Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Estatística

Classificação de dados espectrais de café utilizando análise de discriminantes via mistura finita de distribuições

Paulo Henrique Dourado da Silva **10/03810**

Brasília

2012

Paulo Henrique Dourado da Silva **10/03810**

Classificação de dados espectrais de café utilizando análise de discriminantes via mistura de distribuições

Relatório apresentado à disciplina Estágio Supervisionado II do curso de graduação em Estatística, Departamento de Estatística, Instituto de Exatas, Universidade de Brasília, como parte dos requisitos necessários para o grau de Bacharel em Estatística.

Orientador: Prof. Dr. George Freitas Von Borries

Brasília

2012

Agradecimentos

Este trabalho teve suporte da Embrapa através da bolsa de estágio fornecida pela Embrapa Arroz e Feijão - CNPAF (Centro Nacional de Pesquisa em Arroz e Feijão), durante o período de 10/2011 à 08/2012.

Primeiramente gostaria de agradecer a minha mãe, graças ao seu esforço e trabalho tive o privilégio de ter uma boa educação o que foi de fundamental importância para que eu chegasse até aqui. Tudo que fiz na minha vida acadêmica foi pensando nela para tentar retribuir, nem que seja um pouco, todo o carinho e dedicação que me foi dado. É nela que me espelho para continuar essa caminhada, realmente não sou nada sem ela.

Em segundo lugar agradeço ao meu pai e irmã. Ao meu pai também por sua dedicação e carinho. Vejo nele um exemplo de superação e perseverança por tudo que passou em sua vida desde a sua infância difícil até os problemas pessoais que quase lhe tiraram a vida. Portanto o vejo com um exemplo de homem a que devo seguir. E a minha irmã pelo companheirismo de sempre, tenho-a como exemplo de dedicação que de certa forma me influenciou em vários momentos da minha vida acadêmica.

Também gostaria de agradecer aos meus primos Victor e Maryângela, pelo apoio

em vários momentos difíceis da minha vida, pelo carinho, amor e dedicação. Muita das minhas alegrias devo aos dois que são extremamente importantes na minha vida, tenho um amor incondicional por eles.

Agradeço também aos meus amigos de longa data, Hadson, Arthur Carvalho, Artur Bosco, Bruno Cruz, Filipe Bispo e suas respectivas namoradas, pela paciência e carinho em vários momentos em que tive que sumir devido aos problemas acadêmicos. Também gostaria de agradecer aos meus amigos de graduação, Guilherme Maia, Brunno Augusto, Andreza e João Galvão. Tive muita sorte em dividir a minha vida acadêmica com eles que são pessoas extraordinárias além de eficientes, obrigado pelo aprendizado e companheirismo nesses quatro anos.

Agradeço também ao professor Dr. George Freitas von Borries, meu orientador, pela paciência e atenção dedicada a mim nesse trabalho. Ele me proporcionou aprendizados que vou levar por toda minha vida profissional.

Agradeço também a pesquisadora Fátima Chieppe Parazzi pelos dados fornecidos, os quais permitiram a realização deste trabalho.

Por fim gostaria de fazer um agradecimento especial a uma amiga muito querida por mim, Camila Leal. Na fase final deste trabalho tive alguns problemas com relação a motivação, e com seu carinho e conselhos tive forças para seguir em frente. Tenho muito a agradecer a ela, que se tornou muito importante na minha vida desde o final do ano passado.

Paulo Henrique Dourado da Silva

Resumo

Este trabalho tem por objetivo utilizar análise de discriminante via mistura de distribuições para a classificação de grãos de café em intactos ou defeituosos, e classificação dos grãos para o caso em que os grãos defeituosos estão separados por categorias (Danificados por insetos, quebrados, defeitos gerais e defeitos graves). Esse método permite que as matrizes de covariâncias de cada componente possam variar, tanto dentro das classes como entre as classes, tornando o método mais flexível e geral, com essa generalização é possível determinar a parametrização da matriz de covariâncias e qual o número de componentes da mistura é mais adequado para cada classe. O banco de dados consiste de medidas de absorvância de luz para os grãos intactos e defeituosos captados utilizando a técnica de espectroscopia no infravermelho próximo, variando o comprimento de onda de 500 a 1700 nm. Para a estimação dos parâmetros foi utilizado o algoritmo EM. O pacote estatístico utilizado foi o MCLUST implementado no software R.

Além disso, faz parte dessa monografia a identificação dos comprimentos de onda que possuem maior poder discriminatório baseado em um critério de separabilidade.

Palavras-chaves: Espectroscopia de infravermelho próximo, Distância de

Bhattacharrya, Discriminantes, Classificação, Mistura Finita de Distribuições.

Sumário

Capítulo 1

Introdução

O café é um grão de grande valor no mercado brasileiro e mundial movimentando, anualmente, algo em torno de quatro milhões de toneladas. Isso acaba envolvendo valores altíssimos entre 12 a 15 bilhões de dólares, portanto há uma atenção especial e um interesse do mercado internacional em ofertar tal produto isento de contaminantes devido ao seu valor. Há um interesse em ofertar esse produto isento de um contaminante denominado ochratoxina A (OTA), conhecida micotoxina de ocorrência natural no café [26]. Tal contaminante representa um perigo potencial à saúde do consumidor, além de proporcionar uma substancial perda econômica caso alguma saca venha contaminada.

Países que são produtores e consumidores de café, preocupados com a melhoria da qualidade no mercado interno e externo, vêm implementando medidas voltadas para o aspecto da segurança alimentar a fim de garantir a qualidade desejada. A utilização de mecanismos oficiais de inspeção no mercado internacional de produtos agrícolas é uma prática amplamente utilizada e caracteriza-se pela adoção de procedimentos que permitam avaliar a qualidade e verificar a conformidade destes produtos com os contratos e regulamentações previamente estabelecidas pelos países importadores e

exportadores [26].

Os métodos que são referências para a padronização e classificação dos produtos agrícolas foram criados a fim de facilitar a comercialização destes produtos entre os diversos países. Segundo [26], a detecção visual é um procedimento amplamente utilizado no mercado mundial, sendo utilizado pelos serviços de inspesão para o reconhecimento e remoção de produtos agrícolas infectados por fungos. Mas tal procedimento sozinho não é suficiente para esse reconhecimento, pois muitos produtos isentos de sintomas de infecção perceptíveis a olho nu podem apresentar elevados índices de contaminação por micotoxinas. Similarmente, grãos que são visivelmente infectados por fungos podem não conter contaminantes.

Dentre outras tecnologias disponíveis para a identificação de problemas em produtos agrícolas, os métodos óticos, utilizando alta velocidade de detecção e processamentos de informações, vêm sendo considerados os mais bem sucedidos por permitirem avaliações rápidas e acuradas de diferentes produtos [25, 26, 27]. Neste contexto, a utilização de tecnologias espectrométricas na faixa no infravermelho próximo (NIR) nos permite diferenciar produtos sadios e contaminados por fungos e micotoxinas, principalmente em grãos como milho, trigo, amendoim e café [26]. Outra justificativa para o uso dessa técnica é que outros atributos internos são detectáveis apenas na faixa do infravermelho próximo, não perceptíveis a olho nu.

Com o advento de interfaces computadorizadas houve um favorecimento da aplicação deste método, por permitir a coleta e análise de um grande número de informações, e com isso a estatística vem sendo uma ferramenta de suma importância

na análise dessas informações. Principalmente a estatística multivariada que permite estudar de forma quantitativa as características dos grãos de café, auxiliando na criação de índices de qualidade facilitando a comercialização.

O experimento com o qual trabalharemos foi realizado pela pesquisadora Fátima Chieppe Parizzi e consistiu em uma amostra aleatória de grãos de café. Dessa amostra 540 grãos intactos (sadios) e 540 grãos defeituosos (danificados) foram selecionados por análise visual. Esses grãos foram enviados a um laboratório para a obtenção dos dados espectrais (Absorvância da luz) variando o comprimento de onda, captados por um espectrômetro. Após a obtenção dos dados espectrais dos grãos, considerando cada grão como uma observação, os comprimentos de ondas como variáveis e como temos a informação da classe a que pertence o grão, foi utilizada análise de discriminantes via mistura de distribuições normais multivariadas para realizar a classificação dos grãos em intactos e defeituosos. Para a estimação dos parâmetros foi utilizado o algoritmo EM, considerando o vetor que rotula cada observação a um determinado grupo como não observado.

Análise de discriminantes via mistura de distribuições é uma técnica relativamente nova que vem mostrando resultados eficientes na construção de discriminantes para a classificação [6, 12, 17, 20], o caso particular utilizado nesses trabalhos é quando as componentes da mistura são normais multivariadas. Entretanto em [3], outras misturas foram desenvolvidas utilizando uma família de distribuições assimétricas, baseadas na fórmula de Azzalini [1], proporcionando modelos de mistura mais robustas no sentido não alterar drasticamente o número de grupos na presença

de outliers. Além de fornecer modelos mais parcimoniosos, ou seja, com menos componentes de densidades e consequentemente menos parâmetros para estimar.

O trabalho é descrito da seguinte forma: o segundo capítulo descreve a técnica de espectroscopia no infravermelho próximo, mostrando um pouco da história e algumas definições gerais.

O terceiro capítulo é totalmente dedicado a metodologia utilizada nesse trabalho. Esse capítulo começa falando do uso de mistura de distribuições em agrupamentos. Também é descrito o algoritmo para a estimação dos parâmetros do modelo de mistura (Algoritmo EM). Descreve também mistura de normais, que é um caso especial de mistura de distribuições na qual admitimos que as componentes da mistura são distribuições normais multivariadas, bem como ilustra alguns dos principais critérios para seleção do modelo. A parte da metodologia voltada para mistura de distribuições termina com análise de discriminantes e classificação utilizando mistura de distribuições, denominada por Raftery e Fraley [12] de MclustDA.

Na segunda parte da metodologia é descrito de maneira breve a distância de Bhattacharyya, expondo a sua definição geral e sua forma gaussiana. É exposto também as diferentes formas de se otimizar essa distância computacionalmente.

A terceira, e última parte, da metodologia descreve o pacote MCLUST implementado no software R bem como ilustra um exemplo de algumas de suas funções relacionadas à análise de discriminante via mistura de distribuições aplicadas ao banco de dados iris de Fisher 1936. Esse capítulo finaliza mostrando outros softwares disponíveis para modelagem usando mistura de distribuições.

O quarto capítulo mostra como foram obtidos os dados espectrais para os grãos de café e como é constituído o banco. Além disso, o capítulo expõe uma breve análise exploratória dos dados espectrais através da distância de Bhattacharyya e um pequeno algoritmo para redução da dimensão dos dados. Através dele identificamos os comprimentos de ondas que conseguem discriminar de maneira melhor os grãos em intactos e defeituosos.

O quinto capítulo utiliza os resultados do quarto capítulo e análise de discriminante via mistura de normais (MclustDA), para fazer a discriminação e classificação dos grãos dos dados de café para o caso binário (intacto ou defeituoso) e para o caso em que os grãos defeituosos são separados por categorias (danificados por insetos, quebrados, defeitos gerais, defeitos graves). Desse modo avalia-se também a eficácia do método observando a taxa de acerto gerada.

Por fim, o sexto capítulo mostra as conclusões obtidas nesse trabalho de uma maneira geral e uma discussão sobre todos os resultados obtidos no trabalho. Serão propostas também linhas de pesquisa e projetos futuros envolvendo dados espectrais e a metodologia aqui utilizada.

1.1 Objetivos

O objetivo geral do trabalho é utilizar a técnica de mistura de distribuições para classificação dos grãos de café. Os objetivos específicos são:

- classificar os grãos de café em intactos ou defeituosos, caso binário;
- classificar os grãos para o caso em que os grãos defeituosos são separados em

categorias utilizando a análise de discriminante via mistura de distribuições.

Capítulo 2

Espectroscopia no infravermelho próximo - NIR

Os métodos instrumentais para a aplicação do infravermelho próximo na agricultura foram desenvolvidos pelo departamento de agricultura dos Estados Unidos em meados dos anos 70. Historicamente, Karl Norris iniciou seus trabalhos com a tecnologia NIR procurando por novos métodos para a determinação da umidade nos produtos agrícolas, primeiramente pela extração da água no metanol depois pela suspensão de sementes moídas em CCl_4 [25].

De forma geral Espectroscopia consiste no estudo de radiação eletromagnética emitida ou absorvida por um corpo que pode ser luz visível, infravermelho, raios-X, elétrons, etc. [27]. Quando uma amostra é irradiada, a luz é absorvida seletivamente e dá origem a um espectro. Devido à diferenciação química entre os materiais, existem respostas diferentes a absorção da luz. Os espectros são obtidos através de instrumentos denominados espectrofotômetros, que consiste de uma fonte luminosa, um monocromador que contém os comprimentos de onda tipo prisma, um receptáculo para amostra e uma impressora ou computador. No capítulo relacionado à análise preliminar dos dados será descrito de forma mais detalhada como são obti-

dos os dados espectrais para os grãos de café utilizados nessa monografia.

A região do espectro correspondente ao infravermelho está situada depois da região do visível e abrange a radiação com números de onda no intervalo de comprimento de onda de 780 a 100.000 nm. Por ser uma faixa extensa é dividida em três categorias: Infravermelho próximo (NIR), infravermelho médio (MIR) e infravermelho distante (FIR). Essas categorias estão ilustrados na tabela 2.1.

Região	Comprimento de onda
Próximo (NIR)	780 a 2.500
Médio (MIR)	2.500 a 5.000
Distante (FIR)	5.000 a 100.000

Tabela 2.1: Regiões espectrais do infravermelho [27]

Segundo [25], a espectroscopia no infravermelho próximo tem sido reconhecida como uma poderosa técnica analítica para a determinação, de forma rápida, de vários constituintes em muitos materiais agrícolas e outras matérias-primas. Em seu trabalho Nigoski lista as seguintes vantagens do NIR [25]:

- Análises não destrutivas;
- Não utilização de produtos químicos;
- Design robusto e compacto;
- Análise múltipla de componentes;
- Velocidade de resultados de análise (menos de um minuto);
- Transferência de calibrações entre equipamentos.

e como principal desvantagem, a calibração requer: tempo.

Segundo Parizzi [26], a utilização das tecnologias espectrométricas envolvendo o café vêm sendo descrita, mas estão voltadas para análise de bebidas, grãos, extratos e pós, visando identificar a composição de misturas das espécies *Coffea arabica* e *C. robusta*. Entretanto, além das exigências organolépticas relacionadas às preferências de cada mercado, o setor cafeeiro, acompanhando as tendências mundiais, vem definindo prioridades voltadas à sanidade e qualidade do produto oferecido ao consumidor.

Capítulo 3

Metodologia: Mistura Finita de Distribuições

Mistura finita de distribuições é uma técnica de modelagem estatística bastante flexível capaz de lidar com situações complexas através de elementos simples e tratáveis [18]. Devido a sua grande flexibilidade os modelos de misturas finitas têm recebido bastante atenção nos últimos anos, tanto do ponto de vista prático quanto do ponto de vista teórico. Nas últimas décadas houve um aumento no potencial de aplicações do modelo de mistura de distribuições em várias áreas do conhecimento, como por exemplo, astronomia, biologia, genética, medicina, psiquiatria, economia, engenharia, entre outras áreas. Dentro dessas aplicações o modelo de mistura de distribuições é a base de várias técnicas estatísticas, incluindo análise de agrupamento, análise de classe latente, análise de discriminante, análise de imagens e análise de sobrevivência [22].

Na literatura estatística a primeira análise envolvendo o uso de modelos de mistura de distribuições ocorreu no final do século XIX pelo biometricista Karl Pearson. Em seu artigo Pearson, ajustou uma mistura de densidades de probabilidades normais, com diferentes médias, variâncias e proporções, para um conjunto de dados

que consistiam de medida da razão entre a testa e o corpo de caranguejos fornecido por Weldon.

Em algumas aplicações de mistura de distribuições o agrupamento de dados é o objetivo principal da análise, nestes casos os modelos de misturas estão sendo usados como dispositivo para expor qualquer agrupamento que pode existir na base de dados. Utilizando mistura de distribuições para análise de agrupamentos assumimos que os dados a serem agrupados são realizações de uma mistura com g grupos, os quais devem ser especificados antes da análise. Existe uma relação de correspondência entre componentes da mistura e o grupo, ou seja, cada componente do modelo de mistura define um grupo.

Existem casos em que o número de grupos g não é conhecido, e assim passa a ser o parâmetro mais importante que deve ser estimado. Na prática, para estimar g são usados critérios de seleção de modelos para encontrarmos a mistura que melhor ajuste os dados e consequentemente, o número de grupos em que os dados estão possivelmente distribuídos. O agrupamento dos dados aos g grupos se faz de forma probabilística em termos das probabilidades a *posteriori* ajustadas, como será descrito nas próximas seções.

O modelo de mistura de distribuições descreve o comportamento de uma variável aleatória através da soma de g densidades de probabilidades $f_1, f_2, \ldots, f_g$, onde cada densidade é ponderada por π_i, que no contexto de misturas é definido como a probabilidade a *priori* da variável aleatória pertencer a componente f_i. No contexto de agrupamento usando mistura de distribuições, π_i é visto como a probabilidade

a *priori* da observação pertencer ao grupo i, com isso $\sum_{i=1}^{g} \pi_i = 1$. Portanto a distribuição de uma variável aleatória X que segue uma mistura é dada por (usando a regra da probabilidade total):

$$f(x) = \sum_{i=1}^{g} \pi_i f_i(x) \tag{3.1}$$

Onde $f_i(x)$ é chamado de componente de densidade e π_i é a proporção da mistura. A seguir será dada a formulação paramétrica do modelo de misturas.

3.1 Formulação Paramétrica do Modelo de Misturas Finitas de Distribuições

Em muitas aplicações, as componentes de densidade pertencem a uma família paramétrica de densidades $f_i(x; \theta_i)$, onde $\theta_\mathbf{i}$ é um vetor de parâmetros desconhecido relacionado a i-ésima componente de densidade. Portanto podemos reescrever (3.1) como:

$$f(x; \mathbf{\Psi}) = \sum_{i=1}^{g} \pi_i f_i(x; \theta_i). \tag{3.2}$$

Onde o vetor $\mathbf{\Psi}$ contém todos os parâmetros desconhecidos do modelo de mistura, ou seja,

$$\mathbf{\Psi} = (\pi_1, \ldots, \pi_{g-1}, \theta_1, \ldots, \theta_g)\,. \tag{3.3}$$

Note que através da relação $\sum_{i=1}^{g} \pi_i = 1$, π_g pode ser omitido. Para o nosso estudo estaremos a procura do melhor modelo que ajuste bem os dados e com isso trataremos g como parâmetro, sendo assim,

$$\boldsymbol{\Psi} = (g, \pi_1, \ldots, \pi_{g-1}, \theta_1, \ldots, \theta_g)\,. \tag{3.4}$$

3.2 Problemas

Existem algumas dificuldades na utilização dos modelos de mistura citados por [18], que estão listados abaixo

- A escolha das distribuições $f_1, \ldots, f_g$ a serem empregadas para compor a distribuição que descreve as misturas.

- A procura de técnicas de estimação dos parâmetros da distribuição da mistura, ou seja, estimar os parâmetros $(\boldsymbol{\pi}, \boldsymbol{\Psi})$, em que $\boldsymbol{\Psi}$ pode ser dado por (3.3) ou (3.4).

- A determinação prática do tamanho de amostras para estimar os parâmetros $(\boldsymbol{\pi}, \boldsymbol{\Psi})$.

- A decisão de utilizar g conhecido ou não, e caso se opte por incluí-lo no conjunto paramétrico desconhecido, escolher a técnica mais adequada para estimá-lo [18].

3.3 Interpretação do Modelo de Mistura

Seja $\mathbf{Y}_1, \ldots, \mathbf{Y}_n$, uma amostra aleatória de tamanho n, onde $\mathbf{Y}_j$ é um vetor aleatório com função densidade de probabilidade $f(\mathbf{y}_j)$ sobre $\Re^p$. Na prática, $\mathbf{Y}_j$ é um vetor contendo p medidas feitas sobre o indivíduo i. Seja $\mathbf{Z}_j$ uma variável aleatória

categórica que pode assumir os valores $1, \ldots, g$ com probabilidade $\pi_1, \ldots, \pi_g$, respectivamente. Na prática $\mathbf{Z}_j$ é um vetor indicador aleatório, onde o i-ésimo elemento (z_{ij}) é definido como um ou zero dependendo se a componente de origem de $\mathbf{Y}_j$ na mistura é igual a i ou não $(i = 1, \ldots, g)$. Suponha que a densidade condicional de $\mathbf{Y}_j$ dado que $\mathbf{Z}_j = i$ é $f_i(\mathbf{Y}_j)$, portanto a densidade marginal de $\mathbf{Y}_j$, $f(\mathbf{y}_j)$, usando a regra da probabilidade total será dada por:

$$\begin{aligned} f(\mathbf{y}_j) &= \sum_{i=1}^{g} f(\mathbf{y}_j, \mathbf{z}_j = i) \\ &= \sum_{i=1}^{g} P(\mathbf{z}_j = i) f(\mathbf{y}_j | \mathbf{z}_j = i) \\ &= \sum_{i=1}^{g} \pi_i f(\mathbf{y}_j) \end{aligned} \tag{3.5}$$

Para contextualizar a interpretação acima, considere a seguinte situação. Suponha que $\mathbf{Y}_j$ é retirada de uma população G que consiste de g grupos, $G_1, \ldots, G_g$ em que a proporção de elementos que pertecem ao grupo G_i é dada por π_i. Se a densidade de $\mathbf{Y}_j$ dentro G_i é dada por $f_i(\mathbf{y}_j)$ para $i = 1, \ldots, g$, então a densidade de $\mathbf{Y}_j$ tem a forma de (3.5). Nesta situação fica claro que cada componente da mistura corresponde a um grupo de G. Uma observação importante a se fazer é que que o vetor $\mathbf{Z}_j$ é distribuído como uma distribuição multinomial consistindo de uma retirada sobre as g categorias com probabilidade $\pi_1, \ldots, \pi_g$, ou seja,

$$\mathbf{Z}_j \sim Mult_g(1, \pi_1, \ldots, \pi_g) \tag{3.6}$$

3.4 Misturas de Normais Multivariadas

Dentro dos modelos de mistura de distribuições existentes na literatura, consideramos no trabalho o caso especial em que as componentes densidades são normais multivariadas, o que nos leva a mistura de normais. Logo a componente de densidade é dada por

$$f_i(\mathbf{y}_j;\theta_i) = \phi(\mathbf{y}_j;\boldsymbol{\mu}_i;\boldsymbol{\Sigma}_i) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_i|^{1/2}}\exp\left[\frac{1}{2}(\mathbf{y}_j-\boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}_i^{-1}(\mathbf{y}_j-\boldsymbol{\mu}_i)\right] \quad (3.7)$$

e com isso (3.2) pode ser reescrito como

$$f(\mathbf{y}_j;\boldsymbol{\Psi}) = \sum_{i=1}^{g}\pi_i\phi(\mathbf{y}_j;\boldsymbol{\mu}_i;\boldsymbol{\Sigma}_i). \quad (3.8)$$

Nesta situação o vetor de parâmetros $\boldsymbol{\Psi}$ será $\boldsymbol{\Psi}=(\pi_1,\ldots,\pi_{g-1},\boldsymbol{\xi}^T)^T$, onde $\boldsymbol{\xi}^T$ contém os vetores de médias e as matrizes de covariâncias das componentes de densidades.

Dados gerados por mistura de normais são caracterizados por *clusters* centrados nas médias μ_k, com altas densidades em pontos próximos as médias. As curvas de nível possuem um formato elipsoidal. As características geométricas dos clusters (forma, volume e orientação) são determinadas pelas matrizes de covariâncias $\boldsymbol{\Sigma}_k$, com $k = 1, 2, \ldots, G$. Ao trabalhar-se com mistura de normais deve-se assumir uma estrutura para as matrizes de variância e covariância e também se estas serão iguais para cada componente, ou se serão diferente. Algumas suposições são mais comuns acerca da estrutura da matriz de variância e covariância dos diferentes grupos. Sendo elas:

- $\boldsymbol{\Sigma}_1 = \ldots = \boldsymbol{\Sigma}_k = \sigma^2\mathbf{I}$, com σ desconhecido;
- $\boldsymbol{\Sigma}_1 = \ldots = \boldsymbol{\Sigma}_k = \text{diag}(\sigma_1^2, \ldots, \sigma_k^2)$;
- $\boldsymbol{\Sigma}_1 = \ldots = \boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}$, com $\boldsymbol{\Sigma}$ em forma geral.

A primeira estrutura coloca a matriz de variância e covariância de todos os grupos como sendo iguais a uma matriz diagonal com todos os elementos iguais. Já a segunda permite que os elementos da diagonal sejam diferentes e por último, a terceira estrutura permite a existência de elementos fora da diagonal, o que representa correlação entre as variáveis. Todas estas estruturas são homocedásticas.

Na literatura vários autores propuseram diferentes parametrizações para $\boldsymbol{\Sigma}_k$ [9, 17, 29]. Banfield e Raftery [2], propuseram uma forma geral para estudar a geometria de cada *cluster* em mistura de normais multivariadas parametrizando as matrizes de covariâncias através da decomposição espectral da forma

$$\boldsymbol{\Sigma}_k = \lambda_k \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k^T, \tag{3.9}$$

onde $\lambda_k = \det \boldsymbol{\Sigma}_k^{1/d}$ (d dimensão dos dados), $\mathbf{D}_k$ é matriz de autovetores de $\boldsymbol{\Sigma}_k$ e $\mathbf{A}_k$ é uma matriz diagonal tal que $\det \mathbf{A}_k = 1$, com os autovetores normalizados de $\boldsymbol{\Sigma}_k$ na diagonal em ordem decrescente. Através da decomposição espectral somos capazes de percorrer possíveis estruturas para as matrizes, desde a mais simples que supõe que todos os *clusters* possuem a mesma matriz de variância e covariância e esta é uma matriz diagonal com todos os elementos iguais até a estrutura onde as matrizes são livres para serem totalmente diferentes em cada *cluster*.

Para $\lambda_k, \mathbf{A}_k, \mathbf{D}_k$ fixos, as seguintes propriedades geométricas são consideradas:

Nome	Σ_k	Distribuição	Volume	Formato	Orientação
EII	$\lambda\mathbf{I}$	Esférica	Igual	Igual	NA
VII	$\lambda_k\mathbf{I}$	Esférica	Variável	Igual	NA
EEI	$\lambda\mathbf{D}\mathbf{D}^T$	Diagonal	Igual	Igual	NA
VEI	$\lambda_k\mathbf{D}\mathbf{D}^T$	Diagonal	Variável	Igual	NA
VVI	$\lambda\mathbf{D}_k\mathbf{D}_k^T$	Diagonal	Variável	Variável	NA
EEE	$\lambda\mathbf{D}\mathbf{A}\mathbf{D}^T$	Elipsoidal	Igual	Igual	Igual
EEV	$\lambda\mathbf{D}\mathbf{A}_k\mathbf{D}^T$	Elipsoidal	Igual	Igual	Variável
VEV	$\lambda\mathbf{D}_k\mathbf{A}\mathbf{D}_k^T$	Elipsoidal	Variável	Igual	Variável
VVV	$\lambda_k\mathbf{D}_k\mathbf{A}_k\mathbf{D}_k^T$	Elipsoidal	Variável	Variável	Variável

Tabela 3.1: Tabela de parte dos modelos restritos a parametrização da matriz de covariâncias

- $\mathbf{D}_k$ determina a orientação da k-ésima componente de mistura;
- $\mathbf{A}_k$ determina a forma da k-ésima componente de mistura;
- λ_k determina o volume da k-ésima componente de mistura.

Celeux e Grovaert [9], generalizaram essa parametrização e obtiveram modelos mais parcimoniosos, parte deles estão ilustrados na tabela 3.1. Tais modelos estão implementados no pacote MCLUST do sofware R.

3.5 Estimação dos Parâmetros da Mistura via Algoritmo EM

O grande avanço da computação, nos últimos 20 anos, permitiu um grande avanço em técnicas para ajustar misturas de distribuição. Entre essas técnicas está o algoritmo EM (*Expectation-Maximization*). Este algoritmo talvez seja o método de estimação mais utilizado na prática por produzir boas estimativas e com propriedades assintóticas ótimas. O algoritmo EM foi proposto por Dempster *et al.* [10] e consiste de um método interativo para o cálculo de estimativas de máxima verossimilhança

de um parâmetro θ de uma família de distribuições paramétricas. O algoritmo foi desenvolvido para calcular as estimativas de MV em dados que apresentam *missing values*, ou seja, dados perdidos. Conceito, propriedades, convergência e outros detalhes do algoritmo são dadas de maneira ampla por [21] e para uma leitura mais rápida o leitor pode consultar [16]. Nessa seção serão mostrados os conceitos básicos do algoritmo EM e seu uso em mistura de distribuições numa estrutura de dados incompletos.

3.5.1 Estrutura de Dados incompletos

Considere $\mathbf{y}_1, \ldots, \mathbf{y}_n$ uma realização de n vetores aleatórios *i.i.d* $\mathbf{Y}_1, \ldots, \mathbf{Y}_n$, com distribuição comum $f(\mathbf{y}_j)$. Quando o algoritmo EM é usado, os dados y são vistos como incompletos uma vez que os vetores indicadores $\mathbf{z}$, que indentificam a qual componente a observação pertence, não são conhecidos. Portanto o vetor dos dados completos é dado da seguinte forma:

$$\mathbf{y}_c = (\mathbf{y}^T, \mathbf{z}^T)^T \tag{3.10}$$

Uma observação importante a ser notada é que o vetor $\mathbf{y}$ é visto com sendo completamente observado (sem valores perdidos). Os vetores de indicadores $\mathbf{z}_1, \ldots, \mathbf{z}_n$ são vistos como realizações dos vetores aleatórios $\mathbf{Z}_1, \ldots, \mathbf{Z}_n$, onde para dados de características independentes, é apropriado assumir que os vetores são distribuidos marginalmente por (3.6).

A i-ésima proporção da mistura é vista como a probabilidade a *priori* da observação pertencer a i-ésima componente da mistura. Portanto a probabilidade a

posteriori da observação pertencer a i-ésima componente da mistura é dada por:

$$\tau_i(\mathbf{y}_j) = P(Z_{ij} = 1|\mathbf{y}_j) = \frac{\pi_i f_i(\mathbf{y}_j)}{f(\mathbf{y}_j)} \tag{3.11}$$

onde $f(\mathbf{y}_j)$ é dado por (3.5). A suposição de que o vetor z se distribuí marginalmente como uma multinomial significa que a distribuição do vetor dos dados completos $\mathbf{Y}_c$ implica em uma distribuição apropriada para o vetor dos dados incompletos $\mathbf{Y}$ [22]. Usando essa estrutura de dados incompletos a expressão da função da log-verossimilhança é simplificada e dada por [8, 22]:

$$\log L_c(\mathbf{\Psi}) = \sum_{i=1}^{g}\sum_{j=1}^{n} z_{ij}\left(\log\pi_i + \log f_i(\mathbf{y}_j;\theta_i)\right) \tag{3.12}$$

3.5.2 Aplicação do Algoritmo

O algoritmo EM é um método iterativo de estimação de MV para dados incompletos. Tal algoritmo trabalha iterativamente em duas etapas, a etapa E (Esperança) e a etapa M (Maximização). No contexto de mistura de distribuções o algoritmo EM é aplicado tratando z_{ij} como um dado não observado. A etapa E consiste em calcular a esperança condicional da log verossimilhança dos dados completos, $\mathrm{Log}L_c(\mathbf{\Psi})$, dado os dados observados usando a estimativa atual do vetor $\mathbf{\Psi}$. Usando $\mathbf{\Psi}^{(0)}$ como um vetor com valores iniciais para $\mathbf{\Psi}$, a primeira iteração do algoritmo EM requer o cálculo da esperança condicional de $\mathrm{Log}L_c(\mathbf{\Psi})$ dado $\mathbf{y}$ usando $\mathbf{\Psi}^{(0)}$ no lugar de $\mathbf{\Psi}$, o qual pode ser escrito como

$$Q(\mathbf{\Psi}, \mathbf{\Psi}^{(0)}) = E_{\mathbf{\Psi}^{(0)}}\left(\log L_c(\mathbf{\Psi})|\mathbf{y}\right). \tag{3.13}$$

O subscrito $\boldsymbol{\Psi}^{(0)}$ na esperança (3.13), indica que a esperança está sendo calculada substituindo $\boldsymbol{\Psi}$ por $\boldsymbol{\Psi}^{(0)}$. Após alguma álgebra pode-se chegar ao seguinte resultado

$$\begin{aligned} E_{\boldsymbol{\Psi}^{(0)}}\left(\log L_c(\boldsymbol{\Psi})|\mathbf{y}\right) &= E_{\boldsymbol{\Psi}^{(0)}}\left(\sum_{i=1}^{g}\sum_{j=1}^{n} z_{ij}\left(\log\pi_i + \log f_i(\mathbf{y}_j;\boldsymbol{\theta}_i)\right)|\mathbf{y}\right) \\ &= \sum_{i=1}^{g}\sum_{j=1}^{n} E_{\boldsymbol{\Psi}^{(0)}}\left(z_{ij}\left(\log\pi_i + \log f_i(\mathbf{y}_j;\boldsymbol{\theta}_i)\right)|\mathbf{y}\right) \\ &= \sum_{i=1}^{g}\sum_{j=1}^{n} E_{\boldsymbol{\Psi}^{(0)}}(z_{ij}|\mathbf{y})\left(\log\pi_i + \log f_i(\mathbf{y}_j;\boldsymbol{\theta}_i)\right) \end{aligned} \tag{3.14}$$

onde:

$$\begin{aligned} E_{\boldsymbol{\Psi}^{(0)}}(z_{ij}|\mathbf{y}) = P(z_{ij}=1|\mathbf{y}) &= \frac{\pi_i^{(0)} f_i(\mathbf{y}_j;\boldsymbol{\theta}_i^{(0)})}{\sum_{h=1}^{g}\pi_h^{(0)} f_h(\mathbf{y}_j;\boldsymbol{\theta}_i^{(0)})} \\ &= \tau_i(\mathbf{y}_j|\boldsymbol{\Psi}^{(0)}) \end{aligned} \tag{3.15}$$

Com isso após k interações do algoritmo a etapa E será dada por

$$Q(\boldsymbol{\Psi},\boldsymbol{\Psi}^{(k)}) = \sum_{i=1}^{g}\sum_{j=1}^{n}\tau_i(\mathbf{y}_j|\boldsymbol{\Psi}^{(k)})\left(\log\pi_i + \log f_i(\mathbf{y}_j;\boldsymbol{\theta}_i)\right) \tag{3.16}$$

A próxima etapa do algoritmo é maximizar $Q(\boldsymbol{\Psi},\boldsymbol{\Psi}^{(k)})$ sobre um determinado espaço paramétrico definido para $\boldsymbol{\Psi}$ para obter uma atualização $\boldsymbol{\Psi}^{(k+1)}$. Note que (3.16) pode ser reescrito como:

$$Q(\boldsymbol{\Psi},\boldsymbol{\Psi}^{(k)}) = \sum_{i=1}^{g}\sum_{j=1}^{n}\tau_i(\mathbf{y}_j|\boldsymbol{\Psi}^{(k)})\log\pi_i + \sum_{i=1}^{g}\sum_{j=1}^{n}\tau_i(\mathbf{y}_j|\boldsymbol{\Psi}^{(k)})\log f_i(\mathbf{y}_j;\boldsymbol{\theta}_i). \tag{3.17}$$

Note que $\tau_i(\mathbf{y}_j|\boldsymbol{\Psi}^{(k)})$ é uma constante pelo fato de atribuirmos valores para os parâmetros. Com isso as atualizações de π_i e para $\boldsymbol{\xi} = (\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_n)$ são dadas de forma independente. Para π_i trabalha-se apenas com o primeiro termo de (3.17) e por isso maximiza-se esse termo sujeito a seguinte restrição $\sum_{i=1}^{g}\pi_i = 1$. utilizando multiplicadores de Lagrange para maximizar esse termo, obtemos a seguinte atualização para π_i:

$$\pi_i^{(k+1)} = \frac{\sum_{i=1}^{g} \tau_i(\mathbf{y}_j|\mathbf{\Psi}^{(k)})}{n}. \tag{3.18}$$

Para estimar $\boldsymbol{\xi}$ trabalha-se com o segundo termo de (3.17), logo a atualização para o vetor $\boldsymbol{\xi}$ é dada resolvendo as seguintes equações de verossimilhança:

$$\sum_{i=1}^{g}\sum_{j=1}^{n} \tau_i(\mathbf{y}_j|\mathbf{\Psi}^{(k)}) \frac{\partial \log f_i(\mathbf{y}_j;\boldsymbol{\theta}_i)}{\partial \boldsymbol{\xi}}. \tag{3.19}$$

Pode-se resumir o algoritmo EM através das seguintes etapas [16]:

1. Seja $k = 0$ e dê uma entrada inicial para $\mathbf{\Psi}^{(0)}$ para $\mathbf{\Psi}$

2. Seja $\mathbf{y}_c = (\mathbf{y}, \mathbf{z})$ a representação dos dados completos

3. Para cada interação $k = 0, 1, 2, \ldots$ calcule $Q(\mathbf{\Psi}, \mathbf{\Psi}^{(k)})$

4. Encontre $\mathbf{\Psi}$ que maximize $Q(\mathbf{\Psi}, \mathbf{\Psi}^{(k)})$ e obtenha uma atualização $\mathbf{\Psi}^{(k+1)}$

5. Para $k = k + 1$ repita o passo 3. Alterne entre os passos 3 e 4 até que $|l_c(\mathbf{\Psi}^{(k+1)}) - l_c(\mathbf{\Psi}^{k})| < \epsilon$, onde $\epsilon > 0$ é arbitrariamente pequeno.

As demostrações para os calculos da etapa E e M podem ser encontrados em [8, 21].

3.5.3 Exemplo

Para ilustrar o uso do algoritmo EM, considere um caso especial em que as componentes de densidade são normais multivariadas, ou seja:

$$f_i(\boldsymbol{y}_j;\boldsymbol{\theta}_i) = \phi(\boldsymbol{y}_j;\boldsymbol{\mu}_i;\boldsymbol{\Sigma}_i). \tag{3.20}$$

O passo E se mantem o mesmo, mas agora $\tau_i(\mathbf{y}_j|\boldsymbol{\Psi}^{(k)})$ é dado por

$$\frac{\pi_i^{(0)}\phi(\mathbf{y}_j;\boldsymbol{\mu}_i^k;\boldsymbol{\Sigma}_i^k)}{\sum_{h=1}^{g}\pi_k^{(0)}\phi(\mathbf{y}_j;\boldsymbol{\mu}_h^{(k)};\boldsymbol{\Sigma}_h^{(k)})}. \tag{3.21}$$

A etapa M gera as seguintes atualizações para $\boldsymbol{\mu}_i$ e $\boldsymbol{\Sigma}_i$:

$$\boldsymbol{\mu}_i^{(k+1)} = \frac{\sum_{j=1}^{n}\tau_i(\mathbf{y}_j;\boldsymbol{\Psi}^{(k)})\mathbf{y}_j}{\sum_{j=1}^{n}\tau_i(\mathbf{y}_j;\boldsymbol{\Psi}^{(k)})} \tag{3.22}$$

e

$$\boldsymbol{\Sigma}_i^{(k+1)} = \frac{\sum_{j=1}^{n}\tau_i(\mathbf{y}_j;\boldsymbol{\Psi}^{(k)})(\mathbf{y}_j-\boldsymbol{\mu}_i^{(k+1)})(\mathbf{y}_j-\boldsymbol{\mu}_i^{(k+1)})^T}{\sum_{j=1}^{n}\tau_i(\mathbf{y}_j;\boldsymbol{\Psi}^{(k)})}. \tag{3.23}$$

As demonstrações são dadas por [8].

3.6 Agrupamento de Dados via Mistura de Distribuições

Em muitas aplicações de modelos de misturas, questões relacionadas a agrupamento de dados podem ser levantadas após o modelo ser ajustado. Como exemplo considere que foi usada mistura de três distribuições t de *student* para obter um modelo satisfatório para a distribuição de um conjunto de dados de interesse. Em uma análise de agrupamento de dados, cada componente pode ser vista como um grupo ou subpopulação na qual os dados estão possivelmente divididos.

Em outras aplicações de misturas de distribuições, o agrupamento de dados é o objetivo principal da análise, nestes casos os modelos de misturas estão sendo usados como dispositivo para expor quaisquer agrupamentos que possam existir na base de dados.

Utilizando mistura de distribuições para análise de agrupamentos assumimos que os dados a serem agrupados são realizações de uma mistura com g grupos, com g especificado antes. Existe uma relação de correspondência entre componentes das misturas e o grupo, ou seja, cada componente do modelo de mistura define um grupo. Nas próximas seções será exposta a idéia de como alocar os indíviduos aos grupos bem como alocar novos indivíduos após definidos os grupos.

3.6.1 Abordagem Teórica de Decisão

Após definidos os grupos utilizando o modelo de mistura, considere o problema de classificar um novo indivíduo observado após a agrupamento. No contexto de misturas de distribuições essa classificação é feita utilizando as componentes de mistura. Considere $r(\mathbf{y}_j)$ uma regra para classificar o vetor de característica de um determinado indivíduo $\mathbf{y}_j$ e consequentemente o próprio indivíduo para uma das componentes da mistura (grupo), onde $r(\mathbf{y}_j) = i$ implica que o j-ésimo indivíduo foi classificado na i-ésima componente (grupo) com $i = 1, \ldots, g$. A regra ótima ou de Bayes $r_B(\mathbf{y}_j)$ para a classificação $\mathbf{y}_j$ é definida por:

$$r_B(\mathbf{y}_j) = i \quad \text{se} \quad \tau_i(\mathbf{y}_j) \geq \tau_h(\mathbf{y}_j) \;\; (h = 1, \ldots, g). \tag{3.24}$$

Ou seja:

$$r_B(\mathbf{y}_j) = \arg\max_h \tau_h(\mathbf{y}_j) \tag{3.25}$$

onde $\tau_i(\mathbf{y}_j)$ é definido por (3.11), para caso paramétrico. Portanto monta-se uma matriz $\mathbf{B}$ $n \times g$ na qual em cada linha estão dispostos os indivíduos a serem

classificados e cada coluna representa a probabilidade a *posteriori* do indivíduo pertencer a cada uma das componentes (grupos). A matriz pode ser esquematizada da seguinte forma:

$$\mathbf{B} = \begin{bmatrix} \tau_1(\mathbf{y}_1) & \tau_2(\mathbf{y}_1) & \cdots & \tau_g(\mathbf{y}_1) \\ \tau_1(\mathbf{y}_2) & \tau_2(\mathbf{y}_2) & \cdots & \tau_g(\mathbf{y}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \tau_1(\mathbf{y}_n) & \tau_2(\mathbf{y}_n) & \cdots & \tau_g(\mathbf{y}_n) \end{bmatrix}$$

Nesse caso o indivíduo pode ser classificado arbitrariamente para uma das componentes (grupos) para a qual a probabilidade a *posteriori* correspondente é igual ao valor máximo. Mclachlan [20] assume que os custos de classificação correta são zero e todas as classificações erradas possuem o mesmo custo.

3.6.2 Agrupamento de dados I.I.D - Formulação paramétrica

Suponha que foi ajustado um modelo de mistura de distribuições com a finalidade de agrupar uma amostra aleatória $\mathbf{y}_1, \ldots, \mathbf{y}_n$ em g grupos. Em termos da especificação de dados completos do modelo de mistura, é de interesse inferir $\mathbf{z}_j$ com base no vetor de característica $\mathbf{y}_j$. Após ajustar g componentes de mistura aos dados para obter $\hat{\mathbf{\Psi}}$ do vetor de parâmetros desconhecidos do modelo, pode ser feito um agrupamento probabilístico dos n indivíduos em termos da probabilidade a *posteriori* do indivíduo pertencer a componente. A estimativa para as probabilidades $\tau_1(\mathbf{y}_j; \mathbf{\Psi}), \ldots, \tau_g(\mathbf{y}_j; \mathbf{\Psi})$ é dada por $\tau_1(\mathbf{y}_j; \hat{\mathbf{\Psi}}), \ldots, \tau_g(\mathbf{y}_j; \hat{\mathbf{\Psi}})$.

O agrupamento total dos dados (*hard clustering*) é dado atribuindo cada $\mathbf{y}_j$ para a componente da mistura a qual possui a maior probabilidade a *posteriori*. Ou seja,

essa probabilidade pode ser considerada uma estimativa de máxima verossimilhança para $\mathbf{z}_j$, onde $\hat{z_{ij}} = (\hat{z_j})_i$ é definido por:

$$\hat{z_{ij}} = \begin{cases} 1, & \text{se } \ i = \arg\max_h \tau_h(\mathbf{y}_j; \hat{\mathbf{\Psi}}), \\ 0, & \text{caso contrário.} \end{cases} \tag{3.26}$$

Segue que (3.26) corresponde a versão amostral *plug-in* da regra de Bayes, $r_B(\mathbf{y}_j; \hat{\mathbf{\Psi}})$. Se a mistura ajustou bem os dados e as misturas de proporções π_i são estimados com uma boa precisão, então $r_B(\mathbf{y}_j; \hat{\mathbf{\Psi}})$ deverá ser uma boa aproximação para a regra de Bayes $r_B(\mathbf{y}_j; \mathbf{\Psi})$.

3.7 Seleção do Modelo

Na prática podemos ajustar vários modelos para um conjunto de dados, com isso surge o problema de qual modelo escolher. Na literatura existem vários métodos para seleção de modelo [22]. Nessa seção, serão abordados os quatro principais critérios para seleção do modelo para ajustar os dados, existem outros critérios que serão omitidos aqui pela pouca utilização na prática. Para o leitor interessado em outros critérios as seguintes referências podem ser consultadas [6, 7, 11, 22].

Se uma boa estimativa para a *log* verossimilhança ($L(\hat{\theta})$) esperada puder ser obtida através dos dados observados, esta estimativa poderá ser utilizada como um critério para comparar modelos. Com isso pode-se usar $L(\hat{\theta_i})$ para comparar n modelos quaisquer $g_1(x|\theta_1), \ldots, g_n(x|\theta_n)$, mas existe um problema: tal método não permite a comparação verdadeira entre os modelos já que o modelo verdadeiro $g(x)$ não é conhecido. Primeiramente o método da máxima verossimilhança estima os valores de θ_i e depois os utiliza para estimar $E_G\left[\log f(x|\hat{\theta})\right]$. Mas muitos dos esti-

madores gerados por MV são viesados, e, sendo $\hat{\theta}$ uma estimativa viesada para θ, isso implica em um viés para $L(\hat{\theta})$, sendo que a magnitude desse viés varia de acordo com o número de parâmetros do modelo.

Os critérios de informação são construidos de forma a corrigir o viés de $L(\hat{\theta})$. A forma geral de um critério de informação é dada por [22]:

$$IC(\mathbf{\Psi}) = -2L(\mathbf{\Psi}) + 2C \tag{3.27}$$

onde C é o viés ou penalidade que mede a complexidade do modelo. Escolhemos o modelo que minimiza o critério (3.27).

3.7.1 Critério de Informação Bayesiano - BIC (*Bayesian Information Criterion*)

O Critério de Informação Bayesiano é dado por (3.27) mas fazendo $C = p\ log(n)$, onde p é o número de parâmetros do modelo e n é a dimensão dos dados.

3.7.2 Critério de Akaike - AIC (*Akaike Information Criterion*)

O Critério de Akaike é dado por (3.27) mas fazendo $C = p$, onde p é o número de parâmetros do modelo.

3.7.3 Critério de Determinação Eficiente - EDC (*Efficient Determination Criterion*)

O Critério de Determinação Eficiente é dado por (3.27) fazendo $C = 0,2\sqrt{n}$.

3.7.4 Critério da Verossimilhança Completa Integrada - ICL (*Integrated Complete Likelihood*)

Esse critério difere dos critérios citados nas seções anteriores pois busca maximizar a verossimilhança dos dados completos:

$$p(y, z|M_j) = \int p(y, z|a_j, M_j)p(a_j|M_j)da_j \tag{3.28}$$

onde z é a variável aleatória não-observada que indica o *cluster* de onde se origina a observação y. O critério ICL busca medir a qualidade do ajuste aos dados completos, penalizada da mesma forma que o BIC. Pode-se mostrar que o Critério da Verossimilhança Completa Integrada [5], é dado por:

$$ICL = \log p(y, \hat{z}|\hat{a_j}, M_j) - \frac{p}{2}log(n) \tag{3.29}$$

onde em (3.29), os dados faltantes z foram substituidos pela sua estimativa de MV $\hat{z_{ik}} = \gamma_{ik}$, tal que

$$\gamma_{ik} = \frac{\pi_k f_k(\mathbf{y}_i|\mathbf{\Psi}_k)}{f(\mathbf{y}_i|\mathbf{\Psi})} \tag{3.30}$$

onde $f(\mathbf{y}_i|\mathbf{\Psi})$ é definido por (3.2). Segundo [5] o ICL tende a rejeitar modelos em que haja componentes sobrepostas.

3.8 Análise de discriminante

Nesta seção será dada uma breve descrição sobre análise de discriminate no contexto de mistura de distribuições, que é o foco do trabalho. A descrição será feita segundo Raftery e Fraley [12].

3.8.1 Análise de discriminantes - Uma revisão

Em análise de discriminante, classificações conhecidas de algumas observações (Amostra de treinamento) são utilizadas para classificar outras. O número de classes C é conhecida. Muitas técnicas são probabilísticas baseada na suposição de que as observações na k-ésima classe são geradas por uma distribuição de probabilidade específica da classe, $f_k(.)$. Então se π_k é a proporção de indivíduos da população que pertencem a classe k, o teorema de Bayes diz que a probabilidade a *posteriori* de que a observação y pertença a classe k é

$$P(\mathbf{y} \in \text{Classe } k) = \frac{\pi_k f_k(\mathbf{y})}{\sum_{j=1}^{C} \pi_j f_j(\mathbf{y})} \tag{3.31}$$

Atribuindo $\mathbf{y}$ para a classe a qual possui a maior probabilidade a *posteriori*, minimizando a taxa de erro esperado, esse é chamado de classificador de bayes.

Os métodos mais comuns usados em análise de discriminantes são baseados na suposição de normalidade para as observações em cada classe, ou seja

$$f_k(y) = \phi(\mathbf{y}; \boldsymbol{\mu}_k; \boldsymbol{\Sigma}_k). \tag{3.32}$$

Se a matriz de covariâncias para as diferentes classes iguais e se as estimativas de máxima verossimilhança de $\boldsymbol{\mu}_k$ e $\boldsymbol{\Sigma}_k$ a partir da amostra de treinamento são usadas, então o classificador de Bayes é o discriminante linear de Fisher. Nesse caso, a regra de classificação é definida verificando se uma combinação linear dos componentes de $\mathbf{y}$ excede ou não um limiar [19]. Esse método reduz a discriminação em um problema unidimensional e produz uma regra de classificação que é um limite simples. Se as matrizes de convariâncias $\boldsymbol{\Sigma}_k$ diferem para as diferentes classe, então

o método resultante é a análise de discriminantes quadrática, na qual a função de classificação é uma forma quadrática das componentes de $\mathbf{y}$ [12, 19].

3.8.2 Análise de discriminante via mistura de distribuições

Uma abordagem baseado em modelagem para generalizar o discriminane linear e quadrático é assumir que a densidade para cada classe é uma mistura de normais multivariadas [12], ou seja

$$f_k(\mathbf{y}|\theta_k) = \sum_{j=1}^{C_j} \pi_{kj}\phi\left(\mathbf{y}|\boldsymbol{\mu}_{kj}, \boldsymbol{\Sigma}_{kj}\right) \tag{3.33}$$

Essa técnica foi sugerida inúmeras vezes na literatura [20, 30] e é a base da análise de discriminante por mistura MDA [17], no desenvolvimento do MDA, ou autores fizeram duas suposições: Todas as matrizes de covariâncias de cada componente são iguais (i.e, $\boldsymbol{\Sigma}_{kj} = \boldsymbol{\Sigma}$) e o número de componentes da mistura é conhecido a priori em cada classe.

Raftery e Fraley [12] estenderam essa análise relaxando as suposições impostas acima e aplicando o modelo baseado em agrupamentos citado nas seções anteriores para os membros de cada classe na amostra de treinamento. Isso permite que as matrizes de covariâncias de cada componente possam variar, tanto dentro das classes como entre as classes, tornando o método mais flexível e geral. Com essa generalização é possível determinar a parametrização da matriz de covariâncias e qual o número de componentes da mistura é mais adequado para cada classe através dos critérios mencionados na seção anterior. Essa generalização foi chamada de MclustDA, e está implementada no pacote MCLUST do software R.

Para Rafery e Fraley essa abordagem permite uma aproximação não linear e não monotônica para os limites de classificação. Sob condições fracas, um modelo de mistura pode aproximar uma dada densidade com grande precisão usando um número suficiente de componente, permitindo assim uma grande flexibilidade.

3.9 Distância de Bhattacharyya

A distância de Bhattacharyya fornece uma medida de separabilidade entre duas classes. Ela é caracterizada como ótima quando utilizam-se um par de classes normais e sub-ótima para situações envolvendo mais de duas classes ao mesmo tempo [15]. Nesta seção será apresentada uma introdução sobre essa medida e será apresentada a sua forma geral e gaussiana de acordo com [4, 23, 24], além de suas propriedades e casos especiais.

3.9.1 Distância de Bhattacharyya: Forma Geral

A distância de Bhattacharyya é definida como

$$B = -\ln\left[\int_{-\infty}^{\infty}\sqrt{P(\mathbf{x}|\omega_i)P(\mathbf{x}|\omega_j)}dx\right] \tag{3.34}$$

Sendo $P(\mathbf{x}|\omega_i)$ e $P(\mathbf{x}|\omega_j)$ as densidades a *posteriori* associadas as classes ω_i e ω_j respectivamente. Para interpretar a distância de Bhattacharyya, note que se as funções originais estão bem separadas e a probabilidade de $\mathbf{X}$ com respeito a classe ω_i for alta, a probabilidade de $\mathbf{X}$ com respeito ω_j será próxima de zero e consequentemente $P(\mathbf{x}|\omega_i)P(\mathbf{x}|\omega_j) \to 0$ e $B \to \infty$. Por outro lado se as densidades se sobrepõem então a probabilidade de $\mathbf{X}$ pertencer a classe ω_i e a probabilidade de

$\mathbf{X}$ pertencer a classe ω_j são iguais e $\int_{-\infty}^{\infty} \sqrt{P(\mathbf{x}|\omega_i)P(\mathbf{x}|\omega_j)}dx = 1$ e $B = 0$.

A distância de Bhattacharyya é invariante frente a uma transformação linear do vetor $\mathbf{X}$ e também é aditiva quando os componentes de $\mathbf{X}$ são indepententes, isto é, pode ser expressa como uma soma dos termos similares com cada termo envolvendo somente uma das componentes [23]. Considere que $J_p(\omega_i, \omega_j)$ representa a distância de Bhattacharyya entre duas classes baseada na observação x com p variáveis, então as seguintes propriedades métricas de uma função são apropriadas

$$J_p(\omega_i, \omega_j) > 0 \quad \omega_i \neq \omega_j \tag{3.35}$$

$$J_p(\omega_i, \omega_i) = J_p(\omega_j, \omega_j) = 0 \tag{3.36}$$

$$J_p(\omega_i, \omega_j) = J_p(\omega_j, \omega_i) \tag{3.37}$$

$$J_p(\omega_i, \omega_j) \leq J_{p+1}(\omega_i, \omega_j) \tag{3.38}$$

Tais propriedades não satisfazem a desigualdade triângular, e assim não podem ser classificadas como funções verdadeiras de distâncias [23].

3.9.2 Distância de Bhattacharyya: Forma Gaussiana

Para dados normalmente distribuídos a expressão (3.34) é escrita como [23, 24]

$$B = \frac{1}{8}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T \left(\frac{\boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j}{2}\right)^{-1} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) + \frac{1}{2}\ln\left[\frac{|\boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_j|}{2|\boldsymbol{\Sigma}_i|^{1/2}|\boldsymbol{\Sigma}_j|^{1/2}}\right] \tag{3.39}$$

onde $\boldsymbol{\mu}_i$ e $\boldsymbol{\mu}_j$ são os vetores de médias das classes ω_i e ω_j respectivamente, $\boldsymbol{\Sigma}_i$ e $\boldsymbol{\Sigma}_j$ as matrizes de covariâncias. Segundo [24] a distância B é uma medida bastante

conveniente para estimação da separabilidade entre pares de classes. Em (3.39) a primeira parcela estima a contribuição dos vetores de médias para a separabilidade entre as classes e a segunda a contribuição das matrizes de covariâncias. Note que quando as matrizes de covariância para as duas classes são iguais então

$$B = \frac{1}{8}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \tag{3.40}$$

que é distância de Mahalanobis entre duas classes.

3.9.3 Casos especiais

O método proposto para fins de extração de comprimentos de ondas, implementa um critério que é otimizar a separação entre duas classes usando a distância de Bhattacharyya. A otimização da distância de Bhattacharyya, entretanto, não é tarefa trivial por envolver o traço e determinante de matrizes [23]. Conforme [15], são considerados casos sub-ótimos de otimização dessa distância.

1. Otimização unicamente em função das matrizes de covariâncias. Neste caso assume-se igualdade entre os vetores de médias ($\boldsymbol{\mu}_i = \boldsymbol{\mu}_j$);

2. Otimização unicamente dos vetores de médias. Neste caso assume-se a igualdade entre as matrizes de covariâncias ($\boldsymbol{\Sigma}_i = \boldsymbol{\Sigma}_j$);

3. Otimização conjunta, em função dos vetores de médias e matrizes de covariâncias, com um peso maior à diferença entre os vetores de médias;

4. Otimização conjunta, em função dos vetores de médias e matrizes de covariâncias, com um peso maior à diferença entre as matrizes de covariâncias.

Uma descrição completa destes casos pode ser encontrada em [15].

3.10 O pacote MCLUST

MCLUST é um pacote desenvolvido para o software R para modelagem de mistura de normais, agrupamentos baseada em distribuições normais, estimação de densidades e análise de discriminantes. Nesse pacote foram implementados algoritmos hierárquicos para agrupamento de normais parametrizadas e o algoritmo EM para estimação dos parâmetros da mistura de normais multivariadas parametrizadas pelas matrizes de covariâncias descritas acima. No pacote MCLUST também são incluídas funções que combinam agrupamentos hierárquicos baseados em distribuições, o algoritmo EM e o critério de informação bayesiano (*Bayes information criterion* - BIC) usando as estratégias para agrupamentos, estimação de densidades e análise de discriminantes desenvolvidas por Raftery e Fraley [12].

O MCLUST também possui poderosas funcionalidades relacionadas a visualização dos resultados obtidos no agrupamento e classificação, além de funções relacionadas a visualização gráfica. Resumindo o pacote MCLUST possui as seguintes características:

- EM para quatro matrizes de covariâncias diagonais em modelos de mistura;
- Estimação de densidade via mistura de normais parametrizadas;
- Simulação a partir de mistura de normais;
- Análise de discriminante via MclustDA;

- Métodos para dados unidimensionais;
- Métodos visuais avançados, incluindo gráficos de incertezas e projeções aleatórias.

O foco do nosso trabalho é análise de discriminantes via mistura de normais, portanto na próxima seção será dado um exemplo envolvendo as funções relacionadas à análise de discriminante.

3.10.1 Exemplo

Esta seção procurou explorar a potencialidade das funções relacionadas à análise de discriminante via mistura de normais disponíveis no pacote MCLUST. As funções que serão utilizadas são: mclustDA, mclustDAtrain, mclustDAtest. Para maiores informações o leitor interessado deve consultar [13, 14].

O banco de dados utilizado para a o exemplo é o banco iris. O banco consiste em amostras de tamanho 50 de três espécies da flor *Iris* (*Iris setosa*, *Iris virginica* e *Iris versicolor*). Para cada flor foram medidas quatro características, sendo essas o comprimento e largura das pétalas e sépalas. Esse banco de dados foi apresentado por Fisher em 1936. O banco já está implementado no software R e para carregá-lo os seguintes comandos são requeridos:

```
> data(iris)
> Iris <- iris[,-5]
> require(mclust)
```

Na segunda linha foi eliminada a coluna que fornece as classes e na terceira o pacote mclust foi carregado para uso. Para selecionarmos a amostra de treinamento e amostra de teste podemos, por exemplo, utilizar os seguintes comandos:

```
> odd <- seq(from=1, to=nrow(iris), by=2)
> even <- odd+1
> train <- Iris[odd,]
> test <- Iris[even,]
```

Para a composição da amostra de treinamento foram utilizadas as observações ímpares e para a amostra de teste foram utilizadas as observações pares. Agora, podemos realizar o discriminante através da função mclustDA ou através das funções mclustDAtrain e mclustDAtest. Tais análises serão separadas em duas seções.

Análise via mclustDA

Para essa função as amostras de treinamento e teste são dadas conjuntamente com suas classes. A saída do mclustDA inclui o modelo de mistura para cada classe na amostra de treinamento, a classificação da amostra de treinamento e teste baseada no modelo estimado, as probabilidades a *posteriori* para a amostra de treino e as taxas de erro de classificação para ambas as amostras.

```
> irisMclustDA <- mclustDA(train = list(data = train, labels = iris[odd,5]),
test = list(data = test, labels = iris[even,5]))

> irisMclustDA
```

Usando o argumento train na função mclustDA colocamos as informações referentes a amostra de treinamento, que são a base de dados onde estão armazenados as medidas relacionadas a amostra de treinamento e também as classificações para essas observações para que a taxa de erro de classificação da amostra de treinamento possa ser calculada. Similarmente, no argumento test colocamos as informações referentes à amostra de teste.

```
Modeling Summary:
           trainClass mclustModel numGroups
```

```
setosa         setosa          VEI         2
versicolor versicolor          EEV         2
virginica   virginica          XXX         1

Test Classification Summary:
    setosa versicolor  virginica
        25         24         26

Training Classification Summary:
    setosa versicolor  virginica
        25         25         25

Training Error: 0

Test Error: 0.01333333
```

Observando os resultados gerados pela função temos que a taxa erro de classificação foi de 0% para a amostra treino e de 1,3333% para a amostra teste. Os modelos estimados foram VEI com duas componentes para a espécie *Iris setosa*, EEV com duas componentes para a espécie *Iris versicolor* e XXX com uma componente para a espécie *Iris virginica* ver tabela 3.1. O modelo XXX é uma parametrização para a matriz de covariâncias quando existe apenas uma componente de densidade, esse parametrização indica que o modelo é elipsoidal [14].

Na amostra de treinamento 25 observações foram classificadas na espécie *Iris setosa*, 25 na espécie *Iris versicolor* e 25 na espécie *Iris virginica*. Para a amostra de teste 25 observações foram classificadas na espécie *Iris setosa*, 24 na espécie *Iris versicolor* e 26 na espécie *Iris virginica*.

Um comando que nos permite visualizar alguns outros resultados é o seguinte:

```
names(irisMclustDA)
```

o qual nos fornece o seguinte resumo de objetos componentes

```
[1] "test"    "train"   "summary"
```

Para que possamos visualizar a classificação resultante, a probabilidade de incerteza e a classificação a *priori* para a amostra de teste e treinamento, basta escrever irisMclustDA$test e irisMclustDA$train respectivamente. Ao escrever irisMclustDA$summary teremos o modelo estimado para cada uma das espécies com o respectivo número de componentes estimado para cada espécie, conforme o resultado gerado ainda pouco.

Podemos visualizar a classificação resultante da análise de discriminante através do seguinte comando

```
> plot(irisMclustDA, trainData = train, testData = test)
```

Os resultados estão ilustrados na figura 3.1.

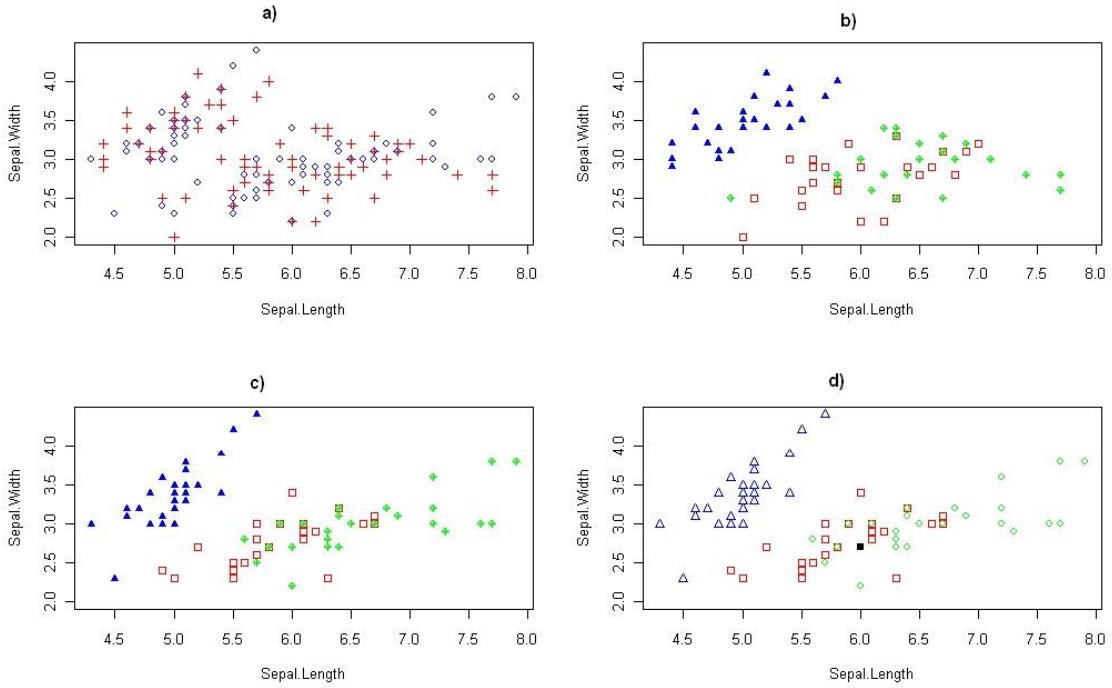

Figura 3.1: Gráficos associados com a função mclustDA sobre o banco de dados iris. a): amostra de treinamento (números ímpares / círculos) e amostra de teste (números pares / cruz). b): amostra de treinamento com as classificações conhecidas. c): classificação resutante da função mclustDA para a amostra de teste. d): classificação errada (símbolo preenchido) ao usar o modelo gerado pela função mclustDA para classificar a amostra de teste.

Análise via mclustDAtrain e mclustDAtest

Podemos fazer essa análise de outra maneira através das funções mclustDAtrain e mclustDAtest. Diferentemente da função mclustDA que realiza as etapas de treinamento e teste simutaneamente, essas funções as realizam de forma separada. Através da função mclustDAtrain utilizamos a amostra de treinamento para modelar a mistura para as diferentes classes e construir a regra de classificação. Após essa etapa usamos a função mclustDAtest para classificar as observações pertencentes a amostra de teste.

As saídas da função mclustDAtrain incluem os modelos estimados e os respectivos números de componentes, a classificação da amostra de treino e a taxa de erro de classificação. As saídas da função mclustDAtest incluem as mesmas saídas da função mclustDAtrain exceto os modelos estimados. Voltando para o exemplo os modelos estimados para cada classe são obtidos através dos seguintes comandos:

```
> irisTrain <- mclustDAtrain(data = train,
+ labels = iris[odd,5])
VEI EEV XXX
  2   2   1
```

que são os mesmos modelos obtidos pela função mclustDA. Agora com os seguintes comandos obtemos as classificações para a amostra teste e a probabilidade a *posteriori* de cada observação da amostra de teste pertencer a cada uma das espécies:

```
> irisTest <- mclustDAtest(models=irisTrain, data=test)
> names(summary(faithfulEvenTest))
[1] "classification" "z"
```

Para obtermos as taxas de erros para as amostras de treino e teste, são usados os seguintes comandos respectivamente:

```
> irisOddTest <- mclustDAtest(models=irisTrain, data=train)
> classError(summary(irisOddTest)$classification,
+ iris[odd,5])$errorRate
[1] 0
> classError(summary(irisTest)$classification,
+ iris[even,5])$errorRate
[1] 0.01333333
```

Observe que as taxas de erros são as mesmas obtidas pela função mclustDA. Para verificarmos os resultados gerados pelas funções utilizadas nestas duas seções podemos usar a função names. Outra vantagem de se usar a função classError é que podemos imprimir quais observações foram erroneamente classificadas na amostra de teste ou na amostra de treinamento, por exemplo:

```
> classError(summary(irisTest)$classification,
+ iris[even,5])$misclassified
[1] 42
```

Ou seja, na amostra de teste apenas a observação 42 foi classificada de forma errada.

3.10.2 Outros softwares disponíveis

Outros softwares que disponibilizam procedimentos relacionados a modelagem por mistura de distribuição são o SAS e o JMP produzidos pelo SAS Institute, Inc.. O SAS em versão 9.3, lançou o *FMM procedure* que ainda está em fase experimental [28]. O procedimento FMM foi criado para modelagem de dados, os quais a distibuição da variável resposta é uma mistura de distribuições univariadas. Algumas aplicações em que podemos utilizar o procedimento FMM para realizar as análises são [28]:

- Estimação de densidades multimodais ou com caldas pesadas;
- Ajustar modelos inflacionados de zeros para dados de contagem;
- Modelar dados com superdispersão;
- Ajustar modelos de regressão com distribuição dos erros complexa;
- Classificar observações baseados nas componentes de probabilidades preditas;
- Entre outras.

Através do procedimento FMM é possível ajustar misturas finitas de modelos de regressão ou mistura finita de modelos lineares generalizados em que a estrutura de regressão e as covariáveis são as mesmas em cada componente ou diferentes. O procedimento FMM permite o ajuste dos parâmetros dos modelos utilizando máxima verossimilhança ou métodos bayesianos. Vale ressaltar que esse procedimento não foi utilizado nesse trabalho porque a versão do SAS 9.3 não estava disponível para a UnB até a presente data. Alguns resultados utilizando esse procedimento foram obtidos no estágio 1, mas devido ao problema da licença decidimos retirar essas análises do trabalho. Portanto no presente trabalho foi utilizado apenas o pacote MCLUST do software R.

Capítulo 4

Análise preliminar dos dados

Como descrito na introdução após obtenção das amostras, 540 grãos danificados (defeituosos) e 540 grãos sadios (intactos) foram selecionados por análise visual. Tais grãos foram armazenados em recipientes plásticos enumerados, de forma a garantir a individualidade dos mesmos durante a realização das análises.

Seguindo uma norma específica, os grãos defeituosos foram classificados em quatro categorias [26]: grãos danificados por insetos, grãos quebrados, grãos com defeitos graves e grãos com defeitos gerais. A descrição de como são constituídos esses grãos podem ser encontrados em [26, pág. 59].

Em seguida os grãos selecionados foram encaminhados para um laboratório para a obtenção dos dados espectrais. Conforme descrito em [26], os grãos de café foram manualmente colocados com a face plana voltada para baixo, no ponto de bifurcação do espectrômetro e da fonte de luz (Figura 4.1), em uma plataforma com 15 mm de diâmetro, localizada a 12 mm acima das fibras de refletância e de iluminação. O feixe de luz era de 7 mm e o feixe de refletância de 2 mm considerando o diâmetro da plataforma de leitura. O equipamento apresentava um dispositivo de interface com o computador e as informações espectrais obtidas foram automaticamente ar-

mazenadas para análises posteriores (Figura 4.2). Foram efetuadas 15 leituras e o espectrômetro armazenou a média.

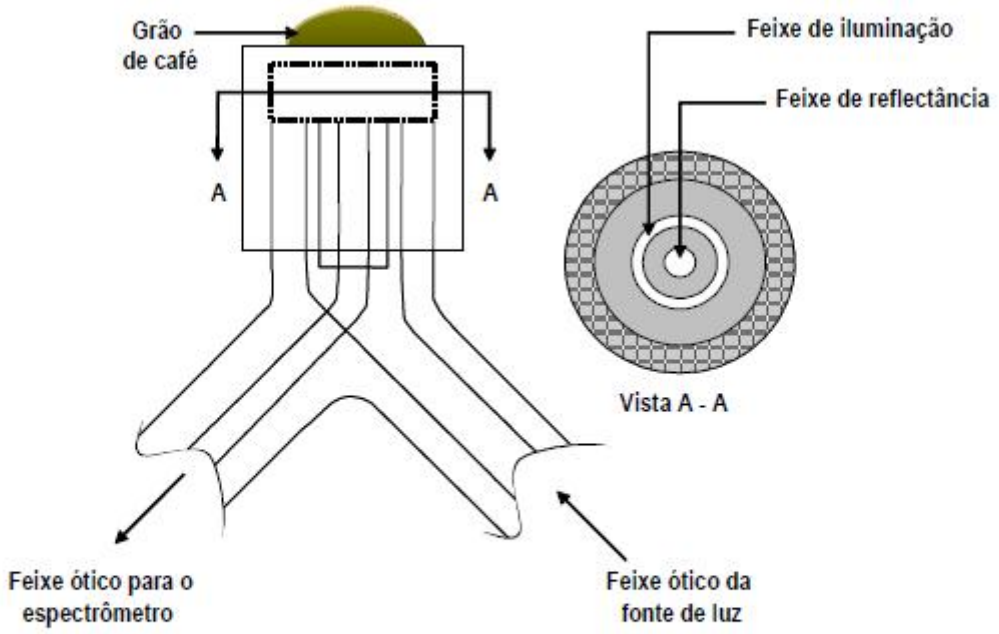

Figura 4.1: Esquema de funcionamento do espectrômetro [26]

Figura 4.2: Espectrômetro de infra-vermelho próximo utilizado na obtenção dos dados [26]

As informações obtidas na faixa de resolução de 500 a 1700 nm foram interpoladas a cada 5 nm resultando em 241 valores de absorvância para cada grão. Portanto, nosso banco de dados é composto por 1080 grãos de café, dos quais 540 são intactos e 540 são defeituosos. A informação captada para cada grão foi a quantidade de luz absorvida, em 241 comprimentos de onda diferentes.

Considerando cada grão como uma observação, os comprimentos de onda como variáveis e tendo a informação a priori da classe a qual o grão pertence, foi utilizado análise de discriminante via mistura de finita de distribuições para realizar a classificação dos grãos em intactos e defeituosos. Como será visto neste capítulo, não há necessidade de utilizar todos os comprimentos de onda como variáveis devido a grande correlação entre eles, este fato será explorado na próxima seção. Portanto, será realizada uma redução na dimensão do banco de dados usando um critério de separabilidade baseada na distância de Bhattacharyya a fim de identificar quais os comprimentos que maximizam a diferença entre as absorvâncias dos grãos intactos e defeituosos. Trabalhar com os dados reduzidos tem como finalidade melhorar a estimação dos parâmetros dos modelos de mistura de normais multivariadas em cada

classe e assim minimizar a taxa de erro de classificação, obtendo assim um bom discriminador.

4.1 Redução de dimensão baseado na distância de Bhattacharyya

Esse trabalho tem como objetivo fazer o discriminante e a classificação dos grãos intactos e defeituosos. Para obter uma melhor regra de classificação se faz necessário trabalhar apenas com os comprimentos de ondas que tenham um alto poder discriminatório. Nota-se na figura 4.3 uma alta correlação entre comprimentos de ondas adjacentes, isso implica que a absorvância captada por comprimentos de ondas adjacentes são praticamente as mesmas, logo isso serve como motivação para se trabalhar com um conjunto reduzido de comprimentos de ondas.

Agora observando a figura 4.4 [26], podemos notar que existe uma faixa de comprimentos de ondas onde há uma diferença visível do comportamento espectral dos grãos intactos e defeituosos, observados pelos valores médios da absorvância.

Juntando essas duas análises, podemos reduzir a dimensão dos dados utilizando um critério de separabilidade. Nesse trabalho foi utilizada a distância de Bhattacharyya, pois na literatura essa distância vem sendo muito usada para reduzir dimensões de dados relacionados a imagens obtidas por satélites [4, 23, 26], além de que em sua versão gaussiana essa separação das classes leva em consideração os vetores de médias, o que é de interesse conforme a figura 4.4. A abordagem aqui é um pouco diferente da utilizada nos trabalhos citados.

Na abordagem desse trabalho estamos interessados nas variáveis originais e não

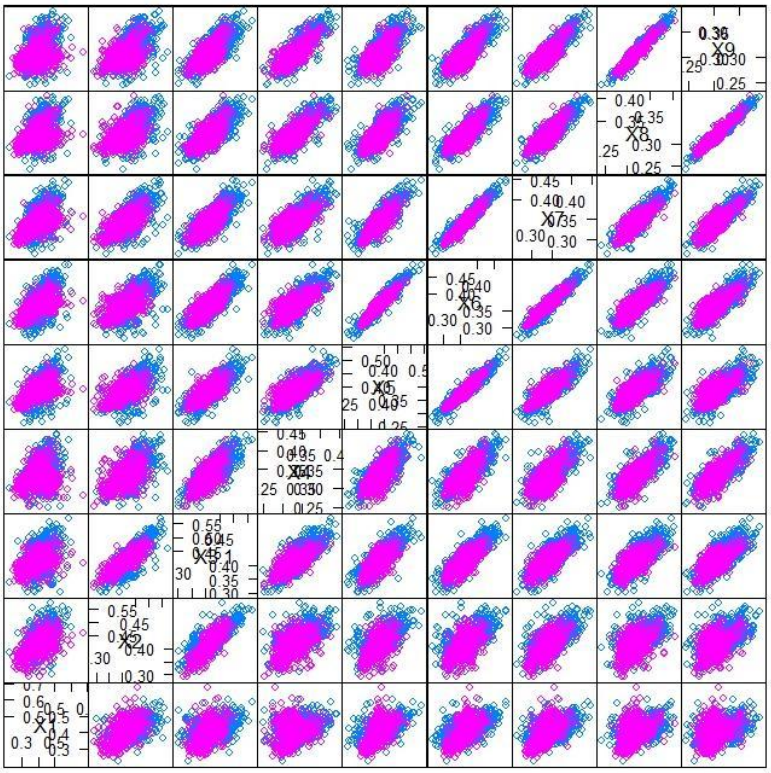

Figura 4.3: Gráficos de dispersão para um conjunto reduzido de comprimentos de onda separado por classe de grão.

em combinações lineares dessas variáveis, portanto a redução se deu da seguinte forma:

1. Observando a figura 4.4 podemos ver que uma diferença significativa entre as absorvâncias médias dos grãos intactos e defeituosos, as curvas são muito parecidas portanto é razoável supor que a distribuição em cada classe se diferem apenas por locação. Logo faremos a suposição de que as matrizes de covariâncias são idênticas com $\mathbf{\Sigma} = \frac{\mathbf{\Sigma}_1 + \mathbf{\Sigma}_2}{2}$;

2. Será calculada a distância de Bhattacharyya usando a fórmula (3.40);

3. Para se obter a contribuição marginal de cada comprimento de onda na distância, foi utilizada apenas os componentes situados na diagonal de Σ^{-1};

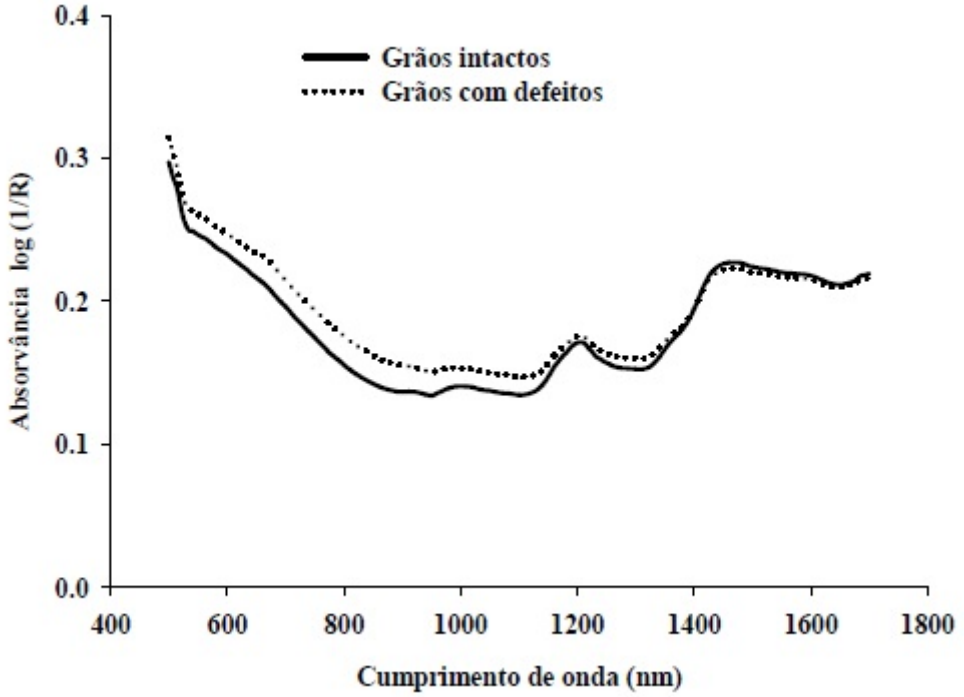

Figura 4.4: Curvas espectrais dos grãos de café intactos e com defeitos [26]

4. Para se obter os m comprimentos de ondas oferecendo a maior separação entre as duas classes ($m < 241$) no espaço original X, serão tomados as m maiores parcelas da distância, ou seja, as m maiores parcelas do seguinte somatório

$$S = \sum_{i=1}^{n} v_{ii}(\mu_{int,i} - \mu_{def,i})^2, \tag{4.1}$$

onde v_{ii} é o i-ésimo elemento da diagonal da matriz Σ^{-1}, $\mu_{int,i}$ é o valor médio da absorvância no i-ésimo comprimento de onda na classe dos grãos intactos e $\mu_{def,i}$ é o valor médio da absorvância no i-ésimo comprimento de onda na classe dos grãos defeituosos.

5. Para se ter uma idéia da contribuição de cada parcela no somatório (4.1), foi feita a razão entre cada parcela e a quantidade S. Em seguida, essa razão foi multiplicada por 100 e assim obtemos a contribuição em porcentagem.

Para a estimação de $\mathbf{\Sigma}^{-1}$ foi utilizado a função qr.solve() encontrado no pacote base do software R, que calcula a inversa de uma matriz utilizando a decomposição QR, essa função foi utilizada pois podemos alterar a tolerância na detecção de dependência linear entre as variáveis, sendo assim temos um cálculo mais preciso para a matriz inversa. A figura 4.5 mostra um gráfico com a contribuição (em porcentagem) de cada comprimento de onda na distância de Bhattacharyya. Com isso surge a dificuldade em se definir um ponto de corte k para a seleção dos comprimentos de ondas.

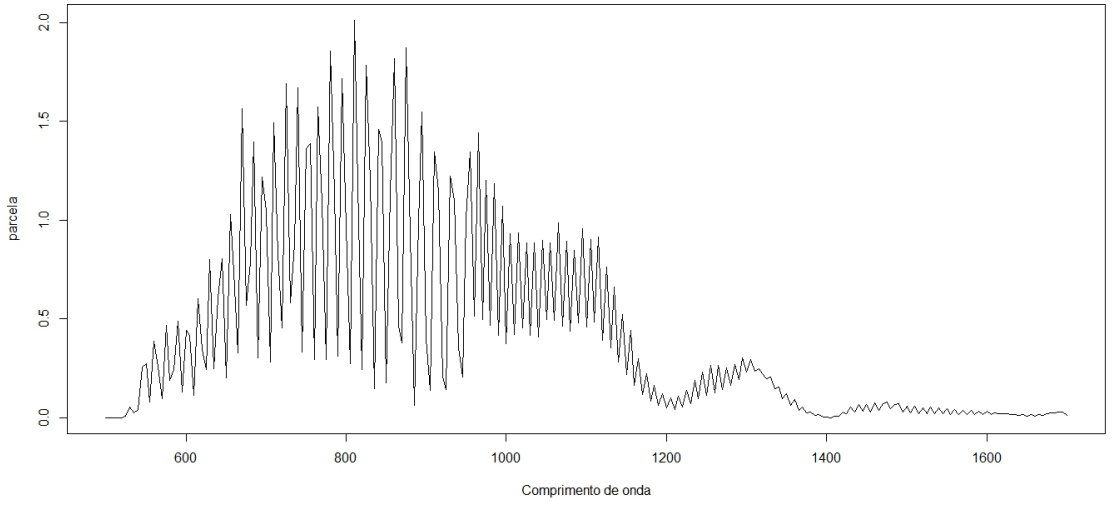

Figura 4.5: Contribuição marginal de cada comprimento de onda na distância de Bhattacharyya

Para contornar essa dificuldade foi criado um pequeno algoritmo para selecionarmos os comprimentos de ondas de forma a minimizar o erro de classificação. O algoritmo pode ser descrito através das seguintes etapas:

1. Dar valores arbitrários entre 0 e o valor máximo das parcelas (em porcentagem) para k;

2. Selecionar as parcelas que possuem valores maiores que cada um dos valores de k;

3. Para cada conjunto reduzido de dados calcular o erro de classificação utilizando mistura de distribuições conforme a seção 4.8;

O algoritmo retornará uma matriz, na qual a primeira coluna consiste nos valores de k. A segunda coluna consiste no erro de classificação da amostra de teste para cada conjunto reduzido de dados. E por fim, a terceira coluna consiste no número de comprimentos de ondas selecionados.

Sendo assim seleciona-se o valor de k que possui o menor erro de classificação, mas há casos em que para diferentes valores de k o erro de classificação é o mesmo, sendo assim dentre esses valores seleciona-se aquele mais parcimonioso, ou seja, o que possui menos comprimentos de ondas. Para ilustração a próxima seção mostrará a aplicação de desse algoritmo no conjunto de dados.

4.2 Aplicação do algoritmo de redução

Para o nosso conjunto de dados os valores das parcelas variam entre 0 e 2.01 aproximadamente, com isso para o algoritmo variamos os valores de k entre 0 e 2 por 0.1. O resultado do algoritmo está ilustrado na tabela 4.1 e na figura 4.6.

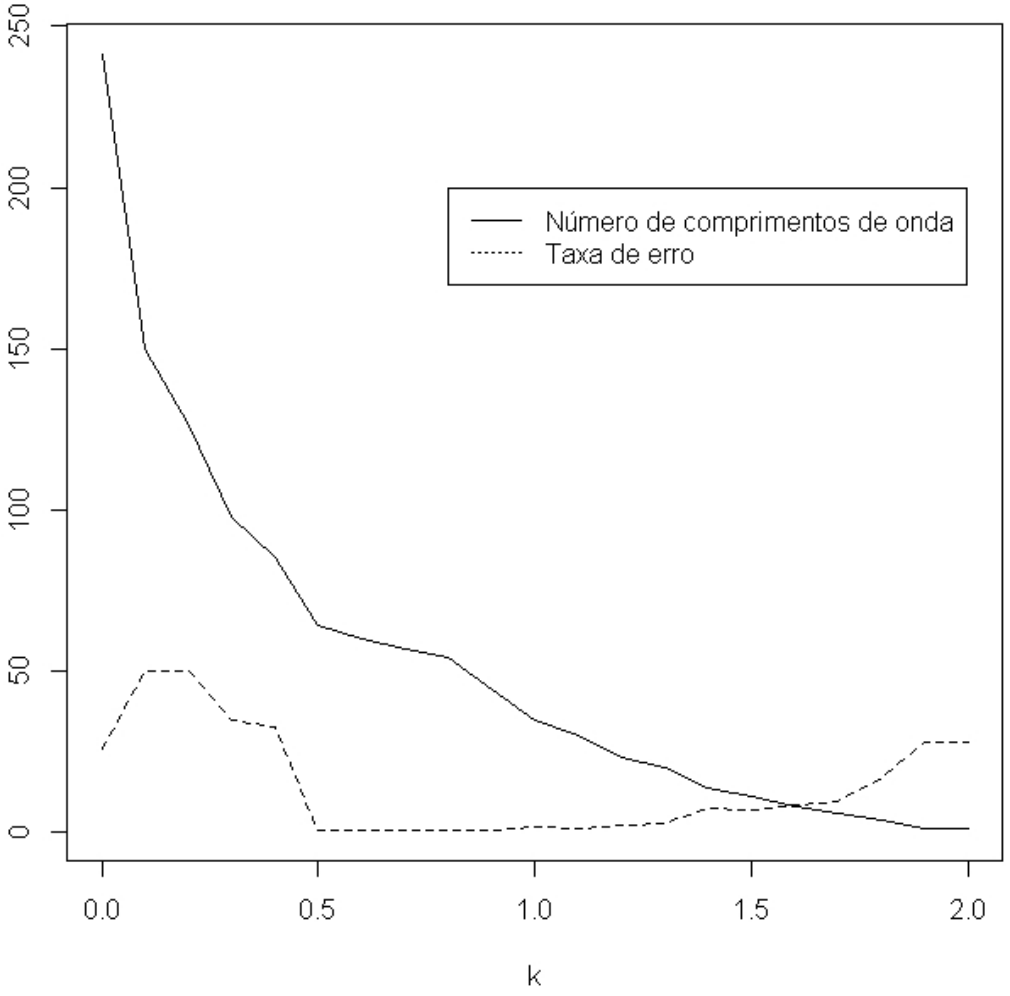

Figura 4.6: Gráfico cruzado - caso binário

Observa-se que para o valor de corte $0,6$ a taxa de erro de classificação foi de $0,5555556\%$. Com isso a base passou de 241 comprimentos de onda para 60. A tabela 4.2 mostra quais os comprimentos de onda que foram selecionados e suas respectivas contribuições (em porcentagem) na distância de Bhattacharyya.

k	Erro	Número de comprimentos de onda selecionados
0.0	0.261111111	241
0.1	0.500000000	150
0.2	0.500000000	126
0.3	0.348148148	98
0.4	0.325925926	85
0.5	0.007407407	64
0.6	**0.005555556**	**60**
0.7	0.007407407	57
0.8	0.007407407	54
0.9	0.007407407	44
1.0	0.014814815	35
1.1	0.012962963	30
1.2	0.024074074	23
1.3	0.029629630	20
1.4	0.074074074	14
1.5	0.070370370	11
1.6	0.085185185	8
1.7	0.096296296	6
1.8	0.168518519	4
1.9	0.279629630	1
2.0	0.279629630	1

Tabela 4.1: Resultado do algoritmo de redução

Podemos notar que os comprimentos que foram selecionados estão situados na faixa em que os níveis de absorvância são nitidamente diferentes para os grãos intactos e defeituosos e a seleção se deu de forma sistemática. O que é interessante pelo fato de que em nenhum momento utilizamos análise visual para a seleção dos comprimentos com maior poder discriminatório.

Comprimento de onda	Parcela (%)	Comprimento de onda	Parcela (%)
615	0.6012324	855	1.1351646
630	0.8019258	860	1.8199125
640	0.6052555	875	1.8750660
645	0.8055759	880	0.9251562
655	1.0289361	890	0.9888831
660	0.7386886	895	1.5468212
670	1.5659385	910	1.3447487
680	0.7861472	915	1.1455316
685	1.3953630	930	1.2215284
695	1.2189614	935	1.1010153
700	1.0450548	950	1.0062749
710	1.4948442	955	1.3464981
715	0.8316387	965	1.4418803
725	1.6915373	975	1.2027886
735	0.8815637	985	1.1832540
740	1.6706941	995	1.0707220
750	1.3636520	1005	0.9296013
755	1.3863977	1015	0.9362338
765	1.5722187	1025	0.8850275
770	1.0872411	1035	0.8866401
780	1.8574051	1045	0.8971600
785	1.1332901	1055	0.8846892
795	1.7179672	1065	0.9848014
800	0.9660773	1075	0.8937952
810	2.0121184	1085	0.8471138
815	1.1789998	1095	0.9562019
825	1.7833703	1105	0.9039029
830	1.1978879	1115	0.9144568
840	1.4599384	1125	0.7648139
845	1.3947614	1135	0.6597154

Tabela 4.2: Comprimentos de onda selecionados pelo algoritmo de redução

Capítulo 5

Resultados

Para discriminar os grãos em intactos ou defeituosos, utilizamos mistura de distribuições normais multivariadas. As justificativas para a utilização dessa densidade são: a efetividade em situações práticas conforme mostrado nos trabalhos [14], a existência de um pacote bem implementado e bem documentado para realizar discriminante via mistura de distribuições normais multivariadas (pacote MCLUST), além da extensa literatura explorando as propriedades desse caso particular de mistura, dando um suporte maior para a aplicação dessa técnica. Portanto as misturas descritas em [3] não foram consideradas nesse trabalho.

Essa técnica também foi utilizada para realizar a classificação dos grãos para o caso em que os grãos defeituosos foram separados em categorias (Danificados por insetos, quebrados, defeitos gerais e defeitos graves). Os resultados aqui gerados se mostraram altamente satisfatórios para a classificação binária (intactos e defeituosos) e os resultados foram satisfatórios para o caso em que mais categorias de defeituosos foram consideradas.

5.1 Análise dos dados espectrais

Utilizou-se a base de dados reduzida para realizar a classificação dos grãos como intactos ou defeituosos. Para realizar a classificação foi utilizada a função MclustDA localizada no pacote mclust do software R e ilustrada no capítulo 4.

Para a constituição da amostra de treinamento foram utilizadas as observações ímpares, portanto 540 grãos foram utilizados na fase de treinamento. Para a fase de teste foram utilizados os demais grãos.

Os resultados gerados pelo MclustDA estão resumidos nas tabelas 5.1 e 5.2, e a classificação está ilustrada na tabela 5.3. A figura 5.1 mostra graficamente a classificação resultante da análise de discriminante.

Base	Grãos	Modelo	Número de componentes
Completa	Intactos	XXX	1
	Defeituosos	XXX	1
Reduzida	Intactos	XXX	1
	Defeituosos	XXX	1

Tabela 5.1: Resumo da modelagem

Base	Amostra	Taxa de erro de classificação
Completa	Treinamento	0,2888889
	Teste	0,275
Reduzida	Treinamento	0,001851852
	Teste	0,005555556

Tabela 5.2: Taxa de erro para a amostra de treinamento e de teste

Para a base completa o modelo utilizado para a parametrização das matrizes de covariâncias para os grãos intactos e defeituosos foi o XXX, e o número de componentes da mistura foi igual a 1 para cada classe. Para a base reduzida o modelo utilizado para a parametrização das matrizes de covariâncias para os grãos intactos

Número de Grãos separados por observação visual	Intactos	Defeituosos
Intactos - 270	**268**	2
	99,26	0,74
Defeituosos - 270	1	**269**
	0,37	**99,63**

Tabela 5.3: Resultado da classificação por análise discriminante via mistura de distribuição para os grãos de café intactos e defeituosos

e defeituosos foi o XXX com número de componentes da mistura igual a 1 para cada classe. O modelo XXX só aparece quando se ajusta apenas uma componente, e significa que o modelo é elipsoidal. Já o erro de classificação para a base completa a taxa de erro ficou próximo de 29% para a amostra de treinamento e 27,5% para a amostra de teste. Já para a base reduzida a taxa de erro foi de 0,1851852% para a amostra de treinamento e 0,5555556% para a amostra de treinamento o que é considerado um resultado altamente satisfatório. Portanto podemos ver que a redução contribuiu significativamente para a diminuição da taxa de erro de classificação.

Para verificar a eficácia do discriminante via mistura de distribuições, foram simulados 4 bases de dados. A primeira base consiste de 540 grãos de café, dos quais 25% dos grãos são defeituosos, a segunda base consiste de 540 grãos dos quais 10% são defeituosos, na terceira base 5% são defeituosos e por fim a última base consiste apenas de grãos intactos. Portanto para verificar a eficácia do método, os grãos foram classificados usando o modelo estimado na fase treinamento. Em seguida, foi calculada a proporção de grãos defeituosos em cada base e esse resultado foi

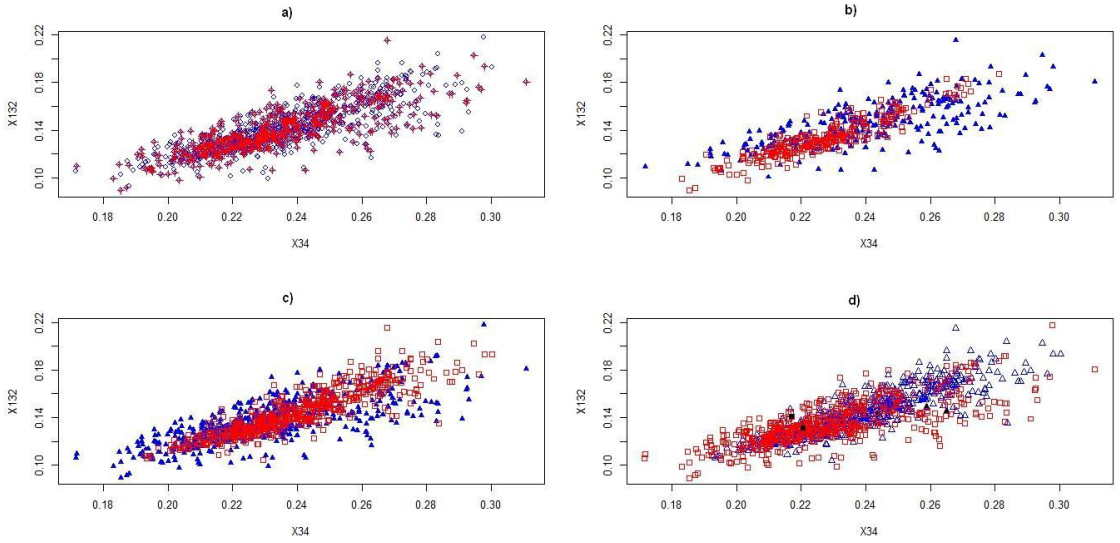

Figura 5.1: Gráficos associados com a funçao mclustDA sobre o banco de dados de café. a): amostra de treinamento e amostra de teste. b): amostra de treinamento com as classificações conhecidas. c): classificação resultante da função mclustDA para a amostra de teste. d): classificação errada (simbolo preenchido) ao usar o modelo gerado pela função mclustDA para classificar a amostra de teste

comparado com a proporção conhecida a priori, por fim foi calculada a porcentagem de grãos defeituosos que foram corretamente classificados para verificar se o modelo detectou de maneira eficaz os grãos defeituosos. Os resultados estão ilustrados na tabela 5.4.

Proporção de defeituosos verdadeira	Proporção de defeituosos após a classificação	Taxa de acerto para os grãos defeituosos
25%	25,37037%	100%
10%	10,18519%	98,15%
5%	5,37037%	100%
0%	0,3703704 %	99,63%

Tabela 5.4: Proporção de defeituosos calculados após a classificação

Observando as proporções geradas na tabela 5.4, podemos ver que estão próximas das proporções reais, com alta taxa de acerto.

Para o caso em que os grãos defeituosos foram separados em categorias con-

forme a gravidade dos defeitos, a amostra de treinamento foi constituída usando a metade das observações em cada categoria (Intactos, danificados por insetos, quebrados, defeitos gerais, defeitos graves), a amostra de teste foi constituída com as observações complementares. O algoritmo de redução foi utilizado novamente já que o número de classes aumentou, então um novo ponto de corte k foi calculado. O valor selecionado foi 1.2 e o número de comprimentos de onda selecionados foi 23. A taxa de erro ficou em aproximadamente 5,55 % para a amostra de treinamento, e aproximadamente 9,83% para a amostra de teste, o que é considerado altamente satisfatório. O resultado do algoritmo está ilustrado na tabela 5.5 e figura 5.2. Os comprimentos de ondas selecionados estão na tabela 5.6. Para a classificação foi utilizada função MclustDA, e os resultados estão ilustrados nas tabelas 5.7 e 5.8

k	Erro	Número de comprimentos de onda selecionados
0.0	0.37847866	241
0.1	0.44712430	150
0.2	0.41187384	126
0.3	0.39332096	98
0.4	0.36549165	85
0.5	0.20408163	64
0.6	0.18367347	60
0.7	0.17439703	57
0.8	0.16697588	54
0.9	0.14842301	44
1.0	0.13914657	35
1.1	0.14285714	30
1.2	**0.09833024**	**23**
1.3	0.11317254	20
1.4	0.13358071	14
1.5	0.15027829	11
1.6	0.15955473	8
1.7	0.16512059	6
1.8	0.31910946	4
1.9	0.44155844	1
2.0	0.44155844	1

Tabela 5.5: Resultado do algoritmo de redução

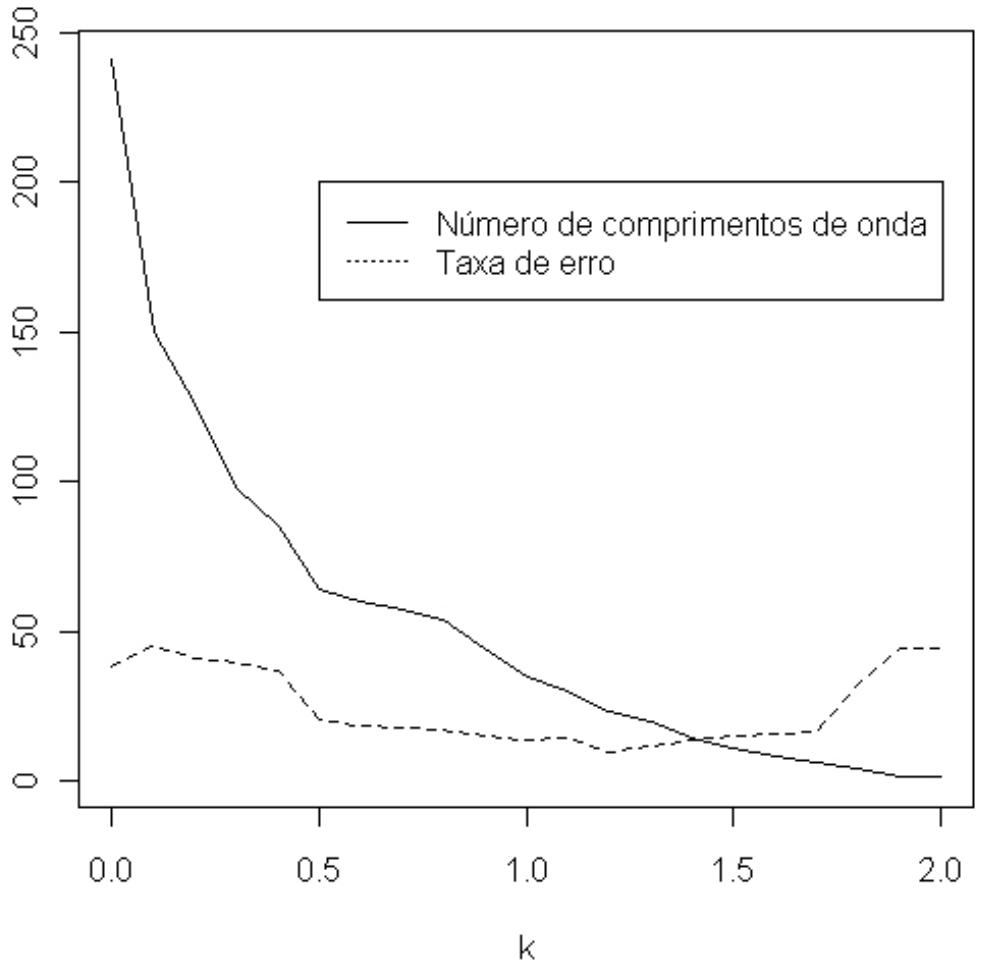

Figura 5.2: Gráfico cruzado

Observa-se que as taxas de acertos são altas para os grãos intactos (98,52%). Para as categorias de defeitos, a taxa de acerto é alta para os grãos quebrados (90,71%), para os grãos com defeitos gerais (100%) e é satisfatória para os grãos com defeitos graves (73,26%). Em contrapartida as taxas são ruins para os grãos danificados por insetos, pois nenhum dos grãos foi classificado de maneira correta.

Comprimento de onda	Parcela (%)	Comprimento de onda	Parcela (%)
670	1.565938	825	1.783370
685	1.395363	840	1.459938
695	1.218961	845	1.394761
710	1.494844	860	1.819912
725	1.691537	875	1.875066
740	1.670694	895	1.546821
750	1.363652	910	1.344749
755	1.386398	930	1.221528
765	1.572219	955	1.346498
780	1.857405	965	1.441880
795	1.717967	975	1.202789
810	2.012118		

Tabela 5.6: Comprimentos de onda selecionados pelo algoritmo de redução

Grãos	Modelo	Número de componentes
Intactos	EEE	9
Danificados por insetos	EII	9
Quebrados	XXX	1
Defeitos gerais	EEE	7
Defeitos graves	XXX	1

Tabela 5.7: Resumo da modelagem para os grãos defeituosos separados em categorias

Número de Grãos separados por observação visual	Número de observações e porcentagem classificadas				
	Intactos	Danificados por insetos	Quebrados	Defeitos gerais	Defeitos graves
Intactos - 270	**266**	0	4	0	0
	98,52	0	1,48	0	0
Danificados por insetos - 13	0	**0**	9	0	4
	0	**0**	69,23	0	30,77
Quebrados - 140	11	0	**127**	0	2
	7,86	0	**90,71**	0	1,43
Defeitos gerais - 30	0	0	0	**30**	0
	0	0	0	**100**	0
Defeitos graves - 86	0	0	23	0	**63**
	0	0	26,74	0	**73,26**

Tabela 5.8: Resultado da classificação por análise discriminante via mistura de distribuições para os grãos de café intactos e defeituosos, separados por categorias de defeitos

Capítulo 6

Considerações finais

Observando os resultados gerados na seção anterior, fica claro que utilizar todo o banco de dados para fazer a classificação dos grãos em intactos ou defeituosos é desnecessário. A justificativa se encontra nos gráficos de dispersão e no gráfico das curvas espectrais, nessa fase concluímos que comprimentos de onda adjacentes são altamente correlacionados e assim há comprimentos de onda que captam medidas redundantes. No gráfico das curvas espectrais podemos ver que existe uma faixa de comprimentos, na qual as absorvâncias médias medidas são nitidamente diferentes o que reforça a hipótese de se trabalhar com um número reduzido de comprimentos de onda.

Para a seleção dos comprimentos de onda que possuem a informação suficiente para se construir um bom classificador, foi desenvolvido um pequeno algoritmo para a escolha do ponto de corte k que será usado para a seleção dos comprimentos de onda, conforme a seção 7.1. A outra justificativa para a redução é a estimação dos parâmetros dos modelos de mistura, pois trabalhando apenas com a informação necessária podemos obter estimativas melhores para cada parâmetro. Observando os comprimentos de onda selecionados para o caso em que classificamos os grãos

apenas em intactos ou defeituosos e para o caso em que os grãos defeituosos foram separados em categorias, estão localizadas na faixa onde existe uma diferença nítida nas medidas de absorvâncias entre os grãos intactos e defeituosos.

Esse fato fica evidente ao compararmos as taxas de erro de classificação para a base completa e para a base reduzida. Trabalhando com os 241 comprimentos de onda a taxa de erro de classificação para a amostra de teste foi de 27,5%, enquanto que para a base reduzida a taxa de erro ficou em torno de 0,5%. Confirmando assim a existência de informação redundante na base de dados.

Para verificar a eficácia do método, foram simuladas quatro bases de dados conforme descrito na seção anterior. Era conhecido a *priori* a quantidade de grãos defeituosos, portanto para verificar a eficácia do método esses grãos foram classificados utilizando o MclustDA e após essa classificação foi calculado a proporção de defeituosos. Podemos ver que as proporções a *posteriori* estão próximos da verdadeira e que a taxa de acerto na classificação é muito alta, conforme a terceira coluna da tabela 5.4.

Para o caso em que os grãos foram separados em categorias o resultado da classificação de uma forma geral se mostrou satisfatório, pois a taxa de erro ficou em torno de 9,83%. Mas olhando para a porcentagem de acerto dentro de cada categoria, vimos que para os grãos que foram danificados por insetos os resultados não são bons. A justificativa encontrada é que devido à pequena quantidade de grãos com essas características, temos pouca informação para a estimação eficiente dos parâmetros do modelo de mistura de normais para essa categoria. Sendo assim pos-

sivelmente o modelo estimado não é o mais adequado para representar a distribuição populacional dos grãos danificados por insetos, implicando assim em uma alta taxa de erro de classificação. Já para as categorias de grãos intactos e grãos quebrados as taxas de acerto foram bastante elevadas, para a categoria de grãos com defeitos gerais todas as observações foram classificadas de maneira correta e para os grãos com defeitos graves a taxa de acerto é razoável.

Portanto podemos ver que análise de discriminante via misturas de normais se mostrou uma técnica bastante eficaz na classificação dos grãos em intactos e defeituosos gerando um acerto de aproximadamente 99,44%, o que é muito elevado. Para o caso em que os grãos defeituosos foram separados em categorias a taxa de acerto ficou em torno de 90%, também uma elevada taxa de acerto. De uma forma geral os resultados produzidos por esse trabalho são melhores do que os resultados obtidos por [26], reforçando assim o fato de que análise de discriminante via mistura de normais é uma técnica extremamente eficaz para classificação dos grãos.

Outra conclusão que podemos tirar é que a técnica de Espectroscopia de infravermelho próximo é uma técnica eficaz para o reconhecimento de um grão intacto ou defeituoso, corroborando assim as conclusões de [25, 26, 27].

Nesse trabalho utilizamos mistura de normais para discriminar os grãos. Portanto, como sugestão de pesquisa, poderia ser utilizada outra componente de densidade como as descritas em [3], para realizar o discriminante, e os resultados podem ser comparados com mistura de normais para ver se há algum ganho expressivo não só em taxa de acerto, mas a maior justificativa para se usar essas densidades é que

tais modelos podem se ajustar melhor aos dados, isso implica em um melhoramento na taxa da classificação. Outro sugestão é trabalhar com as demais otimizações da distância de Bhattacharyya, já que nesse trabalho fizemos a suposição de que as matrizes de covariâncias são iguais. Isso para o caso em que existem apenas duas classes, para o caso com mais classes seria interessante utilizar uma extensão da distância de Bhattacharrya [15] para mais de duas classes, assim possivelmente os resultados poderiam ser melhores para o caso em que os grãos defeituosos são classificados em várias categorias.

Por fim outras medidas de separabilidade também podem ser usadas para maximizar a distância entre a classe dos grãos intactos e a classe dos grãos defeituosos.

Referências Bibliográficas

[1] AZZALINI, A. A class of distributions which includes the normal ones. **Scandinavian Journal of Statistics**, v. 12, p. 171-178, 1985.

[2] BANFIELD, J. D.; RAFTERY, A. E. Model-Based Gaussian and Non-Gaussian Clustering. **Biometrics**, v. 49, p. 803-821, 1993.

[3] BASSO, R. M.; LACHOS, V. H.; CABRAL, C. R. B.; GHOSH, P. Robust mixture modeling based on scale mixtures of skew-normal distributions. **Computational Statistics and Data Analysis**, v. 54, p. 2926-2941, 2010.

[4] BATISTA, M. H.; HAERTEL; V. Classificação hierárquica orientada a objeto em imagens de alta resolução espacial empregando atributos espaciais e espectrais. **Anais XIII Simpósio Brasileiro de Sensoriamento Remoto**, Florianópolis, Brasil, 21-26 abril 2007, INPE, p. 489-497.

[5] BAUDRY, JP.; RAFTERY, A.; CELEUX, G.; LO, K.; GOTTARDO, R. Combining Mixture Components for Clustering. **Journal of Computational and Graphical Statistics**, v. 9, p. 332-353, 2010.

[6] BIERNACKI C.; GOUVAERT G. Choosing Models in Model-based Clustering and Discriminant Analysis. **J. Statis. Comput. Simul.**, v. 64, p. 49-71, 1999.

[7] BIERNACKI C.; CELEUX, G.; GOUVAERT G. Assessing a mixture model for clustering with the integrated completed likelihood. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 22, p. 719-725, 2000.

[8] BILMES, J. A. A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. 1998.

[9] CELEUX, G.; GOVAERT, G. Gaussian Parsimonious Clustering Models. **Pattern Recognition**, v. 28, p. 781-793, 1995.

[10] DEMPSTER, A. P.; LAIRD, N. M.; RUBIN, D. B. Maximum likelihood from incomplete data via the em algorithm. **Journal of the Royal Statistical Society: Series B**, v. 39, p. 1-38, 1977.

[11] FRALEY, C.; RAFTERY, A. E. How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. **The Computer Journal**, v. 41, p. 578-588, 1998.

[12] FRALEY, C.; RAFTERY, A. E. Model-Based Clustering, Discriminant Analysis, and Density Estimation. **Journal of the American Statistical Association**, v. 97, p. 611631, 2002.

[13] FRALEY, C.; RAFTERY, A. E. (2003). Enhanced Model-Based Clustering, Density Estimation, and Discriminant Analysis Software: MCLUST. **Journal of Classification**, 20, 263-286.

[14] FRALEY, C.; RAFTERY, A. E. MCLUST Version 3 for R: Normal Mixture Modeling and Model-Based Clustering. **Technical Report**, University of Washington 2006.

[15] FUKUNAGA, K. **Introduction to Statistical Pattern Recognition**. 2nd ed. Boston: Academic Press. 1990.

[16] GUPTA, M. R.; CHEN, Y. Theory and Use of the EM Algorithm. **Foundations and Trends in Signal Processing**. v. 4, p. 223-296, 2010.

[17] HASTIE, T.; TIBSHIRANI, R. Discriminant Analysis by Gaussian Mixtures. **Journal of the Royal Statistical Society, Ser. B**, v. 58, p. 155-176, 1996.

[18] HORTA, M.M. **Modelos de misturas de distribuições na segmentação de imagens SAR polarimétricas multi-look**. PhD thesis, Universidade de São Paulo, 2009.

[19] JOHNSON, R.A.; WICHERN, D.W. **Applied Multivariate Statistical Analysis**. 6th ed. New Jersey: Pearson Prentice Hall, 2007.

[20] MCLACHLAN, G. J. **Discriminant Analysis and Statistical Pattern Recognition**. New York: John Wiley. 1992.

[21] MCLACHLAN, G. J.; KRISHNAN, T. **The EM Algorithm and Extensions**. 2nd ed. New York: John Wiley, 1996.

[22] MCLACHLAN, G. J.; PEEL, D. **Finite Mixture Models**. New York: John Wiley & Sons, 2000.

[23] MORAES, D. A. O. **Extração de feições em dados imagem com alta dimensão por otimização da distância de Bhattacharyya em um classificador de decisão em árvore**. 2005. 98f. Dissertação (Mestrado em Sensoriamento Remoto) - Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, 2005.

[24] MORAES, D. A. O.; HAERTEL; V. Métodos hierárquicos para redução de dimensões e classificação de imagens AVIRIS. **Anais XIII Simpósio Brasileiro de Sensoriamento Remoto**, Florianópolis, Brasil, 21-26 abril 2007, INPE, p. 6481-6488.

[25] NIGOSKI, S. **Espectroscopia no infravermelho próximo no estudo de características da madeira e papel de *Pinus taeda L***. 2005. 173f. Dissertação (Doutorado em Engenharia Florestal) - Universidade Federal do Paraná, Curitiba, PR, 2005.

[26] PARAZZI, F. C. **Incidência de fungos da pré-colheita ao armazenamento de café**. 2005. 82f. Dissertação (Doutorado em Engenharia Agricola) - Univerdade Federal de Viçosa, Viçosa, MG, 2005.

[27] SANTOS, A. P. **Espectroscopia de infravermelho próximo em análises de solos e plantas**. 2011. 76f. Dissertação (Mestrado em Agronomia) - Universidade Federal de uberlância, Uberlândia, MG, 2011.

[28] SAS Institute. SAS user's guide: Statistics, version 9.3, chapter 37. SAS Institute, Cary, NC. Todd and Browde, 2011.

[29] SCOTT, A. J.; SYMONS, M. J. Clustering Methods Based on Likelihood Ratio Criteria. **Biometrics**, v. 27, p. 387-397, 1971.

[30] SCOTT, D. W. **Multivariate Density Estimation**, New York: Wiley, 1992.

Capítulo 7

APÊNDICE - Programação R

7.1 Caso 1 - Caso Binário

```
####################################
#        LEITURA DOS DADOS         #
####################################

dados <- read.csv(file="J:\\cafe\\dados_cafe.csv", header=T,
stringsAsFactor=F, sep=";", dec=",")
dados1 <- dados[-1, -c(1,2,4)]
dados1 <- dados1[,-c(2:11)]
Dados1 <- dados1[,-1]

#Definição das classes

intact <- dados1[dados1$X3=="intact",-1]
damaged <- dados1[dados1$X3=="damaged",-1]

################################################
#    OTIMIZAÇÃO DA DISTÂNCIA DE BHATTACHARYYA    #
################################################

#Matrizes de covariâncias

cov1 <- cov(intact)
cov2 <- cov(damaged)
```

```
cov1 <- as.matrix(cov1)
cov2 <- as.matrix(cov2)
cov <-  0.5*(cov1+cov2)

#Vetores de médias

med1 <- colMeans(intact)
med2 <- colMeans(damaged)

#Parcelas da distância de Bhattacharyya

require(Matrix)
inv <-qr.solve(cov, tol=1e-17)
inv1 <- Diagonal(n=241, diag(inv))
m <- t(med2-med1)%*%inv1
n <- med2-med1

B <- as.vector(as.numeric())
for (i in 1:ncol(Dados1)){
B[i] <- 0.125*m[i]*n[i]
}

parcela <- 100*(B/sum(B))
i <- seq(2, 242, 1)
c <- seq(500,1700,5)
base <- as.data.frame(cbind(i, c, parcela))

#Gráfico com a contribuição marginal do comprimento de onda na distância

plot(c, parcela, type="l", main="Contribuição marginal do comprimento de onda
na distância de Bhattacharrya", xlab="Comprimento de onda", ylab="parcela")

#########################################################
#             ALGORITMO DE REDUÇÃO                      #
#########################################################

require(mclust) #Carregando o pacote mclust

reducao <- function(min, max, by){
```

```
odd <- c(seq(1, nrow(dados1), 2))
even <- c(odd+1)

corte <- seq(min, max, by)
erro <- as.vector(as.numeric())
nv <- as.vector(as.numeric())

for (k in 1:length(corte)){
i <- seq(2, 242, 1)
basefilt <- base[base$parcela>corte[k],]
nv[k] <- length(basefilt$i)
amostra1 <- dados1[,c(1,basefilt$i)]
train <- mclustDAtrain(amostra1[odd,-1], labels =
amostra1[odd,1], verbose=F) ## training step
test <- mclustDAtest(amostra1[even,-1], train) ## compute model densities
clEven <- summary(test)$class ## classify training set
erro[k] <- classError(clEven,amostra1[even,1])$errorRate

}

return(as.data.frame(cbind(corte, erro, nv)))

}

red <- reducao(0, 2, 0.1)

##Valor selecionado para o corte: 0.6
##60 comprimentos de ondas foram selecionados

##Gráfico Cruzado

plot(red$corte, red$nv, type="l", xlab="k", ylab=" ")
lines(red$corte, red$erro*100, lty="dashed")
legend(0.5, 200, legend=c("Número de comprimentos de onda",
"Taxa de erro"), lty=c(1, 3))

####################
#    REDUÇÃO       #
####################
```

```
basefilt <- base[base$parcela>0.6,]
amostra1 <- dados1[,c(1,basefilt$i)]

##Para se obter os m comprimentos de ondas oferecendo a
##maior separação entre as duas classes (m<241) no espaço original X,
##então podem ser tomados as m maiores parcelas da distância.


#####################################################
#  SELEÇÃO DA AMOSTRA DE TREINAMENTO E DE TESTE  #
#####################################################


odd <- c(seq(1, nrow(dados1), 2))
even <- c(odd+1)


##############################
# DISCRIMINANTE VIA MISTURA #
##############################

#Base completa

GRANMclustDAc <- mclustDA(train=list(data=dados1[odd,-1],labels=dados1[odd,1]),
test= list(data=dados1[,-1],labels=dados1[,1]))

GRANMclustDAc


#############################################################################

#Base Reduzida

GRANMclustDAr <- mclustDA(train=list(data=amostra1[odd,-1],labels=amostra1[odd,1]),
test= list(data=amostra1[even,-1],labels=amostra1[even,1]), verbose=F)

GRANMclustDAr

class <- GRANMclustDAr$test$classification
prop.def <- length(class[class=="damaged"])/length(class)
```

```
table(amostra1[even,1], class)

train <- amostra1[odd,-1]
train1 <- train[,c(1,57)]
test <- amostra1[,-1]
test1 <- test[,c(1,57)]

par(mfrow=c(2,2))
plot(GRANMclustDA, trainData=train1, testData=test1)


############################################################
# SIMULAÇÃO                                                #
############################################################

#Base com 25% de grãos defeituosos

intact1 <- dados1[dados1$X3=="intact",]
damaged1 <- dados1[dados1$X3=="damaged",]

def25 <- seq(1,540,4)
base25 <- rbind(intact1[-def25,],damaged1[def25,])
base25 <- base25[,c(1, basefilt$i)]

MclustDA25 <- mclustDA(train=list(data=amostra1[odd,-1],labels=amostra1[odd,1]),
test= list(data=base25[,-1], labels=base25[,1]), verbose=F)

class25 <- MclustDA25$test$classification
prop.def25 <- length(class25[class25=="damaged"])/length(class25)
table(base25[,1],class25)
#####################################################################



#Base com 10% de grãos defeituosos

intact1 <- dados1[dados1$X3=="intact",]
damaged1 <- dados1[dados1$X3=="damaged",]
```

```
def <- seq(1,540,10)
base1 <- rbind(intact1[-def,],damaged1[def,])
base10 <- base1[,c(1, basefilt$i)]

MclustDA10 <- mclustDA(train=list(data=amostra1[odd,-1],labels=amostra1[odd,1]),
test= list(data=base10[,-1]),
verbose = FALSE)

class10 <- MclustDA10$test$classification
prop.def10 <- length(class1[class10=="damaged"])/length(class10)
table(base10[,1],class10)
##############################################################################


#Base com 5% de grãos defeituosos

intact1 <- dados1[dados1$X3=="intact",]
damaged1 <- dados1[dados1$X3=="damaged",]

def5 <- seq(1,540,20)
base1 <- rbind(intact1[-def5,],damaged1[def5,])
base5 <- base1[,c(1, basefilt$i)]

MclustDA5 <- mclustDA(train=list(data=amostra1[odd,-1],labels=amostra1[odd,1]),
test= list(data=base5[,-1]),
verbose = TRUE)

class5 <- MclustDA5$test$classification
prop.def5 <- length(class5[class5=="damaged"])/length(class5)
table(base5[,1], class5)
##############################################################################


#Base com apenas grãos intactos

intact1 <- dados1[dados1$X3=="intact",]

base0 <- intact1[,c(1, basefilt$i)]

MclustDA0 <- mclustDA(train=list(data=amostra1[odd,-1],labels=amostra1[odd,1]),
```

```
test= list(data=base0[,-1]), verbose=FALSE)

class0 <- MclustDA0$test$classification
prop.def0 <- length(class0[class0=="damaged"])/length(class0)
table(base0[,1], class0)
##############################################################################
```

7.2 Caso 2 - Caso em que os defeituosos foram divididos em categorias

```
#####################################
#        LEITURA DOS DADOS          #
#####################################

dados <- read.csv(file="J:\\cafe\\dados_cafe.csv", header=T,
stringsAsFactor=F, sep=";", dec=",")
dados1 <- dados[-1, -c(1,2,3)]
dados1 <- dados1[,-c(2:11)]
Dados1 <- dados1[,-1]

#Definição das classes
intact <- dados1[dados1$X.2==0,-1]
damaged <- dados1[!dados1$X.2==0,-1]



################################################
#    OTIMIZAÇÃO DA DISTÂNCIA DE BHATTACHARYYA    #
################################################

#Matrizes de covariâncias

cov1 <- cov(intact)
cov2 <- cov(damaged)

cov1 <- as.matrix(cov1)
cov2 <- as.matrix(cov2)
cov <-  0.5*(cov1+cov2)
```

```
#Vetores de médias

med1 <- colMeans(intact)
med2 <- colMeans(damaged)

#Parcelas da distância de Bhattacharyya
require(Matrix)
inv <- Diagonal(n=241, diag(qr.solve(cov,tol=1e-17)))
m <- t(med2-med1)%*%inv
n <- med2-med1

B <- as.vector(as.numeric())
for (i in 1:ncol(Dados1)){
B[i] <- 0.125*m[i]*n[i]
}

parcela <- 100*(B/sum(B))
i <- seq(2, 242, 1)
c <- seq(500,1700,5)
base <- as.data.frame(cbind(i, c, parcela))



#########################################################
#  ALGORITMO REDUÇÃO PARA O CASO EM QUE OS GRÃO      #
#  DEFEITUOSOS FORAM DIVIDIDOS EM CATEGORIAS         #
#########################################################
require(mclust) #Carregando o pacote mclust

reducao1 <- function(min, max, by){
#Amostra de treinamento e teste

intact1 <- amostra1[amostra1$X.2==0,]
damaged1 <- amostra1[amostra1$X.2==1,]
damaged2 <- amostra1[amostra1$X.2==2,]
damaged3 <- amostra1[amostra1$X.2==3,]
damaged4 <- amostra1[amostra1$X.2==4,]

odd1 <- c(seq(1, nrow(intact1), 2))
odd2 <- c(seq(1, nrow(damaged1), 2))
```

```
odd3 <- c(seq(1, nrow(damaged2), 2))
odd4 <- c(seq(1, nrow(damaged3), 2))
odd5 <- c(seq(1, nrow(damaged4), 2))

corte <- seq(min, max, by)
erro <- as.vector(as.numeric())
nv <- as.vector(as.numeric())

for (k in 1:length(corte)){
i <- seq(2, 242, 1)
basefilt <- base[base$parcela>corte[k],]
nv[k] <- length(basefilt$i)
amostra1 <- dados1[,c(1,basefilt$i)]

intact1 <- amostra1[amostra1$X.2==0,]
damaged1 <- amostra1[amostra1$X.2==1,]
damaged2 <- amostra1[amostra1$X.2==2,]
damaged3 <- amostra1[amostra1$X.2==3,]
damaged4 <- amostra1[amostra1$X.2==4,]


train <- rbind(intact1[odd1,], damaged1[odd2,],
   damaged2[odd3,], damaged3[odd4,], damaged4[odd5,])
train <- na.exclude(train)
even <- rbind(intact1[-odd1,], damaged1[-odd2,],
   damaged2[-odd3,], damaged3[odd4,], damaged4[-odd5,])
even <- na.exclude(even)

train <- mclustDAtrain(train[,-1], labels = train[,1], verbose=F) ## training step
test <- mclustDAtest(even[,-1], train) ## compute model densities
clEven <- summary(test)$class ## classify training set
erro[k] <- classError(clEven,even[,1])$errorRate

}

return(as.data.frame(cbind(corte, erro, nv)))

}

red1 <- reducao1(0, 2, 0.1)
```

```
#ponto de corte escolhido 1.2


###################
#    REDUÇÃO      #
###################

basefilt <- base[base$parcela>1.2,]
amostra1 <- dados1[,c(1,basefilt$i)] #Base reduzida

##Para se obter os m comprimentos de ondas oferecendo a
##maior separação entre as duas classes (m<241) no espaço original X,
##então podem ser tomados as m maiores parcelas da distância.

###################################DISCRIMINANTE#################################

#####################################################
#  SELEÇÃO DA AMOSTRA DE TREINAMENTO E DE TESTE  #
#####################################################

intact1 <- amostra1[amostra1$X.2==0,]
damaged1 <- amostra1[amostra1$X.2==1,]
damaged2 <- amostra1[amostra1$X.2==2,]
damaged3 <- amostra1[amostra1$X.2==3,]
damaged4 <- amostra1[amostra1$X.2==4,]

odd1 <- c(seq(1, nrow(intact1), 2))
odd2 <- c(seq(1, nrow(damaged1), 2))
odd3 <- c(seq(1, nrow(damaged2), 2))
odd4 <- c(seq(1, nrow(damaged3), 2))
odd5 <- c(seq(1, nrow(damaged4), 2))


train <- rbind(intact1[odd1,], damaged1[odd2,],
   damaged2[odd3,], damaged3[odd4,], damaged4[odd5,])
train <- na.exclude(train)
even <- rbind(intact1[-odd1,], damaged1[-odd2,],
   damaged2[-odd3,], damaged3[odd4,], damaged4[-odd5,])
even <- na.exclude(even)
```

```
GRANMclustDA1 <- mclustDA(train=list(data=train[,-1],labels=train[,1]),
test= list(data=even[,-1],labels=even[,1]), verbose=F)

GRANMclustDA1
class1 <- GRANMclustDA1$test$classification
table(even[,1], class1)
```