UNIVERSIDADE FEDERAL DE MINAS GERAIS
Instituto de Ciências Exatas
Programa de Pós-Graduação em Estatística

Patrícia Viana da Silva

Modelos de Mistura Normal/Independente via Processo Pontual por Determinante e seu uso para Redução de Dimensionalidade em Variáveis Categóricas

Belo Horizonte
2023

Patrícia Viana da Silva

Modelos de Mistura Normal/Independente via Processo Pontual por Determinante e seu uso para Redução de Dimensionalidade em Variáveis Categóricas

Versão Final

Tese apresentada ao Programa de Pós-Graduação em Estatística da Universidade Federal de Minas Gerais, como requisito à obtenção do título de Doutora em Estatística.

Orientadora: Profa. Dra. Rosangela Helena Loschi
Coorientador: Prof. Dr. Cristiano de Carvalho Santos

Belo Horizonte
2023

Silva, Patrícia Viana da.

S586m Modelos de mistura Normal/Independente via processo pontual por determinante e seu uso para redução de dimensionalidade em variáveis categóricas [recurso eletrônico] / Patrícia Viana da Silva – 2023.
1 recurso online (170 f. il., color.) : pdf.

Orientadora: Rosangela Helena Loschi
Coorientador: Cristiano de Carvalho Santos

Tese (Doutorado) - Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Estatística.
Referências: f.101-109

1. Estatística – Teses. 2. Análise de regressão – Teses. 3. Análise por conglomerados - Teses. 4. Markov, Processos de. – Teses. 5. Variáveis aleatórias – Teses. 6. Estatística Educacional – Teses. I. Loschi, Rosangela Helena. II. Santos, Cristiano de Carvalho. III. Universidade Federal de Minas Gerais, Instituto de Ciências Exatas, Departamento de Estatística. IV.Título.

CDU 519.2(043)

Ficha catalográfica elaborada pela bibliotecária Irénquer Vismeg Lucas Cruz
CRB 6/819 - Universidade Federal de Minas Gerais - ICEx

UNIVERSIDADE FEDERAL DE MINAS GERAIS

PROGRAMA DE PÓS-GRADUAÇÃO EM ESTATÍSTICA

FOLHA DE APROVAÇÃO

Modelos de mistura normal/independente via processo pontual por determinante e seu uso para redução de dimensionalidade em variáveis categoricas

PATRÍCIA VIANA DA SILVA

Tese submetida à Banca Examinadora, designada pelo Colegiado do Programa de Pós-Graduação em ESTATÍSTICA, como requisito para obtenção do grau de Doutor em ESTATÍSTICA, área de concentração ESTATÍSTICA E PROBABILIDADE.

Aprovada, em 07 de agosto de 2023, pela banca constituída pelos membros:

Profa. Rosangela Helena Loschi - Orientador
DEST/UFMG

Prof. Cristiano de Carvalho Santos
DEST/UFMG

Profa. Daiane Aparecida Zuanetti
DEST/UFSCar

Prof. Dani Gamerman
DEST/UFMG

Profa. Márcia D Elia Branco
IME-USP

Prof. Vinícius Diniz Mayrink
DEST/UFMG

Belo Horizonte, 7 de agosto de 2023.

Dedico esse trabalho a todas e todos que tentaram mais de uma vez. Que a perseverança seja sua companheira e inspiração.

Agradecimentos

Pela conclusão deste trabalho tenho muito a agradecer e a muitas pessoas.

Agradeço à professora **Rosangela Helena Loschi** e ao professor **Cristiano de Carvalho Santos** pela disponibilidade, paciência e dedicação conjuntas na orientação e desenvolvimento desse projeto mesmo durante o período desafiador da pandemia.

Aos meus pais **Antonio** e **Terezinha** a quem devo quase tudo o que sou. À minha querida filha **Bianca** que assistiu a mais aulas na universidade que alguns alunos de pós-graduação e ajudou a forjar a mulher e mãe que me tornei. Aos meus irmãos **Henrique** e **José Mário** a quem aprendi a admirar e sentir saudades. À **Elaine** e **Wemily** que se tornaram minha família de coração.

Aos meus grandes amigos **Jony** e **Victor** pelo apoio mesmo à distância. Aos amigos agregados durante o doutorado **Erick**, **Adriana** e **Isabela** pela identificação que virou amizade, e ao **Ricardo** que além da amizade me ofereceu longas sessões de discussões acadêmicas.

À **Faculdade de Matemática/UFU** e a própria **Universidade Federal de Uberlândia** pela oportunidade de crescimento e o tempo de licença para me dedicar a este projeto. À **Universidade Federal de Minas Gerais** pelo acolhimento, ensino de qualidade e aprendizado não apenas acadêmico, mas de vida.

Às minhas eternas professoras **Antonia**, pelo carinho e **Sissi** por me mostrar o caminho da estatística. Aos meus professores da graduação na **Universidade Federal do Ceará**: **Julio**, **Silvia** e **Mauricio** pelos grandes ensinamentos que formaram minha base. Às minhas orientadoras de mestrado na **Universidade de São Paulo**, professora **Denise Aparecida Botter** e professora **Mônica Carneiro Sandoval**. Ao professor **Carlos Alberto de Bragança Pereira** que incutiu em mim o gosto pela estatística Bayesiana e ao professor **José Galvão Leite** pelos ensinamentos e incentivo.

Às agências de fomento à pesquisa e à tecnologia: **CAPES** — Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, **CNPq** — Conselho Nacional de Desenvolvimento Científico e Tecnológico, e **FAPEMIG** — Fundação de Amparo à Pesquisa do Estado de Minas Gerais, pelas contribuições diretas e indiretas aos trabalhos desenvolvidos principalmente nas universidades públicas do país. Com esse apoio possibilitaram não só o desenvolvimento do meu mestrado e doutorado como da maioria dos pós-graduandos do país.

"*O aumento do conhecimento é como uma esfera dilatando-se no espaço: quanto maior a nossa compreensão, maior o nosso contato com o desconhecido.*"

(Blaise Pascal)

Resumo

Com o crescente volume de dados disponíveis atualmente, é fundamental ter métodos eficientes para organizar, explorar e extrair conhecimento desses dados. A análise de agrupamento é uma abordagem poderosa para lidar com essa grande quantidade de informações e identificar padrões úteis e ganhou mais notoriedade com a popularização do aprendizado de máquina. Este trabalho fornece um método flexível de agrupamento não supervisionado que incorpora característica de repulsão no comportamento dos parâmetros de locação que representam os grupos, também chamados de *clusters*. Foi desenvolvido um Modelo de Mistura Finita de Distribuições Normal/Independente considerando o comportamento dos parâmetros de locação como uma realização de um Processo Pontual por Determinante (PPD), distribuição de probabilidade que evita a criação de grupos redundantes através de sua característica natural de repulsão de pontos. A proposta estende um modelo conhecido na literatura fornecendo, ainda, estrutura de incerteza *a priori* aos parâmetros do PPD propondo uma abordagem para sua estimação. A proposta é apresentada para estimação de densidade com um estudo de simulação que exalta a capacidade do modelo em estimar corretamente o número de grupos e a alocação dos indivíduos nestes grupos, além de uma aplicação em dados de demanda de agronegócio no plantio de banana. O modelo também foi utilizado no contexto de regressão linear para lidar com grupos latentes e redução de dimensionalidade, especialmente para variáveis categóricas com muitos níveis. A proposta oferece uma alternativa ao uso de penalidades como as do tipo LASSO. Foram avaliados os efeitos da especificação dos parâmetros do modelo na redução de dimensionalidade, comparando-o com modelos existentes na literatura para análise de dados de educação. O modelo apresentou agrupamentos robustos e se mostrou parcimonioso na estimação do número de grupos em relação aos outros modelos com os quais foi comparado. Além disso, foi desenvolvido um algoritmo Markov Chain Monte Carlo (MCMC) completo para a estimação dos parâmetros do modelo, seguindo o paradigma Bayesiano, e é disponibilizada a implementação em R.

Palavras-chave: Análise de Agrupamento, Processo Pontual por Determinante, Modelo de Mistura Finita, Redução de Dimensionalidade, Distribuição Normal/Independente.

Abstract

With the increasing volume of data currently available, it is essential to have efficient methods to organize, explore and extract knowledge from this data. Cluster analysis is a powerful approach to dealing with this large amount of information and identifying beneficial patterns, which have gained more notoriety with the popularization of machine learning. This work provides a flexible unsupervised clustering method that incorporates the repulsion characteristic in the behavior of the location parameters, the clusters representing: a Finite Mixture Model of Normal/Independent Distributions developed considering the behavior of the location parameters as a realization of a Determinantal Point Process (DPP) probability distribution that avoids the creation of redundant groups through its natural characteristic of repulsion of points. The proposal extends a known model in the literature providing an uncertainty structure *a priori* to the PPD parameters, proposing an approach for its estimation. We introduce the proposal for density estimation with a simulation study that exalts the model's ability to correctly estimate the number of groups and the allocation of individuals in these groups, in addition to an application in agribusiness demand data in banana plantations. The model was also used in the linear regression context to deal with latent groups and dimensionality reduction, especially for categorical variables with many levels. The proposal offers an alternative to the use of LASSO-type penalties. The effects of specifying the model's parameters were evaluated on dimensionality reduction, comparing it with existing models in the literature for analyzing education data. The model allows robust grouping and proven parsimonious estimating of the number of groups compared to the other models. Furthermore, we developed a complete Markov Chain Monte Carlo (MCMC) algorithm to estimate the model parameters, following the Bayesian paradigm, and we available its implementation in R.

Keywords: Clustering, Determinant Point Process, Finite Mixture Model, Dimensionality Reduction, Normal/Independent Distribution.

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

Problemas práticos que envolvem classificação e agrupamento são muito comuns nas mais diversas áreas. Podemos citar aplicações em reconhecimento de imagem [71, 93], controle de pragas em lavouras [72, 38], mapeamento de doenças [95, 23, 107, 31], e monitoramento de incêndios em florestas [12, 112, 110]. A Análise de Agrupamento (do inglês *clustering*) é útil na visualização dos dados permitindo assim entender os padrões espaciais [50] além de reduzir o efeito de possíveis valores atípicos e de diferenças populacionais no processo de inferência, entre outras contribuições [57].

A Análise de Agrupamento é uma área de estudo que pretende classificar os indivíduos/ objetos em categorias as quais não são pré-determinadas. Existem várias técnicas para este fim, as quais são, em geral, baseadas no seguinte princípio: particionar um conjunto de N objetos em K grupos (*clusters*), de forma que haja homogeneidade dos objetos dentro dos grupos com respeito a uma ou mais características de interesse, enquanto diferentes grupos sejam marcados por uma heterogeneidade entre eles segundo a estas mesmas características. Outra forma de interpretação é que estes grupos representam sub-populações não observáveis, as quais são tratadas como uma característica latente dos objetos estudados [101].

Entre os métodos propostos para agrupamento, existem os métodos não-estocásticos baseados em heurísticas de particionamento ou agrupamento hierárquico. Estes métodos são, em geral, construídos prefixando-se o número de grupos e considerando algum critério de otimização, de tal forma que os *clusters* são obtidos quando o critério é atingido. Para citar uns poucos exemplos, em problemas de regionalização, o método AMOEBA (*A Multidirectional Optimum Ecotope-Based Algorithm*) proposto por [4] parte de uma área inicial e novas áreas vizinhas a esta são agregadas até que a autocorrelação espacial pare de crescer. O método SKATER (*Spatial 'K'luster Analysis by Tree Edge Removal*) proposto por [7] baseia-se em remover arestas de um grafo que representa o mapa de interesse até que a variância ou outra medida de dissimilaridade interna do cluster seja minimizada.

Dois métodos de agrupamento que se tornaram muito populares são o algoritmo k-médias e o algoritmo DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [36]. Ambos são considerados algoritmos não-supervisionados. O k-médias parte de um número prefixado de *clusters*, em geral, um valor grande, em seguida, encontrará os centroides dos grupos e, para induzir a redução de variabilidade, re-aloca cada objeto ao *cluster* cujo

centroide está mais próximo a ele motivando assim, uma nova *clusterização*. O processo é repetido até que um critério de otimalidade seja atingido. Já o DBSCAN define clusters como áreas com muita densidade de pontos separados por áreas de baixa densidade. Inicialmente, escolhe-se um ponto arbitrário, em seguida encontra-se todos os seus pontos vizinhos ao ponto escolhido em uma distância pré-fixada. Se o número de vizinhos for menor que o número mínimo pré-definido, este ponto é marcado como ruído e passa-se para o próximo ponto. Caso contrário, cria-se um novo cluster e adicione o ponto atual e seus pontos vizinhos ao cluster. Expande-se o cluster repetindo recursivamente esses passos.

A partir de métodos de agrupamento não-estocásticos pode-se obter uma alocação ou partição dos dados, no entanto, sem possibilidade de avaliação de erro. Assim, qualquer resultado é muito afetado por ambos, pelas escolhas iniciais do número de *clusters* e pela alocação inicial dos objetos aos *clusters*. Esses métodos também podem não conseguir lidar com dados ausentes levando a resultados tendenciosos ou incompletos ou apresentar capacidade reduzida de modelar distribuições complexas como aquelas com formas não lineares ou não convexas. Além disso, uma das grandes limitações destes métodos é a impossibilidade de atribuir-se uma medida de incerteza sobre o número e a posição dos clusters.

Para contornar estes problemas, os métodos probabilísticos são de grande valia, pois além de serem reprodutíveis, permitem realizar inferência estatística completa, incluindo a inferência sobre o número de *clusters* e a realização de teste de hipóteses, o que pode fornecer uma análise mais rigorosa dos dados. O Modelo de Mistura Finita de distribuições (MMF) [109, 115] e métodos Bayesianos não-paramétricos têm se mostrado métodos muito competitivos para a identificação de *clusters*. Em MMF, as estimativas são obtidas principalmente a partir de método de máxima verossimilhança ou considerando-se métodos de inferência Bayesiana [76].

O Modelo de Mistura Finita é um dos modelos estatísticos mais utilizados para inferência em populações heterogêneas e assume que o número de *clusters* é finito e igual a um número K, em geral, pré-fixado, de forma que

$$\mathbf{y}_i \sim \sum_{k=1}^{K} w_k G(\mathbf{y}_i|\boldsymbol{\theta}_k) \tag{1.1}$$

em que as características de interesse para cada indivíduo i denotada por $\mathbf{y}_i$, $i = 1, \ldots, N$, é descrita por uma distribuição de probabilidade diferente para cada subpopulação k tal que $\mathbf{y}_i|k \sim G(\mathbf{y}_i|\boldsymbol{\theta}_k)$. Em geral, $\boldsymbol{\theta}_k$ é um parâmetro de locação, e assume-se que o indivíduo i pertence a um grupo (*cluster*) k com uma probabilidade w_k, $k = 1, \ldots, K$. Isto induz um particionamento dos indivíduos em *clusters* e esta partição pode ser considerada uma característica latente no modelo. A inferência consiste em estimar os parâmetros dos componentes da mistura e os pesos, mas também pode ser de interesse recuperar a distribuição original de cada observação, o que é conhecido como classificação não supervisionada [109].

Este modelo tem sido amplamente utilizado, não apenas para a identificação de *cluster*, mas com diferentes propósitos. Por exemplo, devido a sua flexibilidade, o modelo de mistura

finita é muito aplicado à estimativa de densidade [27]. Em suas primeiras aparições forneceu estrutura para construir distribuições de probabilidade mais complexas, [81] e com heterogeneidade [86].

O apelo prático e vantajoso do algoritmo EM *(Estimation Maximization)* [30] contribuiu para popularizar ainda mais a modelagem baseada em misturas de distribuições. Isto é evidenciado pelas várias publicações relacionadas ao tema em diversas áreas [79, 106, 60, 68, 74, 69] inclusive amplas revisões [109, 75] e editoriais como [16]. Além disso, nas últimas quatro décadas, com o aumento da capacidade computacional disponível e de novos métodos como MCMC *(Monte Carlo Markov Chain)* surgiram trabalhos com maior apelo computacional [55, 56, 42] e considerando distribuições diferentes da normal para os componentes da mistura [67, 70].

O uso de MMF de distribuições normais mostrou-se inadequado quando, por exemplo, os dados apresentavam observações atípicas *(outliers)* ou oriundos de distribuições com caudas pesadas. Nestes casos, observou-se que as estimativas para a média e variância dos componentes da mistura eram seriamente afetadas [37, 76] indicando uma inadequação do modelo. Por ser mais sensível a dados com tais características, o ajuste do modelo de mistura de distribuições normais pode levar a uma superestimação do número de componentes da mistura numa tentativa de melhorar a aproximação da distribuição real dos dados. Isto, consequentemente, afeta a parcimônia da modelagem [94, 75].

A distribuição t-Student tem sido amplamente utilizada como uma alternativa à distribuição normal para modelar dados que apresentam observações atípicas. A distribuição t-Student compartilha com a distribuição normal a propriedade de simetria com relação ao parâmetro de locação. No entanto, ela é mais robusta e flexível por incluir um parâmetro de forma, chamado grau de liberdade, que regula o peso das caudas da distribuição, proporcionando melhores ajustes a dados que possuem valores extremos ou *outliers* [87]. Além de gerar modelos mais flexíveis, os MMF's possuem muitas vantagens tais como tratabilidade analítica simples, boas propriedades assintóticas e ótima capacidade de aproximação para qualquer função de densidade contínua [85, 109].

A distribuição t-Student pertence a uma subclasse de distribuições elípticas que tem sido muito utilizada na construção de modelos estatísticos mais flexíveis e robustos. Esta subclasse é denominada distribuições Normais/Independentes (NI) [65, 97, 5] a qual é construída a partir de uma mistura na escala da distribuição Normal da seguinte forma. Seja U uma variável aleatória não-negativa com distribuição $F_{U|\boldsymbol{\eta}}$. Um vetor aleatório $\mathbf{Y}$ possui distribuição pertencente à família de distribuições Normal/Independente(NI) se sua função densidade de probabilidade é

$$f_{Y|\boldsymbol{\mu};\boldsymbol{\Sigma};\boldsymbol{\eta}}(\mathbf{y}) = \int_0^\infty \frac{u^{n/2}}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{ -\frac{u}{2}(\mathbf{y}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\mathbf{y}-\boldsymbol{\mu})\right\} dF_{\mathbf{U}|\boldsymbol{\eta}}(u), \tag{1.2}$$

em que $\boldsymbol{\mu}$ é o parâmetro de locação, $\boldsymbol{\Sigma}$ é um parâmetro de escala e η é um parâmetro de forma. Diferentes distribuições para $\mathbf{Y}$ são obtidas considerando-se diferentes modelos para U. Se $U \sim Gama(\eta/2, \eta/2)$, (1.2) seguirá uma distribuição t-Student com prâmetro de locação $\boldsymbol{\mu}$, de

escala $\boldsymbol{\Sigma}$ e η graus de liberdade. Se U é uma variável aleatória degenerada tal que $P(U = 1) = 1$, (1.2) terá uma distribuição normal com média $\boldsymbol{\mu}$ e variância $\boldsymbol{\Sigma}$. Outras distribuições de cauda pesada, como, por exemplo, a slash e normal contaminada, podem ser obtidas assumindo-se outros modelos para U.

Se o vetor aleatório $\mathbf{Y}$ possui distribuição Normal/Independente n-variada dada em (1.2), pode-se utilizar a representação estocastica a seguir

$$\mathbf{Y} = \boldsymbol{\mu} + U^{-1/2}\boldsymbol{\Sigma}^{1/2}\mathbf{T}, \tag{1.3}$$

em que $\mathbf{T} \sim N_n(0, I_n)$ e U é uma variável aleatória positiva com f.d.p. $f_{U|\boldsymbol{\eta}}$ (Ver Apêndice A para mais detalhes). Esta representação estocástica permite a modelagem de dados com caudas pesadas de forma elegante e com aspectos práticos que facilitam a sua implementação computacional em modelos Bayesianos. Esta representação será utilizada neste trabalho.

Sob a perspectiva Bayesiana, para completar a especificação do modelo (1.1), atribui-se uma distribuição *a priori* ao vetor $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_K)$ e para os pesos $\mathbf{w} = (w_1, \ldots, w_K)$. Hieraquicamente, o modelo de mistura finita é assim representado:

$$\mathbf{y}_i|\mathbf{w}, \boldsymbol{\theta}, K \overset{ind.}{\sim} G(\cdot|\boldsymbol{\theta}_k); \tag{1.4}$$

$$\boldsymbol{\theta}_k|K \overset{ind.}{\sim} p(\cdot) \tag{1.5}$$

$$\mathbf{w}|K \sim \Pi(\cdot). \tag{1.6}$$

A distribuição de $\boldsymbol{\theta}_k \sim p(\boldsymbol{\theta}_k|\gamma)$, em que γ denota um hiperparâmetro, pertence a uma família de distribuições conhecidas e, usualmente, assume-se uma distribuição Dirichlet para os pesos. O número de componentes da mistura pode ser considerado infinito, isto é, $K = \infty$, o que é assumido em modelos Bayesianos não paramétricos.

Geralmente, $\boldsymbol{\theta}$ é considerado um parâmetro de locação e, em uma abordagem não-paramétrica, atribui-se a ele uma medida de probabilidade *a priori* modelada por um Processo Dirichlet $DP(\alpha, G_0)$ em que α é o parâmetro de dispersão e G_0 uma distribuição paramétrica basal conhecida [92]. O processo Dirichlet (PD) é o mais popular e tem sido amplamente utilizado com diferentes propósitos além da identificação de cluster [6, 40]. Hierarquicamente, o modelo de mistura via processo Dirichlet pode ser representado por

$$\mathbf{y}_i|\boldsymbol{\theta}_i \overset{ind.}{\sim} G(\cdot|\boldsymbol{\theta}_i) \text{ com}$$

$$\boldsymbol{\theta}_i \overset{ind.}{\sim} F,$$

em que F é uma medida de probabilidade, cuja incerteza é descrita por um Processo Dirichlet, isto é, $F \sim DP(\alpha, G_0)$, em que e G_0 é uma distribuição basal conhecida, α é o parâmetro de precisão que controla a variabilidade em torno de G_0 e o número de *clusters*. Se α é grande, então tem-se alta probabilidade de se particionar a população em um número grande de *clusters*.

Uma das críticas mais severas com respeito ao uso do DP para a estimação de *clusters* é que este processo tende a superestimar o número real de clusters. Isso ocorre porque seu uso pressupõe independência entre os parâmetros de locação. Por exemplo, se em (1.4) G é uma distribuição normal com média θ_i e variância σ^2 e assume-se um DP para modelar o comportamento de θ_i em que a normal é escolhida como a distribuição basal G_0, pressupõe-se independência entre os componentes θ_i's. Dessa forma, componentes podem ser considerados diferentes e, na verdade, serem muito similares e, portanto, redundantes. Isso pode acarretar inúmeros problemas em aplicações que demandam separação real entre os grupos como, por exemplo, ocorre quando se deseja agrupar indivíduos segundo suas características biológicas (ver [118, 117] para uma discussão detalhada).

Motivados por este tipo de problema, surgiu a necessidade de se construir modelos capazes de separar grupos que sejam realmente diferentes. Em Estatística Bayesiana não-paramétrica, este tipo de problema é abordado a partir da construção de distribuições de probabilidade com características repulsivas, ou seja, que atribuam probabilidades mais altas para agrupamentos bem separados, evitando redundâncias. Mudanças no modelo de mistura que induzem repulsão foram propostos recentemente na literatura a partir da penalização de componentes próximos. Modelos de mistura repulsivos foram propostos por [89], [118] e [13]. O modelo proposto por [89] utiliza uma convolução de um kernel contínuo com alguma medida de probabilidade aleatória discreta definida como uma mistura infinita de átomos. Uma penalização imposta por uma medida de Gibbs controlando o nível de repulsão com um parâmetro foi proposta por [90]. Além desses, [88] também definiu outra distribuição a partir de uma função penalidade. No entanto, estas distribuições *a priori* dependem da definição explícita de uma métrica de penalização e os cálculos das distribuições *a posteriori* resultantes podem ser bastante complexos.

Uma maneira alternativa de induzir repulsão é considerada no modelo de Processos Pontuais por Determinante - PPP *(Determinantal Point Process)* que foi descrito em 1975 [73] para modelar o comportamento dos férmions, partículas quânticas que obedecem ao princípio de exclusão de Pauli, ou seja, dois férmions no mesmo nível de energia não podem ocupar uma mesma posição do espaço no mesmo instante. Então por definição os PPD's já incorporam a característica de repulsão.

Antes de [73], esse processo foi apenas citado de forma implícita em trabalhos de teoria de matrizes aleatórias [34]. O caso discreto foi primeiro discutido em exercícios do livro [28] e um estudo mais geral com caso discreto e contínuo foi publicado por [102, 103]. Os PPD's vem sendo estudados em várias áreas como probabilidade [54], teoria dos números [98], física estatística [84] e, nos últimos anos, para seleção de itens em aprendizado de máquina [63]. O caso de espaço de estados discretos foi, primeiramente, discutido em exercícios do livro [28] e um estudo mais geral com caso discreto e contínuo foi publicado por [102, 103]. Alguns fenômenos naturais, tais como a localização de árvores e formigas em uma floresta, que também apresentam padrões repulsivos, foram analisados utilizando o PPD por [63]. Entre os exemplos

teóricos de processos pontuais por determinante estão os passeios aleatórios sem interseção[58], os valores próprios de matrizes aleatórias [48, 77], os ladrilhos de diamante asteca [59], entre outros.

Entre as características relevantes do PPD pode-se citar condições de existência, momentos conhecidos, densidade com forma analítica fechada num conjunto compacto em relação ao Processo de Poisson e facilidade de uso em modelos paramétricos [66].

O trabalho de [3] foi pioneiro em propor o PPD como uma alternativa ao DP em modelos de mistura para fazer-se análise de agrupamento. Em seu modelo, esses autores consideram que os dados são modelados por uma mistura de distribuições normais com diferentes médias e variâncias e utilizam o PPD para modelar a incerteza sobre as médias destas distribuições normais. Se no modelo em (1.1) G é uma distribuição normal com média θ_k e variância σ_k^2, [118] mostraram que o uso do Processo Pontual por Determinante como uma distribuição *a priori* para $\boldsymbol{\theta}_k$ contribui para uma melhor estimação das medidas de locação do modelo de mistura e evita criação de grupos redundantes. Isto acontece porque amostras do PPD são conjuntos de pontos que naturalmente se espalham pelo espaço de estados [66]. Consequentemente, este comportamento leva a uma melhor estimação do número de *clusters* evitando a sua superestimação se comparado ao que estimado utilizando-se o Processo Dirichlet e fornece bons resultados e na presença de covariáveis que influenciam a variável resposta [13].

Mesmo com tantas propriedades interessantes, a estimação dos parâmetros do PPD ainda é pouco discutida. Essa tarefa é bastante desafiadora, uma vez que a função de probabilidade não é convexa e o cálculo da função de verossimilhança e seu gradiente podem ser inviáveis em muitos cenários [1]. Mesmo no caso de espaço de estados discreto, em que a distribuição de probabilidade possui fórmula analítica conhecida, a estimação dos parâmetros pode ser considerada um problema NP difícil devido à alta dimensionalidade e capacidade computacional exigida [63].

[66] fez uma primeira tentativa de estimar os parâmetros da função de similaridade do kernel utilizando a otimização Nelder-Mead. Embora seja uma alternativa razoável, esse método não garante convergência para um ponto estacionário e, assim como em outros métodos de otimização, depende do cálculo exato da probabilidade, o que nem sempre é possível para esse processo. [1] conduziu um estudo mais amplo de métodos Bayesianos em modelos envolvendo um kernel em que a função de similaridade depende de um parâmetro e a função qualidade depende de dois parâmetros. Neste estudo, [1] propõe uma alternativa para limitar a verossimilhança e assim contornar a não-convexidade da função. A limitação também permitiu utilizar o algoritmo " *slice-sampler*". Sua ideia consiste em usar um ponto de corte $u \sim Unif(0,1)$ pré-computado e compará-lo com os limitantes. Se u está fora do intervalo limitado, uma decisão é tomada se está entre o limitante superior e o inferior, o algoritmo recalcula os limitantes tentando melhorar a acurácia. Contudo, o espaço paramétrico original não é limitado e não há garantias de que a convergência alcançada seja real [66]. [47] explora a estimação dos parâmetros do kernel de um PPD discreto utilizando o algoritmo EM num contexto não-

paramétrico. Os autores mostram que seus resultados convergem mais rapidamente e são mais robustos que a maximização da verossimilhança via gradiente projetado e ilustram o seu uso em aplicações de dados de recomendação de produtos. [66] propõe contornar o problema da estimação dos parâmetros, a partir de uma aproximação para a função densidade do PPD com base em transformada de Fourier.

Mais recentemente, [20] realizou uma extensa investigação sobre as propriedades geométricas da função de verossimilhança que esclarece problemas enfrentados na estimação de parâmetros do kernel para o PPD discreto. Este estudo chamou a atenção para uma possível "maldição de dimensionalidade" uma vez que a variância assintótica do estimador de máxima verossimilhança aumenta exponencialmente com a dimensão do problema. Além disso, mostra que a função de verossimilhança pode exibir um número exponencial de pontos de sela e evidências de que esses podem ser os únicos pontos críticos.

Em geral, as pesquisas e mesmo os pacotes estatísticos utilizam uma função qualidade constante [8] em que o parâmetro também representa a intensidade do processo. Sob essa configuração, [15] mostram que um estimador pontual baseado na mediana é mais robusto na presença de *outliers* se comparado ao estimador padrão que, em processos pontuais por determinante estacionários, é a quantidade de pontos observados por unidade de volume. Apesar do PPD ser muito utilizado em modelos de mistura, não houve uma investigação do impacto da estimação dos seus parâmetros na estimação dos outros parâmetros do modelo. Além disso, o uso de uma função qualidade constante dificulta o uso do sentido original definido por [64] como medida de preferência dos valores gerados pelo processo. Esse sentido condiz com o paradigma Bayesiano e será explorado nesse trabalho com uma função qualidade baseada na distribuição Gaussiana como sugerido em [118].

Se as observações dentro dos *clusters* tiverem distribuições com caudas pesadas, o modelo de mistura de normais via PD pode levar a uma superestimação do número de clusters. Um dos objetivos deste trabalho é desenvolver um Modelo de Mistura Finita de Distribuições Normal/Independente em que a incerteza sobre os parâmetros de locação dos componentes da mistura serão modeladas pelo Processo Pontual por Determinante. Será considerado o PPD que depende da função kernel C, uma função de covariância definida por dois hiperparâmetros como será detalhado no Capítulo 2. O modelo deste trabalho estende o modelo proposto por [118] também por atribuir uma estrutura de dependência e distribuições *a priori* a estes parâmetros. Além disto, visa dar as seguintes contribuições:

- apresentar um estudo do kernel Gaussiano utilizado Processo Pontual por Determinante, no contexto do modelo de mistura, a fim de melhorar a compreensão do seu comportamento como distribuição *a priori* dos parâmetros de locação;

- fornecer um modelo de mistura com mais flexibilidade para o ajuste dos dados e estimação de densidade a partir de uma distribuição com caudas pesadas baseado na família de

distribuições Normal/Independente, obtendo um método de inferência mais robusto que o Modelo de Mistura de Normais;

- investigar a influência das distribuições *a priori* atribuídas aos parâmetros do PPD na inferência *a posteriori* do modelo proposto;
- desenvolver um algoritmo MCMC (*Monte Carlo Markov Chain*) completo para estimação dos parâmetros do modelo sob o paradigma Bayesiano e fornecer sua implementação no programa R [91].

Os modelos de mistura finita também têm sido amplamente utilizados em modelos de regressão linear em que as variáveis são originárias de grupos latentes desconhecidos. Nestes estudos, misturas finitas de distribuições são utilizadas para modelar os erros em modelos de regressão [10, 105]. Outro problema, no contexto de modelos de regressão linear, em que os métodos de agrupamento podem ser úteis é na redução de dimensionalidade. Este problema é particularmente relevante quando o modelo envolve variáveis categóricas com muitos níveis. Este tipo de problema pode surgir em situações de classificação de profissões como a Classificação Internacional de Ocupações *(HISCO: Historical International Standard Classification of Occupations)*; em classificação de indivíduos segundo a diversidade étnica e cultural, quando houverem muitos níveis diferentes de nacionalidade; em estudos sobre preferências por tipos de alimento em pesquisas alimentares; ou em problemas nos quais a variável resposta é influenciada pela localização geográfica (países, estados, municípios, códigos postais, etc) do indivíduo pesquisado.

Usualmente, este tipo de variável é tratada por meio de variáveis *dummy* e, quando esta têm muitos níveis, a estimação dos coeficientes pode se tornar instável, tornando os resultados difíceis de serem interpretados. Este problema tem sido tratado utilizando técnicas de regularização as quais adicionam um termo de penalização na soma dos quadrados dos resíduos a partir da qual os efeitos são estimados. Estes métodos visam "encolher" para zero coeficientes que são não-significativos. Este princípio é a base do método LASSO introduzido por [108].

Do ponto de vista Bayesiano, este problema tem sido tratado construindo-se uma distribuição *a priori* de encolhimento para os efeitos das covariáveis as quais têm papéis similares àqueles das funções de penalização na estimação dos parâmetros. Para o tratamento de variáveis categóricas com muitos níveis, o método tipo LASSO proposto por [46] e o *Effect Fusion Prior*, um método de regularização alternativo introduzido por [85], mostram-se bastante eficientes para a redução de dimensionalidade. [46] propõem métodos de encolhimento de efeitos para variáveis preditoras categóricas usando dois tipos de penalidade $L1$, uma para preditores nominais e a outra para preditores em escala ordinal, que selecionam fatores e agrupam categorias. Recentemente, [26] propuseram um modelo de agrupamento para redução de dimensionalidade de variáveis categóricas. O modelo proposto por estes autores modela o efeito das variáveis categóricas usando um modelo partição produto [51] e quando comparado com modelos anteriormente propostos mostrou-se muito eficiente. [26] utilizam árvores geradoras aleatórias em

modelos de partição produto para agrupar efeitos de categorias semelhantes considerando uma estrutura de vizinhança entre as diferentes categorias.

Inspirado por estes trabalhos, outra meta deste projeto é propor um modelo para redução de variáveis categóricas com muitos níveis usando o PPD, contribuindo nas seguintes direções:

- Introdução de um modelo de regressão via mistura de distribuições Normal/Independente para tratar variáveis categóricas com muitos níveis, será assumido que o efeito desta covariável é modelado por um PPD;
- Avaliação dos efeitos da especificação de parâmetros deste modelo na redução de dimensionalidade;
- Comparação do modelo proposto a modelos da literatura na análise de dados de educação;
- Desenvolvimento de um algoritmo MCMC (*Monte Carlo Markov Chain*) completo para estimação dos parâmetros do modelo sob o paradigma Bayesiano e sua implementação no programa [91].

O trabalho está organizado da seguinte forma. No Capítulo 2, apresenta-se o PPD como distribuição *a priori* para o parâmetro de locação no modelo de mistura de Normais como definido na literatura. Apresenta-se também um estudo do comportamento do kernel Gaussiano neste contexto. No Capítulo 3, é desenvolvido o modelo de mistura Normal/Independente via PPD para estimação de densidade estendendo o modelo de mistura proposto por [118]. Também é apresentada a proposta para modelar a incerteza sobre os parâmetros do modelo PPD. Para ilustrar o uso do modelo proposto, são apresentados um estudo envolvendo dados simulados e uma aplicação a dados de imagem de satélite para avaliar o comportamento de biomassa no plantio de banana. No Capítulo 4, o modelo de regressão linear que considera uma mistura de distribuições Normal/Independente via PPD para redução de dimensão de variável categórica com muitos níveis é introduzido. Este modelo é aplicado a dados educacionais visando identificar os cursos com mesmo efeito no rendimento acadêmico do aluno. É realizado, também, um estudo de sensibilidade visando avaliar o efeito de modificações na estimação dos parâmetros. O modelo proposto é comparado a modelos previamente introduzidos na literatura. O Apêndice A, contém propriedades e aspectos matemáticos que auxiliaram no trabalho, demonstrações que permitiram obter a fundamentação do modelo, como as distribuições condicionais completas. Além disso, no Apêndice B, são fornecidas as rotinas computacionais utilizadas para obtenção das estimativas desenvolvidas no programa R [91] com linguagem C^{++} a partir do pacote Rcpp [35].

Capítulo 2

Estudo do Kernel do PPD

Nesse capítulo será discutido o Processo Pontual por Determinante (PPD) como distribuição *a priori* para o parâmetro de locação do Modelo de Mistura Finita (MMF) de Normais [118]. A distribuição do PPD depende do determinante de matrizes tendo como base uma função kernel. Nesse caso, será utilizado a função kernel exponencial quadrática definida por duas funções: função de similaridade e função qualidade, e seu comportamento é ilustrado graficamente. Diferentemente do que é assumido em [118], uma estrutura de dependência é proposta para os parâmetros do kernel.

2.1 Processo Pontual por Determinante no contexto de Modelo de Mistura

O processo Dirichlet (PD) é um dos modelos mais usados como distribuição *a priori* para os parâmetros de locação, $\boldsymbol{\theta}$, no Modelo de Mistura Finita (MMF) dado em (1.1). No entanto, o PD induz independência sobre os componentes de $\boldsymbol{\theta}$, podendo causar uma redundância no vetor de parâmetros de locação e, consequentemente, uma superestimação do número de grupos. Isso ocorre porque podem ser estimados θ_k e $\theta_{k'}$ com valores muito próximos e, neste sentido, serão semelhantes, mas são considerados pelo modelo como dois componentes diferentes da mistura. A alocação dos indivíduos aos componentes relacionados a θ_k e $\theta_{k'}$ será comprometida, podendo haver sobreposição das regiões que os dois componentes ocupam no espaço das observações. O Processo Pontual por Determinante (PPD) é um processo repulsivo que passou a ser utilizado como alternativa ao PD visando produzir modelos com uma melhor estimativa para o número de componentes da mistura.

Seja $\mathbf{y}_i = (\mathbf{y}_{i1}, \ldots, \mathbf{y}_{iD})$ um vetor aleatório D-dimensional representando as caracteristicas de interesse (resposta) para o indivíduo i e seja a variável latente z_i em que $z_i = k$ indica que o indivíduo i pertence ao componente k, $k = 1, \ldots, K$, $i = 1, \ldots, N$. No modelo em (1.1), assuma que, dado $z_i = k$, $\mathbf{y}_i$ tem uma distribuição normal D-variada com média $\boldsymbol{\mu}_k$ e variância

$\boldsymbol{\Sigma}_k$ para todos os K componentes da mistura. Assuma que os parâmetros de locação $\boldsymbol{\mu}_k$ seguem um PPD. Este MMF pode ser representado considerando a seguinte estrutura hierárquica:

$$\begin{aligned} \mathbf{y}_i|z_i = k &\sim N_D(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \\ p(z_i = k) &= w_k, \\ \boldsymbol{\mu} = (\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K) &\sim PPD(C). \end{aligned} \tag{2.1}$$

A fim de simplificar o modelo, a Decomposição de Valores Singulares (DVS) foi aplicada à matriz de covariância resultando em $\boldsymbol{\Sigma}_k = E^\top \Lambda_k E$, em que E é uma matriz ortogonal constante e $\Lambda_k = diag\{\lambda_{k1}, \ldots, \lambda_{kD}\}$, em que D é a dimensão dos dados. A partir desta estrutura, para completar a especificação do modelo considere as seguintes distribuições a priori

$$\begin{aligned} \mathbf{w} = (w_1, \ldots, w_K)|K, \boldsymbol{\delta} &\sim Dir(\delta_1, \ldots, \delta_K), \delta_k > 0, k = 1, \ldots, K; \\ \lambda_{kd}|K, a_0, b_0 &\overset{\text{iid}}{\sim} IG(a_0, b_0),\ d = 1, \ldots, D, k = 1, \ldots, \ldots, K. \end{aligned} \tag{2.2}$$

Nesse caso, *IG* representa a distribuição Inversa-Gama. Definições necessárias para entender o PPD são dadas a seguir e uma discussão sobre as distribuições *a priori* relacionadas serão dadas na Subseção 2.5.

Um processo pontual é um modelo estocástico que descreve o comportamento de padrões de pontos em uma determinada região. Nesse caso, tanto o número de pontos quanto a localização específica dos pontos são variáveis aleatórias [32]. Uma realização desse processo, é uma coleção de pontos, dentro da região. Dessa forma, um PPD pode ser descrito como uma distribuição de probabilidade definida em subconjuntos de um conjunto base fixo, Ω [73]. Além de ser um processo pontual regular, sua principal característica é que os subconjuntos com maior probabilidade possuem diversidade, ou seja, contêm observações menos semelhantes em algum sentido [64]. Suas probabilidades são resultados de determinantes de uma matriz kernel calculada para configurações de pontos, $\{x_1, x_2, \ldots, x_K\}$, tais que $x_i \in \Omega$, $K > 0$, inteiro.

Para o modelo em (2.1) é utilizado PPD contínuo definido para um conjunto de Borel qualquer, $B \subseteq \mathbb{R}^D$, considere $\mathcal{X}$ um processo pontual espacial simples e localmente finito em $\mathbb{R}^D$, ou seja, suas realizações podem ser vistas como subconjuntos localmente finitos de B. Antes de outras formalizações para o PPD é necessário estabelecer o conceito do kernel necessário para definir o processo.

Definição 2.1.1. *Função e matriz kernel*

A função complexa C é dita uma função kernel se é definida como $C : B \times B \to \mathbb{C}$. E a matriz $C_{\mathbf{X}}$, de $\mathbf{X} = (x_1, \ldots, x_n)$ é uma matriz kernel, $[C](x_1, \ldots, x_n)$, com dimensão $n \times n$ se sua (i, j)-ésima entrada é calculada por uma função kernel aplicada aos pontos x_i e x_j, ou seja, $C(x_i, x_j)$, com x_i e $x_j \in B$.

A função kernel C está definida no plano complexo, $\mathbb{C}$, no entanto, para a maioria das aplicações, C é uma função real. Neste caso, se C for simétrica, isto é, $C(x_i, x_j) = C(x_j, x_i), \forall\ i, j,$

C é considerada uma função de covariância. No caso complexo, C será uma função de covariância apenas se for Hermitiana. Considere-se a partir deste ponto do trabalho o kernel como função de covariância real.

Diferente das variáveis aleatórias comuns, para as quais os momentos são valores (geralmente) reais, os processos pontuais possuem momentos que são, na verdade, medidas σ-aditivas, sendo representados como funções de borelianos, $B \subset \mathbb{R}^D$ [22]. A seguir, são dadas algumas definições desses momentos que descrevem as características das distribuições espaciais de pontos em um espaço contínuo. Considere para uma matriz quadrada complexa A e denote por $\det(A)$ o seu determinante.

Definição 2.1.2. *Função densidade produto*

O processo $\mathcal{X}$ é um processo pontual por determinante com kernel C, denotado por $\mathcal{X} \sim PPD(C)$, se suas funções de densidade produto satisfazem

$$\rho^{(n)}(x_1, \ldots, x_n) = \det[C(x_1, \ldots, x_n)]; n = 1, 2 \ldots . \tag{2.3}$$

Nesse caso, há a restrição de que $\rho^{(n)}(x_1, \ldots, x_n) \to 0$ se $x_i \to x_j$ para algum $i \neq j$ e C deve ser não negativa definida para que $\rho^{(n)} \geq 0$ [66]

A função densidade produto de ordem n de $\mathcal{X}$ representa a probabilidade de que, para cada $i = 1, \ldots, n$, $\mathcal{X}$ tenha um ponto na região em torno de x_i de volume dx_i. .

Definição 2.1.3. *Função intensidade*

Em particular, se $n = 1$ na Definição 2.1.2, então $\rho = \rho^{(1)}$ é conhecida como função intensidade ou função densidade produto de ordem 1:

$$\rho(\mathbf{x}) = C(\mathbf{x}, \mathbf{x}), \mathbf{x} \in \mathbb{R}^D. \tag{2.4}$$

Pela Definição da Função Intensidade (2.4), se $n = 2$, a função kernel será dada por

$$C(x_i, x_j) = \begin{bmatrix} \rho(x_i) & C(x_i; x_j) \\ C(x_j; x_i) & \rho(x_j) \end{bmatrix}.$$

Além disso, $C(\cdot)$ é simétrica, portanto, $\rho^{(2)}(x_i, x_j)$ que é o determinante da matriz $C(x_i, x_j)$ será dado por

$$\rho^{(2)}(x_i, x_j) = \rho(x_i)\rho(x_j) - C(x_i; x_j)^2. \tag{2.5}$$

Os elementos de C fora da diagonal principal determinam correlação negativa entre os pares de elementos (x_i, x_j) com $i \neq j$, pois reduzem o valor final do determinante. Valores grandes de $C(x_i, x_j)$ implicam que os pontos x_i e x_j tendem a não ocorrer juntos [63].

Definição 2.1.4. *Função de correlação par*

Sejam x_i e $x_j \in B$ a função de correlação par é dada por

$$g(x_i, x_j) = \frac{\rho^{(2)}(x_i, x_j)}{\rho(x_i)\rho(x_j)},$$

em que $g(x, y) = 0$ *se* $\rho(x) = 0$*, se* $\rho(y) = 0$*, ou se* $\rho^{(n)}(x_i, \ldots, x_n) = 0$*.*

De (2.5), tem-se que

$$g(x_i, x_j) = 1 - \frac{C(x_i, x_j)^2}{\rho(x_i)\rho(x_j)},$$

em que $g(x_i, x_j) = 1$ se $C(x_i, x_j) = 0$, ou seja, a correlação par é maior quando a função kernel é *zero* para aquele par de pontos.

A repulsividade do PPD é refletida pelo seguinte resultado. Se C for Hermitiana (caracteristica que generaliza as propriedades de uma função de covariância real para vetores no conjunto dos números complexos), então $g \leq 1$ e, para qualquer $n = 2, 3, \ldots$, resultará

$$\rho^{(n)}(x_1, \ldots, x_n) \leq \rho(x_1) \ldots \rho(x_n)$$

geralmente $\leq$ pode ser substituído por $<$. O determinante de uma matriz de covariância complexa é menor ou igual ao produto de seus elementos diagonais. Se C for contínua, $\rho^{(n)}$ também é, e $\rho^{(n)}(x_1, \ldots, x_n) \to 0$ a medida que a distância Euclidiana entre x_i e x_j se aproxima de 0 para algum $i \neq j$, isso é determinado pela Definição 2.1.2.

O Processo Pontual por Determinante têm vários resultados que enfatizam suas vantagens tais como: estabilidade por restrição, por complementação e por condicionamento, ou seja, sob estas operações o resultado continua sendo um PPD, além de unicidade, momentos finitos, entre outras. Para mais detalhes ver [66].

Neste trabalho será utilizado o kernel exponencial quadrático, um dos poucos que apresentam forma analítica disponíveis para seus autovalores. Além disso, será considerada a decomposição proposta por [64]. Esta proposta constroi a função kernel a partir de uma função similaridade, $\varphi(x_i, x_j)$ e de uma função qualidade, $q(x_i)$ na forma

$$C(x_i, x_j) = q(x_i)\varphi(x_i, x_j)q(x_j). \tag{2.6}$$

em que a função qualidade, $q(\cdot) : \mathbb{R}^n \to \mathbb{R}_+$, mensura a importância (ou intensidade de preferência) de um par de pontos, e a função similaridade, $\varphi(\cdot, \cdot) : \mathbb{R}^n \to [0, 1]$, é responsável pela correlação entre esses pontos.

A decomposição serve tanto para impor implicitamente a restrição de que C deve ser positiva semidefinida quanto permitir uma interpretação amigável da função kernel e a modelagem de forma independente da qualidade e da similaridade.

2.2 Conjuntos L

Apesar da definição formal de processos pontuais serem obtidas pelas densidades produto, para modelagem de dados a construção mais relevante é por meio de conjuntos L. Os conjuntos L (L *ensembles*), são um caso particular de PPD's introduzidos por [19]. Nesse caso, os conjuntos L não definem um PPD através do kernel marginal como em (2.3), mas por uma matriz real positiva semidefinida, L, indexada pelos elementos de $X \subset \mathcal{X}$. Dessa forma, L *ensembles* modelam diretamente as probabilidades de observar cada subconjunto de $\mathcal{X}$, eventos atômicos diretamente, o que é vantajoso para otimização e para a interpretação.

No caso de espaço de estados contínuo, para os conjuntos L a matriz é construída por uma função de covariância C_X de ordem $K \times K$ com entradas $C(i, j)$ dadas por uma função de covariância contínua $C(x_i, x_j)$ e os autovalores λ's da função kernel associados ao operador $\int_S C(x, y)h(y)dy$. Além das vantagens já citadas, as condições de existência do PPD definido por conjuntos L se resumem a C ser positiva semidefinida, e os autovalores associado serem limitados superiormente. A seguir, é apresentada a função densidade de probabilidade de PPD *L-ensembles*.

Definição 2.2.1. *L-emsembles, caso contínuo*

Seja o conjunto limitado $S \subseteq \mathbb{R}^D$ no qual está definido o processo pontual $\mathcal{X}$. A função densidade de probabilidade de $\boldsymbol{X} = \{x_1, \ldots, x_K\}$, $x_K \in S \ \forall\ k$, uma realização de $\mathcal{X}$, é dada por

$$p(\boldsymbol{X}) = \frac{\det(C_{\boldsymbol{X}})}{\prod_{\mathbf{h}} (\lambda_{\mathbf{h}} + 1)}, \tag{2.7}$$

em que $C_{\boldsymbol{X}} = C(x_1, \ldots, x_K)$ é uma matriz de ordem $K \times K$ definida por uma função kernel, função de covariância no caso real, para os pontos de $\mathbf{X}$.

As quantidades $\lambda_{\mathbf{h}_1}, \lambda_{\mathbf{h}_2}, \ldots$ são chamados autovalores do operador kernel (função de covariância contínua complexa) e são indexados pelo índice multivariado $\mathbf{h}$, oriundos da decomposição espectral $C(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{h}}^{\infty} \lambda_{\mathbf{h}} \varphi(\mathbf{x}) \overline{\varphi(\mathbf{y})}, \mathbf{x}, \mathbf{y} \in S \times S$, em que $\overline{\varphi(\mathbf{y})}$ é o conjugado complexo de $\varphi(\mathbf{y})$. Os valores de $\lambda_{\mathbf{h}}$ dependem da função $C(\cdot)$ e seus possíveis parâmetros.

Uma das dificuldades relacionadas aos PPD's em espaços de estados contínuos é lidar com suas funções densidade de probabilidade, pois os autovalores $\lambda_{\mathbf{h}}$ em (2.7) são, geralmente, desconhecidos. Métodos numéricos foram desenvolvidos para se obter resultados aproximados para tais funções densidade [66].

Considerando o modelo de mistura finita via PPD apresentado em (2.1), a distribuição a priori de $\boldsymbol{\mu}|C) \sim PPD(C)$, é dada por

$$p(\boldsymbol{\mu}|\theta, \sigma_q) = \frac{\det(C_\mu)}{\prod_{\mathbf{h}=1}^{\infty} (\lambda_{\mathbf{h}} + 1)},$$

com $\boldsymbol{\mu} = \{\mu_1, \ldots, \mu_K\}$, $\mu_k \in \mathbb{R}^D \ \forall \ k$. Esta função densidade de probabilidade é definida pela função de covariância que pode assumir muitas formas, contanto que mantenha as características que a definem como tal. Na Seção 2.3, será detalhada a função considerada neste trabalho.

2.3 Kernel Gaussiano

O Kernel Exponencial Quadrático ou Kernel Gaussiano é amplamente conhecido não apenas na definição de PPD, mas em Processos Pontuais em geral. Sua utilidade é muito relevante devido as suas propriedades analíticas entre elas a facilidade de se obter os seus autovalores por uma forma analítica fechada [2, 39].

Neste trabalho, será utilizado o Kernel Exponencial Quadrático definido por [118] considerando a decomposição proposta por [64] em que ambas, as funções qualidade e similaridade, são definidas a partir da distribuição normal.

Definição 2.3.1. *Kernel Gaussiano*

Sejam $\boldsymbol{\mu}_k = (\mu_{k1}, \ldots, \mu_{kD})$ *e* $\boldsymbol{\mu}_l = (\mu_{l1}, \ldots, \mu_{lD}) \in B \subseteq \mathbb{R}^D$. *O kernel Exponencial Quadrático do PPD assume forma* $C_{\theta^2,\sigma^2}(\boldsymbol{\mu}_k, \boldsymbol{\mu}_l) = q(\boldsymbol{\mu}_k)\varphi(\boldsymbol{\mu}_k, \boldsymbol{\mu}_l)q(\boldsymbol{\mu}_l)$ *em que a função de qualidade,* $q(\cdot)$*, é dada por*

$$q(\boldsymbol{\mu}_k) = \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi}\sigma_q} \exp\left\{ -\frac{\mu_{kd}^2}{2\sigma_q^2} \right\}, \tag{2.8}$$

e a função similaridade é dada por

$$\varphi(\boldsymbol{\mu}_k, \boldsymbol{\mu}_l) = \exp\left\{ -\sum_{d=1}^{D} \frac{(\mu_{kd} - \mu_{ld})^2}{\theta^2} \right\}. \tag{2.9}$$

Os autovalores do Kernel Exponencial Quadrático, Definição 2.3.1, são conhecidos e podem ser obtidos em função de θ^2 e σ^2 utilizando a fórmula

$$\lambda_{\mathbf{h}} = \left(\frac{2a}{a+b+c} \right)^{D/2} \left(\frac{b}{a+b+c} \right)^{\sum_{d=1}^{D}(h_d - 1)} \tag{2.10}$$

em que $a = \dfrac{1}{4\sigma_q^2}, b = \dfrac{1}{\theta^2}$ e $c = \sqrt{a^2 + 2ab}$ e, ainda, $\mathbf{h} = (h_1, \ldots, h_D)$ é um indice multivariado de dimensão D com componentes, $h_d \in \mathbb{Z}^+$, que representam o número de autovalores iguais.

A partir das expressões (2.8) e (2.9), o kernel $C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_l) = q(\boldsymbol{\mu}_k)\varphi(\boldsymbol{\mu}_k, \boldsymbol{\mu}_l)q(\boldsymbol{\mu}_l)$ assume a seguinte forma

$$C_{\theta^2,\sigma^2}(\boldsymbol{\mu}_k, \boldsymbol{\mu}_l) = (2\pi\sigma_q^2)^{-D/2} \exp\left\{ -\sum_{d=1}^{D} \left[\left(\frac{\theta^2 + 2\sigma_q^2}{2\sigma_q^2\theta^2} \right)(\mu_{kd}^2 + \mu_{ld}^2) - \frac{2\mu_{kd}\mu_{ld}}{\theta^2} \right] \right\}. \tag{2.11}$$

Para o kernel Gaussiano a função intensidade, $\rho(\boldsymbol{\mu}_k)$, e a função correlação par, $g(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j) = 1 - [C(\boldsymbol{\mu}_i; \boldsymbol{\mu}_j)^2]/[\rho(\boldsymbol{\mu}_i)\rho(\boldsymbol{\mu}_j)]$, são dadas por

$$\rho(\boldsymbol{\mu}_k) = [q(\boldsymbol{\mu})]^2 = (2\pi\sigma_q^2)^{-D} \exp\left\{ - \sum_{d=1}^{D} \frac{\mu_{kd}^2}{\sigma_q^2} \right\}$$

e

$$g(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j) = 1 - \varphi^2(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j) = 1 - \exp\left\{ - 2 \sum_{d=1}^{D} \frac{(\mu_{kd} - \mu_{ld})^2}{\theta^2} \right\}.$$

O valor do kernel depende da função similaridade e da função qualidade. A função qualidade controla a ocorrência de pontos, por estar relacionada a função intensidade do processo. Consequentemente, o parâmetro σ_q^2 controla a variabilidade e a extensão de ocorrência desses pontos no espaço em torno da média.

A função qualidade utilizada neste trabalho é o produto de densidades da distribuição normal centrada em zero e com variância σ_q^2. Mudanças na locação desta função podem levar a melhores estimativas. Por exemplo, no contexto de estimação de densidade poderia ser mais eficiente centrar a distribuição na média dos dados. Mas, no contexto mais geral, em que os parâmetros representam efeito de covariáveis, manter a média zero faz mais sentido por que, geralmente, é melhor permitir que os efeitos possam ser negativos ou positivos. A mudança na médias das funções normais utilizadas em $q(\cdot)$ tem a vantagem de não alterar os autovalores do kernel Gaussiano [54].

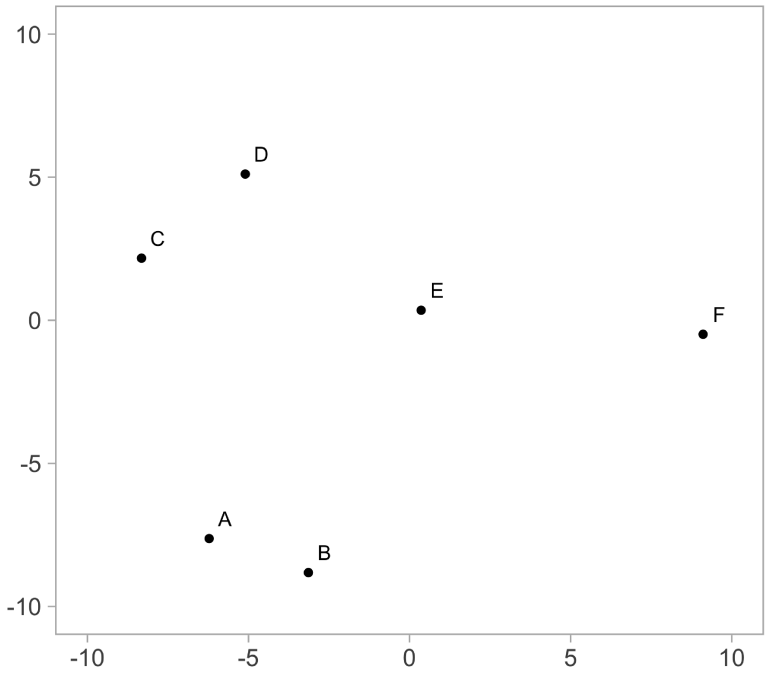
A função similaridade se refere ao relacionamento dos pontos entre si. Essa função carrega a estrutura de repulsão e seu parâmetro θ^2 pode ser utilizado para ajustar a distância mínima de ocorrência entre os pontos. A relação entre esses dois parâmetros é discutida na próxima seção.

A função densidade de probabilidade dada em (2.7) é obtida para o kernel Gaussiano substituindo os elementos da matriz $C_{\mathbf{X}}$ pelos valores da função dada em 2.11. Uma forma fechada para o determinante não é acessível para a maioria dos casos, devido a necessidade de recursão no cálculo para matrizes de dimensões maiores. Apesar disso, é possível ilustrar o comportamento da função kernel Gaussiana para o caso bidimensional considerando um vetor de médias $\mu = (\mu_A, \mu_B, \mu_C, \mu_D, \mu_E, \mu_F) = (-6,22; -7,63); (-3,14; -8,82); (-8,32; 2,17); (-5,10; 5,11); (0,36; 0,35); (9,11; -0,49))$, por exemplo. A configuração dessas médias no plano bidimensional pode ser conferida na Figura 2.1. A mudança de comportamento da função kernel para o caso $D = 2$, a partir da variação dos valores de θ^2 e σ_q^2 é apresentada na Figura 2.2.

Na Figura 2.2, os quadriculados mais claros possuem menor valor para a função kernel, ou seja, uma menor covariância entre os pares de médias da linha e da coluna a qual se associam, assumindo valor *zero* na cor branca. Enquanto os quadriculados mais azuis escuro mostram maior covariância representadas pela função kernel entre as médias.

Identifica-se que, em todos os gráficos da Figura 2.2, o quadriculado mais escuro é o da diagonal referente à média E, assumindo o maior valor de covariância em cada um dos nove

Figura 2.1: Vetor de médias usado como exemplo para ilustrar o comportamento da função kernel Gaussiana.



Fonte: Elaborado pela autora.

casos mostrados. Esta é a média mais próxima da origem, Figura 2.1, portanto tem o maior valor de função qualidade. Isso acontece porque as distribuições normais utilizadas em $q(\cdot)$ são centradas no *zero*, ponto de máximo para a função.

Valores menores de θ^2 impõem uma estrutura de covariância próxima a independência (não correlação) como pode ser visto nas Figuras 2.2a, 2.2b e 2.2c. Enquanto valores maiores de θ^2 impõem maior covariância, consequentemente, maior dependência entre os pares de médias: mais quadriculados com cores mais azul escuras fora da diagonal como nas Figuras 2.2g, 2.2h e 2.2i. Além disso, quanto maior os valores de σ_q^2 maiores são as covariâncias em geral, comparando relativamente ao quadriculado mais azul em cada gráfico.

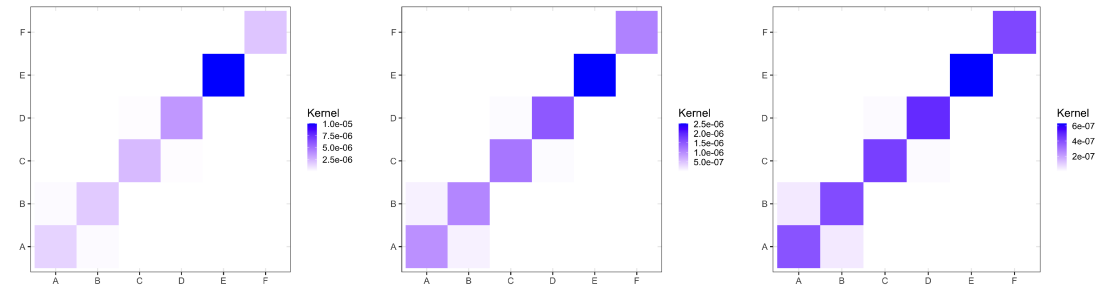
A Figura 2.2 mostra também que os pares de médias mais próximos são os que apresentam valores maiores da função kernel no PPD com o kernel Gaussiano. Por exemplo, na Figura 2.2f, os pares de médias A e B com distância 3,3; e C e D com distância 4,4 que são os mais próximos entre si, tem quadriculados com azul mais escuro. Enquanto o par de médias C e F com a maior distância 17,6, possui valor da função kernel praticamente nulo.

O cálculo analítico e a visualização geral da função densidade em dimensões maiores é comprometida em função da limitação tridimensional. No entanto, é possível mostrar o comportamento utilizando as curvas de níveis da função.

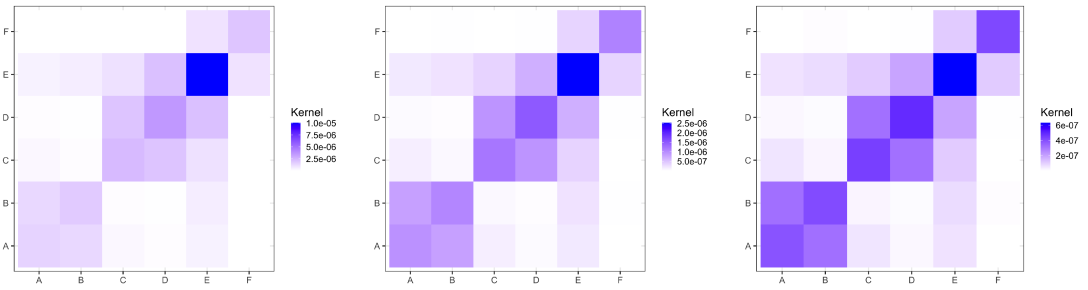
Os PPD's oferecem muitas vantagens práticas, tais como: serem processos pontuais regulares [73], apresentarem ausência de ocorrências múltiplas, possuírem todos os seus momentos finitos e propriedades de suavidade. No entanto, existem muitas dificuldades em termos

Figura 2.2: Matriz de calor da função kernel, considerando o vetor de médias (μ_A, μ_B, μ_C, μ_D, μ_E, μ_F) = $(-6, 22; -7, 63); (-3, 14; -8, 82); (-8, 32; 2, 17); (-5, 10; 5, 11); (0, 36; 0, 35); (9, 11; -0, 49)$).

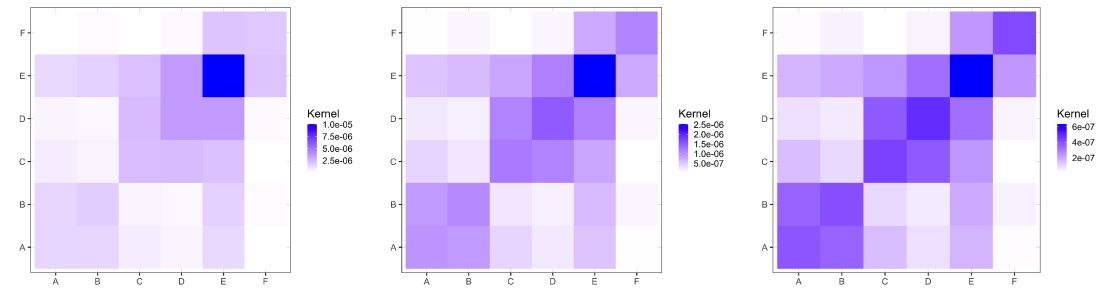
(a) $\theta^2 = 5$ e $\sigma_q^2 = 50$. (b) $\theta^2 = 5$ e $\sigma_q^2 = 100$. (c) $\theta^2 = 5$ e $\sigma_q^2 = 200$.



(d) $\theta^2 = 50$ e $\sigma_q^2 = 50$. (e) $\theta^2 = 50$ e $\sigma_q^2 = 100$. (f) $\theta^2 = 50$ e $\sigma_q^2 = 200$.



(g) $\theta^2 = 100$ e $\sigma_q^2 = 50$. (h) $\theta^2 = 100$ e $\sigma_q^2 = 100$. (i) $\theta^2 = 100$ e $\sigma_q^2 = 200$.



Fonte: Elaborado pela autora.

de estimação dos seus parâmetros. Num caso geral contínuo, seria necessário estimar a função kernel para todos os possíveis conjuntos de pontos do espaço de estados onde o processo é definido.

2.4 Distribuição *a priori* de $\boldsymbol{\mu}$

No entanto, para o caso bivariado $\boldsymbol{\mu} = (\boldsymbol{\mu}_i, \boldsymbol{\mu}_j)$ pode ser obtida a função densidade produto de ordem 2.

A distribuição *a priori* marginal para $\boldsymbol{\mu}|(\theta^2, \sigma_q^2)$ é obtida da distribuição dada em (2.7). Isto implica no cálculo do determinante, ou seja, um cálculo iterativo que depende da dimensão da matriz C. Na próxima definição, será apresentada a distribuição *a priori* marginal para $\boldsymbol{\mu}|(\theta^2, \sigma_q^2)$ no caso bivariado, $\boldsymbol{\mu} = (\boldsymbol{\mu}_i, \boldsymbol{\mu}_j)$, em que cada média é um vetor D-dimensional, $\boldsymbol{\mu}_i = (\mu_{i1}, \ldots, \mu_{iD})$.

Definição 2.4.1. *Densidade Marginal a priori para* $\boldsymbol{\mu}$

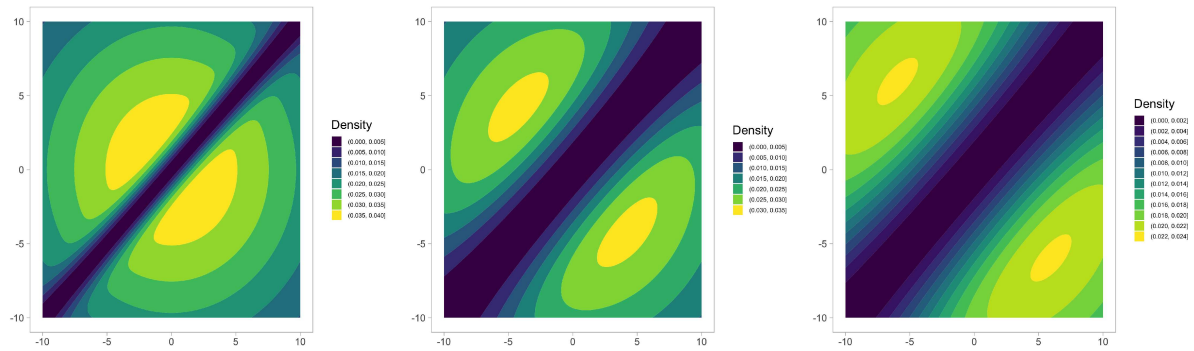
A densidade produto de ordem 2, marginal, para $\boldsymbol{\mu} = (\boldsymbol{\mu}_i, \boldsymbol{\mu}_j)$*, é dada por*

$$p(\boldsymbol{\mu}|\theta, \sigma_q) \propto ae^{-b}[1 - e^{-c}], \tag{2.12}$$

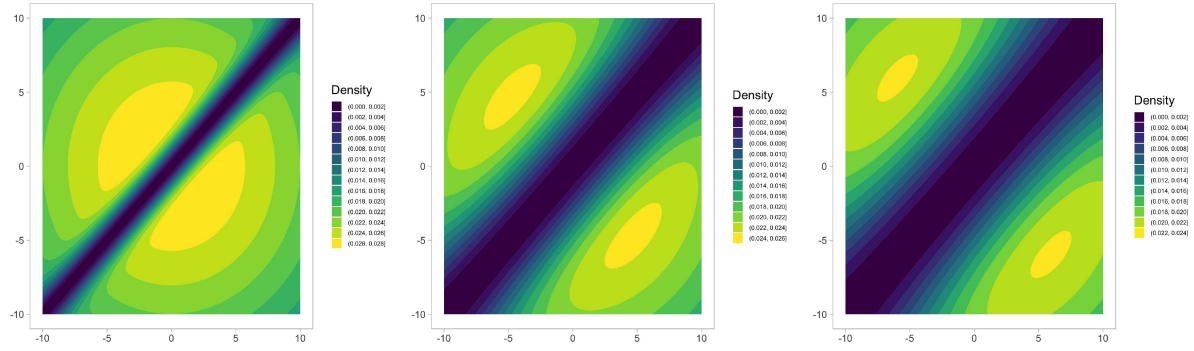
em que $a = \left(\frac{1}{\sqrt{2\pi}\sigma_q}\right)^{2D}$, $b = \sum_{d=1}^{D}\left(\frac{\mu_{id}^2 + \mu_{jd}^2}{2\sigma_q^2}\right)$ *e* $c = 2\sum_{d=1}^{D}\frac{(\mu_{id} - \mu_{jd})^2}{\theta^2}$.

Figura 2.3: Curvas de nível da distribuição marginal de $\boldsymbol{\mu}$, caso unidimensional, utilizando o kernel exponencial quadrático para $\theta^2 = 25$ e $\sigma_q^2 = 12,5$.

(a) $\theta^2 = 5$ e $\sigma_q^2 = 100$. (b) $\theta^2 = 50$ e $\sigma_q^2 = 100$. (c) $\theta^2 = 100$ e $\sigma_q^2 = 100$.



(d) $\theta^2 = 5$ e $\sigma_q^2 = 200$. (e) $\theta^2 = 50$ e $\sigma_q^2 = 200$. (f) $\theta^2 = 100$ e $\sigma_q^2 = 200$.



Fonte: Elaborado pela autora.

Considerando $D = 1$, a Figura 2.3 mostra o comportamento da distribuição marginal *a priori* de $\boldsymbol{\mu}$ para o kernel Gaussiano no caso bivariado, duas médias. As curvas de nível são mostradas para dois valores de σ_q^2 e três valores de θ^2. O comportamento bimodal fica evidente, e é esperado por serem duas médias. Além disso, as regiões mais claras (modas da distribuição) tendem a serem mais afastadas com o aumento do valor de θ^2 isso ilustra o comportamento repulsivo do PPD. As regiões mais escuras são de baixa densidade, ou seja, a probabilidade de duas médias muito próximas ocorrerem conjuntamente nesta distribuição é quase nula. Além disso, a região mais clara aumenta de tamanho a medida que o valor de σ_q^2 aumenta, tornando possível a ocorrência das médias em regiões maiores, mas ainda distantes de acordo com o valor de θ^2. Se houver interesse em diminuir a repulsão valores de $\theta^2 \to 0$ são interessantes. Dessa forma, um estrutura próxima a indepenência será obtida.

2.5 Equilíbrio entre a intensidade e a repulsão

Segundo [8], um PPD válido só é possível se houver equilíbrio entre a intensidade de ocorrência de pontos e o intervalo de repulsão neste tipo de kernel. Considerando um função intensidade, $\rho(\mathbf{x})$, e a função de correlação $R(\mathbf{x}, \mathbf{y}) = \exp\left\{-\dfrac{(\mathbf{x}-\mathbf{y})^2}{\theta^2}\right\}$, um PPD definido pela função kernel

$$C(\mathbf{x}, \mathbf{y}) = \sqrt{\rho(\mathbf{x})\rho(\mathbf{y})}R(\mathbf{x}, \mathbf{y})$$

é válido se

$$\rho(\mathbf{x}) \leq \frac{1}{\pi\theta^2}. \tag{2.13}$$

No contexto do kernel de similaridade Gaussiano em (2.11), θ é parâmetro de escala e pode ser interpretado como a distância mínima entre os pontos para que uma probabilidade razoável de ocorrência seja obtida em um espaço limitado. Dessa forma, grandes valores de θ, exigem pontos separados por uma grande distância e num espaço limitado isso pode inviabilizar probabilidades positivas de ocorrência de pontos.

Assim, é necessário impor um limite à função intensidade $\rho(\cdot) = q^2(\cdot)$ e a função qualidade não deve ser considerada independente de θ^2. Nesse sentido, considerando a função intensidade definida em (2.4) é possível definir uma condição que garanta a validade do PPD. A proposta fornece um meio de relacionar os parâmetros do kernel exponencial com equilíbrio entre a função intensidade e o parâmetro de repulsão, θ. Na proposição é obtida uma condição para se ter um PPD válido com kernel exponencial quadrático.

Proposição 2.5.1. *Considerando que para o kernel exponencial quadrático a função intensidade é dada por* $\rho(\boldsymbol{\mu}) = q^2(\boldsymbol{\mu})$*, a restrição que garante um PPD válido para o modelo proposto em* (2.7) *e* (2.2) *com o kernel exponencial quadrático é dada por*

$$0 \leq \theta^2 \leq \frac{(2\pi\sigma_q^2)^D}{\pi}. \tag{2.14}$$

A condição em (2.14), para $D = 2$, torna-se $0 \leq \theta^2 \leq 4\pi\sigma_q^4$ e, para $D = 1$, é $0 \leq \theta^2 \leq 2\sigma_q^2$. Esta condição será considerada no processo de estimação ao longo desse trabalho.

Capítulo 3

Modelo de Mistura Normal Independente via PPD

Neste capítulo, será considerada a família de distribuições Normal/Independente (NI) para a modelagem do comportamento das observações em um Modelo de Mistura Finita (MMF). Além disso, o Processo Pontual por Determinante (PPD) é utilizado como distribuição *a priori* para o parâmetro de escala. O objetivo é obter boas alocações dos indivíduos aos grupos latentes representados pelos componentes da mistura a que pertencem. O ganho em flexibilidade proporcionado pela NI permite acomodar o comportamento disperso de observações mais afastadas e ainda mantém características analíticas vantajosas em relação como sua estrutura estocástica que ajuda a obtenção de infeR6encia *a posteriori* por conjugação de famílias de distribuições conhecidas. Alem disso, foi implementado um algoritmo de Monte Carlo para estimação de todos os parâmetros do modelo, inclusive do kernel Gaussiano.

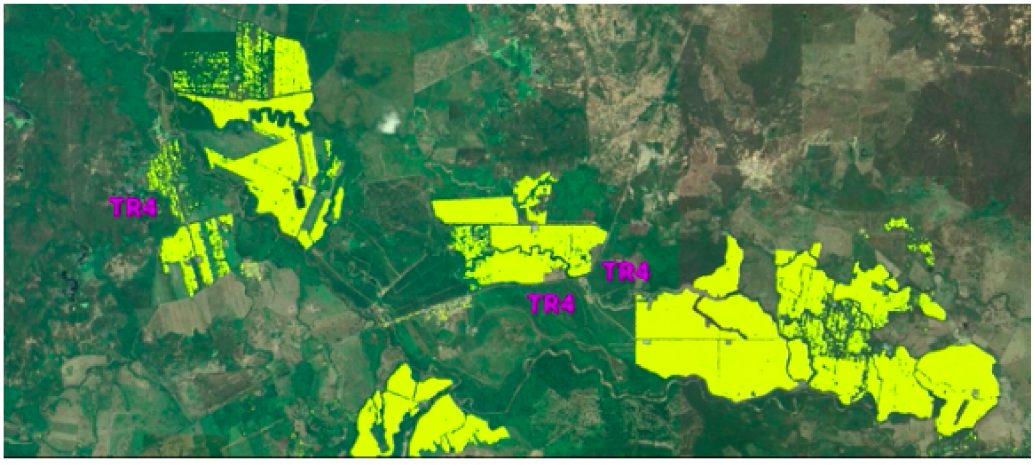
3.1 Introdução

Com a quantidade de dados geradas na atualidade é comum observar conjuntos que apresentam *outliers* e caudas pesadas. Problemas práticos que envolvem classificação e agrupamento ocorrem nas mais diversas áreas. Desde aplicações em *marketing* digital, reconhecimento de imagem para diagnóstico e acompanhamento de doenças, até controle de pragas em lavouras e monitoramento de incêndios em florestas. Conhecida como análise de agrupamento (ou *clustering*) a teoria envolve vários e diferentes métodos, alguns baseados em algoritmos de particionamento como o k-médias [52] ou agrupamento hierárquico [104]. Estas heurísticas, no entanto, têm algumas limitações. O *k*-médias, por exemplo, está mais voltado para a construção de grupos que gozam de maior homogeneidade interna, do que para uma boa separação entre eles. Em outros casos, tais métodos são sensíveis à escolha de sementes de inicialização ou de medidas de similaridade, como ocorre com o agrupamento hierárquico, por exemplo. Além disso, precisam de um número pré-definido de grupos, geralmente escolhido arbitraria-

mente sem levar em conta o problema ou o comportamento dos dados. Já os métodos baseados em densidades [62] são algoritmos não supervisionados que definem grupos como uma região contígua no espaço dos dados com alta ocorrência de pontos. Estas regiões são consideradas separadas por outras regiões com baixa ocorrência de pontos. O principal exemplo é o DBSCAN (*Density Based Spatial Clustering of Application with Noise*) [36], muito utilizado por ser, segundo os autores, aplicável a qualquer base de dados de um espaço métrico usando uma função da distância. Estes métodos, apesar de serem muito difundidos, principalmente por sua velocidade computacional, às vezes são menos interpretáveis e não quantificam a incerteza do agrupamento devido à falta de uma base probabilística. Para fornecer os elementos necessários à inferência, os modelos estatísticos são essenciais.

Um exemplo atual e pertinente da análise de agrupamento é o acompanhamento do plantio comercial da bananeira e seu acometimento pelo mal-do-Panamá. Essa praga é causada pelo fungo de solo *Fusarium oxysporum f. sp. cubense* (Foc) e é considerada uma das dez doenças mais importantes da história da agricultura mundial [24]. Grandes prejuízos foram relatados desde 1904 quando a raça 1 de Foc causou o desaparecimento dos plantios comerciais da variedade de tipo exportação *Gros Michel*. A única solução encontrada foi substituí-la por clones do subgrupo *Cavendish* que é resistente a essa raça do fungo. Apesar de existirem inúmeras variedades de banana para cultivo doméstico, atualmente apenas a *Cavendish*, tem potencial de comercio internacional. Nas últimas décadas uma nova cepa do fungo, a Raça 4 Tropical - R4T (*Tropical Race 4* - TR4), tem se espalhado a partir da Ásia ameaçando a *Cavendish*. Não existem controles químicos da doença e o R4T pode causar estragos irreparáveis na América Latina e Caribe, os maiores produtores da fruta [18].

Figura 3.1: Plantio de banana em La Guajira - Colombia, dezembro de 2019.



Fonte: Projeto Bananaex, [11].

A Figura 3.1 mostra o acompanhamento das plantações (destacadas em amarelo) de banana em La Guajira, um dos 32 departamentos da Colombia, onde R4T chegou no ano de 2019, os locais de infecção são destacados em rosa. O fungo foi registrado três meses antes, no acompanhamento de junho de 2019. O controle da praga é feito por extermínio das plantas

contaminadas e abandono dos campos afetados. Isto ocorre com a região abaixo da área central a esquerda no mapa. Uma parte dessa região de plantil já não apresenta sequer resquícios. Uma importante contribuição tanto para a estatística teórica quanto prática é um modelo eficiente na determinação dos agrupamentos que representam os plantios afetados e que seja capaz de lidar com as margens dessas regiões quando se mostram dispersas, mas ainda pertencem ao grupo. No entanto, modelos de agrupamento que levem em consideração caudas mais pesada ainda são poucos, [107, 82], por exemplo.

Neste capítulo, uma extensão da proposta de [118] é desenvolvida assumindo que cada indivíduo pertence a um grupo ou subpopulação e que sua resposta tem comportamento modelado por uma distribuição da família Normal/Independente. Além disso, foi realizado um estudo sobre o comportamento do kernel exponencial quadrático para o modelo de mistura NI, o que permitiu fornecer uma metodologia de inferência completamente Bayesiana por meio de um algoritmo MCMC para a estimação de todos os parâmetros do modelo, inclusive aqueles que definem o kernel. A organização do trabalho se dá da seguinte forma: na Seção 3.2, é desenvolvido o modelo de mistura Normal/Independente via Processo Pontual por Determinante (NIPPD); a Seção 3.3 detalha a estrutura do método Bayesiano utilizado; o algoritmo MCMC para estimação dos parâmetros é apresentado na Seção 3.4; a Seção 3.5 mostra um estudo de simulação em vários cenários; 3.6 é dedicada a uma aplicação prática do modelo em dados de plantio de banana; e, finalmente, a Seção 3.7 expõe discussões e conclusões do trabalho.

3.2 Modelo Proposto

Seja $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N), \mathbf{y}_i \in \mathbb{R}^D$, $i = 1, \ldots, N$, um conjunto de observações independentes de um modelo de mistura finita D-dimensional, $D \geq 1$, com K componentes cuja densidade é dada por

$$p(\mathbf{y}_i) = \sum_{k=1}^{K} w_k p(\mathbf{y}_i|\boldsymbol{\theta}_k),$$

em que $w_k, k = 1, \ldots, K$, são os pesos da mistura com $0 < w_k < 1$ e sob a restrição $\sum_{k=1}^{K} w_k = 1$. A função densidade, $p(\mathbf{y}|\boldsymbol{\theta}_k)$, do componente k é definida por uma família paramétrica indexada pelo vetor de parâmetros $\boldsymbol{\theta}_k$. A densidade utilizada neste trabalho é da forma:

$$\begin{aligned} p(\mathbf{y}_i|z_i = k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, u_i) &= \det(u_i^{-1}\boldsymbol{\Sigma}_k)^{-1/2}(2\pi)^{-D/2} \\ &\times \exp\Big\{-\frac{u_i}{2}(\mathbf{y}_i - \boldsymbol{\mu}_k)'\boldsymbol{\Sigma}_k^{-1}(\mathbf{y}_i - \boldsymbol{\mu}_k)\Big\}, \end{aligned} \tag{3.1}$$

$i = 1, \ldots, N$, $k = 1, \ldots, K$, em que $\det(\cdot)$ representa o determinante de uma matriz, $\boldsymbol{\mu}_k$ é o parâmetro de locação, $\boldsymbol{\Sigma}_k$ é o parâmetro de escala e u_i é uma variável misturadora. O ve-

tor de parâmetros é denotado por $\boldsymbol{\Theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{u})$ com $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K)$, $\boldsymbol{\Sigma} = (\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K)$, $\mathbf{u} = (u_1, \ldots, u_N)$. Para o parâmetro de escala, a representação $\boldsymbol{\Sigma}_k = E\boldsymbol{\mathcal{T}}_k E'$ pode ser obtida através da Decomposição em Valores Singulares (*Singular Value Decomposition - SVD*) [ver [113]] proposta por [119] para facilitar o processo de estimação de matrizes de variância reduzindo o número de parâmetros no caso de dimensão $D \geq 2$. Neste caso, $\boldsymbol{\mathcal{T}}_k = diag(\boldsymbol{\tau}_k)$, com $\boldsymbol{\tau}_k = (\tau_{k1}, \ldots, \tau_{kD})$ é usado para simplificar a notação e E é uma matriz ortogonal constante de dimensão $D \times D$. Em particular, esta decomposição retorna à forma $\boldsymbol{\Sigma}_k = \sigma^2$, no caso unidimensional. Os detalhes são fornecidos no Apêndice A.

Além disso, será usada a notação $\mathbf{w} = (w_1, \ldots, w_K)$ e $\mathbf{z} = (z_1, \ldots, z_N)$, em que $z_i = k$ é uma variável latente que indica que a i-ésima observação pertence ao k-ésimo componente da mistura, $k = 1, \ldots, K$.

O Modelo de Mistura de distribuições Normais/Independentes via Processo Pontual por Determinante (NIPPD) será desenvolvido a partir de uma estrutura modificada que conta com uma variável misturadora, $\mathbf{u} = (u_1, \ldots, u_N)$ possibilitando o uso da Família de Distribuições e suas propriedades estocásticas:

$$\mathbf{y}_i | z_i = k, \boldsymbol{\mu}_k, u_i, \boldsymbol{\mathcal{T}}_k, K \ldots \sim N_D(\boldsymbol{\mu}_k, u_i^{-1} E\boldsymbol{\mathcal{T}}_k E'); \tag{3.2}$$

$$P(z_i = k | \mathbf{w}, K) = w_k; \tag{3.3}$$

$$\mathbf{w} = (w_1, \ldots, w_K) | K \sim Dirichlet(\boldsymbol{\delta}), \boldsymbol{\delta} = (\delta_1, \ldots, \delta_K); \tag{3.4}$$

$$\mathbf{u} = (u_1, \ldots, u_N) \text{ com } u_i | \eta \overset{\text{iid}}{\sim} F_{U|\boldsymbol{\eta}} \tag{3.5}$$

$$\boldsymbol{\tau}_k = (\tau_{k1}, \ldots, \tau_{kD}) \text{ com } \tau_{kd}^{-1} \overset{\text{iid}}{\sim} Gama(a_0, b_0), k = 1, \ldots, K, d = 1, \ldots, D; \tag{3.6}$$

$$\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K) | \theta^2, \sigma_q^2, K \sim PPD(C, \theta^2, \sigma_q^2) \tag{3.7}$$

$$\theta^2 | \sigma_q^2 \sim GamaT(a_1, b_1, t_1), \theta^2 \in t_1 = \left(0; \frac{(2\pi\sigma^2)^D}{\pi}\right) \tag{3.8}$$

$$\sigma_q^2 \sim Gama(a_2, b_2). \tag{3.9}$$

Em (3.8), $GamaT(a_1, b_1, t_1)$ indica que será usada a distribuição Gama Truncada com parâmetros a_1, b_1 e intervalo de truncamento dado por t_1.

Nesse caso, o vetor aleatório $\mathbf{y_i} | z_i = k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, u_i$ tem distribuiçãoo na classe NI, com parâmetros de locação $\boldsymbol{\mu}_K$, de escala $\boldsymbol{\Sigma}_k$ e de forma η, ou seja, $\mathbf{y_i} | z_i = k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, u_i \sim NI(\boldsymbol{\mu}_K; \boldsymbol{\Sigma}_k; F_{U_i|\eta})$, e sua distribuição tem função densidade de probabilidade dada em (1.2). A média e a variância da distribuição na classe NI são dadas, respectivamente, por

$$E[\mathbf{y_i} | z_i = k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, u_i] = \boldsymbol{\mu}_K \quad \text{e} \quad V[\mathbf{y_i} | z_i = k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, u_i] = E(u_i^{-1})\boldsymbol{\Sigma}_k.$$

A escolha do comportamento probabilístico de u_i leva a diferentes distribuições para $\mathbf{y_i}$. A seguir serão apresentados três exemplos, e para simplificar a notação a dependência de $z_i = k$ e os índices i, k serão omitidos.

1. Se $u|\eta \sim Gama(\eta/2; \eta/2)$ então $\mathbf{y}$ tem distribuição t de Student multivariada com parâmetros de locação $\boldsymbol{\mu} \in \mathbb{R}^D$, de escala $\boldsymbol{\Sigma} \in \mathbb{R}^{D \times D}$ e com graus de liberdade $\eta > 0$. Será usada a

notação $T_n(\boldsymbol{\mu};\boldsymbol{\Sigma};\eta)$ e sua f.d.p. é dada por

$$f_{\mathbf{y}|\boldsymbol{\mu},\boldsymbol{\Sigma},u,\eta}(\mathbf{y}) = \frac{\Gamma((\eta+D)/2)}{|\boldsymbol{\Sigma}|^{1/2}\pi^{D/2}\Gamma(\eta/2)\eta^{D/2}}\left[1+\frac{(\mathbf{y}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}}{\eta}\right]^{-(\eta+D)/2},$$

para $\eta > 2$, em que a matriz de covariância de $\mathbf{y}$ é dada por

$$V[\mathbf{y}] = \left(\frac{\eta}{\eta-2}\right)\boldsymbol{\Sigma}, \text{ para } \eta > 2.$$

2. Se $u|\eta \sim Beta(\eta,1)$ então $\mathbf{y}$ tem distribuição Slash multivariada com parâmetros de locação $\boldsymbol{\mu} \in \mathbb{R}^D$, de escala $\boldsymbol{\Sigma} \in \mathbb{R}^{D\times D}$ e forma $\eta > 0$. A notação $\mathbf{y} \sim SL_D(\boldsymbol{\mu},\boldsymbol{\Sigma},\eta)$ será usada e sua a f.d.p. dada por

$$f_{\mathbf{y}|\boldsymbol{\mu},\boldsymbol{\Sigma},u,\eta}(\mathbf{y}) = \eta\int_0^1 u^{\eta-1}\phi_D(\mathbf{y}|\boldsymbol{\mu},u^{-1}\boldsymbol{\Sigma})du,$$

em que $\phi(\cdot)$ é a função densidade da distribuição Normal e a matriz de covariância de $\mathbf{y}$ é dada por

$$V[\mathbf{y}] = \left(\frac{\eta}{\eta-1}\right)\boldsymbol{\Sigma}, \text{ para } \eta > 1.$$

3. Se $f_{|\eta_1,\eta_2}(u) = \eta_1\mathbf{1}_{\{u=\eta_2\}} + (1-\eta_1)\mathbf{1}_{\{u=1\}}$ então $\mathbf{y}$ tem distribuição Normal Contaminada multivariada com parâmetros de locação $\boldsymbol{\mu}$, de escala $\boldsymbol{\Sigma}$ e em que η_1 e η_2 são parâmetros de forma, a notação utilizada será $\mathbf{y} \sim NC_D(\boldsymbol{\mu},\boldsymbol{\Sigma},\eta_1,\eta_2)$ e cuja a f.d.p. é dada por

$$f_{\mathbf{y}|\boldsymbol{\mu},\boldsymbol{\Sigma},u,\eta_1,\eta_2}(\mathbf{y}) = \eta_1\phi_D(\mathbf{y}|\boldsymbol{\mu},\eta_2^{-1}\boldsymbol{\Sigma}) + (1-\eta_1)\phi_D(\boldsymbol{\mu},\boldsymbol{\Sigma})$$

e a matriz de covariância de $\mathbf{y}$ é dada por

$$V[\mathbf{y}] = \left(\frac{\eta_1}{\eta_2}+1-\eta_1\right)\boldsymbol{\Sigma}.$$

Sem perda de generalidade, neste trabalho, será mostrado o caso em que $u|\eta \sim Gama(\eta/2;\eta/2)$ resultando em $\mathbf{y}_i|z_i = k,\boldsymbol{\mu}_k;\boldsymbol{\Sigma}_k;\eta \sim T_D(\boldsymbol{\mu}_k;\boldsymbol{\Sigma}_k;\eta)$, distribuição t de Student com η graus de liberdade. Assim, o vetor $\boldsymbol{\mu}_k$ será o parâmetro de locação e a matriz $\boldsymbol{\Sigma}_k$ será o parâmetro de escala. Para obter os demais modelos probabilísticos apenas será necessário mudar a distribuição $F_{U|\boldsymbol{\eta}}$.

Dessa forma, as principais diferenças entre o modelo (3.2) - (3.9) e o de [118] se devem a representação estocástica da família NI [65], definida em (1.2) e a estrutura dependência dos parâmetros θ^2 e σ_q^2 proposta e desenvolvida nesta tese e que será mais detalhada na Subseção 3.2.1. Esta extensão resulta num modelo mais flexível que permite caudas pesadas para o comportamento estocástico das observações, além de uma melhor estimacão do parâmetro de locação em consequência da estimção dos parâmetros do Kernel.

A metodologia completamente Bayesiana implica que as quantidades desconhecidas $K,\mathbf{z},\mathbf{w},\boldsymbol{\mu},\theta^2,\sigma_q^2,\mathbf{u},\boldsymbol{\mathcal{T}}$ devem ser estimadas a partir de distribuições *a posteriori* , no entanto,

para isso é necessário a definição de distribuições *a priori* adequadas. A densidade conjunta de todas as variáveis mencionadas no modelo é dada por

$$\begin{aligned} p(K, \mathbf{z}, \mathbf{w}, \boldsymbol{\mu}, \theta^2, \sigma_q^2, \mathbf{u}, \boldsymbol{\mathcal{T}}, \mathbf{y}) &= p(K)p(\mathbf{w}|K)p(\mathbf{z}|\mathbf{w}, K)p(\boldsymbol{\mu}|\theta^2, \sigma_q^2, K)p(\theta^2|\sigma_q^2) \\ &\times p(\sigma_q^2)p(\mathbf{u})p(\boldsymbol{\mathcal{T}}|K). \end{aligned} \tag{3.10}$$

Algumas considerações sobre as distribuições de probabilidade *a priori* são apresentadas a seguir enquanto os cálculos relacionados estão no Apêndice A. A distribuição *a priori* do parâmetro de locação necessita de uma caracterização maior assim será apresentada na Seção 3.2.1.

A maioria das distribuições *a priori* consideradas para os parâmetros do Modelo de Mistura NI em (3.2) são bem conhecidas e com informações consolidadas como é o caso do indicador de alocação, $\mathbf{z} = (z_1, \ldots, z_N)$, e sua distribuição *a priori* discreta, independente e identicamente distribuída para cada indivíduo i: $P(z_i = k|\mathbf{w}, K) = w_k, i = 1, \ldots, N$ e $k = 1, \ldots, K$; com $\sum_{k=1}^{K} P(z_i = k|\mathbf{w}, K) = \sum_{k=1}^{K} w_k = 1$. O mesmo ocorre com os pesos dos grupos, $\mathbf{w} = (w_1, \ldots, w_K)$, a priori, são identicamente distribuídos para todos os indivíduos e sua distribuição depende apenas de K, o número de grupos para definir sua dimensão: $\mathbf{w}|K \sim Dirichlet(\boldsymbol{\delta})$, com $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_K), \delta_k > 0, i = 1, \ldots, K$.

Sobre a variável misturadora, ele é o que garante maior flexibilidade para o modelo em termos de caudas pesadas. Seus componentes $\mathbf{u} = (u_1, \ldots, u_N)$, são considerados, a priori, independentes entre si e dos demais parâmetros do modelo. É exigido que u_i seja positiva, $i = 1, \ldots, N$. Sua distribuição, neste caso, é dada por: $u_i \overset{\text{iid}}{\sim} Gama(\eta/2, \eta/2)$.

A matriz escala, $E\boldsymbol{\mathcal{T}}_k E'$, é uma decomposição *SVD* em que E é uma matriz ortogonal constante, $\boldsymbol{\mathcal{T}}_k = diag(\boldsymbol{\tau}_k)$ é matriz diagonal de autovalores, dados por $\boldsymbol{\tau}_k = (\tau_{k1}, \ldots, \tau_{kD})$. Dessa forma, a estimação se resume às componentes do vetor $\boldsymbol{\tau}_k$ que são considerados independentes a priori.

3.2.1 Processo Pontual por Determinante

O vetor escala é modelado *a priori* por um Processo Pontual por Determinante. Um processo pontual representa o comportamento de uma configuração aleatória de pontos espaço limitado que é determinado principalmente pela sua densidade produto e função intensidade definidas no Capítulo 2.

Para o modelo dado em (3.2)-(3.8), o parâmetro de escala $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K)|K, \theta^2, \sigma_q^2 \sim PPD(C_{\theta^2, \sigma_q^2})$, distribução do PPD para conjuntos L. A distribuição depende da função kernel, C, que por sua vez é indexada pelos hiperparâmetros θ^2 e σ_q^2 e do número de grupos, K. Sua função densidade de probabilidade é dada por

$$p(\boldsymbol{\mu}|\theta^2, \sigma_q^2, K) = \frac{\det(C_{\theta^2,\sigma_q^2}(\boldsymbol{\mu}))}{\prod_{\mathbf{h}} (\lambda_{\mathbf{h}}(\theta^2, \sigma_q^2) + 1)}, \tag{3.11}$$

em que $C_{\theta^2,\sigma_q^2}(\boldsymbol{\mu})$, é uma matriz kernel, $K \times K$, para a qual seus elementos são obtidos a partir do kernel Gaussiano detalhada na Definição 2.3.1 (Seção 2.2, Capítulo 2). Os elementos da matriz $C_{\theta^2,\sigma_q^2}(\boldsymbol{\mu})$ serão representados por $C_{kl} = q(\boldsymbol{\mu}_k)\varphi(\boldsymbol{\mu}_k, \boldsymbol{\mu}_l)q(\boldsymbol{\mu}_l)$, k e $l \in \{1, \ldots, K\}$ em que $q(\cdot)$ e $\varphi(\cdot, \cdot)$ são definidas em (2.8) e (2.9), respectivamente.

Utilizando a partição $\boldsymbol{\mu} = (\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_k)$, em que $\boldsymbol{\mu}_{-k} = \{\boldsymbol{\mu}_j\}_{j\neq k}$, é o vetor $\boldsymbol{\mu}$ sem a k-ésima componente, a matriz $C_{\theta,\sigma_q}(\boldsymbol{\mu})$ pode ser particionada da seguinte forma

$$C_{\theta,\sigma_q}(\boldsymbol{\mu}) = \begin{bmatrix} C(\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_{-k}) & C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})' \\ C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k}) & C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_k) \end{bmatrix}, \tag{3.12}$$

e seu determinante pode ser obtido usando uma identidade de Schur (detalhes em (A.2) no Apêndice A). Dessa forma, além da distribuição *a priori* do vetor $\boldsymbol{\mu}$, pode-se considerar a distribuição *a priori* condicional de uma de suas componentes, $\boldsymbol{\mu}_k$, dadas as demais, $\boldsymbol{\mu}_{-k} = \{\boldsymbol{\mu}_j\}_{j\neq k}$.

Para $\boldsymbol{\mu}_k$, o k-ésimo componente do vetor $\boldsymbol{\mu}$, o núcleo da sua distribuição condicional é

$$p(\boldsymbol{\mu}_k|\boldsymbol{\mu}_{-k}, \theta, \sigma_q, K) \propto C_{\mu_k} - C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})C_{\mu_{-k}}^{-1}C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})', \tag{3.13}$$

em que $C_{\mu_k} = C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_k)$ é um escalar, $C_{\mu_{-k}} = C(\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_{-k})$ é matriz $(K-1) \times (K-1)$ e $C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})$ é vetor $1 \times (K-1)$ composto por $C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_j)$, $j \neq k$.

Segundo [8], o parâmetro θ^2 controla a repulsão em um PPD, determinando uma distância mínima para que os pontos sejam válidos como realização do processo. A partir disto, os autores apresentaram uma limitação para o caso em que a função intensidade é constante garantindo assim o que chamam de validade do PPD. No entanto, o PPD com kernel Gaussiano definido em 2.3.1 não possui intensidade constante.

Uma das contribuições desta tese é trazer para o contexto do PPD com kernel Gaussiano a garantia de equilíbrio entre a intensidade de ocorrência de pontos que é influenciada diretamente pelo parâmetro σ_q^2 (função qualidade) e o intervalo de repulsão controlado pelo parâmetro θ^2 (função similaridade).

Dessa forma, uma condição que garante a validade do PPD para o kernel utilizado no modelo (3.2)-(3.8), considerando uma função qualidade nos termos de (2.8) através de sua função intensidade $\rho(\boldsymbol{\mu}) = q^2(\boldsymbol{\mu})$ é fornecida na Proposição 3.2.1.

Proposição 3.2.1. *Restrição do Kernel Gaussiano.*

A existência de um PPD válido é garantida se $[\rho(t)] \leq 1/(\pi\theta^2), \forall\ t \in B \subseteq \mathbb{R}^D$. *Para o modelo proposto em* (3.2) - (3.8) *com o kernel exponencial quadrático a restrição é dada por*

$$0 < \theta^2 \leq (2\sigma_q^2)^D \pi^{D-1}.$$

A Proposição 3.2.1 foi desenvolvida pela autora, bem como sua prova presente no Apêndice A.

Em particular, para os casos de dados unidimensional e bidimensional, pode ser mostrado que

- se $D = 1$, a restrição será dada por $0 < \theta^2 \leq 2\sigma_q^2$,
- se $D = 2$, a restrição será dada por $0 < \theta^2 \leq 4\sigma_q^4\pi$.

Pela Proposição 3.2.1 fica claro que θ^2 e σ_q^2 não podem ser considerados independentes. Devido à restrição, os parâmetros do kernel são considerados dependentes *a priori* sendo modelados a partir das distribuições Gama Truncada e Gama, respectivamente:

$$\theta^2|\sigma_q^2 \sim Gama_T(a_1, b_1, t_1), \theta^2 \in t_1 = \left(0; (2\sigma^2)^D\pi^{D-1}\right) \tag{3.14}$$

e

$$\sigma_q^2 \sim Gama(a_2, b_2), \sigma_q^2 > 0, \tag{3.15}$$

com $t_1 = \left(0; (2\sigma^2)^D\pi^{D-1}\right)$ sendo o intervalo em que $p(\theta^2|\sigma_q^2) > 0$. Assim, a densidade de θ^2 é dada por

$$p(\theta^2|\sigma_q^2) = \frac{\theta^{2a_1-1}e^{-b_1\theta^2}}{\Gamma_I(a_1, b_1, t_1)}, 0 < \theta^2 < (2\sigma_q^2)^D\pi^{D-1},$$

em que $\Gamma_I(\alpha, \beta, x) = \int_{\{\theta \in x\}} \theta^{\alpha-1}e^{-\beta\theta}d\theta$ é chamada Função Gama Incompleta Inferior. E a densidade de σ_q^2 é dada por $p(\sigma_q^2) = \frac{b_2^{a_2}}{\Gamma(a_2)}\sigma_q^{a_2-1}e^{-b_2\sigma_q^2}, \sigma_q^2 > 0$, ou seja, $E(\sigma_q^2) = \frac{a_2}{b_2}$.

Definidas as distribuições a priori, passaremos a determinação das distribuições condicionais completas essenciais a inferência do modelo.

3.3 Estrutura Bayesiana para o modelo de mistura NIPPD

Inicialmente temos a função de verossimilhança de $\mathbf{y}_i|z_i = k, \boldsymbol{\mu}_k, \mathcal{T}_k, K \ldots \sim N_D(\boldsymbol{\mu}_k, u_i^{-1}E\boldsymbol{\mathcal{T}}_kE')$ dada por

$$\begin{aligned} L(\mathbf{z}, \boldsymbol{\mu}, \mathbf{u}, \boldsymbol{\mathcal{T}}, K, \ldots) &= \prod_{k=1}^{K}\prod_{i:z_i=k} \det\left(u_i^{-1}(E\boldsymbol{\mathcal{T}}_kE')\right)^{-1/2}(2\pi)^{-D/2} \\ &\times \exp\Big\{-\frac{u_i}{2}(\mathbf{y}_i - \boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_kE')^{-1}(\mathbf{y}_i - \boldsymbol{\mu}_k)\Big\}. \end{aligned} \tag{3.16}$$

Para a partição $\mathbf{z} = (z_1, \ldots, z_N)$ a distribuição *a priori* e *a posteriori* são discretas. As componentes $z_i, i = 1, \ldots, N$ são consideradas independentes e sua distribuição é obtida diretamente pela normalização do núcleo abaixo

$$\begin{aligned} p(z_i = k|\ldots) &\propto p(z_i = k)L(\mathbf{z}, \boldsymbol{\mu}, u_i, \boldsymbol{\Sigma}, K, \ldots) \\ &\propto \frac{w_k}{(2\pi)^{n_k D/2}} u_i^{-D/2} det(E\boldsymbol{\mathcal{T}}_k E')^{-1/2} \\ &\times \exp\Big\{ -\frac{u_i}{2}(\mathbf{y}_i - \boldsymbol{\mu}_k)'\boldsymbol{\Sigma}_k^{-1}(\mathbf{y}_i - \boldsymbol{\mu}_k)\Big\}. \end{aligned} \tag{3.17}$$

A constante de normalização é obtida pela soma das quantidades acima para todo $k = 1, \ldots, K$ para cada um dos indivíduos.

Para os pesos $\mathbf{w}$ a distribuição Dirichlet é de família conjugada à considerada na função de verossimilhança, assim a sua distribuição *a posteriori* é dada por

$$p(\mathbf{w}|\ldots) \sim Dir(\delta_1 + n_1, \ldots, \delta_k + n_K), \text{ em que } n_k = \sum_{i=1}^{N} I_{(z_i=k)}. \tag{3.18}$$

Os autovalores da matriz escala possuem distribuição Gama *a priori* que é de família conjugada à Normal na verossimilhança. Assim,

$$p(\tau_{kd}^{-1}|\ldots) \propto \tau_{kd}^{-(a_0/2-1)-n_k/2} \exp\left\{ -(b_0/2)\tau_{kd}^{-1} - \frac{e_d' \sum_{i:z_i=k} u_i(\mathbf{y}_i - \boldsymbol{\mu}_k)'(\mathbf{y}_i - \boldsymbol{\mu}_k)e_d}{2}\tau_{kd}^{-1}\right\}.$$

Logo,

$$\tau_{kd}^{-1}|\ldots \sim Gama(\frac{a_0 + n_k}{2}, \frac{b_0 + e_d' S_k e_d}{2}), \text{ em que } S_k = \sum_{i:z_i=k} u_i(\mathbf{y}_i - \boldsymbol{\mu}_k)(\mathbf{y}_i - \boldsymbol{\mu}_k)'. \tag{3.19}$$

Para as variávies misturadoras, $u_1, \ldots, u_N$, a distribuição *a priori* é da distribuição Gama que é conjugada a família Normal na verossimilhança

$$p(u_i|\ldots) \propto u_i^{(\eta/2-1)} e^{-u_i\eta/2}\, u_i^{D/2} \exp\Big\{ -\frac{u_i}{2}(\mathbf{y}_i - \boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_k E')^{-1}(\mathbf{y}_i - \boldsymbol{\mu}_k)\Big\}.$$

Logo, a distribuição condicional completa de u_i , $i = 1, \ldots, n$, é dada por

$$p(u_i|\ldots) \sim Gama\left(\frac{\eta + D}{2}, \frac{\eta + S_i}{2}\right), \tag{3.20}$$

em que $S_i = (\mathbf{y}_i - \boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_k E')^{-1}(\mathbf{y}_i - \boldsymbol{\mu}_k)$.

Com respeito ao vetor de parâmetros de locação $\boldsymbol{\mu}$, sua condicional completa é desconhecida. Assim, é necessário utilizar o algoritmo de Metropolis-Hastings para amostragem

indireta desta distribuição. O núcleo da condicional foi obtido e é apresentado abaixo.

$$\begin{aligned} p(\boldsymbol{\mu}|\ldots) &\propto \det(C_{\theta,\sigma_q}(\boldsymbol{\mu}))L(\mathbf{z},\boldsymbol{\mu},\mathbf{u},\boldsymbol{\Sigma},K,\ldots) \\ &\propto \det(C_{\theta,\sigma_q}(\boldsymbol{\mu}))\prod_{k=1}^{K}\exp\Big\{-\frac{1}{2}\big(\boldsymbol{\mu}_k - M_k\big)' V_k^{-1}\big(\boldsymbol{\mu}_k - M_k\big)\Big\}, \end{aligned} \tag{3.21}$$

em que $M_k = [\sum_{i:z_i=k} u_i]^{-1}(\sum_{i:z_i=k} u_i \mathbf{y}_i)$ e $V_k = [\sum_{i:z_i=k} u_i]^{-1}(E\,\boldsymbol{\mathcal{T}}_k E)$. Neste caso, a verossimilhança se torna o núcleo de uma normal D-dimensional com média M_k e matriz de variância V_k, $\boldsymbol{\mu}|\ldots \sim N_D(M_k, V_k)$.

Assim como na distribuição *a priori*, a identidade de Schur pode ser usada sobre a partição (3.12) da matriz $C_{\theta,\sigma_q}(\boldsymbol{\mu})$ para decompor o determinante que aparece em (3.21) $p(\boldsymbol{\mu}|\ldots)$ considerando $A_{22} = C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_k)$. A menos das constantes que não dependem do k-ésimo componente do vetor de médias, o núcleo da distribuição de probabilidade condicional de $\boldsymbol{\mu}_k$, $k = 1,\ldots,K$, é dada por

$$\begin{aligned} p(\boldsymbol{\mu}_k|\ldots) &\propto \left(C_{\boldsymbol{\mu}_k} - C(\boldsymbol{\mu}_k,\boldsymbol{\mu}_{-k})C_{\boldsymbol{\mu}_{-k}}^{-1}C(\boldsymbol{\mu}_k,\boldsymbol{\mu}_{-k})'\right) \\ &\times \exp\Big\{-\frac{1}{2}\big(\boldsymbol{\mu}_k - M_k\big)' V_k^{-1}\big(\boldsymbol{\mu}_k - M_k\big)\Big\}, \end{aligned} \tag{3.22}$$

em que $C_{\boldsymbol{\mu}_k} = C(\boldsymbol{\mu}_k,\boldsymbol{\mu}_k)$ e $C_{\boldsymbol{\mu}_{-k}} = C(\boldsymbol{\mu}_{-k},\boldsymbol{\mu}_{-k})$, com $\boldsymbol{\mu}_{-k} = \{\boldsymbol{\mu}_j\}_{j\neq k}$.

Para os parâmetros do kernel as distribuições condicionais completas são dadas por

$$p(\theta^2|\ldots) \propto \frac{\det(C_{\mu,\theta,\sigma_q})}{\prod_{h=1}^{\infty}(\lambda_h(\theta^2,\sigma_q^2)+1)}p(\theta^2|\sigma_q^2), \tag{3.23}$$

em que $\theta^2|\sigma_q^2 \sim Gama_T(a_1,b_1,t_1), \theta^2 \in t_1 = \left(0, \frac{(2\pi\sigma^2)^{D/2}}{\sqrt{\pi}}\right)$ e

$$p(\sigma_q^2|\ldots) \propto \frac{\det(C_{\boldsymbol{\mu},v,\theta,\sigma_q})}{\prod_{h=1}^{\infty}(\lambda_h(\theta^2,\sigma_q^2)+1)}p(\sigma_q^2|\theta^2), \tag{3.24}$$

em que $\sigma_q^2 \sim Gama_T(a_2,b_2,t_2), \sigma_q^2 \in t_2 = \left(\frac{(\theta\sqrt{\pi})^{1/D}}{\sqrt{2\pi}}, \infty\right)$. Assim como a distribuição conjunta é dada por

$$\begin{aligned} p(\theta^2,\sigma_q^2|\ldots) &= p(\theta^2|\sigma_q^2,\ldots)p(\sigma_q^2|\ldots) \\ &\propto \frac{\det(C_{\mu,\theta^2,\sigma_q^2})}{\prod_{h=1}^{\infty}(\lambda_h(\theta^2,\sigma_q^2)+1)}p(\theta^2|\sigma_q^2)p(\sigma_q^2). \end{aligned} \tag{3.25}$$

Para a maioria dos parâmetros é possível amostrar diretamente das condicionais completas. A exceção são os parâmetros de locação e do kernel que terão amostragem realizada via Metropolis-Hastings com a distribuição normal como geradora de candidatas. A amostragem dos parâmetros do kernel será realizada conjuntamente em bloco devido a sua correlação. No que concerne à implementação computacional do modelo proposto, os detalhes serão abordados a seguir na Subseção 3.4. Detalhes sobre a obtenção das distribuições condicionais completas são fornecidos no Apêndice A.

3.4 Algoritmo MCMC para o MNIPPD

O algoritmo MCMC para estimação dos parâmetros do Modelo de Mistura NIPPD é composto de passos de Metropolis-Hastings [53], para o parâmetro de locação e para os parâmetros do Kernel, e passos de Amostrador de Gibbs [45] para os demais parâmetros. É considerado um valor máximo para o número de grupos, K_{max} para o Modelo de Mistura Finita. No entanto, dependendo do processo de alocação, controlado pelo parâmetro $\mathbf{z}$, alguns grupos podem estar vazios em um conjunto de iterações. O que é perfeitamente natural, afinal o verdadeiro número de grupos deve ser alcançado pelo modelo e espera-se que seja menor que K_{max}. A amostra *a posteriori* para K, é dada pelo número de grupos não vazios nas iterações. Desta forma, mesmo para os grupos vazios, são gerados valores para todos os parâmetros a partir das distribuições a priori, possibilitando assim que indivíduos sejam alocados nestes grupos em outras iterações.

No algoritmo será utilizado $K_{max} = 2\log(N)$, N número de observações. O valor de $K_{max} = N$ seria o melhor, mas aumenta consideravelmente o custo computacional e pode gerar erros numéricos. A função log diminui o valor de K_{max} em relação a N consideravelmente, e o dobro desse valor é uma alternativa mais conservadora. No entanto, se um ajuste inicial mostrar muitos valores próximos ao máximo este pode ser aumentado para garantir a eficácia na estimação de K.

Como muitos modelos de mistura este sofre com a troca de rótulos dos *clusters* gerados, problema conhecido como *label switching*. Para verificar a convergência das cadeias a ordenação das medidas de locação foi utilizada. Mas para a verificação da eficiência da alocação foram construídas cadeias para cada indivíduo utilizando os z_i's e as estimativas em cada iteração.

Nesse trabalho η é considerado um hiperparâmetro, no entanto, uma análise de sensibilidade é construída para a escolha do melhor modelo segundo seu valor. Os valores utilizados foram:

- $\eta = 2{,}1$; que é um menor valor para que exista média e variância no modelo t de Student;
- $\eta = 5$; valor que ainda fornece caudas pesadas;
- $\eta = 100$; uma aproximação da distribuição normal.

O algoritmo do MCMC consiste nos seguintes passos:

1. Inicialização das cadeias:
   a) $K_{max} \approx 2\lceil log(N) \rceil$;

b) Cálculo da matriz E, a partir da decomposição SVD da matriz de covariância dos dados: $\mathbf{S} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{Y}_i - \bar{\mathbf{Y}})^\top(\mathbf{Y}_i - \bar{\mathbf{Y}})$;

c) Utilizar o método k-médias para fornecer valores iniciais de $\mathbf{z}^{(0)}$, $\mathbf{w}^{(0)}, \boldsymbol{\mu}^{(0)}$ serão os centroides dos grupos;

d) $\mathbf{u}^{(0)} = (1, 1, \ldots, 1)$, $\boldsymbol{\mathcal{T}}^{(0)} = (1, 1, \ldots, 1)$,

e) $\theta^{2(0)}$, $\sigma_q^{2(0)}$, consistentes com a distribuição *a priori* ou de acordo com seus estimadores de momento:

$\theta^{2(0)}$: mediana da distância Euclidiana dos dados e

$\sigma_q^{2(0)}$: número de observações por unidade de volume ocupada pelos dados.

2. Atualização da alocação, $z_i^{(1)}$, $i = 1, \ldots, N$ via distribuição condicional completa (3.17);

a) Obter a distribuição de $\mathbf{z}$ para cada indivíduo, $P(z_i = k), k = 1, \ldots, K$, a partir da condicional completa em (3.17).

b) Amostrar da distribuição do item (a), uma alocação para cada indivíduo armazenado no vetor $\mathbf{z}$.

3. Atualização do vetor de pesos, $\mathbf{w}$, via distribuição condicional completa em (3.18);

a) Calcular tamanho dos grupos, $n_1, \ldots, n_K$, a partir de $\mathbf{z}$;

b) Amostrar da distribuição condicional completa (3.18), atualizada pelo tamanho dos grupos;

c) Grupos vazios possuem $n_k = 0$, assim seus pesos são gerados como os demais componentes na distribuição Dirichlet, mas com os parâmetros definidos a priori.

4. Atualização da matriz escala $\boldsymbol{\Sigma}_k = E\boldsymbol{\mathcal{T}}E'$.

a) Atualizar os autovalores da matriz escala, $\tau_{kd}^{(1)}$, $k = 1, \ldots, K$ e $d = 1, \ldots, D$, via distribuição condicional completa em (3.19);

b) Calcular a matriz escala a partir de $\Sigma_k = E\boldsymbol{\mathcal{T}}_k E'$.

5. Atualização da variável misturadora, $\mathbf{u}$,

a) Calcular $S_i = (\mathbf{y}_i - \boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_k E')^{-1}(\mathbf{y}_i - \boldsymbol{\mu}_k)$ para cada indivíduo;

b) Amostrar da distribuição em (3.20) para cada indivíduo.

6. Atualização dos componentes do vetor de locação, $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K)$, na m-ésima iteração.

a) Obter um valor proposto $\boldsymbol{\mu}_k^{(n)}$, novo, para $k = 1$, a partir de uma distribuição $N_D(\boldsymbol{\mu}_k^{(m-1)}, \sigma_{pro}^2\boldsymbol{I})$, em que $\boldsymbol{\mu}_k^{(o)}$ é o valor do componente na $(m-1)$-ésima iteração, antigo, σ_{pro}^2 é a variância da proposta.

b) Calcular a probabilidade de aceitação do valor proposto

i. Se o grupo k está vazio, $n_k = 0$, então a probabilidade de aceitação é baseada no núcleo da distribuição *a priori* dada em (3.13);

$$\rho(\boldsymbol{\mu}_k^{(o)}, \boldsymbol{\mu}_k^{(n)}) = \frac{C_{\boldsymbol{\mu}_k^{(n)}} - C(\boldsymbol{\mu}_k^{(n)}, \boldsymbol{\mu}_{-k}) C_{\boldsymbol{\mu}_{-k}}^{-1} C(\boldsymbol{\mu}_k^{(n)}, \boldsymbol{\mu}_{-k})'}{C_{\boldsymbol{\mu}_k^{(o)}} - C(\boldsymbol{\mu}_k^{(o)}, \boldsymbol{\mu}_{-k}) C_{\boldsymbol{\mu}_{-k}}^{-1} C(\boldsymbol{\mu}_k^{(o)}, \boldsymbol{\mu}_{-k})'}$$

ii. Se o grupo k é não vazio, $n_k > 0$, então a probabilidade de aceitação é baseada no núcleo da distribuição condicional completa *a posteriori* em (3.22):

$$\rho(\boldsymbol{\mu}_k^{(o)}, \boldsymbol{\mu}_k^{(n)}) = \frac{C_{\boldsymbol{\mu}_k^{(n)}} - C(\boldsymbol{\mu}_k^{(n)}, \boldsymbol{\mu}_{-k}) C_{\boldsymbol{\mu}_{-k}}^{-1} C(\boldsymbol{\mu}_k^{(n)}, \boldsymbol{\mu}_{-k})'}{C_{\boldsymbol{\mu}_k^{(o)}} - C(\boldsymbol{\mu}_k^{(o)}, \boldsymbol{\mu}_{-k}) C_{\boldsymbol{\mu}_{-k}}^{-1} C(\boldsymbol{\mu}_k^{(o)}, \boldsymbol{\mu}_{-k})'} lr$$

em que $lr = \frac{l(\boldsymbol{\mu}_k^{(n)})}{l(\boldsymbol{\mu}_k^{(o)})}$ é a razão das verossimilhanças calculadas no valor proposto e no valor da $(m-1)$-ésima iteração.

c) Amostrar p de uma distribuição $Unif(0,1)$ e se $p < min\{1, \rho(\boldsymbol{\mu}_k^{(o)}, \boldsymbol{\mu}_k^{(n)})\}$ aceita-se $\boldsymbol{\mu}_k^{(n)}$ como valor atual da cadeia, caso contrário, repete-se o valor anterior na iteração atual.

d) Repetir 6.(a) - 6.(c) para $k = 2, \ldots, K$.

7. Atualização em bloco dos componentes do kernel do PPD, $\theta^{2(m)}$ e $\sigma_q^{2(m)}$, na m-ésima iteração.

a) Obter valores propostos $\theta^{2(n)}$ e $\sigma_q^{2(n)}$, novos, a partir das distribuições $\theta^2_{prop} \sim N(\theta^{2(m-1)}, v_1)$ e $\sigma^2_{q(prop)} \sim N(\sigma_q^{2(m-1)}, v_2)$, em que v_1 e v_2 são as variâncias das propostas, $\theta^{2(m-1)}$ e $\sigma_q^{2(m-1)}$ são os valores das cadeias de θ^2 e de σ_q^2, respectivamente, na $(m-1)$-ésima iteração.

b) Calcular a probabilidade de aceitação dos valores propostos com base na distribuição conjunta do modelo (3.10) e $C(\cdot)$, matriz kernel calculada para os valores atuais do parâmetro de locação, $\boldsymbol{\mu}$. Para o bloco $(\theta^2\sigma_q^2)$, calcula-se

$$\begin{aligned}\rho((\sigma_q^{2(o)}, \theta_q^{2(o)}), (\sigma_q^{2(n)}, \theta_q^{2(n)})) &= \frac{det[C(\boldsymbol{\mu}, \theta^{2(n)}, \sigma_q^{2(n)})]}{det[C(\boldsymbol{\mu}, \theta^{2(o)}, \sigma_q^{2(o)})]} \frac{\prod_{h=1}^{\infty}(\lambda_h(\theta^{2(o)}, \sigma_q^{2(o)}) + 1)}{\prod_{h=1}^{\infty}(\lambda_h(\theta^{2(n)}, \sigma_q^{2(n)}) + 1)} \\ &\times \frac{p(\theta^{2(n)}, \sigma_q^{2(n)})}{p(\theta^{2(o)}, \sigma_q^{2(o)})} \frac{q(\theta^2, \sigma_q^2)}{q(\theta^2, \sigma_q^2)},\end{aligned}$$

em que $\frac{p(\theta^{2(n)}, \sigma_q^{2(n)})}{p(\theta^{2(o)}, \sigma_q^{2(o)})}$ é a razão das prioris conjuntas dadas em (3.25) calculadas nos valores propostos e nos valores da $(m-1)$-ésima iteração. Enquanto $\frac{q(\theta^2, \sigma_q^2)}{q(\theta^2, \sigma_q^2)}$ é a razão das distribuições propostas utilizadas para gerar valores de (θ^2, σ_q^2), neste caso distribuições normais truncadas no zero.

c) Amostrar p de uma distribuição $Unif(0,1)$ e se $p < min\{1, \rho((\sigma_q^{2(o)}, \theta_q^{2(o)}), (\sigma_q^{2(n)}, \theta_q^{2(n)}))$ aceita-se o bloco $\theta^{2(n)}$ e $\sigma_q^{2(n)}$ como valores das cadeias, caso contrário, repetem-se os valores anteriores na iteração atual.

8. Repetir os passos $1-7$ até obter m iterações.

O algoritmo foi implemenado no R [91] e as rotinas computacionais estão disponíveis no Apêndice B. Nas Seções 3.5 e 3.6, o algoritmo é utilizado para o ajuste do Modelo de Mistura NIPPD, para estimação de densidade em dados simulados e em uma aplicação real em dados de plantio de banana, respectivamente.

3.5 Dados Simulados

Foram considerados conjuntos de dados unidimensionais, $D = 1$, simulando quatro cenários distintos com 300 indivíduos cada:

a) Cenário base: médias separadas, variâncias constantes e pesos próximos;

b) Pesos diferentes, demais características do cenário base mantidas;

c) Duas medidas de locação próximas, demais características do cenário base mantidas, e

d) Variâncias diferentes, demais características do cenário base mantidas.

O cenário a) foi considerado base para os demais sendo construído com $K = 3$ grupos, pesos $\mathbf{w}$ =(0,30; 0,35; 0,35), $\boldsymbol{\mu} = (18, 35, 55)$, $u_i \sim Gama(5/2, 5/2)$, ou 5 graus de liberdade, σ^2 = (1,2; 1,2; 1,2).

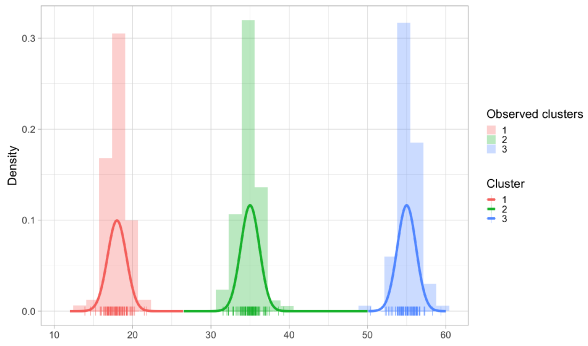
Tabela 3.1: Cenários simulados.

Cenário	$\boldsymbol{\mu}$	$\boldsymbol{\sigma}^2$	$\mathbf{w}$	$\mathbf{u}$	N
a	(18, 35, 55)	(1,2; 1,2; 1,2)	(0,30; 0,35; 0,35)	Gama(5/2, 5/2)	(97; 102, 101)
b	(18, 35, 55)	(1,2; 1,2; 1,2)	**(0,5; 0,3; 0,2)**	Gama(5/2, 5/2)	**(152; 95, 53)**
c	**(18, 25, 55)**	(1,2; 1,2; 1,2)	(0,30; 0,35; 0,35)	Gama(5/2, 5/2)	(97; 102, 101)
d	(18, 35, 55)	**(5,0; 7,0; 15,5)**	(0,30; 0,35; 0,35)	Gama(5/2, 5/2)	(97; 102, 101)

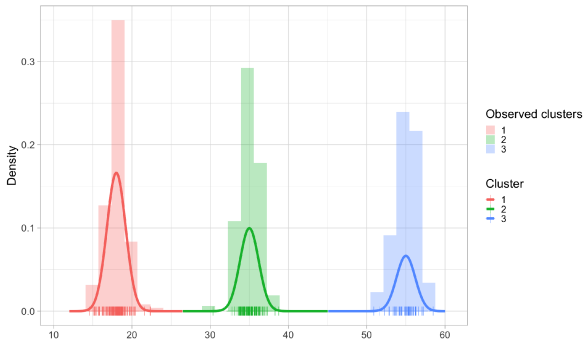
As cadeias do MCMC foram obtidas com 50000 iterações, aquecimento de 10000, saltos de 20 observações resultando em uma amostra de tamanho 2000. Para as distribuições *a priori* foram considerados os seguintes valores de hiperparâmetros: $\delta = 1$, equivalente à distribuição uniforme *a priori* para os pesos das componentes, $a_0 = b_0 = 0{,}01$, equivalente a uma priori vaga (sem média ou variância finita) para os parâmetros de escala $\sigma_k^2, k = 1, \ldots, K$, $K_{max} = 6$

Figura 3.2: Histogramas e funções densidade dos cenários simulados pelos dados artificiais. As observações são representadas pelos traços verticais (|) abaixo, no eixo horizontal.

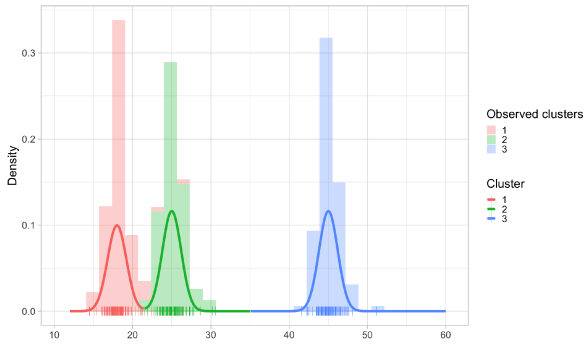
(a) Cenário base.



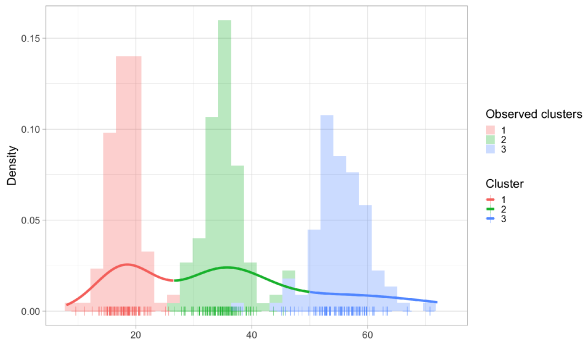
(b) Cenário com pesos diferentes.



(c) Cenário com medidas de locação próximas.



(d) Cenário com variância diferentes.



Fonte: Elaborado pela autora.

e $\eta = 5$. A convergência foi verificada por teste de Geweke e função de autocorrelação, os gráficos relacionados bem como as taxas de aceitação são apresentadas no Apendice B.

Para a estimativa da alocação dos *clusters*, a matriz de similaridade foi computada a partir do modelo NIPPD através de $\mathbf{z}$. Esta matriz estima as probabilidades de dois indivíduos quaisquer do conjunto de dados estarem no mesmo *cluster* [41]. A partir da matriz de similaridade, foram calculadas três funções de perda recomendadas na literatura para determinar a melhor partição dos efeitos dos cursos: a Perda Variação de Informação (VI) [78], a Perda de Binder N-invariante (B) [14] e a Perda omARI (One minus Adjusted Rand Index), denotada neste trabalho por ARI [114].

Sejam $\mathbf{z}$ e $\hat{\mathbf{z}}$ duas configurações de alocação, ou seja, $\mathbf{z} = (z_1, \ldots, z_N)$, tal que $z_i = k$ indica que o indivíduo i foi alocado no *cluster* k. Além disso, defina K e $\hat{K}$ o número de *clusters* em $\mathbf{z}$ e $\hat{\mathbf{z}}$, respectivamente, e $n_{i,j}$ o número de indivíduos no grupo i da configuração $\mathbf{z}$ que está no grupo j da configuração $\hat{\mathbf{z}}$, e n_{+j} o número de indivíduos que estão no grupo j da configuração $\hat{\mathbf{z}}$, com $i = 1, \ldots, K$ e $j = 1, \ldots, \hat{K}$..

Perda de Binder N-invariante

$$\bar{B}(\mathbf{z},\hat{\mathbf{z}}) \quad = \quad \sum_{i=1}^{K}\left(\frac{n_{i+}}{N}\right)^2 + \sum_{j=1}^{\hat{k}_N}\left(\frac{n_{+j}}{N}\right)^2 - 2\sum_{i=1}^{K}\sum_{j=1}^{\hat{K}}\left(\frac{\hat{n}_{ij}}{N}\right)^2. \tag{3.26}$$

Indice de Rand (1971):

$$R(\mathbf{z},\hat{\mathbf{z}}) = \frac{A}{A+D},$$

em que A é o número de concordâncias entre pares de observações ($A + D = \binom{N}{2}$). Já o Índice de Rand Ajustado (*Adjusted Rand Index*) é dado por $ARI(\mathbf{z},\hat{\mathbf{z}}) = \dfrac{R - E[R]}{max[R] - E[R]}$,

Perda Variação de Informação (VI)

$$VI(\mathbf{z},\hat{\mathbf{z}}) = H(\mathbf{z}) + H(\hat{\mathbf{z}}) - 2I(\mathbf{z},\hat{\mathbf{z}}), \tag{3.27}$$

em que $H(\mathbf{z})$ é a medida de entropia para a configuracção $\mathbf{z}$ e $I(\mathbf{z},\hat{\mathbf{z}})$ uma medida de informação mútua (entropia relativa ou ainda a Divergência de Kullback-Leibler do produto das distribuições marginais para a distribuição conjunta) entre as duas configurações de clusters $\mathbf{z}$ e $\mathbf{z}'$ e são dadas por $I(\mathbf{z},\hat{\mathbf{z}}) = \sum_{j=1}^{k_N}\sum_{j'=1}^{k'_N}\frac{n_{jj'}}{N}\log\left(\frac{n_{jj'}N}{n_{j+}n_{+j}}\right)$ e $H(\mathbf{z}) = -\sum_{i=1}^{k_N}\frac{n_{i+}}{N}\log\left(\frac{n_{i+}}{N}\right)$ com $\log = log_2$.

As perdas de Binder e ARI estão relacionadas, sendo que a ARI possui uma correção que compensa alocações completamente por acaso, enquanto a perda de Binder permite penalizações em diferentes erros de alocação. A perda VI, baseada na teoria da informação, favorece alocações com menor entropia, resultando em clusters estimados mais homogêneos (mais detalhes em [114]).

3.5.1 Resultados

A Tabela 3.2 mostra o número de *clusters* estimado pela moda da distribuição *a posteriori* para os cenários simulados, a probabilidade associada a esse valor e o intervalo *Highest Posterior Density* (HPD) de 95% de probabilidade, calculado considerando os valores de K com as maiores probabilidade em ordem decrescente. Em todos os casos o número verdadeiro de *clusters* foi recuperado pelo modelo e os intervalos são precisos. Quanto às alocações, a tabela também apresenta um número muito baixo de indivíduos classificados incorretamente, sendo no máximo 4 considerando a perda VI e, no máximo 5, considerando todas as perdas.

Para o Cenário base (a) e o Cenário (b), que foi simulado considerando pesos diferentes, as alocações tiveram 100% de acerto usando qualquer uma das funções de perda: VI, Binder e ARI. Na alocação dos indivíduos para o Cenários (c), que considera duas médias próximas e para o Cenário (d), que considera variâncias diferentes ocorreram 3 e 4 erros, respectivamente,

Tabela 3.2: Número de alocações incorretas segundo a perda VI, e estatísticas *a posteriori* para o número de *clusters*.

Cenário	Alocação Incorreta			Moda	HPD 95%	
	VI	Binder	ARI	(prob.)	Inf	Sup
(a)	0	0	0	3 (0,63)	3	4
(b)	0	0	0	3 (0,99)	3	3
(c)	2	3	2	3 (0,99)	3	3
(d)	4	5	4	3 (0,99)	3	3

utilizando a perda VI. Ainda no Cenário (c) com duas médias próximas, a perda de Binder apresentou 3 erros de alocação dos indivíduos e a perda ARI apresentou 2 erros.

O cenário (d) apresentou o maior número de alocações incorretas, mesmo assim, um número bem pequeno correspondendo a no máximo 1,7% das observações. A Tabela 3.3 mostra as alocações segundo as três perdas e a comparação com os verdadeiros *clusters* para o Cenário (d). A perda de Binder apresentou o maior número de alocações incorretas, 5 indivíduos, e a perda ARI alocou incorretamente 4 indivíduos.

Tabela 3.3: Avaliação da alocação das observações nos *clusters* para o cenário (d).

	Alocado									
	VI			Binder				ARI		
Verdadeirio	1	2	3	1	2	3	4	1	2	3
1	96	1	0	95	0	0	2	96	1	0
2	0	100	2	0	100	2	0	0	100	2
3	0	1	100	0	1	100	0	0	1	100

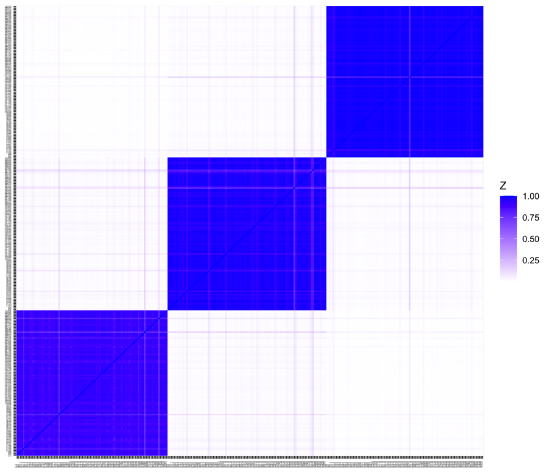
As matrizes de similaridade estimadas pelo modelo são representadas na Figura 3.3 pelos gráficos de calor. Os eixos horizontal e vertical representam os indivíduos ordenados segundo a alocação produzida pela Perda VI. Um padrão bloco-diagonal representa uma alocação bem definida, esse padrão está presente nos gráficos de todos os cenários. O gráfico é formado por pequenos quadriculados e a intensidade da sua cor é proporcional a probabilidade da dupla de indivíduos associados (na linha e na coluna) estarem no mesmo *cluster*, cores mais escuras representam maiores probabilidades. Como em outros modelos de mistura, é possível ocorrer troca de rótulos (*label switching*) entre os *clusters* estimados e os *clusters* reais. As legendas das Figuras 3.3a, 3.3b, 3.3c e 3.3d especificam os rótulos dos blocos diagonais em relação aos verdadeiros componentes simulados pelos cenários.

Para o Cenário base (a), Figura 3.3a, as alocações estão bem definidas como mostra o padrão bloco-diagonal. A alocação estimada conta com altas probabilidades dentro dos *clusters*: no mínimo 0,687 entre os indivíduos do *cluster* 1, no mínimo 0,692 entre os indivíduos do *cluster* 2 e 0,746 entre os indivíduos do *cluster* 3. Enquanto as probabilidades entre os *clusters* são baixas, por exemplo, os indivíduos alocados no *cluster* 1, tem no máximo probabilidade de 0,193 de estar com as observações dos *clusters* 2 ou 3. Observações do *cluster* 2 tem no máximo 0,209 de probabilidade de estarem no mesmo *cluster* que observações alocadas nos

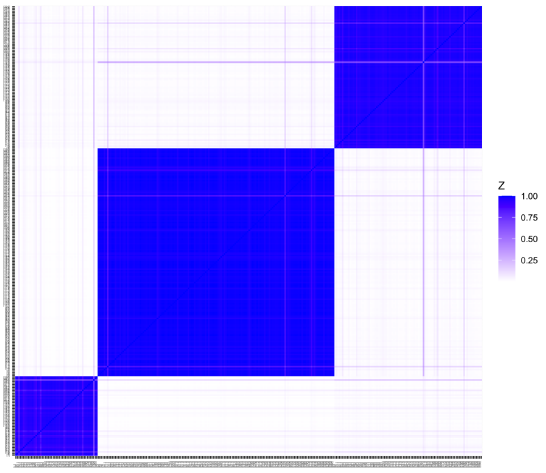
clusters 1 ou 3. O mesmo valor 0,209, foi estimado para as observações do *cluster* 3 de estarem no mesmo *cluster* que observações alocadas no *cluster* 1 ou 2.

Figura 3.3: Gráfico de calor da matriz de similaridade estimada pelo modelo para os dados de todos os cenários considerando a ordem de alocação da perda VI. Rótulos dos bloco-diagonais: Cenário base-Rótulos dos blocos-diagonais (de baixo para cima), correspondem aos componentes simulados com médias 18, 35 e 55; Cenário com pesos diferentes - de baixo para cima, correspondem aos componentes com médias 55, 18 e 35; Cenário com duas médias próximas: de baixo para cima, correspondem aos componentes com médias 55, 18 e 25, respectivamente; Cenário com variâncias diferentes - de baixo para cima, correspondem aos componentes com médias 55, 18 e 35, respectivamente.

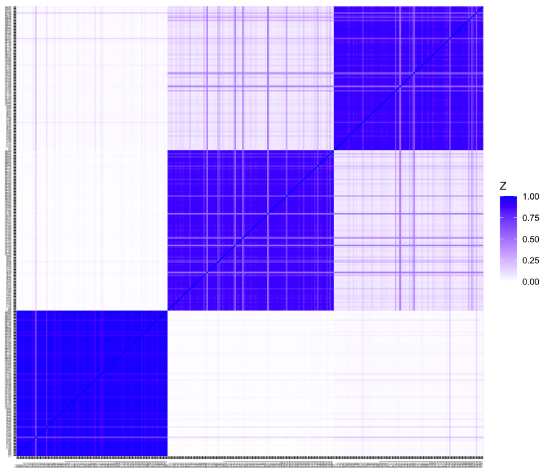
(a) Cenário base.



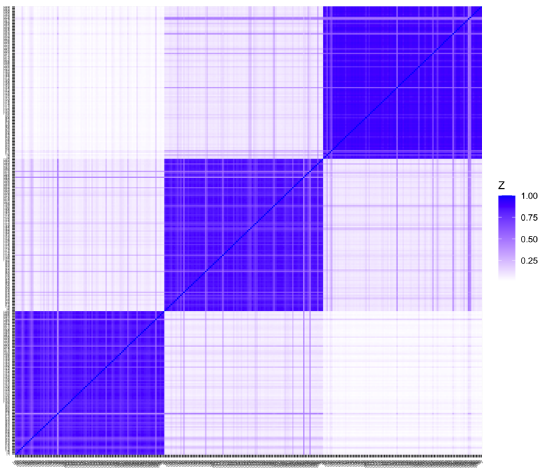
(b) Cenário com pesos diferentes.



(c) Cenário com duas médias próximas.



(d) Cenário com variâncias diferentes.



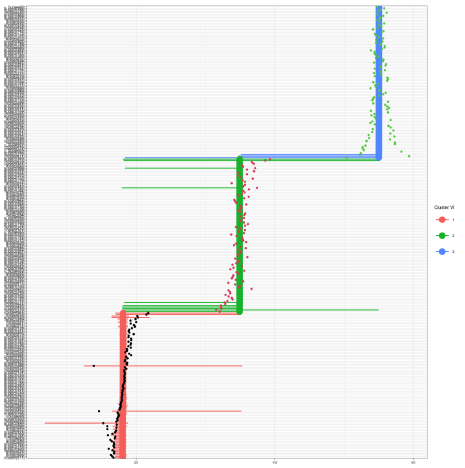
Fonte: Elaborado pela autora.

O Cenário (b), Figura 3.3b, também tem um padrão bem definido e os diferentes tamanhos dos blocos na diagonal refletem os diferentes pesos considerados na simulação. Possui probabilidades altas entre os indivíduos no mesmo *cluster*: no mínimo 0,617; 0,746 e 0,631 para os *clusters* 1, 2 e 3, respectivamente. Também possui probabilidades baixas entre os in-

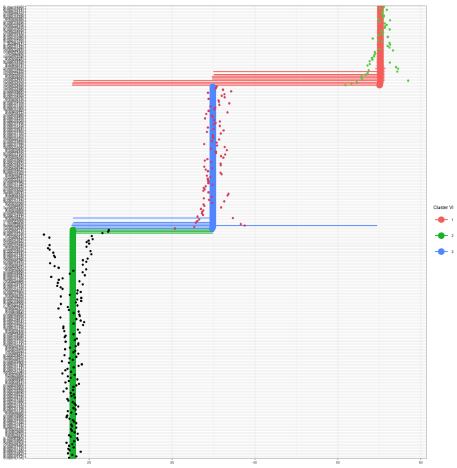
divíduos de *clusters* diferentes: no máximo 0,203; 0,315 e 0,315 para os indivíduos dos *clusters* 1, 2 e 3, respectivamente.

Figura 3.4: Medianas e Intervalos HPD de 95% de probabilidade (linhas horizontais) da distribuição *a posteriori* de $\boldsymbol{\mu}$. As cores dos intervalos representam a alocação estimada pela perda VI. Os pontos representam as respostas observadas e suas diferentes cores mostram os verdadeiros *clusters* a que pertencem.

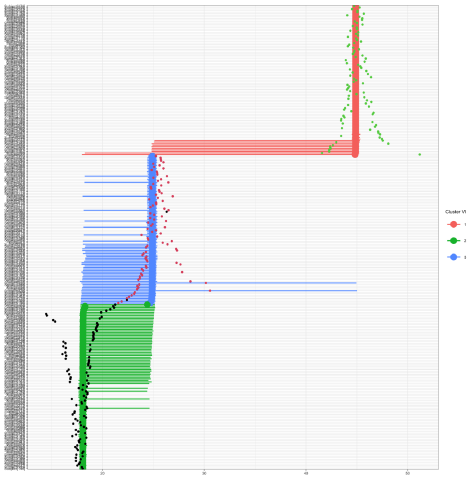
(a) Cenário base.



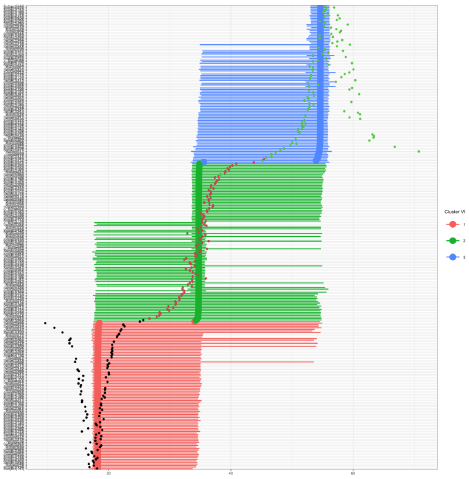
(b) Cenário com pesos diferentes.



(c) Cenário com duas médias próximas.



(d) Cenário com variâncias diferentes.



Fonte: Elaborado pelo autor(a).

Para o Cenário (c), a Figura 3.3c mostra padrão de alocação bem definido, mas as probabilidades de alocação entre os diferentes *clusters* estimados são maiores considerando aqueles simulados com médias próximas. As probabilidades dentro dos *clusters* estimados são: 0,642 para o *cluster* 1 que é relacionado ao componente de média 55, 0,487 para o *cluster* 2 que é relacionado ao componente com média 25 e 0,492 para o 3, sendo relacionado ao componente com média 18. As probabilidades entre observações de *clusters* diferentes são: no máximo 0,230 entre o *cluster* 1 e os demais, mas é no máximo 0,534 entre os *clusters* 2 e 3, que possuem médias próximas. Esse comportamento é esperado devido à característica dos dados com médias próximas, o que pode gerar algum confundimento na alocação dos *clusters*. Mesmo

assim foi possível diferenciar as observações e realizar alocação muito próxima à real. As duas observações alocadas incorretamente, 66 e a 268, deveriam estar no *cluster* 2, mas foram alocadas no *cluster* 3. Suas probabilidades eram realmente maiores com as das observações do *cluster* 3: no mínimo 0,558 para a 66 e 0,546 para 268, enquanto com o *cluster* 2 eram no máximo: 0,490, para a 66 e 0,509 para a 268.

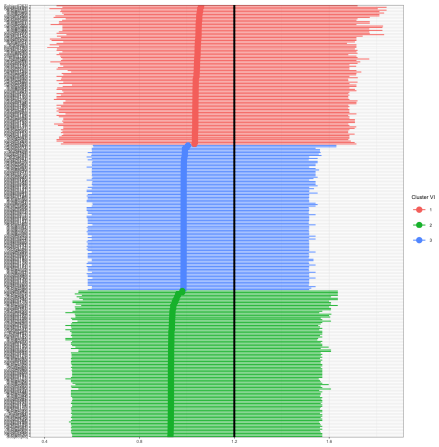
O Cenário (d), Figura 3.3d, que considera variâncias diferentes e com valores mais altos, também apresentou probabilidades maiores entre os *clusters* estimados, mas também apresenta um padrão bloco-diagonal e probabilidades moderadas dentro dos *clusters*: no mínimo 0,422 para o *cluster* 1, relacionado ao componente que considerada variância de 5; 0,414 para o *cluster* 2, relacionado ao componente que considera variância de 7; e 0,482 para o *cluster* 3, relacionado ao componente que considera variância de 15,5. Entre os indivíduos de *clusters* diferentes as probabilidades foram no máximo 0,471 entre os *clusters* 1 e 2; 0,517 entre os *clusters* 2 e 3; e 0,305 entre os *clusters* 1 e 3.

A Figura 3.4 apresenta as estimativas de $\boldsymbol{\mu}$ pela mediana da distribuição a *posteriori* e seus intervalos HPD de 95% de probabilidade. Os pontos são as observações dos dados simulados e as cores deles são os *clusters* verdadeira alocação Nos cenários (a), Figura 3.4a, e (b), Figura 3.4b, possuem estimativas bem próximas aos verdadeiros valores das medidas de locação (18, 35 e 55) e intervalos são mais precisos, com amplitudes menores. O cenário (c), Figura 3.4c, também apresenta estimativas próximas aos valores verdadeiros das medidas de locação (18, 25 e 55), mas o intervalos são mais amplos entre o *cluster* 1 (verde) e *cluster* 2 (azul), sendo os que possuem médias próximas. O cenário (d), Figura 3.4d, apresenta mais intervalos amplos devido às variâncias maiores, mas também estimam bem as medidas de locação. Em todos os cenários os valores observados, representados pelos pontos, estão dispostos em torno dos valores estimados para as medidas de locação.

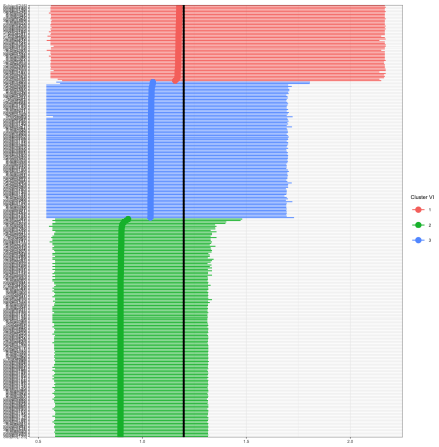
As variâncias dos *clusters* também são estimadas pelo modelo tendo sido mostradas na Figura 3.5. O modelo permite diferentes valores de variâncias para cada *cluster*, $\sigma_k^2, k = 1, \ldots, K$, mas na maioria dos cenários os dados foram gerados com variâncias iguais a 1,2. O modelo apresenta flexibilidade nas estimativas, com diferentes valores de variâncias para cada *cluster*, além de intervalos HPD 95% e estimativas com valores próximos para indivíduos do mesmo *cluster*. Em todos os cenários os valores reais estão contidos nos intervalos HPD de 95% de probabilidade e com exceção do Cenário base (a) o modelo estima valor com pouco erro (próximo do verdadeiro) para pelo menos um *cluster*.

Figura 3.5: Medianas da distribuição *a posteriori* e Intervalos HPD de 95% de probabilidade para σ_k^2 em todos os cenários. As cores dos intervalos representam a alocação estimada pela perda VI e na cor preta está o verdadeiro valor de σ_k^2.

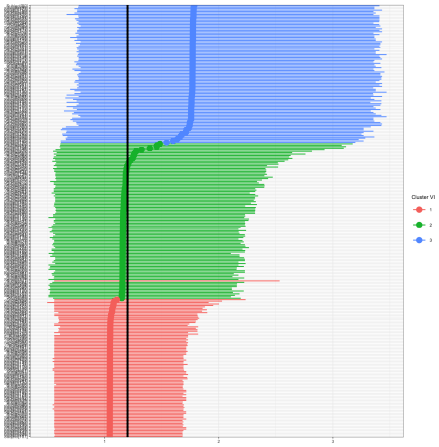
(a) Cenário base.



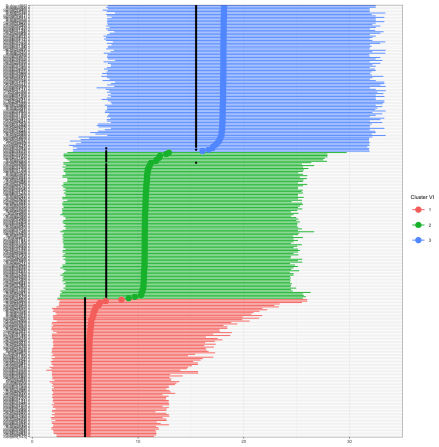
(b) Cenário com pesos diferentes.



(c) Cenário com duas médias próximas.



(d) Cenário com variâncias diferentes.



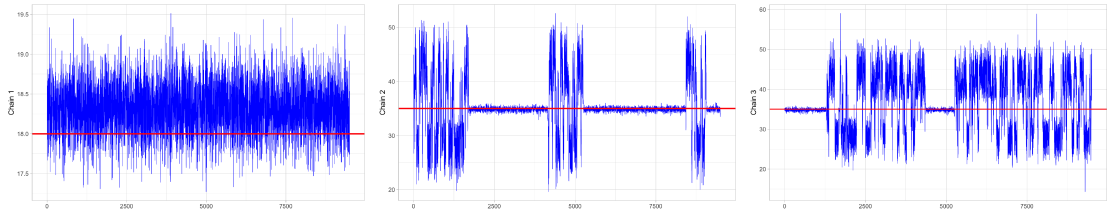
Fonte: Elaborado pela autora.

3.5.1.1 Comentários adicionais sobre a parte computacional

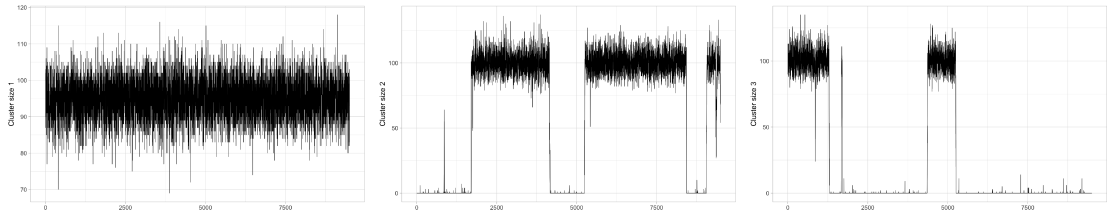
Os parâmetros do kernel, θ^2 e σ_q^2, apresentaram taxas de aceitação entre 31% e 49%. O parâmetro de locação, possui características específicas quanto a taxa de aceitação ao fixar K_{max}, o máximo do número de grupos, em valor maior do que realmente os dados apresentam. Taxas mais baixas ocorrem para as cadeias de componentes da mistura associados a médias verdadeiras de *clusters*, essas cadeias tendem a ter quase sempre indivíduos alocados a elas ao longo das iterações. Enquanto isso as cadeias que não apresentam valores próximos aos valores de médias reais dos *clusters* tendem não ter indivíduos alocados a elas (vazias) e suas taxas de aceitação são maiores, atingindo 88%. Se a variância da distribuição proposta é inflada para controlar a taxa de aceitação das cadeias de grupos vazios, isso diminui excessivamente a taxa de aceitação das outras cadeias. Esta característica pode ser contornada usando propostas com

Figura 3.6: Cadeias para o parâmetro de locação e os tamanhos de grupos no Cenário d). Linhas vermelhas são verdadeiros valores dos parâmetros, grupo 1 com valor do μ_1 =, grupo 4 com valor de μ_3, grupos 2, 3 e 6 com valores de μ_2. A linha amarela é a média dos dados como referência na cadeia do grupo 5 (predominantemente vazio).

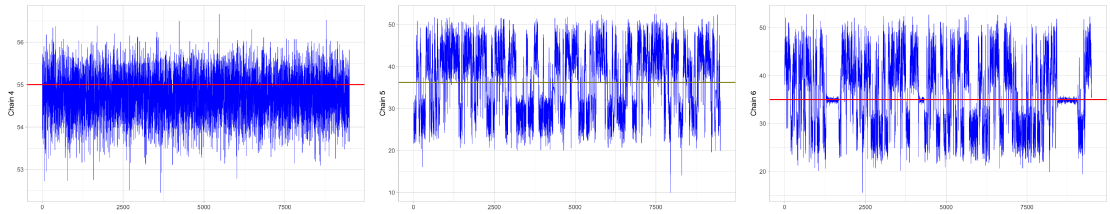
(a) Cadeia para μ do grupo 1. (b) Cadeia para μ do grupo 2. (c) Cadeia para μ do grupo 3.



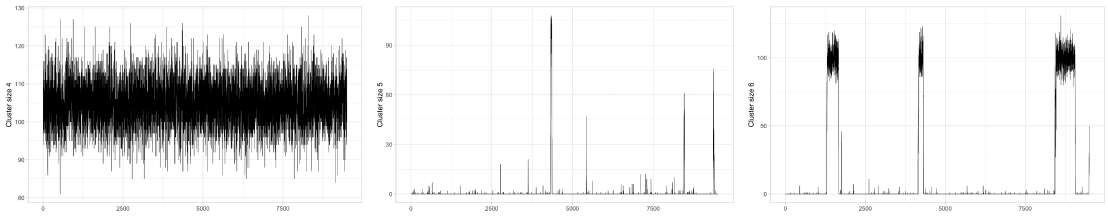
(d) Tamanho do grupo 1. (e) Tamanho do grupo 2. (f) Tamanho do grupo 3.



(g) Cadeia para μ do grupo 4. (h) Cadeia para μ do grupo 5. (i) Cadeia para μ do grupo 6.



(j) Tamanho do grupo 4. (k) Tamanho do grupo 5. (l) Tamanho do grupo 6.



Fonte: Elaborado pela autora.

variâncias diferentes para as cadeias vazias e para as cadeias não vazias, ou podem ser utilizadas como indicação do número de *clusters*: cadeias não vazias.

Para ilustrar o comportamento das cadeias vazias e não vazias, os dados simulados no cenário (d) também foram ajustados utilizando uma modificação do kernel. *a priori* foi baseada numa função qualidade centrada no valor da média dos dados: 36,23. Dessa forma, $q(\mu_i) = \frac{1}{\sqrt{2\pi}\sigma_q} \exp\left\{ -\frac{(\mu_{kd}-36{,}23)^2}{2\sigma_q^2} \right\}$. Na Figura 3.6, são apresentadas as cadeias para este ajuste com os parâmetros de locação e os respectivos tamanhos de grupos ao longo das iterações para o Cenário (d). Os verdadeiros valores do vetor de parâmetros, (μ_1, μ_2, μ_3) = (18, 35, 55), foram mostradas como linhas de referência nas cadeias que mais se adequavam. Com exceção da cadeia do grupo 5 (Figura 3.6h), sendo predominantemente vazia. Nesse caso, o valor de

referência foi a média dos dados. Os comportamentos evidenciam que as cadeias quando vazias são dominadas pela distribuição designada *a priori* e que a distribuição do PPD é influenciada diretamente pela função qualidade, $q(\cdot)$, e, consequentemente, pelo valor central nela especificado.

3.6 Aplicação: análise da biomassa no plantio de banana

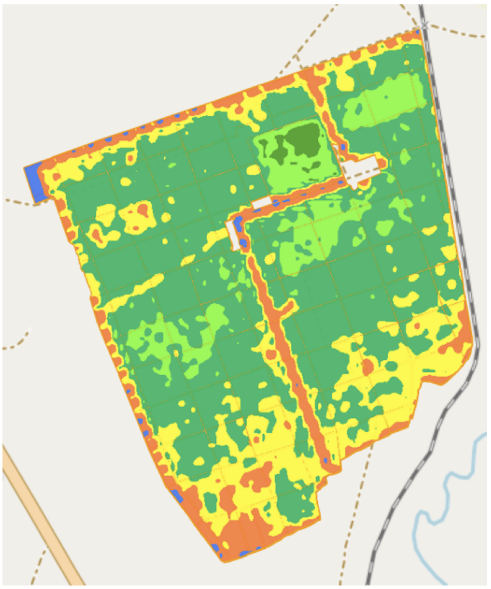
A biomassa é responsável pela decomposição e mineralização de resíduos vegetais e orgânicos, esses são fonte de energia e nutrientes para sua manutenção e multiplicação [43]. A partir da sua quantificação é possível avaliar alterações na quantidade de matéria orgânica que o sistema de cultivo pode causar [44], isto permite identificar possíveis mudanças no funcionamento do ciclo de nutrientes e, consequentemente, na produtividade do sistema de plantio [111]. Além disso, os níveis baixos de biomassa mostram sinais de empobrecimento do solo que podem ser associados a pragas como o mal-do-Panamá, um grande vilão do cultivo de banana em larga escala.

O Índice de Vegetação da Diferença Normalizada (Normalized Difference Vegetation Index) - NVDI, possui uma forte correlação com a biomassa e serve como um indicador de atividade fotossintética, ou seja, crescimento de vegetação sadia. Ele é muito útil no monitoramento das lavouras, na obtenção de estimativas de potencial de produção agrícola e na tomada de decisões associadas ao manejo da cultura. É possível obter valores de NVDI a partir de imagens de satélite, evitando medições extensas e demoradas, no entanto, geralmente é necessário um tratamento da imagem para uso da informação, por exemplo, com valores para a biomassa. Na Figura 3.7b, é possível observar a proporção de cada nível de biomassa segundo as cores para a propriedade. Neste caso, os valores estão no intervalo de 19,71 a 85, a maioria da área permanece saudável. As cores azul, laranja e amarela representam menores níveis de biomassa, logo, são consideradas mais ligadas à presença do mal-do-Panamá. Enquanto os tons de verde possuem nível de biomassa saudável.

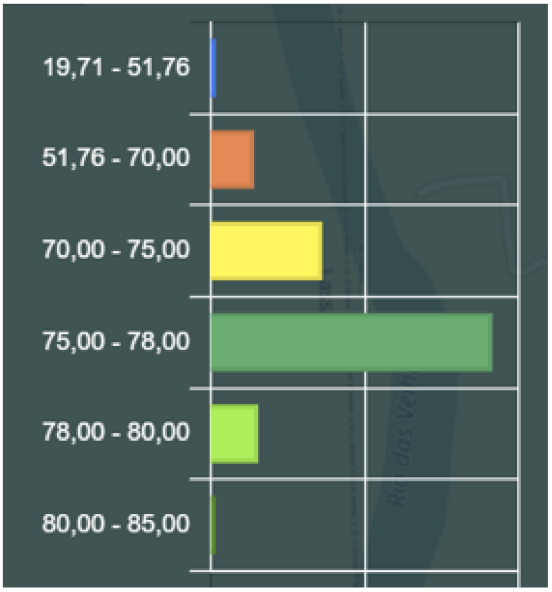
Nesse contexto, serão analisados dados referentes à área plantada de uma fazenda dedicada ao plantio de banana. Esta fazenda está localizada na região central de Minas Gerais, e as informações são referentes a imagem de satélite obtida em julho de 2021, mostrando a distribuição da biomassa segundo o NVDI na área plantada (Figura 3.7a). A fazenda é dividida em partes aproximadamente quadriculadas chamadas talhões. No entanto, algumas regiões possuem construções humanas, como a estrada que aparece como uma linha laranja na parte central da Figura 3.7a. Estas regiões foram excluídas das análises, pois não são áreas plantáveis e tem baixa biomassa por motivos alheios ao mal-do-Panamá. Dessa forma, foram removidos os talhões: CABO5 FS1 AM1, CABO 5 FS1 AM2, CABO 5 FS1 AM3, CABO 5 FS1 AM4,

Figura 3.7: Imagem original da biomassa segundo o NVDI medidas por talhões.

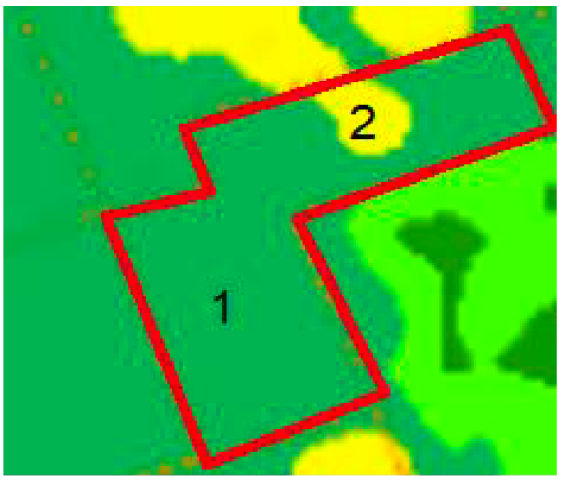
(a) Imagem de satélite referente ao NVDI na propriedade.



(b) Níveis de biomassa na fazenda.



(c) NCABO 4 POS 4.



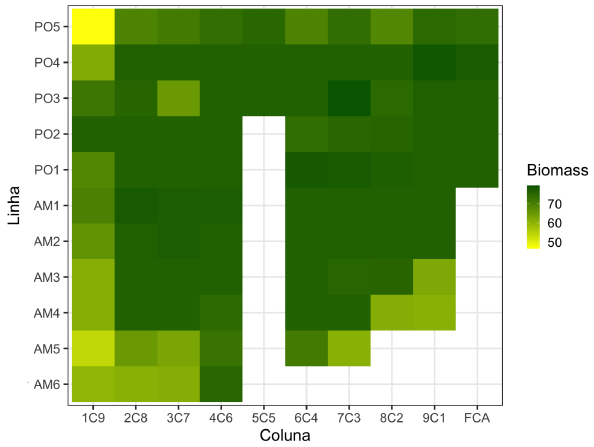
Fonte: Inceres https://inceres.com.br/

CABO 5 FS1 AM5, CABO 5 FS2 POS1 e CABO 5 FS2 POS2 e CABO 3 POS3, resultando em 92 talhões na amostra.

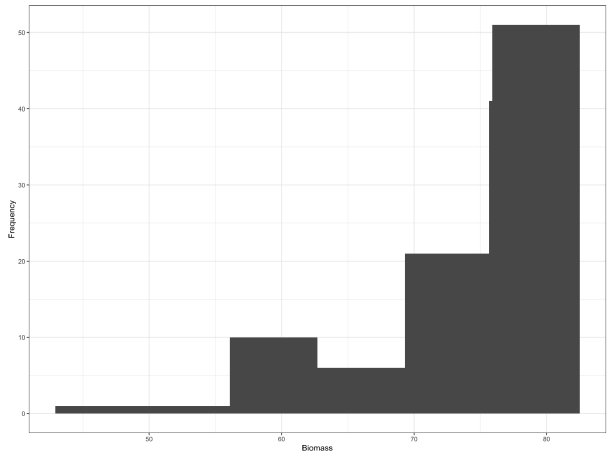
Os dados foram obtidos via tratamento da imagem de satélite para fornecer a cada um dos talhões uma medida da biomassa segundo o NVDI. Cada talhão foi separado em uma imagem individual e lido para identificar número de píxeis de cada cor. Este trabalho foi realizado com o pacote *png* do programa *R* [91]. As proporções de cores foram usadas para o cálculo de uma média ponderada dos valores centrais dos intervalos que representam as cores da Figura 3.7b. Valores baixos serão considerados relacionados à presença do mal-do Panamá. Em cada talhão, a cor indica o nível médio de biomassa medido pelo NVDI. Um tratamento prévio foi realizado com o talhão CABO 4 FS2 POS4, por possuir um formato não retangular, dificultando o tratamento da imagem contígua. Para obter uma melhor leitura, o talhão foi separado em parte

Figura 3.8: Biomassa segundo o NVDI na fazenda.

(a) Imagem tratada.



(b) Distribuição da biomassa.



Fonte: Elaborado pela autora.

1 e 2, conforme a Figura 3.7c, no entanto, a média ponderada obtida como resposta das duas partes não apresentou diferença, comprovando que as partes podem ser tratadas como um talhão apenas. Além desse, os talhões CABO 7 FS1 AM.6 e CABO 7 FS1 AM.EXP foram considerados um só talhão, por serem áreas menores praticamente equivalentes, juntas, a um talhão da coluna em que estão localizados e suas medidas de biomassa eram muito semelhantes.

Este estudo visa analisar os dados da área onde ocorre o plantio de banana para determinar possíveis agrupamentos que identifiquem a presença do mal-do-Panamá caracterizado pelo declínio da biomassa. É possível observar na Figura 3.8b, que há uma cauda mais alongada na distribuição dos dados, assim será utilizado o ajuste do modelo de mistura Normal/Independente apresentado na Seção 3.2. Além disso, a biomassa apresenta grande heterogeneidade dentro de um mesmo talhão (unidade amostral) essa é uma dificuldade inerente dos dados. A imagem tratada, Figura 3.8a, mostra dados mais homogêneos por ter sido contruída pelas médias das medidas das cores.

3.6.1 Resultados

Para o conjunto de dados da Biomassa foi considerado $K_{max} = 5$, δ=1,0, $a_0 = b_0$ =0,01, a_1 =105, $b_1 = 150$, $a_2 = 170$, $b_2 = 130$. Os valores iniciais foram $\theta^{2(0)} = 2{,}6$, a mediana das distâncias entre os dados, $\sigma_q^{2(0)} = 125$, $\boldsymbol{\mu}$=(76,4; 59,6; 78,4; 73,0; 67,8), centroides definidos pelo método de k-médias com 5 grupos. As cadeias foram geradas com 100000 iterações, 40000 iterações iniciais foram descartadas como aquecimento e realizados saltos a cada 30 valores para evitar autocorrelações fortes. Dessa forma, foram utilizadas 2000 amostras para estimação dos parâmetros de todos os modelos. A convergência foi verificada por teste de Geweke e função de autocorrelação, os gráficos das cadeias, bem como as taxas de aceitação são apresentadas no Apendice B.

A Tabela 3.4 mostra, para os modelos ajustados com $\eta = 2.1$, η =5 e $\eta = 100$, as distribuições dos números de *clusters* bem como seus intervalos HPD de 95% de probabilidade em que se verifica que todos os ajustes estimam 4 *clusters* com probabilidades maiores que 0,5. A avaliação dos modelos será realizada utilizando o DIC (*Deviance Information Criterion*) observado [21]. O cálculo do DIC refere-se ao interesse em um parâmetro especificado, θ, sendo obtido por $DIC = -2\overline{D(\theta)} - D(\tilde{\theta})$ em que $D(\theta) = -2log\, f(\mathbf{y}|\theta)$ e $\tilde{\theta}$ é uma estimativa para θ *a posteriori* , por exemplo, a média ou a mediana *a posteriori* . Para o modelo NIPPD é dado por

$$\begin{aligned} \overline{D(\theta)} &\approx -\frac{2}{m}\sum_{l=1}^{m} \log f(y_i, z_i, u_i|\mu^{(l)}, \sigma^{2(l)}) \\ &= -\frac{2}{m}\sum_{l=1}^{m}\sum_{i=1}^{n} \log\left\{\sum_{k=1}^{K} w_k^{(l)} \phi(y_i|\mu^{(l)}, \sigma^{2(l)}, u_i^{(l)}) g(u_i|\eta)\right\}, \end{aligned} \tag{3.28}$$

em que $\phi(\cdot)$ é a função densidade de probabilidade de uma distribuição normal e $g(\cdot)$ é a função densidade da distribuição gama. Segundo o DIC, o melhor modelo é o de $\eta = 100$ com menor valor.

Tabela 3.4: Distribuição do número de grupos e DIC para os modelos de mistura NIPPD ajustados aos dados de biomassa.

η	K					HPD 95%		DIC
	1	2	3	4	5	Inf	Sup	
2.1	0,00	0,00	0,20	**0,52**	0,28	3	5	4103568
5.0	0,00	0,00	0,35	**0,53**	0,12	3	5	21425,06
100	0,00	0,00	0,01	**0,66**	0,34	4	5	-42038,41

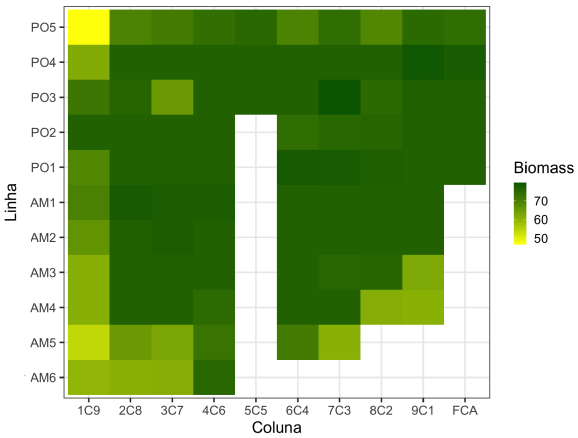
Fonte: Elaborado pela autora.

Na Figura 3.9, as cores representam a intensidade da biomassa observada na fazenda e a biomassa mediana estimada pelos modelos para cada um dos talhões, quanto mais verde-escuro

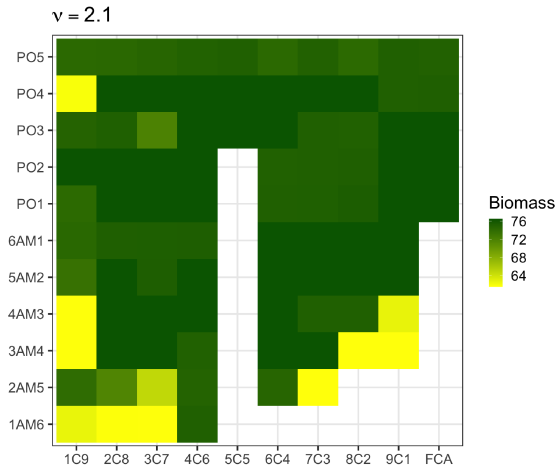
é a cor, maior é o valor da biomassa, quanto mais amarela é a cor, menor é o valor da biomassa. As medidas de locação apresentam coerência com os valores reais da Biomassa para a fazenda, Figura 3.9a. Nas Figuras 3.9b, 3.9c e 3.9d, é possível visualizar que os talhões localizados nas bordas da fazenda tiveram medianas estimadas com valores menores, assim como nos dados originais.

Figura 3.9: Biomassa observada pelo NVDI para cada talhão da fazenda e estimativas pelas medianas *a posteriori* para cada talhão obtidas pelos modelos NIPPD para η =2,1; 5,0 e 100.

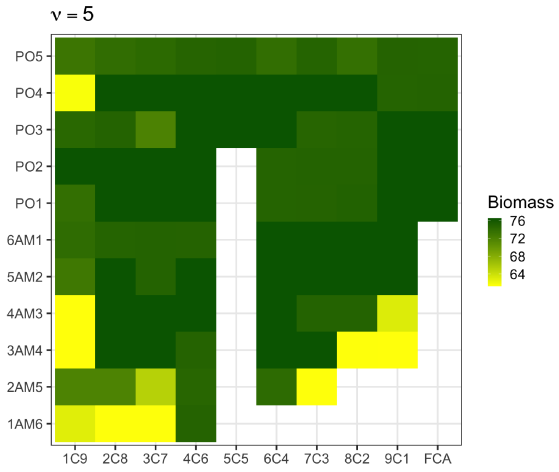
(a) Fazenda.



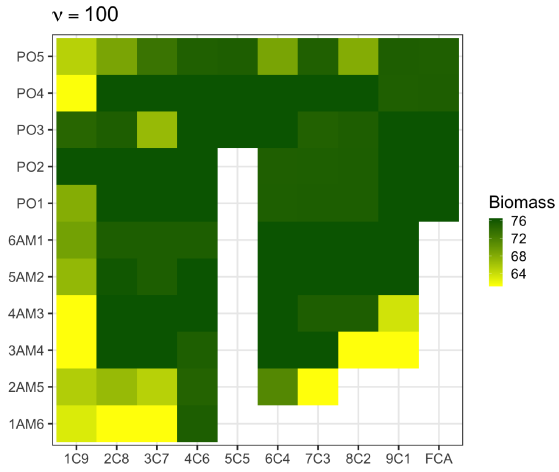
(b)



(c)



(d)



Fonte: Elaborado pela autora.

As funções de perda, VI, ARI e de Binder, utilizadas para resumir a alocação dos *clusters*, $\mathbf{z}$, definiram diferentes números de *clusters*. A perda ARI foi a única que apresentou 4 grupos como estimado pelos modelos, mesmo assim, só para os modelos com η =2,1 e com η = 100. A concordância da alocação definida pelas perdas é apresentada ma Tabela 3.5. Unicamente, para o modelo com η = 5 as três perdas concordam, não apenas com o número de *clusters*, como também sobre os talhões alocados a eles.

Tabela 3.5: Concordância da alocação pelas três perdas VI, ARI e de Binder para os modelos NIPPD ajustados com η = 2,1; 5; 100 para os dados da biomassa.

η		Binder					ARI			
	VI	1	2	3	4	5	1	2	3	4
2,1	1	41	0	0	0	0	41	0	0	0
	2	0	11	33	3	2	0	11	36	2
5,00	1	41	0	0			41	0	0	
	2	0	11	0			0	11	0	
	3	0	0	38			0	0	38	
100	1	39	0	0	0	2	41	0	0	0
	2	0	3	8	0	0	0	3	8	0
	3	0	10	0	28	0	0	10	0	28

Fonte: Elaborado pela autora.

As Figuras 3.10, 3.11 e 3.12, apresentam as matrizes de similaridade estimada pelos modelos. As matrizes representam as probabilidades das observações estarem no mesmo *cluster* e as observações foram ordenadas conforme as perdas consideradas para a estimação de alocação: Perda VI, Perda ARI e Perda de Binder. Considerando as diferentes perdas, a mudança é na alocação e ordenação dos talhões nos gráficos, não nos valores das probabilidades estimadas pelas matrizes de similaridade para cada um dos três modelos. Todas as figuras mostram um padrão bloco-diagonal, que define a forma de alocação das observações nos *clusters*, mas alguns são melhor definidos que outros.

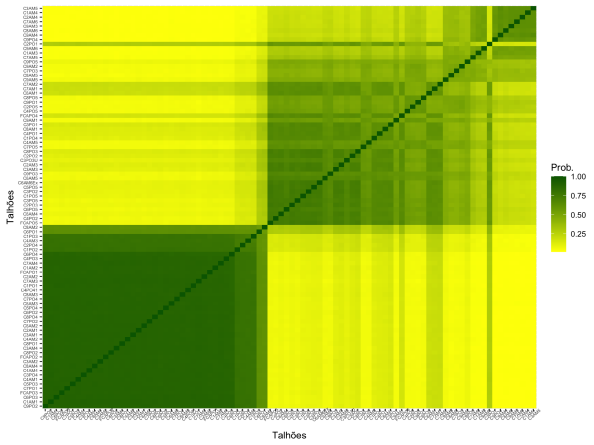
Considerando a perda VI, Figura 3.10, um dos *clusters* é melhor identificado em todos os modelos (canto inferior esquerdo), com observações com probabilidades mais altas de estarem no mesmo *cluster* entre si: no mínimo 0,563 para o modelo com η = 2,1, no mínimo 0,505 para o modelo com η =5,0 e no mínimo 0,461 para o modelo com η = 100. As probabilidades mais baixas em todos esses casos são relacionadas aos talhões C6PO1 e C8AM2, que possuem probabilidades razoáveis de estarem alocadas com talhões de outros *clusters*, mas que mesmo assim são menores que as dos talhões no *cluster* 1. Estes dois talhões repetiram esse comportamento em todos os modelos para todas as perdas. Enquanto isso, dentro dos outros *clusters* encontram-se observações com probabilidades menores entre si de estarem no mesmo *cluster*. Para o modelo com η = 2,1, Figura 3.10a, o segundo *cluster* possui probabilidade de no mínimo 0,133 até 0,758 entre as observações que o compõem. Entre talhões de *clusters* diferentes essa probabilidade é de até 0,418, entre observações do *cluster* 1 e do *cluster* 2.

Nos modelos com η = 5 e η = 100, a perda VI definiu 3 *clusters*. Para o modelo com η =5,0, Figura 3.10b, o *cluster* 2 tem probabilidade de no mínimo 0,491, entre observações do mesmo *cluster* de serem alocadas juntas, e probabilidade de no mínimo 0,457 para as observações do *cluster* 3. Mas entre observações de *clusters* diferentes, tem-se que, entre o cluster 1 e o 2 as probabilidades são mais baixas e estão no intervalo 0,001-0,157, e entre as observações do *cluster* 1 e do *cluster* 3, as probabilidades estão no intervalo [0,008—0,409].

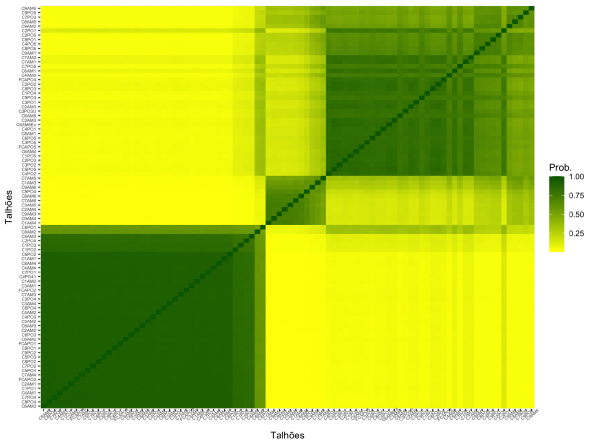
Ainda sob a perda VI, para o modelo com η = 100, Figura 3.10c, nos dois outros *clus-*

Figura 3.10: Gráfico coroplético das Matrizes de similaridade, estimadas pelo modelo NIPPD para estimação de densidade ajustado considerando valores de graus de liberdade $\eta = 2{,}1; 5, 0$ e 100, para os dados da biomassa nos talhões. A ordenação dos talhões foi obtida segundo a perda VI.

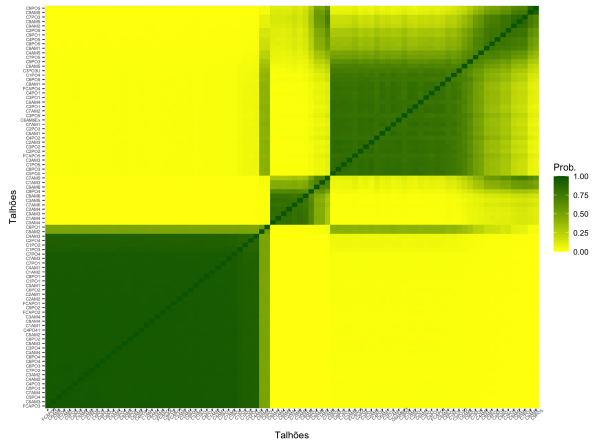
(a) $\eta = 2{,}1$. Dois *clusters* identificados.



(b) $\eta = 5$. Três *clusters* identificados.



(c) $\eta = 100$. Três *clusters* identificados.



Fonte: Elaborado pela autora.

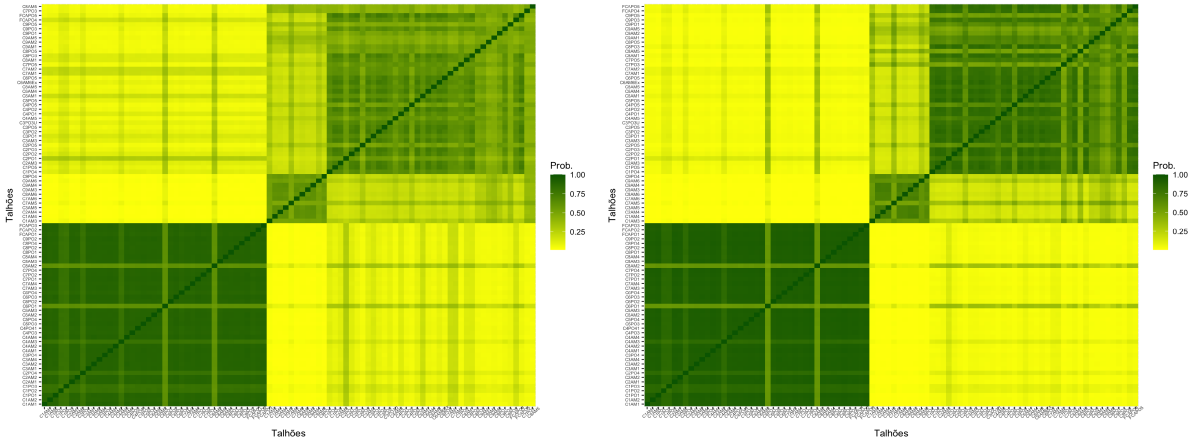
ters, os talhões possuem probabilidades de alocação entre talhões do mesmo *cluster*, variando muito. Dentro do *cluster* 2 as probabilidades estão no intervalo [0,260—0,835], e no *cluster* 3 [0,042—0,871]. Os *clusters* 1 e 2 tem observações com pouca probabilidade de estarem agrupadas, no máximo 0,046, os *clusters* 1 e 3 tem probabilidade até de 0,453. Enquanto entre os indivíduos dos *clusters* 2 e 3 tem algumas probabilidades maiores de serem alocadas juntas variando entre 0,004 e 0,711.

Para o modelo que considera $\eta = 5$, tanto a perda ARI, Figura 3.11b, quanto a perda de Binder, Figura 3.12b, definiram 3 *clusters*, nestes casos, os *clusters* foram os mesmos definidos pela perda VI, consequentemente, cabem as mesmas considerações. Enquanto isso, foram definidos 4 *clusters* para o modelo considerando $\eta = 2{,}1$, a partir da perda ARI. O primeiro *cluster* é o que apresenta melhores resultados de definição, assim como nos casos da perda VI. Para o modelo com $\eta = 2{,}1$, este primeiro *cluster* tem probabilidade no mínimo de $0, 563$ entre as

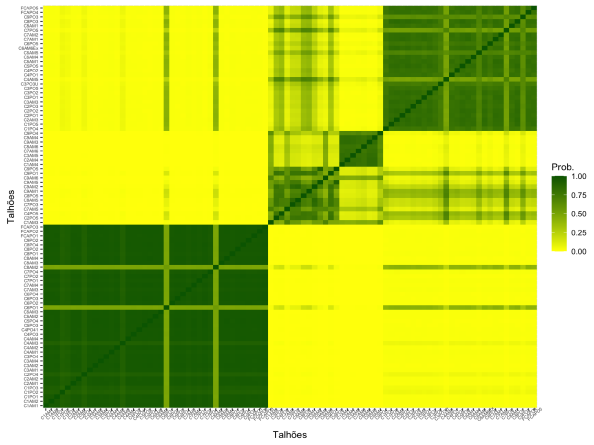
Figura 3.11: Gráfico coroplético da Matriz de similaridade, para os grupos de talhões pela biomassa, valores de graus de liberdade $\eta = 2{,}1$; 5 e 100, ordenação dos talhões segundo a perda ARI.

(a) $\eta = 2{,}1$. Quatro *clusters* identificados pela perda ARI: *cluster* 1 a esquerda em baixo e *cluster* 2, menor no meio com 11 talhões, o *cluster* 3 com 36 talhões, e o *cluster* 4 direita e acima composto por 2 talhões.

(b) $\eta = 5$. Três *clusters* identificados pela perda ARI: *cluster* 1 (41 talhões), a esquerda em baixo, *cluster* 2, menor no meio (11 talhões), e *cluster* 3 a direita e acima (38 talhões).



(c) $\eta = 100$. Quatro *clusters* identificados pela perda ARI: *cluster* 1 a esquerda em baixo e *cluster* 2, no meio abaixo com 13 talhões, o *cluster* 3 é o menor no meio com 8 talhões, e o *cluster* 4 direita e acima, a direita, composto por 28 talhões.



Fonte: Elaborado pela autora.

observações que o compõem, no máximo $0,156$ de probabilidade de alocação com os talhões no *cluster* 2 e no máximo $0,419$ de probabilidade de alocação com os talhões no *cluster* 3. Os demais *clusters*, apresentam probabilidades de no mínimo $0,483$ entre os talhões no *cluster* 2, entre si, e no mínimo $0,336$ entre os talhões no *cluster* 3, entre si, e $0,513$ para o *cluster* 4. Já as probabilidades entre talhões desses diferentes *clusters* são de até $0,469$, entre talhões do *cluster* 2 e 3, e $0,511$ entre os talhões dos *clusters* 3 e 4. Esta probabilidade $0,469$ é referente ao talhão C7AM5, do *cluster* 2, mas as probabilidades de alocação com os outros talhões do

mesmo *cluster* ainda são maiores que esta, no mínimo $0,484$. O *cluster* 4 foi definido com os talhões C7PO3, C8AM5 e C9AM2, que na perda VI estavam no *cluster* 2 e na perda ARI estavam no *cluster* 3. Enquanto o *cluster* 5 foi definido com os talhões C9AM5 e C9PO5.

Sobre o modelo considerando $\eta = 100$ e sob a perda ARI, a alocação de talhões ao primeiro *cluster* foi comum ao modelo com $\eta = 2{,}1$ para esta perda, com probabilidade mínima de $0,477$, de seus talhões estarem no mesmo *cluster*. A probabilidade de talhões do *cluster* 1 estarem no mesmo *cluster* que talhões de outros *clusters* foi no máximo $0,186$ com talhões do *cluster* 2, $0,021$ com talhões do *cluster* 3 e $0,439$ com talhões do *cluster* 4. Para os componentes dos outros três *clusters*, a probabilidade mínima de seus talhões estarem juntas foi de $0,410$, $0,639$ e $0,485$, respectivamente. Entre talhões de *clusters* diferentes a probabilidade máxima foi de 0,622 entre os *clusters* 2 e 3; 0,703 entre os *clusters* 2 e 4; e 0,198 entre os *clusters* 3 e 4. Para o *cluster* 2 as probabilidades mostram que houve dificuldade na locação pela perda ARI. Por exemplo, a probabilidade máxima $0,622$ é relacionada ao talhão C1AM3 do *cluster* 2 com o talhão C9PO4 do *cluster* 3, as probabilidades de C1AM3 estar alocado com outros talhões do *cluster* 2 são todas abaixo deste valor exceto duas $0,712$ e $0,695$ que devem ter influenciado sua alocação ao *cluster* 2. Algo parecido ocorre entre o *cluster* 2 e 4, a probabilidade $0,703$ é relacionada ao talhão C8PO5 do *cluster* 2 com o C4AM5 do *cluster* 4, mas as probabilidades relacionadas ao talhão C8PO5 estar agrupado com os outros de seu próprio *cluster* são em sua maioria maiores que $0,703$.

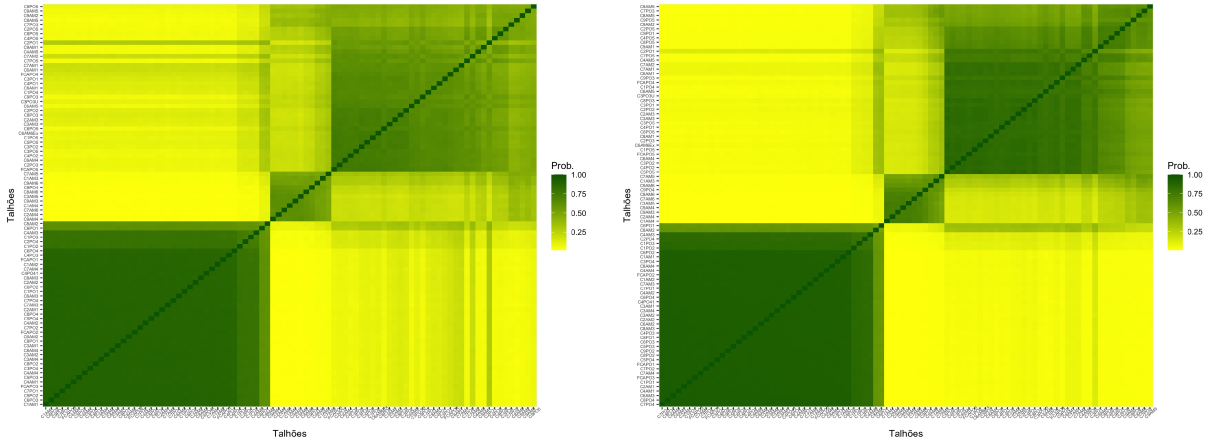
Considerando a perda de Binder, o modelo com $\eta = 5$ já foi comentado. Para os modelos com η =2,1 e $\eta = 100$ a perda de Binder identificou 5 *clusters*. O modelo considerando η =2,1 possui o mesmo *cluster* 1 com os mesmos 41 talhões de perdas anteriores e mesma probabilidade mínima entre seus talhões de 0,563 de estarem no mesmo *cluster*. Entre suas observações e as de outros *clusters*, 2 a 5, as probabilidades máximas são de 0,156; 0,418; 0,200 e 0,228; respectivamente. Os outros *clusters* apresentam probabilidades de seus talhões, entre si, de estarem no mesmo *cluster* de no mínimo 0,484; 0,411; 0,492 e 0,485; para os *clusters* 2 a 5, respectivamente. Esta configuração para a alocação dos talhões foi a que permitiu probabilidades mais altas de talhões estarem com talhões de *clusters* diferentes: no máximo 0,418 para os *clusters* 2 e 3, 0,481 para os *clusters* 2 e 4, 0,469 para os *clusters* 2 e 5, 0,522 para os *clusters* 3 e 4 0,504 e 0,482 para os *clusters* 4 e 5.

O modelo com $\eta = 100$, sob a perda de Binder, possui 39 talhões no *cluster* 1. Comparando com as outras perdas, as duas observações que tinham menos probabilidade de estarem alocadas com os outros deste *cluster* foram separadas num *cluster* 5. Nesta configuração, a probabilidade mínima entre os talhões do *cluster* 1 serem alocadas juntas é de 0,878, com probabilidades bem pequenas de seus talhões serem alocados com os de outros *clusters*: 0,024 no máximo com os talhões do *cluster* 2; 0,003 no máximo com os talhões do *cluster* 3; 0,062 no máximo com os talhões do *cluster* 4 e maiores com os talhões do *cluster* 5 no máximo 0,499. Os outros *clusters* tem probabilidades relacionadas aos seus talhões entre si, de no mínimo 0,409 para o *cluster* 2, no mínimo 0,639 para o *cluster* 3, no mínimo 0,485 para o *cluster* 4 e

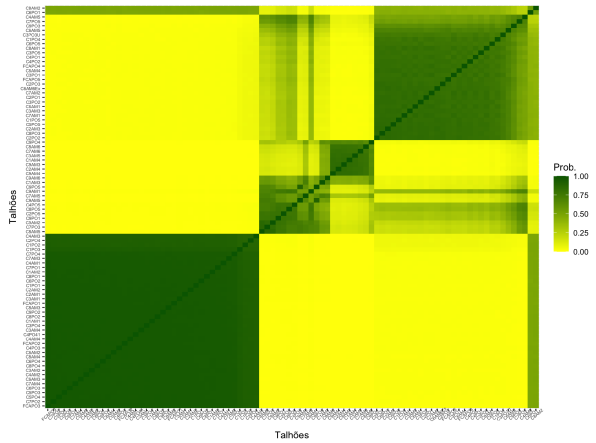
Figura 3.12: Gráfico coroplético da Matriz de similaridade, para os grupos de talhões pela biomassa, valores de graus de liberdade η =2,1; 5 e 100, ordenação dos talhões segundo a perda de Binder.

(a) η = 2,1. Cinco *clusters* identificados pela perda de Binder: *cluster* 1 a esquerda em baixo e *cluster* 2, no meio abaixo com 11 talhões, o *cluster* 3 é o maior no meio com 33 talhões, o *cluster* 4 direita e acima, composto por 3 talhões e o *cluster* 5 composto por 2 talhões.

(b) η = 5. Três *clusters* identificados pela perda de Binder: *cluster* 1 (41 talhões) a esquerda em baixo, *cluster* 2, menor no meio (11 talhões), e *cluster* 3 a direita e acima (38 talhões).



(c) η = 100. Cinco *clusters* identificados pela perda de Binder: *cluster* 1 a esquerda em baixo com 39 talhões, *cluster* 2, no meio abaixo com 13 talhões, o *cluster* 3 é o menor no meio com 8 talhões, o *cluster* 4 direita e acima, composto por 28 talhões e o *cluster* 5 composto por 2 talhões.

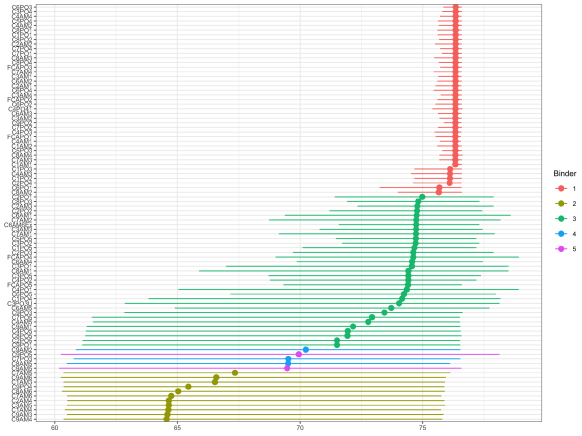


Fonte: Elaborado pela autora.

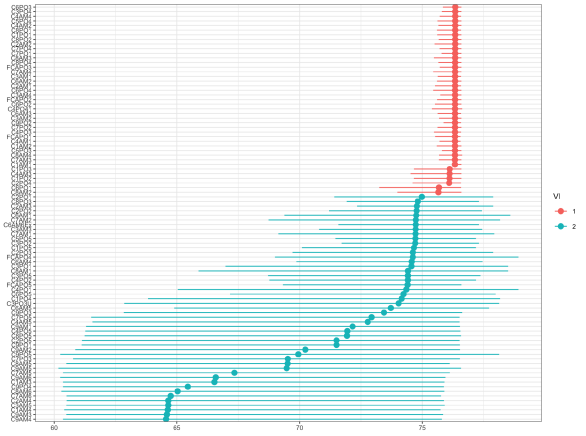
no mínimo 0,549 para o *cluster* 5. Já as probabilidades de talhões de *clusters* diferentes serem alocados juntos são no máximo 0,622 entre o *cluster* 2 e o *cluster* 3, 0,703 entre o *cluster* 2 e 4, 0,186 entre o *cluster* 2 e 5, 0,198 3 e 4, 0,021 3 e 5 0,439 4 e 5. Os casos mais difíceis de alocação são em relação ao *cluster* 2, mas se referem as mesmas observações discutidas na perda ARI. A novidade é probabilidade de 0,439 entre os *clusters* 4 e 5, mas que são menores que aquelas entre os talhões do *cluster* 4 entre si e as do *cluster* 5 entre si.

Figura 3.13: Intervalos HPD de 95% e a mediana como estimador pontual da medida de Biomassa para cada talhão. Grupos de talhões estimados pelas Perdas de Binder, VI e ARI para $\eta = 2{,}1$. Matriz de similaridade ordenada pelas medianas *a posteriori* estimadas pelo modelo.

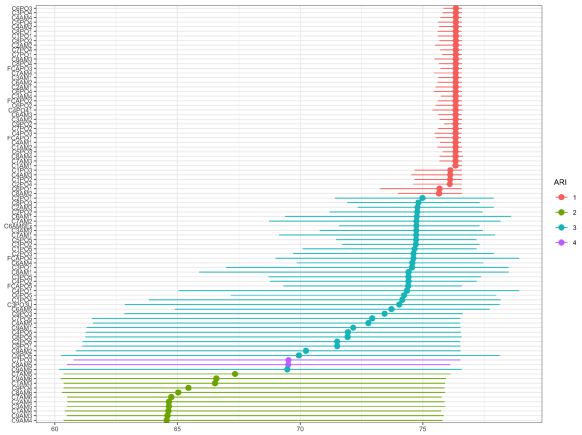
(a) Perda de Binder.



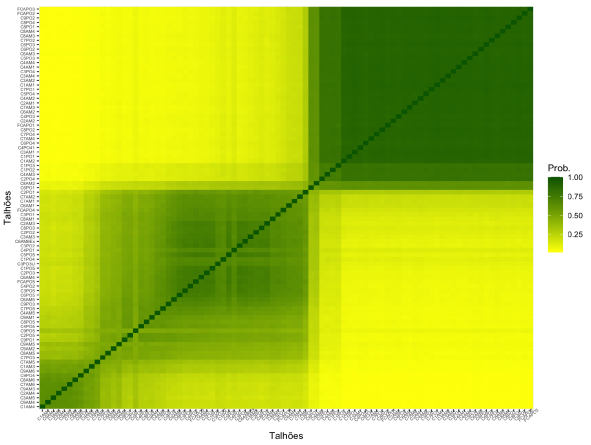
(b) Perda VI.



(c) Perda omARI.



(d) Matriz de similaridade estimada pelo modelo considerando $\eta = 2{,}1$ com talhões ordenados pelos valores das medianas *a posteriori* .



Fonte: Elaborado pela autora.

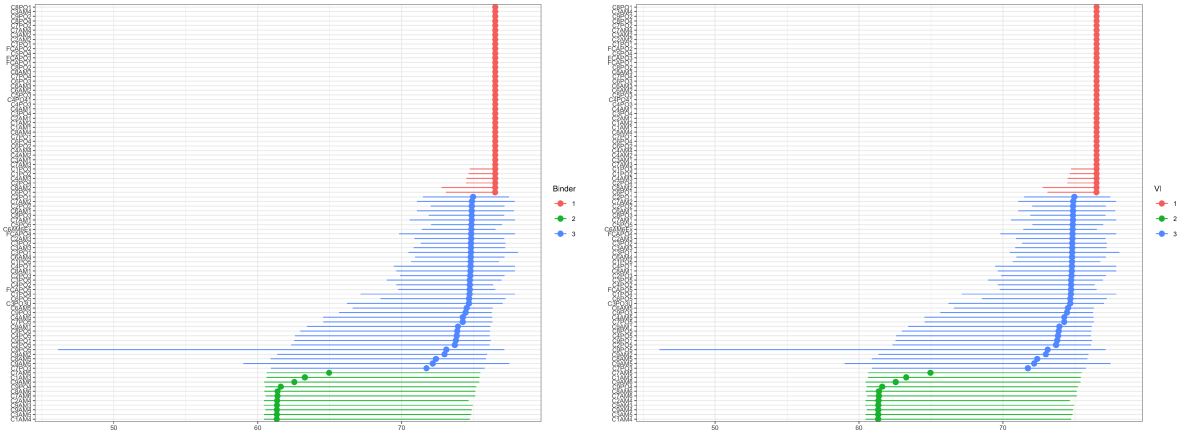
Além das alocações, as estimativas dos demais parâmetros foram avaliadas e serão discutidas. As medidas de locação do modelo foram estimadas usando as medianas das distribuições *a posteriori* e são apresentadas nas Figuras 3.13, 3.14 e 3.15 para os modelos com $\eta = 2{,}1$, $\eta = 5$ e $\eta = 100$. São apresentados os intervalos HPD de 95% de probabilidade para as medidas de locação de cada talhão. Além disso, as matrizes de similaridade foram ordenadas segundo a ordem das medianas a posteriori para discutir os diferentes números de *clusters* segundo as perdas e a estimação do modelo NIPPD.

Para todos os modelos as matrizes de similaridade ordenadas pelas medianas *a posteriori* possuem um cluster bem definido com altas probabilidades de alocação conjunta (cores mais escuras) e pouca probabilidade de alocação com talhões fora deste grupo, os que é condizente com as matrizes ordenadas segundo as perdas. Além disso, os intervalos HPD de 95% de probabilidade são mais precisos, com amplitudes menores para os talhões que coincidem com

Figura 3.14: Intervalos HPD de 95% e a mediana como estimador pontual da medida de Biomassa para cada talhão. Grupos de talhões estimados pelas Perdas de Binder, VI e ARI para $\eta = 5$. Matriz de similaridade ordenada pelas medianas *a posteriori* estimadas pelo modelo.

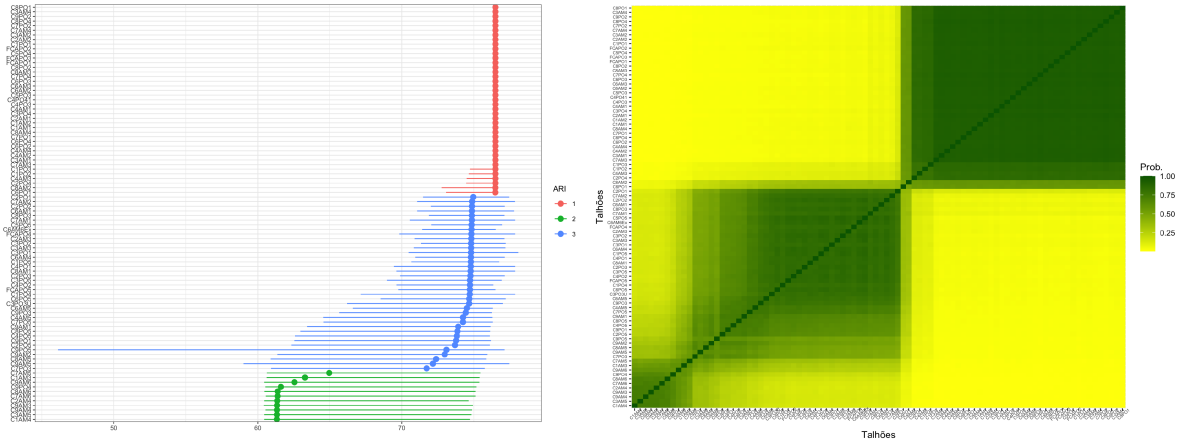
(a) Perda de Binder.

(b) Perda VI.



(c) Perda ARI.

(d) Matriz de similaridade estimada pelo modelo considerando $\eta = 5$ com talhões ordenados pelos valores das medianas *a posteriori* .



Fonte: Elaborado pela autora.

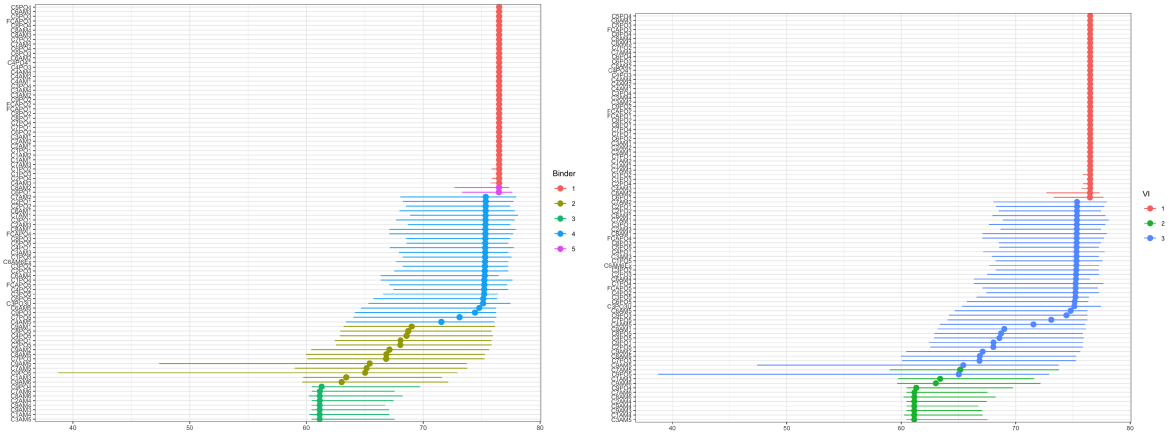
este *cluster* mais bem definido. Os outros talhões apresentam intervalos HPD que, em geral, tem suas amplitudes aumentamdo de acordo com a dimnuição do valor da mediana. Esse comportamento não ocorre para o modelo com $\eta = 100$ que possui intervalos HPD mais amplos para os talhões no meio dos gráficos, Figuras 3.15b, 3.15c e 3.15a.

Considerando o modelo NIPP com η=2,1, os intervalos HPD de 95% de probabilidade são os mais amplos entre os três modelos, Figuras 3.13b, 3.13c e 3.13a, podendo chegar ao comprimento de 17 unidades da medida de biomassa. Os *clusters* são contíguos em relação a ordenação pelas estimativas das medianas da distribuição *a posteriori* apenas segundo a perda VI. Para perda ARI, o talhão C9AM5 foi separado do seu *cluster* e para perda de Binder há uma mistura entre os talhões do *cluster* 4 e 5. Pela matriz de similaridade ordenada, Figura 3.13d, pelas medianas *a posteriori* , é possível identificar além do *cluster* mais bem definido um outro com probabilidades de alocação conjunta mais fracas ou dois *clusters* com uma carga

Figura 3.15: Intervalos HPD de 95% e a mediana como estimador pontual da medida de Biomassa para cada talhão. Grupos de talhões estimados pelas Perdas de Binder, VI e ARI para $\eta = 100$. Matriz de similaridade ordenada pelas medianas *a posteriori* estimadas pelo modelo.

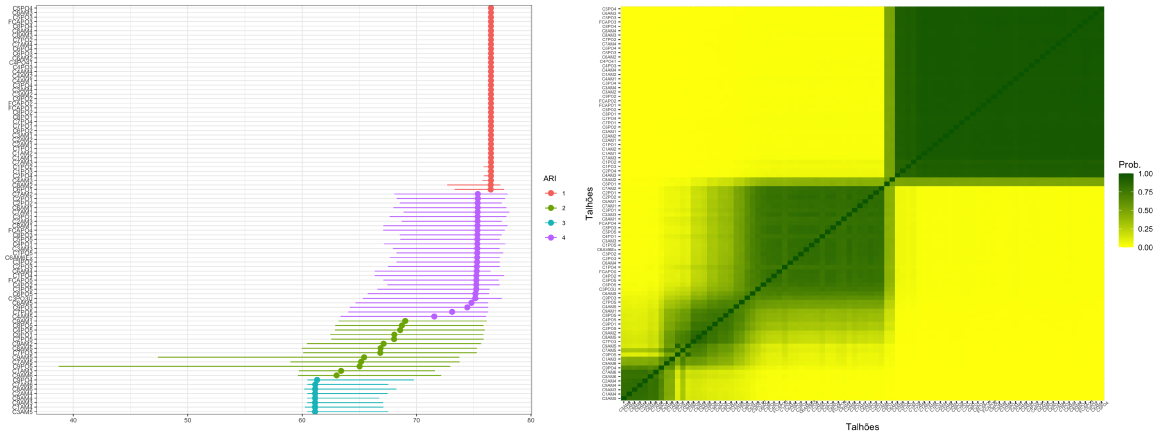
(a) Perda de Binder.

(b) Perda VI.



(c) Perda ARI.

(d) Matriz de similaridade estimada pelo modelo considerando $\eta = 100$ com talhões ordenados pelos valores das medianas *a posteriori* .



Fonte: Elaborado pela autora.

de confundimento, ou seja, com probabilidades maiores entre talhões desses diferentes *clusters*.

O modelo com $\eta = 5$, Figuras 3.14b, 3.14c e 3.14a, apresenta, em geral, intervalos HPD de 95% de probabilidade mais curtos que o modelo com $\eta = 2{,}1$. No entanto, para os talhões C9PO5 e C9AM5 os intervalos HPD foram bem maiores que os outros com amplitudes de 31 e 18.5 unidades de biomassa, respectivamente. Os outros talhões apresentaram intervalos HPD de no máximo 14.9 unidades. A matriz de similaridade ordenada pelas medidas das medianas *a posteriori* , Figura 3.14d, mostra três *clusters* de forma mais clara que no modelo com $\eta = 2{,}1$, mas ainda com algumas probabilidades de alocação entre os talhões que alocados a diferentes *clusters* para os dois abaixo no gráfico.

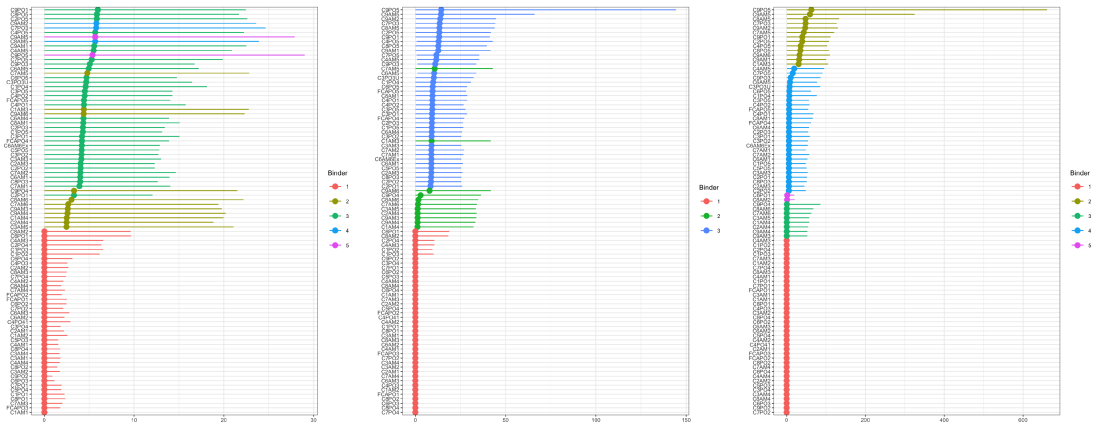
O modelo que considera $\eta = 100$, apresenta intervalos HPD com amplitudes mais homogêneas que os outros dois, isto desconsiderando os talhões associados ao *cluster* mais bem definido que possuem intervalos HPD muito menores em amplitude. Os talhões C9PO5

e C9AM5, assim como no modelo com $\eta = 5$, apresentam intervalos HPD com amplitudes discrepantes, bem maiores. Seus intervalos possuem amplitudes de 34,2 para o talhão C9PO5 e 26,3 para o talhão C9AM5. Os outros talhões possuem intervalos com amplitude de no máximo 15.3 unidades. Esstes talhões foram separados dos outros em um *cluster* pela perda ARI para o modelo com η =2,1.

Os parâmetros de escala do modelo, σ^2, são apresentados na Figura 3.16, com seus intervalos HPD de 95% de probabilidade e a alocação fornecida pela perda de Binder para os modelos com η = 2,1, 5 e 100. Nesse caso, os intervalos HPD tem amplitudes que diminuem com a diminuição do valor de η. Para o modelo com η =2,1 os intervalos possuem cumprimento de no máximo 30, Enquanto para o modelo com $\eta = 100$ os intervalos são, em sua maioria, de amplitude até 150, mas dois talhões se destacam novamente: C9AM5 com amplitude maior que 300 e C9PO5, com amplitude maior que 600,

Figura 3.16: Intervalos HPD de 95% e a mediana como estimativa pontual das medidas dos parâmetros de escala, σ^2, para cada talhão. As cores são relacionadas ao *clusters* estimados pela Perda de Binder para η = 2,1; 5 e 100.

(a) η = 2,1. (b) η = 5.0. (c) η = 100.



Fonte: Elaborado pela autora.

A Tabela 3.6 mostra as estimativas para os parâmetros do kernel utilizando as medianas das distribuições *a posteriori* com seus respectivos intervalos HPD de 95% de probabilidade. Os valores das estimativas são próximos para os três modelos, bem como os limites dos intervalos HPD. O fato da estimação não apresentar grandes mudanças é coerente com nível hierárquico desses parâmetros no modelo. O kernel do PPD define a distribuição do vetor de locação e seu comportamento e as estimativas de $\boldsymbol{\mu}$ não apresentaram muita variação em relação aos três modelos.

Tabela 3.6: Estimativas pelas medianas das distribuições *a posteriori* e Intervalos HPD de 95% de probabilidade para os parâmetros do kernel do PPD, θ e σ_q.

η	θ			σ_q		
	Mediana	HPD 95%		Mediana	HPD 95%	
		Inf	Sup		Inf	Sup
2,1	0,83	0,75	0,91	127,34	125,18	128,67
5	0,84	0,76	0,92	129,25	127,29	130,46
100	0,84	0,76	0,91	126,32	124,52	128,78

Fonte: Elaborado pela autora.

3.7 Conclusões

Neste capítulo desenvolvemos um modelo de mistura finita para estimação densidade com base na distribuição Normal/Independente para os dados e Processos Pontuais por Determinante para modelar o vetor de parâmetros de locação. O modelo consegue estimar os parâmetros de locação, escala e de alocação das observações nos *clusters*, além de estimar os parâmetros do kernel do PPD. Todo o processo de estimação permite avaliação de incerteza a partir das distribuições a *a posteriori* dos parâmetros fornecidas por um MCMC proposto e implementado.

Um estudo de simulação foi conduzido no qual se verificou que o modelo retorna o número de *clusters* e a alocação para os dados simulados, além disso, apresenta bons resultados de estimação e ajuste aos dados. Apesar de não possuir uma estrutura espacial, mostrou bons resultados de ajuste para os dados de aplicação em biomassa obtido por imagens em que as observações são subáreas definidas de uma fazenda.

Para os parâmetros do kernel do PPD, θ^2 e σ_q^2, foi proposta uma configuração de dependência. A distribuição de θ^2 *a priori* é limitada tanto inferior quanto superiormente e o uso da mediana das distâncias entre os dados como valor inicial para as cadeias de θ^2, apesar de se mostrar superior ao estimado pelo MCMC, não dificulta a convergência. O parâmetro σ_q^2 apresenta valores mais elevados e necessitou de valores iniciais também maiores. O uso do número de observações por unidade de volume não se mostrou um bom valor inicial para as cadeias de σ_q^2, após ajustes prévios usou-se valores em torno de 125^2 o que ajudou a obter convergência mais rapidamente.

As cadeias das distribuições *a posteriori* são apresentadas no Apêndice B bem como as taxas de aceitação do Metropolis-Hastings.

Capítulo 4

Redução de dimensão de Variável Categórica via Modelo NIPPD

No geral, o uso de Processos Pontuais por Determinante (PPDs) em Modelos de Mistura Finita (MMFs) para agrupar níveis de variáveis categóricas representa uma abordagem promissora para identificar subgrupos significativos em grandes conjuntos de dados. Ao capturar a diversidade e a coerência entre os subgrupos, os MMFs baseados em PPD podem fornecer uma ferramenta poderosa para descobrir insights e informar a tomada de decisões em vários campos. No contexto de modelos de regressão os processos pontuais por determinante se mostram promissores, especialmente para a seleção de modelos [61, 96].

Variáveis explicativas categóricas muitas vezes não tem sua informação bem aproveitada devido à dificuldade de lidar com os muitos níveis ou categorias presentes em sua configuração. As propostas para atacar esse problema ainda são escassas e nem sempre permitem a avaliação da incerteza sobre os resultados obtidos. Uma das contribuições deste trabalho é oferecer como alternativa um modelo que permite agrupar níveis da variável categórica e, ao mesmo tempo, a obtenção das estimativas dos efeitos desta e das demais variáveis explicativas comuns aos indivíduos. A proposta é baseada na distribuição Normal/Independente, flexível e robusta em termos de *outliers* e os efeitos dos muitos níveis da variável categórica são representados por uma realização de um Processo Pontual por Determinante que possui características repulsivas e protege contra o superajuste do modelo.

4.1 Introdução

Num mundo onde *big data* é uma realidade, a redução de dimensionalidade de dados é uma necessidade inata. Uma rápida pesquisa em buscadores na internet gera dezenas de milhares resultados (Aproximadamente 36100000 resultados no Google, busca simples pelo termo “big data”). Fóruns de discussão relacionados a análise de dados estão cheios de questionamentos sobre possíveis ou melhores formas de reduzir dimensionalidade dos dados. Tratando-se de

variáveis categóricas com um número muito grande de níveis, isto tem um apelo ainda maior. Contudo, esse é um grande desafio, principalmente, se no processo da redução desse número de categorias, busca-se também uma boa interpretabilidade dos resultados ou pelo menos que o critério de redução tenha relação com características de interesse dos dados utilizados.

Usualmente, variáveis categóricas são incluídas no modelo por meio de variáveis *dummy*, na qual são construídas matrizes de zeros e uns com o número de colunas sendo o número de categorias menos 1. Para variáveis categóricas que envolvem muitos níveis, essa matrizes se tornam esparsas, levando a estimativas instáveis para os efeitos das categoria, dificultando a interpretação dos resultados. Além disto, estas matrizes com alta dimensão, ocupam muita memória computacional, causando quebras nas operações numéricas utilizadas.

Para obter melhor interpretabilidade dos resultados e melhor eficiência computacional, uma estratégia razoável é reduzir o número de categorias. Segundo [33] o agrupamento das categorias é mais vantajoso do que lidar com dados esparsos e pode trazer melhorias na convergência dos algoritmos.

Uma abordagem simples e bem estruturada, teoricamente, foi desenvolvida por [29] visando obter uma interpretação dos resultados em estudos envolvendo grandes tabelas de contingência. Nesta abordagem, algumas categorias de fatores são combinadas de tal forma que não altere as propriedades Markoviana do grafo envolvido na modelo.

A técnica denominada *codificação-alvo* [80] reduz dimensionalidade agrupando os níveis da variável categórica com base na variável resposta. [80] sugere manter apenas as categorias mais frequentes enquanto as demais são agrupadas em uma categoria genérica. Embora seja bem simples de ser aplicado e esteja implementado em vários pacotes, esse método pode causar sobreajuste e carece de uma base teórica mais sólida. O uso de critérios *ad-hoc* para agregar categorias é muito utilizado. Por exemplo, agrupar categorias com poucos indivíduos para evitar problemas computacionais como a singularidade na inversão de matrizes [99] é uma prática comum. No entanto, uma redução que permita avaliar seus efeitos sobre as inferências não é obtida facilmente [33].

Na última década, a redução de dimensionalidade em modelos de regressão tem recebido grande atenção devido a massiva criação de dados e a necessidade de técnicas eficientes para analisá-los. Uma abordagem que tem se destacado é a técnica de regularização ou penalização, tais como a regularização L1 e L2. Esses métodos podem ser usados para evitar o superajuste (*overfitting*) em espaços de alta dimensão. O mais conhecido entre esses métodos é o LASSO *(Least Absolute Shrinkage and Selection Operator)* proposto por [108]. A ideia básica é usar um algoritmo de otimização para minimizar a soma dos quadrados dos resíduos penalizada pela adição de um termo de penalidade para grandes valores absolutos dos coeficientes. Essa penalidade é controlada por um parâmetro de ajuste λ. O LASSO tende a "encolher" para zero efeitos não-significativos de variáveis preditoras, permitindo selecionar as variáveis relevantes para a análise entre inúmeras possíveis. Isto previne *overfitting*.

Alguns métodos para agrupamento, como k-médias e análise discriminante [116], são

muito utilizados na alocação de indivíduos, mas não são adequados para lidar com agrupamento de níveis de variáveis categóricas. No entanto, existem outros algoritmos de aprendizado de máquina: como Random Forest e Support Vector Machines (SVMs) [25], capazes de lidar com dados de alta dimensão e podem ser usados para tarefas como classificação e regressão. *Random Forest* propõe construir uma infinidade de árvores de decisão durante o treinamento e gerar a moda das classes (classificação) ou a previsão média (regressão) das árvores individuais. Enquanto o *Support Vector Machines* (SVMs) visa encontrar um hiperplano que separe os dados em diferentes classes. Este hiperplano é conhecido como suporte, sendo obtido maximizando sua distância até os pontos de dados mais próximos de cada classe.

Outro algoritmo de agrupamento, usado por [4], é o AMOEBA (sigla para *Multidirectional Optimum Ecotope-Based Algorithm*). Este algoritmo depende do uso de uma estatística de autocorrelação espacial local. O método identifica agrupamentos por meio de uma matriz de pesos espaciais, que, por sua vez, são vetores que identificam as unidades espaciais relacionadas e não relacionadas às unidades contíguas.

Uma abordagem Bayesiana comum para lidar com o problema de variáveis categóricas com muitos níveis é construir uma distribuição *a priori* de encolhimento para os efeitos das covariáveis. Esta distribuição *a priori* desempenha um papel semelhante às funções de penalização na estimação dos parâmetros "encolhendo" para zero os efeitos não significativos. Com esse intuito, [83] propõem uma abordagem Bayesiana para o método LASSO (*Least Absolute Shrinkage and Selection Operator*) proposto por [108]. Além de permitir a seleção de variáveis de maneira automática, com a inclusão de informações *a priori* sobre os coeficientes, é possível melhorar a precisão da estimativa e o poder de predição do modelo.

Nesta linha de métodos de regularização, um método tipo LASSO alternativo foi proposto por [46] como uma extensão do trabalho de [17]. Nesta proposta, a forma desse encolhimento consegue reduzir os níveis dentro de um fator, definindo seus efeitos como iguais, ao mesmo tempo, em que obtém a seleção de fatores zerando fatores inteiros. O uso dessa abordagem também leva à identificação de uma estrutura dentro de cada fator, pois os níveis podem ser automaticamente recolhidos para formar grupos. O método proposto por [46] também é baseado em penalidade L1 para agrupamento de categorias, contudo não é uma abordagem unificada, apresenta dois métodos diferentes: um para níveis de escala nominal e outro para escala ordinal. Na abordagem *Effect Fusion (The effect fusion prior)* [85], propõem uma nova distribuição *a priori* de regularização para, especificamente, tratar de variáveis categóricas a qual é uma generalização do método *spike-slab*. Além do encolhimento para zero de efeitos não significativos, este método também agrupa efeitos semelhantes.

Recentemente, [26] propuseram um modelo de agrupamento para redução de dimensionalidade de variáveis categóricas baseado no modelo de partição produto. Na estrutura considerada, os efeitos dos níveis da variável categórica são representados por um grafo e a técnica de particionamento desenvolvida por [107] baseada em árvores geradoras aleatórias é considerada. Este método mostrou-se bastante eficiente em comparação com métodos existentes para a

redução de dimensionalidade neste tipo de problema.

Inspirado pelos resultados obtidos pelos autores [26], o objetivo deste capítulo é propor um modelo para tratar variáveis categóricas com muitos níveis assumindo que a incerteza sobre os efeitos dos níveis desta covariável sejam ajustados por um modelo de mistura finita via Processo Pontual por Determinante. O objetivo é particionar não o conjunto de observações, mas as categorias pré-definidas desses dados em um número menor de subpopulações, ou *clusters*, com base em suas características observadas. Os PPDs oferecem uma vantagem única nesse contexto porque podem capturar tanto a diversidade quanto a coerência entre os *clusters* [54]. Especificamente, os PPDs fornecem uma maneira de modelar a tendência dos efeitos se repelirem, o que auxilia na obtenção de *clusters* distintos entre si, mas internamente homogêneos. Dadas estas características do PPD, espera-se que apenas níveis muito diferentes sejam alocados a grupos diferentes.

Um dos principais benefícios do uso de PPDs em Modelos de Mistura Finita é que eles podem ajudar a evitar o *overfitting*, problema comum em algoritmos de agrupamento e em modelos de regressão que envolvem matrizes esparsas. Ao incorporar informações sobre a diversidade e a similaridade dos *clusters*, os PPDs podem ajudar a garantir que o modelo de mistura resultante seja preciso e generalizável para novos dados.

Esse método enfatiza os efeitos principais dos preditores categóricos e utiliza estrutura de PPD's para identificar grupos de categorias. Essa abordagem é particularmente útil quando se trata de preditores que possuem um grande número de categorias, pois também permite identificar categorias que exibem um efeito distinto na variável resposta sem onerar o processo de estimação com um número excessivo de parâmetros.

No geral, o uso de PPD"s em MMF"s para agrupar níveis de variáveis categóricas representa uma abordagem promissora para identificar subgrupos significativos em grandes conjuntos de dados. Ao capturar a diversidade e a coerência entre os subgrupos, os MMF's baseados em PPD podem fornecer uma ferramenta poderosa para obter descobertas e apoiar a tomada de decisões em vários campos. No contexto de modelos de regressão, os processos pontuais por determinante se mostram promissores, especialmente para a seleção de modelos [61, 96].

Outra contribuição deste trabalho, é que diferentemente do que é considerado em [26], assumiremos a distribuição Normal/Independente para modelar o comportamento das variáveis respostas. As caudas pesadas desta distribuição proporcionam mais flexibilidade para lidar com possíveis *outliers* dentro dos *clusters*. O modelo proposto é apresentado na Seção 4.2.1, na qual também se discute a implementação computacional. Na Seção 4.3 apresenta comparações da proposta com outras da literatura: um modelo Bayesiano via Partição Produto (PPRM) proposto por [26] e o modelo LASSO Bayesiano proposto por [83]. Estes modelos serão utilizados na análise de dados de rendimento semestral global médio (RSGM) dos estudantes da UFMG ingressantes no ano de 2008, onde pretende-se investigar que cursos tem efeitos similares no rendimento do aluno. Além desta comparação, avalia-se o desempenho do modelo proposto para diferentes especificações dos parâmetros.

Para a implementação do LASSO foi utilizado o pacote *monomvn* do R [91] e para o PPRM utiliza-se o pacote disponibilizado pelos autores.

4.2 Modelo proposto

Seja $\mathbf{Y} = (y_1, \ldots, y_N)' \subset \mathbb{R}^N$ um vetor coluna de ordem N contendo as variáveis respostas de N indivíduos selecionados independentemente. Seja $\mathbf{X}$ a matriz de delineamento de ordem $N \times p$ contento a informação das p variáveis preditoras (covariáveis) para os N indivíduos onde a i-ésima linha corresponde ao valor das covariáveis para o indivíduo i. Denote por $\boldsymbol{\beta}$ o vetor dos efeitos das covariáveis com dimensão $p \times 1$. Será assumido que $\mathbf{X}$ contem informações sobre as variáveis quantitativas e sobre variáveis categóricas com poucos níveis.

Admita que se tenha uma variável categórica com J categorias, em que J é grande. Define-se $\mathbf{V}^*$ uma matriz de ordem $N \times J$ cuja coordenada $v_{ij} = 1$ se o i-ésimo indivíduo está na categoria j desta variável e $v_{ij} = 0$, caso contrário, para todo $i = 1, \ldots, N$ e $j = 1, \ldots, J$, com $\sum_{j=1}^{J} v_{ij} = 1$. Denote por $\boldsymbol{\alpha}^* = (\alpha_1^*, \ldots, \alpha_J^*)'$, o vetor dos efeitos de cada uma das J categorias. Um modelo de regressão linear será adotado da seguinte forma:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{V}^*\boldsymbol{\alpha}^* + \boldsymbol{\varepsilon}, \tag{4.1}$$

em que os erros aleatórios são independentes e identicamente distribuídos (i.i.d.) com uma distribuição centrada em zero. Sob esta estrutura, assume-se que a variável resposta para o indivíduo i que pertence à categoria j da variável categórica de interesse é tal que

$$y_{ij} = \mathbf{X}_i\boldsymbol{\beta} + \alpha_j^* + \varepsilon_{ij}.$$

A meta neste trabalho é reduzir a dimensionalidade da matriz $\mathbf{V}^*$ agrupando efeitos α_i^* que sejam similares. Admite-se que $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ são parâmetros comuns a todos os indivíduos e a variância dentro dos grupos é σ_y^2 para todos os grupos.

Para reduzir a dimensionalidade de $\mathbf{V}^*$, admita que as observações são particionadas aleatoriamente em K, $K < J$, grupos de indivíduos $G_1, \ldots, G_K$. Denote por $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)$, $K < J$, o vetor de efeitos dos níveis agregados da variável categórica e $\mathbf{V}$ é uma matriz $N \times K$ de alocação dos indivíduos para as categorias agregadas. Se o indivíduo i pertence à categoria agregada k, a i-ésima linha da matriz $\mathbf{V}$ tem entradas zeros em todas as colunas, exceto na coluna k onde assume o valor 1. Assim, tem-se que $\sum_{k=1}^{K} V_{ik} = 1$. Ou seja, $V_{ik} = 1$, se, e somente se, a categoria a qual o indivíduo pertence foi alocada ao *cluster* k, $k = 1, \ldots, K$.

Para um ajuste mais flexível e robusto em relação a *outliers*, será assumido que os erros ε_{ij} são i.i.d. com distribuição normal/independente centrada em zero, com parâmetro de escala σ_y^2 e parâmetro de forma η. A distribuição dos erros será denotada por $\varepsilon_{ij} \overset{iid}{\sim} NI(0, \sigma_y^2, \eta)$.

Dentro desta estrutura e, considerando a relação linear entre a variável resposta e as covariáveis dada em (4.1), assume-se que a resposta para o i-ésimo indivíduo é modelada como uma mistura finita de distribuições na família Normal/Independente como segue

$$\mathbf{y}_{ij}|u_{ij},\ldots \sim \sum_{k=1}^{K} w_k N(y_{ij}|\mathbf{X_i}\boldsymbol{\beta}+\alpha_k,\sigma_y^2 u_{ij}^{-1}), \tag{4.2}$$

em que u_{ij} é uma variável aleatória não-negativa considerada na representação estocástica da família NI para o indivíduo i na categoria j original, $w_k \in (0,1)$ representa o peso da componente k da mistura e $N(y|a,b)$ denota a densidade da distribuição normal com média a e variância b avaliada no ponto y. A representação estocástica da família NI discutida no Capítulo 1 será utilizada aqui. Este resultado é muito útil na estrutura hierárquica do modelo e, principalmente, na implementação computacional, para fazer inferências e na geração de amostras de distribuições desta família [65].

Para incluir a estrutura de agrupamento no modelo, será utilizado o Processo Pontual por Determinante para a modelagem do vetor de parâmetros $\boldsymbol{\alpha}$ cujas componentes representam os efeitos dos níveis da variável categórica com muitas categorias. Será assumido que $u_{ij} \overset{iid}{\sim} Gama(\eta/2,\eta/2)$, ou seja, o Modelo t de Student para a resposta dos indivíduos.

Na próxima seção faremos uma apresentação hierárquica do modelo proposto apresentando as distribuições *a priori* para os demais parâmetros .

4.2.1 Representação hierárquica do modelo proposto

O vetor de observações pode ser escrito para refletir a estrutura imposta pela variável categórica da seguinte forma $\mathbf{Y} = (\mathbf{y}_1,\ldots,\mathbf{y}_J)$ em que $\mathbf{y}_j = (y_{1j},\ldots,y_{n_j j})$, assim cada y_{ij} é a observação do i-ésimo indivíduo da j-ésima categoria nos dados, $i = 1,\ldots,n_j$ e $j = 1,\ldots,J$. Além disso, o número de observações é dado por $N = \sum_{j=1}^{J} n_j$, em que n_j é o número de observações na j-ésima categoria. O agrupamento é relacionado à variável categórica, logo, a partição se dá sobre as J categorias e o vetor aleatório latente $\mathbf{z} = (z_1,\ldots,z_J)$ é responsável pela alocação. Sua distribuição *a priori* é discreta, e tal que $P(z_j = k|K) = w_k, k = 1,\ldots,K$.

Para apresentar hierarquicamente o modelo de regressão via mistura finita de distribuições na Família de Distribuições Normal/Independente será considerada variável misturadora, $\mathbf{u} = (u_{11},\ldots,u_{n_1 1},u_{12},\ldots,u_{n_2 2},\ldots,u_{1J},\ldots,u_{n_J J})$. Desta forma, o modelo proposto é

$$\begin{aligned}
\mathbf{y}_{ij}|\mathbf{X}, \boldsymbol{\beta}, z_j = k, \boldsymbol{\alpha}_k, u_i, \sigma_y^2, K &\overset{ind.}{\sim} N(\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\alpha}_k, u_{ij}^{-1}\sigma_y^2); && (4.3)\\
\boldsymbol{\beta}|\sigma_y^2 &\sim N_p(\boldsymbol{\mu}_\beta, \sigma_y^2\boldsymbol{\Sigma}_\beta); && (4.4)\\
P(z_j = k|\mathbf{w}, K) &= w_k, k = 1, \ldots, K; && (4.5)\\
\mathbf{w} = (w_1, \ldots, w_K) &\sim Dirichlet(\boldsymbol{\delta}), \boldsymbol{\delta} = (\delta_1, \ldots, \delta_K); && (4.6)\\
\mathbf{u} = (u_1, \ldots, u_N) \text{ com } u_i &\overset{\text{iid}}{\sim} Gama(\eta/2, \eta/2); && (4.7)\\
\sigma_y^2 &\sim IG(a_0, b_0), && (4.8)\\
\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)|\theta^2, \sigma_q^2, K &\sim PPD(C, \theta^2, \sigma_q^2); && (4.9)\\
\theta^2|\sigma_q^2 &\sim Gama_T(a_1, b_1, T), \theta^2 \in T = (0; 2\sigma_q^2); && (4.10)\\
\sigma_q^2 &\sim Gama(a_2, b_2); && (4.11)
\end{aligned}$$

em que $\mathbf{X}$ é matriz $(N \times p)$ de covariáveis comuns a todos os indivíduos, $\boldsymbol{\beta}$ é o vetor de efeitos destas covariáveis, α_k é o efeito da k-ésima categoria da variável categórica e funciona como um intercepto aleatório do modelo e seu valor depende da alocação z_j segundo o grupo ao qual foi designado. Para evitar problemas de identificabilidade, não é considerado um intercepto relacionado no vetor $\boldsymbol{\beta}$, ou seja, não há um β_0 nem uma coluna relacionada a ele na matriz $\mathbf{X}$. Cada categoria original da variável com muitos níveis terá seu próprio intercepto.

O agrupamento é relacionado aos efeitos, $\boldsymbol{\alpha}$, das categorias, mas as unidades de observação são os indivíduos. Para facilitar a notação e manuseio computacional, é definida a seguir uma estrutura matricial que converte o agrupamento das categorias para o nível do indivíduo. A matriz $\mathbf{Z}$, de dimensão $(N \times K)$, será responsável pela distribuição dos componentes do vetor $\boldsymbol{\alpha}$ para os indivíduos. Neste caso, $\mathbf{Z} = [Z_{(ij)k}]$, $j = 1, 2, \ldots, J$ e $i = 1, 2, \ldots, n_j$, é uma matriz que tem como componentes variáveis indicadoras, ou seja, $Z_{(ij)k} = 1$, se o i-ésimo indivíduo da categoria original j for agrupado no *cluster* k, com $k = 1, \ldots, K$, e $Z_{(ij)k} = 0$, caso contrário.

A partir da estrutura definida e considerando a família NI para a variável resposta, a função de verossimilhança é dada por

$$\begin{aligned}
L(\boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\alpha}, \mathbf{u}, \sigma_y^2, \mathbf{w}, \theta^2, \sigma_q^2|\mathbf{y}) &= (2\pi)^{-N/2}\det(\mathbf{U}^{-1}\sigma_y^2)^{-1/2} && (4.12)\\
&\times \exp\Big\{-\frac{1}{2\sigma_y^2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})\Big\},
\end{aligned}$$

em que $\mathbf{U} = diag(\mathbf{u})$, matriz diagonal dos componentes do vetor $\mathbf{u}$.

No modelo, $\boldsymbol{\alpha}$ é o vetor de efeitos da variável categórica com seus níveis já agrupados e é modelado como a realização de um Processo Pontual por Determinante, ou seja, $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_K)|\ K, \theta^2, \sigma_q^2 \sim PPD(C, \theta, \sigma_q)$. A distribuição de $\boldsymbol{\alpha}$, depende da função kernel, C, e, consequentemente, dos seus hiperparâmetros, θ^2 e σ_q^2, além do número de grupos, K. Sua função densidade de probabilidade *a priori* é dada por

$$p(\boldsymbol{\alpha}|\theta, \sigma_q, K) = \frac{\det(C_{\theta,\sigma_q}(\boldsymbol{\alpha}))}{\prod_{\mathbf{h}} (\lambda_{\mathbf{h}}(\theta, \sigma_q) + 1)}, \tag{4.13}$$

com $C_{\theta,\sigma_q}(\boldsymbol{\alpha}) = [C_{kl}]_{k,l\in\{1,\ldots,K\}}$, definida como um operador kernel, função não negativa e simétrica utilizada para obter os elementos C_{kl}. O kernel utilizado neste trabalho será considerado o kernel exponencial quadrático apresentado no Capítulo 3.

Para um k qualquer, $k = 1, 2, \ldots, K$, a partir da da partição do vetor $\boldsymbol{\alpha} = (\alpha_k, \boldsymbol{\alpha}_{-k})$, em que $\boldsymbol{\alpha}_{-k}$ é o vetor de efeitos da variável categórica sem o componente α_k, a matriz $C_{\theta,\sigma_q}(\boldsymbol{\alpha})$ também pode ser particionada e o seu determinante pode ser obtido utilizando-se a Identidade de Schur na decomposição de (4.13). Dessa forma, o núcleo da distribuição *a priori* de α_k dado o restante do vetor de efeitos, α_{-k}, é

$$p(\alpha_k|\boldsymbol{\alpha}_{-k}, \theta, \sigma_q, K) \propto C_{\alpha_k} - C(\alpha_k, \boldsymbol{\alpha}_{-k}) C^{-1}_{\boldsymbol{\alpha}_{-k}} C(\alpha_k, \boldsymbol{\alpha}_{-k})', \tag{4.14}$$

em que $C_{\boldsymbol{\alpha}_{-k}} = C_{\theta,\sigma_q}(\boldsymbol{\alpha}_{-k}\boldsymbol{\alpha}_{-k})$ é matriz $(K-1)\times(K-1)$.

A função densidade de probabilidade *a priori* dos parâmetros do Kernel θ^2 e σ_q^2 são, respectivamente, distribuições Gama e Gama Truncada. Suponha $\theta \sim Gama(a, b, t)$, em que a é o parâmetro de forma, b é o parâmetro de escala e t é o intervalo de truncamento de θ. Sua função densidade de probalidade é dada por:

$$f(\theta^2|a, b, T) = \begin{cases} \dfrac{\theta^{-a-1} exp\{-\theta/b\}}{b^a \Gamma(a)}, & \text{se } \theta \in T = (0, 2\sigma_q^2). \\ 0, & \text{caso contrário.} \end{cases} \tag{4.15}$$

e sua esperança e a variância são dadas por $E(\theta^2) = \dfrac{\Gamma_T(a+1)}{b\Gamma_T(a)}$ e $Var(\theta^2) = \dfrac{1}{b^2}\Big[\dfrac{\Gamma_T(a+2)}{\Gamma_T(a)} - \dfrac{\Gamma_T(a+1)}{\Gamma_T(a)}\Big]$ é a função Gama Truncada Superior: $\Gamma_T(a) = \int_T (\theta^2)^\alpha e^{-\theta/b} d\theta^2$.

4.2.2 Estrutura Bayesiana para o Modelo NIPPD para Redução de Dimensão da Variável Categórica

A distribuição conjunta relacionada ao modelo apresentado na seção anterior é dada por

$$\begin{aligned} p(\mathbf{y}, \boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\alpha}, \mathbf{u}, \sigma_y^2, \mathbf{w}, \theta^2, \sigma_q^2) &= p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\alpha}, \mathbf{u}, \sigma_y^2) p(\boldsymbol{\beta}|\sigma_y^2) p(\mathbf{z}|\mathbf{w}) p(\mathbf{w}) \\ &\times\ p(\boldsymbol{\alpha}|\theta^2, \sigma_q^2) p(\theta^2|\sigma_q^2) p(\sigma_q^2) p(\sigma_y^2). \end{aligned} \tag{4.16}$$

A distribuição condicional completa *a posteriori* relacionada ao agrupamento de níveis da variável categórica e responsável pela alocação, $\mathbf{z} = (z_1, \ldots, z_J)$, uma vez que as componentes são consideradas independentes, é obtida diretamente pela normalização da distribuição

proporcional a seguir

$$\begin{aligned} P(z_j = k|\ldots) &\propto w_K(2\pi)^{-n_j/2}\det(\mathbf{U}_j^{-1}\sigma_y^2)^{-1/2} \\ &\times \exp\Big\{-\frac{1}{2\sigma_y^2}(\mathbf{y}_j - \mathbf{X}_j\boldsymbol{\beta} - \mathbf{Z}_j\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})\Big\}. \end{aligned} \tag{4.17}$$

A constante de normalização é obtida pela soma das quantidades acima para todo $k = 1, \ldots, K$ e para cada uma das j categorias.

Para os pesos $\mathbf{w}$ a distribuição Dirichlet é conjugada à distribuição multinomial considerada na função de verossimilhança, assim a sua distribuição condicional completa é dada por

$$\mathbf{w}|\ldots \sim Dir(\delta_1 + n_1, \ldots, \delta_k + n_K), \tag{4.18}$$

em que $n_k = \sum_{i=1}^{N} I_{(z_j=k)}$.

Dada a representação estocástica da família NI, a distribuição condicional completa dos efeitos das covariáveis $\boldsymbol{\beta}$ que são comuns a todos os indivíduos tem forma fechada sendo dada por

$$\boldsymbol{\beta}|\sigma_y^2, \ldots \sim N_p(\mathbf{M}_\beta, \mathbf{S}_\beta), \tag{4.19}$$

em que a média é $\mathbf{M}_\beta = \left(\boldsymbol{\Sigma}_\beta^{-1} + \mathbf{X}'\mathbf{U}^{-1}\mathbf{X}\right)^{-1}\left(\boldsymbol{\Sigma}_\beta^{-1}\boldsymbol{\mu}_\beta + \mathbf{X}'\mathbf{U}^{-1}\dagger - \mathbf{X}'\mathbf{U}^{-1}\mathbf{Z}\boldsymbol{\alpha}\right)^{-1}$ e sua variância é $\mathbf{S}_\beta = \sigma_y^2\left(\boldsymbol{\Sigma}_\beta^{-1} + \mathbf{X}'\mathbf{U}^{-1}\mathbf{X}\right)^{-1}$.

Já a distribuição condicional completa da variável misturadora, u_{ij}, $i = 1, \ldots, n_j$, $j = 1, \ldots, J$, componente do vetor $\mathbf{u}$, é dada por

$$u_{ij}|\ldots \sim Gama\left(\frac{\eta + 1}{2}, \frac{\eta + S_{ij}}{2}\right), \tag{4.20}$$

em que $S_{ij} = \dfrac{(y_{ij} - \mathbf{X}_{ij}\boldsymbol{\beta} - \alpha_{z_j})^2}{\sigma_y^2}$ e $\mathbf{X}_{ij}$ é a linha da matriz de covariáveis relacionadas a observação do i-ésimo indivíduo da j-ésima categoria.

O parâmetro de escala, σ_y^2, possui distribuição Inversa Gama *a priori*, ou ainda, $\frac{1}{\sigma_y^2} \sim Gama(a_0, b_0)$ e sua condicional completa é dada por

$$\frac{1}{\sigma_y^2}\bigg|\ldots \sim Gama\Big(a_0 + N/2 + p/2,\ b_0 + \mathcal{S}/2 + B/2\Big), \tag{4.21}$$

em que N é o tamanho total da amostra, p é o comprimento do vetor $\boldsymbol{\beta}$, $\mathcal{S} = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})$ e $\mathbf{B} = \left(\boldsymbol{\beta} - \boldsymbol{\mu}_\beta\right)'\boldsymbol{\Sigma}_\beta^{-1}\left(\boldsymbol{\beta} - \boldsymbol{\mu}_\beta\right)$.

Para o vetor dos efeitos de *clusters* específico da variável categórica, a distribuição condicional completa conjunta *a posteriori* é

$$\begin{aligned} p(\boldsymbol{\alpha}|\ldots) &\propto \det(C_{\theta,\sigma_q}(\boldsymbol{\alpha}))L(\boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\alpha}, \mathbf{u}, \sigma_y^2, \mathbf{w}, \theta^2, \sigma_q^2|\mathbf{y}) \\ &\propto \det(C_{\theta,\sigma_q}(\boldsymbol{\alpha}))(2\pi)^{-N/2}\det(\mathbf{U}^{-1}\sigma_y^2)^{-1/2} \\ &\times \exp\Big\{-\frac{1}{2\sigma_y^2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})\Big\}. \end{aligned} \tag{4.22}$$

Utilizando o núcleo da distribuição *a priori* de α_k dado em (4.14), é possível obter a distribuição condicional completa de α_k dado o restante do vetor de efeitos, $\boldsymbol{\alpha_{-k}}$, e os demais parâmetros. Assim, para um k qualquer,

$$\begin{aligned} p(\alpha_k|\boldsymbol{\alpha}_{-k},\ldots) &\propto \left(C_{\alpha_k} - C(\alpha_k,\boldsymbol{\alpha}_{-k})C^{-1}_{\boldsymbol{\alpha}_{-k}}C(\alpha_k,\boldsymbol{\alpha}_{-k})'\right) \\ &\times \exp\Big\{-\frac{1}{2}\Big(\boldsymbol{\alpha}_k - M_k\Big)' V_k^{-1}\Big(\boldsymbol{\alpha}_k - M_k\Big)\Big\} \end{aligned} \tag{4.23}$$

em que $M_k = [\sum_{j:z_j=k} u_{ij}]^{-1} \sum_{j:z_j=k} (\mathbf{y}_j - \mathbf{X}_j\boldsymbol{\beta})\mathbf{U}_j$, $V_k = \sigma_y^2[\sum_{j:z_j=k} u_{ij}]^{-1}$ e $\mathbf{U}_j = diag(u_{1j},\ldots,u_{n_jj})$, com a verossimilhança sendo proporcional ao núcleo de um distribuição normal unidimensional com média M_k e matriz de variância V_k.

Para os parâmetros do kernel do PPD as distribuições condicionais completas são dadas por

$$p(\theta^2|\ldots) \propto \frac{\det(C_{\boldsymbol{\alpha},\theta^2,\sigma_q^2})}{\prod_{h=1}^{\infty}(\lambda_h(\theta^2,\sigma_q^2)+1)} p(\theta^2|\sigma_q^2), \tag{4.24}$$

para $\theta^2 \in t_1 = \left(0, 2\sigma_q^2\right)$ em que $p(\theta^2|\sigma_q^2)$ é a densidade da distribuição $Gama_T(a_1,b_1,t_1)$,e

$$p(\sigma_q^2|\ldots) \propto \frac{\det(C_{\boldsymbol{\alpha},\theta^2,\sigma_q^2})}{\prod_{h=1}^{\infty}(\lambda_h(\theta^2,\sigma_q^2)+1)} p(\sigma_q^2|\theta^2), \tag{4.25}$$

em que $\sigma_q^2 \sim Gama_T(a_2,b_2,t_2)$, $\sigma_q^2 \in t_2 = \left(\frac{(\theta\sqrt{\pi})^{1/D}}{\sqrt{2\pi}}, \infty\right)$. Além disso, conjuntamente

$$\begin{aligned} p(\theta^2,\sigma_q^2|\ldots) &= p(\boldsymbol{\alpha}|\theta^2,\sigma_q^2,\ldots)p(\theta^2|\sigma_q^2,\ldots)p(\sigma_q^2|\ldots) \\ &\propto \frac{\det(C_{\boldsymbol{\alpha},\theta^2,\sigma_q^2})}{\prod_{h=1}^{\infty}(\lambda_h(\theta^2,\sigma_q^2)+1)} p(\theta^2|\sigma_q^2)p(\sigma_q^2), \end{aligned} \tag{4.26}$$

com $\theta^2 \in t_1 = \left(0, 2\sigma_q^2\right)$ e $\sigma_q^2 > 0$.

O processo de estimação dos parâmetros é apresentado na Seção 4.2.3 com a estrutura do MCMC para obtenção das amostras das distribuiçõesa *posteriori* do modelo.

4.2.3 MCMC para o MNIPPD

As distribuições condicionais completas são conhecidas para os parâmetros de locação $\boldsymbol{\beta}$ e variabilidade σ_y^2 comuns a todos os indivíduos, para os pesos da mistura $\mathbf{w}$, para o vetor de alocação $\mathbf{z}$ e para a variáveis misturadoras u_i A amostragem de valores destes parâmetros é realizada usando Amostrador de Gibbs no MCMC. Para o parâmetro $\boldsymbol{\alpha}$ modelado com um PPD e os parâmetros θ^2 e σ_q^2 do kernel do PPD, para os quais a distribuição condicional completa não têm forma fechada conhecida, serão utilizados passos de Metropolis-Hastings.

A amostragem dos parâmetros do kernel foi mais desafiadora. Não há muitos estudos sobre sua estimação, principalmente, na forma das funções de similaridade e de qualidade que são utilizadas neste trabalho. Para melhorar o aspecto da correlação entre θ^2 e σ_q^2, eles foram amostrados em bloco. Além disso, na estabilização das cadeias foi utilizado Metropolis-Hastings Adaptativo, com a variância da proposta sendo dependente da variabilidade de valores de passos anteriores da cadeia. Para valores iniciais foram utilizadas estimativas tomadas de processos pontuais homogêneos com intensidade constante, mas uma avaliação parcial das cadeias pode ser utilizada para direcionar os valores.

Em cada iteração do MCMC, gera-se amostras *a posteriori* de $\hat{\Theta} = (\mathbf{z}, \mathbf{w}, \sigma_y^2, \mathbf{u}, \boldsymbol{\beta}, \boldsymbol{\alpha}, \theta^2, \sigma_q^2)$. Os valores iniciais utilizados são fixados. Os hiperparâmetros $\mathbf{H} = (\boldsymbol{\delta}, a_0, b_0, \eta, \mu_\beta, \Sigma_\beta, a_1, b_1, a_2, b_2)$ são considerados conhecidos.

Descrição: Pseudo código para MCMC do modelo NIPPD.
Entrada: $\mathbf{X}$, $\mathbf{y}$, V_{cat}, $\mathbf{H}$
Saída: $\{\Theta^1, \ldots, \Theta^M\}$

Inicialize a cadeia: K_{max} e Θ^0 ▷ *inicio*
Faça $m = 1$
enquanto $m < M$ **faça** ▷ *x*
AMOSTRE $\mathbf{z}^{(m)}|\mathbf{y}, \mathbf{X}, \Theta^{(m-1)}, \mathbf{H}$, da densidade (4.17);
ATUALIZE $\mathbf{Z}^{(m)}$ a partir de $\mathbf{z}^{(m)}$;
AMOSTRE $\mathbf{w}^{(m)}|\mathbf{z}^{(m)}, \mathbf{H}$, da densidade (4.18);
AMOSTRE ${\sigma_y}^{2(m)}|\mathbf{y}, \mathbf{X}, \mathbf{z}^{(m)}, \boldsymbol{\beta}^{(m-1)}, \boldsymbol{\alpha}^{(m-1)}, \mathbf{H}$, da densidade (4.21);
AMOSTRE $\boldsymbol{\beta}^{(m)}|\mathbf{y}, \mathbf{X}, \mathbf{z}^{(m)}, \sigma_y^{(m)}, \boldsymbol{\alpha}^{(m-1)}, \mathbf{H}$, da densidade (4.19);
AMOSTRE $\mathbf{u}^{(m)}|\mathbf{y}, \mathbf{X}, \mathbf{z}^{(m)}, \boldsymbol{\beta}^{(m)}, {\sigma_y}^{2(m)}, \boldsymbol{\alpha}^{(m-1)}, \mathbf{H}$, da densidade (4.20);

para k de 1 até K **faça**
$\alpha_k^{(m)}|\mathbf{y}, \mathbf{X}, \mathbf{z}^{(m)}, \boldsymbol{\beta}^{(m)}, {\sigma_y}^{2(m)}, \mathbf{u}^{(m)}, \theta^{2(m-1)}, \sigma_q^{2(m-1)}, \mathbf{H}$, da densidade (4.23);
fim para

AMOSTRE de $(\theta^{2(m)}, \sigma_q^{2(m)})|\boldsymbol{\alpha}^{(m)}$ conjuntamente da distribuição (4.26);

FAÇA $m = m + 1$;
fim enquanto
Imprima $\{\Theta^{(1)}, \ldots, \Theta^{(m)}\}$. ▷ *final*

Atualização dos componentes do vetor de parâmetros de locação e os parâmetros do kernel dependem de passos de Metropolis-Hastings e serão atualizados a partir do núcleo da distribuição condicional completa conjunta de (θ^2, σ_q^2) dada em (4.16). Para amostrar de $\boldsymbol{\alpha} =$

$(\alpha_1, \ldots, \alpha_K)$, na m-ésima iteração, seguem-se os seguintes passos:

1. Obtenha um valor proposto $\boldsymbol{\alpha}_k^{(n)}$, n de novo, para $k = 1$, a partir de uma distribuição $N_D(\boldsymbol{\alpha}_k^{(m-1)}, \sigma^2_{pro})$, em que $\boldsymbol{\alpha}_k^{(o)}$ é o valor do k-ésimo componente na $(m-1)$-ésima iteração, anterior, σ^2_{pro} é a variância da proposta.

2. Calcule a probabilidade de aceitação do valor proposto.

   a) Se o grupo k está vazio, $n_k = 0$, então a probabilidade de aceitação é baseada no núcleo da distribuição *a priori* dada em (4.14);

   $$\rho(\boldsymbol{\alpha}_k^{(o)}, \boldsymbol{\alpha}_k^{(n)}) = \frac{C_{\boldsymbol{\alpha}_k^{(n)}} - C(\boldsymbol{\alpha}_k^{(n)}, \boldsymbol{\alpha}_{-k}) C^{-1}_{\boldsymbol{\alpha}_{-k}} C(\boldsymbol{\alpha}_k^{(n)}, \boldsymbol{\alpha}_{-k})^\top}{C_{\boldsymbol{\alpha}_k^{(o)}} - C(\boldsymbol{\alpha}_k^{(o)}, \boldsymbol{\alpha}_{-k}) C^{-1}_{\boldsymbol{\alpha}_{-k}} C(\boldsymbol{\alpha}_k^{(o)}, \boldsymbol{\alpha}_{-k})^\top}$$

   b) Se o grupo k é não vazio, $n_k > 0$, então a probabilidade de aceitação é baseada no núcleo da distribuição condicional completa em (4.23):

   $$\rho(\boldsymbol{\alpha}_k^{(o)}, \boldsymbol{\alpha}_k^{(n)}) = \frac{C_{\boldsymbol{\alpha}_k^{(n)}} - C(\boldsymbol{\alpha}_k^{(n)}, \boldsymbol{\alpha}_{-k}) C^{-1}_{\boldsymbol{\alpha}_{-k}} C(\boldsymbol{\alpha}_k^{(n)}, \boldsymbol{\alpha}_{-k})'}{C_{\boldsymbol{\alpha}_k^{(o)}} - C(\boldsymbol{\alpha}_k^{(o)}, \boldsymbol{\alpha}_{-k}) C^{-1}_{\boldsymbol{\alpha}_{-k}} C(\boldsymbol{\alpha}_k^{(o)}, \boldsymbol{\alpha}_{-k})'} lr$$

   em que $lr = \frac{l(\boldsymbol{\alpha}_k^{(n)})}{l(\boldsymbol{\alpha}_k^{(o)})}$ é a razão das verossimilhanças calculadas no valor proposto e no valor da $(m-1)$-ésima iteração.

3. Amostre p de uma distribuição $Unif(0,1)$ e se $p < min\{1, \rho(\boldsymbol{\alpha}_k^{(o)}, \boldsymbol{\alpha}_k^{(n)})\}$ aceita-se $\boldsymbol{\alpha}_k^{(n)}$ como valor atual da cadeia, caso contrário, repete-se o valor anterior na iteração atual.

4. Repita 6.(a) - 6.(c) para $k = 2, \ldots, K$.

Atualização em bloco dos componentes do kernel do PPD depende da distribuição conjunta dada em (4.26), e é feita via um passo de Metropolis-Hastings Adaptativo (MHA) na estrutura do MCMC. Esta estratégia permitiu melhorar a estabilidade das cadeias geradas. Denote por $\theta^{2(m)}$ e $\sigma_q^{2(m)}$ o valor gerado para os parâmetros θ^2 e σ_q^2 na m-ésima iteração.

1. Obter a variância da Proposta. Para o MHA a variância da proposta, V_{prop}, depende das amostras anteriores. Para os parâmetros do Kernel foi utilizada a variância das 100 últimas observações, V_{100}. Antes da cadeia completar 100 iterações um valor fixo foi atribuído V_{fix}, a partir da centésima iteração a variância da proposta foi uma média entre V_{100} e V_{fix}, permitindo ainda algum controle sobre a variância da proposta.

2. Obter valores propostos $\theta^{2(n)}$ e $\sigma_q^{2(n)}$, novos, a partir das distribuições $\theta^2_{prop} \sim N(\theta^{2(m-1)}, V_{prop})$ e $\sigma_q^{2(prop)} \sim N(a_2, a_2)$, respectivamente, com a_2 e b_2 hiperparâmetros. Neste caso, $\theta^{2(m-1)}$ é o valor da cadeia de θ^2 na $(m-1)$-ésima iteração, antigo, e $\eta_{1(prop)}$ é a variância da proposta. Como foi utilizado o MH adaptativo, essa variância depende dos valores anteriores da cadeia. Enquanto, $\sigma_q^{2(m-1)}$ é o valor da cadeia de σ^2 na $(m-1)$-ésima iteração, antigo, e $\eta_{2(prop)}$ é a variância da proposta.

3. Calcular a probabilidade de aceitação dos valores propostos com base na distribuição conjunta do modelo (4.26) e $C(\cdot)$, matriz kernel calculada para os valores atuais do parâmetro de locação, $\boldsymbol{\alpha}$ para os grupos não vazios.

   Para o bloco $(\theta^2\sigma_q^2)$, calcula-se

$$\begin{aligned}\rho((\sigma_q^{2(o)},\theta_q^{2(o)}),(\sigma_q^{2(n)},\theta_q^{2(n)})) &= \frac{det[C(\boldsymbol{\alpha},\theta^{2(n)},\sigma_q^{2(n)})]}{det[C(\boldsymbol{\alpha},\theta^{2(o)},\sigma_q^{2(o)})]}\frac{\prod_{h=1}^{\infty}(\lambda_h(\theta^{2(o)},\sigma_q^{2(o)})+1)}{\prod_{h=1}^{\infty}(\lambda_h(\theta^{2(n)},\sigma_q^{2(n)})+1)}\\ &\times \frac{p(\theta^{2(n)},\sigma_q^{2(n)})}{p(\theta^{2(o)},\sigma_q^{2(o)})}\frac{q(\theta^2,\sigma_q^2)}{q(\theta^2,\sigma_q^2)},\end{aligned}$$

   em que $\dfrac{p(\theta^{2(n)},\sigma_q^{2(n)})}{p(\theta^{2(o)},\sigma_q^{2(o)})}$ é a razão da distribuição *a priori* conjunta dada em (4.26) calculada nos valores propostos e nos valores da $(m-1)$-ésima iteração. Enquanto

$$\frac{q(\theta^{2(n)},\sigma_q^{2(n)}|\theta^{2(o)},\sigma_q^{2(o)})}{q(\theta^{2(o)},\sigma_q^{2(o)}|\theta^{2(n)},\sigma_q^{2(n)})}$$

   é a razão da distribuição proposta utilizada para gerar valores de (θ^2,σ_q^2), neste caso distribuições normais truncadas no zero e obedecendo à restrição.

4. Amostrar p de uma distribuição $Unif(0,1)$ e se $p < min\{1,\rho((\sigma_q^{2(o)},\theta_q^{2(o)}),(\sigma_q^{2(n)},\theta_q^{2(n)}))$ aceita-se o bloco $\theta^{2(n)}$ e $\sigma_q^{2(n)}$ como valores das cadeias, caso contrário, repetem-se os valores anteriores na iteração atual.

4.3 Aplicação: Análise do RSGM dos alunos da UFMG

A Universidade Federal de Minas Gerais (UFMG) é uma das principais universidades públicas do Brasil e tem como objetivo promover a excelência acadêmica e contribuir para o desenvolvimento social e cultural do país. Para alcançar esses objetivos, é fundamental que a UFMG tenha um sistema eficaz de acompanhamento de desempenho dos alunos, que permita identificar os principais desafios enfrentados pelos estudantes e propor soluções para melhorar a qualidade da educação oferecida pela instituição.

Uma das principais maneiras de acompanhar o desempenho dos alunos é por meio de avaliações regulares, que permitem que os professores e coordenadores de curso avaliem o conhecimento adquirido pelos alunos ao longo do semestre ou ano letivo. A UFMG conta com um sistema de acompanhamento acadêmico que permite que os estudantes consultem suas notas e outras informações relevantes sobre seu desempenho acadêmico, como o Rendimento Semestral Global Médio (RSGM). Esse é um índice atualizado semestralmente que considera,

além das notas, as frequências nas disciplinas, sendo calculado como uma média ponderada do desempenho acadêmico em cada semestre.

Sendo uma das maiores universidades do Brasil, a UFMG tem em seu corpo discente características socio-econômicas diversas que podem influenciar o desempenho acadêmico. A universidade possui ações afirmativas no âmbito de reservas de vagas, mas também de permanência dos alunos, como bolsas e programas de apoio a estudantes em situação de risco de evasão ou expostos a risco social. No entanto, acompanhar o desempenho dos alunos sempre é necessário para verificar a eficácia destas ações.

Os dados analisados são referentes a 4600 alunos da Universidade Federal de Minas Gerais - UFMG que ingressaram no curso de graduação no ano de 2008. Destes, 71 alunos, possuem valores faltantes, restando 4529 discentes na amostra. Estes dados foram fornecidos pela Pró-reitoria de Graduação (PROGRAD). A variável resposta de interesse é o RSMG. As covariáveis consideradas no estudo são: a pontuação obtida no vestibular, que varia de 0 a 100 (SAdmEx); a classificação social medida pela escala ABIPEME de 5 estratos, que varia de A, mais alto, até E, mais baixo (SocEcoS); o número de créditos já cursados (TcredC); o gênero (Sex) e o tipo de escola frequentada pelo estudante durante o ensino médio, que pode ser estadual, federal, municipal ou particular (THighSch). Uma variável que pode influenciar no desempenho e sobre a qual se tem um interesse especial é o curso de graduação em que o aluno está matriculado. São 71 cursos na UFMG e com tantas categorias, o modelo proposto pode contribuir na identificação de cursos com efeito semelhante no rendimento do aluno, agrupando-os, e estimando seus efeitos no desempenho dos alunos ao mesmo tempo, pode identificar influências das demais covariáveis neste rendimento.

4.3.1 Distribuições a *priori* e valores iniciais

Os hiperparâmetros para o modelo de mistura NIDPP foram definidos para obter distribuições *a priori* vagas sempre que possível. Para os parâmetros comuns a todos os indivíduos foram definidos $\mu_\beta = \mathbf{0}$, $\Sigma_\beta = \upsilon_\beta \mathbf{I}$ com $\upsilon_\beta = 10^4$, $a_0 = b_0 = 0{,}01$ com indicado por [49]. Para os pesos da mistura, $\delta_k = 1, k = 1, \ldots, K$, e para os parâmetros do kernel $a_1 = 100, b_1 = 1$, para θ^2, $a_2 = 200$ e $b_2 = 0{,}5$. As cadeias do MCMC foram inicializadas com os seguintes valores: $\sigma_y^{2(0)} = var(Y) = 1,12$, $\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma_y^{2(0)}\mathbf{I})$, $\boldsymbol{\alpha}^0 \sim N(\mathbf{0}, \mathbf{I})$ e $u_{ij} = 1, j = 1, \ldots, J, i = 1, \ldots, n_j$. Para os parâmetros do kernel os valores iniciais foram obtidos por método de momentos para processos pontuais regulares [32] adaptados ao PPD. Para θ^2 foi utilizada a mediana das distâncias Euclidianas dos dados da variável resposta, 0,9. Enquanto o valor inicial para σ_q^2 seria o número de pontos por unidade de volume ocupada pelos dados, 905,8, [15].

Para avaliar o efeito do peso da cauda da distribuição do RSGM na qualidade do ajuste

do modelo, o hiperparâmetro da variável misturadora, u, foi fixado em três valores diferentes, $\eta = 2{,}1$; 5,0 e 100. O primeiro valor ajusta uma distribução com valor de η no limite da existência da variância da distribuição t de Student, $\eta > 2$, uma distribuição com caudas mais pesadas possíveis, mais ainda tem os dois primeiros momentos finitos. O segundo valor, $\eta = 5$, continua tendo caudas pesadas um pouco mais moderadas. Já o terceiro valor, $\eta = 100$, apresenta a distribuição aproximadamente normal o que é uma boa comparação com outros modelos que consideram apenas a distribuição Gaussiana para a variável resposta. No MMF há a necessidade de definir um K_{max}, número máximo de grupos, antes do ajuste do modelo. Neste trabalho foram utilizados os valores 16 e 29. O primeiro valor foi escolhido como o dobro do número de grandes áreas de cursos na UFMG, enquanto o valor 29 foi o número esperado indicado no trabalho de [26] para efeito de comparação. Também foram realizados outros ajustes variando valores do parâmetro dos pesos da mistura com $\delta = 0{,}01$.

Para os ajustes dos modelos NIDPP foram obtidas 20000 iterações de MCMC. Foram descartadas 2500 primeiras iterações para o modelo com $\eta = 2{,}1$ e 5000 para os modelos com $\eta = 5$ e $\eta = 100$ e saltos de tamanho 10 para evitar fortes autocorrelações. Dessa forma, as inferências *a posteriori* foram analisadas a partir de amostras de tamanho 1750 para o modelo com $\eta = 2{,}1$ e 1500 para os outros modelos.

Para sumarizar a informação sobre as alocações dos grupos fornecidas por $\boldsymbol{z}$ estimada pelo modelo NIPPD serão utilizadas algumas funções de perda recomendadas na literatura. Para escolher o melhor particionamento dos efeitos dos cursos utiliza-se a Perda Variação de Informação (VI) [78], a Perda de Binder N-invariante (B) [14] e a Perda omARI (One minus Adjusted Rand Index) aqui denotada por ARI [114]. As perdas de Binder e ARI são relacionadas. A perda ARI inclui uma correção para compensar o acaso em alocações completamente aleatórias, enquanto a perda de Binder permite realizar penalizações em diferentes erros de alocação. A perda VI é baseada em teoria da informação e favorece alocações com menos entropia, definindo *clusters* estimados mais homogêneos (detalhes ver [114]). As perdas são calculadas a partir da matriz de similaridade que estima as probabilidades de dois indivíduos estarem no mesmo *cluster* [41]. Para comparar o desempenho preditivo dos modelos, foi utilizada a técnica de validação cruzada *5-fold* e calcularam-se as medidas *Root Mean Squared Error* (RMSE), *Mean Absolut Error* (MAE) e o *Deviance Information Criterion* (DIC) a partir da verossimilhança condicional dada em (3.28) [21].

Os modelos LASSO Bayesiano (LB) [83] e Product Partition Regression Model (PPRM) [26] para agrupamento de níveis de variáveis categóricas foram ajustados e os resultados foram comparados com aqueles obtidos pelo modelo proposto, NIPPD. Os ajustes do LB e PPRM foram obtidos considerando-se os mesmo valores de hiperparâmetros e as mesmas distribuições a *priori* de $\boldsymbol{\beta}$ e σ_y^2. O PPRM foi ajustado assumindo-se que $\rho \sim Beta(4, 6)$, *a priori*. Para o ajuste do LASSO Bayesiano assumiu-se $\lambda = 1$. Para ambos os modelos, foram geradas 10000 iterações com aquecimento de 1000 e saltos de 5 resultando em amostras *a posteriori* de tamanho 1800.

4.3.2 Estimação dos parâmetros do modelo

Nesta seção, avalia-se o comportamento do modelo proposto considerando-se diferentes especificações *a priori* para alguns dos parâmetros. Foram ajustados 12 modelos diferentes com as variações do número máximo de *clusters* $K_{max} = 16$ e $K_{max} = 29$, do hiperparâmetro da distribuição do peso da mistura, $\delta = 1{,}0$ e $\delta = 0{,}01$, e do hiperparâmetro η envolvido na distribuição da variável misturadora $\eta = 2{,}1$, $\eta = 5{,}0$ e $\eta = 100$.

Apesar de alguns dos modelos propostos permitirem até 29 grupos, por assumir $K_{max} = 29$, em nenhum deles o valor estimado de K foi maior que 15. A Tabela 4.1 apresenta a distribuição *a posteriori* do número de grupos K fixando $K_{max} = 16$ e $K_{max} = 29$ e considerando os hiperparâmetros $\delta = 1{,}0$ e $\delta = 0{,}01$ e $\eta = 2{,}1$; 5,0 e 100. Considerando a moda *a posteriori*, a estimativa *a posteriori* para K tende a aumentar com o aumento do grau de liberdade. Isso ocorre tanto para ajustes considerando $\delta = 1{,}0$ quanto $\delta = 0{,}01$. A única exceção foi observada quando se ajustou o modelo proposto com $K_{max} = 29$ e $\delta = 1{,}0$. Este resultado é esperado, de certa forma, uma vez que o modelo com caudas mais pesadas acomoda melhor observações atípicas, não as alocando para um *cluster* diferente. Além disso, o valor das estimativas variam entre 4 e 9 grupos, esses valores estão condizentes com o que se espera para os cursos de graduação que, dentro da universidade, estão organizados em oito grandes áreas acadêmicas: Ciências Naturais, Engenharias, Ciências da Saúde, Ciências Agrárias, Letras e Artes, Ciências Sociais Aplicadas e Ciências Humanas.

Tabela 4.1: Distribuição *a posteriori* do número de *clusters*, K, para os modelos propostos.

	$K_{max} = 16$ e $\delta = 1{,}0$										HPD 95%	
η	3	4	5	6	7	8	9	10	11	Moda	Inf	Sup
2,1	0,01	0,05	0,33	**0,39**	0,18	0,04	0,00			6	4	7
5,0		0,00	0,06	0,29	**0,41**	0,22	0,02			7	5	8
100				0,01	0,19	**0,55**	0,21	0,04	0,00	8	7	10
	$K_{max} = 16$ e $\delta = 0,01$										HPD 95%	
2,1	0,17	**0,56**	0,24	0,03						4	3	5
5,0	0,00	0,26	**0,57**	0,16	0,01					5	4	6
100		0,00	0,13	0,33	**0,42**	0,11	0,01			7	5	8
	$K_{max} = 29$ e δ =1,0											
2,1	0,01	0,23	0,08	0,19	**0,28**	0,14	0,05	0,01	0,00	7	2	10
5,0	0,18	**0,29**	0,02	0,12	0,18	0,13	0,06	0,01	0,00	4	3	9
100	0,12	0,14	0,10	0,13	0,04	0,14	**0,20**	0,09	0,04	9	2	10
	$K_{max} = 29$ e δ =0,01											
2,1	0,07	**0,50**	0,37	0,06	0,00					4	3	6
5,0		0,11	**0,48**	0,34	0,07	0,00				5	4	7
100			0,04	0,25	**0,42**	0,27	0,03	0,00		7	5	8

As áreas acadêmicas podem trazer informação sobre semelhança entre os cursos quanto a atuação,

mas não necessariamente sobre o desempenho dos alunos. Considerando os modelos com $K_{max} = 16$, $\eta = 2{,}1$ e estimando os parâmetros com mediana *a posteriori*, observa-se que os efeitos dos cursos no RSGM do aluno são todos negativos como mostrado na Figura 4.1, indicando que todos os cursos causam o efeito de diminuir o RSGM dos alunos. Em cada curso a nota de partida, ou seja, com o valor das outras covariáveis sendo nulos (ou assumindo as categorias de referência se a covariável também é qualitativa), o aluno já de uma defasagem no seu desempenho (rendimento negativo). O que faz o aluno ter uma nota positiva são as outras covariáveis ou o efeito aleatório.

O curso de Biblioteconomia (*Library Science*) apresenta o menor efeito no rendimento do aluno, seguido de outros cursos da área de Letras e Artes, mas também dos cursos de Pedagogia (*Pedagogy*), História (*History*) e Ciências Sociais (*Social Science*) os quais são da área de Ciências Humanas. O curso de Medicina apresenta-se isolado, tendo maior influência negativa no rendimento do aluno com efeito estimado, sendo o maior em valor absoluto, precedido pelo curso de Medicina Veterinária e antes deste o curso de Geologia, ambos de áreas diversas. A maioria dos cursos da área de Engenharia possuem valores de efeito semelhante, com excessões dos cursos de Engenharia Ambiental e Engenharia de Controle e Automação cujos efeitos estão mais próximos aos efeitos de cursos da área de Ciências da Natureza e a maioria dos cursos de Ciências Saúde.

Os intervalos HPD com 95% de probabilidade para o efeito dos cursos apresentam amplitude próxima de 3,0 para a maioria, indicando uma variabilidade similar das distribuições *a posteriori* destes efeitos. Os cursos que mostraram maior incerteza *a posteriori* sobre os seus efeitos no RSGM, são Farmácia noturno (*Pharmacy (night)*), de Medicina (*Medicine*), de Antropologia noturno (*Anthropology (night)*) e Dança noturno (*Dance (night)*) para os quais o HPD tem a amplitude de cerca de 4,0.

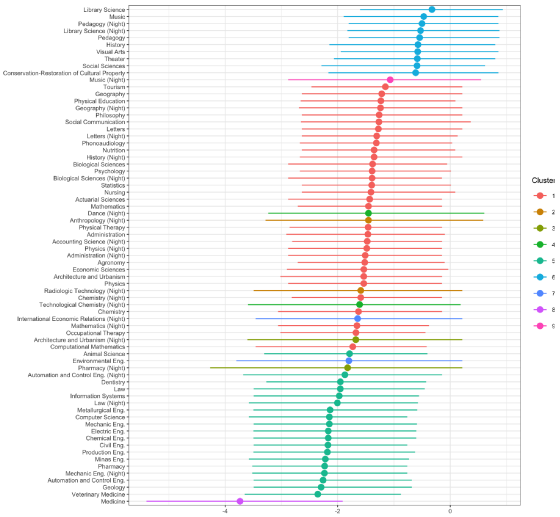
As Figuras 4.1a, 4.1b e 4.1c também mostram os agrupamentos dos efeitos dos cursos obtidos considerando as perdas ARI (um menos o valor do Índice de Rand Ajustado), a perda de Binder e a Perda Variação de Informação. Na alocação estimada segundo a Perda VI (Figura 4.1a) para o modelo NIDPP ajustado com $\eta = 2{,}1$, a maioria dos cursos da área de Engenharia foram alocados no mesmo grupo, o que era de se esperar por possuírem valores de efeito semelhantes. As exceções são Engenharia Ambiental e Engenharia de Controle e Automação que foram alocados junto a cursos da área de Ciências da Natureza e a maioria dos cursos de Ciências da Saúde. Esse último, foi o maior entre os 4 *clusters* definidos pela perda VI para esse modelo e também é composto pelos cursos das demais áreas (Ciências Biológicas e Ciências Agrárias). O curso de Medicina compõe um grupo individual e se destaca por ter o maior efeito negativo, estar bem separado dos demais e possui um intervalo HPD com maior amplitude do que os efeitos com medianas próximas.

Segundo as outras funções de perda, ARI (Figura 4.1b) e de Binder (Figura 4.1c), são definidos 9 grupos. O grupo cujos efeitos são estimados em valores maiores, que compreende até o curso de Conservação e Restauração de Patrimônio Cultural, foi mantido. O curso de Medicina, cuja estimativa do efeito foi a mais negativa entre todos os cursos, foi alocado com o curso de Farmácia (*Pharmacy*) pela perda de Binder, mas não pela perda ARI. Em geral, para as duas perdas, os cursos alocados em *clusters* diferentes em relação à perda VI são, principalmente, os que possuem intervalos HPD de 95% com amplitudes maiores, ou seja, que possuem maior variabilidade *a posteriori*. Por exemplo, a perda de Binder agregou os Cursos de Dança noturno (*Dance (night)*) e Música noturno (*Music (night)*) em um *cluster* mesmo não possuindo valores de efeitos vizinhos, ou seja, contíguos no gráfico pela ordenação.

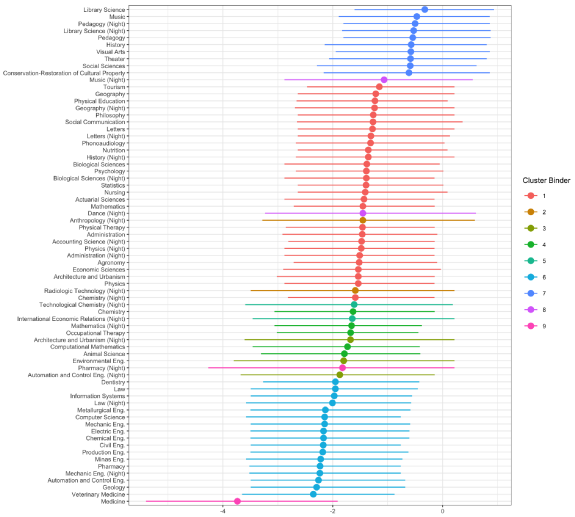
Figura 4.1: Medianas *a posteriori* e intervalo HPD 95% de probabilidade para os efeitos dos cursos ajustando o modelo NIPPD com $\delta = 1{,}0$ e $\eta = 2{,}1$, alocados com as perdas VI, ARI e de Binder. Os clusteres são indicados por diferentes cores.

(a) *clusters* pela perda VI.

(b) *clusters* pela ARI.



(c) *clusters* pela perda de Binder.



Fonte: Elaborado pela autora.

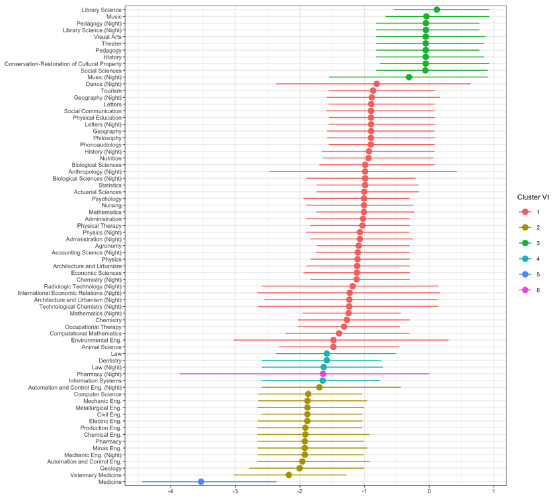
Assim como a perda ARI agregou Antropologia noturna (*Antropology (night)*) e Tecnologia em Radiologia noturna (*Radiologic Tecnology (night)*), os dois grupos não possuem efeitos com valores vizinhos, mas possuem intervalos HPD mais amplos.

O modelo ajustado assumindo $\eta = 2{,}1$ graus de liberdade mostrou-se mais parcimonioso com estimação *a posteriori* de um menor número de *clusters* para efeitos dos cursos e gerando efeitos mais homogêneos. O aumento de η no modelo NIPPD, em geral, estimou valores maiores para os efeitos dos cursos sobre o RSGM. Por exemplo, o grupo com os maiores efeitos, possui medianas *a posteriori* próximas de 0 (zero) no ajuste com $\eta = 5$ e acima de 0,5 no ajuste com $\eta = 100$, enquanto no ajuste com $\eta = 2{,}1$ os valores foram todos abaixo de -0,1. Na Figura 4.2 são apresentados os efeitos dos cursos sob o ajuste do modelo NIPPD com $\eta = 5{,}0$ e na Figura 4.3 o ajuste com $\eta = 100$. Neste caso, o número

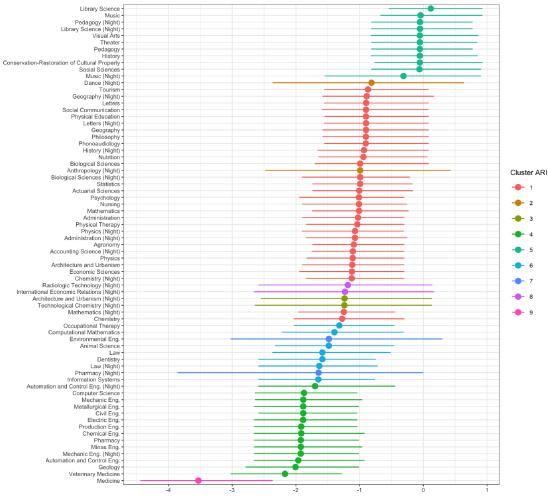
de *clusters* aumenta mesmo sob a perda VI, que aponta $K = 6$ para $\eta = 5$ e $K = 11$ para o modelo com $\eta = 100$. Diferente do ajuste com $\eta = 2{,}1$, foi definido um *cluster* apenas com o curso de Farmácia noturno para o modelo com $\eta = 5{,}0$. Este curso possui o maior intervalo HPD de 95% neste ajuste, e outro grupo de quatro cursos diversos foi definido. Além disso, nos ajustes com maiores graus de liberdade, o comprimento dos intervalos HPD de 95% dos efeitos dos cursos sobre o RSGM diminuiu, mas a diminuição não afetou todos os efeitos da mesma forma.

Figura 4.2: Mediana *a posteriori* e Intervalos HPD de 95% de probabilidade para os efeitos dos cursos estimados pelo modelo NIPPD com δ =1,0 e η = 5,0, alocados pela Perda VI, ARI e de Binder.

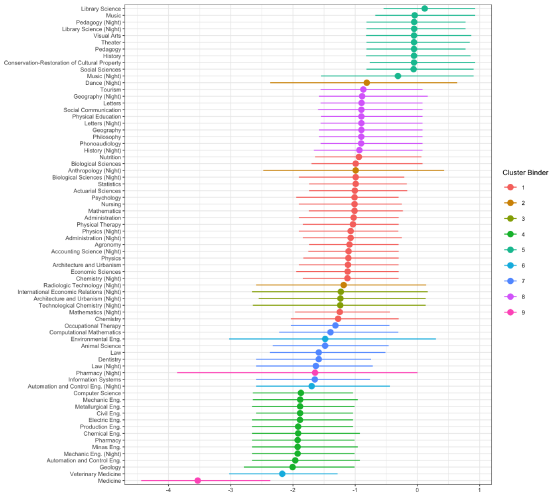
(a) *clusters* pela perda VI.



(b) *clusters* pela perda ARI.



(c) *clusters* pela perda Binder.

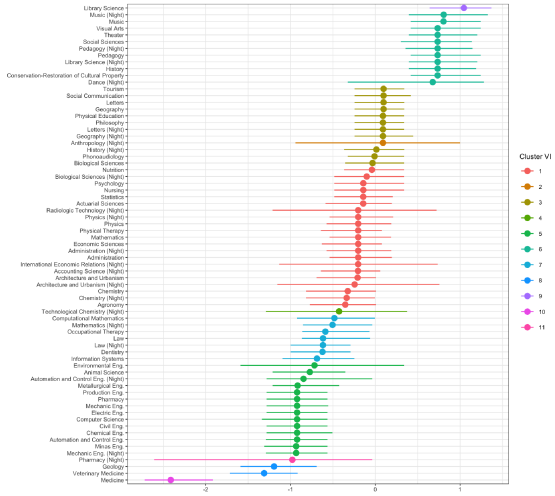


Fonte: Elaborado pela autora.

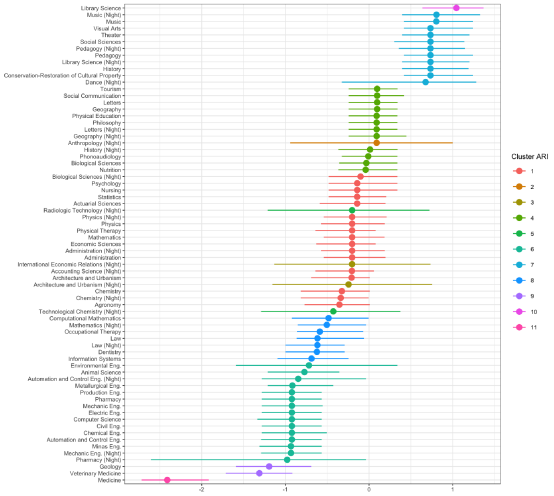
Alguns cursos apresentam maior variabilidade na distribuição *a posteriori* e seus intervalos são mais discrepantes dos outros que no ajuste com $\eta = 2{,}1$. Enquanto a maioria possui intervalos com comprimento em torno de 1,5 no ajuste com $\eta = 5$, o curso de Medicina, por exemplo, tem intervalo de comprimento maior que 2,0 e o curso de Farmácia possui intervalo maior que 4,0. Ambos os cursos foram separados em *clusters* individuais segundo a perda VI, destacando suas diferenças não apenas pelo

Figura 4.3: Mediana *a posteriori* e Intervalos HPD de 95% de probabilidade para os efeitos dos cursos estimados pelo modelo NIPPD com δ =1,0 e η = 100, alocados pela Perda VI, ARI e de Binder.

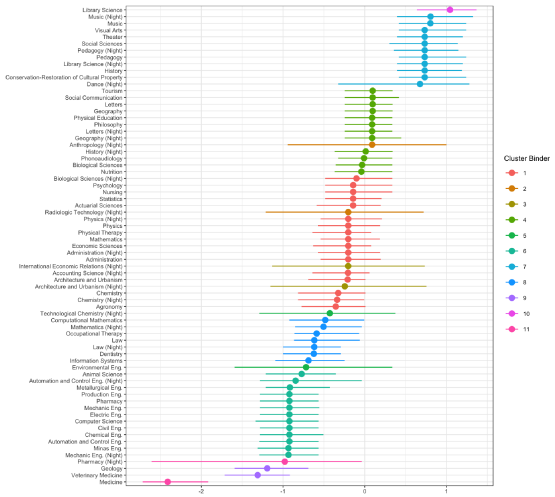
(a) *clusters* pela perda VI.



(b) *clusters* pela perda ARI.



(c) *clusters* pela perda de Binder.



Fonte: Elaborado pela autora.

tamanho do efeito, mas também pela variabilidade. Isto os torna mais difíceis de agrupar de maneira única.

4.3.3 Estimação dos parâmetros do kernel, parâmetro de escala e avaliação dos modelos

Uma novidade do modelo proposto é a estimação dos parâmetros do kernel do PPD. Muitas dificuldades são descritas na literatura sobre esta estimação [1, 20, 66], pois suas funções dependerem

diretamente de determinantes de matrizes que podem ser onerosos computacional e analiticamente. Além disso, a forma dos autovalores que aparecem no denominador da função densidade do PPD em (4.13) nem sempre é conhecida. Alguns trabalhos contornam estimação dos parâmetros usando uma função densidade aproximada para o PPD [66, 13]. Outros trabalhos consideram tais quantidades como fixas ou hiperparâmetros [118, 9]. A partir da restrição proposta em 3.2.1 foram estimados os parâmetros usando o algoritmo da Subseção 4.2.3 com bons resultados para o modelo e convergência (Apêndice B).

A Tabela 4.2 apresenta as medianas e os intervalos HPD de 95% *a posteriori* para os parâmetros θ^2 e σ_q^2 do kernel do PPD assumindo-se $K_{max} = 16$ e 29, $\eta = 2{,}1$; 5,0 e 100 e δ =0,01 e 1,0. Os valores estimados não são afetados pela variação dos outros parâmetros. Isto pode ocorrer porque os parâmetros do kernel são voltados para o relacionamento entre os efeitos dos cursos e não para as medidas da distribuição da variável resposta.

A Tabela 4.2 também apresenta alguns resumos da distribuição *a posteriori* para o parâmetro de forma σ_y^2. Pelos valores, é possível identificar um comportamento de aumento da mediana *a posteriori* de σ_y^2, conforme a diminuição do grau de liberdade η para todos os modelos ajustados. Este resultado é esperado uma vez que σ_y^2 é o parâmetro de forma da família NI o qual traz informação sobre a variabilidade condicional do RSGM. Para o modelo com $\eta = 100$, os valores são pequenos e próximos do estimado assumindo normalidade condicional para o RSGM, como adotado no trabalho de [26].

Os resultados do DIC (*Deviance Information Criterion*) também foram calculados para todos os modelos ajustados sendo apresentados na Tabela 4.2. O DIC aponta o modelo que assume um valor alto para o parâmetro de grau de liberdade, $\eta = 100$. Ou seja, o modelo com melhor ajuste segundo o DIC é aquele que mais se aproxima a um modelo de mistura finita da distribuição normal, sendo que o melhor resultado foi o que considera $K_{max} = 29$ e δ =0,01. Esse modelo será comparado ao PPRM e ao LASSO Bayesiano para o agrupamento de níveis de variáveis categóricas na próxima subseção.

4.3.4 Comparação com outros modelos

No trabalho de [26] os dados foram ajustados considerando distribuição normal, além disso, o modelo NIPPD melhor avaliado pelo DIC foi aquele com $\eta = 100$, aproximadamente normal. Dessa forma, trataremos principalmente, mas não exclusivamente, dos resultados referentes ao modelo NIPPD com $K_{max} = 29$, $\eta = 100$ e δ =0,01 em comparação com os modelos LASSO Bayesiano [83] e *Production Partition Regression Model* [26].

A Tabela 4.3 mostra que o modelo NIPPD possui características preditivas não muito distantes daquelas obtidas ajustando-se o modelo PPRM e é superior ao LB. Entre os três modelos o PPRM possui os melhores resultados. Os modelos NIPPD e PPRM têm o menor valor para o MAE. O modelo LASSO também tem bons resultados quanto ao RMSE e MAE, no entanto, sua proposta não é realmente de agrupamento e não dispõe de um mecanismo proprio para agregar os cursos. O procedimento consiste em estimar os parâmetros do modelo, depois utilizar outro método auxiliar como o k-médias para agrupar os cursos. Diferente dos outros dois modelos, o NIPPD não faz imposição aos dados e nem demanda

Tabela 4.2: Medianas e intervalos HPD de 95% para os parâmetros θ^2, σ_q^2 e o parâmetro de forma σ_y^2 ajustando o modelo proposto com K_{max} = 16 e 29, η = 2,1; 5,0 e 100 e δ = 0,01 e 1,0.

		K_{max} = 16			K_{max} = 29		
		Mediana	HPD 95%		Mediana	HPD 95%	
			Inf	Sup		Inf	Sup
η	δ = 1,0						
	θ^2	103,475	82.955	123,513	103,932	91,146	127,903
2,1	σ_q^2	94,931	81,434	108,007	94,134	81,784	109,058
	σ_y^2	3,61	1,72	12,30	3,55	1,73	10,11
	DIC		16071,71			16714,96	
	δ = 1,0						
	θ^2	103,439	81,971	121,931	101,768	10,691	127,753
5,0	σ_q^2	93,627	80,634	107,700	94,279	57,024	14532.964
	σ_y^2	1,27	1,05	1,61	1,28	1,04	1,68
	DIC		13031,23			13300,20	
	δ = 1,0						
	θ^2	102.837	83,442	123,534	91,202	3,135	108,542
100	σ_q^2	92.391	80,188	108,050	93,046	69,514	13542.855
	σ_y^2	0,50	0,47	0,52	0,50	0,47	0,60
	DIC		9365,123			9458,95	
	δ = 0,01						
	θ^2	102.673	84,459	122.253	102.538	83,627	121,963
2,1	σ_q^2	97,359	82.611	110,165	96,191	83,432	109,551
	σ_y^2	4,08	1,94	12.43	3,878	2.04	13,62
	DIC		17523,19			13682.13	
	δ = 0,01						
	θ^2	104,527	81,946	147,180	101,061	41,228	124,814
5,0	σ_q^2	96,693	74,849	14369,622	96,832	78,382	12925,098
	σ_y^2	1,34	1,09	1,78	1,321	1,08	1,73
	DIC		13428,92			12992,64	
	δ = 0,01						
	θ^2	101,896	79,567	125,260	103,280	68,271	154,361
100	σ_q^2	95,057	73,486	13425,164	94,540	71,810	13540,410
	σ_y^2	0,51	0,48	0,55	0,503	0,48	0,53
	DIC		9638,90			9243,67	

Fonte: Elaborado pela autora.

um processo externo de agrupamento, possui método probabilístico que permite a avaliação de incerteza sobre o agrupamento e sobre todos os efeitos estimados.

Tabela 4.3: Critérios de comparação dos modelos.

Modelo	RMSE	MAE	DIC
NIPPD	0,67	0,51	9243,67
PPRM	0,45	0,51	9238,52
LASSO	0,50	0,54	21031.55

Fonte: Elaborado pela autora.

A Figura 4.4 mostra as medianas *a posteriori* para os efeitos dos cursos e os intervalos HPD de 95% estimados pelos três modelos, bem como a alocação dos *clusters* via perda VI. As estimativas para o modelo NIPPD são praticamente todas negativas, com medianas entre -3,5 a 0,5 (Figura 4.4a). Para o modelo PPRM, os efeitos são positivos apenas para os dois grupos com efeitos de valores maiores entre 0,5 e 1,0 (Figura 4.4b). Ajustando-se o modelo LASSO os efeitos são estimadas entre -1,25 e 0,75 (Figura 4.4c). No entanto, quase que a totalidade dos efeitos são estimados em valores bem próximo de zero, indicando não haver efeito dos cursos sobre o rendimento do aluno.

Quanto ao número de *clusters* os modelos apresentam diferentes estimativas. Segundo a perda VI, o PPRM apresentou 10 grupos, enquanto o NIPPD apresentou 6 grupos. O modelo LASSO não estima o número de *clusters*, foi pré-fixado o valor $K = 4$ por ser próximo ao que a visualização dos efeitos permite identificar, Figura 4.4c. No método LASSO, os coeficientes não relevantes são "encolhidos" até zero, dessa forma, o agrupamento não é muito específico. Apenas os cursos com efeitos muito diferentes sobre o RSGM são separados. Mesmo o curso de Comunicação Social que teve um efeito diferente de zero foi alocado no maior *cluster* com os demais cursos considerados com efeito zero. No modelo PPRM, Figura 4.4b, 4 cursos foram alocados em *clusters* individuais, e outro foi formado com apenas dois cursos. Enquanto o modelo NIPPD estimou apenas um *cluster* individual e um com dois cursos.

Nos três modelos, o curso de Medicina compõe um *cluster* individual com o efeito mais negativo sobre RSGM, enquanto o curso de Biblioteconomia possui o efeito mais positivo sobre o desempenho dos alunos estudados no modelo NIPPD e no PPRM. Isto já havia sido verificado nos ajustes do modelo NIPPD das subseções anteriores. Para o modelo LASSO o curso de Biblioteconomia não foi o maior efeito e sim o quarto maior. Os 12 cursos com maiores efeitos no modelo NIPPD também são os maiores para o modelo PPRM, mas no primeiro formam um único *cluster*, enquanto no segundo foram separados em dois *clusters*. O modelo LASSO tem quase todos os 6 cursos do *cluster* com maiores efeitos entre estes 12, a exceção é o curso de Comunicação Social. Os três modelos possuem um *cluster* de dois cursos logo acima do curso de Medicina, com os cursos de Medicina Veterinária e Geologia para os modelos NIPPD e PPRM e com os cursos de Medicina Veterinária e Engenharia Civil para o modelo LASSO. Todos os outros cursos foram considerados do mesmo *cluster* pelo modelo LASSO.

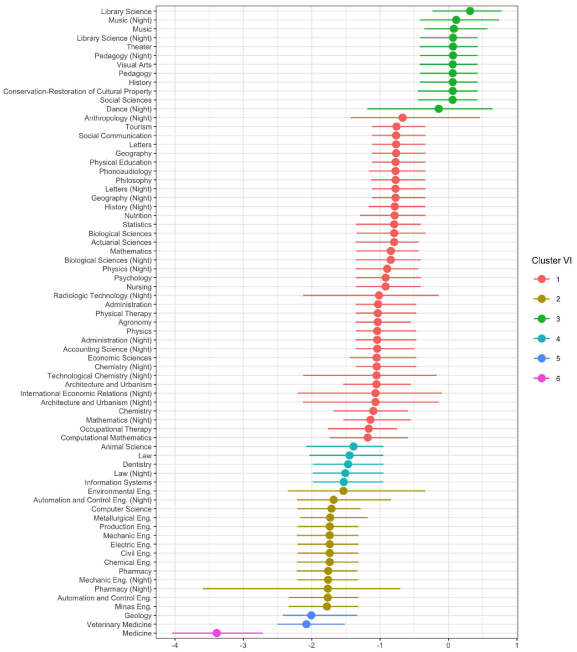
Os modelos NIPPD e PPRM possuem um grupo com efeitos negativos sobre o RSGM formado em sua maioria por cursos da área de engenharia, além dos cursos de Ciências da Computação, Farmácia e Farmácia noturno. Um único curso que difere entre o *cluster* de um modelo e do outro é o de Zoologia para o modelo NIPPD e o curso de Engenharia Ambiental para o modelo PPRM. O modelo NIPPD agrupa os demais cursos em um único *cluster*, mas o modelo PPRM, além de separar o curso de Engenharia Ambiental em um *cluster* individual, cria o outro com efeitos próximos de zero com cursos de várias áreas: Educação Física, Antropologia noturna, História, Filosofia, Letras, Letras noturno, Ciências Biológicas, Nutrição, Fonoaudiologia, Geografia, Geografia noturno, Turismo e Comunicação Social.

Em geral, o modelo PPRM possui intervalos HPD de 95% com amplitude menor que o modelo NIPPD e que o modelo LASSO. Este último apresenta muitos intervalos com amplitude elevada para os efeitos de curso, alguns com amplitude maior que 7: Antropologia noturno, Arquitetura e Urbanismo noturno, Dança noturno, Relações Econômicas Internacionais noturno, Farmácia noturno, Tecnologia em Radiologia noturno e Química Tecnológica noturno. Amplitudes elevadas indicam maior variabilidade na distribuição *a posteriori* dos efeitos, o que pode dificultar a alocação desses cursos. Intervalos

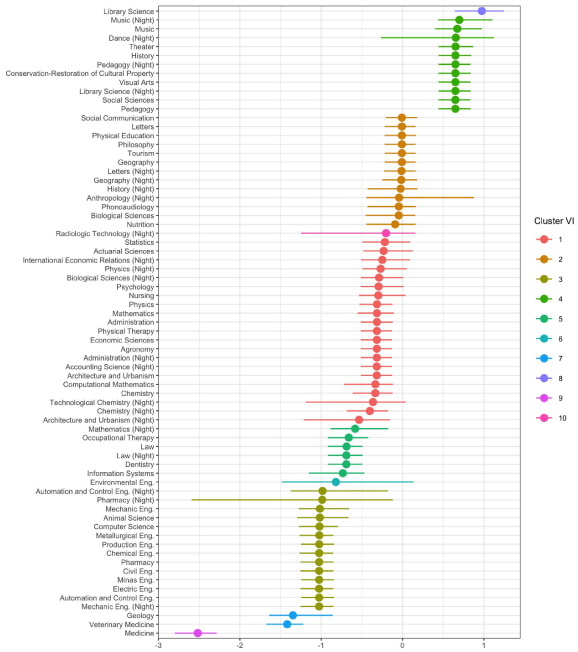
mais amplos ocorreram também nos outros modelos. No entanto, os intervalos maiores são proporcionalmente mais discrepantes para o modelo NIPPD, com tamanhos até 5 vezes maiores que comprimento da maioria. Por exemplo, o intervalo HPD do curso de Psicologia é de 0,5 e o de Engenharia Ambiental é 3 vezes maior, enquanto o de Farmácia é 5 vezes maior. Os cursos de Tecnologia em Radiologia e Engenharia ambiental foram alocados em *clusters* individuais neste modelo e seus efeitos no rendimento

Figura 4.4: Mediana *a posteriori* e Intervalos HPD de 95% de probabilidade para os efeitos dos cursos estimados pelo modelo NIPPD, PPRM e pelo modelo LASSO Bayesiano.

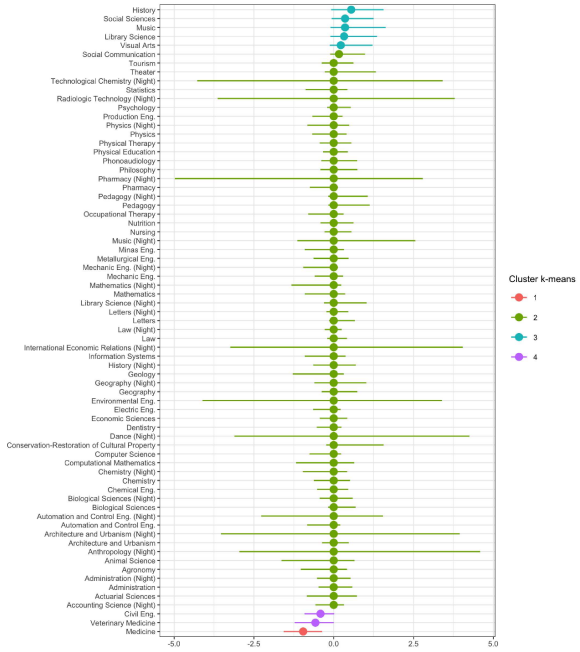
(a) Modelo NIPPD, $\eta = 100$ e $\delta = 0,01$.



(b) *clusters* pela perda VI para o modelo PPRM.



(c) *Clusters* via k-médias para o modelo LASSO.



Fonte: Elaborado pela autora.

do aluno possuem intervalos de maiores amplitudes.

A Figura 4.5 apresenta os gráficos de calor das matrizes de similaridade com base nas alocações probabilísticas definidas pelos modelos NIPPD e PPRM. O LASSO não possui um mecanismo aleatório de alocações. A matriz de similaridade é obtida a partir dos respectivos MCMC's dos modelos e os seus elementos representam a estimativa da probabilidade *a posteriori* de dois cursos estarem no mesmo *cluster*. A intensidade da cor é proporcional a probabilidade, quanto mais azul a cor do quadriculado no gráfico, maior essa probabilidade e quanto mais branco menor ela é. Os gráficos dos dois modelos, NIPPD, Figura 4.5a e PPRM, Figura 4.5b, são parecidos. Não há um padrão bloco-diagonal claro no geral, o que indicaria haver um agrupamento bem definido, mas existem alguns agrupamentos para alguns dos cursos de Ciências Sociais e Artes e outro composto pelos Cursos de Engenharia. As Ciências Humanas mostram agrupamentos entre si, mas também com cursos das Ciências Sociais. Já curso de Medicina aparece isolado de todos, apenas com uma pequena probabilidade com o Curso de Farmácia Noturno. No modelo, NIPPD essa probabilidade é 0,21 e no modelo PPRM é 0,07. Esses dois cursos não foram agrupados juntos, mas o curso de Farmácia noturno é um dos melhores exemplos em que a variabilidade na distribuição *a posteriori* do seu efeito se traduz em probabilidade de alocação dispersa.

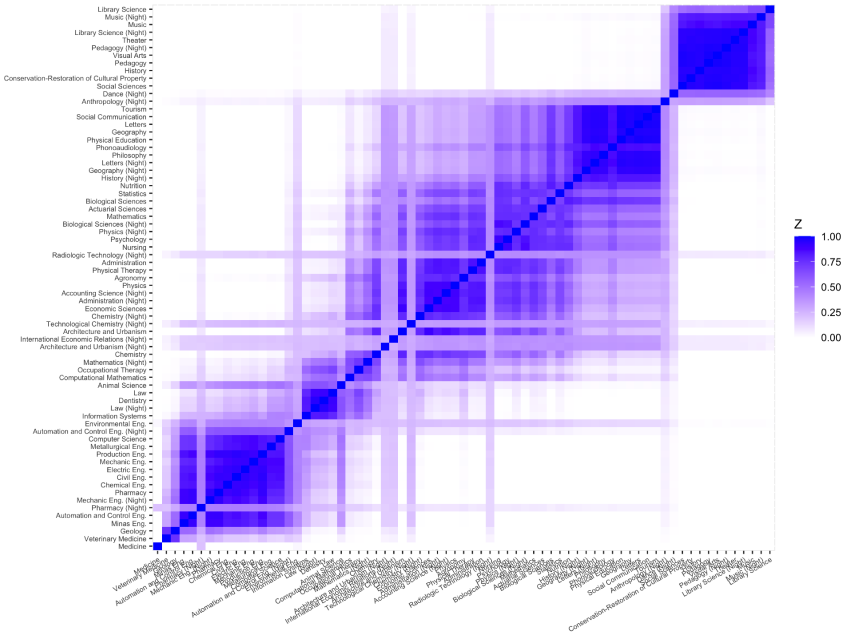
A Figura 4.5a mostra que no modelo NIPPD existe uma pequena probabilidade de alocação do curso de Farmácia noturno com quase todos os cursos exceto os do *cluster* com efeitos maiores. Sua maior probabilidade de alocação é com o curso de Engenharia de Produção (0,46). Para o modelo PPRM o comportamento é parecido, Figura 4.5b, só não tem probabilidade positiva de alocação com os dois *clusters* de efeitos maiores, a maior probabilidade é com o curso de Ciência da Computação (0,61). O curso de Tecnologia Radiológica noturno possui este tipo de comportamento no modelo PPRM, probabilidade de alocação baixa, mas positiva com 68 cursos. A maior destas probabilidades é com o curso de Geografia noturno (0,47) mas não foi suficiente para agrupá-lo a nenhum outro curso. No modelo NIPPD, o curso de Tecnologia Radiológica noturno possui probabilidade positiva com todos os outros cursos e acima de 0,35 com 28 deles. A maior probabilidade é com o curso de Matemática com 0,41. E para esse modelo foi suficiente para agrupá-lo com outros cursos de Ciências da Natureza e os demais desse grupo. O curso de Engenharia Ambiental, que também ficou isolado no modelo PPRM, possui maior probabilidade (0,43) de alocação com o curso de Engenharia de Controle e Automação (Night), também possui maior probabilidade de estar no mesmo grupo com o curso de Engenharia Civil (0,44) no modelo NIPPD e foi alocado com a maioria dos outros cursos da área de Engenharia. Isso tudo indica que o modelo NIPPD aloca os cursos com probabilidade moderada enquanto o PPRM apresenta probabilidades mais altas.

Na Tabela 4.4, são apresentados os coeficientes dos modelos para as demais covariáveis que são comuns a todos os indivíduos. O modelo LASSO Bayesiano apresenta quase todos os coeficientes com valor "zero", exceto Total de Créditos Cursados, o sexo, sendo o sexo masculino um efeito negativo para o desempenho dos alunos. Estas covariáveis foram consideradas significativas para o RSGM em quase todos os modelos, a exceção é o modelo NIPPD com $\eta = 2{,}1$ que não concorda com o efeito do sexo, mas também é negativo. A classe socio econômica (SocEco) foi considerada com influência significativa sobre RSGM apenas pelo modelo NIPPD $\eta = 100$ e δ =0,01. O efeito do tipo de escola de ensino médio (THighSCH) foi considerado significativo apenas para o modelo NIPPD com $\eta = 100$ e $\delta = 1{,}0$.

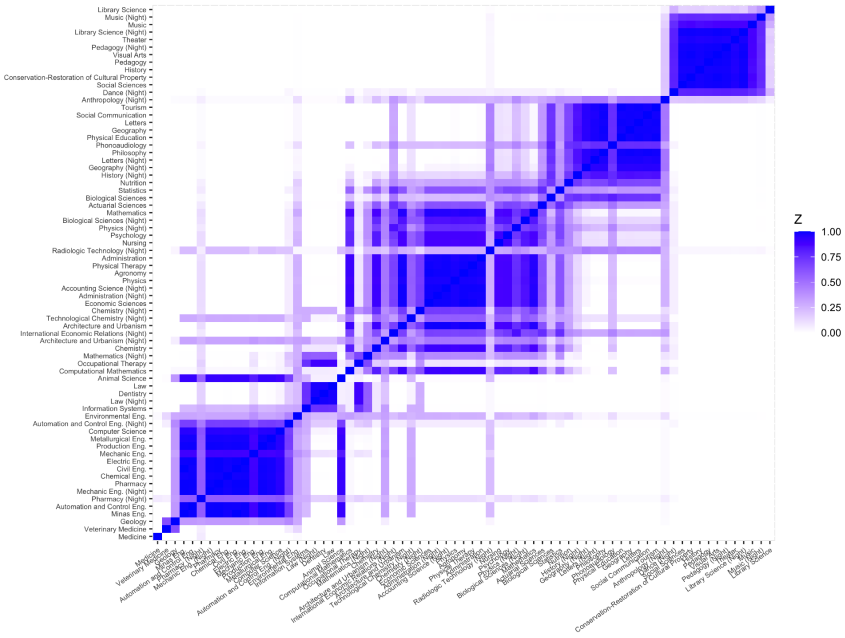
O parâmetro σ_y^2 foi estimado com valor maior para o modelo LASSO (11,43) devido aos coefi-

Figura 4.5: Matriz de similaridade modelos NIPPD e PPRM.

(a) Modelo NIPPD com $\eta = 100$ e $\delta = 0,01$.



(b) Modelo PPRM.



Fonte: Elaborado pela autora.

Tabela 4.4: Mediana *a posteriori* e intervalo HPD 95% para as covariáveis comuns aos indivíduos nos modelos NIPPD, LASSO e PPRM.

	NIPPD - η =2,1 δ =1,0			NIPPD - η = 5 δ =1,0			NIPPD - η = 100 δ =1,0		
	Med	Inf	Sup	Med	Inf	Sup	Med	Inf	Sup
SocEco									
B	0,52	-0,46	1,70	0,25	-0,22	0,72	-0,01	-0,20	0,22
C	0,55	-0,42	1,64	0,29	-0,17	0,75	0,03	-0,15	0,26
D	0,53	-0,57	1,72	0,31	-0,19	0,82	0,03	-0,20	0,23
E	0,58	-0,67	1,86	0,32	-0,24	0,89	0,02	-0,23	0,30
TcredC	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01
sex									
Male	-0,29	-0,74	0,14	-0,26	-0,41	-0,11	-0,28	-0,33	-0,23
SAdnEx	0,03	0,01	0,04	0,03	0,02	0,03	0,02	0,02	0,02
THighSch									
Federal	0,19	-0,74	1,32	0,10	-0,34	0,52	0,00	-0,13	0,14
Estadual	0,20	-0,74	1,13	0,06	-0,30	0,42	-0,03	-0,17	0,10
Particular	0,04	-0,82	0,98	-0,08	-0,44	0,28	-0,17	-0,28	-0,05
σ_y^2	3,61	1.72	12,30	1,27	1,05	1,61	0,50	0,47	0,52
	LASSO			PPRM			NIPPD - η = 100 δ =0,01		
	Med	HPD 95%			HPD 95%			HPD 95%	
		Inf	Sup	Med	Inf	Sup	Med	Inf	Sup
SocEco									
B	0,00	-0,15	0,10	0,02	-0,11	0,15	0,23	-0,02	0,46
C	0,00	-0,10	0,14	0,05	-0,07	0,20	0,27	0,05	0,53
D	0,00	-0,20	0,17	0,04	-0,09	0,20	0,27	0,03	0,51
E	0,00	-0,48	0,36	0,04	-0,15	0,26	0,25	-0,05	0,51
TcredC	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01
sex									
Male	-0,34	-0,56	-0,11	-0,28	-0,33	-0,24	-0,27	-0,33	-0,22
SAdmEx	0,00	-0,00	0,01	0,02	0,02	0,02	0,02	0,02	0,03
THighSch2									
Federal	0,00	-0,18	0,21	0,01	-0,11	0,12	0,08	-0,06	0,24
Estadual	0,00	-0,10	0,27	-0,02	-0,12	0,08	0,07	-0,09	0,20
Particular	0,00	-0,26	0,06	-0,15	-0,25	-0,06	-0,08	-0,22	0,05
σ_y^2	11,43	10,96	11,91	0,44	0,42	0,46	0,50	0,48	0,53

Fonte: Elaborado pela autora.

cientes "encolhidos" no modelo. Para o modelo PPRM o valor foi o mais baixo (0,44). Para o modelo NIPPD foi de 0,50 como este é um modelo aproximadamente normal seu valor é um pouco maior do que o estimado pelo PPRM.

A maioria dos cursos que apresentaram mais variabilidade na distribuição *a posteriori* dos efeitos são noturnos. Muitos alunos que estudam em cursos noturnos trabalham durante o dia o que diminui o tempo e a disposição para os estudos e pode afetar o desempenho. Uma investigação sobre o efeito dessa covariável pode melhorar a explicação da variação do RSGM. Algo que poderia ser investigado

em próximos trabalhos.

4.4 Considerações finais

Uma observação relevante sobre todos os ajustes do modelo NIPPD é que nenhuma das cadeias de K, número de grupos, após a exclusão do período de aquecimento, apresentou mais que 13 grupos em sua distribuição *a posteriori*, mesmo nos modelos com $K_{max} = 29$. Isto mostra que o NIPPD apresenta informação consistente sobre o número de grupos. Comparado com os outros modelos, possui capacidade preditiva semelhante, mas mostra mais parcimônia na formação de grupos, nem tantos, quanto o modelo PPRM, nem tão poucos, quanto o modelo LASSO. Em geral, os efeitos estimados e grupos são mais parecidos com os do modelo PPRM.

Sobre os aspectos Computacionais, o trabalho faz indicações sobre os valores iniciais dos parâmetros do kernel θ^2 e σ_q^2 e as escolha de hiperparâmetros das suas prioris e apresenta algoritmo MCMC capaz de estimar seus valores. Apesar de não ter um limitante superior teórico para os valores de σ_q^2 a relação proposta com os valores de θ^2 forneceu bons resultados de estimação.

Como objetivos futuros tem-se a comparação com o modelo Dirichlet que não impõe ordenação a variável categórica e o ajuste a dados que possuam alguma ordenação para avaliar o comportamento do modelo.

Capítulo 5

Conclusão e Propostas de Continuidade

Este trabalho fornece um método flexível de agrupamento não supervisionado que incorpora uma característica de repulsão no comportamento dos parâmetros de locação que representam os *clusters*. A proposta é um Modelo de Mistura Finita de Distribuições Normal/Independente desenvolvido considerando o comportamento dos parâmetros de locação como uma realização de um Processo Pontual por Determinante (PPD), distribuição de probabilidade que evita a criação de grupos redundantes através de sua característica natural de repulsão de pontos e com uma abordagem Bayesiana para estimação dos seus parâmetros.

O modelo foi avaliado em um estudo de simulação e em uma aplicação em dados de demanda de agronegócio, mostrando-se eficiente na estimação de densidade. Foi avaliado também na identificação de grupos e na redução de dimensionalidade para variáveis categóricas com muitos níveis em dados de desempenho educacional. Os resultados mostram que o modelo oferece uma alternativa ao uso de penalidades como as do tipo LASSO, é parcimonioso na estimação do número de grupos e apresenta agrupamentos e que fazem sentido. Além disso, foram desenvolvidos algoritmos de Markov Chain Monte Carlo (MCMC) para a estimação dos parâmetros do modelo, seguindo o paradigma Bayesiano, e as rotinas foram impletadas em R e C^{++} por meio da biblioteca *Rcpp*.

Os resultados deste trabalho motivam sua continuidade em estudos futuros, por exemplo, podem ser explorados outros aspectos do modelo como a estimação dos graus de liberdade a partir de uma distribuição *a priori* no modelo; a dependência espacial definida a partir dos pesos da mistura, no caso em que a variável resposta esteja indexada por coordenadas; e/ou a dependência temporal considerando Processo Pontual por Determinante Condicional. Em termos computacionais um algoritmo de *Reversible Jump Markov Chain Monte Carlo* (RJMCMC) pode ser desenvolvido para comparar com os resultados já obtidos e a necessidade de definir um número máximo de componentes da mistura.

References

[1] Raja Hafiz Affandi, Emily Fox, Ryan Adams, and Ben Taskar. Learning the parameters of determinantal point process kernels. In *In International Conference on Machine Learning*, pages 1224–1232, 2014.

[2] Raja Hafiz Affandi, Emily B. Fox, and Ben Taskar. Approximate inference in continuous determinantal point processes. In *Advances in Neural Information Processing Systems*, 2013.

[3] Raja Hafiz Affandi, Alex Kulesza, and Emily B. Fox. Markov determinantal point processes. In *Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence, UAI*, pages 26–35, 10 2012.

[4] Jared Aldstadt and Arthur Getis. Using amoeba to create a spatial weights matrix and identify spatial clusters. *Geographical Analysis*, 38(4):327–343, 2006.

[5] David F. Andrews and Colin L. Mallows. Scale mixtures of normal distributions. *Journal of the Royal Statistical Society: Series B (Methodological)*, 36(1):99–102, 1974.

[6] Charles E. Antoniak. Mixtures of Dirichlet Processes with Applications to Bayesian Nonparametric Problems. *The Annals of Statistics*, 2(6):1152 – 1174, 1974.

[7] Renato M. Assunção, Marcos Corrêa Neves, Gilberto Câmara, and Corina da Costa Freitas. Efficient regionalization techniques for socio-economic geographical units using minimum spanning trees. *International Journal of Geographical Information Science*, 20(7):797–811, 2006.

[8] Adrian Baddeley, Ege Rubak, and Rolf Turner. *Spatial point patterns: methodology and applications with R*. Chapman & Hall CRC, London, 2015.

[9] Rémi Bardenet, Frédéric Lavancier, Xavier Mary, and Aurélien Vasseur. On a few statistical applications of determinantal point processes. *ESAIM: Proceedings and Surveys*, 60:180–202, 2017.

[10] Francesco Bartolucci and Luisa Scaccia. The use of mixtures for dealing with non-normal regression errors. *Computational Statistics & Data Analysis*, 48(4):821–834, 2005.

[11] Dan Bebber, Sarah Gurr, Steve McCorriston, Gero Steinberg, and Gillian Petrokofsky. Toward a global banana map. https://bananex.org/2020/03/04/toward-a-global-banana-map/, August 2020.

[12] Salil Bharany, Sandeep Sharma, Jaroslav Frnda, Mohammed Shuaib, Muhammad Irfan Khalid, Saddam Hussain, Jawaid Iqbal, and Syed Sajid Ullah. Wildfire monitoring based on energy efficient clustering approach for fanets. *Drones*, 6(8):193, 2022.

[13] Ilaria Bianchini, Alessandra Guglielmi, and Fernando A. Quintana. Determinantal Point Process Mixtures Via Spectral Density Approach. *Bayesian Analysis*, 15(1):187 – 214, 2020.

[14] David A. Binder. Bayesian cluster analysis. *Biometrika*, 65(1):31–38, 1978.

[15] Christophe AN Biscio and Jean-François Coeurjolly. Standard and robust intensity parameter estimation for stationary determinantal point processes. *Spatial Statistics*, 18:24–39, 2016.

[16] Dankmar Böhning, Christian Hennig, Geoffrey J McLachlan, and Paul D McNicholas. The 2nd special issue on advances in mixture models. *Computational Statistics & Data Analysis*, 71:1–2, 2014.

[17] Howard D. Bondell and Brian J. Reich. Simultaneous factor selection and collapsing levels in anova. *Biometrics*, 65(1):169–177, 2009.

[18] Ana Lúcia Borges. Palestras técnico-científicas apresentadas na embrapa mandioca e fruticultura 2010 e 2011. *Embrapa Mandioca e Fruticultura-Documentos (INFOTECA-E)*, 2014.

[19] Alexei Borodin. *Determinantal point processes*. In Oxford Handbook of Random Matrix Theory. Oxford University Press, New York, 2010.

[20] Victor-Emmanuel Brunel, Ankur Moitra, Philippe Rigollet, and John Urschel. Maximum likelihood estimation of determinantal point processes. *arXiv preprint arXiv:1701.06501*, 01 2017.

[21] G. Celeux, F. Forbes, C. P. Robert, and D. M. Titterington. Deviance information criteria for missing data models. *Bayesian Analysis*, 1(4):651 – 673, 2006.

[22] Sung Nok Chiu, Dietrich Stoyan, Wilfrid S Kendall, and Joseph Mecke. *Stochastic geometry and its applications*. John Wiley & Sons, New Jersey, 2013.

[23] Samantha Cockings and David Martin. Zone design for environment and health studies using pre-aggregated data. *Social Science & Medicine*, 60(12):2729–2742, 2005.

[24] Zilton José Maciel Cordeiro, AP DE Matos, and Paulo Ernesto Meissner Filho. Doenças e métodos de controle. *O Cultivo da Bananeira*, 1:146–182, 2004.

[25] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[26] Tulio L. Criscuolo, Renato M. Assunção, Rosangela H. Loschi, Wagner Meira J.R., and Danna Cruz-Reyes. Handling categorical features with many levels using a product partition model. *The Annals of Applied Statistics*, pages 2150–2180, 2023.

[27] Adele Cutler and Olga I. Cordero-Brana. Minimum hellinger distance estimation for finite mixture models. *Journal of the American Statistical Association*, 91(436):1716–1723, 1996.

[28] Daryl J. Daley, David Vere-Jones, et al. *An introduction to the theory of point processes: volume I: elementary theory and methods*. Springer, 2003.

[29] Petros Dellaportas and Claudia Tarantola. Model determination for categorical data with factor level merging. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):269–283, 2005.

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

[31] David G.T. Denison and Christofer C. Holmes. Bayesian partitioning for estimating disease risk. *Biometrics*, 57(1):143–149, 2001.

[32] Peter J Diggle. *Statistical analysis of spatial and spatio-temporal point patterns*. Chapman & Hall ,CRC, London, 2013.

[33] Christine DiStefano, Dexin Shi, and Grant B. Morgan. Collapsing categories is often more advantageous than modeling sparse data: Investigations in the cfa framework. *Structural Equation Modeling: A Multidisciplinary Journal*, 28(2):237–249, 2021.

[34] Freeman J Dyson. Statistical theory of the energy levels of complex systems. i. *Journal of Mathematical Physics*, 3(1):140–156, 1962.

[35] Dirk Eddelbuettel and Romain François. Rcpp: Seamless R and C++ integration. *Journal of Statistical Software*, 40(8):1–18, 2011.

[36] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press, 1996.

[37] Brian S. Everitt, Sabine Landau, Morven Leese, and Daniel Stahl. *Cluster Analysis*. John Wiley & Sons, New Jersey, 2011.

[38] Fina Faithpraise, Philip Birch, Rupert Young, J. Obu, Bassey Faithpraise, and Chris Chatwin. Automatic plant pest detection and recognition using k-means clustering algorithm and correspondence filters. *International Journal of Advanced Biotechnology and Research*, 4(2):189–199, 2013.

[39] Gregory E Fasshauer and Michael J McCourt. Stable evaluation of gaussian radial basis function interpolants. *SIAM Journal on Scientific Computing*, 34(2):A737–A762, 2012.

[40] Thomas S. Ferguson. A bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209 – 230, 1973.

[41] Arno Fritsch and Katja Ickstadt. Improved criteria for clustering based on the posterior similarity matrix. *Bayesian Analysis*, 4(2):367 – 391, 2009.

[42] Sylvia Frühwirth-Schnatter. Markov chain monte carlo estimation of classical and dynamic switching and mixture models. *Journal of the American Statistical Association*, 96(453):194–209, 2001.

[43] E.F. da Gama-Rodrigues and A.C. da Gama-Rodrigues. Biomassa microbiana e ciclagem de nutrientes. *Fundamentos da matéria orgânica do solo: ecossistemas tropicais e subtropicais. Porto Alegre: Gênesis*, pages 227–243, 1999.

[44] Emanuela Gama-Rodrigues and Antonio Gama-Rodrigues. *Biomassa Microbiana e Ciclagem de Nutrientes*, pages 1–12. Genesis, Porto Alegre, 08 2008.

[45] Alan E Gelfand and Adrian FM Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85(410):398–409, 1990.

[46] Jan Gertheiss and Gerhard Tutz. Sparse modeling of categorial explanatory variables. *The Annals of Applied Statistics*, 4(4):2150–2180, 2010.

[47] Jennifer A Gillenwater, Alex Kulesza, Emily Fox, and Ben Taskar. Expectation-maximization for learning determinantal point processes. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27, page 3149–3157. Curran Associates, Inc., 2014.

[48] Jean Ginibre. Statistical ensembles of complex, quaternion, and real matrices. *Journal of Mathematical Physics*, 6(3):440–449, 1965.

[49] Flávio B. Gonçalves, Marcos O. Prates, and Victor Hugo Lachos. Robust Bayesian model selection for heavy-tailed linear regression using finite mixtures. *Brazilian Journal of Probability and Statistics*, 34(1):51 – 70, 2020.

[50] Diansheng Guo, Donna J. Peuquet, and Mark Gahegan. Iceage: Interactive clustering and exploration of large and high-dimensional geodata. *GeoInformatica*, 7:229–253, 2003.

[51] John A. Hartigan. Partition models. *Communications in Statistics-Theory and Methods*, 19(8):2745–2756, 1990.

[52] John A Hartigan and Manchek A Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.

[53] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 04 1970.

[54] J Ben Hough, Manjunath Krishnapur, Yuval Peres, Bálint Virág, Bálint ág, et al. Determinantal processes and independence. *Probability Surveys*, 3:206–229, 2006.

[55] Lynette Hunt and Murray Jorgensen. Theory & methods: Mixture model clustering using the multimix program. *Australian & New Zealand Journal of Statistics*, 41(2):154–171, 1999.

[56] A Jasra, CC Holmes, and DA Stephens. Markov chain monte carlo methods and the label switching problem in bayesian mixture modeling. *Statistical Science*, 20(1):50–67, 2005.

[57] Mon-Fong Jiang, Shian-Shyong Tseng, and Chih-Ming Su. Two-phase clustering process for outliers detection. *Pattern Recognition Letters*, 22(6-7):691–700, 2001.

[58] Kurt Johansson. Determinantal processes with number variance saturation. *Communications in Mathematical Physics*, 252(1-3):111–148, 2004.

[59] Kurt Johansson. The arctic circle boundary and the airy process. *The Annals of Probability*, 33(1):1–30, 2005.

[60] Christine Keribin. Consistent estimation of the order of mixture models. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 49–66, 2000.

[61] Mutsuki Kojima and Fumiyasu Komaki. Determinantal point process priors for bayesian variable selection in linear regression. *Statistica Sinica*, pages 97–117, 2016.

[62] Hans-Peter Kriegel, Peer Kröger, Jörg Sander, and Arthur Zimek. Density-based clustering. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge discovery*, 1(3):231–240, 2011.

[63] Alex Kulesza and Ben Taskar. Determinantal point processes for machine learning. 5(2–3).

[64] Alex Kulesza and Ben Taskar. Structured determinantal point processes. In *Advances in Neural Information Processing Systems*, pages 1171–1179, 2010.

[65] Kenneth Lange and Janet S. Sinsheimer. Normal/independent distributions and their applications in robust regression. *Journal of Computational and Graphical Statistics*, 2(2):175–198, 1993.

[66] Frédéric Lavancier, Jesper Møller, and Ege Rubak. Determinantal point process models and statistical inference. *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, pages 853–877, 2015.

[67] Sharon Lee and Geoffrey J McLachlan. Finite mixtures of multivariate skew t-distributions: some recent and new results. *Statistics and Computing*, 24:181–202, 2014.

[68] Jia Li. Clustering based on a multilayer mixture model. *Journal of Computational and Graphical Statistics*, 14(3):547–568, 2005.

[69] Bruce G. Lindsay. Mixture models: Theory, geometry and applications. In *NSF-CBMS Regional Conference Series in Probability and Statistics*, volume 5, pages i–163. JSTOR, 1995.

[70] Chuanhai Liu and Donald B Rubin. Ml estimation of the t distribution using em and its extensions, ecm and ecme. *Statistica Sinica*, pages 19–39, 1995.

[71] David G. Lowe. Local feature view clustering for 3d object recognition. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.

[72] Zenghong Ma, Zeyi Tao, Xiaoqiang Du, Yaxin Yu, and Chanyu Wu. Automatic detection of crop root rows in paddy fields based on straight-line clustering algorithm and supervised learning method. *Biosystems Engineering*, 211:63–76, 2021.

[73] Odile Macchi. The coincidence approach to stochastic point processes. *Advances in Applied Probability*, 7(1):83–122, 1975.

[74] Gertraud Malsiner-Walli, Sylvia Frühwirth-Schnatter, and Bettina Grün. Model-based clustering based on sparse finite gaussian mixtures. *Statistics and Computing*, 26(1-2):303–324, 2016.

[75] Geoffrey J McLachlan and Kaye E Basford. *Mixture models: Inference and applications to clustering*, volume 38. M. Dekker, New York, 1988.

[76] Geoffrey J. McLachlan and David Peel. *Finite mixture models*. John Wiley & Sons, New Jersey, 2004.

[77] Madan Lal Mehta and Michel Gaudin. On the density of eigenvalues of a random matrix. *Nuclear Physics*, 18:420–427, 1960.

[78] Marina Meilă. Comparing clusterings—an information based distance. *Journal of multivariate analysis*, 98(5):873–895, 2007.

[79] Kerrie L Mengersen, Christian Robert, and Mike Titterington. *Mixtures: estimation and applications*. John Wiley & Sons, New Jersey, 2011.

[80] Daniele Micci-Barreca. A preprocessing scheme for high-cardinality categorical attributes in classification and prediction problems. *ACM SIGKDD Explorations Newsletter*, 3(1):27–32, 2001.

[81] Simon Newcomb. A generalized theory of the combination of observations so as to obtain the best result. *American Journal of Mathematics*, pages 343–366, 1886.

[82] Garritt L. Page, Fernando Andrés Quintana, and David B. Dahl. Spatio-temporal random partition models. *arXiv: Methodology*, 2019.

[83] Trevor Park and George Casella. The bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686, 2008.

[84] Raj Kumar Pathria and Paul D Beale. *Statistical Mechanics*. Elsevier, Boston, 2011.

[85] Daniela Pauger and Helga Wagner. Bayesian effect fusion for categorical predictors. *Bayesian Analysis*, 14(2):341–369, 2019.

[86] Karl Pearson. Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London. A*, 185:71–110, 1894.

[87] David Peel and Geoffrey J McLachlan. Robust mixture modelling using the t distribution. *Statistics and Computing*, 10:339–348, 2000.

[88] Francesca Petralia, Vinayak Rao, and David B Dunson. Repulsive mixtures. In *Advances in Neural Information Processing Systems*, pages 1889–1897, 2012.

[89] José J Quinlan, Garritt L Page, and Fernando A Quintana. Density regression using repulsive distributions. *Journal of Statistical Computation and Simulation*, 88(15):2931–2947, 2018.

[90] José J Quinlan, Fernando A Quintana, and Garritt L Page. On a class of repulsive mixture models. *TEST*, pages 1–17, 2020.

[91] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2023.

[92] Douglas A Reynolds. Gaussian mixture models. In *Encyclopedia of biometrics*, pages 1889–1897. Springer US, Boston, MA, New York, 2009.

[93] Patricia B. Ribeiro, Roseli A.F. Romero, Patrícia R. Oliveira, Homero Schiabel, and Luciana B. Verçosa. Automatic segmentation of breast masses using enhanced ica mixture model. *Neuro-Computing*, 120:61–71, 2013.

[94] Sylvia Richardson and Peter J. Green. On bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society: series B (statistical methodology)*, 59(4):731–792, 1997.

[95] Thomas C. Ricketts. *Using geographic methods to understand health issues*. Agency for Health Care Policy and Research, Department of Health and Human Services, 1997.

[96] Veronika Ročková and Edward I. George. *Determinantal Priors for Variable Selection*, pages 129–136. Springer International Publishing, Cham, 2022.

[97] William H. Rogers and John W. Tukey. Understanding some long-tailed symmetrical distributions. *Statistica Neerlandica*, 26(3):211–226, 1972.

[98] Zeév Rudnick, Peter Sarnak, et al. Zeros of principal l-functions and random matrix theory. *Duke Mathematical Journal*, 81(2):269–322, 1996.

[99] Leslie Rutkowski, Dubravka Svetina, and Yuan-Ling Liaw. Collapsing categorical variables and measurement invariance. *Structural Equation Modeling: A Multidisciplinary Journal*, 26(5):790–802, 2019.

[100] J Schur. Über potenzreihen, die im innern des einheitskreises beschränkt sind. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 1918(148):122–145, 1918.

[101] A. J. Scott and M. J. Symons. Clustering methods based on likelihood ratio criteria. *Biometrics*, 27(2):387–397, 1971.

[102] Tomoyuki Shirai and Yoichiro Takahashi. Random point fields associated with certain fredholm determinants i: fermion, poisson and boson point processes. *Journal of Functional Analysis*, 205(2):414–463, 2003.

[103] Tomoyuki Shirai and Yoichiro Takahashi. Random point fields associated with certain fredholm determinants ii: fermion shifts and their ergodic and gibbs properties. *The Annals of Probability*, 31(3):1533–1564, 2003.

[104] Robin Sibson. Slink: an optimally efficient algorithm for the single-link cluster method. *The Computer Journal*, 16(1):30–34, 1973.

[105] Gabriele Soffritti and Giuliano Galimberti. Multivariate linear regression with non-normal errors: a solution based on mixture models. *Statistics and Computing*, 21:523–536, 2011.

[106] Matthew Stephens. *Bayesian methods for mixtures of normal distributions.* PhD thesis, University of Oxford, 1997.

[107] Leonardo Vilela Teixeira, Renato M Assunção, and Rosangela Helena Loschi. Bayesian space-time partitioning by sampling and pruning spanning trees. *J. Mach. Learn. Res.*, 20:85–1, 2019.

[108] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[109] D Michael Titterington, Adrian FM Smith, and Udi E Makov. *Statistical analysis of finite mixture distributions*. John Wiley, New Jersey, 1985.

[110] Marj Tonini, Devis Tuia, and Frédéric Ratle. Detection of clusters using space–time scan statistics. *International Journal of Wildland Fire*, 18(7):830–836, 2009.

[111] M.R. Tótola and G.M. Chaer. Microrganismos e processos microbiológicos como indicadores da qualidade dos solos. *Tópicos em ciência do solo*, 2(3):195–276, 2002.

[112] Bulent Tutmez, Mert G. Ozdogan, and Ahmet Boran. Mapping forest fires by nonparametric clustering analysis. *Journal of Forestry Research*, 29:177–185, 2018.

[113] Charles F. Van Loan and G. Golub. *Matrix computations (Johns Hopkins studies in mathematical sciences)*. Johns Hopkins University Press, Baltimore and London, 3rd edition, 1996.

[114] Sara Wade and Zoubin Ghahramani. Bayesian cluster analysis: Point estimation and credible balls (with discussion). *Bayesian Analysis*, 13(2):559–626, 06 2018.

[115] Mike West and Michael D Escobar. *Hierarchical priors and mixture models, with application in regression and density estimation*. Institute of Statistics and Decision Sciences, Duke University, 1993.

[116] Petros Xanthopoulos, Panos M. Pardalos, Theodore B. Trafalis, Petros Xanthopoulos, Panos M. Pardalos, and Theodore B. Trafalis. Linear discriminant analysis. *Robust data mining*, pages 27–33, 2013.

[117] Fangzheng Xie and Yanxun Xu. Bayesian repulsive gaussian mixture model. *Journal of the American Statistical Association*, 115(529):187–203, 2020.

[118] Yanxun Xu, Peter Müller, and Donatello Telesca. Bayesian inference for latent biologic structure with determinantal point processes (dpp). *Biometrics*, 72(3):955–964, 2016.

[119] Zhihua Zhang, Kap Luk Chan, Yiming Wu, and Chibiao Chen. Learning a multivariate gaussian mixture model with the reversible jump mcmc algorithm. *Statistics and Computing*, 14(4):343–355, 2004.

Apêndice A

Material Complementar

Neste capítulo, serão apresentados as bases matemáticas e cálculos necessários para obtenção dos resultados presentes na tese. Os lemas e teoremas matriciais podem ser encontrados em [113].

A.1 Aspectos Matemáticos

A.1.1 Família de Distribuições Normal/Independente

Pode-se dizer que um vetor aleatório $\mathbf{X}$ possui distribuição pertencente à Família de Distribuições Normal Independente - NI, se sua distribuição tem função densidade de probabilidade dada por

$$f_{X|\boldsymbol{\mu};\boldsymbol{\Sigma};\boldsymbol{\nu}}(\mathbf{x}) = \int_0^\infty \frac{u^{n/2}}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{ -\frac{u}{2}(\mathbf{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\} dF_{\mathbf{U}|\boldsymbol{\nu}}(u), \tag{A.1}$$

com $\boldsymbol{\mu}$ sendo o parâmetro de locação, $\boldsymbol{\Sigma}$ o parâmetro de escala e $\boldsymbol{\nu}$ parâmetro de forma, $\boldsymbol{X} \sim NI(\boldsymbol{\mu};\boldsymbol{\Sigma};F_{U|\boldsymbol{\nu}})$.

Esta família também pode ser definida por sua forma estocástica [65] apresentada no Lema a seguir.

Lema A.1.1. *Seja X um vetor aleatório que possui distribuição Normal/Independente - NI, D-variada com parâmetros de locação $\boldsymbol{\mu}$ e de escala $\boldsymbol{\Sigma}$ então*

$$X = \boldsymbol{\mu} + \boldsymbol{U}^{-1/2}\boldsymbol{\Sigma}^{1/2}\mathbf{T},$$

em que $T \sim N_D(\mathbf{0};\mathbf{I}_n)$ e U é uma variável positiva com f.d.p. $f_{U|\nu}(u)$.

Proposição A.1.1. *Se $U \sim Gama(\nu/2;\nu/2)$ então $\mathbf{X}$ tem distribuição t-Student multivariada com parâmetros de locação $\boldsymbol{\mu}$ e de escala $\boldsymbol{\Sigma}$ e com graus de liberdade $\nu > 0$, $\boldsymbol{X} \sim T_D(\boldsymbol{\mu};\boldsymbol{\Sigma};\nu)$ e com f.d.p. dada por*

$$f_{X|\boldsymbol{\mu};\boldsymbol{\Sigma};\nu}(x) = \frac{\Gamma[(\nu+n)/2]}{|\boldsymbol{\Sigma}|^{1/2}\pi^{n/2}\Gamma(\nu/2)\nu^{n/2}}$$

em que a matriz de covariância de X é dada por

$$V[X] = \frac{\nu}{\nu-2}\boldsymbol{\Sigma}.$$

A.1.2 Decomposição em Valores Singulares de $\boldsymbol{\Sigma}$

Proposição A.1.2. *Decomposição em Valores Singulares*

Seja A uma matriz real com dimensões $m \times n$, então existem matrizes ortogonais

$$U = [u_1, \ldots, u_m] \in \mathbb{R}^{m \times m} \text{ e } V = [v_1, \ldots, v_n] \in \mathbb{R}^{n \times n}$$

tais que

$$U'AV = diag(s_1, \ldots, s_p) \in \mathbb{R}^{m \times n} \text{ com } p = min\{n, m\}.$$

em que $s_1 \geq \cdots \geq s_p \geq 0$.

Lema A.1.2. *Decomposição Simétrica de Schur*

Se A é uma matriz simétrica real com dimensões $n \times n$, então existe matriz ortogonal Q, $n \times n$, tal que

$$Q'AQ = diag(\lambda_1, \ldots, \lambda_n)$$

e $\lambda_1, \ldots, \lambda_n$ são os autovalores de A.

Como a matriz escala do modelo, $\boldsymbol{\Sigma}_k$, $k = 1, 2, \ldots, K$, é simétrica, então existe E ortogonal, $D \times D$, tal que $E'\boldsymbol{\Sigma}_k E = diag(\tau_1, \ldots, \tau_D)$, logo $\boldsymbol{\Sigma}_k$ pode ser escrita na forma

$$\boldsymbol{\Sigma}_k = E\, \mathcal{T}_k E',$$

pois $E^{-1} = E'$, pela sua ortogonalidade.

A.1.3 Complemento de Schur na Distribuição Condicional Completa de $\boldsymbol{\mu}$

Proposição A.1.3. *Complemento de Schur*

Seja A uma matriz real de dimensão $n \times n$, particionada da seguinte forma

$$A = \begin{bmatrix} A_{11} & A_{21} \\ A_{12} & A_{22} \end{bmatrix},$$

em que A_{11} é submatriz $r \times r$ não singular. A matriz $S = A_{22} - A_{21}A_{11}^{-1}A_{12}$ é chamada Complemento de Schur de A_{11} em A.

Dessa forma, a matriz A tem a seguinte decomposição

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & A_{11}^{-1}A_{12} \\ A_{21} & A_{22} \end{bmatrix}.$$

Lema A.1.3. *Determinante de Schur*

Seja A nas condições da Proposição A.1.3, seu determinante é dado por

$$det(A) = det(A_{11})det(A_{22} - A_{21}A_{11}^{-1}A_{12}). \tag{A.2}$$

Considerando a função densidade a priori do vetor de médias dada em (3.11), $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K)|$ $K, \theta, \sigma_q \sim DPP(C, \theta, \sigma_q)$, sua função densidade é proporcional ao determinante

$$p(\boldsymbol{\mu}|\theta, \sigma_q, K) \propto \det(C_{\theta,\sigma_q}(\boldsymbol{\mu})).$$

Utilizando a partição $\boldsymbol{\mu} = (\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_k)$, em que $\boldsymbol{\mu}_{-k} = \{\boldsymbol{\mu}_j\}_{j \neq k}$, a matriz pode ser particionada da seguinte forma

$$C_{\theta,\sigma_q}(\boldsymbol{\mu}) = \begin{bmatrix} C(\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_{-k}) & C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})' \\ C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k}) & C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_k) \end{bmatrix},$$

e seu determinante pode ser obtido usando (A.2):

$$\begin{aligned} \det\left(C_{\theta,\sigma_q}(\boldsymbol{\mu})\right) &= \det\left(C(\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_{-k})\right) \\ &\times \det\left(C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_k) - C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})C(\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_{-k})^{-1}C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})'\right), \end{aligned}$$

mas a matriz contida no segundo determinante é 1×1, logo

$$\begin{aligned} \det\left(C_{\theta,\sigma_q}(\boldsymbol{\mu})\right. &= \det\left(C(\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_{-k})\right) \\ &\times \left(C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_k) - C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})C(\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_{-k})^{-1}C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})'\right). \end{aligned}$$

Este resultado permite obter o núcleo da priori para um componente k qualquer do vetor de locação, $\boldsymbol{\mu}_k$. Observa-se que $C(\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_{-k})$ não dependende do componente $\boldsymbol{\mu}_k$, logo

$$p(\boldsymbol{\mu}_k|\boldsymbol{\mu}_{-k}) \propto \left(C_{\mu_k} - C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})C_{\mu_{-k}}^{-1}C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})'\right).$$

Além disso, $p(\boldsymbol{\mu}_k|\ldots)$ pode ser utilizada na distribuição condicional completa de $\boldsymbol{\mu}_k$.

A.1.4 Restrição no Kernel

Segundo [8], um DPP válido só é possível se houver equilíbrio entre a intensidade de ocorrência de pontos e o intervalo de repulsão neste tipo de kernel. Assim, para função intensidade definida em 2.4 é essencial definir uma condição que garanta a validade do DPP.

Em particular, para o kernel exponencial quadrático no modelo proposto temos

$$\rho(\boldsymbol{\mu}) = q^2(\boldsymbol{\mu}).$$

No contexto do kernel exponencial quadrático, θ é parâmetro de escala e pode ser interpretado como a distância mínima entre os pontos para que uma probabilidade razoável de ocorrência seja obtida em um espaço limitado. Dessa forma, grandes valores de θ, exigem pontos separados por uma grande distância e num espaço limitado isso pode inviabilizar probabilidades positivas.

Assim, é necessário impor um limite à função intensidade, $\rho = q^2(\cdot)$ e a função qualidade não pode ser considerada independente de θ.

Proposição A.1.4. *A restrição que garante um DPP válido com o kernel exponencial quadrático definido com funções qualidade dada em 2.8 e similaridade dada em 2.9, é dada por*

$$\theta^2 \leq \frac{(2\pi\sigma_q^2)^D}{\pi}. \tag{A.3}$$

Prova A.1.5. *O equilíbrio entre a função intensidade e o intervalo de repulsão definido por [8], neste caso, é*

$$[q(\boldsymbol{\mu})]^2 \leq 1/(\pi\theta^2), \forall\ \boldsymbol{\mu} \in B \subseteq \mathbb{R}^D. \tag{A.4}$$

Usando o máximo da função $q(\cdot)$ como limitante, ou seja, fazendo $\boldsymbol{\mu} = 0$, temos

$$\left(\prod_{d=1}^{D} \frac{1}{\sqrt{2\pi}\sigma_q}\right)^2 \leq \frac{1}{\pi\theta^2}.$$

Mas $\theta \geq 0$, então

$$0 \leq \theta \leq \frac{(2\pi\sigma_q^2)^{D/2}}{\sqrt{\pi}}.$$

Para $D = 2, 0 \leq \theta \leq \dfrac{2\pi\sigma_q^2}{\sqrt{\pi}}$ e para $D = 1, 0 \leq \theta \leq \sqrt{\dfrac{2\pi\sigma_q^2}{\pi}}$.

A.2 Distribuições Condicionais Completas do Modelo de Mistura NIDPP para estimação de densidade

A.2.0.1 Função de verossimilhança

Para $\mathbf{y}_i|z_i = k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, K \ldots \sim N_D(\boldsymbol{\mu}_k, u_k^{-1}\boldsymbol{\Sigma}_k)$ a função de verossimilhança dada por

$$L(\mathbf{z}, \boldsymbol{\mu}, \mathbf{u}, \boldsymbol{\Sigma}, K, \ldots) = \prod_{k=1}^{K} \prod_{i:z_i=k} \frac{\exp\left\{-\frac{u_i}{2}(\mathbf{y}_i - \boldsymbol{\mu}_k)'\boldsymbol{\Sigma}_k^{-1}(\mathbf{y}_i - \boldsymbol{\mu}_k)\right\}}{\left|u_i^{-1}\boldsymbol{\Sigma}_k\right|^{1/2}(2\pi)^{D/2}} \tag{A.5}$$

A.2.0.2 Distribuição Condicional Completa para w

Para os pesos **w** a distribuição Dirichlet é de família conjugada à distribuição Normal considerada na função de verossimilhança, assim a sua distribuição posteriori é dada por

$$p(\mathbf{w}|\dots) = Dir(\delta + n_1, \dots, \delta + n_K), \text{ em que } n_k = \sum_{i=1}^{N} I_{(z_1=k)}. \tag{A.6}$$

A.2.0.3 Condicional Completa de z

Para a partição **z** sua distribuição a priori é discreta, independente e identicamente distribuída para cada indivíduo i, mas a distribuição condicional completa também é discreta definida em um conjunto finito de valores, portanto, sua distribuição é obtida diretamente pela normalização da distribuição proporcional abaixo.

$$p(z_i = k|\dots) \propto p(z_i = k)L(\mathbf{z}, \boldsymbol{\mu}, \mathbf{u}, \boldsymbol{\Sigma}, K, \dots) \propto w_k \prod_{i=1}^{N} \prod_{k=1}^{K} N_D(y_i|\boldsymbol{\mu}_k, u_k\boldsymbol{\Sigma}_k). \tag{A.7}$$

$$\text{com } \sum_{k=1}^{K} P(z_i = k|\mathbf{w}, K) = \sum_{k=1}^{K} w_k = 1.$$

A.2.0.4 Condicional Completa de u

Para o parâmetro misturador, a priori, temos uma distribuição Gama para cada componente do vetor **u**, considerados independentes entre si e dos demais parâmetros.

Para o parâmetro misturador, $\mathbf{u} = \mathbf{u} = (u_1, \dots, u_n)$, a distribuição a priori é da família Gama, $u_i \sim Gama(\eta/2, \eta/2)$, que é conjugada à família Normal da verossimilhança.

$$\begin{aligned}
p(u_i|\dots) &\propto p(u_i)L(\mathbf{z}, \boldsymbol{\mu}, \mathbf{u}, \boldsymbol{\Sigma}, K, \dots) \\
&\propto u_i^{(\eta/2-1)} e^{-u_i\eta/2} \frac{\exp\Big\{-\frac{u_i}{2}(\mathbf{y}_i - \boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_k E')^{-1}(\mathbf{y}_i - \boldsymbol{\mu}_k)\Big\}}{\left|u_i^{-1} E\boldsymbol{\mathcal{T}}_k E'\right|^{1/2}} \\
&\propto u_i^{(\eta/2-1)} e^{-u_i\eta/2} \left|u_i^{-1} E\boldsymbol{\mathcal{T}}_k E'\right|^{-1/2} \exp\Big\{-\frac{u_i}{2}(\mathbf{y}_i - \boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_k E')^{-1}(\mathbf{y}_i - \boldsymbol{\mu}_k)\Big\}.
\end{aligned}$$

Temos u_i no determinante, assim $\left|u_i^{-1} E\boldsymbol{\mathcal{T}}_k E'\right| = u_i^{-D}|E\boldsymbol{\mathcal{T}}_k E'\Big|$ como os outros termos são constantes em relação a $u_i\, u_i^{-D} \prod_{d=1}^{D} \tau_{kd}$, pois a matriz E é constante. E para a distribuição de u_i a quantidade do produtório também não depende de u_i. Logo,

$$\begin{aligned} p(u_i|\ldots) &\propto u_i^{(\eta/2-1)} e^{-u_i\eta/2}\, u_i^{D/2} \\ &\times \exp\Big\{-\frac{u_i}{2}(\mathbf{y}_i-\boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_k E')^{-1}(\mathbf{y}_i-\boldsymbol{\mu}_k)\Big\}. \end{aligned}$$

Logo, a distribuição condicional completa de u_i , $i = 1, \ldots, n$, é dada por

$$p(u_i|\ldots) \sim Gama\left(\frac{\eta+D}{2}, \frac{\eta+S_i}{2}\right), \tag{A.8}$$

em que $S_i = (\mathbf{y}_i-\boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_k E')^{-1}(\mathbf{y}_i-\boldsymbol{\mu}_k)$, para $k : z_i = k$.

A.2.0.5 Condicional Completa de $\boldsymbol{\tau}$

Os autovalores da matriz de escala possuem distribuição Gama a priori que é de família conjugada à distribuição Normal da verossimilhança.

$$\begin{aligned} \boldsymbol{\tau}_k = (\tau_{k1}, \ldots, \tau_{kD}) \text{ com } \tau_{kd}^{-1} &\sim Gama(a_0/2, b_0/2), \\ \boldsymbol{\Sigma}_k = E\boldsymbol{\mathcal{T}}_k E', k = 1, \ldots, K \text{ e } d &= 1, \ldots, D. \end{aligned}$$

A distribuição condicional completa de τ_{kd} é obtida de

$$\begin{aligned} p(\tau_{kd}^{-1}|\ldots) &\propto p(\tau_{kd}^{-1}) L(\mathbf{z}, \boldsymbol{\mu}, \mathbf{u}, \boldsymbol{\Sigma}, K, \ldots) \\ &\propto \tau_{kd}^{-(a_0/2-1)} e^{-b_0/2\tau_{kd}} \prod_{i:z_i=k} \frac{\exp\Big\{-\frac{u_i}{2}(\mathbf{y}_i-\boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_k E')^{-1}(\mathbf{y}_i-\boldsymbol{\mu}_k)\Big\}}{\left|u_i^{-1} E\boldsymbol{\mathcal{T}}_k E'\right|^{1/2}} \\ &\propto \tau_{kd}^{-(a_0/2-1)} e^{-b_0/2\tau_{kd}} \Big(\prod_{i:z_i=k} \left|u_i^{-1} E\boldsymbol{\mathcal{T}}_k E'\right|^{1/2} \Big) \\ &\times \exp\Big\{-\frac{1}{2}\sum_{i:z_i=k} u_i(\mathbf{y}_i-\boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_k E')^{-1}(\mathbf{y}_i-\boldsymbol{\mu}_k)\Big\} \\ &\propto \tau_{kd}^{-(a_0/2-1)} \Big(\prod_{i:z_i=k} \left|u_i^{-1} E\boldsymbol{\mathcal{T}}_k E'\right|^{-1/2} \Big) \\ &\times \exp\Big\{-\frac{1}{2}\sum_{i:z_i=k} u_i(\mathbf{y}_i-\boldsymbol{\mu}_k)'(E\boldsymbol{\mathcal{T}}_k E')^{-1}(\mathbf{y}_i-\boldsymbol{\mu}_k) - b_0/2\tau_{kd}\Big\}. \end{aligned}$$

No determinante que permanece no produtório, u_i^{-1} é uma constante multiplicando a matriz quadrada de dimensão D, logo

$$E\left|u_i^{-1}E\boldsymbol{\mathcal{T}}_kE'\right| = u_i^{-D}\left|E\boldsymbol{\mathcal{T}}_kE'\right|.$$

Além disso, a matriz E é constante em relação a τ_{kd} e a matriz $\boldsymbol{\mathcal{T}}_k$ é diagonal com determinante sendo o produto dos elementos da diagonal. Assim,

$$\left|u_i^{-1}E\boldsymbol{\mathcal{T}}_kE'\right| \propto u_i^{-D}\prod_{d=1}^{D}\tau_{kd} = u_i^{-D}\tau_{kd}.$$

Logo,

$$\prod_{i:z_i=k}\left|u_i^{-1}E\boldsymbol{\mathcal{T}}_kE'\right|^{-1/2} = \prod_{i:z_i=k}u_i^{D/2}\tau_{kd}^{-1/2}.$$

Como u_i^{-D} constante em relação a τ_{kd},

$$\prod_{i:z_i=k}u_i^{D/2}\tau_{kd}^{-1/2} \propto \tau_{kd}^{-n_k/2}.$$

Na inversa do produto de matrizes dentro da exponencial a matriz E é ortogonal, ou seja, $E^{-1} = E'$. E por propriedade das inversas, temos $(E\boldsymbol{\mathcal{T}}_kE')^{-1} = (E')^{-1}\boldsymbol{\mathcal{T}}_k^{-1}(E)^{-1} = E\boldsymbol{\mathcal{T}}_k^{-1}E' = E\ diag\{1/\tau_{k1},\ldots,1/\tau_{kD}\}E'$.

$$\begin{aligned} p(\tau_{kd}^{-1}|\ldots) &\propto \tau_{kd}^{-(a_0/2-1)}\tau_{kd}^{-n_k/2}\exp\{-(b_0/2)\tau_{kd}^{-1}\} \\ &\times \exp\Big\{-\frac{1}{2}\sum_{i:z_i=k}u_i(E'(\mathbf{y}_i-\boldsymbol{\mu}_k))'diag\Big\{\tau_{k1}^{-1},\ldots,\tau_{kD}^{-1}\Big\}((\mathbf{y}_i-\boldsymbol{\mu}_k)'E)'\Big\}. \end{aligned}$$

Na soma dentro da exponencial, para um dado indivíduo $i : z_i = k$, temos $u_i(E'(\mathbf{y}_i-\boldsymbol{\mu}_k))'$ $diag\Big\{\tau_{k1}^{-1},\ldots,\tau_{kD}^{-1}\Big\}((\mathbf{y}_i-\boldsymbol{\mu}_k)'E)' = u_i[e_1'(\mathbf{y}_i-\boldsymbol{\mu}_k)\tau_{k1}^{-1}(\mathbf{y}_i-\boldsymbol{\mu}_k)'e_1 + \ldots + e_D'(\mathbf{y}_i-\boldsymbol{\mu}_k)\tau_{kD}^{-1}(\mathbf{y}_i-\boldsymbol{\mu}_k)'e_D]$, em que e_d é a d-ésima coluna da matriz E. Assim, apenas as parcelas relacionadas a dimensão d da matriz E não são constantes em relação a τ_{kd}.

$$\begin{aligned} p(\tau_{kd}^{-1}|\ldots) &\propto \tau_{kd}^{-(a_0/2-1)}\tau_{kd}^{-n_k/2}\exp\{-(b_0/2)\tau_{kd}^{-1}\} \\ &\times \exp\Big\{-\frac{e_d'\sum_{i:z_i=k}u_i(\mathbf{y}_i-\boldsymbol{\mu}_k)'(\mathbf{y}_i-\boldsymbol{\mu}_k)e_d}{2}\tau_{kd}^{-1}\Big\}. \end{aligned}$$

Logo,

$$\tau_{kd}^{-1}|\ldots \sim Gama(\frac{a_0+n_k}{2}, \frac{b_0+e_d'S_ke_d}{2}), \text{ em que } S_k = \sum_{i:z_i=k}u_i(\mathbf{y}_i-\boldsymbol{\mu}_k)(\mathbf{y}_i-\boldsymbol{\mu}_k)'. \tag{A.9}$$

A.2.0.6 Distribuição Condiconal Completa para μ_k

Para atualizar μ_k, será utilizado o algoritmo de Metropolis-Hastings com a distribuição condicional completa dada abaixo

$$p(\boldsymbol{\mu}_k|\ldots) \propto \det(C_{\boldsymbol{\mu}}) \prod_{i:z_i=k} N(y_i, \boldsymbol{\mu}_k, u_k^{-1}\boldsymbol{\Sigma}_k).$$

Para o cálculo da distribuição de probabilidade condicional de $\boldsymbol{\mu}$ foi utilizada a identidade de Schur [100] que permite decompor o determinante da matriz $C_{\boldsymbol{\mu}}$ e facilitando obtenção da condicional da componente $\boldsymbol{\mu}_k$ do vetor de médias, dado os valores de todos os outros parâmetros, inclusive as componentes restantes do vetor de médias.

$$p(\boldsymbol{\mu}_k|\ldots) \propto \left(C_{\boldsymbol{\mu}_k} - C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})C_{\boldsymbol{\mu}_{-k}}^{-1}C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_{-k})'\right) \prod_{i:z_i=k} N(\boldsymbol{\mu}_k, u_k^{-1}\boldsymbol{\Sigma}_k), \tag{A.10}$$

em que $C_{\boldsymbol{\mu}_k} = C(\boldsymbol{\mu}_k, \boldsymbol{\mu}_k)$ e $C_{\boldsymbol{\mu}_{-k}} = C(\boldsymbol{\mu}_{-k}, \boldsymbol{\mu}_{-k})$, com $\boldsymbol{\mu}_{-k} = \{\boldsymbol{\mu}_j\}_{j\neq k}$.

A.2.0.7 Distribuição Condicional Completa para θ^2 e σ_q^2

Devido a restrição da Seção A.1.4, os parâmetros do kernel são considerados dependentes a priori e são modelados a partir de distribuições Gama truncadas. O truncamento é obtido a partir da restrição.

$$\theta^2|\sigma_q^2 \sim Gama_T(a_1, b_1, t_1), \theta^2 \in t_1 = \left(0, \frac{(2\pi\sigma^2)^{D/2}}{\sqrt{\pi}}\right)$$

e

$$\sigma_q^2|\theta^2 \sim Gama_T(a_2, b_2, t_2), \sigma_q^2 \in t_2 = \left(\frac{(\theta\sqrt{\pi})^{1/D}}{\sqrt{2\pi}}, \infty\right).$$

Para as distribuições condicionais completas dos parâmetros são utilizadas as distribuições a priori condicionais

$$p(\theta^2|\ldots) \propto \frac{\det(C_{\mu,v,\theta,\sigma_q})}{\prod_{h=1}^{\infty}(\lambda_h(\theta,\sigma_q)+1)} p(\theta^2|\sigma_q), \tag{A.11}$$

em que $\theta^2|\sigma_q^2 \sim Gama_T(a_1, b_1, t_1), \theta^2 \in t_1 = \left(\frac{(2\pi\sigma^2)^{D/2}}{\sqrt{\pi}}; \infty\right)$.

$$p(\sigma_q|\ldots) \propto \frac{\det(C_{\boldsymbol{\mu},v,\theta,\sigma_q})}{\prod_{h=1}^{\infty}(\lambda_h(\theta,\sigma_q)+1)} p(\sigma_q|\theta), \tag{A.12}$$

em que

$$\sigma_q^2 \sim GamaTruncEsq(a_2, b_2, t_2), \sigma_q \in t_2 = \left(0; \frac{(\theta\sqrt{\pi})^{1/D}}{\sqrt{2\pi}}\right).$$

A.3 Modelo NIDPP para Variável Categórica

A.3.0.1 Função de Verossimilhança

$$\begin{aligned} L(\boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\alpha}, \mathbf{u}, \sigma_y^2, \mathbf{w}, \theta^2, \sigma_q^2 | \mathbf{y}) &= (2\pi)^{-N/2} \det(\mathbf{U}^{-1}\sigma_y^2)^{-1/2} \\ &\times \exp\Big\{ -\frac{1}{2\sigma_y^2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})\Big\} \end{aligned}$$

A.3.0.2 Distribuição Condiconal Completa para z

O agrupamento é relacionada a variável categórica, logo, a partição se dá sobre as J categorias e vetor $\mathbf{z} = (z_1, \ldots, z_J)$ é responsável pela alocação. As componentes são consideradas independentes e sua distribuição é obtida diretamente pela normalização da distribuição proporcional a seguir

$$\begin{aligned} P(z_j = k | \ldots) &\propto w_K (2\pi)^{-N/2} \det(\mathbf{U}_j^{-1}\sigma_y^2)^{-1/2} \\ &\times \exp\Big\{ -\frac{1}{2\sigma_y^2}(\mathbf{y}_j - \mathbf{X}_j\boldsymbol{\beta} - \mathbf{Z}_j\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})\Big\} \end{aligned}$$

A constante de normalização é obtida pela soma das quantidades acima para todo $k = 1, \ldots, K$ para todas as categorias.

A.3.0.3 Distribuição Condiconal Completa para w

Para os pesos $\mathbf{w}$ a distribuição Dirichlet é de família conjugada à distribuição Normal considerada na função de verossimilhança, assim a sua distribuição posteriori é dada por

$$\begin{aligned} p(\mathbf{w}|\ldots) &= p(\mathbf{w}|\mathbf{z})p(\mathbf{z}|\mathbf{w}) \\ &= \frac{\Gamma(\sum_{k=1}^K \delta_k)}{\prod_{k=1}^K \Gamma(\delta_k)} \prod_{k=1}^K w_k^{\delta_k - 1} P(z_j = k)^{\sum_{k=1}^k \mathbf{I}_{\{z_j = k\}}} \\ &\propto \prod_{k=1}^K w_k^{\delta_k - 1 + \sum_{k=1}^k \mathbf{I}_{\{z_j = k\}}}. \end{aligned}$$

Logo, a condicional completa dos pesos será $\mathbf{w}|\cdots \sim Dir(\delta_1 + n_1, \ldots, \delta_k + n_K)$, em que $n_k = \sum_{i=1}^N I_{(z_1 = k)}$.

A.3.0.4 Distribuição Condiconal Completa para $\boldsymbol{\beta}$

Os efeitos das covariáveis comuns a todos os indivíduos representados pelo vetor $\boldsymbol{\beta}$,

$$\begin{aligned} p(\boldsymbol{\beta}|...) &\propto \exp\Big\{-\frac{1}{2\sigma_y^2}(\boldsymbol{\beta}-\boldsymbol{\mu}_\beta)'\boldsymbol{\Sigma}_\beta{}^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}_\beta)\Big\} \\ &\times \exp\Big\{-\frac{1}{2\sigma_y^2}(\mathbf{y}-\mathbf{X}\boldsymbol{\beta}-\mathbf{Z}\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y}-\mathbf{X}\boldsymbol{\beta}-\mathbf{Z}\boldsymbol{\alpha})\Big\} \\ &\propto \exp\Big\{-\frac{1}{2\sigma_y^2}\Big[\boldsymbol{\beta}'(\Sigma_\beta^{-1}+\mathbf{X}'\mathbf{U}\mathbf{X})\boldsymbol{\beta}-2\boldsymbol{\beta}'\Big(\Sigma_\beta^{-1}\boldsymbol{\mu}_\beta+\mathbf{X}'\mathbf{U}\mathbf{y}-\mathbf{X}'\mathbf{U}\boldsymbol{\alpha}\Big)\Big]\Big\} \end{aligned}$$

Assim, possuem distribuiçào condicional completa é conjugada à família NI e é fornecida a seguir:

$$\boldsymbol{\beta}|\sigma_y^2,\ldots \sim N_p(\mathbf{M}_\beta,\mathbf{S}_\beta),$$

em que a média é $\mathbf{M}_\beta = \Big(\boldsymbol{\Sigma}_\beta^{-1}+\mathbf{X}'\mathbf{U}^{-1}\mathbf{X}\Big)^{-1}\Big(\boldsymbol{\Sigma}_\beta^{-1}\boldsymbol{\mu}_\beta+\mathbf{X}'\mathbf{U}^{-1}\dagger-\mathbf{X}'\mathbf{U}^{-1}\mathbf{Z}\boldsymbol{\alpha}\Big)^{-1}$ e sua variância é $\mathbf{S}_\beta = \sigma_y^2\Big(\boldsymbol{\Sigma}_\beta^{-1}+\mathbf{X}'\mathbf{U}^{-1}\mathbf{X}\Big)^{-1}$.

A.3.0.5 Distribuição Condicional Completa para u

As componentes do vetor da variável misturadora são considerados independentes e identicamente distribuidos a priori. A distribuição condicional completa de u_{ij} , $i = 1,\ldots,n_j, j = 1,\ldots,J$, é dada por

$$\begin{aligned} p(u_{ij}|\ldots) &= p(u_{ij})L(\boldsymbol{\beta},\mathbf{z},\boldsymbol{\alpha},\mathbf{u},\sigma_y^2,\mathbf{w},\theta^2,\sigma_q^2|\mathbf{y}) \\ &\propto \exp(-u_{ij}\eta/2)u_{ij}^{\eta/2-1}\det(\mathbf{U}_j^{-1}\sigma_y^2)^{-1/2} \\ &\times \exp\Big\{-\frac{1}{2\sigma_y^2}(\mathbf{y}_j-\mathbf{X}_j\boldsymbol{\beta}-\mathbf{Z}_j\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y}-\mathbf{X}\boldsymbol{\beta}-\mathbf{Z}\boldsymbol{\alpha})\Big\} \\ &\propto exp(-u_{ij}\eta/2)u_{ij}^{\eta/2-1}u_{ij}^{1/2-1}, \\ &\times \exp\Big\{-\frac{1}{2\sigma_y^2}(y_{ij}-\mathbf{X}_{ij}\boldsymbol{\beta}-\alpha_k)^2\Big\} \end{aligned}$$

Logo, $u_{ij}|\ldots \sim Gama\Big(\frac{\eta+1}{2},\frac{\eta+S_{ij}}{2}\Big)$, em que $S_{ij} = \dfrac{(y_{ij}-\mathbf{X}_{ij}\boldsymbol{\beta}-\alpha_k)^2}{\sigma_y^2}$, com $\mathbf{X}_{ij}$ a linha da matriz de covariáveis relacionadas a i-ésima observação da j-ésima categoria.

A.3.0.6 Distribuição Condiconal Completa para σ_y^2

O parâmetro de escala, σ_y^2, possui distribuição Inversa-Gama a priori, ou ainda, $\frac{1}{\sigma_y^2} \sim Gama(\frac{a_0}{2}, \frac{b_0}{2})$ e sua condicional completa é dada por

$$\begin{aligned}
p(\sigma_y^2|\ldots) &= p(\sigma_y^2)p(\boldsymbol{\beta}|\sigma_y^2)L(\boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\alpha}, \mathbf{u}, \sigma_y^2, \mathbf{w}, \theta^2, \sigma_q^2|\mathbf{y}) \\
&\propto \exp(-b_0/\sigma_y^2)(\sigma_y^2)^{-(a_0-1)} \det(\mathbf{U}^{-1}\sigma_y^2)^{-1/2} \det({\Sigma_{\boldsymbol{\beta}}}^{-1}\sigma_y^2)^{-1/2} \\
&\times \exp\Big\{-\frac{1}{2\sigma_y^2}(\boldsymbol{\beta}-\boldsymbol{\mu}_{\boldsymbol{\beta}})'{\boldsymbol{\Sigma}_{\boldsymbol{\beta}}}^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}_{\boldsymbol{\beta}})\Big\} \\
&\times \exp\Big\{-\frac{1}{2\sigma_y^2}(\mathbf{y}_j - \mathbf{X}_j\boldsymbol{\beta} - \mathbf{Z}_j\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})\Big\} \\
&\propto exp(-b_0/\sigma_y^2)(\sigma_y^2)^{-N/2-p/2-(a_0-1)} \\
&\times \exp\Big\{-\frac{1}{2\sigma_y^2}\Big[(\mathbf{y}_j - \mathbf{X}_j\boldsymbol{\beta} - \mathbf{Z}_j\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha}) + (\boldsymbol{\beta}-\boldsymbol{\mu}_{\boldsymbol{\beta}})'{\boldsymbol{\Sigma}_{\boldsymbol{\beta}}}^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}_{\boldsymbol{\beta}})\Big]\Big\}
\end{aligned}$$

$$\frac{1}{\sigma_y^2}\Big|\ldots \sim Gama\Big(a_0 + (N+p)/2, b_0 + (\mathcal{S}+B)/2\Big),$$

em que N é o tamanho total da amostra, p é o tamanho do vetor $\boldsymbol{\beta}$, $\mathcal{S} = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})$ e $\mathbf{B} = \Big(\boldsymbol{\beta} - \boldsymbol{\mu}_{\boldsymbol{\beta}}\Big)'\boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{-1}\Big(\boldsymbol{\beta} - \boldsymbol{\mu}_{\boldsymbol{\beta}}\Big)$.

Para o parâmetro de efeito do agrupamento da variável categórica, temos

$$\begin{aligned}
p(\boldsymbol{\alpha}|\ldots) &\propto \det(C_{\theta,\sigma_q}(\boldsymbol{\alpha}))L(\boldsymbol{\beta}, \mathbf{z}, \boldsymbol{\alpha}, \mathbf{u}, \sigma_y^2, \mathbf{w}, \theta^2, \sigma_q^2|\mathbf{y}) \\
&\propto \det(C_{\theta,\sigma_q}(\boldsymbol{\alpha}))(2\pi)^{-N/2} \det(\mathbf{U}^{-1}\sigma_y^2)^{-1/2} \qquad \text{(A.13)} \\
&\times \exp\Big\{-\frac{1}{2\sigma_y^2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})'\mathbf{U}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\alpha})\Big\}.
\end{aligned}$$

Utilizando o núcleo da distribuição a priori de α_k dado em (4.14), é possível obter a distribuição condicional completa de α_k dado o restante do vetor de efeitos, $\boldsymbol{\alpha_{-k}}$, e os demais parâmetros. Assim, para um k qualquer, $k = 1, 2, \ldots, K$,

$$\begin{aligned}
p(\alpha_k|\boldsymbol{\alpha}_{-k}, \ldots) &\propto \Big(C_{\alpha_k} - C(\alpha_k, \boldsymbol{\alpha}_{-k})C_{\boldsymbol{\alpha}_{-k}}^{-1}C(\alpha_k, \boldsymbol{\alpha}_{-k})'\Big) \qquad \text{(A.14)} \\
&\times \exp\Big\{-\frac{1}{2}\Big(\boldsymbol{\alpha}_k - M_k\Big)'V_k^{-1}\Big(\boldsymbol{\alpha}_k - M_k\Big)\Big\} \qquad \text{(A.15)}
\end{aligned}$$

obtido utilizando-se a Identidade de Schur na decomposição do determinante em (4.14), em que $M_k = [\sum_{j:z_j=k} u_{ij}]^{-1} \sum_{j:z_j=k} (\mathbf{y}_j - \mathbf{X}_j\boldsymbol{\beta})\mathbf{U}_j$, $V_k = \sigma_y^2[\sum_{j:z_j=k} u_{ij}]^{-1}$ e $\mathbf{U}_j = diag(u_{1j}, \ldots, u_{n_{n_j}j})$, com a verossimilhança sendo proposrcional ao núcleo de um distribuição normal unidimensional com média M_k e matriz de variância V_k, ou seja, $N(\boldsymbol{\alpha}_k, M_k, V_k)$.

Para os parâmetros do kernel as distribuições condicionais completas são dadas por

$$p(\theta^2|\ldots) \propto \frac{\det(C_{\boldsymbol{\alpha},\theta^2,\sigma_q^2})}{\prod_{h=1}^{\infty}(\lambda_h(\theta^2,\sigma_q^2)+1)} p(\theta^2|\sigma_q^2), \tag{A.16}$$

em que $\theta^2|\sigma_q^2 \sim Gama_T(a_1,b_1,t_1), \theta^2 \in t_1 = \left(0, \frac{(2\pi\sigma^2)^{D/2}}{\sqrt{\pi}}\right)$ e

$$p(\sigma_q^2|\ldots) \propto \frac{\det(C_{\boldsymbol{\alpha},\theta^2,\sigma_q^2})}{\prod_{h=1}^{\infty}(\lambda_h(\theta^2,\sigma_q^2)+1)} p(\sigma_q^2|\theta^2), \tag{A.17}$$

em que $\sigma_q^2 \sim Gama_T(a_2,b_2,t_2), \sigma_q^2 \in t_2 = \left(\frac{(\theta\sqrt{\pi})^{1/D}}{\sqrt{2\pi}}, \infty\right)$. Além disso, conjuntamente

$$\begin{aligned} p(\theta^2,\sigma_q^2|\ldots) &= p(\theta^2|\sigma_q^2,\ldots)p(\sigma_q^2|\ldots) \\ &\propto \frac{\det(C_{\boldsymbol{\alpha},\theta^2,\sigma_q^2})}{\prod_{h=1}^{\infty}(\lambda_h(\theta^2,\sigma_q^2)+1)} p(\theta^2|\sigma_q^2)p(\sigma_q^2). \end{aligned} \tag{A.18}$$

Apêndice B

Aspectos Computacionais

B.1 Rotinas Implementadas para o Ajuste dos Modelos para Estimação de Densidade

```
# Runing fitting
mcmc <- dpp_uvar_uni(X, hparams, store, control, fixed, initial =
list(nu = 5), verbose)
# X is data

# -------- Fit NIDPP model function ####
dpp_uvar_uni <- function(X, hparams=NULL, store=NULL, control=NULL,
fixed=NULL, initial=NULL, verbose=TRUE){

stime <- proc.time();

# nu paramether
if(is.null(initial$nu)){
nu <- 2.1;
} else{nu <- initial$nu;}

# if(is.null(initial$mu_y)){
#   mu_y <- 0;
# } else{mu_y <- initial$mu_y;}

# Data dimension
D = ncol(X);

# Hiperparameters
hparams.default <- list(
a0 = 0.1,     b0 = 0.1, # var_y
delta = 1.0, # or 0.01, w
```

```
a1 = 100,   b1 = 1.0, # theta
a2 = 200,   b2 = 0.5, #sigma_q
a3 = nu/2,   b3 = nu/2,
mu.prop = list(si = 0, th = 5, mu = 0),
v.prop = list(si = 500, th = 0.5, mu = sqrt(1))
);


if(!is.null(hparams)){
hparams <- c(hparams, hparams.default);
}else{hparams <- hparams.default}
#cat("\n a1, b1, a2, b2: ", hparams$a1, hparams$b1,
hparams$a2, hparams$b2);

control.default <- list(
niter = 40000,
burnin = 0,
thin = 1
);

if(!is.null(control)){
control <- c(control, control.default);
} else{control <- control.default}


E <- compute_E_uvar(X);
# svd(sigma2)$v;

# ---- Incicializando a cadeia ----
# Parâmetros do DPP na mesma função
smpl <- initialize_uvar(X, E, fixed, initial);

# Adaptative mcmc mean to sigmaq
mcsigq2 <- c(smpl$sigmaq^2, rep(NA, control$niter-1));

# acceptance counting (mu, theta and sigma_q)
rhos <- list(rhots=0, rhomu = rep(0,smpl$K));

# Calculando a matrix de covariância pela decomposição
if(!is.null(initial$Sigma)){
Sigma <- initial$Sigma;
```

```
smpl$Sigma <- Sigma;
} else{
Sigma <- update_Sigma_mod(smpl$lambda, smpl$K, E);
smpl$Sigma <- Sigma;
}

smpl$Sigma <- Sigma;


# Armazenamento de valores das posterioris
if (is.null(store)) {
store <- c(names(smpl));
} else if (length(store) == 1 && is.na(store)) {
store <- c("z", "K");
}

# Número de iterações e aquecimento da cadeia
if (control$burnin <= 0) {
mcmc <- list(smpl);
} else {
mcmc <- list();
}

#### ---- Loop ---- ####

i <- 1; # first iteration was inicialize to mcmc
mcmc[[i]] <- smpl[store]

for (iter in 2:control$niter) { # iter = iter + 1

if (is.null(fixed$z)) {
smpl$z <- update_z_uvar(smpl, X, E);
# cat("atualização de z \n"); # print(smpl$z);
}

if (is.null(fixed$w)) {
smpl$w <- update_w_uvar(smpl$z, hparams, smpl$K);
# cat("atualização de w \n");
}

if (is.null(fixed$lambda)) {
```

```
smpl$lambda <- update_lambda_uvar(smpl, X, E, hparams);
# cat("atualização de lambda \n"); # print(smpl$lambda);
}

if (is.null(fixed$Sigma)) {
smpl$Sigma <- update_Sigma_mod(smpl$lambda, smpl$K, E);
# cat("atualização de Sigma \n"); # print(smpl$Sigma);
} # apenas o cálculo de Sigma, os lambdas com u foram calculados antes

# ---- u loop ----
if (is.null(fixed$u)) {
smpl$u <- update_u_uvar(smpl, X, hparams);
# cat("atualização de u \n", smpl$u);
}

if (is.null(fixed$mu)) {
stemp_mu <- update_mu_uvar(smpl, X, hparams);
smpl$mu <- stemp_mu$mu;
rhos$rhomu <- rhos$rhomu + stemp_mu$rhomu;
# cat("atualização de mu \n", smpl$mu, "\n");
}



# Adaptative mcmc
if(iter<51){
mesigq2 <- mean(mcsigq2, na.rm =TRUE);
}else{mesigq2 <- mean(mcsigq2[(iter-50):(iter-1)])}

# ---- kernel loop----
if(is.null(fixed$theta)|is.null(fixed$sigmaq)){
h_up <- update_kernel(smpl$theta^2, smpl$sigmaq^2, mesigq2,
smpl$mu, smpl$z, hparams, fixed)
rhos$rhots <- rhos$rhots + h_up$rhots;
smpl$theta <- sqrt(h_up$t2);
smpl$sigmaq <- sqrt(h_up$sq);
mcsigq2[iter] <- smpl$sigmaq^2;
}

if (is.null(fixed$K)) {
smpl <- update_K_uvar(smpl, X, E, hparams);
# all other parameters will change if K changes
```

```
# cat("atualização de K, Sigma: \n", smpl$Sigma, "\n");
}

if (iter > control$burnin && iter %% control$thin == 0) {
i <- i + 1;
mcmc[[i]] <- smpl[store];
}

if (verbose && iter %% floor(control$niter/100) == 0) {
message("iter: ", iter, ";  K = ", smpl$K)
}
}

if (verbose) {
message("Elapsed time: ", proc.time()[3] - stime[3])
}

return(list(mcmc=mcmc, rhos = rhos))

}




#
compute_E_uvar <- function(X) {
X.c <- scale(X, center=TRUE, scale=FALSE);
svd(X.c)$v
}

#
initialize_uvar <- function(X, E, fixed, initial) {
N <- nrow(X);
D <- ncol(X);

if (is.null(fixed$K)) {
if (!is.null(fixed$mu)) {
if(!is.null(nrow(fixed$mu))&!is.null(fixed$mu)){
fixed$K <- nrow(fixed$mu)
} else {fixed$K <- length(fixed$mu)};
} else if (!is.null(fixed$w)) {
```

```
fixed$K <- length(fixed$w);
} else if (!is.null(fixed$lambda)) {
if(!is.null(now(fixed$lambda))){
fixed$K <- nrow(fixed$lambda);
} else {
fixed$K <- length(fixed$lambda);
}
} else if (!is.null(fixed$z)) {
fixed$K <- max(fixed$z);
}
}

if (is.null(fixed$K)) {
if(is.null(initial$K)){
K <- ceiling(log(nrow(X)));
} else{K <- initial$K}
} else {K <- fixed$K};

if(D == 1 &!is.null(fixed$mu)){
fixed$mu <- matrix(fixed$mu, nrow = K)
}

if (!all(c("z", "mu", "w") %in% names(fixed)) ) {
cl <- kmeans(X, K);
}

if (is.null(fixed$z)) {
if(is.null(initial$z)){
z <- cl$cluster;
} else {z <- initial$z}
} else {z <- fixed$z};

if (is.null(fixed$w)) {
if(is.null(initial$w)){
w <- cl$size / sum(cl$size);
} else{w <- initial$w}
} else {w <- fixed$w};

if (is.null(fixed$mu)) {
if(is.null(initial$mu)){
mu <- matrix(rnorm(K);
```

```
} else{mu <- initial$mu}
} else {mu <- fixed$mu};

if (is.null(fixed$lambda)) {
if(is.null(initial$lambda)){
lambda <- matrix(1.0, K, D);
} else{lambda <- initial$lambda;}
} else {lambda <- fixed$lambda;}

if (is.null(fixed$u)){
if(is.null(initial$u)){
u <- rep(1.0, N);
} else{u <- initial$u}
} else {u <- fixed$u};

if (is.null(fixed$sigmaq)) {
if(is.null(initial$sigmaq)){
sigmaq <- N/(max(Y)-min(Y));
} else{sigmaq <- initial$sigmaq}
} else {sigmaq <- fixed$sigmaq};

if (is.null(fixed$theta)) {
if(is.null(initial$theta)){
theta <- quantile(dist(X), 0.5);
} else{theta <- initial$theta}
} else {theta <- fixed$theta};

list(
K = K,    z = z,    mu = mu,    w = w,    lambda = lambda,
u = u,    sigmaq = sigmaq,    theta = theta
)
}

# -------

update_z_uvar <- function(smpl, X, E) {
K <- smpl$K; mu <- smpl$mu; w <- smpl$w; lambda <- smpl$lambda;
N <- nrow(X); u <- smpl$u; Sigma <- smpl$Sigma;
kz <- 1:K;

# compute unnormlized density
```

```
dz <- matrix(
unlist(lapply(kz,
function(k) {
log(w[k]) + mvtnorm::dmvnorm((X - mu[k, ])*u^(1/2), 0,
as.matrix(Sigma[ , , k]),log =T)*u^(1/2)
}
)),
nrow = nrow(X)
);

M <- log(.Machine$double.xmax)/ncol(dz) - apply(dz,1,max);
dz <- exp(M + dz);
dzsum <- apply(dz,1,sum)
dz <- dz/dzsum

z <- apply(dz, 1,
function(d) {
# ‘sample‘ will normalize d to sum to 1
sample(kz, 1, prob=d)
}
);

z
}

# -------
update_Sigma_mod <- function(lambda, K, E) {
D <- nrow(E);
Sigma <- array(0, c(D, D, K));

for (k in 1:K) {
Sigma[, , k] <- E %*% diag(lambda[k, ],D) %*% t(E);
}
Sigma
}

# --------
update_w_uvar <- function(z, hparams, K){
# z must be have components 1 ... K
m <- rep(0, K);
idz <- as.numeric(names(table(z)));
```

```
# s.t. table(z) has size K
m[idz] <- as.numeric(table(z));

rdirichlet(1, hparams$delta + m);
}

#-----

compute_Sk_uvar <- function(X, mu, idx, k, u) {
nidx <- length(idx);
d <- t((X[idx, , drop=FALSE] - matrix(mu[k, ],nrow=nidx,byrow = T))*
u[idx]^(1/2));
d %*% t(d)
}

#---------
update_lambda_uvar <- function(smpl, X, E, hparams) {
K <- smpl$K; z <- smpl$z; mu <- smpl$mu;  u <- smpl$u;
D <- ncol(X); hparams <- hparams;

lambda <- matrix(0, K, D);
for (k in 1:K) {
for(d in 1:D){
idx <- which(z == k);
if(length(idx)==0){
lambda[k, d] <- 1/rgamma(1, shape=hparams$a0/2,
rate=hparams$b0/2);
} else{
Sk <- compute_Sk_uvar(X, mu, idx, k, u);

aa <- hparams$a0 + length(idx);
bb <- hparams$b0 + (t(E[,d]) %*% Sk %*% E[,d]);
lambda[k, d] <- 1 / rgamma(1, shape = aa/2, rate = bb/2);
}

}

}

lambda
}
```

```
#-------
update_u_uvar <- function(smpl, X, hparams) {
K <- smpl$K; z <- smpl$z; mu <- smpl$mu; Sigma <- smpl$Sigma;
D <- ncol(X); N <- nrow(X);
u <- rep(0, N);

for (k in 1:K) {
idx <- which(z == k);
nidx <- length(idx);

if(nidx==0){
u[idx] <- rgamma(nidx, shape=hparams$a3/2, rate=hparams$b3/2);
} else {
d <- (X[idx, , drop=FALSE] - matrix(mu[k, ], nrow = nidx,
byrow = T));
dd <- unlist(sapply(d, function(v){v%*%solve(Sigma[,, k])%*%v},
simplify = F));

aa <- hparams$a3 + D;
bb <- hparams$b3 + dd;
u[idx] <- unlist(sapply(bb, function(v){rgamma(1, shape = aa/2,
rate = v/2)}))

}
}

u
}

#------
update_mu_uvar <- function(smpl, X, hparams) {
K = smpl$K; z = smpl$z; mu = smpl$mu; lambda = smpl$lambda;
theta = smpl$theta; sigmaq = smpl$sigmaq; Sigma <- smpl$Sigma;
u <- smpl$u;
D <- ncol(X);
rhomu <- rep(0, K) # counting acceptance

S.pro <- (hparams$v.prop$mu)^2 * diag(D); # parametro do kernel

# covariance matrix for DPP
```

```
C <- cov_matrix(mu, theta^2, sigmaq^2);

if (K == 1) {

mk <- nrow(X);

mu.k.old <- mu[1, , drop=FALSE];
mu.k.new <- mvtnorm::rmvnorm(1, mu.k.old, S.pro);

S.inv <- solve(Sigma[,,K]); # print(S.inv );

D.old <- (X - matrix(mu.k.old, mk, D, byrow=TRUE))*
u[idx]^(1/2);
D.new <- (X - matrix(mu.k.new, mk, D, byrow=TRUE))*
u[idx]^(1/2);

ll.old <- -0.5 * sum(apply(D.old, 1, function(d) t(d) %*%
S.inv
%*% d));
ll.new <- -0.5 * sum(apply(D.new, 1, function(d) t(d) %*%
S.inv %*% d));

Cm <- log(cov_matrix(as.matrix(mu.k.new), theta^2, sigmaq^2))-
log(cov_matrix(as.matrix(mu.k.old), theta^2, sigmaq^2));
ratio <- exp(ll.new - ll.old + Cm);

if (runif(1) < ratio) {
mu[1,] <- mu.k.new;
alpmu[K] <- alpmu[K] + 1
}

} else {

for (k in 1:K) {

C.sinv <- Hmisc::solvet(C[-k,-k]);
mu.k.old <- mu[k, , drop=FALSE];

S.pro <- (hparams$v.prop$mu)^2 * diag(D);
mu.k.new <- matrix(mvtnorm::rmvnorm(1, mu.k.old, S.pro),
nrow = D);
```

```
b.old <- C[k, -k, drop=FALSE];
b.new <- cov_rvec_mat(mu.k.new, mu[-k, , drop=FALSE],
theta^2, sigmaq^2);

Cm.old <- cov_matrix(mu.k.old, theta^2, sigmaq^2);
Cm.new <- cov_matrix(mu.k.new, theta^2, sigmaq^2);

ratio <- (Cm.new - b.new %*% C.sinv %*% t(b.new)) /
(Cm.old - b.old %*% C.sinv %*% t(b.old));

idx <- which(z == k);
mk <- length(idx);

if(mk > 0){ # no empty clusters

S.inv <- Hmisc::solvet(Sigma[,,k]);
D.old <- (X[idx,] - matrix(mu.k.old, mk, D, byrow=TRUE))*
u[idx]^(1/2);
D.new <- (X[idx,] - matrix(mu.k.new, mk, D, byrow=TRUE))*
u[idx]^(1/2);

ll.old <- -0.5 * sum(apply(D.old, 1, function(d) t(d) %*%
S.inv %*% d));
ll.new <- -0.5 * sum(apply(D.new, 1, function(d) t(d) %*%
S.inv %*% d));

ratio.like <- exp(ll.new - ll.old);
ratio <- ratio * ratio.like;

}

# probabilistic step
if (runif(1) < ratio) {
mu[k,] <- mu.k.new;
C <- cov_matrix(mu, theta^2, sigmaq^2);
rhomu[k] <- rhomu[k] + 1
}

} # end for
```

```
} # end else

return(list(mu = mu, rhomu = rhomu))
}


### ---- Update kernel ---- ####################

flts <- function(t2, sq2, D=1){
h <- rowSums(expand.grid(1:500,1:500));
h <- h[order(h)];
a <- 1/(4*sq2);
b <- 1/(t2);
c <- sqrt(a^2+2*a*b); # já modificado por a e b
l <- (2*a/(a+b+c))^(D/2)*cumprod((b/(a+b+c))^(h-1));
return(prod(l[which(l>0)]+1))
}

# Log-Determinant
logdet <- function(A){
2*sum(log(diag(chol(A))))
}


# ------ Jointly update \theta and \sigma_q --------- ###
update_kernel <- function(t2, sq2, mesg2, mu, z, hparams, fixed){
rho.ts <- 0;
mu <- mu[sort(unique(z))];

ts = (t2)/(2);# 2/t2;
sigmaq2.new <- EnvStats::rnormTrunc(1, mean = mesg2, sd =
hparams$v.prop$si, min = ts);
sigmaq.new <- sqrt(sigmaq2.new);

tt = (2*sigmaq2.new);
theta2.new <- EnvStats::rnormTrunc(1, mean = t2, sd =
hparams$v.prop$th, max = tt);
theta.new <- sqrt(theta2.new);

if (is.null(fixed$theta) | is.null(fixed$sigmaq)){
lt.det <- logdet(cov_matrix(mu, theta2.new, sigmaq2.new)) -
```

```
logdet(cov_matrix(mu, t2, sq2));
lt.eng <- - log(flts(theta2.new, sigmaq2.new)) + log(flts(t2,
sq2));
lt.pri <- log(cascsim::dtgamma(theta2.new, shape =
hparams$a1, scale=hparams$b1, max = tt)) +
dgamma(sigmaq2.new, shape=hparams$a2, scale=hparams$b2,
log=T) -  log(cascsim::dtgamma(t2, shape=hparams$a1,
scale = hparams$b1, max = tt)) -  dgamma(sq2,
shape=hparams$a2, scale=hparams$b2, log=T);
lt.prop <- log(EnvStats::dnormTrunc(theta2.new, mean = t2, sd =
hparams$v.prop$si, min =0, max = 2*sigmaq2.new)) +
log(EnvStats::dnormTrunc(sigmaq2.new, mean = mesg2, sd =
hparams$v.prop$si, min=0))-
log(EnvStats::dnormTrunc(t2, mean = theta2.new, sd =
hparams$v.prop$si, min =0, max = 2*sq2)) -
log(EnvStats::dnormTrunc(sq2, mean = mesg2, sd =
hparams$v.prop$si, min=0));
ratio.t <- (sum(c(lt.det, lt.eng, lt.pri, lt.prop)))
} else{ratio.t <- 1; theta2.new <- t2; sigmaq2.new <- sq2}

if (log(runif(1)) < ratio.t){t2 <- theta2.new; sq2 <- sigmaq2.new;
rho.ts <- 1}

return(list(t2 = t2, sq2 = sq2, rhots = rho.ts));
}

#### ---- Covariance Functions to DPP's kernel ---- ####

# Similiarty kernel with the covariance function
similarity_kernel <- function(x, y, t2) {
if(is.matrix(x)){
theta <- rep(t2, nrow(x))}else{t2 <- rep(t2, length(x))}
-(x - y)^2/t2
}


# Quality function for kernel with the covariance function
quality_kernel <- function(x, sq2) {
dnorm(x, mean=0, sd = sqrt(sq2), log = TRUE)
}
```

```
cov_matrix <- function(mu, t2, sq2) {
if(is.matrix(mu)){
KK <- nrow(mu);
} else{KK <- length(mu); mu <- matrix(mu,length(mu),1)}
C <- matrix(1.0, KK, KK)
# matrix is symmetric
for (i in 1:KK) {
for (j in i:KK) {
cij <- exp(sum(similarity_kernel(mu[i,], mu[j,], t2)));
qij <- exp(sum(quality_kernel(mu[i, ], sq2) +
quality_kernel(mu[j, ], sq2)));
C[i,j] <- C[j,i] <- cij*qij;
}
}

C
}

cov_rvec_mat <- function(rvec, mat, t2, sq2) {
if(length(rvec)==1){mat<-matrix(mat,ncol=1)}
# replicate row vector by row-wise to match dimension of mat
rmat <- matrix(rvec, nrow(mat), ncol(mat), byrow=TRUE);
matrix(exp(rowSums(quality_kernel(rmat, sq2)+
similarity_kernel(rmat, mat, t2)+
quality_kernel(mat,sq2))), nrow=1)
}
```

Rotina para extrair as cadeias das posteriores do MCMC.

```
estimate.postV3 <- function(object){
N <- length(object[[1]]$z) # tamanho da amostra
m <- length(object); # iterações
D <- dim(object[[1]]$Sigma)[1]
J <- object[[1]]$K[1] # K máximo

mcmc.K <- rep(NA, m);
mcmc.z <- matrix(0, m, N);
mcmc.mu <-array(NA, dim=c(m, J));
mcmc.Sigma <-array(NA, dim=c(m, J));
mcmc.lambda <- matrix(NA, m, J);
mcmc.w <-matrix(NA, m, J);
```

```
mcmc.theta <- rep(NA, m);
mcmc.sigmaq <- rep(NA, m);
mcmc.u <- matrix(NA, m, N);
mui <- array(NA, dim=c(m, N));
Sigmai <- array(NA, dim = c(m, N));
mcmc.param <- list();

for (i in 1:m) {
mcmc.z[i,] <- object[[i]]$z;
mcmc.K[i] <- length(unique(mcmc.z[i,])); #object[[i]]$K;
mcmc.mu[i,] <- c(object[[i]]$mu);
mcmc.Sigma[i,] <- object[[i]]$Sigma;
mcmc.lambda[i,] <- object[[i]]$lambda;
mcmc.w[i,] <- object[[i]]$w;
mcmc.theta[i] <- object[[i]]$theta;
mcmc.sigmaq[i] <- object[[i]]$sigmaq;
mcmc.u[i,] <- object[[i]]$u;
mui[i, ] <- mcmc.mu[i, mcmc.z[i,]];
Sigmai[i, ] <- mcmc.Sigma[i, mcmc.z[i,]]
}

mcmc.param$K <- mcmc.K;
mcmc.param$z <- mcmc.z;
mcmc.param$mu <- mcmc.mu;
mcmc.param$Sigma <- mcmc.Sigma;
mcmc.param$lambda <- mcmc.lambda;
mcmc.param$w <- mcmc.w;
mcmc.param$theta <- mcmc.theta;
mcmc.param$sigmaq <- mcmc.sigmaq;
mcmc.param$u <- mcmc.u;
mcmc.param$mui <- mui;
mcmc.param$Sigmai <- Sigmai;

return(mcmc.param)
}

# Runing the function
post_mcmc <- estimate.postV3(mcmc)
```

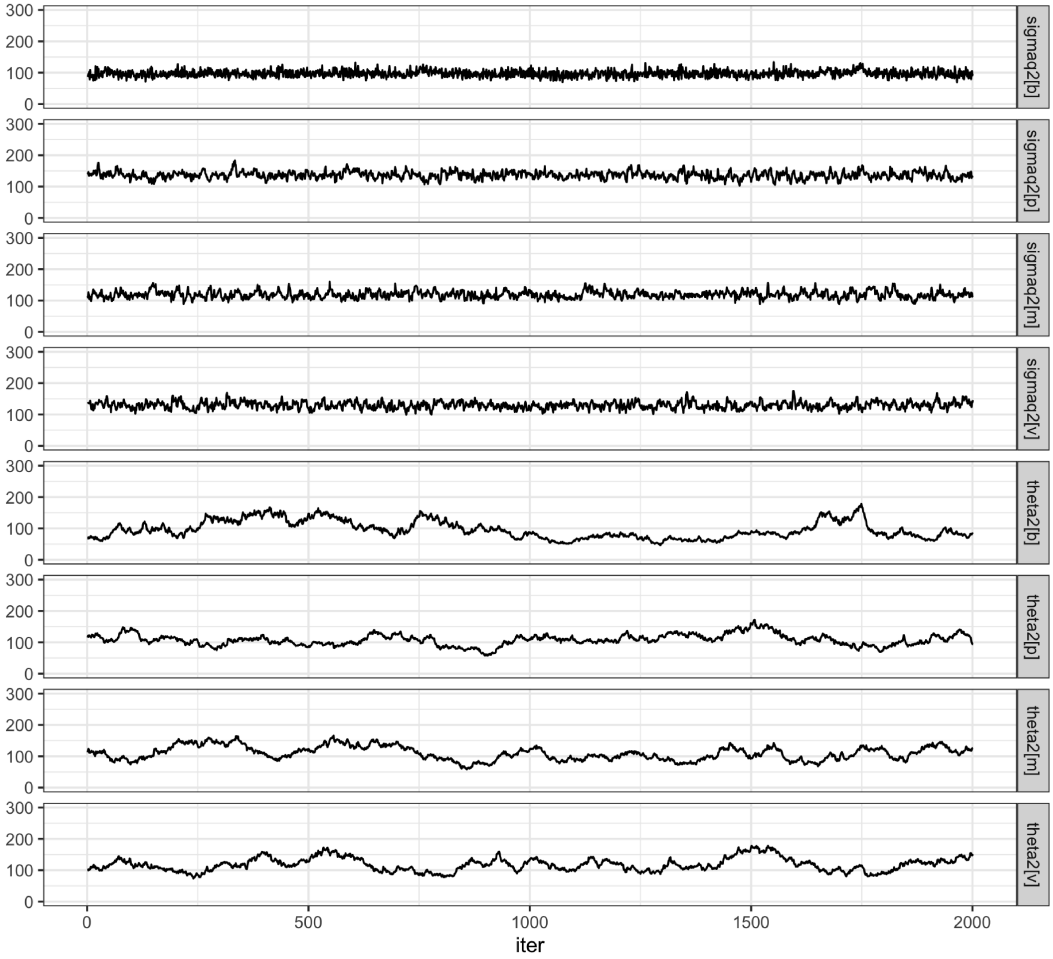
Dados simulados

```
## ---- Base ---- #
```

```
m4 <- c(18, 35, 55); sigma4 <- rep(1.2,3); w4 <- c(.3,.35,.35);
set.seed(83643399)
clust4 = sample(c(1,2,3),replace = T, 300, prob = w4)
set.seed(83643399)
u <- rgamma(300, shape = 2.5, rate = 2.5)
set.seed(83643399)
X4 <- rnorm(300, m4[clust4], sqrt(sigma4/u))
clust4 <- as.factor(clust4)

# ---- Pesos diferentes #
m5 <- c(18, 35, 55); sigma5 = rep(1.2,3); w5 <- c(.5,.3,.2)
set.seed(83643399)
clust5 = sample(c(1,2,3),replace = T, 300, prob = w5)
set.seed(83643399)
u5 <- rgamma(300, shape = 2.5, rate = 2.5)
set.seed(83643399)
X5 <- rnorm(300, m5[clust5], sqrt(sigma5/u5))
clust5 <- as.factor(clust5)


# ---- Duas médias próximas e uma diferente #
m7 <- c(18, 25, 45); sigma7 = rep(1.2,3); w7 <- c(.3,.35,.35)
set.seed(36433998)
clust7 = sample(c(1,2,3),replace = T, 300, prob = w7)
set.seed(36433998)
u7 <- rgamma(300, shape = 2.5, rate = 2.5)
set.seed(36433998)
X7 <- rnorm(300, m7[clust7], sqrt(sigma7/u7))
clust7 <- as.factor(clust7)

# ---- Variâncias muito diferentes #
m8 <- c(18, 35, 55); sigma8=c(5,7,15.5); w8 <- c(.3,.35,.35)
set.seed(83643399)
clust8 = sample(c(1,2,3),replace = T, 300, prob = w8)
set.seed(83643399)
u8 <- rgamma(300, shape = 2.5, rate = 2.5)
set.seed(83643399)
X8 <- rnorm(300, m8[clust8], sqrt(sigma8[clust8]/u8))
clust8 <- factor(clust8)
```

B.1.1 Cadeias *a posteriori*

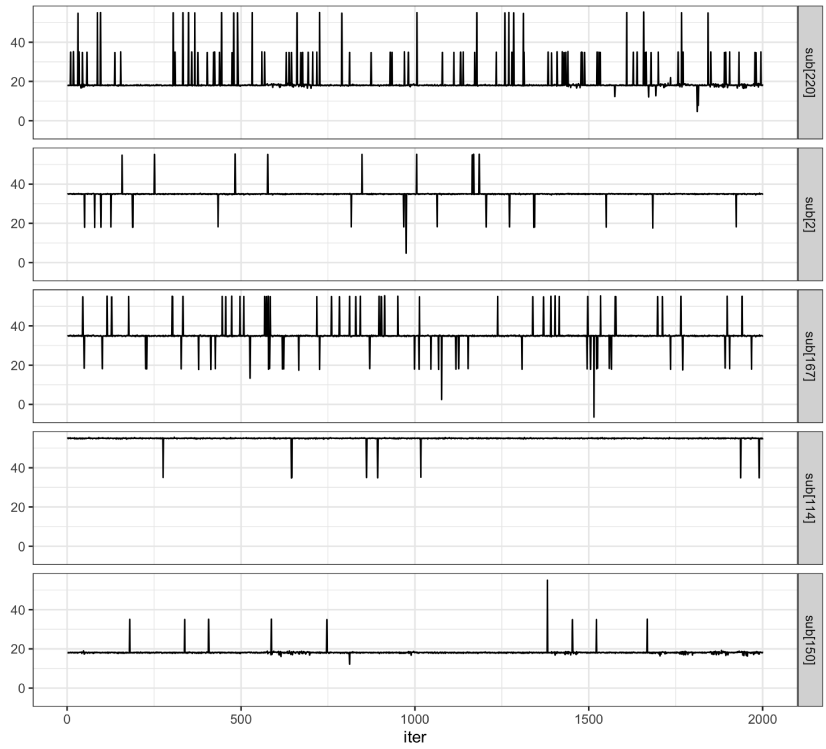
Figura B.1: Cadeias das distribuições a posteriori dos parâmetros do kernel dos modelos ajustados aos Cenários de dados simulados.



Fonte: Elaborado pela autora.

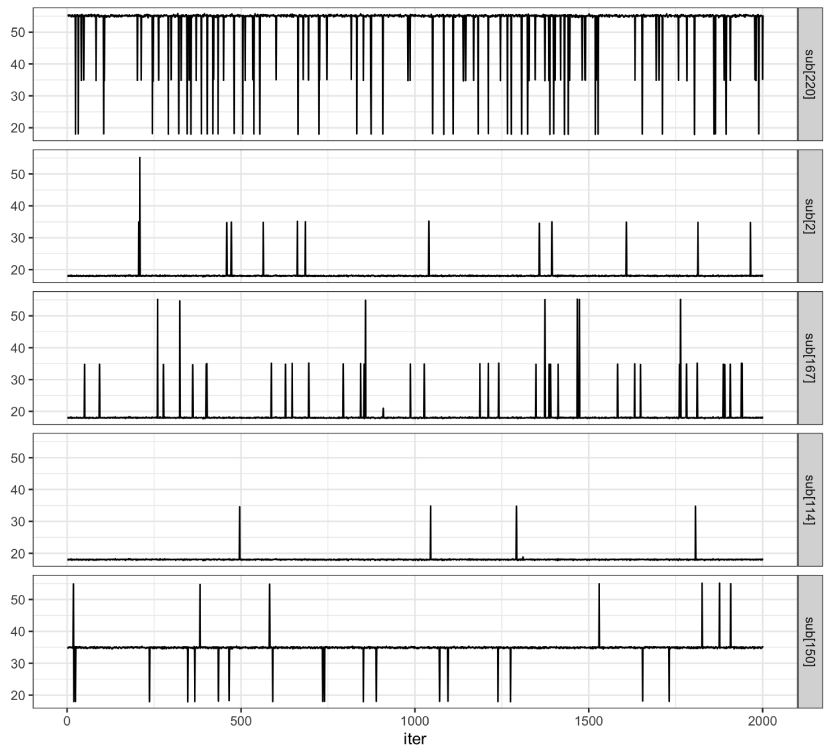
Cadeias para indivíduos selecionados aleatoriamente: 220, 2, 167, 114 e 150. No Cenário (a) são dos *clusters* 1, 2, 2, 3 e 1; no Cenário (b): 3, 1, 1, 1 e 2; no Cenário (c): 1, 1, 1, 1 e 2; e no Cenário (d): 1, 2, 2, 3 e 1, respecivamente.

Figura B.2: Cadeias das distribuições a posteriori dos parâmetros de locação do modelo ajustado cenário (a) dos indivíduos 220, 2, 167, 114 e 150.



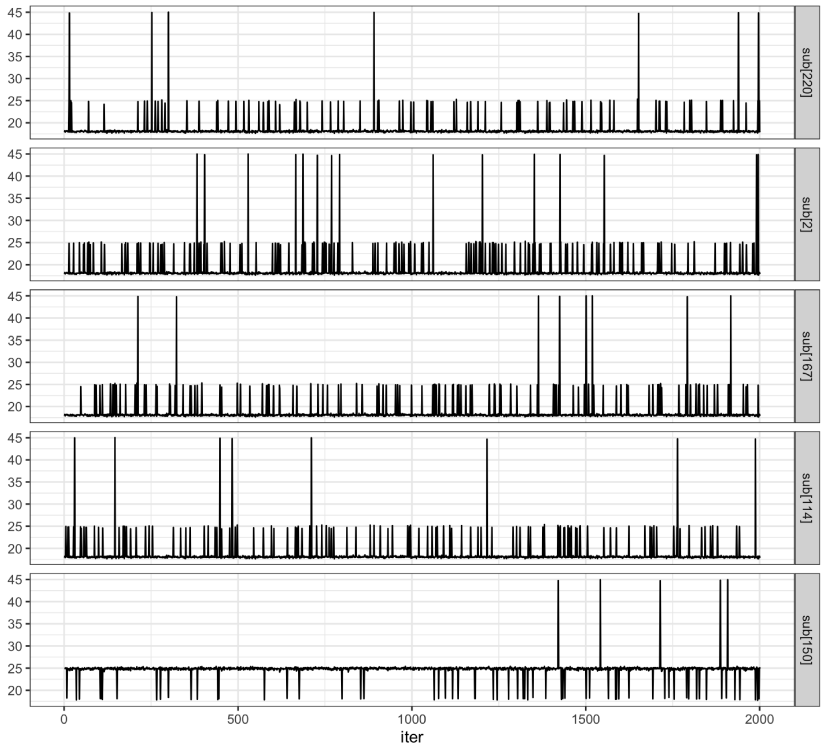
Fonte: Elaborado pela autora.

Figura B.3: Cadeias das distribuições a posteriori dos parâmetros de locação do modelo ajustado cenário (b) dos indivíduos 220, 2, 167, 114 e 150.



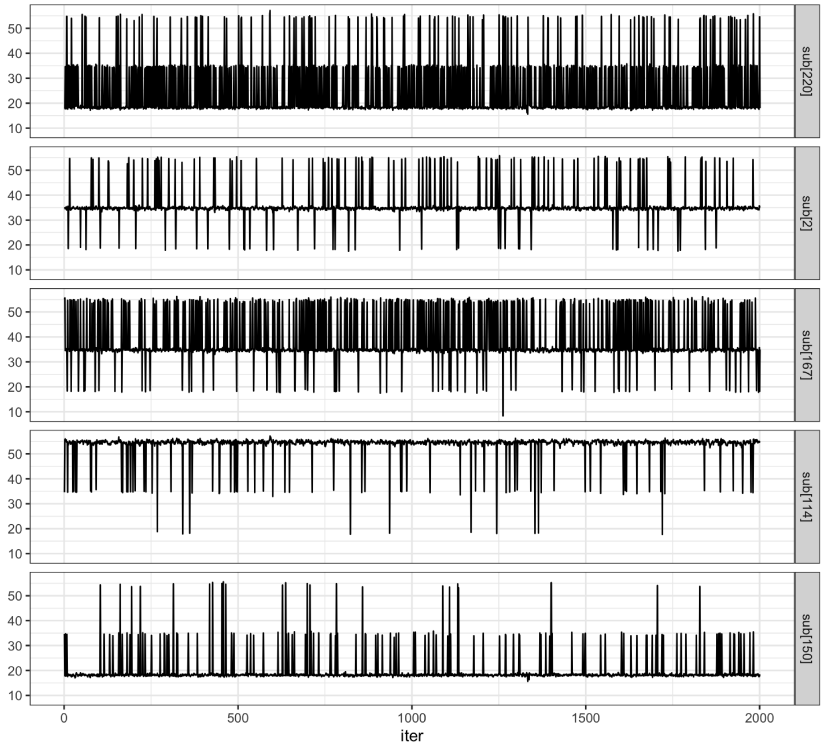
Fonte: Elaborado pela autora.

Figura B.4: Cadeias das distribuições a posteriori dos parâmetros de locação do modelo ajustado cenário (c) dos indivíduos 220, 2, 167, 114 e 150.



Fonte: Elaborado pela autora.

Figura B.5: Cadeias das distribuições a posteriori dos parâmetros de locação do modelo ajustado cenário (d) dos indivíduos 220, 2, 167, 114 e 150.



Fonte: Elaborado pela autora.

Tabela B.1: Taxas de aceitação dos ajustes do modelo para dados da biomassa.

η	θ^2 e σ_q^2	μ_1	μ_2	μ_3	μ_4	μ_5
2.1	0,185	0,014	0,645	0,627	0,588	0,596
5.0	0,185	0,673	0,012	0,653	0,640	0,633
100	0,186	0,515	0,014	0,620	0,677	0,583

Fonte: Elaborado pela autora.

B.2 Rotinas para o Modelo para redução de Dimensão de Variável Categórica

```
library(Rcpp)
library(RcppArmadillo)
library(RcppDist)

# Arquivos com rotinas no C
Rcpp::sourceCpp("~beta_sample.cpp")
Rcpp::sourceCpp("~u_sample.cpp")
Rcpp::sourceCpp("~Sigma_sample.cpp")
Rcpp::sourceCpp("~z_sample.cpp")
Rcpp::sourceCpp("~alpha_sample.cpp")
Rcpp::sourceCpp("~Z_cate.cpp")

#### ---- Dados: leitura e preparação ----
load("~/dataE.RData") #  data with X model factors
Y = dataE$Y; X = dataE[,2:6]; vcat = dataE$vcat;


########## ---- Fit Model Function  ---- ############

dpp_cate <- function(Y, X, vcat, hparams=NULL, store =
NULL, control=NULL, fixed=NULL, initial=NULL, verbose =
TRUE){

stime <- proc.time();

# Linear Model matrix
Xm <- compute_X_cate(Y, X);
```

```
# hparams=NULL; store=NULL; control=NULL; fixed=NULL;
verbose=TRUE;
# covariates
p <- ncol(Xm);

# categories in a factor
ycat <- as.numeric(as.factor(vcat));
#
# if(is.null(initial$mu_y)){
#   mu_y <- 0;
# } else{mu_y <- initial$mu_y;}

if(is.null(initial$nu)){
nu <- 5;
} else{nu <- initial$nu;}

# Hiperparameters
hparams.default <- list(
a0 = 2.3,     b0 = 1.461, # para Sigma
delta = 1, # para w
mubeta = rep(0,p), sigbeta = 10000*diag(p), sb_inv = solve(10000*diag(p)),
# para betas
a1 = 105,   b1 = 1, # para theta
a2 = 200,   b2 = 0.5, # para sigmaq
a3 = nu/2,   b3 = nu/2,  # para u,
mu.prop <- list(si = 0, th= 0), # means to adaptative M-H
v.prop = list(si = 5, th = 5, alpha = 10) # dp's das propostas p/ MH
);

if(!is.null(hparams)){
hparams <- c(hparams, hparams.default);
}else{hparams <- hparams.default};
cat(" nu: ", nu, ", delta: ", hparams$delta, " \n ");
# "var prop: ", hparams$v.prop$si,

control.default <- list(
niter = 40000,
burnin = 0,
thin = 1
);
```

```
if(!is.null(control)){
control <- c(control, control.default);
} else{control <- control.default}


# ---- Inicialize a cadeia ---- #
sm <- initialize_cate(Y, Xm, vcat, fixed, initial, hparams);
smpl <- sm$sp;
V <- sm$V;

Z <- Z_cate(z = smpl$z, K = smpl$K, V = V);
# Adaptative mcmc to sigmaq
mcsigq2 <- c(smpl$sigmaqˆ2,rep(NA, control$niter-1));
mcthet2 <- c(smpl$thetaˆ2,rep(NA, control$niter-1));

if(!is.null(initial$rhos)){
rhos <- initial$rhos;
}else{rhos <- list(rhots=0, rhoalpha = rep(0, smpl$K))};

# Store posteriori values
if (is.null(store)) {
store <- c(names(smpl));
} else if (length(store) == 1 && is.na(store)) {
store <- c("z", "K");
}

if (control$burnin <= 0) {
mcmc <- list(smpl);
} else {
mcmc <- list();
}

#### ---- Loop ---- ####

i <- 1;
mcmc[[i]] <- smpl[store]

for (iter in 2:(control$niter+1)) {

if (is.null(fixed$z)) {
smpl$z <- as.vector(z_sample(alpha = smpl$alpha, u = smpl$u,
```

```
beta = smpl$beta, Sigma = smpl$Sigma, V = V, Y = Y, X = Xm,
w = smpl$w, K = smpl$K,
ycat = ycat));
Z <- Z_cate(z = smpl$z, K = smpl$K, V = V);
}

if (is.null(fixed$w)) {
smpl$w <- update_w_cate(Z, hparams);
}

if (is.null(fixed$Sigma)) {
smpl$Sigma <- Sigma_sample(alpha = smpl$alpha, u = smpl$u,
beta = smpl$beta,
Y = Y, X = Xm, Z = Z, sb_inv = hparams$sb_inv,
mb = hparams$mubeta, a0 = hparams$a0, b0 = hparams$b0)
}

if (is.null(fixed$beta)) {
smpl$beta <- beta_sample(alpha = smpl$alpha, u = smpl$u,
Sigma = smpl$Sigma,
Y = Y, X = Xm, Z = Z, sigbeta = hparams$sigbeta,
mubeta = hparams$mubeta)
}

##### ---- u ---- ####
if (is.null(fixed$u)) {
# smpl$u <- update_u_cate(smpl, Y, Xm, Z, hparams);
smpl$u <- as.vector(u_sample(alpha = smpl$alpha,
beta = smpl$beta, Sigma = smpl$Sigma, Y = Y, X = Xm, Z = Z,
a3 = hparams$a3, b3 = hparams$b3));
}

if (is.null(fixed$alpha)) {
stemp_alpha <- alpha_sample(alpha = smpl$alpha, u = smpl$u,
beta = smpl$beta, Sigma =  smpl$Sigma, theta2 = smpl$theta^2,
sigmaq2 = smpl$sigmaq^2, Y = Y, X = Xm, Z = Z, K = smpl$K,
sigprop = hparams$v.prop$alpha);
smpl$alpha <- stemp_alpha$alpha;
rhos$rhoalpha <- rhos$rhoalpha + stemp_alpha$rho_alpha;
}
```

```
# Adaptative mcmc
if(iter<102){
mesigq2 <- hparams$v.prop$si; # mean(mcsigq2, na.rm =TRUE);
methet2 <- hparams$v.prop$th;
}else{
mesigq2 <- mean(c(sd(mcsigq2[(iter-100):(iter-1)], na.rm = T),
hparams$v.prop$si), na.rm = T);
methet2 <- mean(c(sd(mcthet2[(iter-100):(iter-1)], na.rm = T),
hparams$v.prop$th), na.rm = T);
}

##### ---- Kernel ---- ####
if(is.null(fixed$theta)|is.null(fixed$sigmaq)){
h_up <- update_kernel(smpl$theta^2, smpl$sigmaq^2, mesigq2, methet2,
smpl$alpha, smpl$z, hparams, fixed)
rhos$rhots <- rhos$rhots + h_up$rhots;
smpl$theta <- sqrt(h_up$t2);
smpl$sigmaq <- sqrt(h_up$sq);
mcsigq2[iter] <- smpl$sigmaq^2;
}

if (is.null(fixed$K)) {
smpl <- update_K_cate(smpl, Xm, E, hparams);
}

if (iter > control$burnin && iter %% control$thin == 0) {
i <- i + 1;
mcmc[[i]] <- smpl[store];
}

if (verbose && iter %% floor(control$niter/100) == 0) {#
message("iter: ", iter, ";  K = ", smpl$K, ";  rhos = ", rhos$rhots)
}
}

if (verbose) {
message("Elapsed time: ", proc.time()[3] - stime[3])
}

return(list(mcmc=mcmc, rhos = rhos))
```

```
}

#### ------------------------####
compute_X_cate <- function(Y, X) {
N <- nrow(X);
df <- data.frame(Y,X);
l <- lm(as.formula(df), data = df);
# z is N x 1
# alpha is K x 1
# w is K x 1
m <- model.matrix(lm(as.formula(df), data = df));
p <- ncol(m);
return(m[,2:p])
}

# Initialize
initialize_cate <- function(Y, X, vcat, fixed, initial, hparams) {
N <- nrow(X);
p <- ncol(X);

# One-hot encoding matrix V: categorical variable for subject
cat <- as.factor(vcat);
v <- as.matrix(model.matrix(~vcat));
v[,1] <- 2 - rowSums(v);
colnames(v)[1] <- "cat1";

if (is.null(fixed$K)) {
if (!is.null(fixed$alpha)) {
if(!is.null(nrow(fixed$alpha))&!is.null(fixed$alpha)){
fixed$K <- nrow(fixed$alpha)
} else {fixed$K <- length(fixed$alpha)};
} else if (!is.null(fixed$w)) {
fixed$K <- length(fixed$w);
} else if (!is.null(fixed$z)) {
fixed$K <- max(fixed$z);
}
}

if (is.null(fixed$K)) {
if(is.null(initial$K)){
K <- ceiling(2*log(nrow(X)));
```

```
} else{K <- initial$K}
} else {K <- fixed$K};


if (!all(c("z", "alpha", "w") %in% names(fixed)) ) {
cl <- kmeans(tapply(Y,vcat,mean), K);
}

if (is.null(fixed$z)) {
if(is.null(initial$z)){
z <- cl$cluster;
} else {z <- initial$z}
} else {z <- fixed$z};

nz <- as.numeric(names(table(z)));

if (is.null(fixed$w)) {
if(is.null(initial$w)){
w <- cl$size / sum(cl$size);
} else{w <- initial$w}
} else {w <- fixed$w};

if (is.null(fixed$alpha)) {
if(is.null(initial$alpha)){
alpha <- rnorm(K); #cl$centers;
} else{alpha <- initial$alpha}
} else {alpha <- fixed$alpha};
alpha <- matrix(alpha, nrow = K)

if (is.null(fixed$Sigma)) {
if(is.null(initial$Sigma)){
Sigma <- var(Y); #0.6655 verdadeiro
} else{Sigma <- initial$Sigma;}
} else {Sigma <- fixed$Sigma;}

if (is.null(fixed$u)) {
if(is.null(initial$u)){
u <- rep(1.0, N);
} else{u <- initial$u}
} else {u <- fixed$u};
```

```
if (is.null(fixed$beta)) {
if(is.null(initial$beta)){
beta <- rep(0, p);
} else{beta <- initial$beta}
} else {beta <- fixed$beta};

if (is.null(fixed$sigmaq)) {
if(is.null(initial$sigmaq)){
sigmaq <- N/(max()-min());
} else{sigmaq <- initial$sigmaq}
} else {sigmaq <- fixed$sigmaq};

if (is.null(fixed$theta)) {
if(is.null(initial$theta)){
theta <- quantile(dist(Y),0.5);
} else{theta <- initial$theta}
} else {theta <- fixed$theta};

list(sp = list(K = K,    z = z,    alpha = alpha,    w = w,
Sigma = Sigma, u = u,    beta = beta,    sigmaq = sigmaq,
theta = theta), V = v)
}

#------- Update w -----------------#

update_w_cate <- function(Z, hparams){
# Z must be have columns 1 ... K
m <- colSums(Z);

rdirichlet(1, hparams$delta + m);
}

flts <- function(t2, sq2, D=1){
h <- rowSums(expand.grid(1:500,1:500));
h <- h[order(h)];
a <- 1/(4*sq2);
b <- 1/(t2);
c <- sqrt(a^2+2*a*b); # já modificado por a e b
l <- (2*a/(a+b+c))^(D/2)*cumprod((b/(a+b+c))^(h-1));
return(prod(l[which(l>0)]+1))
}
```

```
logdet <- function(A){
2*sum(log(diag(chol(A))))
}

# ---------------------------------------- #

update_kernel <- function(t2, sq2, mu, z, hparams, fixed){
# counting acceptance for theta and sigmaq in block
rho.ts <- 0;
mu <- mu[sort(unique(z))];

# limiting \sigma_q
ts = (t2)/(2);
# proposals
sigmaq2.new <- EnvStats::rnormTrunc(1, mean = sq2, sd =
hparams$v.prop$si, min = 0) #min = ts
sigmaq.new <- sqrt(sigmaq2.new);

# limiting \theta
tt = 2*(sigmaq2.new);
# proposal
theta2.new <- EnvStats::rnormTrunc(1, mean = t2, sd =
hparams$v.prop$th, min = 0, max = tt)
theta.new <- sqrt(theta2.new);

if (is.null(fixed$theta) | is.null(fixed$sigmaq)){
lt.det <- logdet(cov_matrix(mu, theta2.new, sigmaq2.new)) -
logdet(cov_matrix(mu, t2, sq2));
lt.eng <- - log(flts(theta2.new, sigmaq2.new)) + log(flts(t2, sq2));
lt.pri <- log(cascsim::dtgamma(theta2.new, shape=hparams$a1, scale =
hparams$b1, max = tt)) +
dgamma(sigmaq2.new, shape=hparams$a2,
scale=hparams$b2, log=T) -
log(cascsim::dtgamma(t2, shape=hparams$a1, scale=hparams$b1,
max = tt)) -
dgamma(sq2, shape=hparams$a2, scale=hparams$b2, log=T);
lt.prop <- log(EnvStats::dnormTrunc(theta2.new, mean = t2,
sd = hparams$v.prop$si, min =0, max = 2*sigmaq2.new)) +
log(EnvStats::dnormTrunc(sigmaq2.new, mean = sq2, sd =
hparams$v.prop$si, min=0))-
```

```
log(EnvStats::dnormTrunc(t2, mean = theta2.new, sd =
hparams$v.prop$si, min =0, max = 2*sq2)) -
log(EnvStats::dnormTrunc(sq2, mean = sigmaq2.new, sd =
hparams$v.prop$si, min=0));

} else{ratio.t <- 1; theta2.new <- t2; sigmaq2.new <- sq2}

if (log(runif(1)) < ratio.t){t2 <- theta2.new;
sq2 <- sigmaq2.new; rho.ts <- 1}

return(list(t2 = t2, sq2 = sq2, rhots = rho.ts));
}

#### ---- Funçoes de Cov para DPP ---- ####

# Similiarty kernel with the covariance function
similarity_kernel <- function(x, y, t2) {
if(is.matrix(x)){
theta <- rep(t2, nrow(x))}else{t2 <- rep(t2, length(x))}
-(x - y)^2/t2
}


# Quality function
quality_kernel <- function(x, sq2) {
dnorm(x, mean=0, sd = sqrt(sq2), log = TRUE)#72.97333
}

# Covariance function on all pairs of rows of X
cov_matrix <- function(mu, t2, sq2) {
if(is.matrix(mu)){
KK <- nrow(mu);
} else{KK <- length(mu); mu <- matrix(mu,length(mu),1)}
C <- matrix(1.0, KK, KK)

# matrix is symmetric
for (i in 1:KK) {
for (j in i:KK) {
cij <- exp(sum(similarity_kernel(mu[i,], mu[j,], t2)));
qij <- exp(sum(quality_kernel(mu[i, ], sq2) +
quality_kernel(mu[j, ], sq2)));
```

```
C[i,j] <- C[j,i] <- cij*qij;
}
}

C
}

# Covariance function for a row vector (rvec) against each row of
# a matrix.
cov_rvec_mat <- function(rvec, mat, t2, sq2) {
if(length(rvec)==1){mat<-matrix(mat,ncol=1)}
# replicate row vector by row-wise to match dimension of mat
rmat <- matrix(rvec, nrow(mat), ncol(mat), byrow=TRUE);
matrix(exp(rowSums(quality_kernel_cate(rmat, sq2)+
similarity_kernel_cate(rmat, mat, t2)+
quality_kernel_cate(mat,sq2))), nrow=1)
}




# ----------------- #

rdirichlet <- function(n, delta) {
l <- length(delta);
# x <- matrix(rgamma(l * n, delta), ncol = l, byrow = TRUE);
x <- matrix(rgamma(l * n, shape = delta), ncol = l, byrow = TRUE);
sm <- x %*% rep(1, l);

x / as.vector(sm)
}
```

B.2.1 Rotinas em C pelo pacote Rcpp no R

Rotina para estimar o vetor de locação α.

```
######  Rotinas em C pelo pacote Rcpp do R  ######
```

```
// [[Rcpp::depends(RcppArmadillo)]]
#include <RcppArmadillo.h>
#include <RcppArmadilloExtensions/sample.h>

using namespace Rcpp;

/* cov cate */
// [[Rcpp::export]]
arma::mat cov_cate(arma::vec alpha, double theta2, double sigmaq2) {
int K = alpha.size();
arma::mat C(K,K);
double pi = arma::datum::pi;

// C cov
for (int i = 0; i < K; i++) {

// C(i,i) = sigmaq2;

C(i,i) = 1/(2*pi*sigmaq2)*exp(-alpha(i)*alpha(i)/sigmaq2);

for (int j = i+1; j < K; j++) {

//C(i,j) = (sigmaq2)*exp(-pow((alpha(i)-alpha(j)),2)/theta2);
C(i,j) = 1/(2*pi*sigmaq2)*exp(-(pow(alpha(i),2)+pow(alpha(j),2))
/(2*sigmaq2)-pow((alpha(i)-alpha(j)),2)/theta2);
C(j,i) = C(i,j);

}
}

return C;

}



/* cov vec cate */
// [[Rcpp::export]]
arma::rowvec cov_vec_cate(double alphak, arma::vec alpha, double theta2,
double sigmaq2) {
```

```
int K = alpha.size();
arma::rowvec C(K);
double pi = arma::datum::pi;

for (int i = 0; i < K; i++) {
// C(i) = (sigmaq2)*exp(-pow((alphak-alpha(i)),2)/theta2);
C(i) = 1/(2*pi*sigmaq2)*exp(-(pow(alphak,2)+pow(alpha(i),2))/
(2*sigmaq2)-pow((alphak-alpha(i)),2)/theta2);
}

return C;
}


/* alpha sample */
// [[Rcpp::export]]
List alpha_sample(arma::vec alpha, arma::vec u, arma::vec beta, double
Sigma, double theta2,
double sigmaq2, arma::vec Y, arma::mat X, arma::Mat<int> Z, int
K, double sigprop) {


arma::vec su = Z.t()*u;
arma::vec Va = arma::sqrt(Sigma/su);
arma::vec Ma = Z.t()*((Y-X*beta)%u)/su;
arma::vec L(K);
arma::vec alpha_prop(K);
double pi = arma::datum::pi;
arma::vec rho_alpha(K);
rho_alpha.zeros();

arma::mat C = cov_cate(alpha, theta2, sigmaq2);

for(int k = 0; k < K; k++){
L(k) = Rf_dnorm4(alpha(k), Ma(k), Va(k), true);
alpha_prop(k) = Rf_rnorm(alpha(k), sigprop);

double ratio_like = 0;

int mk = sum(Z.col(k));
if(mk > 0){
```

```
ratio_like = Rf_dnorm4(alpha_prop[k], Ma[k], sqrt(Va[k]), true) -
L[k];
}

arma::rowvec b_old = C.row(k);
b_old.shed_col(k);

arma::vec alpha_ = alpha;

alpha_.shed_row(k);

arma::rowvec b_new = cov_vec_cate(alpha_prop(k), alpha_, theta2,
sigmaq2);

arma::mat Cs = C;
Cs.shed_col(k);
Cs.shed_row(k);
arma::mat C_sinv = Cs.i();

double Cm_old = C(k, k);
double Cm_new = 1/(2*pi*sigmaq2)*exp(-alpha_prop(k)*
alpha_prop(k)/sigmaq2);

double Cv_old = Cm_old -
arma::as_scalar(b_old * C_sinv * b_old.t());
double Cv_new = Cm_new -
arma::as_scalar(b_new * C_sinv * b_new.t());

double ratio = exp(ratio_like)*Cv_new/Cv_old;

double u = Rf_runif(0.0, 1.0);

if (u < ratio) {
alpha.row(k) = alpha_prop(k);
C = cov_cate(alpha, theta2, sigmaq2);
rho_alpha(k) = 1;
}

}

List out;
```

```
out["alpha"] = alpha;
out["rho_alpha"] = rho_alpha;

return out;
}
```

Rotina em C para estimar os parâmetros comuns aos indivíduos $\boldsymbol{\beta}$.

```
// [[Rcpp::depends(RcppArmadillo)]]
#include <RcppArmadillo.h>
#include <RcppArmadilloExtensions/sample.h>

using namespace Rcpp;

#include <mvnorm.h>
// [[Rcpp::depends(RcppDist)]]


/* beta sample */
// [[Rcpp::export]]
arma::vec beta_sample(arma::vec alpha, arma::vec u, double Sigma,
arma::vec Y, arma::mat X, arma::Mat<int> Z,
arma::mat sigbeta, arma::vec mubeta) {

arma::mat sb_inv = arma::inv(sigbeta);
arma::vec mb = mubeta;
arma::vec alpha_est = Z * alpha;
arma::mat U = arma::diagmat(1/u);
arma::mat Vb = inv(X.t()*U*X + sb_inv);
arma::vec Mb = Vb * (sb_inv*mb + X.t()*U*Y - X.t()*U*alpha_est);

// beta sample
arma::vec beta = rmvnorm(1, Mb , Vb*Sigma).t();

return beta;

}
```

Rotina para estimar a variável misturadora $\mathbf{u}$.

```
// [[Rcpp::depends(RcppArmadillo)]]
```

```
#include <RcppArmadillo.h>
#include <RcppArmadilloExtensions/sample.h>

using namespace Rcpp;

#include <mvnorm.h>
// [[Rcpp::depends(RcppDist)]]

/* u sample */
// [[Rcpp::export]]
arma::vec u_sample(arma::vec alpha, arma::vec beta, double Sigma,
arma::vec Y, arma::mat X, arma::Mat<int> Z,
double a3, double b3) {

int N = X.n_rows;
arma::vec u(N);

arma::vec d = Y - X*beta - Z*alpha;
arma::vec Sy = (d%d)/Sigma;

arma::vec bb = b3 + Sy;

// u sample
for (int i = 0; i < N; i++) {
u(i) = Rf_rgamma( (a3 + 1)/2 , 2/bb(i) );
}

return u;

}
```

Rotinas em C para estimar o vetor de alocação e criar a matriz de alocação por indivíduo.

```
// [[Rcpp::depends(RcppArmadillo)]]
#include <RcppArmadillo.h>
#include <RcppArmadilloExtensions/sample.h>

using namespace Rcpp;


/* z sample */
```

```
// [[Rcpp::export]]
arma::rowvec z_sample(arma::vec alpha, arma::vec u, arma::vec beta,
double Sigma, arma::mat V, arma::vec Y, arma::mat X, arma::vec w,
double K, arma::vec ycat) {

int N = X.n_rows;
int J = V.n_cols;
arma::mat dlog(J,K);
arma::rowvec z(J);
const double log_max_K = arma::datum::log_max/K;
const arma::uvec kvec = arma::linspace<arma::uvec>(1, K, K);

arma::vec U = arma::sqrt(Sigma/u);
arma::vec Xb = X*beta;

dlog.zeros();

for (int i = 0; i < N; i++) {
for (int k = 0; k < K; k++) {
dlog(ycat(i)-1,k) += Rf_dnorm4(Y(i), Xb(i)+alpha(k), U(i), true);
}
}

// z sample
for (int j = 0; j < J; j++) {
dlog.row(j) += log(w.t());
double max_log_i = log_max_K - max(dlog.row(j));
arma::vec d = exp( dlog.row(j).t() + max_log_i );
d = d/sum(d);
z(j) = RcppArmadillo::sample(kvec, 1, true, d).at(0);
}

return z;

}


// [[Rcpp::depends(RcppArmadillo)]]
#include <RcppArmadillo.h>
#include <RcppArmadilloExtensions/sample.h>
```

```
using namespace Rcpp;

// [[Rcpp::depends(RcppDist)]]

/* Z cate */
// [[Rcpp::export]]
arma::mat Z_cate(arma::vec z, int K, arma::mat V) {

int J = z.size();
arma::mat Zv(J,K);
Zv.zeros();


for (int j = 0; j < J; j++){
Zv(j,z(j)-1) += 1;
}

arma::mat Z = V*Zv;

return Z;

}
```

Rotina em C para estimar a variância do modelo σ_y^2

```
// [[Rcpp::depends(RcppArmadillo)]]
#include <RcppArmadillo.h>
#include <RcppArmadilloExtensions/sample.h>

using namespace Rcpp;

#include <mvnorm.h>
// [[Rcpp::depends(RcppDist)]]

/* Sigma sample */
// [[Rcpp::export]]
double Sigma_sample(arma::vec alpha, arma::vec u, arma::vec beta,
arma::vec Y, arma::mat X, arma::Mat<int> Z,
arma::mat sb_inv,
arma::vec mb, double a0, double b0) {
```

```
int N = X.n_rows;
int p = X.n_cols;
double Sigma;

arma::mat U = arma::diagmat(1/u);
arma::vec dy = Y - X*beta - Z*alpha;
double sy = arma::as_scalar(dy.t()*U*dy);
arma::vec db = beta - mb;
double sbeta = arma::as_scalar(db.t()*sb_inv*db);

// Sigma sample
Sigma = 1/Rf_rgamma( (a0 + N + p)/2 , 2/(b0 + sbeta + sy) );

return Sigma;

}
```

Função para extrair as cadeias do MCMC para dados com variáveis categóricas com muitos níveis.

```
#### ---- Posterioris VCate ---- ####
estimate.post_cate <- function(object, fixed){
#object <- mcmc;
N <- length(object[[1]]$u) # tamanho da amostra
m <- length(object); # iterações
J <- length(object[[1]]$z); # categorias
K.mx <- object[[1]]$K; # 29 ou 16 # K máximo
p <- length(object[[1]]$beta); # covariáveis

mcmc.K <- rep(NA, m);
mcmc.z <- matrix(0, m, J);
mcmc.alpha <- matrix(NA, m, K.mx);
mcmc.Sigma <- rep(NA, m);
mcmc.w <- matrix(0, m, K.mx);
if(!is.null(fixed$theta)){
mcmc.theta <- object[[1]]$theta;
}else{mcmc.theta <- rep(NA, m)};
if(!is.null(fixed$sigmaq)){
mcmc.sigmaq <- object[[1]]$sigmaq;
}else{mcmc.sigmaq <- rep(NA, m)};
```

```
if(!is.null(fixed$u)){
mcmc.u <- object[[1]]$u;
}else{mcmc.u <- matrix(NA, m, N)};

alphaj <- array(NA, dim=c(m, J));
mcmc.beta <- matrix(0, m, p);

mcmc.param <- list();

for (i in 1:m) {
mcmc.alpha[i,] <- c(object[[i]]$alpha);
mcmc.beta[i,] <- c(object[[i]]$beta);
mcmc.z[i,] <- object[[i]]$z;
mcmc.K[i] <- length(unique(object[[i]]$z));
mcmc.Sigma[i] <- object[[i]]$Sigma;
mcmc.w[i,] <- object[[i]]$w;
if(is.null(fixed$theta)){mcmc.theta[i] <- object[[i]]$theta};
if(is.null(fixed$sigmaq)){mcmc.sigmaq[i] <- object[[i]]$sigmaq};
if(is.null(fixed$u)){mcmc.u[i,] <- object[[i]]$u};
alphaj[i,] <- mcmc.alpha[i, mcmc.z[i,]];
}

mcmc.param$K <- mcmc.K;
mcmc.param$z <- mcmc.z;
mcmc.param$alpha <- mcmc.alpha;
mcmc.param$beta <- mcmc.beta;
mcmc.param$Sigma <- mcmc.Sigma;
mcmc.param$w <- mcmc.w;
mcmc.param$theta <- mcmc.theta;
mcmc.param$sigmaq <- mcmc.sigmaq;
mcmc.param$u <- mcmc.u;
mcmc.param$alphaj <- alphaj;

return(mcmc.param)
}

# Run the function
post_Vcat <- estimate.post_cate(mcmc_vcat, fixed = NULL)
```

B.2.2 Rotina para a Avaliação de Reprodutibilidade

```
#### Reprodutibilidade ####
#### ---- 5 dobras da amostra ----####
# 4529/5 = 905.8, 4 grupos de 906 e 1 grupo de 905 indivíduos

#.Random.seed[1506]
set.seed(-1764602873)
seed_5dobras <- sample(1:4529, 4529, replace = F)
# Divisão das observações em 5 dobras
dobra <- matrix(c(as.numeric(seed_5dobras), NA), 906, 5)
vcat1<- factor(vcatec[-(dobra[,1])],exclude=NA)
vcat2<- factor(vcatec[-(dobra[,2])],exclude=NA)
vcat3<- factor(vcatec[-(dobra[,3])],exclude=NA)
vcat4<- factor(vcatec[-(dobra[,4])],exclude=NA)
vcat5<- factor(vcatec[-(dobra[-906,5])],exclude=NA)


vcat_dobra1 <- dpp_cate(Yc[-c(dobra[,1])], Xc[-c(dobra[,1]),], vcat1,
fixed = list(K=16), initial = list(nu = 100, sigmaq=30),
control=list(niter = 20000))
vcat_dobra2 <- dpp_cate(Yc[-(dobra[,2])], Xc[-(dobra[,2]),], vcat2,
fixed = list(K=16), initial = list(nu = 100, sigmaq=30),
 control=list(niter = 20000))
vcat_dobra3 <- dpp_cate(Yc[-(dobra[,3])], Xc[-(dobra[,3]),], vcat3,
fixed = list(K=16), initial = list(nu = 100, sigmaq=30),
control=list(niter = 20000))
vcat_dobra4 <- dpp_cate(Yc[-c(dobra[,4])], Xc[-c(dobra[,4]),], vcat4,
fixed = list(K=16), initial = list(nu = 100, sigmaq=30),
control=list(niter = 20000))
vcat_dobra5 <- dpp_cate(Yc[-c(dobra[-906,5])], Xc[-c(dobra[-906,5]),], vcat5,
fixed = list(K=16), initial = list(nu = 100, sigmaq=30),
control=list(niter = 20000))
saveRDS(vcat_dobra5,"vcat_dobra5.rds")
saveRDS(vcat_dobra4,"vcat_dobra4.rds")
saveRDS(vcat_dobra3,"vcat_dobra3.rds")
saveRDS(vcat_dobra2,"vcat_dobra2.rds")
saveRDS(vcat_dobra1,"vcat_dobra1.rds")

pos_db1 <- estimate.post_cate(vcat_dobra1$mcmc, NULL)
```

```
pos_db2 <- estimate.post_cate(vcat_dobra2$mcmc, NULL)
pos_db3 <- estimate.post_cate(vcat_dobra3$mcmc, NULL)
pos_db4 <- estimate.post_cate(vcat_dobra4$mcmc, NULL)
pos_db5 <- estimate.post_cate(vcat_dobra5$mcmc, NULL)


# ---- Anãlise das dobras ----

# ----Dobra 1 ####
#### Model matrix to vcat ####

#
id <- as.numeric(vcatec[dobra[,1]]) %in%
which(levels(vcatec)%in%vcat1)
lvs_alp <- which(levels(vcatec)%in%vcat1)
alpha <- rep(0,71)

# ---- Paramether estimate ----
slag<- seq(10001,20001,10)
beta_db1 <- apply(pos_db1$beta[slag,],2,quantile,probs=0.5)
alpha_db1 <- apply(pos_db1$alphaj[slag,],2,quantile,probs=0.5)
alpha[lvs_alp] <- alpha_db1

X_model1 <- compute_X_cate(Yc[dobra[,1]], Xc[dobra[,1],])
X_model1 <- X_model1[id,]

y_pred1<- (X_model1%*%beta_db1) + alpha[as.numeric(vcatec[dobra[,1]])][id]

y <- Yc[(dobra[,1])][id]


# Calculando RMSE e MAE

rmse1 <- sqrt(mean((y - y_pred1)^2))
mae1 <- mean(abs(y - y_pred1))
cat("RMSE:", rmse1, "\n")
cat("MAE:", mae1, "\n")

# ---- Dobra 2 ----
#### Model matrix to vcat ####
```

```
id <- as.numeric(vcatec[dobra[,2]]) %in% which((levels(vcatec)%in%vcat2))
lvs_alp <- which(levels(vcatec)%in%vcat2)
alpha <- rep(0,71)

# ---- Paramether estimate ----
slag<- seq(10001,20001,10) # retira burnin e lag
beta_db2 <- apply(pos_db2$beta[slag,],2,quantile,probs=0.5)
alpha_db2 <- apply(pos_db2$alphaj[slag,],2,quantile,probs=0.5)
alpha[which((levels(vcatec)%in%vcat2))] <- alpha_db2

X_model2 <- compute_X_cate(Yc[dobra[,2]], Xc[dobra[,2],])
X_model2 <- X_model2[id,]

y_pred2<-(X_model2%*%beta_db2) +
alpha[as.numeric(vcatec[dobra[,2]])][id]

y <- Yc[(dobra[,2])][id]

# Calculando RMSE e MAE

rmse2 <- sqrt(mean((y - y_pred2)^2))
mae2 <- mean(abs(y - y_pred2))
cat("RMSE:", rmse2, "\n")
cat("MAE:", mae2, "\n")
# RMSE: 0.6392672
# MAE: 0.485154

#   DIC

loglike <- function(y_p, Sg, estu){
sum(dnorm(y, mean=y_p, sd = sqrt((Sg/estu)), log=T))
}

Sig2 <- quantile(pos_db2$Sigma[slag], probs=0.5)
alp <- pos_db2$alphaj[slag,]
bt <- pos_db2$beta[slag,]
S2 <- pos_db2$Sigma[slag]
u  <- quantile(pos_db2$u[slag,id], probs=0.5)
uM <- apply(pos_db2$u[slag,id], 1, quantile, probs=0.5)
```

```
y <- Yc[(dobra[, 2])][id]

llikeh2 <- sum(dnorm(y, mean=y_pred2, sd = sqrt(Sig2/u), log=T))

pd<-0; y_p<-0;

for (i in 1:length(uM)){
alpha[lvs_alp] <- alp[i,];
y_p <- (X_model2%*%bt[i,]) +
alpha[as.numeric(vcatec[dobra[,2]])][id];
pd <- pd + loglike(y_p, S2[i], uM[i])
}

pd <- pd/length(uM)

pDIC <- 2*(llikeh2 - pd )
dic2 <- -2*llikeh2 + 2*pDIC

cat("DIC:", dic1, "\n")


# ---- Dobra 3 ----
#### Model matrix to vcat ####

id <- as.numeric(vcatec[dobra[,3]]) %in%
which((levels(vcatec)%in%vcat3))
lvs_alp <- which(levels(vcatec)%in%vcat3)
alpha <- rep(0,71)

# ---- Paramether estimate ----
slag<- seq(10001,20001,10) # retira burnin e lag
beta_db3 <- apply(pos_db3$beta[slag,],2,quantile,probs=0.5)
alpha_db3 <- apply(pos_db3$alphaj[slag,],2,quantile,probs=0.5)
alpha[which((levels(vcatec)%in%vcat3))] <- alpha_db3

X_model3 <- compute_X_cate(Yc[dobra[,3]], Xc[dobra[,3],])
X_model3 <- X_model3[id,]

y_pred3<- (X_model3%*%beta_db3) +
alpha[as.numeric(vcatec[dobra[,3]])][id]
```

```
y <- Yc[(dobra[,3])][id]

# Calculando RMSE e MAE

rmse3 <- sqrt(mean((y - y_pred3)^2))
mae3 <- mean(abs(y - y_pred3))
cat("RMSE:", rmse3, "\n")
cat("MAE:", mae3, "\n")

# Calculando  DIC

loglike <- function(y_p, Sg, estu){
sum(dnorm(y, mean=y_p, sd = sqrt((Sg/estu)), log=T))
}

Sig2 <- quantile(pos_db3$Sigma[slag], probs=0.5)
alp <- pos_db3$alphaj[slag,]
bt <- pos_db3$beta[slag,]
S2 <- pos_db3$Sigma[slag]
u  <- quantile(pos_db3$u[slag,id], probs=0.5)
uM <- apply(pos_db3$u[slag,id], 1, quantile, probs=0.5)

y <- Yc[(dobra[, 3])][id]

llikeh3 <- sum(dnorm(y, mean=y_pred3, sd = sqrt(Sig2/u), log=T))

pd<-0; y_p<-0;

for (i in 1:length(uM)){
alpha[lvs_alp] <- alp[i,];
y_p <- (X_model3%*%bt[i,]) + alpha[as.numeric(vcatec[dobra[,3]])][id];
pd <- pd + loglike(y_p, S2[i], uM[i])
}

pd <- pd/length(uM)

pDIC <- 2*(llikeh3 - pd )
dic3 <- -2*llikeh3 + 2*pDIC

cat("DIC:", dic3, "\n")
```

```
# ---- Dobra 4 ----
#### Model matrix to vcat ####

id <- as.numeric(vcatec[dobra[,4]]) %in%
which((levels(vcatec)%in%vcat4))
lvs_alp <- which(levels(vcatec)%in%vcat4)
alpha <- rep(0,71)

# ---- Paramether estimate ----
slag<- seq(10001,20001,10)
beta_db4 <- apply(pos_db4$beta[slag,],2,quantile,probs=0.5)
alpha_db4 <- apply(pos_db4$alphaj[slag,],2,quantile,probs=0.5)
alpha[which((levels(vcatec)%in%vcat4))] <- alpha_db4

X_model4 <- compute_X_cate(Yc[dobra[,4]], Xc[dobra[,4],])
X_model4 <- X_model4[id,]

y_pred4<- (X_model4%*%beta_db4) +
alpha[as.numeric(vcatec[dobra[,4]])][id]

y <- Yc[(dobra[,4])][id]

# Calculando RMSE e MAE

rmse4 <- sqrt(mean((y - y_pred4)^2))
mae4 <- mean(abs(y - y_pred4))
cat("RMSE:", rmse4, "\n")
cat("MAE:", mae4, "\n")



# ---- Dobra 5 ----
#### Model matrix to vcat ####

id <- as.numeric(vcatec[dobra[-906,5]]) %in%
which((levels(vcatec)%in%vcat5))
lvs_alp <- which(levels(vcatec)%in%vcat5)
alpha <- rep(0,71)

# ---- Paramether estimate ----
slag<- seq(10001,20001,10)
```

```
beta_db5 <- apply(pos_db5$beta[slag,],2,quantile,probs=0.5)
alpha_db5 <- apply(pos_db5$alphaj[slag,],2,quantile,probs=0.5)
alpha[which((levels(vcatec)%in%vcat5))] <- alpha_db5

X_model5 <- compute_X_cate(Yc[dobra[,5]], Xc[dobra[,5],])
X_model5 <- X_model5[id,]

y_pred5 <- (X_model5%*%beta_db5) +
alpha[as.numeric(vcatec[dobra[-906,5]])][id]

y <- Yc[(dobra[-906,5])][id]

# Calculando RMSE e MAE

rmse5 <- sqrt(mean((y - y_pred5)^2))
mae5 <- mean(abs(y - y_pred5))
cat("RMSE:", rmse5, "\n")
cat("MAE:", mae5, "\n")
# RMSE: 0.6579063
# MAE: 0.5015344

#### RMSE e MAE 5-fold ###

RMSE =(rmse1 + rmse2 + rmse3 + rmse4 + rmse5)/5
MAE = (mae1 + mae2 + mae3 + mae4 + mae5)/5
cat("RMSE:", RMSE, "\n")
cat("MAE:", MAE, "\n")



#### DIC  ####

# To  calcule  DIC

fit_test <- pvcat29_nu100d0;
slag<- seq(5001,20001,10)

loglike <- function(y_p, Sg, estu){
sum(-2*dnorm(y, mean=y_p, sd = sqrt((Sg/estu)), log=T))
}

Sig2 <- fit_test$Sigma[slag]
```

```
alp  <- fit_test$alphaj[slag,]
bt   <- fit_test$beta[slag,]

ui   <- fit_test$u[slag,]

y        <- Yc
S2_pred  <- mean(Sig2)
alp_pred <- apply(alp, 2, mean)
bt_pred  <- apply(bt, 2, mean)
ui_pred  <- apply(ui,2, mean)

# model matrix
X_model <- compute_X_cate(Yc, Xc) # model matrix

y_pred   <- X_model%*%bt_pred + alp_pred[vcatec]
# sum((y-y_pred)^2/4529)

loglike <- function(y_p, Sg, estu){
sum(-2*dnorm(y, mean=y_p, sd = sqrt((Sg/estu)), log=T))
}

#### Deviance
di<-numeric(length(Sig2)); y_p <- 0;

for (i in 1:length(Sig2)){
y_p <- (X_model%*%bt[i,]) + alp[i,][vcatec];
di[i] <- loglike(y_p, Sig2[i], ui[i,])
}

# Calculate the average deviance
dev_bar <- mean(di)

dev_est <- loglike(y_pred, S2_pred, ui_pred)


# Compute DIC
DIC <- 2*dev_bar-dev_est

# Print DIC
cat("DIC:", DIC, "\n")
```

B.3 Taxas de aceitação do MCMC para Modelo Variável Categórica

Tabela B.2: Taxas de aceitação do modelo NIPPD para variável categórica, dados educacionais, $K_{max} = 29$ e $K_{max} = 16$, $\delta = 1.0$, $a_0 = b_0 = 0,01$, $a_1 = 100, b_1 = 1$, $a_2 = 200, b_2 = 0,5$.

	$K_{max} = 29$			$K_{max} = 16$		
Parâmetro	2.1	5.0	100	2.1	5.0	100
θ^2 e σ_q^2	0,661	0,660	0,659	0,662	0,656	0,658
α_1	0,264	0,240	0,229	0,204	0,163	0,132
α_2	0,278	0,261	0,237	0,295	0,202	0,103
α_3	0,236	0,239	0,224	0,204	0,144	0,146
α_4	0,263	0,283	0,239	0,210	0,122	0,104
α_5	0,279	0,258	0,252	0,211	0,132	0,126
α_6	0,263	0,251	0,238	0,211	0,201	0,142
α_7	0,256	0,233	0,238	0,200	0,202	0,121
α_8	0,261	0,247	0,228	0,207	0,102	0,141
α_9	0,260	0,267	0,247	0,205	0,138	0,126
α_{10}	0,247	0,236	0,232	0,258	0,242	0,146
α_{11}	0,275	0,240	0,206	0,133	0,122	0,133
α_{12}	0,265	0,238	0,213	0,226	0,212	0,116
α_{13}	0,250	0,268	0,181	0,148	0,143	0,103
α_{14}	0,254	0,256	0,235	0,225	0,276	0,113
α_{15}	0,252	0,265	0,252	0,222	0,127	0,149
α_{16}	0,264	0,233	0,238	0,219	0,111	0,147
α_{17}	0,257	0,254	0,214			
α_{18}	0,264	0,248	0,241			
α_{19}	0,271	0,236	0,212			
α_{20}	0,259	0,246	0,231			
α_{21}	0,259	0,248	0,247			
α_{22}	0,266	0,263	0,222			
α_{23}	0,251	0,246	0,231			
α_{24}	0,262	0,256	0,253			
α_{25}	0,260	0,220	0,232			
α_{26}	0,235	0,264	0,237			
α_{27}	0,283	0,253	0,225			
α_{28}	0,282	0,273	0,244			
α_{29}	0,268	0,261	0,243			

Fonte: Elaborado pela autora.