI$_5$_80% | 0.27 | 0.30 |
| | ADNI$_3$_80% | 0.32 | 0.30 |
| | ADNI$_2$_80% | 0.30 | 0.32 |
| test MMM | ADNI$_{10}$_80% | 0.55 | 0.43 |
| | ADNI$_5$_80% | 0.57 | 0.61 |
| | ADNI$_3$_80% | 0.63 | 0.62 |
| | ADNI$_2$_80% | 0.60 | 0.65 |
| BIC of the training model | ADNI$_{10}$_80% | 9423 | 9730 |
| | ADNI$_5$_80% | 4163 | 4323 |
| | ADNI$_3$_80% | 1725 | 1848 |
| | ADNI$_2$_80% | 1016 | 1140 |

Table 45: Comparison of **VCMM** to **GMM** when the same variables are used.

the three most heavily weighted variables being **S13**, **mPd** and **mPt**. Therefore, we also applied the **GMM** and **VCMM** to these three variables, denoted as ADNI$_3$, and compared the results. Table 45 shows the comparison of **VCMM** to **GMM** when the same variables are used.

# 6 Conclusion

In this thesis, we discussed variable selection approaches for clustering. To reduce the complexity of the model and increase the accuracy, we introduced several different approaches of variable selection for the mixture model clustering and applied them to the dataset obtained from Alzheimer's Disease Neuroimaging Initiative to compare results. First of all, we fitted with different types of variable selection approaches, i.e., penalization-based, information criterion-based, and hybrid-based approaches. To be more specific, we used four different variable selection approaches for GMM, including **Clustervarsel**, **VarSelLCM**, **VSCC** and **Selvarmix**, where **Clustervarsel** and **VarSelLCM** are information criterion approaches based on the BIC and MICL respectively, **VSCC** is hybrid approach based on within-group variance and **Selvarmix** is hybrid approach based on lasso penalization. Besides, we also applied **Sparcl** to the dataset, which is a penalization kmeans approach, a special case for GMM-based approach. After that, we started with a vine copula mixture model and use the R package **vineclust** Sahin [2022]. For them, we allowed Gaussian and non-Gaussian marginal distributions and different parametric copula family sets. For model comparison, two measurements were used, namely **ARI** and **MMM**.

## 6.1 Findings

In Section 4, we performed the clustering for the Gaussian mixture model without variable selection and the 5 different variable selection approaches for clustering. We found that in most cases the accuracy of the model improved after removing some variables for clustering, indicating that the presence of some irrelevant and redundant variables had a negative impact on the results of the model. Among the five approaches of variable selection, the approach **Sparcl** which is a penalization kmeans approach, always gave the highest **ARI** and **MMM**. Although it may select all variables, we can exclude those with weights close to zero. And according to the previous analysis, this does not affect the clustering results much or even improve them. **Clustervarsel** performs stable and concise. It always keep some relevant variables and remove some irrelevant ones. But it cannot distinguish well between **MCI** and **DEM** in our dataset. **VSCC** sometimes selects all variables for clustering and the results are the same as the traditional Gaussian model estimated by **mclust**. However, in many cases, **VSCC** is still able to select the most relevant variables for clustering. **Selvarmix** does not perform as well as other approaches. It removes some variables but its results sometimes tend to be lower than classic GMM approaches even though it needs to improve the clustering performance by selecting variables. For **VarselLCM**, it outperforms the other three Gaussian model-based approaches for a fixed number of clusters. But if we want to choose the best model among different numbers of clusters and variables, it tends to obtain a larger number of clusters, making the results worse. In addition, according to our analysis by using **GMM** we found that the variable **mPd**, **mpt** and **Ent** has the most significant effect on clustering in our dataset, while **S13**, **mPd** , **mPt** are more important for clustering if we analysed the results of penalization approach **sparcl**. We also applied the **GMM** to these most relevant variables estimated by **sparcl**, the **ARI** and **MMM** are satisfactory.

Most clustering approaches based on Gaussian mixture models do not identify **MCI** and **DEM** well and they are always concentrated in the same component. According to the boxplots of all variables, there are some non-Gaussian dependencies between variables in the normalized contour plot. Therefore, we applied the vine copula mixture model approach to the dataset in Section 5. Comparing **ARI** and **MMM** of these models, it showed that the results of the vine copula mixture models using all variables are worse. However, it performed much better after reducing some variables. Unlike the classic Gaussian mixture model-based approach, **VCMM** after variable selection can distinguish patients more clearly between the **MCI** and **DEM**. For variable selection, we used the stepwise approach based on the BIC criterion and got the model containing two most relevant variables **mPd** and **mPt**. It has higher **ARI** and **MMM** than the models estimated by GMM. Besides, we also applied the **VCMM** to the most three relevant variables estimated by **sparcl**, the results are better than most of Gaussian mixture model approaches. Furthermore, we also found that all approaches are less effective in identifying disease status in males than in females and married than non-married.

## 6.2 Future work

- The approach based on penalization, **sparcl**, always chooses all variables since none of the variables has a weight equal to zero. But if we remove the variables with very little weights, the result remains the same or even improves. An unsatisfied value of $s$ may lead to this situation, i.e., the weight of some variables is close to but not equal to zero, and thus the variable cannot be excluded. When we applied our proposed method for selecting the value of $s$ to our dataset, we found that our method calculates a slightly larger value for $s$ using our dataset, resulting in a situation where almost no variables weight zero, and therefore all variables can be retained. We may therefore need to consider a more appropriate method to choose a better $s$ to adjust the weights of the variables and the number of variables selected.

- Comparing the clustering approaches for Gaussian mixture model and vine copula mixture models, we found that **VCMM** does not perform as well as **GMM** when there are too many variables. But as the number of variables are reduced, the performance of **VCMM** improves significantly. Therefore variable selection is important for vine copula mixture models. In our thesis, we used a stepwise search based on BIC to select the variable and estimate the model that minimizes BIC. However, it can be computationally expensive if many variables are observed. Furthermore, the lowest BIC may not imply a good fit of the model to the observations. As we plotted in Section 5.3, when **WhB** was removed to reduce BIC in the second step, the **ARI** and **MMM** did not improve. Even though we ended up with a model with the greatest **ARI** and **MMM** of the process, this model is not the best model with highest **ARI** and **MMM** we observed in Figure 44(a) and Figure 44(b). Therefore, we may be able to find a more reliable and time-saving approach to variable selection for **VCMM** in future studies.

# References

Özge Sahin and Claudia Czado. Vine copula mixture models and clustering for non-gaussian data. *Econometrics and Statistics*, 22:136–158, 2022.

Claudia Czado. Analyzing dependent data with vine copulas. *Lecture Notes in Statistics, Springer*, 2019.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

Chris Fraley and Adrian E Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American statistical Association*, 97(458):611–631, 2002.

Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

David JC MacKay, David JC Mac Kay, et al. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

Juergen Altfeld. *tryCatchLog: Advanced 'tryCatch()' and 'try()' Functions*, 2021. URL `https://CRAN.R-project.org/package=tryCatchLog`. R package version 1.3.1.

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2 (1):193–218, 1985.

Ka Yee Yeung and Walter L Ruzzo. Details of the adjusted rand index and clustering algorithms, supplement to the paper an empirical study on principal component analysis for clustering gene expression data. *Bioinformatics*, 17(9):763–774, 2001.

Silke Wagner and Dorothea Wagner. *Comparing clusterings: an overview*. Universität Karlsruhe, Fakultät für Informatik Karlsruhe, 2007.

Wei Pan and Xiaotong Shen. Penalized model-based clustering with application to variable selection. *Journal of machine learning research*, 8(5), 2007.

Sijian Wang and Ji Zhu. Variable selection for model-based high-dimensional clustering and its application to microarray data. *Biometrics*, 64(2):440–448, 2008.

Daniela M Witten and Robert Tibshirani. A framework for feature selection in clustering. *Journal of the American Statistical Association*, 105(490):713–726, 2010.

Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Morton Slater. Lagrange multipliers revisited. In *Traces and emergence of nonlinear programming*, pages 293–306. Springer, 2014.

Florent Langrognet, Remi Lebret, Christian Poli, Serge Iovleff, Benjamin Auder, and Serge Iovleff. *Rmixmod: Classification with Mixture Modelling*, 2020. URL `https://CRAN.R-project.org/package=Rmixmod`. R package version 2.1.4.

Michael Fop, Thomas Brendan Murphy, et al. Variable selection methods for model-based clustering. *Statistics Surveys*, 12:18–65, 2018.

Cathy Maugis, Gilles Celeux, and Marie-Laure Martin-Magniette. Variable selection for clustering with gaussian mixture models. *Biometrics*, 65(3):701–709, 2009.

Matthieu Marbac and Mohammed Sedki. Varsellcm: an r/c++ package for variable selection in model-based clustering of mixed-data with missing values. *Bioinformatics*, 35(7):1255–1257, 2019.

Jeffrey L Andrews and Paul D McNicholas. Variable selection for clustering and classification. *Journal of Classification*, 31(2):136–153, 2014.

Gilles Celeux, Cathy Maugis-Rabusseau, and Mohammed Sedki. Variable selection in model-based clustering and discriminant analysis with a regularization approach. *Advances in Data Analysis and Classification*, 13(1):259–278, 2019.

Hui Zhou, Wei Pan, and Xiaotong Shen. Penalized model-based clustering with unconstrained covariance matrices. *Electronic journal of statistics*, 3:1473, 2009.

ADNI group. Variables description, 2010. URL `https://adni.bitbucket.io/reference/adnimerge.html`.

Thomas Nagler, Ulf Schepsmeier, Jakob Stoeber, Eike Christian Brechmann, Benedikt Graeler, and Tobias Erhardt. *VineCopula: Statistical Inference of Vine Copulas*, 2021. URL `https://CRAN.R-project.org/package=VineCopula`. R package version 2.4.3.

Luca Scrucca, Michael Fop, T. Brendan Murphy, and Adrian E. Raftery. mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, 8(1):289–317, 2016. URL `https://doi.org/10.32614/RJ-2016-021`.

Daniela M. Witten and Robert Tibshirani. *sparcl: Perform Sparse Hierarchical Clustering and Sparse K-Means Clustering*, 2018. URL `https://CRAN.R-project.org/package=sparcl`. R package version 1.0.4.

Luca Scrucca and Adrian E. Raftery. clustvarsel: A package implementing variable selection for gaussian model-based clustering in R. *Journal of Statistical Software*, 84(1):1–28, 2018. doi: 10.18637/jss.v084.i01.

Matthieu Marbac and Mohammed Sedki. Variable selection for model-based clustering using the integrated completed-data likelihood. *Statistics and Computing*, 27(4):1049–1063, 2017.

Jeffrey L. Andrews and Paul D. McNicholas. *vscc: Variable selection for clustering and classification*, 2013. R package version 1.

Mohammed Sedki, Gilles Celeux, and Cathy Maugis-Rabusseau. *SelvarMix: Regularization for Variable Selection in Model-Based Clustering and Discriminant Analysis*, 2017. URL `https://CRAN.R-project.org/package=SelvarMix`. R package version 1.2.1.

Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

Stephen A Ramsey, Sandy L Klemm, Daniel E Zak, Kathleen A Kennedy, Vesteinn Thorsson, Bin Li, Mark Gilchrist, Elizabeth S Gold, Carrie D Johnson, Vladimir Litvak, et al. Uncovering a macrophage transcriptional program by integrating evidence from motif scanning and expression dynamics. *PLoS computational biology*, 4(3):e1000021, 2008.

David P Hofmeyr. Degrees of freedom and model selection for k-means clustering. *Computational Statistics & Data Analysis*, 149:106974, 2020.

Oezge Sahin. *vineclust: Model-Based Clustering with Vine Copulas*, 2022. URL `https://github.com/oezgesahin/vineclust`. R package version 0.1.0.