UNIVERSIDADE FEDERAL DE JUIZ DE FORA
INSITUTO DE CIÊNCIAS EXATAS
CURSO DE ESTATÍSTICA

Davi Oliveira Chaves

Mistura Finita de Modelos de Regressão Assimétricos com Resposta Censurada

Juiz de Fora
2022

Davi Oliveira Chaves

Mistura Finita de Modelos de Regressão Assimétricos com Resposta Censurada

Monografia apresentada ao Curso de Estatística da Universidade Federal de Juiz de Fora, como requisito parcial para obtenção do grau de Bacharel em Estatística

Orientadora: Camila Borelli Zeller

Juiz de Fora
2022

Ficha catalográfica elaborada através do Modelo Latex do CDC da UFJF com os dados fornecidos pelo(a) autor(a)

Oliveira Chaves, Davi .
Mistura Finita de Modelos de Regressão Assimétricos com Resposta Censurada / Davi Oliveira Chaves. - 2022.
30 f. : il.

Orientadora: Camila Borelli Zeller
Trabalho de Conclusão de Curso - Universidade Federal de Juiz de Fora, Instituto de Ciências Exatas. Curso de Estatística, 2022.

1. Algoritmo EM. 2. Mistura Finita de Distribuições. 3. Classe de Distribuições Mistura de Escala Skew-Normal. 4. Censura. I. Borelli Zeller, Camila, orient. II. Título.

Davi Oliveira Chaves

Mistura Finita de Modelos de Regressão Assimétricos com Resposta Censurada

Monografia apresentada ao Curso de Estatística da Universidade Federal de Juiz de Fora, como requisito parcial para obtenção do grau de Bacharel em Estatística

Aprovada em:

BANCA EXAMINADORA

Profª. Dra. Camila Borelli Zeller - Orientadora
Universidade Federal de Juiz de Fora

Professor Dr. Clécio da Silva Ferreira
Universidade Federal de Juiz de Fora

Professor Dr. Lupércio França Bessegato
Universidade Federal de Juiz de Fora

AGRADECIMENTOS

Primeiramente aos familiriares, fonte infinita de carinho e afeto em minha vida. Especialmente aos meus pais, Leninha e Helvinho, sempre muito solícitos, permitiram que meu tempo fosse dedicado exclusivamente aos estudos, e a minha irmã caçula, Sarah, e minha gata, Tovis, por serem as melhores companhias possíveis nas horas de descanço.

Aos professores e pesquisadores que colaboraram de alguma forma em minha caminhada acadêmica. Principalmente à Dra. Camila, por ser uma excelente professora, orientadora e conselheira, ao Dr. Glauco, meu primeiro orientador, o qual levo como exemplo para vida profissional e pessoal, e ao Dr. Lupércio, por sempre estar disponível quando precisei de ajuda.

Aos amigos que fiz ao longo do curso, por todos almoços no RU, conversas jogadas fora, momentos de estudo, jogos de baralho e de sinuca. Especialmente ao Matheusão, Chandon e ao meu inseparável amigo Ian, que esteve comigo em todos os momentos desde o primeiro dia de faculdade, são mais de 100 partidas de sinuca jogadas. Também aos meus eternos amigos de Rio Pomba, Matheus, Gabriel, Koala, Vinição, Rodrigo, Cacá, Marcos, Igão e Yuri pelo companheirismo e por todos importantes momentos de descontração.

Por fim, gostaria de agradecer à banca examinadora, Dr. Clécio e Dr. Lupércio, por aceitarem o convite de contribuir com este trabalho e ao CNPq, pelo financiamento do projeto de iniciação científica que culminou nessa monografia.

RESUMO

O processo de modelagem via regressão linear para analisar dados com resposta censurada pode se tornar uma tarefa árdua quando o conjunto de dados apresenta certas especificidades como assimetria, caudas pesadas e heterogeneidade não observada. Alguns modelos para examinar dados com essas características já foram propostos na literatura, porém, nenhum deles busca acomodar as três ao mesmo tempo para dados censurados. O modelo apresentado neste trabalho visa cumprir esse desafio através da utilização da mistura finita de regressões lineares no contexto da classe de distribuições mistura de escala skew-normal (MESN), alcançando um modelo com alta flexibilidade que será útil em várias áreas de pesquisa. Devido à complexidade da função de verossimilhança nesses casos e à conveniente representação hierárquica da classe de distribuições MESN, um Algoritmo EM foi desenvolvido para obtenção dos estimadores de máxima-verossimilhança dos parâmetros originais. Para mostrar a aplicabilidade do modelo tanto no contexto de ajuste quanto no de classificação, foram realizados estudos de simulação e uma base de dados real foi analisada.

Palavras-chave: Algoritmo EM; Mistura Finita de Distribuições; Classe de Distribuições Mistura de Escala Skew-Normal; Censura.

ABSTRACT

The process of modeling via linear regression to analyze data with censored response can become an arduous task when the data set presents certain particularities such as asymmetry, heavy tails and unobserved heterogeneity. Some models to examine data with these characteristics have already been proposed in the literature, however, none of them seeks to accommodate all three at the same time for censored data. The model presented in this work aims to meet this challenge by using the finite mixture of linear regressions in the context of the scale mixture of skew-normal (SMSN) class of distributions, achieving a model with high flexibility that will be useful in several areas of research. Due to the complexity of the likelihood function in these cases and the convenient hierarchical representation of the SMSN class of distributions, an EM type algorithm was developed to obtain the maximum likelihood estimators of the original parameters. To show the applicability of the model both in the context of fit data and in the classification context, simulation studies were performed and a real dataset was analyzed.

Keywords: EM type algorithm; Finite Mixture of Distributions; Scale Mixture of Skew-Normal Class of Distributions; Censored Data.

Lista de ilustrações

Figura 1 – Seleção do parâmetro ν 26
Figura 2 – Boxplots das estimativas dos parâmetros do modelo MFR-SN-CR 29
Figura 3 – Histograma da variável resposta dos dados gerados pelo modelo MFR-SN-CR e linha horizontal representando o percentil 35% 30
Figura 4 – Erro quadrático médio dos estimadores dos parâmetros do modelo MFR-SN-CR. Nível de censura 8%: linha preta, 20%: cinza escuro e 35%: cinza claro 31
Figura 5 – Histograma das estimativas de alguns parâmetros de locação do modelo MFR-SN-CR para N = 500 32
Figura 6 – Histograma das estimativas de alguns parâmetros de locação do modelo MFR-SN-CR para N = 3000 32
Figura 7 – Boxplots das estimativas dos parâmetros do modelo MFR-ST-CR 33
Figura 8 – Erro quadrático médio dos estimadores dos parâmetros do modelo MFR-ST-CR. Nível de censura 8%: linha preta, 20%: cinza escuro e 35%: cinza claro 34
Figura 9 – Histograma das estimativas de alguns parâmetros de locação do modelo MFR-ST-CR para N = 500 36
Figura 10 – Histograma das estimativas de alguns parâmetros de locação do modelo MFR-ST-CR para N = 3000 37
Figura 11 – Valores de AIC e BIC para 100 amostras com 15% de censura à esquerda. Linha verde (do meio): MFR-SN-CR(3), vermelha (no topo): MFR-ST-CR(1), laranja (entre o meio e o topo): MFR-ST-CR(2), preta: MFR-ST-CR(3) e azul: MFR-ST-CR(1) 38
Figura 12 – Histograma e curva de densidade estimada da variável horas trabalhadas fora de casa dividida por 1000 40
Figura 13 – Seleção do parâmetro ν para ajuste do modelo MFR-ST-CR com um componente nos dados de Mroz (1984) 41

Lista de tabelas

Tabela 1 – Valor-p do teste de Shapiro-Wilk para verificar a normalidade dos estimadores dos parâmetros do modelo MFR-SN-CR 30
Tabela 2 – Valor-p do teste de Shapiro-Wilk para verificar a normalidade dos estimadores dos parâmetros do modelo MFR-ST-CR 35
Tabela 3 – Porcentagem de vezes em que cada modelo foi selecionado segundo cada nível de censura e critério 36
Tabela 4 – Média da acurácia dos modelos. Dados simulados de uma MFR-ST-CR com dois componentes e $\nu = 4$ 39
Tabela 5 – Média da acurácia dos modelos. Dados simulados de uma MFR-SSL-CR com dois componentes e $\nu = 2$ 39
Tabela 6 – Critérios de seleção para os modelos MFR-SN-CR e MFR-ST-CR com G = 1,...,3 para os dados de Mroz (1984) 41
Tabela 7 – Parâmetros estimados pelo melhor modelo (MFR-ST-CR com dois componentes e $\nu = 4$), erros-padrão aproximados e valor-p para os dados de Mroz (1984) 42

Lista de Abreviaturas

MEN - Mistura de escala normal
MESN - Mistura de escala skew-normal
MESNT - Mistura de escala skew-normal truncada
HN - Half-normal
SN - Skew-normal
NT - Normal truncada
SNT - Skew-normal truncada
ST - Skew-t
TT - t truncada
STT - Skew-t truncada
SSL - Skew-slash
EQM - Erro quadrático médio
ACC - Acurácia
CC - Classificações corretas
EM - Expectativa-maximização
AIC - Critério de informação de Akaike
BIC - Critério de informação de Bayes
fdp - Função densidade de probabilidade
EMV - Estimador de máxima verossimilhança

Sumário

1	**Introdução**	**11**
1.1	Objetivos	12
1.2	Descrição dos capítulos	12
2	**Conceitos iniciais**	**13**
2.1	Algoritmo EM	13
2.2	Mistura finita de distribuições	14
2.3	Misturas de Escala Skew-Normal	15
2.4	Dados censurados	16
3	**O modelo MFR-MESN-CR**	**17**
3.1	Especificação do modelo	17
3.2	Estimação do parâmetro via Algoritmo EM	18
3.3	Valores iniciais e critério de convergência	20
3.4	Erro-padrão aproximado	21
4	**Skew-normal**	**22**
5	**Skew-t**	**24**
6	**Aspectos computacionas**	**27**
7	**Estudos de simulação**	**28**
7.1	Recuperação dos parâmetros	28
7.1.1	MFR-SN-CR	28
7.1.2	MFR-ST-CR	32
7.2	Critério de seleção	35
7.3	Classificação	37
8	**Análise de dados reais**	**40**
9	**Considerações finais**	**43**
	REFERÊNCIAS	**44**
A	**ALGORITMO MFR-SN-CR**	**46**
B	**ALGORITMO MFR-ST-CR**	**55**

1 Introdução

A modelagem via misturas finitas de distribuições é uma técnica que vem sendo muito utilizada nas mais diversas áreas do conhecimento para lidar com conjuntos de dados que apresentam multimodalidade e heterogeneidade não observada. Nos casos em que é possível observar multimodalidade através do histograma e da curva de densidade estimada da variável, a modelagem supondo que a amostra seja proveniente de apenas um grupo não é recomendada, sendo interessante estimar os parâmetros considerando dois ou mais grupos e escolher a situação que melhor se ajusta aos dados através de critérios de seleção, como o AIC e o BIC. O mesmo vale no contexto de regressão linear quando a variável resposta apresenta multimodalidade. É imprudente supor que a relação entre esta e as variáveis explicativas seja fielmente ajustada por um único modelo linear. Portanto, é recomendável a estimação dos coeficientes supondo a presença de dois ou mais grupos de observações.

Essa metodologia foi pela primeira vez no final do século XIX em Pearson (1894), quando o aclamado estatístico Karl Pearson propôs um modelo de mistura finita de distribuições normais com dois componentes. Porém, devido à alta complexidade da função de verossimilhança nesse contexto, essa abordagem só foi consolidada após o desenvolvimento do Algoritmo EM por Dempster, Laird e Rubin (1977) e do trabalho de McLachlan e Basford (1988), que utilizou este algoritmo para facilitar a obtenção dos estimadores de máxima-verossimilhança nos modelos de mistura finita e aplicou-os a várias situações práticas. Atualmente, há várias obras disponíveis que tratam com excelência do assunto, dentre as quais podemos citar McLachlan e Peel (2004), Frühwirth-Schnatter (2006), Mengersen, Robert e Titterington (2011), McNicholas (2016) e Bouguila e Fan (2020).

Esse método de modelagem também é bem aceito no contexto de dados censurados, ou seja, aqueles em que os valores da variável resposta não estão completamente disponíveis para algumas observações (COLOSIMO; GIOLO, 2006). Esses dados são encontrados em várias pesquisas devido à restrições nos equipamentos de coleta. Por exemplo, variáveis como pH, carga viral de HIV e umidade do ambiente sofrem com censura nesse sentido. Portanto, como é comum a necessidade de modelar dados censurados, vários trabalhos foram desenvolvidos para tratá-los, porém, sua grande maioria no contexto simétrico, como Karlsson e Laitila (2014), que ajustou um modelo de mistura de regressões lineares para dados censurados considerando os erros normalmente distribuídos e Zeller et al. (2019), que fez o mesmo, no entanto, abrangiu a distribuição dos erros aleatórios para a classe de mistura de escala normal (MEN). Mais recentemente, Galarza, Matos e Lachos (2022) propôs um modelo para dados censurados multivariados que considera os erros distribuídos de acordo com uma skew-normal. Dessa forma, este trabalho visa expandir esse modelo para a classe de distribuições mistura de escala skew-normal (MESN) no contexto univariado e de misturas finitas, abordagem que ainda não foi apresentada na literatura.

O grande sucesso da classe de distribuições MESN é justificado pela flexibilidade oferecida ao lidar conjuntamente com assimetria e caudas pesadas em um conjunto de dados. Portanto, considerando uma mistura finita de regressões lineares com erros distribuídos na classe MESN alcançamos uma maneira de acomodar heterogeneidade não observada, assimetria e caudas pesadas. Um modelo que engloba todas essas características foi apresentado por Zeller, Cabral e Lachos (2016) para tratar dados sem censura. Assim, o presente trabalho pode ser visto tanto como uma extensão do artigo de Galarza, Matos e Lachos (2022) para a classe MESN no contexto univariado quanto como uma extensão do artigo de Zeller, Cabral e Lachos (2016) para o contexto de dados censurados.

1.1 Objetivos

O objetivo deste trabalho é desenvolver um modelo flexível para tratar dados censurados no contexto univariado, o qual seja capaz de acomodar conjuntamente heterogeneidade não observada, assimetria e caudas pesadas. Além disso, temos o propósito de apresentar expressões fechadas para as Etapas E e M do Algoritmo EM que foi aplicado para estimação dos parâmetros do modelo, assim como para os erros-padrão dessas estimativas. Esses resultados serão apresentados no contexto das distribuições skew-normal e skew-t, casos particulares da classe de distribuições MESN. Por fim, esperamos ilustrar os resultados através de estudos de simulação e de uma análise de dados reais.

1.2 Descrição dos capítulos

O restante do trabalho está organizado como segue. No Capítulo 2, são explicados os principais conceitos deste trabalho: Algoritmo EM, Mistura Finita de Distribuições, classe de distribuições MESN e dados censurados. No Capítulo 3 é apresentado o modelo de mistura finita de regressão com erros aleatórios distribuídos na classe MESN para análise de dados censurados, o qual foi denominado como MFR-MESN-CR. Nos Capítulos 4 e 5 são apresentadas as expressões fechadas da Etapa E para os casos particulares skew-normal e skew-t, respectivamente. No Capítulo 6, são exibidos os aspectos computacionais referentes ao desenvolvimento do algoritmo. No Capítulo 7 são feitos estudos de simulação para verificar a funcionalidade do modelo proposto e no Capítulo 8 uma base de dados reais é analisada. Por fim, no Capítulo 9 são apresentadas algumas conclusões e perspectivas de estudos futuros.

2 Conceitos iniciais

2.1 Algoritmo EM

Neste trabalho, a obtenção do estimador de máxima verossimilhança do vetor paramétrico de interesse $\boldsymbol{\theta}$ foi baseada no algoritmo de Expectativa-Maximização (DEMPSTER; LAIRD; RUBIN, 1977). Este, popularmente conhecido como Algoritmo EM, é uma ferramenta amplamente aplicada para a computação iterativa de estimadores de máxima-verossimilhança, útil em uma variedade de situações que apresentam dados incompletos, nas quais outros algoritmos iterativos, como o de Newton-Raphson, podem vir a ser mais trabalhosos (MCLACHLAN; KRISHNAN, 2007).

A ideia fundamental é reformular o problema, constituído de dados incompletos, em termos de uma situação com dados completos, os quais correspondem aos dados observados acrescidos dos dados não observáveis ou perdidos. Dessa forma, em vez de trabalhar com a log-verossimilhança dos dados observados, utiliza-se a log-verossimilhança dos dados completos, facilitando os cálculos.

O algoritmo é composto por duas etapas, comumente conhecidas como Etapa E, relativa ao cálculo da esperança da log-verossimilhança dos dados completos condicionada ao vetor de dados observados e à estimativa do vetor paramétrico, e a Etapa M, relativa à maximização da expressão encontrada no passo anterior. A esperança condicional da função de log-verossimilhança dos dados completos, mais conhecida como função Q, é dada abaixo

$$Q(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}}) = E[l_c(\boldsymbol{\theta}|\boldsymbol{y_c})|\boldsymbol{y_{obs}}, \hat{\boldsymbol{\theta}}]. \tag{1}$$

Logo, seja $\hat{\boldsymbol{\theta}}^{(\boldsymbol{k})}$ o vetor paramétrico estimado na k-ésima iteração, a (k+1)-ésima iteração é definida da seguinte forma:

- Etapa E: para $\hat{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}}^{(\boldsymbol{k})}$, calcular a função Q;
- Etapa M: obter $\hat{\boldsymbol{\theta}}^{(\boldsymbol{k+1})}$ a partir da maximização da função Q.

Outros aspectos importantes sobre o Algoritmo EM são a escolha dos valores iniciais e do critério de convergência. A estimativa obtida ao final do algoritmo depende do valor inicial. Portanto, sugere-se que sejam testados diferentes métodos de obtenção dos valores iniciais e seja escolhido aquele que faça o algoritmo convergir para o máximo global. Em situações de análise de dados reais, também é viável a utilização de estimações dos parâmetros que foram encontradas em estudos anteriores como valores iniciais.

Quanto ao critério de convergência, há várias opções, comparando-se a estimativa obtida na iteração atual com a obtida nas anteriores. Essa comparação pode ser feita utilizando-se tanto os vetores paramétricos quanto as funções de log-verossimilhança dos dados observados. Nesse trabalho, usou-se o seguinte critério

$$\left| \frac{l(\hat{\boldsymbol{\theta}}^{(\boldsymbol{k+1})}|\boldsymbol{y_{obs}})}{l(\hat{\boldsymbol{\theta}}^{(\boldsymbol{k})}|\boldsymbol{y_{obs}})} - 1 \right| < \epsilon, \tag{2}$$

em que ϵ é uma constante predeterminada com valor próximo de 0.

Em geral, são encontradas soluções fechadas na Etapa M, porém, quando isso não é possível ou consiste em uma tarefa custosa, podemos contar com algumas extensões do Algoritmo EM, como o Algoritmo ECM (MENG; RUBIN, 1993) e o ECME (LIU; RUBIN, 1994). Este último vem sendo constantemente utilizado na literatura para estimar o valor do parâmetro $\boldsymbol{\nu}$ em casos onde se trabalham com as distribuições da classe

Mistura de Escala Skew-Normal (MESN), veja por exemplo Zeller, Cabral e Lachos (2016) e Basso et al. (2010). Em ambas extensões, o vetor de parâmetros $\boldsymbol{\theta}$ é dividido em dois subvetores $\boldsymbol{\theta_1}$ e $\boldsymbol{\theta_2}$ e a Etapa M é dividida em duas, como mostra o esquema a seguir:

- ECM $\begin{cases} \text{Etapa CM-1} : \widehat{\boldsymbol{\theta_1}}^{(k+1)} = \underset{\boldsymbol{\theta_1}}{\arg\max}\ Q(\boldsymbol{\theta}|\widehat{\boldsymbol{\theta_2}}^{(k)}), \\ \text{Etapa CM-2} : \widehat{\boldsymbol{\theta_2}}^{(k+1)} = \underset{\boldsymbol{\theta_2}}{\arg\max}\ Q(\boldsymbol{\theta}|\widehat{\boldsymbol{\theta_1}}^{(k+1)}). \end{cases}$
- ECME $\begin{cases} \text{Etapa CM} : \widehat{\boldsymbol{\theta_1}}^{(k+1)} = \underset{\boldsymbol{\theta_1}}{\arg\max}\ Q(\boldsymbol{\theta}|\widehat{\boldsymbol{\theta_2}}^{(k)}), \\ \text{Etapa CME} : \widehat{\boldsymbol{\theta_2}}^{(k+1)} = \underset{\boldsymbol{\theta_2}}{\arg\max}\ l(\boldsymbol{\theta}|\boldsymbol{y_{obs}}, \widehat{\boldsymbol{\theta_1}}^{(k+1)}). \end{cases}$

2.2 Mistura finita de distribuições

A mistura finita de distribuições é um método de modelagem extremamente flexível que vêm recebendo crescente atenção ao passar dos anos devido a sua aplicação em diversas áreas da estatística como agrupamento, análise discriminante, análise de sobrevivência, inferência sobre dados heterogêneos, entre outros (MCLACHLAN; PEEL, 2004). Comprovação de sua importância e abrangência é a grande quantidade de obras sobre o assunto, dentre as quais podemos citar Everitt (2013), McNicholas (2016), Dávila, Cabral e Zeller (2018) e Bouguila e Fan (2020).

Essa abordagem foi utilizada pela primeira vez há mais de um século, quando Pearson (1894) implementou um modelo de mistura finita de distribuições normais, contendo dois componentes. Apesar disso, o amplo desenvolvimente deste método de modelagem só foi alcançado após a públicação de Dempster, Laird e Rubin (1977) sobre o Algoritmo EM, que facilitou o processo de estimação de máxima verossimilhança, o qual pode vir a ser bem complicado no contexto de mistura finita. Após isso, McLachlan e Basford (1988) aplicaram o modelo de mistura a vários problemas de interesse real como, por exemplo, a classificação de pessoas em portadoras e não-portadoras de uma certa doença através de outras variáveis mensuradas em cada indivíduo.

Neste trabalho, o modelo de mistura é utilizado devido a sua grande serventia no cenário de modelos de regressão aplicados em dados com respostas censuradas e presença de multimodalidade e heterogeneidade não observável , tanto no contexto de estimação dos parâmetros da distribuição quanto no contexto de classificação das observações.

Definição 2.1. Seja $Y_i, i = 1, ..., n$ uma variável aleatória distribuída de acordo com uma mistura finita de G componentes, sua função densidade de probabilidade (fdp) é dada por

$$f(y_i|\boldsymbol{\theta}) = \sum_{j=1}^{G} p_j g(y_i|\boldsymbol{\theta_j}), i = 1, ..., n, \tag{3}$$

onde $g(.|\boldsymbol{\theta_j})$ é a fdp de uma distribuição qualquer com vetor de parâmetros $\boldsymbol{\theta_j}, j = 1, ..., G$, o peso p_j representa a probabilidade da i-ésima observação pertencer ao componente j, tal que $\sum_{j=1}^{G} p_j = 1$ e $\boldsymbol{\theta} = (\boldsymbol{\theta_1}, ..., \boldsymbol{\theta_G}, p_1, ..., p_{G-1})$.

Com o intuito de simplificar os cálculos e as interpretações futuras, foi considerada nesse trabalho uma variável auxiliar $\boldsymbol{Z_i} = (Z_{i1}, ..., Z_{iG})$ que representa o componente relativo à i-ésima observação, onde

$$Z_{ij} = \begin{cases} 1, \text{se a i-ésima observação pertence ao j-ésimo componente;} \\ 0, \text{caso contrário.} \end{cases} \tag{4}$$

Nesse sentido, o vetor aleatório $\boldsymbol{Z_i}, i = 1, ..., n$ tem distribuição multinomial considerando-se G eventos mutuamente exclusivos com probabilidades $p_1, ..., p_G$, isto é,

$$P(\boldsymbol{Z_i} = \boldsymbol{z_i}) = p_1^{z_{i1}} p_2^{z_{i2}} ... p_G^{z_{iG}}. \tag{5}$$

2.3 Misturas de Escala Skew-Normal

A classe de distribuições misturas de escala skew-normal (MESN) é uma excelente alternativa para modelar dados que apresentam assimetra e caudas pesadas simultaneamente.

Definição 2.2. Uma variável aleatória Y tem distribuição MESN com parâmeto de locação $\mu \in \mathbb{R}$, de escala $\sigma^2 \in \mathbb{R}$ e de assimetria $\lambda \in \mathbb{R}$ se sua fdp é dada por

$$f(y) = 2 \int_0^\infty \phi(y|\mu, \kappa(u)\sigma^2) \Phi(\kappa^{1/2}(u) A) dH(u), \tag{6}$$

tal que $A = \lambda \frac{y-\mu}{\sigma}$, U é uma variável aleatória positiva com fda $H(.|\boldsymbol{\nu})$ e fdp $h(.|\boldsymbol{\nu})$, $\boldsymbol{\nu}$ é um parâmetro escalar ou vetorial que caracteriza a distribuição do fator de escala U, $\kappa(u)$ é uma função peso bem definida e é igual a u^{-1} no contexto das distribuições MESN, $\phi(.|\mu, \sigma^2)$ representa a fdp de uma normal com média μ e variância σ^2 e, por fim, $\Phi(.)$ é a função de distribuição acumulada (fda) de uma normal padrão.

No contexto de modelos de mistura com estimação paramétrica via Algoritmo EM, a representação estocástica da distribuição é muito utilizada.

Definição 2.3. Uma variável aleatória $Y \sim MESN(\mu, \sigma^2, \lambda, \boldsymbol{\nu})$ tem representação estocástica dada por

$$Y = \mu + \Delta T + \kappa^{1/2}(u) \Gamma^{1/2} T_0, \tag{7}$$

tal que $T = \kappa^{1/2}(u)|T_0|$, $T_0 \sim N(0, 1)$, $\delta = \frac{\lambda}{\sqrt{1+\lambda^2}}$, $\Delta = \sigma\delta$ e $\Gamma = \sigma^2(1 - \delta^2)$.

Como exemplos, podemos citar os seguintes casos particulares da distribuição MESN:

- Skew-normal, quando U é uma variável aleatória degenerada em 1;
- skew-t, quando $U \sim Gamma(\frac{\nu}{2}, \frac{\nu}{2})$;
- skew-slash, quando $U \sim Beta(\nu, 1)$;
- skew-normal contaminada, quando U é uma variável aleatória discreta que assume γ com probabilidade ν e 1 com probabilidade $1 - \nu$.

Um dos objetivos desse trabalho é tratar dados censurados, que serão apresentados na próxima Seção, logo, a definição da distribuição MESN truncada se faz necessária.

Definição 2.4. Seja $Y \sim MESN(\mu, \sigma^2, \lambda, \boldsymbol{\nu})$, com $P(a < Y < b) > 0$, para valores fixos de $a < b$. Uma variável aleatória X com distribuição MESNT, denotada por $X \sim MESNT(\mu, \sigma^2, \lambda, \boldsymbol{\nu}; [a, b])$, tem a mesma distribuição que $Y|Y \in [a, b]$.

Consequentemente, a fdp de X pode ser escrita como

$$f_T(x) = \frac{f(x|\mu, \sigma^2, \lambda, \boldsymbol{\nu})}{\int_a^b f(x|\mu, \sigma^2, \lambda, \boldsymbol{\nu})\partial x} \mathbb{I}_{[a,b]}(x), \tag{8}$$

onde $f(.|\mu, \sigma^2, \lambda, \boldsymbol{\nu})$ é a fdp de uma distribuição da classe MESN, introduzida na Definição 2.2 e $\mathbb{I}_{[a,b]}(x)$ é uma função indicadora, que assume o valor 1 caso $x \in [a, b]$ e 0 caso contrário.

Vale ressaltar que, mesmo apresentando o modelo em função da classe de distribuições MESN, as aplicações só foram realizadas no contexto da skew-normal e skew-t, pois os dois primeiros momentos dessas distribuições estão presentes no pacote do R *MomTrunc* (GALARZA et al., 2021), o qual, até o momento, é o único método disponível para computar os momentos de uma distribuição da classe MESN truncada.

2.4 Dados censurados

Neste trabalho, trataremos a censura que ocorre quando dados sobre a variável resposta não estão completamente disponíveis para algumas unidades da amostra por um motivo específico, como a limitação dos equipamentos de medição ou do projeto experimental, no entanto, para estas unidades, os dados sobre as variáveis regressoras são totalmente conhecidos (GARAY, 2014). Essas observações parciais estão presentes em várias áreas de pesquisa, como em acompanhamentos clínicos de pacientes, em processos industriais e na análise de características socioeconômicas da população. Seguindo a ideia apresentada em Louzada-Neto, Mazucheli e Achcar (2001), podemos citar como exemplo um processo industrial em que somente uma proporção dos componentes em estudo falham até o final. Logo, não é possível saber quanto tempo as peças que não falharam ainda durariam e, portanto, esses dados são censurados à direita. Além desse tipo de censura citado no exemplo, também existe censura à esquerda e intervalar.

Seguindo esse raciocínio, seja $\boldsymbol{Y} = (Y_1, ..., Y_n)$ o vetor resposta censurado. A variável resposta Y_i pode ser representada pelas variáveis conjuntas (V_i, ρ_i), onde $V_i, i = 1, ..., n$ é a variável que foi de fato observada, definida como

$$V_i = \begin{cases} (c_{i1}, c_{i2}), \text{ se } \rho_i = 1, \\ y_i, \text{ se } \rho_i = 0, \end{cases} \tag{9}$$

para algum intervalo conhecido $(c_{i1}, c_{i2}), i = 1, ..., n$. A variável $\rho_i, i = 1, ..., n$, é o indicador de censura, assumindo o valor 1 no caso em que a resposta seja censurada e 0 caso contrário.

Assim, caso a variável aleatória $Y_i, i = 1, ..., n$ seja censurada, sua contribuição na log-verossimilhança é dada pela $P(c_{i1} < Y_i < c_{i2})$. Caso contrário, sua contribuição é dada pela própria fdp de Y_i.

Definição 2.5. Seja $Y_i, i = 1, ..., n$ uma variável aleatória de um vetor de dados censurados. A log-verossimilhança dos dados é dada por

$$l(\boldsymbol{\theta}|\boldsymbol{V}) = \sum_{i=1}^{n} log\left[[F_{0i}]^{\rho_i} [f_0(y_i|\boldsymbol{\theta})]^{1-\rho_i} \right], \tag{10}$$

onde $f_0(y_i|\boldsymbol{\theta})$ é a fdp da distribuição original de Y_i, $F_{0i} = \int_{c_{i1}}^{c_{i2}} f_0(y_i|\boldsymbol{\theta})\partial y_i$ e $\boldsymbol{V} = (V_1, ..., V_n)$.

O vetor resposta foi enunciado acima considerando-se uma censura intervalar, porém, para representar casos de censura à direita basta substituir c_{i2} por ∞ e, em casos de censura à esquerda, basta substituir c_{i1} por $-\infty$. Mais ainda, para representar dados faltantes (missing data), basta substituir c_{i1} e c_{i2} por $-\infty$ e ∞, respectivamente, no entanto, esse caso particular foge do escopo deste trabalho.

3 O modelo MFR-MESN-CR

3.1 Especificação do modelo

Seja um modelo de mistura finita de regressões lineares com variável resposta distribuída no contexto da classe de distribuição MESN, apresentado em Zeller, Cabral e Lachos (2016). A observação $Y_i, i = 1, ..., n$ é descrita como

$$Y_i|Z_{ij} = 1 \sim MESN(\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j, \boldsymbol{\nu_j}) \quad j = 1, ..., G, \tag{11}$$

onde $\boldsymbol{x_i^T} = (x_{i1}, ..., x_{ip})$ é o vetor p-dimensional de variáveis explicativas, $\boldsymbol{\beta_j^T} = (\beta_{j1}, ..., \beta_{jp})$ é o vetor p-dimensional de coeficientes de regressão, G é o número de componentes da mistura finita e a variável Z_{ij} descrita na Seção 2.2.

De acordo com a distribuição da variável resposta, deve-se tomar certo cuidado no momento de estimar os valores da variável resposta, pois, de Basso et al. (2010), temos a seguinte proposição.

Proposição 3.1. *Seja $Y \sim MESN(\mu, \sigma^2, \lambda, \boldsymbol{\nu})$ e o fator de escala $U \sim H$,se $\kappa_1 < \infty$, então*

$$E[Y] = \mu + \sqrt{\frac{2}{\pi}}\kappa_1\Delta, \tag{12}$$

onde $\kappa_1 = \frac{2g_0(y)}{g(y)}E[\kappa^{-1}(U)\Phi(\kappa^{-1/2}(U)A)]$, $g_0(.)$ a fdp de $Y_0 \sim MEN(\mu, \sigma^2, \boldsymbol{\nu})$ e Δ apresentado na Definição 2.3.

Logo, o modelo não é centrado e, caso sejam feitas estimações ou predições para a variável resposta, o valor esperado de $Y_i, i = 1, ..., n$ não pode ser calculado somente através do preditor linear $\boldsymbol{x_i^T\beta_j}, j = 1, ..., G$. Deve-se somar o valor $\sqrt{\frac{2}{\pi}}\kappa_1\Delta_j, j = 1, ..., G$ ao preditor linear.

Além disso, por facilidade computacional, assumiu-se que $\boldsymbol{\nu_1} = ... = \boldsymbol{\nu_G} = \boldsymbol{\nu}$, ou seja, o parâmetro referente à variável fator de escala não varia entre as componentes. Essa estratégia foi utilizada em trabalhos predecessores e, além de simplificar enormemente o problema de otimização, funcionou muito bem nos estudos empíricos, fato que também foi observado durante a construção desse trabalho. Para mais detalhes, veja Lin, Ho e Lee (2014) e Zeller, Cabral e Lachos (2016).

A proposta desta tese é estender o modelo apresentado acima para o contexto de respostas censuradas, trabalho que ainda não se faz presente na literatura. Portanto, o vetor resposta é definido como mostrado na Seção 2.4 e a fdp da variável resposta é dada por

$$f(y_i|\boldsymbol{\theta}) = \sum_{j=1}^{G} p_j g(y_i|\boldsymbol{\theta_j}), \tag{13}$$

onde $\theta = (p_1, ..., p_{G-1}, \boldsymbol{\theta_1}, ..., \boldsymbol{\theta_G})$, $\boldsymbol{\theta_j} = (\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j, \boldsymbol{\nu_j})$, $g(y_i|\boldsymbol{\theta_j}) = [F_{0ij}]^{\rho_i}[f_0(y_i|\boldsymbol{\theta_j})]^{1-\rho_i}$, $f_0(.|\boldsymbol{\theta_j})$ é a fdp de uma $MESN(\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j, \boldsymbol{\nu_j})$, $F_{0ij} = \int_{c_{i1}}^{c_{i2}} f_0(y_i|\boldsymbol{\theta_j})dy_i$ e ρ_i é a variável dicotômica indicadora de censura, apresentada na Seção 2.4

No contexto da inferência clássica, o parâmetro desconhecido $\boldsymbol{\theta}$ seria estimado pelo estimador de máxima verossimilhança (EMV)

$$\hat{\boldsymbol{\theta}} = argmax_{\boldsymbol{\theta}} \sum_{i=1}^{n} log[f(y_i|\boldsymbol{\theta})]. \tag{14}$$

Entretanto, no contexto da classe de distribuição MESN, a maximização da expressão acima se torna muito complexa, então, propôs-se o uso do Algoritmo EM para encontrar o EMV do parâmetro.

3.2 Estimação do parâmetro via Algoritmo EM

A partir da representação estocástica de uma variável aleatória com distribuição na classe MESN apresentada na Definição 2.3 e lembrando que $\kappa(u) = u^{-1}$, a variável resposta $Y_i, i = 1, ..., n$ pode ser escrita hierarquicamente como

1. $Y_i|T_i = t_i, U_i = u_i, Z_{ij} = 1 \sim N(\boldsymbol{x_i^T\beta_j} + \Delta_j t_i, u_i^{-1}\Gamma_j)$,
2. $T_i|U_i = u_i, Z_{ij} = 1 \sim HN(0, u_i^{-1})$,
3. $U_i|Z_{ij} = 1 \sim H(u_i|\boldsymbol{\nu})$,
4. $\boldsymbol{Z_i} \sim \text{Multinomial}(1; p_1, ..., p_G)$,

onde $HN(a, b)$ denota a distribuição normal truncada com parâmetro de locação a e parâmetro de escala b no intervalo $(0, \infty)$.

Dada a representação hierárquica, suponha que temos m observações censuradas, as quais podem ser consideradas como realizações não observadas da variável $Y_i, i = 1, ..., n$. Portanto, para o desenvolvimento do Algoritmo EM, considera-se que as variáveis latentes $\boldsymbol{y_L} = (y_1, ..., y_m)^T$, $\boldsymbol{u} = (u_1, ..., u_n)^T$, $\boldsymbol{t} = (t_1, ..., t_n)^T$ e $\boldsymbol{z} = (\boldsymbol{z_1^T}, ..., \boldsymbol{z_n^T})^T$ tenham sido, de fato, observadas. Dessa forma, a função de log-verossimilhança completa associada ao vetor $\boldsymbol{y_c} = (\boldsymbol{y_{obs}^T}, \boldsymbol{y_L^T}, \boldsymbol{u^T}, \boldsymbol{t^T}, \boldsymbol{z^T})^T$ é

$$l(\boldsymbol{\theta}|\boldsymbol{y_c}) = c + \sum_{i=1}^{n}\sum_{j=1}^{G} z_{ij} log p_j - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{G} z_{ij} log\Gamma_j - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{G} z_{ij}\frac{(y_i - \boldsymbol{x_i^T\beta_j} - \Delta_j t_i)^2}{u_i^{-1}\Gamma_j} + \sum_{i=1}^{n}\sum_{j=1}^{G} z_{ij} log(u_i|\boldsymbol{\nu}), \tag{15}$$

em que c é uma constante independente do vetor de parâmetros $\boldsymbol{\theta}$.

Dada a função de log-verossimilhança completa, as etapas do Algoritmo EM utilizado no modelo proposto são apresentadas a seguir.

Etapa E

Seja $\hat{\boldsymbol{\theta}}^{(\boldsymbol{k})}$ a estimativa de $\boldsymbol{\theta}$ na k-ésima iteração do algoritmo. Após simples álgebra, a esperança condicional da função de log-verossimilhança completa $Q(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}}^{(\boldsymbol{k})}) = E[l(\boldsymbol{\theta}|\boldsymbol{y_c})|\boldsymbol{y_{obs}}, \hat{\boldsymbol{\theta}}^{(\boldsymbol{k})}]$, conhecida como função Q, é dada por

$$\begin{aligned} Q(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}}^{(\boldsymbol{k})}) = c &+ \sum_{i=1}^{n}\sum_{j=1}^{G} \widehat{z_{ij}}^{(k)} log\widehat{p_j}^{(k)} - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{G} \widehat{z_{ij}}^{(k)} log\widehat{\Gamma_j}^{(k)} - \\ &\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{G}\frac{1}{\widehat{\Gamma_j}}[\widehat{\epsilon_{02ij}}^{(k)} - 2\widehat{\epsilon_{01ij}}^{(k)}(\boldsymbol{x_i^T}\widehat{\boldsymbol{\beta}}_{\boldsymbol{j}}^{(\boldsymbol{k})}) + \widehat{\epsilon_{00ij}}^{(k)}(\boldsymbol{x_i^T}\widehat{\boldsymbol{\beta}}_{\boldsymbol{j}}^{(\boldsymbol{k})})^2 + \widehat{\epsilon_{20ij}}^{(k)}\widehat{\Delta_j^2}^{(k)} - 2\widehat{\Delta}^{(k)}\widehat{\epsilon_{11ij}}^{(k)} + \\ &2\widehat{\epsilon_{10ij}}^{(k)}\widehat{\Delta}^{(k)}(\boldsymbol{x_i^T}\widehat{\boldsymbol{\beta}}_{\boldsymbol{j}}^{(\boldsymbol{k})})] + \sum_{i=1}^{n}\sum_{j=1}^{G}\widehat{z_{ij}}^{(k)} E[log\big(h(U_i|\boldsymbol{\nu})\big)|\boldsymbol{y_{obs}}, \widehat{\boldsymbol{\theta}}_{\boldsymbol{j}}^{(\boldsymbol{k})}], \end{aligned} \tag{16}$$

onde $\widehat{z_{ij}}^{(k)} = E[Z_{ij}|y_{obsi}, \hat{\boldsymbol{\theta}}_{\boldsymbol{j}}^{(\boldsymbol{k})}]$ e $\widehat{\epsilon_{rsij}}^{(k)} = E[Z_{ij}U_iT_i^rY_i^s|y_{obsi}, \hat{\boldsymbol{\theta}}_{\boldsymbol{j}}^{(\boldsymbol{k})}], r, s = 0, 1, 2.$

Como considerou-se o $\boldsymbol{\nu}$ fixo, não é necessário calcular a esperança $E[log\big(h(U_i|\boldsymbol{\nu})\big)|\boldsymbol{y_{obs}}, \widehat{\boldsymbol{\theta}}_{\boldsymbol{j}}^{(\boldsymbol{k})}]$.

Usando propriedades conhecidas de esperança condicional, temos

$$\widehat{z_{ij}}^{(k)} = \frac{\widehat{p_j}^{(k)} g(y_i|\hat{\boldsymbol{\theta}}^{(\boldsymbol{k})})}{\sum_{j=1}^{G} \widehat{p_j}^{(k)} g(y_i|\hat{\boldsymbol{\theta}}^{(\boldsymbol{k})})} \quad e \quad \widehat{\epsilon_{rsij}}^{(k)} = \widehat{z_{ij}}^{(k)} E[U_iT_i^rY_i^s|y_{obsi}, \hat{\boldsymbol{\theta}}^{(\boldsymbol{k})}], r, s = 0, 1, 2. \tag{17}$$

Vale ressaltar que os valores de $\widehat{z_{ij}}$ são utilizados para classificar as observações em cada Grupo, fato que será melhor explicado na Seção 7.3.

Antes de prosseguir com as esperanças condicionais, é necessário enunciar uma proposição, cujos resultados estão presentes em Cabral, Lachos e Prates (2012).

Proposição 3.2. *Seja* $Y_i \sim MESN(\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j, \boldsymbol{\nu_j})$, $T_i = U_i^{-1/2}|T_0|$, $T_0 \sim N(0,1)$,U_i *o fator de escala de* Y_i, $Z_{ij} \sim Multinomial(1; p_1, ...p_G)$ *e* $X \sim HN(a, b^2)$. *Então*

1. $T_i|U_i = u_i, Z_{ij} = 1 \sim HN(0, u_i^{-1})$,
2. $T_i|Y_i = y_i, U_i = u_i, Z_{ij} = 1 \sim HN(\mu_{Tij}, u_i^{-1}M_{Tj}^2)$,
3. $E[X] = a + W_\phi(\frac{a}{b})b$,
4. $E[X^2] = a^2 + b^2 + W_\phi(\frac{a}{b})ab$,

onde $\mu_{Tij} = \frac{\Delta_j}{\Gamma_j+\Delta_j^2}(y_i - \boldsymbol{x_i^T\beta_j})$, $M_{Tj}^2 = \frac{\Gamma_j}{\Gamma_j+\Delta_j^2}$, $W_\phi(.) = \frac{\phi(.)}{\Phi(.)}$, $\phi(.)$ *é a fdp da distribuição normal padrão e* $\Phi(.)$ *é a fda da mesma.*

Em posse da proposição acima, seguem as esperanças condicionais para as observações sem censura e censuradas colocadas de uma forma geral para a classe MESN. Nos Capítulos 4 e 5, as expressões fechadas no contexto da skew-normal e da skew-t, respectivamente, podem ser encontradas.

Observações sem censura

Nesse caso, $Y_{obsi} = Y_i \sim MESN(\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda, \boldsymbol{\nu})$. Logo,

- $\widehat{\epsilon_{00ij}}^{(k)} = \widehat{z_{ij}}^{(k)} E[U_i|Y_i, \widehat{\boldsymbol{\theta}}_{\boldsymbol{j}}^{(\boldsymbol{k})}]$, necessitando-se do conhecimento do fator de escala U_i,
- $\widehat{\epsilon_{01ij}}^{(k)} = \widehat{\epsilon_{00ij}}^{(k)} y_i$
- $\widehat{\epsilon_{02ij}}^{(k)} = \widehat{\epsilon_{00ij}}^{(k)} y_i^2$
- $\widehat{\epsilon_{10ij}}^{(k)} = \widehat{z_{ij}}^{(k)} [\frac{\widehat{\epsilon_{00ij}}^{(k)}}{\widehat{z_{ij}}^{(k)}} \widehat{\mu_{Tij}}^{(k)} + \widehat{M_{Tj}}^{(k)} \widehat{\tau_{Tij}}^{(k)}]$, onde $\tau_{Tij} = E[U_i^{1/2} W_\phi(U_i^{1/2} A_{ij})|Y_i, \widehat{\boldsymbol{\theta}}_{\boldsymbol{j}}^{(\boldsymbol{k})}]$,
- $\widehat{\epsilon_{20ij}}^{(k)} = \widehat{z_{ij}} [\frac{\widehat{\epsilon_{00ij}}^{(k)}}{\widehat{z_{ij}}^{(k)}} (\widehat{\mu_{Tij}}^{(k)})^2 + (\widehat{M_{Tj}}^{(k)})^2 + \widehat{\mu_{Tij}}^{(k)} \widehat{M_{Tj}}^{(k)} \widehat{\tau_{Tij}}^{(k)}]$,
- $\widehat{\epsilon_{11ij}}^{(k)} = \widehat{\epsilon_{01ij}}^{(k)} [\frac{\widehat{\epsilon_{00ij}}^{(k)}}{\widehat{z_{ij}}^{(k)}} \widehat{\mu_{Tij}}^{(k)} + \widehat{M_{Tj}}^{(k)} \widehat{\tau_{Tij}}^{(k)}]$.

Dados censurados

Nesse caso, $Y_{obsi} = (c_{i1}, c_{i2})$, ou seja, $c_{i1} \leq Y_i \leq c_{i2}$. Portanto, precisa-se da informação sobre a distribuição exata de Y_i para o cálculo das esperanças condicionais, que apresentam o seguinte formato:

- $\widehat{\epsilon_{rsij}}^{(k)} = \widehat{z_{ij}}^{(k)} E[Y_i^s E[U_i E[T_i^r | U_i, Y_i, \widehat{\boldsymbol{\theta}_j}^{(k)}] | Y_i, \widehat{\boldsymbol{\theta}_j}^{(k)}] | c_{i1} \leq Y_i \leq c_{i2}, \widehat{\boldsymbol{\theta}_j}^{(k)}], r, s = 0, 1, 2.$

Etapa M

Nessa etapa, os valores dos parâmetros são atualizados maximizando a função Q sobre cada um deles. Então, para $j = 1, ..., G$, temos

- $\widehat{p_j}^{(k+1)} = \frac{\sum_{i=1}^{n} \widehat{z_{ij}}^{(k)}}{n}$,
- $\widehat{\boldsymbol{\beta_j}}^{(k+1)} = (\sum_{i=1}^{n} \widehat{\epsilon_{00ij}}^{(k)} \boldsymbol{x_i x_i^T})^{-1} \sum_{i=1}^{n} (\widehat{\epsilon_{01ij}}^{(k)} - \widehat{\epsilon_{10ij}}^{(k)} \widehat{\Delta_j}^{(k)}) \boldsymbol{x_i}$,
- $\widehat{\Delta_j}^{(k+1)} = \frac{\sum_{i=1}^{n} [\widehat{\epsilon_{11ij}}^{(k)} - \widehat{\epsilon_{10ij}}^{(k)} (\boldsymbol{x_i^T} \widehat{\boldsymbol{\beta}_j}^{(k+1)})]}{\widehat{\epsilon_{20ij}}^{(k)}}$,
- $\widehat{\Gamma_j}^{(k+1)} = (\sum_{i=1}^{n} \widehat{z_{ij}}^{(k)})^{-1} \sum_{i=1}^{n} [\widehat{\epsilon_{02ij}}^{(k)} - 2\widehat{\epsilon_{01ij}}^{(k)} (\boldsymbol{x_i^T} \widehat{\boldsymbol{\beta}_j}^{(k+1)}) + \widehat{\epsilon_{00ij}}^{(k)} (\boldsymbol{x_i^T} \widehat{\boldsymbol{\beta}_j}^{(k+1)})^2 + \widehat{\epsilon_{20ij}}^{(k)} \widehat{\Delta_j^2}^{(k+1)} - 2\widehat{\Delta}^{(k+1)} \widehat{\epsilon_{11ij}}^{(k)} + 2\widehat{\epsilon_{10ij}}^{(k)} \widehat{\Delta}^{(k+1)} (\boldsymbol{x_i^T} \widehat{\boldsymbol{\beta}_j}^{(k+1)})]$,
- $\widehat{\sigma_j^2}^{(k+1)} = \widehat{\Gamma_j}^{(k+1)} + \widehat{\Delta_j^2}^{(k+1)}$,
- $\widehat{\lambda_j}^{(k+1)} = \widehat{\Delta_j}^{(k+1)} + \widehat{\Gamma_j^{-1/2}}^{(k+1)}$.

Como o $\boldsymbol{\nu}$ é considerado fixo, ele não é estimado. Porém, a escolha do melhor valor para esse parâmetro pode ser vista em detalhes no Capítulo 5.

3.3 Valores iniciais e critério de convergência

A escolha adequada dos valores iniciais é o que garante a convergência do Algoritmo EM para o máximo global. Portanto, essa é uma etapa crucial do processo. Os valores iniciais do algoritmo proposto nesse estudo são encontrados da seguinte maneira:

- A amostra é particionada em G grupos através do algoritmo de agrupamento k-means (HARTIGAN; WONG, 1979) e um modelo de regressão linear é traçado em cada um deles;
- $p_j^{(0)}$ é dado pela proporção de observações presentes em cada grupo j, j = 1,...,G;
- $\boldsymbol{\beta_j}^{(0)}$ é dado pelos coeficientes da regressão linear traçada no grupo j, j = 1,...,G;
- $\sigma_j^{2(0)}$ é dada pela variância estimada a partir do modelo traçado no grupo j, j = 1,...,G;
- $\lambda_j^{(0)}$ é dada pelo cálculo da assimetria dos resíduos do modelo traçado no grupo j, j = 1,...,G.

O critério de convergência utilizado é o mesmo que foi apresentado na Equação 2, na Seção 2.1.

3.4 Erro-padrão aproximado

A matriz de covariância assintótica de $\hat{\boldsymbol{\theta}}$ pode ser aproximada pela inversa da matriz de informação empírica, definida como $\boldsymbol{I_0} = \sum_{i=1}^{n} s(Y_i|\boldsymbol{\theta})s(Y_i|\boldsymbol{\theta})^T - n^{-1}S(\boldsymbol{Y}|\boldsymbol{\theta})S(\boldsymbol{Y}|\boldsymbol{\theta})^T$, em que $S(\boldsymbol{Y}|\boldsymbol{\theta}) = \sum_{i=1}^{n} s(Y_i|\boldsymbol{\theta})$ e $s(Y_i|\boldsymbol{\theta}) = \frac{\partial Q_i(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}})}{\partial \boldsymbol{\theta}}$. Substituindo $\boldsymbol{\theta}$ por $\hat{\boldsymbol{\theta}}$ em $\boldsymbol{I_0}$, obtém-se a aproximação $\hat{\boldsymbol{I}}_{\boldsymbol{0}} = \sum_{i=1}^{n} \hat{\boldsymbol{s}}_{\boldsymbol{i}}\hat{\boldsymbol{s}}_{\boldsymbol{i}}^{\boldsymbol{T}}$, onde $\hat{\boldsymbol{s}}_{\boldsymbol{i}}$ é o vetor escore $\hat{\boldsymbol{s}}_{\boldsymbol{i}} = (\hat{s}_{i,p_1}, ..., \hat{s}_{i,p_{G-1}}, \hat{s}_{i,\boldsymbol{\beta}_1}, ..., \hat{s}_{i,\boldsymbol{\beta}_G}, \hat{s}_{i,\sigma_1^2}, ..., \hat{s}_{i,\sigma_G^2}, \hat{s}_{i,\lambda_1}, ..., \hat{s}_{i,\lambda_G})^T$. As expressões fechadas para os elementos de $\hat{\boldsymbol{s}}_{\boldsymbol{i}}$, são dadas por

- $\hat{s}_{i,p_j} = \frac{\hat{z}_{ij}^{(k)}}{p_j} - \frac{\hat{z}_{iG}^{(k)}}{p_G}$;
- $\hat{s}_{i,\boldsymbol{\beta}_j} = \frac{1}{\Gamma_j}[\widehat{\epsilon_{01ij}}^{(k)}\boldsymbol{x_i} - \widehat{\epsilon_{00ij}}^{(k)}(\boldsymbol{x_i^T\beta_j})\boldsymbol{x_i} - \widehat{\epsilon_{10ij}}^{(k)}\Delta_j\boldsymbol{x_i}]$;
- $\hat{s}_{i,\sigma_j^2} = -\frac{1}{2\sigma_j^2} + \frac{1+\lambda_j^2}{2\sigma_j^4}[\widehat{\epsilon_{02ij}}^{(k)} - 2\widehat{\epsilon_{01ij}}^{(k)}(\boldsymbol{x_i^T\beta_j}) + \widehat{\epsilon_{00ij}}^{(k)}(\boldsymbol{x_i^T\beta_j})^2 + \widehat{\epsilon_{20ij}}^{(k)}\Delta_j^2 - 2\Delta\widehat{\epsilon_{11ij}}^{(k)} + 2\widehat{\epsilon_{10ij}}^{(k)}\Delta(\boldsymbol{x_i^T\beta_j})] - \frac{\delta_j}{2\Gamma_j}\left[\delta_j\widehat{\epsilon_{20ij}}^{(k)} + \frac{1}{\sigma_j}(\widehat{\epsilon_{11ij}}^{(k)} - \widehat{\epsilon_{10ij}}^{(k)}(\boldsymbol{x_i^T\beta_j}))\right]$;
- $\hat{s}_{i,\lambda_j} = \frac{\lambda_j}{1+\lambda_j^2} - \frac{\lambda_j}{\sigma_j^2}[\widehat{\epsilon_{02ij}}^{(k)} - 2\widehat{\epsilon_{01ij}}^{(k)}(\boldsymbol{x_i^T\beta_j}) + \widehat{\epsilon_{00ij}}^{(k)}(\boldsymbol{x_i^T\beta_j})^2 + \widehat{\epsilon_{20ij}}^{(k)}\Delta_j^2 - 2\Delta\widehat{\epsilon_{11ij}}^{(k)} + 2\widehat{\epsilon_{10ij}}^{(k)}\Delta(\boldsymbol{x_i^T\beta_j})] - \frac{\lambda_j}{1+\lambda_j^2}\widehat{\epsilon_{20ij}}^{(k)} + \frac{1+2\lambda_j^2}{\sigma_j\sqrt{1+\lambda_j^2}}(\widehat{\epsilon_{11ij}}^{(k)} - \widehat{\epsilon_{10ij}}^{(k)}(\boldsymbol{x_i^T\beta_j}))$.

Os erros-padrão são obtidos pela raiz quadrada da diagonal principal da inversa da matriz $\hat{\boldsymbol{I}}_{\boldsymbol{0}}$.

4 Skew-normal

A distribuição skew-normal é um caso particular da classe de distribuições MESN quando o fator de escala U, apresentado na Definição 2.2, é uma variável aleatória degenerada em 1, ou seja, $P(U = 1) = 1$. Portanto, não há presença do parâmetro $\boldsymbol{\nu}$, que funcionava como um fator de acomodação para valores extremos. Esse modelo foi denominado como MFR-SN-CR.

Definição 4.1. Seja Y uma variável aleatória que segue uma distribuição $SN(\mu, \sigma^2, \lambda)$, a função densidade de probabilidade de Y é dada por

$$g(y) = 2\phi(y|\mu, \sigma^2)\Phi\left(\lambda\frac{(y-\mu)}{\sigma}\right), y \in \mathbb{R}. \tag{18}$$

As fórmulas fechadas apresentadas na Etapa M e na Seção de estimação dos erros-padrão permanecem as mesmas, porém, o valor esperado obtido na Etapa E $\epsilon_{00ij} = z_{ij}E[U_i|y_i, \boldsymbol{\theta_j}]$ pode ser substituído simplesmente por z_{ij}, pois agora o valor esperado da variável $U_i, i = 1, ..., n$ é igual a 1 em qualquer situação.

Portanto, em posse da distribuição do erro aleatório do modelo (skew-normal), podemos encontrar fórmulas fechadas para as esperanças condicionais $z_{ij} = E[Z_{ij}|y_{obsi}, \boldsymbol{\theta_j}]$ e $\epsilon_{rsij} = z_{ij}E[T_i^r Y_i^s|y_{obsi}, \boldsymbol{\theta_j}], r, s = 0, 1, 2$. Porém, antes disso, devemos enunciar uma importante Proposição que será necessária no cálculo dos resultados acima para o caso das observações censuradas. A demonstração no contexto multivariado encontra-se em Galarza, Matos e Lachos (2022), que estudou a distribuição skew-normal multivariada com respostas censuradas.

Proposição 4.1. *Seja uma variável aleatória* $Y \sim SNT(\mu, \sigma^2, \lambda; [a, b])$*, então*

$$E\left[h(Y)W_\phi\left(\lambda\frac{Y-\mu}{\sigma}\right)\right] = \frac{P_0}{\sqrt{\frac{\pi}{2}(1+\lambda^2)}R_0}E[r(W_0)], \tag{19}$$

onde $P_0 = \int_a^b \phi(y|\mu, \sigma^2)\partial y$, $R_0 = \int_a^b g(y|\mu, \sigma^2, \lambda)\partial y$, $\phi(.|\mu, \sigma^2)$ *é a fdp de uma distribuição normal,* $g(.|\mu, \sigma^2, \lambda)$ *é a fdp de uma distribuição skew-normal,* $W_0 \sim NT(\mu, \Gamma; [a, b])$*, SNT representa uma distribuição skew-normal truncada, NT uma distribuição normal truncada e* $r(.)$ *é uma função bem definida.*

Usando os resultados do Capítulo 3 e a Proposição 4.1, as fórmulas fechadas são apresentadas a seguir.

Observações sem censura:

- $z_{ij} = \frac{p_j g(y_i|\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j)}{\sum_{j=1}^G p_j g(y_i|\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j)}$;
- $\epsilon_{01ij} = z_{ij}y_i$;
- $\epsilon_{02ij} = z_{ij}y_i^2$;
- $\epsilon_{10ij} = z_{ij}[\mu_{Tij} + M_{Tj}W_\phi(A_{ij})]$;
- $\epsilon_{20ij} = z_{ij}[\mu_{Tij}^2 + M_{Tj}^2 + \mu_{Tij}M_{Tj}W_\phi(A_{ij})]$;
- $\epsilon_{11ij} = z_{ij}y_i[\mu_{Tij} + M_{Tj}W_\phi(A_{ij})]$.

Observações censuradas:

- $z_{ij} = \frac{p_j R_{0ij}}{\sum_{j=1}^G p_j R_{0ij}}$, em que $R_{0ij} = \int_{c_{i1}}^{c_{i2}} g(y_i|\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j)\partial y_i$

- $\epsilon_{01ij} = z_{ij}E[W_i]$, em que $W_i \sim SNT(\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j; [c_{i1}, c_{i2}])$;
- $\epsilon_{02ij} = z_{ij}E[W_i^2]$;
- $\epsilon_{10ij} = z_{ij}\left[\frac{M_{Tj}^2\Delta_j}{\Gamma_j}(\frac{\epsilon_{01ij}}{z_{ij}} - \mu_{Tij}) + M_{Tj}\gamma_{ij}\right]$, em que $\gamma_{ij} = \frac{P_{0ij}}{\sqrt{\frac{\pi}{2}(1+\lambda^2)R_{0ij}}}$ e $P_{0ij} = \int_{c_{i1}}^{c_{i2}} \phi(y_i|\boldsymbol{x_i^T\beta_j}, \Gamma_j)\partial y_i$;
- $\epsilon_{20ij} = z_{ij}\left[\left(\frac{M_{Tj}^2\Delta_j}{\Gamma_j}\right)^2(\frac{\epsilon_{02ij}}{z_{ij}} - 2\frac{\epsilon_{01ij}}{z_{ij}}\mu_{Tij} + \mu_{Tij}^2) + \frac{M_{Tj}^3\Delta_j}{\Gamma_j}(E[W_0] - \mu_{Tij})\gamma_{ij} + M_{Tj}^2\right]$, em que $W_0 \sim NT(\boldsymbol{x_i^T\beta_j}, \Gamma_j; [c_{i1}, c_{i2}])$;
- $\epsilon_{11ij} = z_{ij}\left[\frac{M_{Tj}^2\Delta_j}{\Gamma_j}(\frac{\epsilon_{02ij}}{z_{ij}} - \frac{\epsilon_{01ij}}{z_{ij}}\mu_{Tij}) + M_{Tj}E[W_0]\gamma_{ij}\right]$.

Lembrando que o primeiro e o segundo momento das variáveis W_i e W_0 são calculados por meio do pacote *MomTrunc* (GALARZA et al., 2021).

5 Skew-t

A distribuição skew-t é outro caso particular da classe de distribuições MESN que abordou-se nesse estudo. Ela ocorre quando o fator de escala U, apresentado na Definição 2.2, é uma variável aleatória com distribuição $Gamma(\frac{\nu}{2}, \frac{\nu}{2})$. Nesse caso, o fator de acomodação de valores extremos $\nu > 0$ é um escalar. Esse modelo foi denominado como MFR-ST-CR.

Definição 5.1. Seja Y uma variável aleatória que segue uma distribuição $ST(\mu, \sigma^2, \lambda, \nu)$, a função densidade de probabilidade de Y é dada por

$$g(y) = 2t(y|\mu, \sigma^2, \nu)T\left(\sqrt{\frac{\nu+1}{\nu+d}}A|\nu+1\right), y \in \mathbb{R}, \tag{20}$$

onde $t(.|\mu, \sigma^2, \nu)$ é a fdp da distribuição t-Student locação-escala, $T(.|\nu)$ é a fda da distribuição t-Student e $d = \frac{(y-\mu)^2}{\sigma^2}$ é a distância de Mahalanobis.

A seguir, é enunciada uma proposição composta de dois resultados necessários para o desenvolvimento das fórmulas fechadas para as esperanças condicionais $z_{ij} = E[Z_{ij}|y_{obsi}, \boldsymbol{\theta_j}]$ e $\epsilon_{rsij} = z_{ij}E[U_i T_i^r Y_i^s|y_{obsi}, \boldsymbol{\theta_j}], r, s = 0, 1, 2$, no contexto da skew-t. A demonstração pode ser encontrada em Basso et al. (2010).

Proposição 5.1. *Seja Y uma variável aleatória com distribuição $ST(\mu, \sigma^2, \lambda, \nu)$ e seja U uma variável aleatória com distribuição $Gamma(\frac{\nu}{2}, \frac{\nu}{2})$. Então, para r = 1, 2, ..., temos que*

1. $E[U^r|Y = y] = \frac{2^r\Gamma(\frac{\nu+1+2r}{2})(\nu+d)^{-r}}{\Gamma(\frac{\nu+1}{2})} \frac{T\left((\frac{\nu+1+2r}{\nu+d})^{1/2}A|\nu+1+2r\right)}{T\left((\frac{\nu+1}{\nu+d})^{1/2}A|\nu+1\right)}$,

2. $E[U^{r/2}W_\phi(\frac{U^{1/2}A}{U^{1/2}A})|Y = y] = \frac{2^{(r-1)/2}\Gamma(\frac{\nu+1+r}{2})(\nu+d)^{(\nu+1)/2}}{\pi^{1/2}\Gamma(\frac{\nu+1}{2})(\nu+d+A^2)^{(\nu+1+r)/2}} \frac{1}{T\left((\frac{\nu+1}{\nu+d})^{1/2}A|\nu+1\right)}$,

onde $\Gamma(.)$ é a função gama.

Em posse dos resultados obtidos na Capítulo 3, na Proposição 5.1, apresentada acima, e após intensivo cálculo, chega-se nas expressões fechadas abaixo. Lembrando que, nesse estudo, consideramos o parâmetro ν constante entre os grupos $\nu = \nu_1 = ... = \nu_G$.

Observações sem censura:

- $z_{ij} = \frac{p_j g(y_i|\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j, \nu)}{\sum_{j=1}^G p_j g(y_i|\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j, \nu)}$;
- $\epsilon_{00ij} = z_{ij}\left[\frac{4\nu^{\nu/2}\Gamma(\frac{\nu+3}{2})(\nu+d_{ij})^{-(\nu+3)/2}}{\sqrt{\pi}\sigma_j\Gamma(\frac{\nu}{2})} \frac{T\left((\frac{\nu+3}{\nu+d_{ij}})^{1/2}A_{ij}|\nu+3\right)}{g(y_i|\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j, \nu)}\right]$;
- $\epsilon_{01ij} = \epsilon_{00ij}y_i$;
- $\epsilon_{02ij} = \epsilon_{00ij}y_i^2$;
- $\epsilon_{10ij} = z_{ij}\left[\mu_{Tij}\frac{\epsilon_{00ij}}{z_{ij}} + \frac{2M_j\nu^{\nu/2}\Gamma(\frac{\nu+2}{2})(\nu+d_{ij}+A_{ij}^2)^{-(\nu+2)/2}}{\pi\sigma_j\Gamma(\frac{\nu}{2})} \frac{1}{g(y_i|\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j, \nu)}\right]$
- $\epsilon_{20ij} = z_{ij}\left[\mu_{Tij}^2\frac{\epsilon_{00ij}}{z_{ij}} + M_j^2 + \mu_{Tij}\frac{2M_j\nu^{\nu/2}\Gamma(\frac{\nu+2}{2})(\nu+d_{ij}+A_{ij}^2)^{-(\nu+2)/2}}{\pi\sigma_j\Gamma(\frac{\nu}{2})} \frac{1}{g(y_i|\boldsymbol{x_i^T\beta_j}, \sigma_j^2, \lambda_j, \nu)}\right]$

- $\epsilon_{11ij} = \epsilon_{10ij} y_i$

Observações censuradas:

- $z_{ij} = \frac{p_j F_{0ij}}{\sum_{j=1}^{G} p_j F_{0ij}}$, em que $F_{0ij} = \int_{c_{i1}}^{c_{i2}} g(y_i | \boldsymbol{x_i^T \beta_j}, \sigma_j^2, \lambda_j, \nu) dy_i$;
- $\epsilon_{00ij} = z_{ij} \left[\frac{P_{0ij}}{F_{0ij}} \right]$, em que $P_{0ij} = \int_{c_{i1}}^{c_{i2}} g(y_i | \boldsymbol{x_i^T \beta_j}, \sigma_j^{2*}, \lambda_j, \nu + 2) dy_i$ e $\sigma_j^{2*} = \frac{\nu}{\nu+2} \sigma_j^2$;
- $\epsilon_{01ij} = \epsilon_{00ij} E[W_i]$, em que $W_i \sim STT(\boldsymbol{x_i^T \beta_j}, \sigma_j^{2*}, \lambda_j, \nu + 2; [c_{i1}, c_{i2}])$ e STT representa a distribuição skew-t truncada;
- $\epsilon_{02ij} = \epsilon_{00ij} E[W_i^2]$;
- $\epsilon_{10ij} = z_{ij} \left[\frac{\Delta_j}{\Gamma_j + \Delta_j^2} (\frac{\epsilon_{01ij}}{z_{ij}} - \frac{\epsilon_{00ij}}{z_{ij}} \boldsymbol{x_i^T \beta_j}) + M_{Tj} c(\nu) W_\Phi \right]$, em que $c(\nu) = \frac{2\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu+1}{2}) \sqrt{\pi \nu (1+\lambda_j^2)}}$, $W_\Phi = \frac{R_{0ij}}{F_{0ij}}$, $R_{0ij} = \int_{c_{i1}}^{c_{i2}} g(y_i | \boldsymbol{x_i^T \beta_j}, \sigma_j^{2**}, 0, \nu + 1) dy_i$ e $\sigma_j^{2**} = \frac{\nu}{(\nu+1)(1+\lambda^2)} \sigma_j^2$;
- $\epsilon_{20ij} = z_{ij} \left[(\frac{\Delta_j}{\Gamma_j + \Delta_j^2})^2 (\frac{\epsilon_{02ij}}{z_{ij}} - 2 \frac{\epsilon_{01ij}}{z_{ij}} \boldsymbol{x_i^T \beta_j} + \frac{\epsilon_{00ij}}{z_{ij}} (\boldsymbol{x_i^T \beta_j})^2) + \frac{\Delta_j}{\Gamma_j + \Delta_j^2} (E[W_0] - \boldsymbol{x_i^T \beta_j}) M_{Tj} c(\nu) W_\Phi + M_{Tj}^2 \right]$, em que $W_0 \sim TT(\boldsymbol{x_i^T \beta_j}, \sigma_j^{2**}, \nu + 1; [c_{i1}, c_{i2}])$ e TT representa a distribuição t-Student truncada;
- $\epsilon_{11ij} = z_{ij} \left[\frac{\Delta_j}{\Gamma_j + \Delta_j^2} (\frac{\epsilon_{02ij}}{z_{ij}} - \frac{\epsilon_{01ij}}{z_{ij}} \boldsymbol{x_i^T \beta_j}) + M_{Tj} c(\nu) W_\Phi E[W_0] \right]$.

Nesse contexto, o primeiro e o segundo momento das variáveis W_i e W_0 também são calculados através do pacote *MomTrunc* (GALARZA et al., 2021), disponível no R.

A Etapa M permanece inalterada, porém, agora há a necessidade de se estimar o parâmetro ν. A obtenção de uma expressão fechada para atualizar o valor desse parâmetro a cada iteração é inviável. Uma opção para contornar esse contratempo seria o uso do Algoritmo ECME (LIU; RUBIN, 1994), uma extensão do Algoritmo EM que foi citada na Seção 2.1. No entanto, seguindo a abordagem utilizada por Zeller, Lachos e Vilca-Labra (2011) e Massuia et al. (2015), esse estudo considerou o parâmetro ν como fixo e um procedimento de seleção de modelo baseado no critério de informação Bayesiano (BIC) foi utilizado para identificar o valor mais apropriado para ν. Essa escolha foi feita pois alguns problemas surgiram durante a estimação dos graus de liberdade, como a convergência para máximos locais diferentes. Uma análise geral dos possíveis problemas decorrentes dessa estimação, principalmente no contexto da distribuição t-Student, pode ser encontrada em Fernández e Steel (1999). Além disso, Lucas (1997) mostra que as estimativas dos parâmetros só se comportam robustamente quando o parâmetro ν é considerado fixo.

A imagem abaixo exemplifica a seleção do parâmetro ν. Nesse caso, foram testados os valores 4, 5, 6 e 7, sendo que, o valor verdadeiro é 5. O menor BIC foi encontrado para $\nu = 5$ e, portanto, esse seria o valor escolhido para o parâmetro.

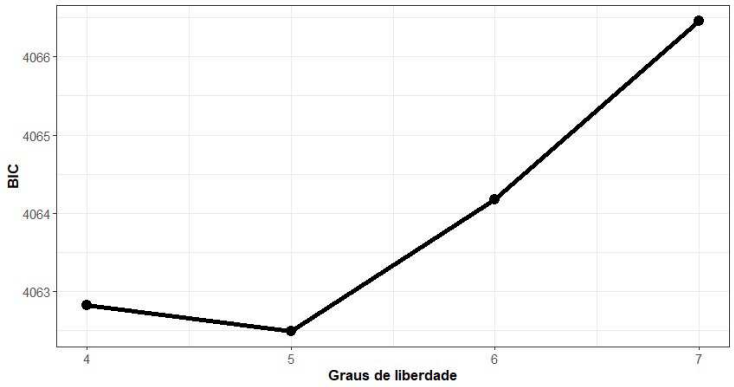


Figura 1 – Seleção do parâmetro ν

6 Aspectos computacionas

Os modelos apresentados neste trabalho, MFR-SN-CR e MFR-ST-CR, foram implementados pelos próprios autores na linguagem de programação R (R Core Team, 2022), pois eles não se encontram disponíveis em nenhum pacote existente. Para realizar essa tarefa, contou-se com o auxílio de funções dos pacotes *MomTrunc* (GALARZA et al., 2021) para calcular os momentos das distribuições skew-normal truncada e skew-t truncada, *mixsmsn* (PRATES et al., 2021) para simular os dados, *sn* (AZZALINI; AZZALINI, 2015) para calcular as funções de densidade de probabilidade e as funções acumuladas da skew-normal e skew-t, *numDeriv* (GILBERT; VARADHAN; GILBERT, 2009) como uma opção computacional para o cálculo dos erros-padrão e *moments* para o cálculo da assimetra no chute inicial.

A implementação dos modelos, assim como os estudos de simulação e a análise de dados reais, que também foram conduzidos através da linguagem R, foram disponibilizados em um repositório do GitHub que pode ser acessado através do link https://github.com/Davioc2/MFR_MESN_CR. Os algoritmos referentes à aplicação dos dois modelos também estão presentes nos Apêndices A (MFR-SN-CR) e B (MFR-ST-CR).

É importante relatar que o algoritmo pode demorar muito a convergir em alguns casos, como: caudas muitos pesadas (válido para o modelo MFR-ST-CR), tamanho amostral muito grande e alto nível de censura. Portanto, ajustar os modelos a dados com 3000 observações e nível de censura de 35%, o que foi feito no estudo de simulação que será apresentado no próximo Capítulo, foi uma tarefa muito demorada. Caso o parâmetro ν fosse igual a 2 (caudas muito pesadas) nessa configuração, o algoritmo poderia rodar por mais de um dia. O motivo dessa lentidão é a parte da Etapa E relativa aos dados censurados, que necessita da função meanvarTMD() do pacote *MomTrunc* para encontrar o primeiro e o segundo momento das distribuições skew-normal truncada e skew-t truncada. Essa função não roda instantaneamente e não trabalha vetorialmente. Portanto, dentro de cada iteração, há um loop para estimar esses momentos.

A função meanvarTMD() também retorna erros em alguns casos, como: ν e número de componentes distantes dos valores reais.

7 Estudos de simulação

Três estudos de simulação de Monte Carlo foram conduzidos para ilustrar a funcionalidade do modelo MFR-MESN-CR na prática. Para simplificar o estudo, somente casos de censura à esquerda foram abordados.

7.1 Recuperação dos parâmetros

O objetivo desta Seção é verificar se os algoritmos propostos para os modelos MFR-SN-CR e MFR-ST-CR recuperam corretamente os parâmetros originais simulados. As propriedades assintóticas dos estimadores de máxima-verossimilhança, como a normalidade, a consistência e a eficiência, foram verificadas para diferentes tamanhos de amostra (N = 500, 1000 e 3000) e níveis de censura à esquerda (C = 8%, 20% e 35%).

7.1.1 MFR-SN-CR

Foram gerados 500 conjuntos de dados de um modelo MFR-SN-CR com dois componentes e $\boldsymbol{x}_i^T = (1, x_{i1}, x_{i2})$, tal que $x_{i1} \sim N(5, 9)$ e $x_{i2} \sim N(20, 16)$. Os parâmetros utilizados foram os seguintes: $\boldsymbol{\beta}_1 = (-1, -4, -3)$, $\boldsymbol{\beta}_2 = (3, 7, 9)$, $\sigma_1^2 = 3$, $\sigma_1^2 = 1$, $\lambda_1 = 4$ e $\lambda_2 = 3$. Vale ressaltar que foram simulados 500 conjuntos de dados para cada tamanho de amostra e nível de censura citados no início dessa Seção (7.1).

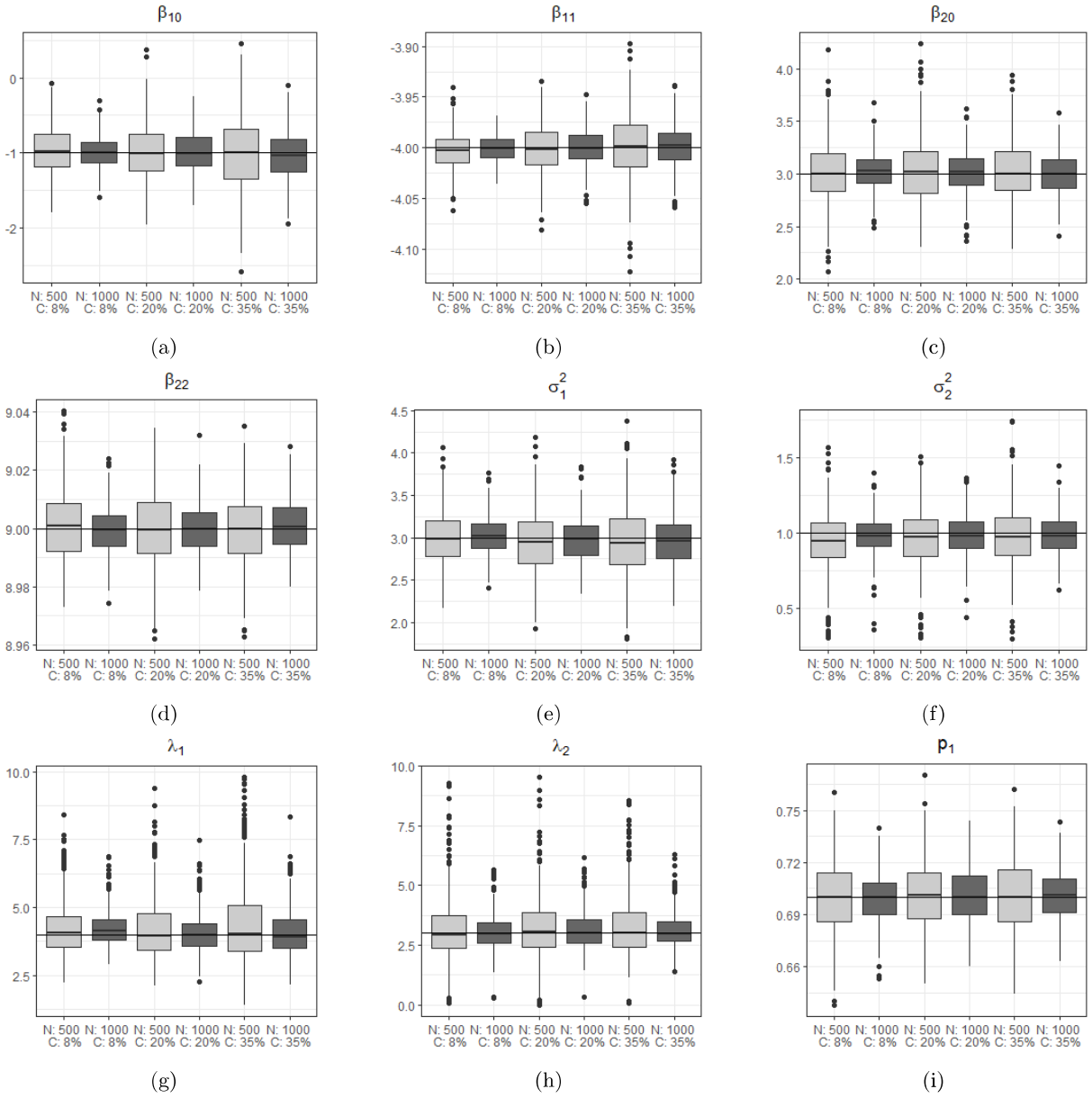
O nível de censura desejado é obtido através do percentil c% da seguinte maneira: calcula-se o percentil c% ($P_{c\%}$) e os valores menores ou iguais a ele são transformados em $P_{c\%}$.

Na Figura 2, observam-se os boxplots referentes às estimativas de alguns dos parâmetros para N = 500 e 1000 e níveis de censura de 8%, 20% e 35%. Para um mesmo nível de censura, pode-se perceber a redução da variabilidade das estimativas com o aumento do tamanho da amostra, confirmando a eficiência do estimador. Além disso, para um mesmo tamanho da amostra percebe-se que, quanto maior o nível de censura, maior a variabilidade.

É interessante salientar que, devido às configurações do modelo simulado, os valores de Y pertencentes ao Grupo 1 vão de aproximadamente -135 a 0, enquanto aqueles pertencentes ao Grupo 2 vão de aproximadamente 100 a 320. Portanto, de acordo com o processo de censura explicado no início desta Seção, os valores censurados estarão todos no Grupo 1 e somente as estimativas referentes aos parâmetros desse grupo sofrerão com o aumento do nível de censura. Veja que nas Figuras 2.c, 2.d, 2.f e 2.h, que se referem às estimativas dos parâmetros do Grupo 2, se tomarmos um mesmo N, não é visível um crescimento na variabilidade à medida que o nível de censura aumenta. Veja a Figura 3, que representa o histograma da variável Y de uma amostra e o corte de censura a um nível de 35%.

Comparando-se as estimativas para os diferentes tamanhos amostrais (n) e níveis de censura (c) através do erro quadrático médio (EQM), notou-se que elas melhoram à medida que N aumenta. Essa comparação pode ser vista na Figura 4. Ademais, seguindo o que foi dito no parágrafo anterior, o nível de censura só afeta a qualidade das estimativas dos parâmetros do Grupo 01, sendo que, quanto maior o c, pior a estimativa. Por fim, é possível visualizar que os parâmetros de locação, exceto o intercepto, possuem EQM consideravelmente inferior aos parâmetros de escala e, principalmente, de forma, sendo estes últimos considerados de mais difícil estimação. Vale ressaltar que a dificuldade em estimar o valor do intercepto é causada por estarmos considerando um modelo não centrado, como foi explicado no início da Seção 3.1.

A normalidade assintótica dos estimadores foi verificada através do teste de normalidade de Shapiro-Wilk, o qual tem como hipótese nula a normalidade do vetor. Os valores-p referentes aos testes podem ser verificados na Tabela 1. A partir dos resultados, constatou-se que os estimadores dos parâmetros de locação e de proporção já alcançam a normalidade com tamanho amostral igual a 500, fato que pode ser visualizado



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figura 2 – Boxplots das estimativas dos parâmetros do modelo MFR-SN-CR

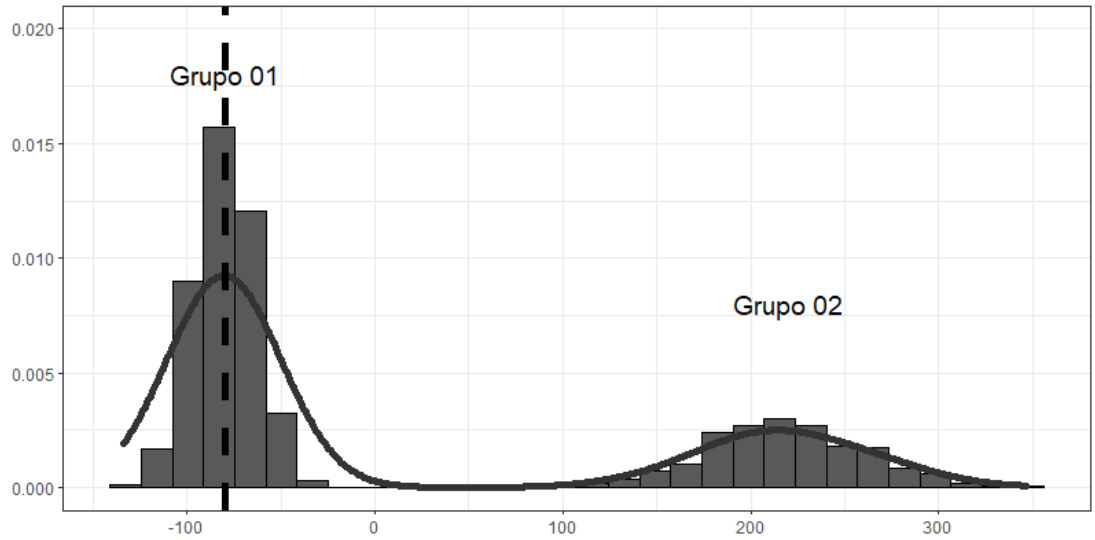


Figura 3 – Histograma da variável resposta dos dados gerados pelo modelo MFR-SN-CR e linha horizontal representando o percentil 35%

Tabela 1 – Valor-p do teste de Shapiro-Wilk para verificar a normalidade dos estimadores dos parâmetros do modelo MFR-SN-CR

Parâmetros		**C: 8%**			**C: 35%**	
	N: 500	N: 1000	N: 3000	N: 500	N: 1000	N: 3000
β_{10}	0,6144	0,9494	0,2451	0,9549	0,8337	0,1329
β_{11}	0,8514	0,2528	0,1323	0,0609	0,9134	0,9351
$\beta 12$	0,9448	0,1697	0,4632	0,7588	0,6340	0,2386
$\beta 20$	0,1238	0,5416	0,1242	0,2137	0,5823	0,6822
β_{21}	0,0629	9,40x10-4	0,9154	0,1865	0,4255	0,3575
β_{22}	0,0237	0,6433	0,8072	0,8995	0,3556	0,9478
σ_1^2	0,3365	0,5425	0,2651	0,0901	0,7881	0,0504
σ_2^2	5,73x10-4	2,00x10-6	0,8196	0,0010	0,5718	0,1115
λ_1	6,10x10-12	6,95x10-9	1,60x10-3	1,17x10-15	1,22x10-9	1,20x10-7
λ_2	5,25x10-16	5,28x10-8	1,37x10-4	6,99x10-15	1,21x10-9	4,20x10-5
p_1	0,7411	0,5644	0,3317	0,0866	0,6491	0,1290

através dos histogramas das estimativas de alguns dos parâmetros de locação para N = 500 (Figura 5) e N = 3000 (Figura 6) para um nível de censura de 35%. Os estimadores dos parâmetros de escala também atingiram a normalidade assintótica, porém, ela só foi alcançada em ambos os grupos para N = 3000 com C = 8% e a partir de N = 1000 para C = 35%, mostrando que o alcance dessa propriedade exige tamanhos amostrais maiores em comparação com os estimadores dos parâmetros de locação. Por fim, o estimador dos parâmetros de forma não atingiu a normalidade assintótica em nenhum caso, logo, não é possível realizar análises que exigem essa propriedade, como verificar a significância das estimativa, para tamanhos amostrais menores ou iguais a 3000. Maiores valores de N não foram testados devido às dificuldades computacionais relatadas no Capítulo 6 para N muito grande.

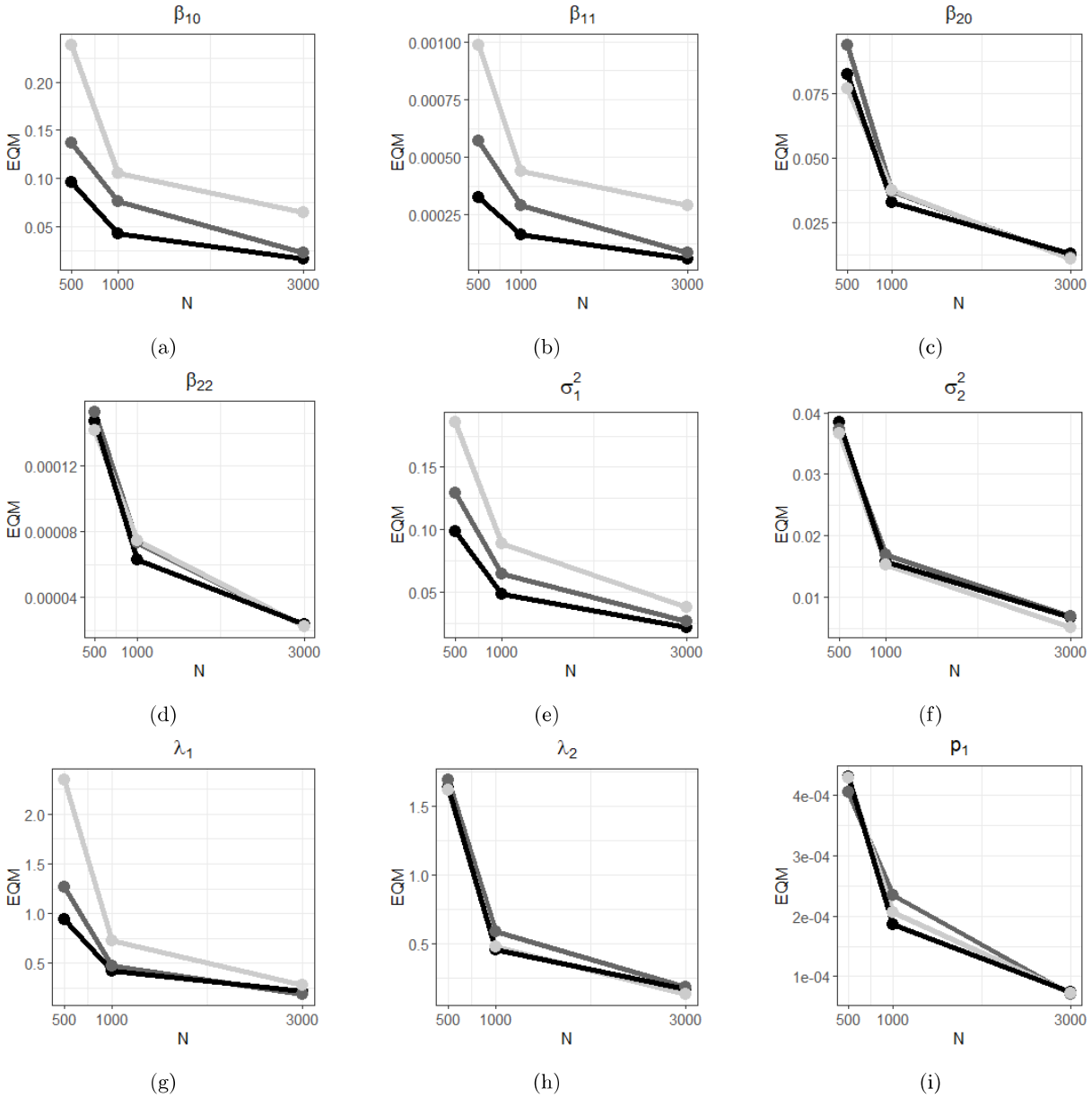
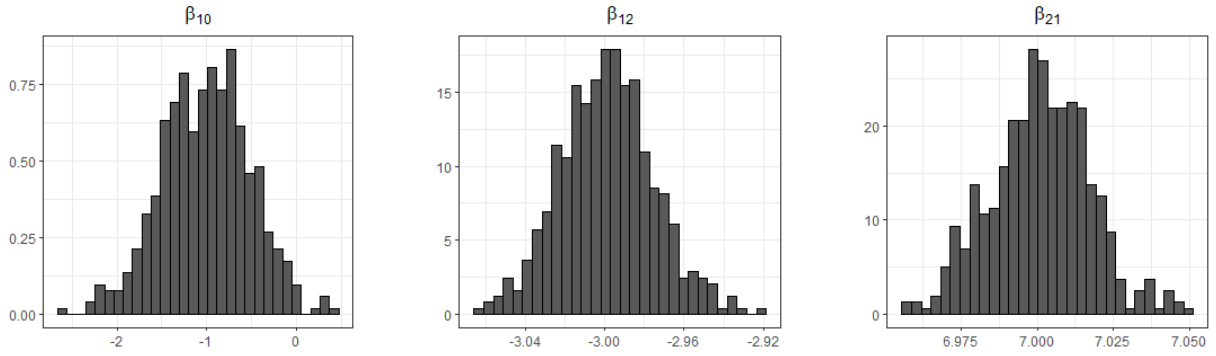
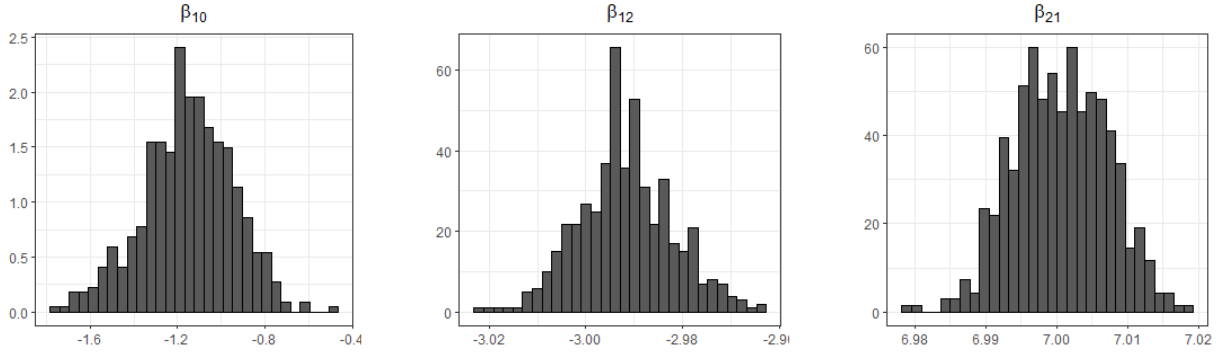


Figura 4 – Erro quadrático médio dos estimadores dos parâmetros do modelo MFR-SN-CR. Nível de censura 8%: linha preta, 20%: cinza escuro e 35%: cinza claro



(a) (b) (c)

Figura 5 – Histograma das estimativas de alguns parâmetros de locação do modelo MFR-SN-CR para N = 500



(a) (b) (c)

Figura 6 – Histograma das estimativas de alguns parâmetros de locação do modelo MFR-SN-CR para N = 3000

7.1.2 MFR-ST-CR

Foram gerados 500 conjuntos de dados de um modelo MFR-ST-CR com dois componentes e $\boldsymbol{x}_i^T = (1, x_{i1}, x_{i2}, x_{i3})$, tal que $x_{i1} \sim U(1,5)$, $x_{i2} \sim U(-2,2)$ e $x_{i3} \sim U(1,4)$. Os parâmetros utilizados foram os seguintes: $\boldsymbol{\beta}_1 = (-1,1,2,3)$, $\boldsymbol{\beta}_2 = (0,-1,-2,-3)$, $\sigma_1^2 = 2$, $\sigma_1^2 = 1$, $\lambda_1 = 3$, $\lambda_2 = -2$ e $\nu = 5$. Vale ressaltar que foram simulados 500 conjuntos de dados para cada tamanho de amostra e nível de censura citados no início dessa Seção (7.1).

Na Figura 7, observam-se os boxplots referentes às estimativas de alguns dos parâmetros para N = 500 e 1000 e níveis de censura de 8%, 20% e 35%. Assim como no caso anterior (MFR-SN-CR) é possível notar que, para um mesmo nível de censura, houve redução da variabilidade das estimativas com o aumento do tamanho da amostra, confirmando a eficiência do estimador. Além disso, para um mesmo tamanho da amostra percebeu-se que, quanto maior o nível de censura, maior a variabilidade. Porém, diferente do caso anterior, o Grupo 2 foi atingido pela censura, portanto, houve aumento de variabilidade devido à censura somente nas estimativas dos parâmetros referentes ao Grupo 2.

Visualizando-se a Figura 8, é possível verificar através do EQM o efeito positivo do aumento do tamanho amostral nas estimativas dos parâmetros. Além disso, nota-se que, para um mesmo N, as estimativas referentes aos parâmetros do Grupo 2, o qual sofreu a censura, pioram à medida que o nível de censura aumenta. Ao

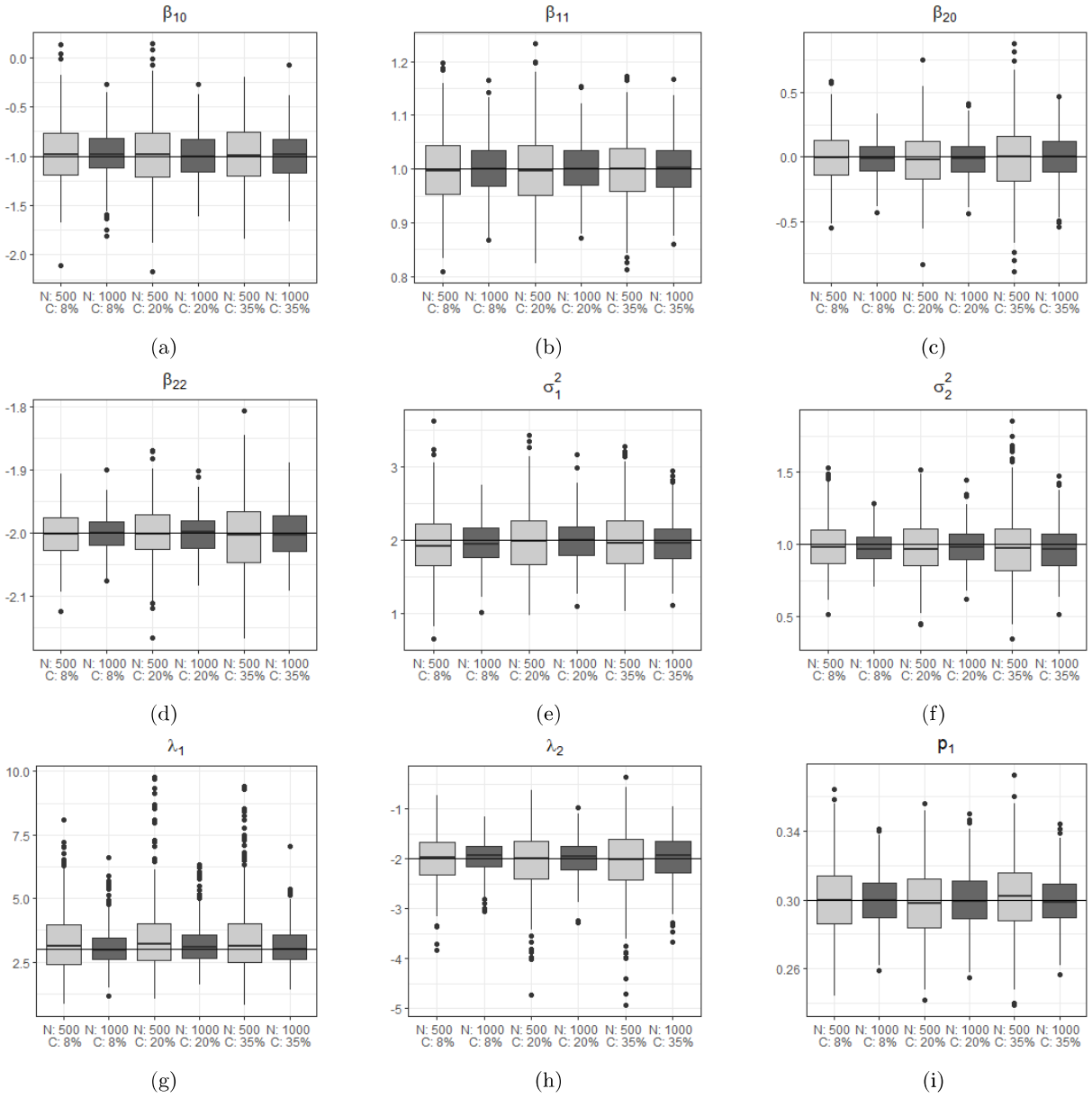


Figura 7 – Boxplots das estimativas dos parâmetros do modelo MFR-ST-CR

encontro dos resultados obtidos no caso do modelo MFR-SN-CR, o EQM para as estimativas dos parâmetros de locação, com execeção do intercepto, foram menores que para os parâmetros de escala e forma.

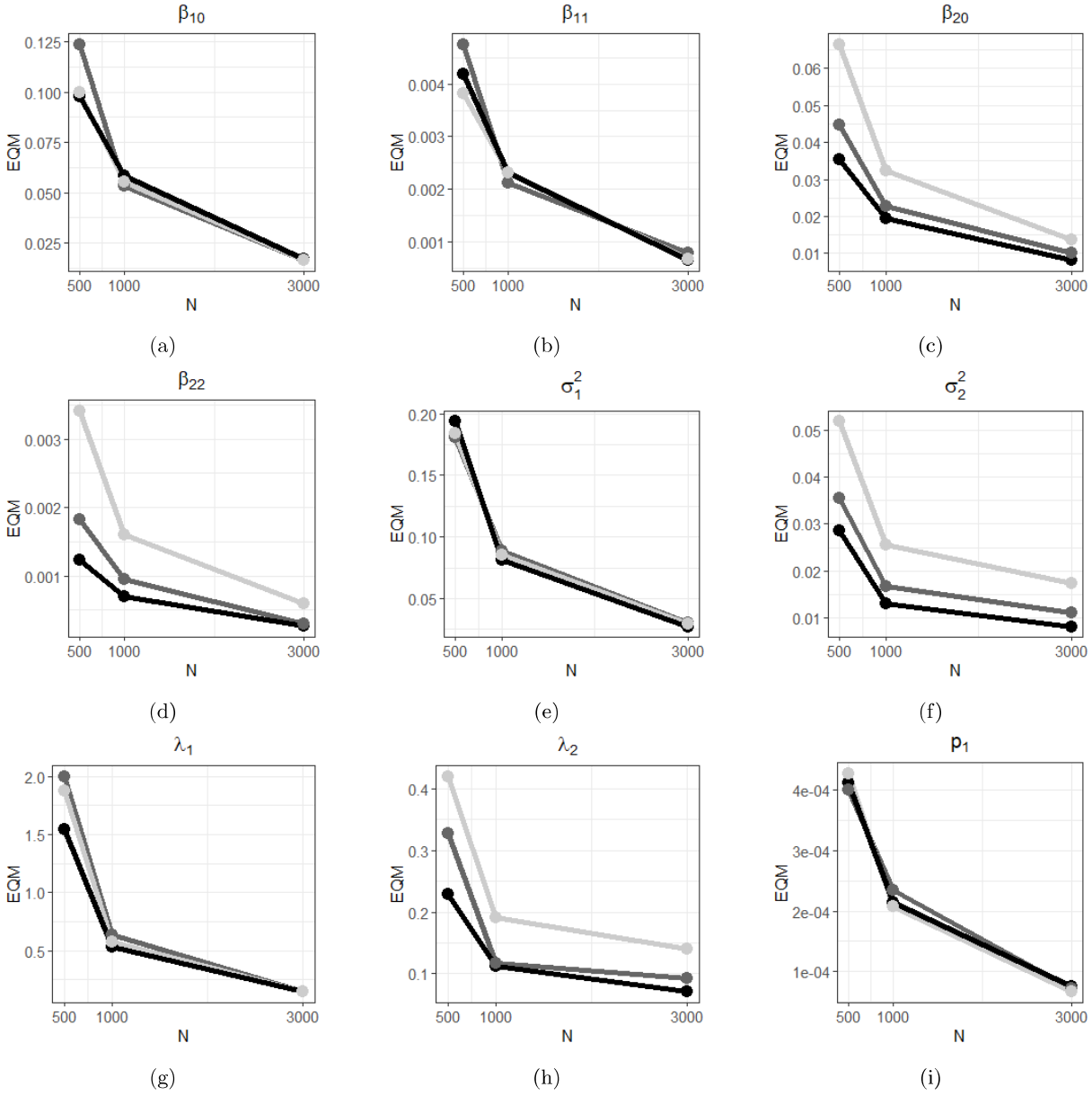


Figura 8 – Erro quadrático médio dos estimadores dos parâmetros do modelo MFR-ST-CR. Nível de censura 8%: linha preta, 20%: cinza escuro e 35%: cinza claro

A partir dos valores-p referentes ao teste de Shapiro-Wilk apresentados na Tabela 2, constatou-se que os estimadores dos parâmetros de locação e de proporção já alcançam a normalidade com tamanho amostral igual a 500, fato que pode ser visualizado através dos histogramas das estimativas de alguns dos parâmetros de locação para N = 500 (Figura 9) e N = 3000 (Figura 10) para um nível de censura de 35%. Os estimadores dos parâmetros de escala também atingiram a normalidade assintótica, porém, ela só foi alcançada em ambos os grupos para N = 3000 com C = 35% e a partir de N = 1000 para C = 8%, mostrando que o alcance dessa propriedade exige tamanhos amostrais maiores em comparação com os estimadores dos parâmetros de

Tabela 2 – Valor-p do teste de Shapiro-Wilk para verificar a normalidade dos estimadores dos parâmetros do modelo MFR-ST-CR

Parâmetros		**C: 8%**			**C: 35%**	
	N: 500	N: 1000	N: 3000	N: 500	N: 1000	N: 3000
β_{10}	0,0123	0,3275	0,8677	0,1006	0,7622	0,0975
β_{11}	0,1730	0,3148	0,6375	0,8992	0,6334	0,6356
β_{12}	0,6739	0,2838	0,1327	0,3027	0,3897	0,1732
β_{13}	0,6709	0,4958	0,5796	0,5944	0,5159	0,9781
β_{20}	0,8915	0,5898	0,9442	0,6622	0,4829	0,7228
β_{21}	0,5908	0,2236	0,9730	0,5273	0,6081	0,5533
β_{22}	0,8240	0,4190	0,0185	0,4548	0,0107	0,2252
β_{23}	0,9291	0,6588	0,7215	0,5046	0,8619	0,3455
σ_1^2	0,0042	0,6282	0,5682	0,0016	0,1859	0,1801
σ_2^2	0,0079	0,0947	0,3084	1,63x10-5	0,0050	0,9549
λ_1	2,63x10-10	2,05x10-11	0,0396	4,39x10-17	1,04x10-7	4,88x10-5
λ_2	6,14x10-4	4,99x10-5	0,0899	3,52x10-7	2,27x10-5	0,3237
p_1	0,7584	0,5858	0,8861	0,5907	0,6487	0,2055

locação. Por fim, o estimador dos parâmetros de forma não atingiu a normalidade assintótica em nenhum caso, logo, não é possível realizar análises que exigem essa propriedade, como verificar a significância das estimativa, para tamanhos amostrais menores ou iguais a 3000.

7.2 Critério de seleção

Nesta Seção, dois conhecidos critérios de seleção serão avaliados quanto a sua capacidade de escolher o modelo apropriado para certo conjunto de dados. São eles o critério de informação de Akaike (AIC), dado por $-2l(\hat{\boldsymbol{\theta}}) + 2k$, onde $l(\hat{\boldsymbol{\theta}})$ é a função de log-verossimilhança e k é o número de parâmetros, e o critério de informação Bayesiano (BIC), dado por $-2l(\hat{\boldsymbol{\theta}}) + log(n)k$.

Foram simuladas 100 amostras de tamanho 1000 de um modelo MFR-ST-CR com três componentes e vetor de variáveis explicativas $\boldsymbol{x}_i^T = (1, x_{i1}), i = 1, ..., n$, tal que $x_{i1} \sim U(-2, 2)$. Os parâmetros originais foram dados por $\boldsymbol{\beta_1} = (-4, 4)^T$, $\boldsymbol{\beta_2} = (0, -2)^T$, $\boldsymbol{\beta_3} = (0, 4)^T$, $\sigma_1^2 = 1$, $\sigma_2^2 = 4$, $\sigma_3^2 = 1$, $\lambda_1 = -2$, $\lambda_2 = 3$, $\lambda_3 = 4$, $p_1 = 0.5$, $p_2 = 0.2$ e $\nu = 4$. A simulação foi feita para três níveis diferentes de censura à esquerda: 5%, 15% e 30%.

Para cada amostra, foram ajustados os seguintes modelos: MFR-SN-CR(1), MFR-SN-CR(2), MFR-SN-CR(3), MFR-SN-CR(4), MFR-ST-CR(1), MFR-ST-CR(2), MFR-ST-CR(3) e MFR-ST-CR(4), onde o número entre parênteses representa a quantidade de componentes. Para cada um deles, os valores de AIC e BIC foram armazenados e o modelo escolhido foi aquele que exibiu o menor valor em cada caso.

Na Tabela 3 é possível verificar a porcentagem de vezes que cada modelo foi selecionado por cada critério em cada nível de censura. Por exemplo, para um nível de censura de 5%, o AIC escolheu o modelo MFR-ST-CR (3), que é o correto, para 86 das 100 amostras. Observa-se que ambos os critérios fizeram a escolha correta na grande maioria das vezes. Além disso, é possível notar que para os níveis de 5% e 15% os critérios sempre escolheram corretamente a distribuição dos erros aleatórios, porém, nem sempre acertaram o número de componentes. Comparando-se os dois, é nítida a superioridade do BIC, e, portanto, este será utilizado para a escolha do melhor modelo no Capítulo seguinte, referente à análise de dados reais. Quando o nível de censura passou para 30%, a tarefa de selecionar o modelo correto ficou mais difícil, sendo que, nesse caso, os critérios enganaram-se, mesmo que poucas vezes, na distribuição, no número de componentes e, algumas

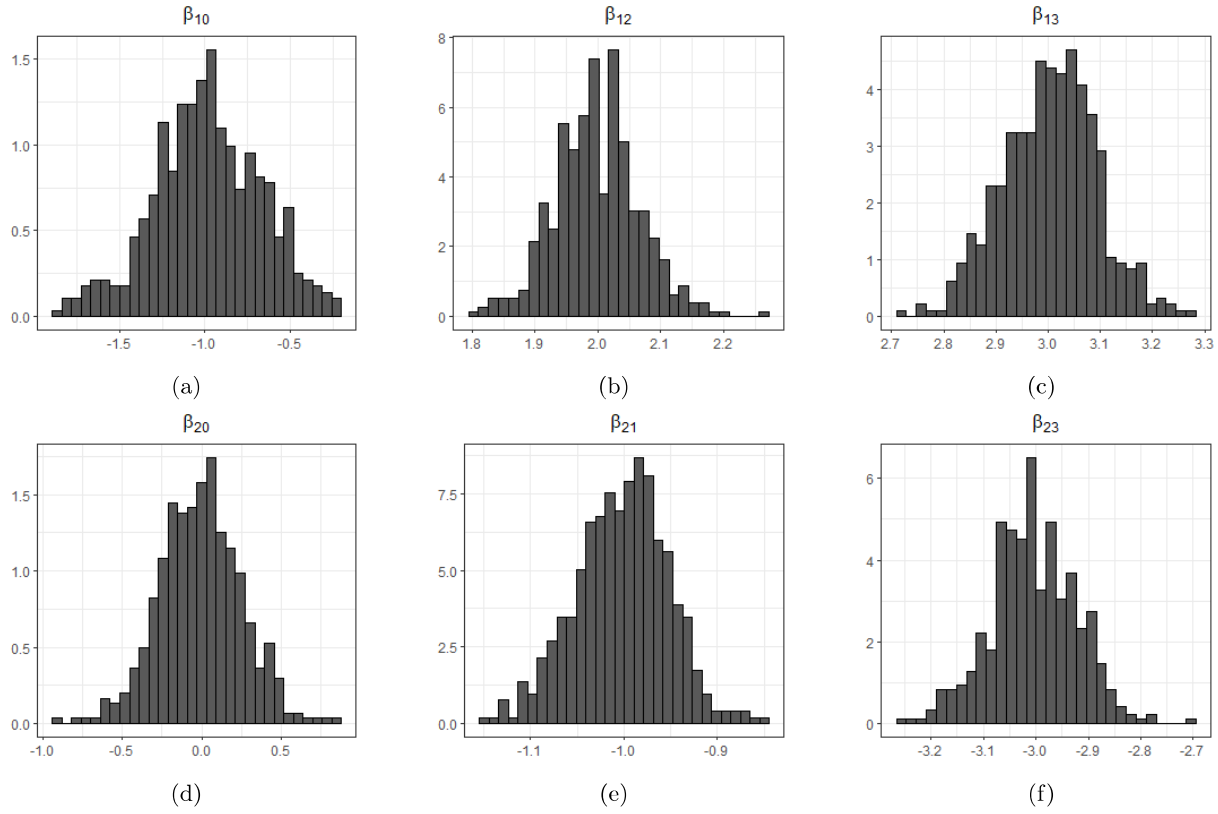


Figura 9 – Histograma das estimativas de alguns parâmetros de locação do modelo MFR-ST-CR para N = 500

Tabela 3 – Porcentagem de vezes em que cada modelo foi selecionado segundo cada nível de censura e critério

Censura	**Critério**	**SN(1)**	**SN(2)**	**SN(3)**	**SN(4)**	**ST(1)**	**ST(2)**	**ST(3)**	**ST(4)**
5%	AIC	0	0	0	0	0	0	86	14
	BIC	0	0	0	0	0	0	98	2
15%	AIC	0	0	0	0	0	0	89	11
	BIC	0	0	0	0	0	0	98	2
30%	AIC	0	6	17	0	0	1	73	3
	BIC	0	8	15	0	0	3	71	3

vezes, nos dois ao mesmo tempo.

Na Figura 11 observa-se os valores de AIC e BIC para o ajuste dos modelos MFR-SN-CR(3), MFR-ST-CR(1), MFR-ST-CR(2), MFR-ST-CR(3) e MFR-ST-CR(4). Ambos critérios acusaram o modelo com distribuição skew-t e 1 componente como aquele com o pior ajuste, seguido do skew-t com 2 componentes e do skew-normal com 3 componentes. No gráfico referente ao AIC, percebe-se que há sobreposição entre as linhas que representam o modelo skew-t com 3 e 4 componentes, enquanto que no gráfico de BIC, é possível diferenciar as duas linhas, sendo que a linha preta, referente ao modelo original dos dados (skew-t com 3 componentes) está um pouco abaixo. Essa análise reafirma a soberania do BIC na seleção do modelo correto.

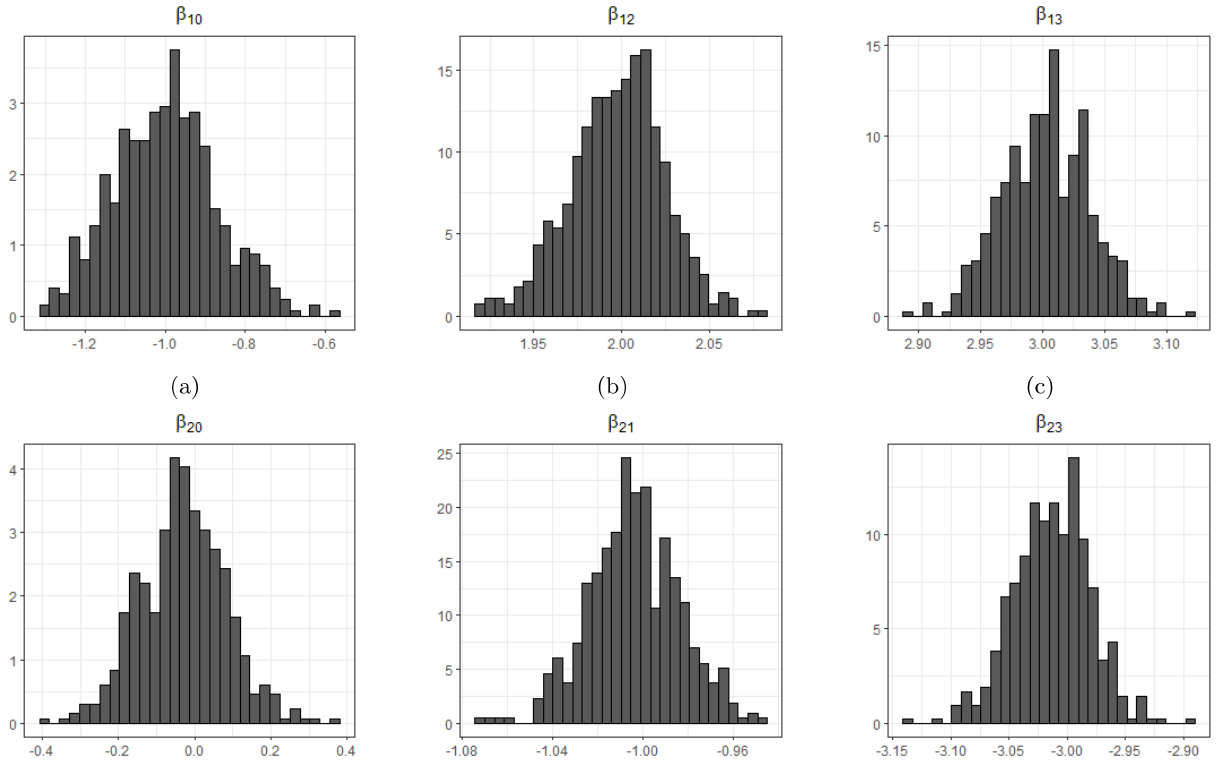


Figura 10 – Histograma das estimativas de alguns parâmetros de locação do modelo MFR-ST-CR para N = 3000

7.3 Classificação

O método de mistura finita de distribuições possui outra aplicabilidade importante além do ajuste de modelos a dados com heterogeneidade não observada, a classificação. A partir da matriz $\boldsymbol{Z}$ encontrada na Etapa E do Algoritmo EM, podemos classificar as observações em cada grupo. Por exemplo, caso tenha sido ajustado um modelo MFR-MESN-CR considerando-se três componentes, cada observação i, i = 1,...,n será classificada em um dos três grupos de acordo com o valor máximo do vetor $\boldsymbol{Z_i} = (z_{i1}, z_{i2}, z_{i3})^T$. Então, se o máximo de $\boldsymbol{Z_i}$, nesse caso, for o valor z_{i2}, a observação será considerada como pertencente ao Grupo 2.

Vale relatar que no contexto de misturas finitas pode haver troca de label no algoritmo. Portanto, os resultados referentes ao Grupo 1 podem sair denominados como Grupo 2 na saída do algoritmo. Para evitar isso, foi criado um parâmetro que define a ordem desejada de acordo com as proporções. Por exemplo, imagine que foram simulados dois grupos, o Grupo 1 com $p_1 = 0.7$ e o Grupo 2 com $p_2 = 0.3$ Para mantermos o mesmo label, passamos como parâmetro os índices dos grupos em ordem crescente de proporção, nesse caso, $ordem = c(2, 1)$. Assim, após o uso do algoritmo kmeans no chute inicial, o índice 2 será integrado ao Grupo com o menor número de observações (menor p) e o índice 1 ou Grupo com o maior número de observações. Esse artifício evitou a troca de label em casos onde as proporções são diferentes, caso sejam 50% a 50%, por exemplo, não é possível evitar a troca de label utilizando esse recurso.

Nesta Seção, os modelos considerados neste trabalho, MFR-SN-CR e MFR-ST-CR serão avaliados quanto à capacidade de classificar corretamente as observações. A acurácia (ACC) será utilizada para comparar os dois modelos, essa métrica de desempenho é calculada pelo total de classificações feitas corretamente (CC)

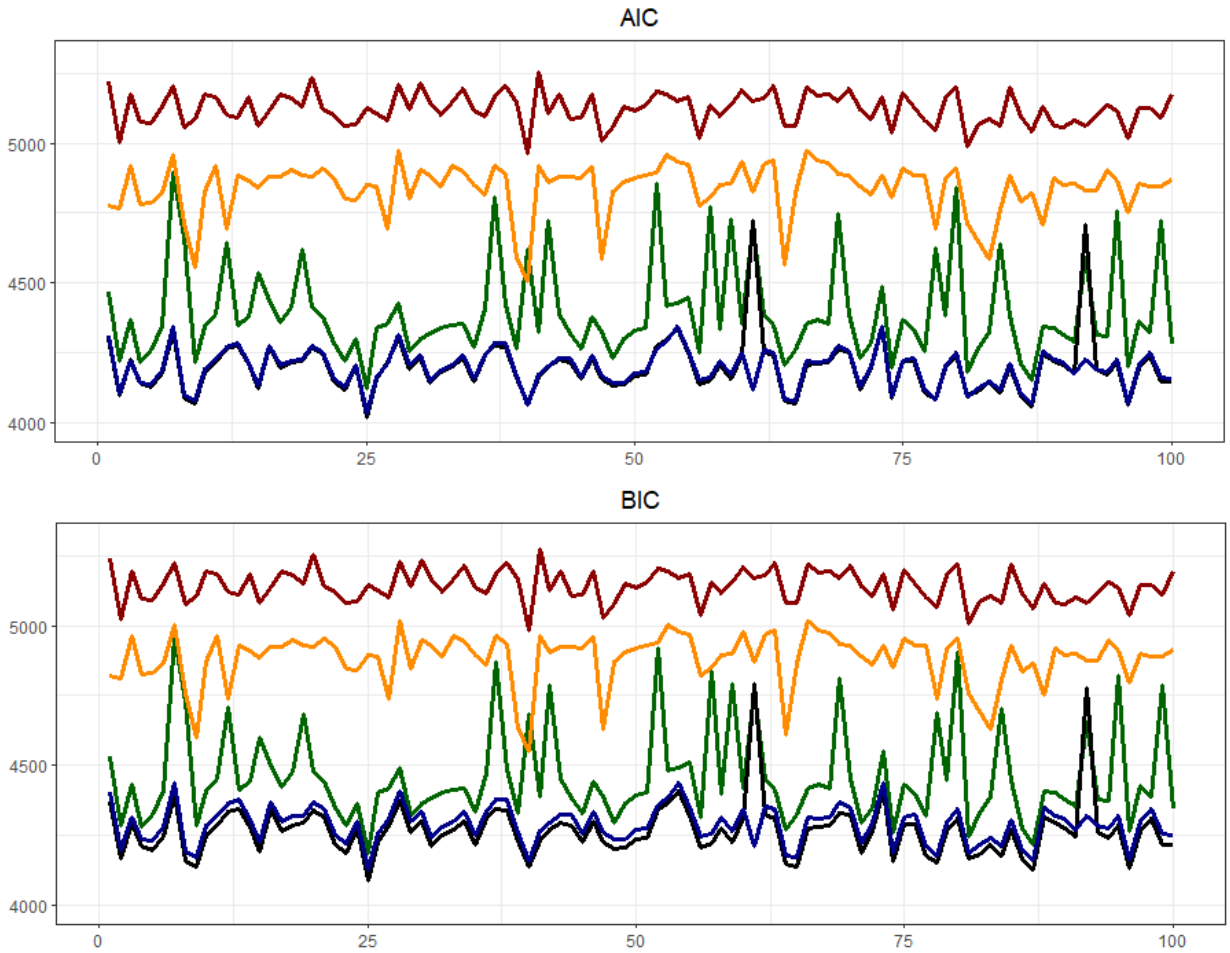


Figura 11 – Valores de AIC e BIC para 100 amostras com 15% de censura à esquerda. Linha verde (do meio): MFR-SN-CR(3), vermelha (no topo): MFR-ST-CR(1), laranja (entre o meio e o topo): MFR-ST-CR(2), preta: MFR-ST-CR(3) e azul: MFR-ST-CR(1)

dividido pelo total de observações, ou seja, $ACC = \frac{1}{n}\sum_{i=1}^{n} CC_i$.

Foram simulados dados de uma MFR-ST-CR com dois componentes, $\nu = 4$, censura à esquerda de 5% e 10% e vetor de variáveis explicativas $\boldsymbol{x}_i^T = (1, x_{i1}), i = 1, ..., n$, tal que $x_{i1} \sim U(-2, 2)$. Os parâmetros foram dados por $\boldsymbol{\beta_1} = (0, -2)^T$, $\boldsymbol{\beta_2} = (0, 4)^T$, $\sigma_1^2 = 4$, $\sigma_2^2 = 1$, $\lambda_1 = 3$, $\lambda_2 = 4$ e $p_1 = 0.6$. A esses dados, foram ajustados os modelos MFR-SN-CR e MFR-ST-CR com $\nu = 4$, ambos com dois componentes. Os resultados podem ser vistos na Tabela 4.

O mesmo foi feito para dados simulados de uma MFR-SSL-CR (erros distribuídos segundo uma skew-slash) com dois componentes, $\nu = 2$ e com a mesma configuração utilizada acima. A esses dados também foram ajustados os modelos MFR-SN-CR e MFR-ST-CR com $\nu = 15$, ambos com dois componentes. O valor de graus de liberdade para o modelo MFR-ST-CR foi escolhido através do método apresentado no Capítulo 5. Os resultados podem ser vistos na Tabela 5.

Neste estudo de simulação, vimos que não houve difença perceptível entre a acurácia média de classificação dos dois modelos. Portanto, caso esse modelo seja usado com o intuito de agrupar observações nesse contexto, de misturas finitas de regressões lineares com respostas censuradas e erros assimétricos, não há necessidade

Tabela 4 – Média da acurácia dos modelos. Dados simulados de uma MFR-ST-CR com dois componentes e $\nu = 4$

Censura	**MFR-SN-CR**	**MFR-ST-CR**
5%	0,8219	0,8282
10%	0,7583	0,7676

Tabela 5 – Média da acurácia dos modelos. Dados simulados de uma MFR-SSL-CR com dois componentes e $\nu = 2$

Censura	**MFR-SN-CR**	**MFR-ST-CR**
5%	0,8023	0,7912
10%	0,7453	0,7441

de ajustar os dados aos modelos mais complexos como o MFR-ST-CR, pois o MFR-SN-CR cumprirá o mesmo papel. Além disso, percebe-se que à medida que o nível de censura aumenta, a tarefa de classificar as observações corretamente se torna mais difícil.

8 Análise de dados reais

A análise de dados reais foi feita em um conjunto de dados obtido em Mroz (1984), o qual está disponível no pacote *SMNCensReg* do R. Esse conjunto de dados contém informações sobre o total de horas trabalhadas fora de casa por mulheres casadas, além de outras informações sobre elas. Estão disponíveis 753 observações, das quais 325 não trabalharam fora de casa, ou seja, apresentam o valor 0 na variável resposta (horas trabalhadas fora de casa). Portanto, os dados apresentam censura à direita a um nível de 43,16%.

Esses dados foram analisados por Caudill (2012) e Karlsson e Laitila (2014) no contexto de mistura finita de regressões com erros distribuídos normalmente e por Zeller et al. (2019) com erros distribuídos na classe MEN. Seguindo a ideia dos artigos supracitados, as variáveis utilizadas para explicar as horas anuais de trabalho fora de casa dividida por 1000 (Y) foram a escolaridade da mulher em anos (X_1), a idade (X_2), a experiência anterior no mercado de trabalho (X_3) e o quadrado desta última (X_4), com o intuito de realçar as maiores experiências no mercado de trabalho.

A Figura 12 sugere a presença de multimodalidade na variável resposta, portanto, o uso de modelos com mistura finita para ajustar esses dados é recomendável. Pelo histograma, parece que os dados são provenientes de dois grupos distintos, ou seja, a escolha ideal seria ajustá-los através de um modelo de mistura finita com duas componentes. Por segurança, os modelos MFR-SN-CR e MFR-ST-CR foram testados para G = 1,...,4.

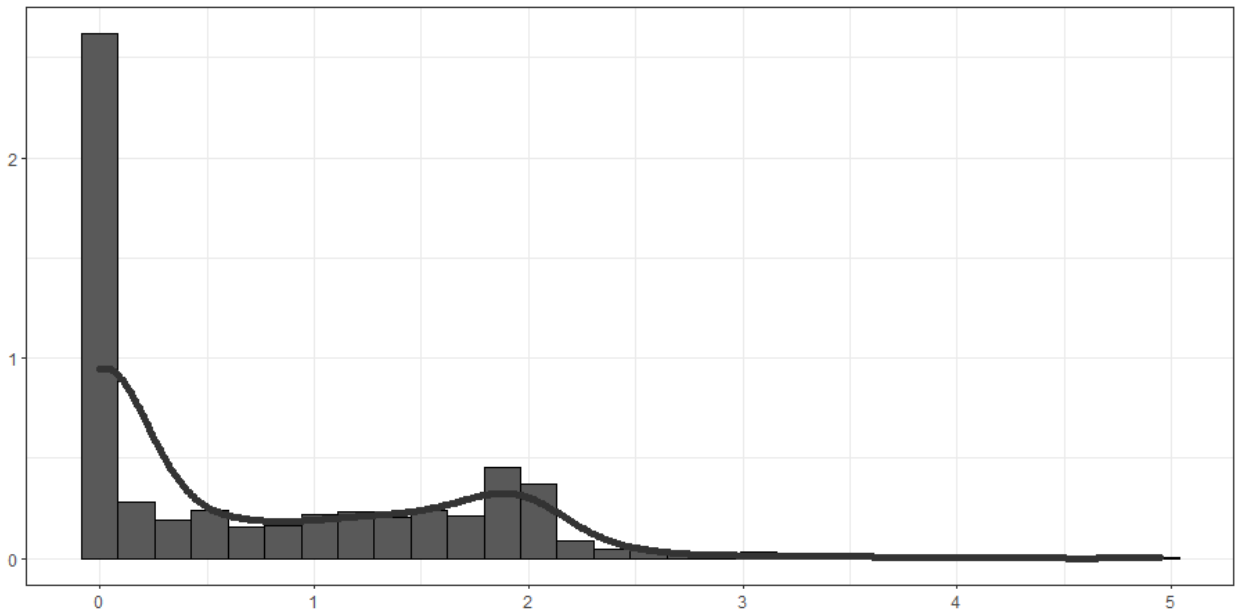


Figura 12 – Histograma e curva de densidade estimada da variável horas trabalhadas fora de casa dividida por 1000

Os modelos MFR-ST-CR foram ajustados com $\nu = 3$, pois, para todas as quantidades de componentes testadas, esse número de graus de liberdade foi o que obteve os menores valores de BIC, como pode ser visto na Figura 4 para o caso da MFR-ST-CR com dois componente, que como será visto, foi o melhor modelo.

Na Tabela 6, nota-se através do BIC que o modelo que melhor se ajustou aos dados foi o MFR-ST-CR com dois componentes.

Os parâmetros estimados, assim como os erros-padrão e os valores-p encontrados com o ajuste do modelo MFR-ST-CR com dois componentes podem ser vistos na Tabela 7. A coluna de significância indica se os

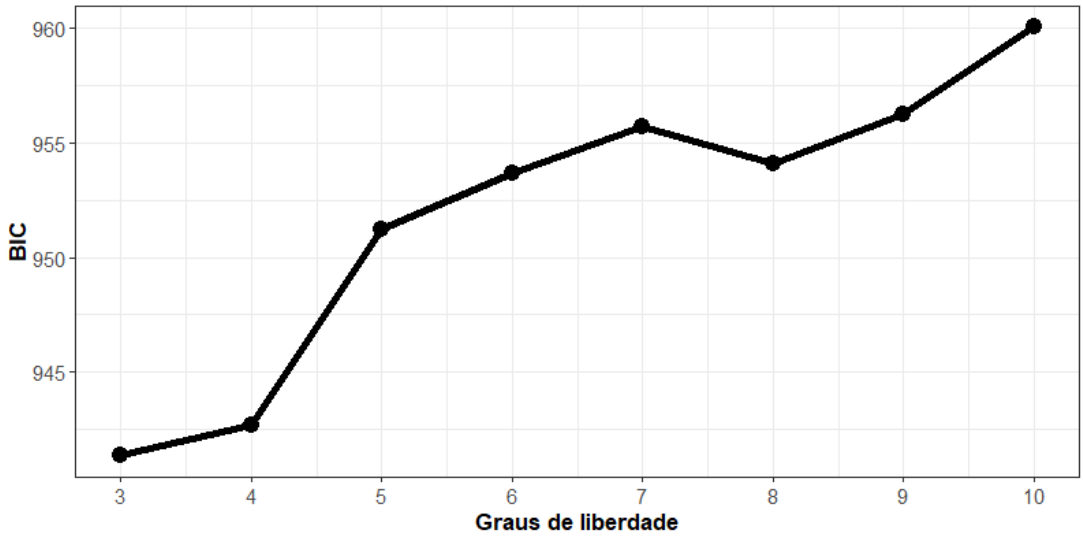


Figura 13 – Seleção do parâmetro ν para ajuste do modelo MFR-ST-CR com um componente nos dados de Mroz (1984)

Tabela 6 – Critérios de seleção para os modelos MFR-SN-CR e MFR-ST-CR com G = 1,...,3 para os dados de Mroz (1984)

Modelo	G	BIC
MFR-SN-CR	1	1019,32
MFR-SN-CR	2	972,54
MFR-SN-CR	3	979,82
MFR-SN-CR	4	1035,44
MFR-ST-CR	1	1067,82
MFR-ST-CR	2	**941,35**
MFR-ST-CR	3	951,89
MFR-ST-CR	4	987,20

parâmetros são significantes a um nível de 10% (.), 5% (*), 1% (**) ou 0,1% (***). Essa significância não é mostrada para os parâmetros de escala e de forma, pois como visto no Capítulo de simulação, não é seguro afirmar que esses estimadores atingem a normalidade assintótica para N $<$ 1000, que é o caso do conjunto de dados analisados. Os resultados mostram as variáveis estudo, experiência e experiência ao quadrado foram significativas para os dois grupos, enquanto a variável idade não contribuiu significativamente com o ajuste da variável resposta. Os valores dos coeficientes retratam que o estudo e a experiência ao quadrado interferem positivamente na quantidade de horas trabalhadas fora de casa por mulheres casadas. Como exemplo, o aumento de 1 ano de estudo para mulheres do Grupo 1 leva a um aumento de aproximadamente 30 horas de trabalho fora de casa.

Tabela 7 – Parâmetros estimados pelo melhor modelo (MFR-ST-CR com dois componentes e $\nu = 4$), erros-padrão aproximados e valor-p para os dados de Mroz (1984)

Parâmetro	**Estimativa**	**Erro-padrão**	**Valor-p**	**Significância**
β_{10}	-1,6798	0,2349	0,0000	***
β_{11} (estudo)	0,0306	0,0114	0,0038	**
β_{12} (idade)	0,0046	0,0041	0,1329	
β_{13} (experiência)	-0,0517	0,0106	0,0000	***
β_{14} (experiência^2)	0,0009	0,0003	0,0017	**
β_{20}	-0,2794	0,1787	0,0590	.
β_{21} (estudo)	0,0239	0,0089	0,0034	**
β_{22} (idade)	-0,0004	0,0024	0,4390	
β_{23} (experiência)	-0,0182	0,0087	0,0177	*
β_{24} (experiência^2)	0,0005	0,0003	0,0481	*
σ_1^2	0,0907	0,0260		
σ_2^2	0,3490	0,1093		
λ_1	-0,7410	0,6012		
λ_2	-7,8500	1,7325		
p_1	0,5341	0,0853		
ν	4			

9 Considerações finais

Concluindo, o presente trabalho apresentou um modelo capaz de ajustar dados com a variável resposta censurada na presença de assimetria, caudas pesadas e heterogeneidade não observada, estendendo o modelo proposto por Zeller, Cabral e Lachos (2016), o qual lidava com as mesmas características, porém, para dados sem censura, e o proposto por Galarza, Matos e Lachos (2022) para dados multivariados, o qual lidava com dados censurados e erros distribuídos por uma skew-normal, porém, sem considerar o contexto de misturas finitas e da classe de distribuições MESN. A utilização do Algoritmo EM foi crucial para a obtenção das estimadores de máxima-verossimilhança para os parâmetros e dos erros-padrão dos mesmos, que foram obtidos através da função Q, viabilizando a obtenção de fórmulas fechadas para as funções escore. Além disso, os estudos de simulação mostraram que esses estimadores são consistentes e eficientes, tanto no caso da distribuição skew-normal quanto no caso da skew-t, que foram as duas distribuições da classe MESN abordadas.

Como perspectivas futuras, podemos citar a extensão desse trabalho para outras distribuições da classe MESN, como a slash e a normal-contaminada, além de uma abordagem multivariada. Também é de interesse a otimização dos algoritmos computacionais, os quais podem demorar muito em casos de alto nível de censura, caudas muito pesadas e grandes amostras.

Referências

AZZALINI, A.; AZZALINI, M. A. Package 'sn'. *The skew-normal and skew-t distributions*, p. 1–3, 2015.

BASSO, R. M. et al. Robust mixture modeling based on scale mixtures of skew-normal distributions. *Computational Statistics & Data Analysis*, Elsevier, v. 54, n. 12, p. 2926–2941, 2010.

BOUGUILA, N.; FAN, W. *Mixture models and applications*. [S.l.]: Springer, 2020.

CABRAL, C. R. B.; LACHOS, V. H.; PRATES, M. O. Multivariate mixture modeling using skew-normal independent distributions. *Computational Statistics & Data Analysis*, Elsevier, v. 56, n. 1, p. 126–142, 2012.

CAUDILL, S. B. A partially adaptive estimator for the censored regression model based on a mixture of normal distributions. *Statistical Methods & Applications*, Springer, v. 21, n. 2, p. 121–137, 2012.

COLOSIMO, E. A.; GIOLO, S. R. *Análise de sobrevivência aplicada*. [S.l.]: Editora Blucher, 2006.

DÁVILA, V. H. L.; CABRAL, C. R. B.; ZELLER, C. B. *Finite mixture of skewed distributions*. [S.l.]: Springer, 2018.

DEMPSTER, A. P.; LAIRD, N. M.; RUBIN, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, Wiley Online Library, v. 39, n. 1, p. 1–22, 1977.

EVERITT, B. *Finite mixture distributions*. [S.l.]: Springer Science & Business Media, 2013.

FERNÁNDEZ, C.; STEEL, M. F. Multivariate student-t regression models: Pitfalls and inference. *Biometrika*, Oxford University Press, v. 86, n. 1, p. 153–167, 1999.

FRÜHWIRTH-SCHNATTER, S. *Finite mixture and Markov switching models*. [S.l.]: Springer, 2006. v. 425.

GALARZA, C. E. et al. Package 'momtrunc'. *R package version*, 2021.

GALARZA, C. E.; MATOS, L. A.; LACHOS, V. H. An em algorithm for estimating the parameters of the multivariate skew-normal distribution with censored responses. *METRON*, Springer, p. 1–23, 2022.

GARAY, A. W. M. *Modelos de regressao para dados censurados sob distribuiçoes simétricas*. Tese (Doutorado) — Universidade de Sao Paulo, 2014.

GILBERT, P.; VARADHAN, R.; GILBERT, M. P. Package 'numderiv'. *differential equations*, v. 3, p. 203–267, 2009.

HARTIGAN, J. A.; WONG, M. A. Algorithm as 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, JSTOR, v. 28, n. 1, p. 100–108, 1979.

KARLSSON, M.; LAITILA, T. Finite mixture modeling of censored regression models. *Statistical papers*, Springer, v. 55, n. 3, p. 627–642, 2014.

LIN, T.-I.; HO, H. J.; LEE, C.-R. Flexible mixture modelling using the multivariate skew-t-normal distribution. *Statistics and Computing*, Springer, v. 24, n. 4, p. 531–546, 2014.

LIU, C.; RUBIN, D. B. The ecme algorithm: a simple extension of em and ecm with faster monotone convergence. *Biometrika*, Oxford University Press, v. 81, n. 4, p. 633–648, 1994.

LOUZADA-NETO, F.; MAZUCHELI, J.; ACHCAR, J. A. *Uma introdução à análise de sobrevivência e confiabilidade*. [S.l.]: Sociedad Chilena de Estadística, 2001.

LUCAS, A. Robustness of the student t based m-estimator. *Communications in Statistics-Theory and Methods*, Taylor & Francis, v. 26, n. 5, p. 1165–1182, 1997.

MASSUIA, M. B. et al. Influence diagnostics for student-t censored linear regression models. *Statistics*, Taylor & Francis, v. 49, n. 5, p. 1074–1094, 2015.

MCLACHLAN, G. J.; BASFORD, K. E. *Mixture models: Inference and applications to clustering*. [S.l.]: M. Dekker New York, 1988. v. 38.

MCLACHLAN, G. J.; KRISHNAN, T. *The EM algorithm and extensions*. [S.l.]: John Wiley & Sons, 2007.

MCLACHLAN, G. J.; PEEL, D. *Finite mixture models*. [S.l.]: John Wiley & Sons, 2004.

MCNICHOLAS, P. D. *Mixture model-based classification*. [S.l.]: Chapman and Hall/CRC, 2016.

MENG, X.-L.; RUBIN, D. B. Maximum likelihood estimation via the ecm algorithm: A general framework. *Biometrika*, Oxford University Press, v. 80, n. 2, p. 267–278, 1993.

MENGERSEN, K. L.; ROBERT, C.; TITTERINGTON, M. *Mixtures: estimation and applications*. [S.l.]: John Wiley & Sons, 2011.

MROZ, T. A. *The sensitivity of an empirical model of married women's hours of work to economic and statistical assumptions*. [S.l.]: Stanford University, 1984.

PEARSON, K. Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London. A*, JSTOR, v. 185, p. 71–110, 1894.

PRATES, M. et al. Package 'mixsmsn'. 2021.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2022. Disponível em: <https://www.R-project.org/>.

ZELLER, C. B.; CABRAL, C. R.; LACHOS, V. H. Robust mixture regression modeling based on scale mixtures of skew-normal distributions. *Test*, Springer, v. 25, n. 2, p. 375–396, 2016.

ZELLER, C. B. et al. Finite mixture of regression models for censored data based on scale mixtures of normal distributions. *Advances in Data Analysis and Classification*, Springer, v. 13, n. 1, p. 89–116, 2019.

ZELLER, C. B.; LACHOS, V. H.; VILCA-LABRA, F. E. Local influence analysis for regression models with scale mixtures of skew-normal distributions. *Journal of Applied Statistics*, Taylor & Francis, v. 38, n. 2, p. 343–368, 2011.

A Algoritmo MFR-SN-CR

Algoritmo em R para aplicação do modelo MFR-SN-CR.

```
MFMRCSN <- function(y, X, cen, g, maxIter = 100, tol = 1e-6, ordem,
                    showEP = T, printOutput = T){
  ki <- cen
  n <- length(y)
  phi <- as.numeric(y == ki)

  p <- ncol(X)

  ## Função auxiliar

  dUnc <- function(yUnc, XUnc, beta, sigma, lambda, prop){
    auxMatriz <- matrix(NA, nrow = n-m, ncol = g)
    for(j in 1:g){
      mediaUnc <- as.numeric(XUnc%*%beta[j,])
      auxMatriz[,j] <- prop[j]*dsn(yUnc, mediaUnc, sigma[j], lambda[j])
    }
    return(rowSums(auxMatriz))
  }

  dCen <- function(yCen, XCen, beta, sigma, lambda, prop){
    auxMatriz <- matrix(NA, nrow = m, ncol = g)
    for(j in 1:g){
      mediaCen <- as.numeric(XCen%*%beta[j,])
      auxMatriz[,j] <- prop[j]*psn(yCen, mediaCen, sigma[j], lambda[j])
    }
    return(rowSums(auxMatriz))
  }

  logVero <- function(y, X, beta, sigma, lambda, prop){
    yUnc <- y[phi == 0]
    yCen <- y[phi == 1]

    XUnc <- X[phi == 0,]
    XCen <- X[phi == 1,]

    lv <- sum(log(dUnc(yUnc, XUnc, beta, sigma, lambda, prop))) +
      sum(log(dCen(yCen, XCen, beta, sigma, lambda, prop)))
```

```
  return(lv)
}

EP_MFMRCSN <- function(X, y, phi, beta, sigma2, lambda, prop){

  # Funções auxiliares

  dUncEP <- function(yUnc, XUnc, beta, sigma, lambda, prop){
    g <- nrow(beta)
    aux <- numeric(g)

    for(j in 1:g){
      mediaUnc <- as.numeric(t(XUnc)%*%beta[j,])
      aux[j] <- prop[j]*dsn(yUnc, mediaUnc, sigma[j], lambda[j])
    }
    return(sum(aux))
  }

  dCenEP <- function(yCen, XCen, beta, sigma, lambda, prop){
    g <- nrow(beta)
    aux <- numeric(g)

    for(j in 1:g){
      mediaCen <- as.numeric(t(XCen)%*%beta[j,])
      aux[j] <- prop[j]*psn(yCen, mediaCen, sigma[j], lambda[j])
    }
    return(sum(aux))
  }

  logVeroEP <- function(yi, xi, g, phi, theta){
    p <- length(xi)

    beta <- matrix(theta[(1:(p*g))], nrow = g, byrow = T)
    sigma <- sqrt(theta[(p*g + 1):(p*g + g)])
    lambda <- theta[(p*g + g + 1):(p*g + 2*g)]
    pAux <- theta[(p*g + 2*g + 1):(p*g + 3*g - 1)]

    prop <- c(pAux, 1-sum(pAux))

    if(phi == 1){
      lv <- log(dCenEP(yi, xi, beta, sigma, lambda, prop))
    }else{
      lv <- log(dUncEP(yi, xi, beta, sigma, lambda, prop))
```

```
  }

  return(lv)
}


# Encontrando o erro-padrão

n <- length(y)
g <- nrow(beta)
p <- ncol(X)

theta <- c(as.vector(t(beta)), sigma2, lambda, prop[-g])
i <- 1

while(i < n+1){
  score <- grad(logVeroEP, theta, yi = y[i], xi = X[i,], g = g, phi = phi[i])

  w <- score%*%t(score)

  if(i == 1){
    InfObs <- w
  }else{
    InfObs <- InfObs + w
  }

  i <- i + 1
}

EPGrad <- round(sqrt(diag(solve(InfObs))), 4)

names(EPGrad)[(1:(p*g))] <- paste0('beta', rep(1:g, rep(p, g)), rep(0:(p-1), g))
names(EPGrad)[(p*g + 1):(p*g + g)] <- paste0('sigma2_', 1:g)
names(EPGrad)[(p*g + g + 1):(p*g + 2*g)] <- paste0('lambda', 1:g)
names(EPGrad)[(p*g + 2*g + 1):(p*g + 3*g - 1)] <- paste0('p', 1:(g-1))


# Output

valCritico <- theta/EPGrad
valorP <- round(1 - pnorm(abs(valCritico)), 4)
names(valorP) <- names(EPGrad)
```

```
  output <- rbind(EPGrad, valorP)
  rownames(output) <- c('Erro-padrão', 'Valor-p')

  return(output)
}

printMFMRCSN <- function(theta, ep, lv, aic, bic){
  sig <- ifelse(ep[2,] > 0.05, ' ',
                ifelse(ep[2,] > 0.01, '*', ifelse(ep[2,] > 0.001, '**', '***')))
  tabela <- data.frame(Estimate = theta, `Std error` = ep[1,],
                       `p-value` = paste0(ep[2,], sig))

  cat('------------------------------------------------------------------------ \n')
  cat('  \n')
  print(tabela)
  cat('  \n')
  cat('Loglikelihood =', lv, '   AIC =', aic, '   BIC =', bic, '\n')
  cat('  \n')
  cat('  \n')
  cat('Ps: the results take into account the normality of the ML estimators for large n.  \n')
  cat('------------------------------------------------------------------------')
}


## Chute inicial

dados <- as.data.frame(cbind(Y = y, X[,-1]))
km <- kmeans(dados, centers = g, nstart = 50, iter.max = 100)
aux <- as.numeric(names(sort(table(km$cluster))))

betaNova <- matrix(NA, nrow = g, ncol = p)
dpNova <- pNova <- assNova <- numeric(g)

for(i in 1:g){
  dados0 <- dados[km$cluster == aux[ordem[i]],]
  modelo <- lm(Y ~ ., data = dados0)
  betaNova[i,] <- as.numeric(modelo$coefficients)
  dpNova[i] <- sqrt(sum((dados0$Y - modelo$fitted.values)^2)/(nrow(dados0) - p))
  assNova[i] <- skewness(modelo$residuals)
  pNova[i] <- nrow(dados0)/n
}

rm(dados, km, aux, ordem, dados0, modelo, i)
```

```
## Processo iterativo

yUnc <- y[phi == 0]
yCen <- y[phi == 1]

XUnc <- X[phi == 0,]
XCen <- X[phi == 1,]

m <- sum(phi == 1)

crit <- 1
c <- 0

while(crit > tol & c < maxIter){
  ## Etapa E

  betaAtual <- betaNova
  dpAtual <- dpNova
  assAtual <- assNova
  pAtual <- pNova

  Z <- ZY <- ZY2 <- ZT <- ZT2 <- ZTY <- matrix(NA, nrow = n, ncol = g)

  delta <- assAtual/sqrt(1 + assAtual^2)
  D <- dpAtual*delta
  G <- (dpAtual^2)*(1 - delta^2)

  for(j in 1:g){
    M <- sqrt(G[j]/(G[j] + D[j]^2))

    ### Dados sem censura

    mediaUnc <- as.numeric(XUnc%*%betaAtual[j,])
    aUnc <- assAtual[j]*(yUnc - mediaUnc)/dpAtual[j]
    aux <- ifelse(pnorm(aUnc) == 0, .Machine$double.xmin, pnorm(aUnc))

    #### E[Zi | Yi]

    ZUnc <- pAtual[j]*dsn(yUnc, mediaUnc, dpAtual[j], assAtual[j])/
      dUnc(yUnc, XUnc, betaAtual, dpAtual, assAtual, pAtual)
```

```
#### E[ZiYi | Yi]

YUnc <- yUnc

#### E[ZiYi² | Yi]

Y2Unc <- yUnc^2

#### E[ZiTi | Yi]

p01_T1 <- (M^2)*D[j]*(yUnc - mediaUnc)/G[j]
p02_T1 <- M*dnorm(aUnc)/aux

TUnc <- p01_T1 + p02_T1

#### E[ZiTi² | Yi]

T2Unc <- p01_T1^2 + p01_T1*p02_T1 + M^2

rm(p01_T1, p02_T1, mediaUnc, aUnc, aux)

#### E[ZiYiTi | Yi]

TYUnc <- YUnc*TUnc


### Dados censurados

mediaCen <- as.numeric(XCen%*%betaAtual[j,])

auxMean <- auxEYY <- w0 <- numeric(m)
for(i in 1:m){
  aux02 <- meanvarTMD(lower = -Inf, upper = yCen[i], mu = mediaCen[i],
                      Sigma = dpAtual[j]^2, lambda = assAtual[j], dist = 'SN')
  auxMean[i] <- aux02$mean
  auxEYY[i] <- aux02$EYY

  w0[i] <- meanvarTMD(lower = -Inf, upper = yCen[i], mu = mediaCen[i],
                      Sigma = G[j], dist = 'normal')$mean
}

P0 <- pnorm(yCen, mediaCen, sqrt(G[j]))
R0 <- ifelse(psn(yCen, mediaCen, dpAtual[j], assAtual[j]) == 0,
```

```
              .Machine$double.xmin,
              psn(yCen, mediaCen, dpAtual[j], assAtual[j]))
gi <- P0/(sqrt(pi*(1 + assAtual[j]^2)/2)*R0)

#### E[Zi | Yi < ki]

ZCen <- pAtual[j]*psn(yCen, mediaCen, dpAtual[j], assAtual[j])/
  dCen(yCen, XCen, betaAtual, dpAtual, assAtual, pAtual)

#### E[ZiYi | Yi < ki]

YCen <- auxMean

#### E[ZiYi² | Yi < ki]

Y2Cen <- auxEYY

#### E[ZiTi | Yi < ki]

TCen <- (M^2)*D[j]*(YCen - mediaCen)/G[j] + M*gi

#### E[ZiTi² | Yi < ki]

p01_T2 <- (M^4)*(D[j]^2)*(Y2Cen - 2*YCen*mediaCen + mediaCen^2)/(G[j]^2)
p02_T2 <- (M^3)*D[j]*(w0 - mediaCen)*gi/G[j] + M^2

T2Cen <- p01_T2 + p02_T2

rm(p01_T2, p02_T2)

#### E[ZiTiYi | Yi < ki]

TYCen <- (M^2)*D[j]*(Y2Cen - YCen*mediaCen)/G[j] + M*w0*gi


### Vetores finais

Z[phi == 0, j] <- ZUnc; Z[phi == 1, j] <- ZCen
ZY[phi == 0, j] <- ZUnc*YUnc; ZY[phi == 1, j] <- ZCen*YCen
ZY2[phi == 0, j] <- ZUnc*Y2Unc; ZY2[phi == 1, j] <- ZCen*Y2Cen
ZT[phi == 0, j] <- ZUnc*TUnc; ZT[phi == 1, j] <- ZCen*TCen
ZT2[phi == 0, j] <- ZUnc*T2Unc; ZT2[phi == 1, j] <- ZCen*T2Cen
ZTY[phi == 0, j] <- ZUnc*TYUnc; ZTY[phi == 1, j] <- ZCen*TYCen
```

```
    rm(ZUnc, YUnc, Y2Unc, TUnc, T2Unc, TYUnc, ZCen, YCen, Y2Cen, TCen,
       T2Cen, TYCen, auxEYY, auxMean, gi, i, j, P0, R0, w0, mediaCen, aux02, M)
  }


  ## Etapa M

  betaNova <- matrix(NA, nrow = g, ncol = p)
  DNova <- GNova <- pNova <- numeric(g)

  for(j in 1:g){
    pNova[j] <- sum(Z[,j])/n

    betaNova[j,] <- solve(t(X)%*%diag(Z[,j])%*%X)%*%(t(X)%*%(ZY[,j] - ZT[,j]*D[j]))

    DNova[j] <- sum(ZTY[,j] - ZT[,j]*(X%*%betaNova[j,]))/sum(ZT2[,j])

    GNova[j] <- sum(ZY2[,j] - 2*ZY[,j]*(X%*%betaNova[j,]) - 2*ZTY[,j]*DNova[j] +
                      Z[,j]*(X%*%betaNova[j,])^2 +
                      2*ZT[,j]*(X%*%betaNova[j,])*DNova[j] + ZT2[,j]*(DNova[j]^2))/
      sum(Z[,j])
  }

  dpNova <- sqrt(GNova + DNova^2)

  assNova <- DNova/sqrt(GNova)


  ## Critério de parada

  crit <- abs(logVero(y, X, betaNova, dpNova, assNova, pNova)/
                logVero(y, X, betaAtual, dpAtual, assAtual, pAtual) - 1)
  c <- c + 1
}


## Resultados

theta <- round(c(as.vector(t(betaNova)), dpNova^2, assNova, pNova[g-1]), 4)


### Desempenho
```

```
  nPar <- length(theta)
  lv <- logVero(y, X, betaNova, dpNova, assNova, pNova)
  aic <- -2*lv + 2*nPar
  bic <- -2*lv + log(n)*nPar

  desempenho <- c(lv, aic, bic)
  names(desempenho) <- c('Loglikelihood', 'AIC', 'BIC')


  ### Classficação das observações

  class <- apply(Z, 1, FUN = function(x) which.max(x))


  ### Erro-padrão

  if(showEP){
    erroPadrao <- EP_MFMRCSN(X, y, phi, betaNova, dpNova^2, assNova, pNova)

    resultado <- list(Iterations = c, Prop = pNova, Beta = betaNova,
                      Sigma2 = dpNova^2, Lambda = assNova,
                      StdError = erroPadrao, Performance = desempenho,
                      Classification = class)
  }else{
    resultado <- list(Iterations = c, Prop = pNova, Beta = betaNova,
                      Sigma2 = dpNova^2, Lambda = assNova,
                      Performance = desempenho, Classification = class)
  }


  ### Output

  if(showEP & printOutput){
    printMFMRCSN(theta, erroPadrao, lv, aic, bic)
  }


  return(resultado)
}
```

B Algoritmo MFR-ST-CR

Algoritmo em R para aplicação do modelo MFR-ST-CR.

```
MFMRCST03 <- function(y, X, cen, g, nu, maxIter = 100, tol = 1e-6, showEP = T,
                      printOutput = T, ordem){
  ki <- cen
  n <- length(y)
  phi <- as.numeric(y == ki)

  p <- ncol(X)


  ## Função auxiliar

  dUnc <- function(yUnc, XUnc, beta, sigma, lambda, nu, prop){
    auxMatriz <- matrix(NA, nrow = n-m, ncol = g)
    for(j in 1:g){
      mediaUnc <- as.numeric(XUnc%*%beta[j,])
      auxMatriz[,j] <- prop[j]*dst(yUnc, mediaUnc, sigma[j], lambda[j], nu)
    }
    return(rowSums(auxMatriz))
  }

  dCen <- function(yCen, XCen, beta, sigma, lambda, nu, prop){
    auxMatriz <- matrix(NA, nrow = m, ncol = g)
    for(j in 1:g){
      mediaCen <- as.numeric(XCen%*%beta[j,])
      auxMatriz[,j] <- prop[j]*pst(yCen, mediaCen, sigma[j], lambda[j], nu)
    }
    return(rowSums(auxMatriz))
  }

  logVero <- function(y, X, beta, sigma, lambda, nu, prop){
    yUnc <- y[phi == 0]
    yCen <- y[phi == 1]

    XUnc <- X[phi == 0,]
    XCen <- X[phi == 1,]

    lv <- sum(log(dUnc(yUnc, XUnc, beta, sigma, lambda, nu, prop))) +
      sum(log(dCen(yCen, XCen, beta, sigma, lambda, nu, prop)))
```

```
    return(lv)
  }

  EP_MFMRCST <- function(X, y, phi, beta, sigma2, lambda, prop, nu){

    # Funções auxiliares

    dUncEP <- function(yUnc, XUnc, beta, sigma, lambda, prop, nu){
      g <- nrow(beta)
      aux <- numeric(g)

      for(j in 1:g){
        mediaUnc <- as.numeric(t(XUnc)%*%beta[j,])
        aux[j] <- prop[j]*dst(yUnc, mediaUnc, sigma[j], lambda[j], nu)
      }
      return(sum(aux))
    }

    dCenEP <- function(yCen, XCen, beta, sigma, lambda, prop, nu){
      g <- nrow(beta)
      aux <- numeric(g)

      for(j in 1:g){
        mediaCen <- as.numeric(t(XCen)%*%beta[j,])
        aux[j] <- prop[j]*pst(yCen, mediaCen, sigma[j], lambda[j], nu)
      }
      return(sum(aux))
    }

    logVeroEP <- function(yi, xi, g, phi, nu, theta){
      p <- length(xi)

      beta <- matrix(theta[(1:(p*g))], nrow = g, byrow = T)
      sigma <- sqrt(theta[(p*g + 1):(p*g + g)])
      lambda <- theta[(p*g + g + 1):(p*g + 2*g)]
      pAux <- theta[(p*g + 2*g + 1):(p*g + 3*g - 1)]

      prop <- c(pAux, 1-sum(pAux))

      if(phi == 1){
        lv <- log(dCenEP(yi, xi, beta, sigma, lambda, prop, nu))
      }else{
```

```
    lv <- log(dUncEP(yi, xi, beta, sigma, lambda, prop, nu))
  }

  return(lv)
}


# Encontrando o erro-padrão

n <- length(y)
g <- nrow(beta)
p <- ncol(X)

theta <- c(as.vector(t(beta)), sigma2, lambda, prop[-g])
i <- 1

while(i < n+1){
  score <- grad(logVeroEP, theta, yi = y[i], xi = X[i,], g = g, phi = phi[i],
                nu = nu)

  w <- score%*%t(score)

  if(i == 1){
    InfObs <- w
  }else{
    InfObs <- InfObs + w
  }

  i <- i + 1
}

EPGrad <- round(sqrt(diag(solve(InfObs))), 4)

names(EPGrad)[(1:(p*g))] <- paste0('beta', rep(1:g, rep(p, g)), rep(0:(p-1), g))
names(EPGrad)[(p*g + 1):(p*g + g)] <- paste0('sigma2_', 1:g)
names(EPGrad)[(p*g + g + 1):(p*g + 2*g)] <- paste0('lambda', 1:g)
names(EPGrad)[(p*g + 2*g + 1):(p*g + 3*g - 1)] <- paste0('p', 1:(g-1))


# Output

valCritico <- theta/EPGrad
valorP <- round(1 - pnorm(abs(valCritico)), 4)
```

```
  names(valorP) <- names(EPGrad)

  output <- rbind(EPGrad, valorP)
  rownames(output) <- c('Erro-padrão', 'Valor-p')


  return(output)
}

printMFMRCST <- function(theta, nu, ep, lv, aic, bic){
  sig <- ifelse(ep[2,] > 0.05, ' ',
                ifelse(ep[2,] > 0.01, '*', ifelse(ep[2,] > 0.001, '**', '***')))
  tabela <- data.frame(Estimate = theta, `Std error` = ep[1,],
                       `p-value` = paste0(ep[2,], sig))

  cat('--------------------------------------------------------------------- \n')
  cat('  \n')
  print(tabela)
  cat('  \n')
  cat('nu =', nu, '\n')
  cat('  \n')
  cat('Loglikelihood =', lv, '   AIC =', aic, '   BIC =', bic, '\n')
  cat('  \n')
  cat('  \n')
  cat('Ps: the results take into account the normality of the ML estimators for large n. \n')
  cat('---------------------------------------------------------------------')
}


## Chute inicial

dados <- as.data.frame(cbind(Y = y, X[,-1]))
km <- kmeans(dados, centers = g, nstart = 50, iter.max = 100)
aux <- as.numeric(names(sort(table(km$cluster))))

betaNova <- matrix(NA, nrow = g, ncol = p)
dpNova <- pNova <- assNova <- numeric(g)

for(i in 1:g){
  dados0 <- dados[km$cluster == aux[ordem[i]],]
  modelo <- lm(Y ~ ., data = dados0)
  betaNova[i,] <- as.numeric(modelo$coefficients)
  dpNova[i] <- sqrt(sum((dados0$Y - modelo$fitted.values)^2)/(nrow(dados0) - p))
```

```
  assNova[i] <- skewness(modelo$residuals)
  pNova[i] <- nrow(dados0)/n
}

nuNova <- nu

rm(dados, km, aux, ordem, dados0, modelo, i, nu)


## Processo iterativo

yUnc <- y[phi == 0]
yCen <- y[phi == 1]

XUnc <- X[phi == 0,]
XCen <- X[phi == 1,]

m <- sum(phi == 1)

crit <- 1
c <- 0

while(crit > tol & c < maxIter){
  ## Etapa E

  betaAtual <- betaNova
  dpAtual <- dpNova
  assAtual <- assNova
  nuAtual <- nuNova
  pAtual <- pNova

  Z <- ZU <- ZUY <- ZUY2 <- ZUT <- ZUT2 <- ZUTY <- matrix(NA, nrow = n, ncol = g)

  delta <- assAtual/sqrt(1 + assAtual^2)
  D <- dpAtual*delta
  G <- (dpAtual^2)*(1 - delta^2)

  for(j in 1:g){
    M <- sqrt(G[j]/(G[j] + D[j]^2))

    ### Dados sem censura

    mediaUnc <- as.numeric(XUnc%*%betaAtual[j,])
```

```
aUnc <- assAtual[j]*(yUnc - mediaUnc)/dpAtual[j]
d2Unc <- ((yUnc - mediaUnc)^2)/(dpAtual[j]^2)

#### E[Zi | Yi]

ZUnc <- pAtual[j]*dst(yUnc, mediaUnc, dpAtual[j], assAtual[j], nuAtual)/
  dUnc(yUnc, XUnc, betaAtual, dpAtual, assAtual, nuAtual, pAtual)

#### E[Ui | Yi]

p01_U <- 4*(nuAtual^(nuAtual/2))*gamma((nuAtual + 3)/2)/
  (sqrt(pi)*gamma(nuAtual/2)*dpAtual[j])
p02_U <- (nuAtual + d2Unc)^(-(nuAtual + 3)/2)*
  pt(aUnc*sqrt((nuAtual + 3)/(nuAtual + d2Unc)), nuAtual + 3)
p03_U <- ifelse(dst(yUnc, mediaUnc, dpAtual[j], assAtual[j], nuAtual) == 0,
                .Machine$double.xmin,
                dst(yUnc, mediaUnc, dpAtual[j], assAtual[j], nuAtual))

UUnc <- p01_U*p02_U/p03_U

rm(p01_U, p02_U)

#### E[UiYi | Yi]

UYUnc <- UUnc*yUnc

#### E[UiYi² | Yi]

UY2Unc <- UUnc*(yUnc^2)

#### E[UiTi | Yi]

p01_UT <- (M^2)*D[j]*(yUnc - mediaUnc)*UUnc/G[j]
p02_UT <- 2*(nuAtual^(nuAtual/2))*gamma((nuAtual + 2)/2)/
  (pi*gamma(nuAtual/2)*dpAtual[j])
p03_UT <- (nuAtual + d2Unc + aUnc^2)^(-(nuAtual + 2)/2)

UTUnc <- p01_UT + M*p02_UT*p03_UT/p03_U

rm(p01_UT)

#### E[UiTi² | Yi]
```

```
p01_UT2 <- (((M^2)*D[j]*(yUnc - mediaUnc)/G[j])^2)*UUnc
p02_UT2 <- M^2 + (M^3)*D[j]*(yUnc - mediaUnc)*p02_UT*p03_UT/(p03_U*G[j])

UT2Unc <- p01_UT2 + p02_UT2

rm(p03_U, p02_UT, p03_UT, p01_UT2, p02_UT2)

#### E[UiYiTi | Yi]

UTYUnc <- yUnc*UTUnc

rm(d2Unc, aUnc)


### Dados censurados

mediaCen <- as.numeric(XCen%*%betaAtual[j,])

dpAtual02 <- sqrt((nuAtual*(dpAtual[j]^2))/(nuAtual + 2))

dpAtual03 <- sqrt((nuAtual*(dpAtual[j]^2))/((nuAtual + 1)*(1 + assAtual[j]^2)))

cNu <- 2*gamma((nuAtual + 1)/2)/
  (gamma(nuAtual/2)*sqrt(nuAtual*(1 + assAtual[j]^2)*pi))

aux <- ifelse(pst(yCen, mediaCen, dpAtual[j], assAtual[j], nuAtual) == 0,
              .Machine$double.xmin,
              pst(yCen, mediaCen, dpAtual[j], assAtual[j], nuAtual))

wPhi <- pst(yCen, mediaCen, dpAtual03, 0, nuAtual + 1)/aux

auxMean <- auxEYY <- w0 <- numeric(m)
for(i in 1:m){
  aux02 <- meanvarTMD(lower = -Inf, upper = yCen[i], mu = mediaCen[i],
                      Sigma = dpAtual02^2, lambda = assAtual[j],
                      nu = nuAtual + 2, dist = 'ST')
  auxMean[i] <- aux02$mean
  auxEYY[i] <- aux02$EYY

  w0[i] <- meanvarTMD(lower = -Inf, upper = yCen[i], mu = mediaCen[i],
                      Sigma = dpAtual03^2, lambda = 0, nu = nuAtual + 1,
                      dist = 'ST')$mean
}
```

```
#### E[Zi | Yi < ki]

ZCen <- pAtual[j]*pst(yCen, mediaCen, dpAtual[j], assAtual[j], nuAtual)/
  dCen(yCen, XCen, betaAtual, dpAtual, assAtual, nuAtual, pAtual)

#### E[Ui | Yi < ki]

UCen <- pst(yCen, mediaCen, dpAtual02, assAtual[j], nuAtual + 2)/aux

#### E[UiYi | Yi < ki]

UYCen <- auxMean*UCen

#### E[UiYi² | Yi < ki]

UY2Cen <- auxEYY*UCen

#### E[UiTi | Yi < ki]

p01_UT <- D[j]*(UYCen - UCen*mediaCen)/(G[j] + D[j]^2)
p02_UT <- M*cNu*wPhi

UTCen <- p01_UT + p02_UT

rm(p01_UT)

#### E[UiTi² | Yi < ki]

p01_UT2 <- ((D[j]/(G[j] + D[j]^2))^2)*
  (UY2Cen - 2*UYCen*mediaCen + UCen*(mediaCen^2))
p02_UT2 <- p02_UT*D[j]*(w0 - mediaCen)/(G[j] + D[j]^2) + M^2

UT2Cen <- p01_UT2 + p02_UT2

rm(p01_UT2, p02_UT2)

#### E[UiTiYi | Yi < ki]

p01_UTY <- D[j]*(UY2Cen - UYCen*mediaCen)/(G[j] + D[j]^2)
p02_UTY <- p02_UT*w0

UTYCen <- p01_UTY + p02_UTY
```

```
  rm(p02_UT, p01_UTY, p02_UTY, dpAtual02, dpAtual03, cNu, aux, wPhi, aux02,
     auxMean, auxEYY, w0)


  ### Vetores finais

  Z[phi == 0, j] <- ZUnc; Z[phi == 1, j] <- ZCen
  ZU[phi == 0, j] <- ZUnc*UUnc; ZU[phi == 1, j] <- ZCen*UCen
  ZUY[phi == 0, j] <- ZUnc*UYUnc; ZUY[phi == 1, j] <- ZCen*UYCen
  ZUY2[phi == 0, j] <- ZUnc*UY2Unc; ZUY2[phi == 1, j] <- ZCen*UY2Cen
  ZUT[phi == 0, j] <- ZUnc*UTUnc; ZUT[phi == 1, j] <- ZCen*UTCen
  ZUT2[phi == 0, j] <- ZUnc*UT2Unc; ZUT2[phi == 1, j] <- ZCen*UT2Cen
  ZUTY[phi == 0, j] <- ZUnc*UTYUnc; ZUTY[phi == 1, j] <- ZCen*UTYCen

  rm(ZUnc, UUnc, UYUnc, UY2Unc, UTUnc, UT2Unc, UTYUnc, ZCen, UCen, UYCen, UY2Cen,
     UTCen, UT2Cen, UTYCen, i, j, mediaCen)
}


## Etapa M

betaNova <- matrix(NA, nrow = g, ncol = p)
DNova <- GNova <- pNova <- numeric(g)

for(j in 1:g){
  pNova[j] <- sum(Z[,j])/n

  betaNova[j,] <- solve(t(X)%*%diag(ZU[,j])%*%X)%*%(t(X)%*%(ZUY[,j] - ZUT[,j]*D[j]))

  DNova[j] <- sum(ZUTY[,j] - ZUT[,j]*(X%*%betaNova[j,]))/sum(ZUT2[,j])

  GNova[j] <- sum(ZUY2[,j] - 2*ZUY[,j]*(X%*%betaNova[j,]) - 2*ZUTY[,j]*DNova[j] +
                    ZU[,j]*(X%*%betaNova[j,])^2 +
                    2*ZUT[,j]*(X%*%betaNova[j,])*DNova[j] + ZUT2[,j]*(DNova[j]^2))/
    sum(Z[,j])
}

dpNova <- sqrt(GNova + DNova^2)

assNova <- DNova/sqrt(GNova)

nuNova <- nuAtual
```

```
  ## Critério de parada

  crit <- abs(logVero(y, X, betaNova, dpNova, assNova, nuNova, pNova)/
              logVero(y, X, betaAtual, dpAtual, assAtual, nuAtual, pAtual) - 1)
  c <- c + 1
}

## Resultados

theta <- round(c(as.vector(t(betaNova)), dpNova^2, assNova, pNova[-g]), 4)


### Desempenho

nPar <- length(theta)
lv <- logVero(y, X, betaNova, dpNova, assNova, nuNova, pNova)
aic <- -2*lv + 2*nPar
bic <- -2*lv + log(n)*nPar

desempenho <- c(lv, aic, bic)
names(desempenho) <- c('Loglikelihood', 'AIC', 'BIC')


### Classficação das observações

class <- apply(Z, 1, FUN = function(x) which.max(x))


### Erro-padrão

if(showEP){
  erroPadrao <- EP_MFMRCST(X, y, phi, betaNova, dpNova^2, assNova, pNova,
                           nuNova)

  resultado <- list(Iterations = c, Prop = pNova, Beta = betaNova,
                    Sigma2 = dpNova^2, Lambda = assNova, Nu = nuNova,
                    StdError = erroPadrao, Performance = desempenho,
                    Classification = class)
}else{
  resultado <- list(Iterations = c, Prop = pNova, Beta = betaNova,
                    Sigma2 = dpNova^2, Lambda = assNova, Nu = nuNova,
```

```
                    Performance = desempenho, Classification = class)
  }


  ### Output

  if(showEP & printOutput){
    printMFMRCST(theta, nuNova, erroPadrao, lv, aic, bic)
  }


  return(resultado)
}
```