**Universidade de São Paulo
Escola Superior de Agricultura “Luiz de Queiroz”**

**Um estudo sobre reconhecimento de padrões:
um aprendizado supervisionado com classificador bayesiano**

Pedro Henrique Ramos Cerqueira

Dissertação apresentada para obtenção do título de
Mestre em Ciências. Área de concentração: Estatística
e Experimentação Agronômica

**Piracicaba
2010**

Pedro Henrique Ramos Cerqueira
Estatístico

**Um estudo sobre reconhecimento de padrões:
um aprendizado supervisionado com classificador bayesiano**

Orientadora:
Profa. Dra. **ROSELI APARECIDA LEANDRO**

Dissertação apresentada para obtenção do título de
Mestre em Ciências. Área de concentração: Estatística
e Experimentação Agronômica

**Piracicaba
2010**

**Dados Internacionais de Catalogação na Publicação
DIVISÃO DE BIBLIOTECA E DOCUMENTAÇÃO - ESALQ/USP**

Cerqueira, Pedro Henrique Ramos

Um estudo sobre reconhecimento de padrões: um aprendizado supervisionado com classificador bayesiano / Pedro Henrique Ramos Cerqueira. - - Piracicaba, 2010.
108 p. : il.

Dissertação (Mestrado) - - Escola Superior de Agricultura "Luiz de Queiroz", 2010.
Bibliografia.

1. Aprendizagem 2. Classificação 3. Estatística computacional 4. Inferência bayesiana
5. Marcador molecular 6. Mineração de dados 7. Probabilidade 8. Reconhecimento de padrões
9. Softwares I. Título

CDD 519.50285
C416u

"Permitida a cópia total ou parcial deste documento, desde que citada a fonte – O autor"

Dedicatória

Em memória de minha mãe, Juraci Ramos Cerqueira.

A minha vó, Adelaide Ramos Cerqueira,
A minha tia, Roseli Ramos Cerqueira,
que sempre me apoiaram e estiveram ao meu lado,
suprindo as funções de Pai e Mãe simultaneamente,
sempre com amor incondicional.

AGRADECIMENTOS

Primeiramente aos meus familiares: minha avó, Adelaide Ramos Cerqueira, minhas tias, Roseli Ramos Cerqueira e Carmem, meu tio CID Ramos Cerqueira e meu primo Rodrigo Saffuan, por estarem ao meu lado, mesmo a a quilômetros de distância durante toda esta jornada, por brigarem comigo quando estava errado e principalmente por serem minha família.

À minha namorada, Camila, por todo o seu apoio, ajuda e pensamento positivo. Sempre me fazendo olhar para frente, nunca me deixando abalar com adversidades e NUNCA desistir.

Aos meus parentes: meu primo Felipe e sua esposa Camila, meus primos Gustavson e Bruno, minhas tias Dirce e Alice, minha madrinha Irene, por todo apoio durante todas as etapas de minha vida.

Aos meus amigos: Diego “Cavalo”, Aline, Richard “Leôncio”, Beto, Camila, Daniel e Juliana, pelos momentos de alegria descontração e discussões sobre valores que sempre fizeram abrir meus olhos para outros pontos de vista.

Aos meus amigos de pós-graduação: Adriana, Marcelino “Popó”, Cássio, Guilherme Biz, Luiz Ricardo, Ana Julia, e Everton, Simone, Renata, Ítalo, Gabriel, João, Daniel, Ricardo e Thiago.

Em especial aos meus amigos de pós-graduação: Rodrigo “Digão”, Raphael “Kconde” por sempre estarem ao meu lado nos momentos de descontração, nas discussões sobre ensino e como ser profissionais melhores e principalmente por estarem ao meu lado nos momentos de aperto sempre oferecendo ajuda.

Agradeço minha amiga Mariana por ter me acolhido quando eu ainda era um estranho em Piracicaba, e também, pelas caronas para o hospital, rodoviária entre outras e apesar de todas as divergências sempre ver em mim um potencial.

Às secretárias do programa de pós graduação: Luciane e Solange por toda a ajuda nos momentos cruciais e na correria na reta final.

Aos funcionários do programa de pós graduação: Eduardo, Jorge e Rosni por me ajudar quando eu precisava de alguma ajuda.

Aos professores do programa de pós graduação em Estatística e Experimentação Agronômica: Carlos Tadeu dos Santos Dias, Edwin Moisés Marcos Ortega, Silvio Sandoval Zocchi, Sônia Maria De Stefano Piedade e Taciana Villela Savian pelos conselhos e ajuda nos momentos de dúvidas.

À professora doutora Juliana Garcia Cespedes, por toda ajuda e colaboração para o desenvolvimento dessa dissertação.

Agradeço em especial à minha orientadora professora Roseli Aparecida Leandro, por nunca ter desistido de mim mesmo com toda a minha aparente falta de dedicação, por sempre estar presente para me auxiliar e direcionar o meu caminho.

Agradeço em especial à professora Clarice Garcia Borges Demétrio, os puxões de orelha e as conversas em sua sala, sempre me chamando a atenção quando eu estava errado, fazendo eu modificar a minha postura, torando-me um profissional e pessoa melhor, com seus conselhos.

Ao Conselho Nacional de desenvolvimento Científico e Tecnológico - CNPq pelo apoio financeiro em forma de bolsa de estudos.

Por fim a todos que sempre acreditaram em mim.

OBRIGADO!!!!!!

SUMÁRIO

RESUMO	9
ABSTRACT	11
LISTA DE FIGURAS	13
LISTA DE TABELAS	15
1 INTRODUÇÃO	17
2 DESENVOLVIMENTO	21
2.1 Revisão Bibliográfica	21
2.1.1 Reconhecimento de Padrões	21
2.1.2 Aprendizagem	25
2.1.2.1 Aprendizado não supervisionado	26
2.1.2.2 Aprendizado supervisionado	26
2.1.3 Classificadores	27
2.1.3.1 Classificador bayesiano	29
2.1.3.2 Função discriminante para duas classes	33
2.1.3.3 Função discriminante para dados binários e multinomiais	34
2.1.3.4 Função discriminante para distribuição Normal	37
2.1.3.5 Função discriminante para normal multivariada	37
2.1.3.5.1 Caso 1: $\Sigma_i = \sigma^2 \mathbf{I}$	39
2.1.3.5.2 Caso 2: $\Sigma_i = \Sigma$	42
2.1.3.5.3 Caso 3: Σ_i =arbitrária	43
2.1.3.6 Avaliação da classificação	44
2.1.3.6.1 Índice Kappa	44
2.1.3.6.2 Probabilidade de erro geral	47
2.1.4 Problemas de classificação	48
2.1.5 Seleção de Variáveis	49
2.1.5.1 Método do passo atrás (<i>backward</i>)	50
2.1.5.2 Método do passo a frente (<i>forward</i>)	51
2.1.5.3 Método do passo a frente passo atrás (<i>stepwise</i>)	52
2.2 Material e Métodos	53

2.2.1 Marcadores Moleculares	54
2.2.2 Dados de soja	55
2.2.3 Dados de peças de carros	55
2.2.4 <i>Software</i>	57
2.3 Exemplo	57
3 RESULTADOS E DISCUSSÃO	61
3.1 Análise de marcadores moleculares	61
3.2 Análise para dados de soja	66
3.3 Análise das peças de Carro	68
4 CONCLUSÕES	75
REFERÊNCIAS	77
APÊNDICES	81

RESUMO

Um estudo sobre reconhecimento de padrões: um aprendizado supervisionado com classificador bayesiano

A facilidade que temos para reconhecer um rosto, compreender palavras faladas, ler manuscritos, identificar chaves do carro no bolso e decidir se uma maçã está madura pelo seu cheiro, desmentem os processos complexos que estão por trás desses atos de reconhecer estes padrões. Estes reconhecimentos têm sido cruciais para a nossa sobrevivência, e ao longo das últimas dezenas de milhões de anos desenvolvemos sistemas sofisticados para a realização dessas tarefas. O reconhecimento de padrões tem por objetivo realizar a classificação de determinado conjunto de dados em determinadas classes ou grupos, considerando os seus padrões e os das classes, permitindo diversas aplicações, como por exemplo: processamento de documentos, leitores de código de barra; identificação de pessoas, leitores óticos ou de impressão digital; automação industrial, processamento de imagens e aplicações agronômicas, análise de marcadores moleculares e classificação de plantas, tornando-se nos últimos anos, uma técnica de grande importância. Para uma melhor classificação é necessário realizar aprendizados, que podem ser elaborados pelo método supervisionado ou não supervisionado, a fim de desenvolver os classificadores, tais como o classificador bayesiano e as redes neurais, que permitem a tomada de decisões. Para verificar a qualidade das classificações devem ser utilizadas medições específicas, como o índice kappa ou a probabilidade de erro geral. Deste modo é essencial a utilização de software para a tomada de decisões, entre eles o *R* e o *WEKA*, desenvolvido especificamente para resolução de problemas de reconhecimento de padrões. Com o intuito de solucionar os problemas específicos das áreas de automação e agronomia, foi utilizado o método de aprendizado supervisionado, com classificador bayesiano e para verificar a qualidade das classificações foram utilizados o índice de kappa e a probabilidade de erro geral por meio dos *software R* e *WEKA*, para as classificações foram utilizados dados de marcadores moleculares, dados de soja e tipos de embalagens de peças de automóvel.

Palavras-chave: Aprendizagem; Classificação; Estatística computacional; Marcador molecular; Reconhecimento de padrões

ABSTRACT

A study on pattern recognition: supervised learning with a Bayesian classifier

The facility we have to recognize a face, to understand spoken words, reading manuscripts, identifying car keys in our pocket and deciding whether an apple is ripe by its smell, belie the complex processes that are behind the act to recognize these patterns. These recognitions have been crucial to our survival, and over the past tens of millions of years sophisticated systems were developed to accomplish these tasks. The pattern recognition has aimed to carry out the rating of a given set of data in certain classes or groups, considering their standards and those of their classes, allowing several applications, such as: document processing, bar code readers, identification of people, optical and fingerprint drivers, industrial automation, image processing and agronomic applications, molecular markers analysis and plant's classification, which became a technique of great importance in recent years. For a better rating is necessary to perform some studies, which can be formulated by supervised and unsupervised methods, in order to develop classifiers, such as the Bayesian classifier and neural networks, which enable the decision-making. To check the quality of ratings, specific measurements must be used, such as the kappa index or the general error probability. Thus it is essential to use software to make a decision, including the R and WEKA, developed specifically to solve pattern recognition problems. In order to solve problems of specific areas, as automation and agronomy, the supervised learning method with Bayesian classifier was used, and to verify the ratings quality, the kappa index and the general error probability were used. Software R and WEKA were used, in order to perform ratings for molecular markers data, soybean data and types of packaging for auto parts.

Keywords: Classification; Learning; Molecular marker; Pattern Recognition; Statistical computing

LISTA DE FIGURAS

Figura 1 - Estrutura para criação do algoritmo de classificação	23
Figura 2 - Estrutura de um reconhecimento de padrões	24
Figura 3 - Universo das classes	25
Figura 4 - Estrutura do classificador de MAP	31
Figura 5 - Valores para $P(\omega_i \mathbf{x})$ em relação as observações	33
Figura 6 - Valores da probabilidade para os diferentes tamanhos de conjunto de treino	64
Figura 7 - Valores do índice kappa para os diferentes tamanhos de conjunto de treino .	65

LISTA DE TABELAS

Tabela 1 - Classificações em relação as classes verdadeiras	46
Tabela 2 - Classificações em relação as classes verdadeiras	46
Tabela 3 - Interpretação do índice Kappa	47
Tabela 4 - Origem das 18 linhagens	55
Tabela 5 - Probabilidade de erro geral para o tamanho do treino pelo programa PRP .	62
Tabela 6 - Índice kappa para o tamanho do treino no pelo programa PRP	62
Tabela 7 - Probabilidade de erro geral para o tamanho do treino no <i>WEKA</i>	63
Tabela 8 - Índice kappa para o tamanho do treino no <i>WEKA</i>	63
Tabela 9 - Probabilidade de erro em relação ao tamanho do treino	64
Tabela 10 -Índice kappa em relação ao tamanho do treino	64
Tabela 11 -Probabilidade de erro	68
Tabela 12 -Índice kappa	68
Tabela 13 -Probabilidade de erro	73
Tabela 14 -Valores e classificação do índice kappa	73

1 INTRODUÇÃO

O conceito de reconhecer padrões está diretamente relacionado com nossas ações cotidianas, desde a simples diferenciação de chaves em nosso bolso pelo tato como a consulta do valor de um produto por meio de um leitor de código de barras (DUDA et al, 2000; FRIEDMAN et al, 2008).

O campo da estatística é constantemente desafiado por problemas que a ciência e a indústria propõem. Inicialmente, esses problemas surgiam muitas vezes de experiências agrícolas e industriais e eram relativamente simples. Com o avanço computacional e maior disponibilidade de informações, os dados aumentaram em relação à quantidade e complexidade. Dificuldades no armazenamento de dados levaram ao desenvolvimento do *data mining*. Com a finalidade de estudar os problemas na biologia e na medicina, que necessitam do uso de estatística e informática, foi desenvolvida a bioinformática (FRIEDMAN et al, 2008; MITCHELL, 1997).

Com o aprimoramento das técnicas para armazenamento de dados e a facilidade de se obter acesso à informação, muitos dados têm sido coletados em diversas áreas. O trabalho estatístico consiste em utilizar as ferramentas disponíveis para conseguir extrair padrões importantes e tendências, deste modo, é possível entender o que os “dados dizem” (FRIEDMAN et al, 2008).

O armazenamento de dados possibilita, realizar uma grande quantidade de análises por meio de métodos diferentes. Com os dados disponíveis há o interesse em se obter informações que possam ajudar e facilitar a tomada de decisões, ou seja, descobrir padrões em relação aos dados. O problema da procura de padrões tem uma longa história de sucesso, as observações astronômicas de Tycho Brahe, no século XVI, permitiram Johannes Kepler descobrir as leis empíricas do movimento planetário, que por sua vez, serviram para o desenvolvimento da mecânica clássica. Da mesma forma, a descoberta de regularidades nos espectros atômicos desempenhou um papel fundamental no desenvolvimento e verificação da física quântica no início do século XX (BISHOP, 2006).

O campo do reconhecimento de padrões tem como objetivo a verificação de regularidades nos dados, por meio da utilização de algoritmos computacionais e com o uso desses recursos realizar a tomada de decisões, tais como classificar os dados em diferentes

categorias. Assim sendo, o reconhecimento de padrões pode ser definido como o ato de utilizar os dados brutos e com base nos padrões da “categoria” para tomar decisões. O ato da tomada de decisões tem sido crucial para a nossa sobrevivência, ao longo dos últimos anos (BISHOP, 2006; DUDA et al, 2000; JAIN et al, 2000).

A área de segurança, tem utilizado muito os conceitos de reconhecimento de padrões, por meio da identificação de pessoas como, por exemplo: reconhecimento de vozes, digitais, face entre outras características dos indivíduos. Atualmente este procedimento tem sido utilizado em diversas ocasiões, como por exemplo: eleições, acesso a computadores, substituição de senhas em agências bancárias, dentre outras situações em que é imprescindível ter certeza que o acesso será dado à pessoa correta (CAMPOS, 2001; JAIN et al, 2000; MORAIS, 2010).

Os desafios para a classificação dos dados realizaram uma revolução nas ciências estatísticas, levando a estudos específicos para a tomada de decisões, sempre com o objetivo de melhorar os resultados obtidos, ou seja, tomar a decisão correta (FRIEDMAN et al, 2008; MARQUES, 2005).

Visando aperfeiçoar esses resultados, foram desenvolvidos os aprendizados, que consiste em utilizar os dados disponíveis para elaborar critérios para a tomada de decisões, pois deste modo podem ser obtidos melhores resultados de classificação. O aprendizado pode ser supervisionado ou não supervisionado. No aprendizado supervisionado, o objetivo é prever a classificação de uma observação com base em uma série de medidas de entrada, sendo que as quantidades de categorias são conhecidas, no aprendizado não supervisionado o objetivo é descrever as associações e os padrões do conjunto de medidas de entrada, e nesse caso, a quantidade de categorias não é conhecida (DUDA et al, 2000; FRIEDMAN et al, 2008; MARQUES, 2005; MORAIS, 2010).

Dentre as técnicas disponíveis para realizar as classificações dos dados, a técnica de reconhecimento de padrões merece um estudo mais detalhado. Esta pode ser utilizada para auxiliar as mais diversas áreas, tais como:

- 1) Processamento de documentos: Reconhecimento de caracteres impressos ou escritos (Máquinas de leitura para deficientes visuais, análise de documentos financeiros e leitores de códigos de barras);

- 2) Automação industrial: Inspeção e montagem de objetos complexos (inspeção de circuitos impressos, inspeção de partes de máquinas e processamento de imagem);
- 3) Detecção remota (*Remote sensing*): Observação do planeta por meio de sensores em satélites ou aviões (previsão da evolução de culturas, planejamento de uso de terras, monitorização ambiental, meteorologia, exploração de minas e mapas topográficos);
- 4) Medicina e biologia: Processamento de diversos sinais e imagens médicas (contagem de células no sangue, detecção de tumores em imagens de Raios-X, caracterização de tecidos usando ultra-sons, análise de imagens de cromossomos, interpretação de eletrocardiogramas e diagnóstico médico;
- 5) Identificação de pessoas: Restrição de acesso em instalações de segurança (reconhecimento de voz, identificação de impressões digitais e reconhecimento de caras);
- 6) Aplicações científicas (interpretação de ondas sísmicas para previsão de terremotos e análise da composição molecular por meio de imagens de microscópio electrónico);
- 7) Aplicações na agricultura (direcionamento de equipamento, inspeção de produtos, ordenação e empacotamento de produtos).

Para a realização do reconhecimento dos padrões de um conjunto de dados, é necessário utilizar uma abordagem como, por exemplo: morfologia matemática com aprendizado computacional e estatística. Outros procedimentos fundamentais são: 1) os aprendizados 2) a escolha do classificador, que são necessários para elaborar os algoritmos de tomada de decisões, entre eles têm-se: o classificador bayesiano, redes neurais, árvore de decisão ou a regra dos k -vizinhos mais próximos (GADI, 2008; MARQUES, 2005; MORAIS, 2010).

Realizadas as classificações é de suma importância verificar a qualidade das classificações, por meio de métodos que avaliem os erros e acertos cometidos, entre os métodos possíveis pode-se utilizar o índice kappa ou a probabilidade de erro geral. Para a classificação, devem ser utilizados *software*, como por exemplo o *R*, para procedimentos específicos do reconhecimento de padrões e programar os algoritmos para tomada de decisões, e o *WEKA*, *software* desenvolvido para o reconhecimento de padrões. Atualmente existe uma interface entre *WEKA* e *R*: biblioteca (RWEKA) (HALL et al, 2009; LOBÃO et al, 1995).

Neste trabalho será apresentada uma revisão sobre o reconhecimento de padrões por meio de uma abordagem estatística com aprendizado supervisionado com classificador bayesiano. Como a aplicação de reconhecimento de padrões pode ocorrer em diversas áreas, foram analisados dados de diferentes áreas, tais como: genética, com dados de marcadores moleculares; agricultura, com dados referentes a diferentes tipos de doença em soja; e automação, com dados referentes a embalagens de peças de automóveis. Foram utilizados para a classificação algoritmos implementados em dois *software*: *R* e *WEKA*.

2 DESENVOLVIMENTO

Para o desenvolvimento deste trabalho será feita uma revisão sobre reconhecimento de padrões e seleção de variáveis.

2.1 Revisão Bibliográfica

2.1.1 Reconhecimento de Padrões

Para entender melhor o reconhecimento de padrões é interessante entender a morfologia destas palavras, por definição reconhecimento é considerado o ato de reconhecer, que por sua vez é definido como distinguir por meio de certos caracteres, e padrões são modelos que servem de referência, que representa a relação de umas coisas com outras. Deste modo pode-se pensar em reconhecimento de padrões como uma técnica que por meio de estruturas de referência realiza a distinção destas. Para Duda et al (2000) o reconhecimento de padrões é definido como o ato de utilizar um conjunto de observações (dados) para tomar uma ação, baseada no padrão de sua “categoria”.

O reconhecimento de padrões está relacionado com diversas áreas de pesquisa e abrange desde a detecção de padrões da escolha mais simples entre dois objetos, como a complexa realização de aprendizagem, este está presente em nosso cotidiano em diversas ocasiões, tais quais a escolha de objetos por meio do tato ou o fato de utilizar leitores de códigos de barra para o pagamento de contas (CAMPOS, 2001; FRIEDMAN et al, 2008; JAIN et al, 2000; MARQUES, 2005; MORAIS, 2010).

Com a tecnologia disponível atualmente é possível armazenar uma grande quantidade de dados, deste modo há uma maior possibilidade que sejam encontrados padrões, tendências e anomalias e assim transformar estes dados em informações, e as informações, em base para as classificações (WITTEN; FRANK, 2005).

Para a realização do reconhecimento dos padrões de um conjunto de dados, é necessário definir qual abordagem será utilizada, dentre elas têm-se:

- 1) Casamento ou associação (*template matching*);
- 2) Abordagem sintática;

- 3) Lógica nebulosa;
- 4) Morfologia matemática com aprendizado computacional;
- 5) Estatística.

Depois da definição da abordagem a ser utilizada, é necessário definir o aprendizado de interesse, o aprendizado supervisionado, em que, são conhecidas as classes para a classificação, ou, não supervisionado, em que não são conhecidas as classes, deste modo são criados grupos com características semelhantes.

Para serem tomadas as decisões os aprendizados são de suma importância, pois, por meio deles são definidos os algoritmos de decisão, que são ferramentas que realizam a tomada de decisões. Estes aprendizados são realizados a partir de um conjunto de dados retirados do original, e são conhecidos como conjunto de treino, este conjunto possui exemplos de padrões de todas as classes existentes no sistema, para verificar a eficiência da decisão é utilizado um conjunto para validação, preferencialmente com dados que não estejam no conjunto de treino, ou seja, as observações restantes do conjunto de dados original (CAMPOS, 2001; DUDA et al, 2008; MARQUES, 2005; MORAIS, 2010).

Existem diferentes métodos para selecionar o conjunto de treino e realizar a validação do algoritmo de decisão dentre eles têm-se:

- 1) Porcentagem do conjunto de dados: Consiste em utilizar uma amostra do conjunto de dados para elaborar o algoritmo de classificação e com a mostra não utilizada realizar a validação do algoritmo utilizado;
- 2) Validação cruzada: Consiste em dividir o conjunto de dados em p subconjuntos, em sequência utilizar $p - 1$ subconjuntos para criar o algoritmo de classificação, em seguida utiliza-se o subconjunto que não foi utilizado para a validação. Este processo é realizado para todos os p subconjuntos, ou seja, todos os subconjuntos serão utilizados para a validação e para criar o algoritmo de classificação.

Pode-se verificar pela Figura 1 como é a estrutura para se elaborar o algoritmo de decisão.

Figura 1 - Estrutura para criação do algoritmo de classificação

- 1) Aquisição de dados: Obtenção do conjunto de dados para análise;
- 2) Pré-processamento: Serve para eliminar ruídos ou as distorções;
- 3) Extração e seleção de características: A extração cria um vetor com as características dos dados, transformando-os em atributos, propriedades ou características, quando necessário o seletor elimina as características mais redundantes;
- 4) Com base na decisão de interesse é elaborado o algoritmo de decisão;

O primeiro passo é a obtenção dos dados, e assim definir, de acordo com suas estrutura qual o tipo de aprendizado será elaborado, quantos e de que maneira os dados serão definidos como pertencente ao conjunto de treino ou de validação e qual tipo de validação que será utilizada.

O pré-processamento e a seleção de características são realizados com o objetivo de acelerar o processo computacional, por exemplo, se o objetivo de uma classificação for detecção de uma face em tempo real, o computador necessita lidar com um grande número de *pixels* por segundo, e necessita apresentar estes diretamente a um complexo algoritmo de reconhecimento de padrões, que pode se tornar em alguns casos computacionalmente inviável, deste modo com a seleção de características e com a eliminação de ruídos este procedimento pode ser simplificado (BISHOP, 2006; FRIEDMAN et al, 2008; JIANG et al, 2005; MARQUES, 2005; MORAIS, 2010).

É importante mencionar que alguns cuidados devem ser tomados quando for feita a seleção de características, pois em algumas situações a informação que é descartada é

importante para o pesquisador, que não pode deixar de analisá-la, e para a resolução do problema, ou seja, pode influenciar na classificação do sistema podendo prejudicar a sua exatidão (CAMPOS, 2001; JAIN et al, 2000; MORAIS, 2010).

Após utilizar os dados do conjunto de treino para elaborar o algoritmo de decisão é utilizado o conjunto de dados de validação para realizar as decisões de interesse.

Pela Figura 2 pode-se verificar como é a estrutura para a tomada de decisão nos reconhecimento de padrões.

Figura 2 - Estrutura de um reconhecimento de padrões

Para as etapa de Classificação têm-se:

- 1) Pré-Processamento: Serve para eliminar ruídos ou as distorções;
- 2) Medição das características: São realizadas as medições das características de interesse e assim cria-se o vetor de características;
- 3) Resultado do algoritmo: Os dados medidos são utilizados no algoritmo para obtenção de valores;
- 4) Decisão: Com os valores obtidos por meio dos algoritmos é realizada a tomada de decisão.

A medição da característica deve realizar o mesmo processo que foi feito para o conjunto de treino, ou seja, as variáveis devem ser as mesmas que as que foram extraídas e para os casos que forem realizadas a seleção de característica serão avaliadas somente as características de interesse.

As características medidas juntamente com o algoritmo de classificação geram valores que são fundamentais para a tomada de decisão.

Na abordagem de reconhecimento de padrões, cada padrão é representado em termos de d características ou atributos e é representado por um vetor $\mathbf{x} = [x_1, \dots, x_d]^t$, em que cada x_i é uma característica. O espaço formado pelos vetores de características é chamado de espaço de características (DUDA et al, 2000; JAIN et al, 2000, MARQUES, 2005).

O conjunto de padrões “similares”, ou seja, os valores assumidos para cada uma das d características, que podem ser categóricas, discretas ou contínuas geram uma classe ω_i , que representa i -ésima classe de um conjunto de classes Ω (Figura 3).

Figura 3 - Universo das classes

Cada classe contém padrões que possuem alguma relação ou peculiaridade em comum. Para exemplificar como são esses padrões pode-se pensar em um exemplo simples de biometria: Seja o espaço de padrões $\mathbf{x} = (x_1, x_2, x_3)$ em que x_1 representa altura, x_2 representa peso e x_3 representa o tamanho dos pés. Nesse espaço de três dimensões, cada observação $\mathbf{x}$ representa as medições de uma pessoa em um determinado instante e cada classe representa uma família. Nesse caso o problema de classificação define-se por: dada uma pessoa desconhecida, extrair suas características para obter seu vetor de características $\mathbf{x}$ e determinar a qual família essa pessoa pertence (DUDA et al, 2000; MARQUES, 2005; MORAIS, 2010; WITTEN; FRANK, 2005).

2.1.2 Aprendizagem

O aprendizado ou treino é fundamental para os problemas de classificação de padrões, de acordo com o conjunto de dados disponível pode-se definir qual o tipo de aprendi-

zado será utilizado. A aprendizagem consiste em utilizar os dados disponíveis para poder criar os algoritmos de classificação e em alguns casos grupos com padrão semelhante (DUDA et al, 2000; FRIEDMAN et al, 2008; MORAIS, 2010).

O aprendizado pode-se desenvolvido de duas maneiras:

- 1) Aprendizado não supervisionado, os padrões de treino não se encontram classificados, ou seja, não se conhecem as classes que geraram os padrões;
- 2) Aprendizado supervisionado, em que as classes que geraram os padrões são conhecidas.

2.1.2.1 Aprendizado não supervisionado

Para o aprendizado não supervisionado, o conjunto possui somente padrões de entrada, ou seja, não há o conhecimento sobre como são os dados e relação às classes, nem informações sobre a quantidade, dos nomes e outras informações, ao contrário do aprendizado supervisionado, que o conjunto possui entrada e saída (DUDA et al, 2000; GADI, 2008; FRIEDMAN et al, 2008; MARQUES, 2005; MORAIS, 2010).

Para o desenvolvimento do aprendizado não supervisionado, pode-se utilizar:

- 1) Técnicas de análise de agrupamentos
- 2) Técnicas de análise de componentes principais

Essas técnicas podem ser utilizadas tendo em vista que o objetivo neste tipo de aprendizado está em criar grupos que possuam características semelhantes (DUDA et al, 2000; FRIEDMAN et al, 2008; JAIN et al, 2000; MARQUES, 2005; MORAIS, 2010; SUN, 2009).

2.1.2.2 Aprendizado supervisionado

O aprendizado supervisionado é o tipo de aprendizado em as categorias são conhecidas antecipadamente, sabe-se como estas estão dispostas, com a quantidade de observações para cada uma das classes e a quantidade de classes que deseja-se classificar. Neste tipo de aprendizado um especialista pode fornecer as informações sobre o conjunto em que há o interesse de se classificar, com por exemplo a quantidade de classes, seus nomes, o custo para cada tomada de decisão entre outras informações. Deste modo o objetivo no aprendizado supervisionado está em realizar a classificação de um determinado dado, pertencente ao conjunto,

utilizando um conjunto de dados como treino, para elaborar os critérios de decisão (DUDA et al, 2000; MORAIS, 2010; FRIEDMAN et al, 2008; MARQUES, 2005; WITTEN; FRANK, 2005).

Embora não seja de interesse, neste caso, realizar a comparação entre os tipos de aprendizado, alguns autores defendem que as técnicas de aprendizado supervisionado geram melhores resultados que os do aprendizado não supervisionado (GADI, 2008; JAIN et al, 2000, MARQUES, 2005).

Para realizar a classificação utilizando o aprendizado supervisionado podem ser utilizados dois métodos (MARQUES, 2005).

- 1) Estimar o modelo probabilístico a partir dos dados disponíveis, substituindo o modelo estimado pela classificação utilizando o teorema de Bayes;
- 2) Estimar diretamente as funções discriminantes que definem as fronteiras de decisão do classificador a partir dos dados.

Esse aprendizado é utilizado para execução dos algoritmos de classificação para elaborar as funções discriminantes que servem para determinar as fronteiras de decisão, que são as regiões que delimitam as diferenças entre as características, ou seja, dividem o espaço de características em c regiões, para um problema de c classes, sendo que cada região corresponde a uma classe, como pode ser visto pela Figura 3 (BISHOP, 2006; DUDA et al, 2000; FRIEDMAN et al, 2008). Este foi o aprendizado definido para este trabalho, deste modo é essencial realizar uma revisão sobre os classificadores e os métodos de como elaborar os seus algoritmos de decisões, suas funções discriminantes, assim como os critérios para a tomada da mesma.

2.1.3 Classificadores

Seja x_i um padrão desconhecido, pertencente ao conjunto de padrões X em um espaço de características $\mathbf{x}$, e o conjunto Ω com todas as classes existentes, o classificador é uma função $Y : X \longrightarrow \Omega$ tal que $Y(\mathbf{x}) = \omega_i$, em que ω_i representa a i -ésima classe de Ω .

Os classificadores são funções que utilizam como entrada os padrões desconhecidos, e como saída as classes à que estes padrões provavelmente pertencem, são as ferramentas que realizam o reconhecimento de padrões em um aprendizado supervisionado (DUDA et al, 2000; MARQUES, 2005; MORAIS, 2010; PÉREZ, 2009; TANAKA, 1995).

A diferença entre os classificadores está no método em que se criam as fronteiras de decisões, porém as diferentes técnicas para estimar as fronteiras de decisões têm em comum os seguintes objetivos:

- 1) Minimizar o erro de classificação;
- 2) Permitir que a classificação seja eficiente computacionalmente, otimizando o processo, fazendo com que leve o menor tempo possível.

A importância desses objetivos varia de acordo com o classificador, sendo que o melhor resultado possível ocorre quando um classificador é rápido e preciso, mas esta situação dificilmente ocorre em problemas complexos.

Outro fator que pode influenciar nas classificações são as consequências que cada decisão pode ocasionar. Por exemplo, em um sistema de detecção de incêndio em que a cada momento deve-se verificar se há ou não algum foco de incêndio, sente caso, podem ocorrer dois tipos de erro: detectar incêndio quando não existe (alarme falso) e não detectar quando existe (falha no sistema). Ambos erros são prejudiciais, porém, quando ocorre a falha no sistema, esta leva à consequências mais graves. Outro exemplo: é o de sistema de detecção de avião em radar, fica evidente que é mais prejudicial não detectar um avião do que detectar um falso avião.

Em ambos os casos é preferível que o erro de ocorrer falha no sistema seja menor, mesmo que para isso seja necessário aumentar o erro para alarmes falsos, deste modo para resolver este problema pode-se atribuir um custo para cada decisão e desta maneira utilizar o classificador para minimizar o custo médio (DUDA et al, 2000; MARQUES, 2005).

Para realizar as classificações é necessário utilizar um classificador dentre eles pode-se destacar:

- 1) Árvores de Decisão
- 2) Redes Neurais
- 3) Classificador bayesiano ou Naive Bayes
- 4) Sistemas Imunológicos Artificiais

5) Algoritmo Genético

6) Regra dos K-vizinhos mais próximos

Para o seguimento desta foi escolhido o classificador bayesiano, por isso a seguir será realizado um estudo mais detalhado sobre o mesmo.

2.1.3.1 Classificador bayesiano

O classificador bayesiano, também conhecido como naive Bayes, é uma abordagem fundamental para os problemas de classificação de padrões. Esta abordagem é baseada na quantificação das comparações entre as várias decisões utilizando probabilidades e custos de tais decisões, assumindo que os problemas de decisão são colocados em termos probabilísticos e que estes valores são conhecidos (DUDA et al, 2000; JIANG et al, 2005; MARQUES, 2005; MITCHELL, 1997).

Este classificador é muito utilizado devido a sua simplicidade e sua eficiência, ou seja, a implementação de seu algoritmo de estimação é fácil e conseguem-se bons resultados de modo rápido. Porém é limitado à independência de suas características (GADI, 2008, RENNIE et al, 2003; WEBB et al, 2010).

Ao utilizar classificador bayesiano há o interesse em classificar uma observação em uma determinada classe, para tal é utilizado o conceito da probabilidade condicional e para serem desenvolvidas as suas funções discriminantes é utilizado o teorema de Bayes dado por:

$$P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{p(\mathbf{x})} \quad i = 1, \dots, c$$

em que $P(\omega_i|\mathbf{x})$ é representa a probabilidade a posteriori do item ser da classe ω_i dado que foi observado o padrão $\mathbf{x}$, $p(\mathbf{x}|\omega_i)$ representa a função densidade de probabilidade condicional, para dados contínuos e função de probabilidade condicional, para dados discretos, $P(\omega_i)$ é considerada a probabilidade a priori para cada classe e o valor de $P(\mathbf{x})$ é dado por:

$$\sum_{i=1}^c p(\mathbf{x}|\omega_i)P(\omega_i)$$

este representa uma constante de proporcionalidade e pode ser descartado no momento de elaborar a classificação (DUDA et al, 2000; JIANG et al, 2005; MARQUES, 2005; MITCHELL,

1997; PÉREZ, 2007; RENNIE et al, 2003; WEBB et al, 2010). Pode-se representa-lo de modo informal como:

$$\text{posteriori} = \frac{\text{verossimilhança} \times \text{priori}}{\text{evidência}}$$

Este teorema mostra que, observado o vetor $\mathbf{x}$ pode-se modificar a probabilidade a priori $P(\omega_i)$ para a probabilidade a posteriori $P(\omega_i|\mathbf{x})$, a probabilidade de uma classe ω_i ocorrer, sabendo que foram observados valores de $\mathbf{x}$. A função $p(\mathbf{x}|\omega_i)$ é chamada de verossimilhança de ω em relação a $\mathbf{x}$. Este produto entre a verossimilhança e a probabilidade a priori é o mais importante para determinar a probabilidade a posteriori (DUDA, 200; MARQUES, 2005).

A função discriminante para cada classe representa o logaritmo de sua probabilidade a posteriori, deste modo a função discriminante é dada por:

$$\ln P(\omega_i|\mathbf{x}) = g_i(\mathbf{x}) = \ln p(\mathbf{x}|\omega_i) + \ln P(\omega_i) - \ln P(\mathbf{x})$$

Nesse tipo de classificação há o interesse em escolher a classe que maximize a probabilidade a posteriori, ou seja, que minimize o erro de uma escolha, deste modo é utilizado o classificador de máximo a posteriori (MAP) como critério para o tomada de decisão (DUDA et al, 2000; JIAN et al, 2005; MARQUES, 2005; MITCHELL, 1997; PÉREZ, 2007; RENNIE et al, 2003; WEBB et al, 2010). Este critério pode ser visto como:

$$\hat{\omega} = \arg \max_{\omega \in \Omega} P(\omega|\mathbf{x})$$

sendo que $\arg \max$ representa o argumento que maximiza a expressão

Ao utilizar a função discriminante para a tomada de decisão com o MAP o interesse permanece em tomar uma decisão que maximize a probabilidade a posteriori (DUDA et al, 2000; MORAIS, 2010; WEBB et al, 2010). Seja $(g_1(\mathbf{x}), \dots, g_c(\mathbf{x}))$ uma sequência de funções discriminantes para cada uma das c classe, então a decisão de classificação pode ser vista como:

$$\hat{\omega}_i = \omega_k : k = \arg \max_{i=1, \dots, c} g_i(\mathbf{x})$$

Como $\arg \max$ representa o argumento que maximiza a expressão, a decisão será em prol do valor de i que maximiza $g_i(\mathbf{x})$, ou seja, a classe i que obtiver o maior valor para

a função discriminante será considerada a verdadeira. A estrutura do classificador de MAP pode ser vista na Figura 4.

Figura 4 - Estrutura do classificador de MAP

Para exemplificar como é utilizado o classificador bayesiano juntamente com o classificador de MAP, pode-se pensar em um problema que consiste em classificar duas flores, flor A e flor B , de acordo com algumas características.

Na classificação bayesiana é dito que cada flor surge de um estado da natureza ou seja, as flores só podem ser de dois tipos ou ela é A ou B . A classe a qual a flor pertence é denotada por: ω_i , com $\omega_i = \omega_1$ para a flor A e $\omega_i = \omega_2$ para a flor B (DUDA et al, 2000; MORAIS, 2010; PÉREZ, 2007).

Se a flor observada é flor A ou flor B , pode-se dizer que a próxima será A ou B , com as mesmas chances. Para este caso pode-se dizer que a probabilidade a priori para $P(\omega_1)$, a próxima flor é A , é a mesma que $P(\omega_2)$. Caso não haja outra espécie de flor então a soma de $P(\omega_1)$ e $P(\omega_2)$ será igual a 1.

Quando há o interesse em realizar uma classificação, afirmar se a flor é a A ou B , assumindo que ambas as decisões tem o mesmo custo, a única informação disponível será a chance da flor ser ou não A ou B . Deste modo usando uma regra de decisão pode se classificar como ω_1 se $P(\omega_1) > P(\omega_2)$, caso contrário classifica-se como ω_2 . Para casos que se possuam mais de duas classe pode-se elaborar a decisão como $P(\omega_i) > \dots > P(\omega_j)$, ou seja, a decisão será em prol da classe que possuir maior probabilidade a priori (DUDA et al, 2000).

Como a decisão de classificação é baseada em probabilidades, se $P(\omega_1)$ é muito maior que $P(\omega_2)$, a decisão será em favor de ω_1 , e na maioria dos casos esta decisão será a correta, quando ocorre $P(\omega_1)=P(\omega_2)$, a probabilidade da flor pertencer a qualquer umas das classes será a mesma.

Geralmente para se tomar as decisões não são utilizadas poucas informações, ou seja, no exemplo das flores, pode-se observar diferentes características para uma melhor classificação, como por exemplo comprimento e largura da sépala, que pode ser medida em $\mathbf{x}$, para aprimorar o classificador. É esperado que flores diferentes tenham comprimento e largura da sépala diferentes, deste modo pode-se expressar a variabilidade em termos estatísticos.

Seja $\mathbf{x}$ um vetor de variáveis aleatórias contínuas cuja as distribuições dependem dos estados da natureza, pode-se expressar esta distribuição como $p(\mathbf{x}|\omega_i)$, a função densidade de probabilidade condicional para as classes, a função de probabilidade condicional $p(\mathbf{x}|\omega_i)$ deve ser escrita como $p_X(\mathbf{x}|\omega_i)$ para indicar que está sendo falado sobre uma função de densidade particular para o vetor X . Esta é função densidade de probabilidade para $\mathbf{x}$ dado que ocorreu o estado da natureza ω_i (DUDA et al, 2000; MARQUES, 2005).

Sejam conhecidas as probabilidades a priori de $P(\omega_i)$ e a densidade condicional $p(\mathbf{x}|\omega_i)$, em que i representa a classe que pertence a flor. Há o interesse em medir a variabilidade das sépalas e descobrir o valor de $\mathbf{x}$, para verificar como esta medida pode influenciar na tomada decisão sobre classe da planta. Utiliza-se a densidade de probabilidade, para encontrar o padrão que pertence a cada classe ω_i , assim o valor $\mathbf{x}$ pode ser escrito como

$$P(\omega_i, \mathbf{x}) = P(\omega_i|\mathbf{x})P(\mathbf{x}) = p(\mathbf{x}|\omega_i)P(\omega_i)$$

para este caso, em que há somente duas classe, o valor de $P(\mathbf{x})$ é dado por:

$$\sum_{i=1}^2 p(\mathbf{x}|\omega_i)P(\omega_i)$$

Deste modo a variação de $P(\omega_i|\mathbf{x})$, para o caso de $P(\omega_1) = 3/4$ e $P(\omega_2) = 1/4$, pode ser ilustrada pela figura 5.

Se para a observação $\mathbf{x}$ o valor de $P(\omega_1|\mathbf{x})$ for maior que $P(\omega_2|\mathbf{x})$ então a decisão será em prol de ω_1 caso contrário a decisão será tomada em favor de ω_2 . Para justificar a decisão tomada é calculada a probabilidade de erro para cada uma das decisões (DUDA et al, 2000; JAIN et al, 2000). Para um vetor particular $\mathbf{x}$ tem-se:

$$P(erro|\mathbf{x}) = \begin{cases} P(\omega_1|\mathbf{x}) & \text{se decidir } \omega_2; \\ P(\omega_2|\mathbf{x}) & \text{se decidir } \omega_1. \end{cases}$$

Figura 5 - Valores para $P(\omega_i|\mathbf{x})$ em relação as observações

Pode-se verificar que a probabilidade de erro é minimizada quando forem tomadas as decisões: ω_1 se $P(\omega_1|\mathbf{x}) > P(\omega_2|\mathbf{x})$ e ω_2 caso contrário. Para o exemplo da visto na Figura 5 pode-se dizer que, para comprimentos de sápal menores que 5,45 cm escolhe-se a classe 1, caso contrário escolhe-se a classe 2.

2.1.3.2 Função discriminante para duas classes

Um caso particular de classificação ocorre quando há o interesse em classificar uma observação e há somente duas classes possíveis, deste modo a classificação será mais simples. Utilizando o exemplo visto anteriormente em que o objetivo era o de classificar um vetor observado $\mathbf{x}$ entre duas classes ω_1 e ω_2 , que representavam as flores A e B , respectivamente, pode-se modificar o critério de decisão utilizando a razão de verossimilhança (DUDA et al, 2000; MARQUES, 2005).

Para ser feita a classificação do vetor $\mathbf{x}$ há duas opções, ou pertence a classe ω_1 ou a classe ω_2 , para a observação pertencer à ω_1 é necessário que $P(\omega_1|\mathbf{x}) > P(\omega_2|\mathbf{x})$, pelo teorema de Bayes foi visto que $P(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)P(\omega_i)}{P(\mathbf{x})}$, como o valor $P(\mathbf{x})$ será o mesmo para

ambas as classes não há a necessidade de sua utilização (BISHOP, 2006; DUDA et al, 2000; MARQUES, 2005). Deste modo pode-se reescrever o critério para a tomada de classificação como:

$$p(\mathbf{x}|\omega_1)P(\omega_1) > p(\mathbf{x}|\omega_2)P(\omega_2)$$

deste modo tem-se que:

$$\frac{p(\mathbf{x}|\omega_1)P(\omega_1)}{p(\mathbf{x}|\omega_2)P(\omega_2)} > 1$$

Pode ser verificado que para os valores superiores a 1 a observação será classificada como pertencente à classe ω_1 para os valores inferiores 1 a observação será classificada como pertencente à classe ω_2 .

Utilizando a função discriminante para realizar a tomada de decisão e para facilidades de implementação o critério para para classificar a observação como pertencente a ω_1 será:

$$\ln p(\mathbf{x}|\omega_1) + \ln P(\omega_1) > \ln p(\mathbf{x}|\omega_2) + \ln P(\omega_2)$$

Caso contrário, se $\ln p(\mathbf{x}|\omega_1) + \ln P(\omega_1) < \ln p(\mathbf{x}|\omega_2) + \ln P(\omega_2)$ a observação será classificada como pertencente à classe ω_2 .

O critério de classificação pode se tornar mais simples para os casos em que as probabilidades a priori forem iguais, pode-se observar que neste caso particular a observação será classificada utilizando somente a sua função densidade de probabilidade.

2.1.3.3 Função discriminante para dados binários e multinomiais

Para a classificação de padrões utilizando os classificadores bayesianos são utilizados dados que podem ser de dois tipos: discretos e contínuos, entre os dados discretos um caso particular ocorre quando os dados apresentam uma característica binária, ocorre quando o vetor observado é composto somente por dois valores: 0 e 1. Por exemplo, seja um vetor de dimensão 4, que corresponde ao número de características para cada dado do conjunto utilizado, $x = (1, 1, 1, 0)^t$ (BISHOP, 2006; DUDA et al, 2000; MORAIS, 2010).

Utilizando um exemplo para uma melhor visualização dessa classificação binária utilizando duas classes. Sejam duas fábricas que produzem a mesma máquina agrícola, em que

cada um dos d componentes produzidos podem ser defeituoso ou não, gerando valores 1 e 0, respectivamente. Este é um modelo problema de classificação em que cada característica tem como resposta sim / não (1 ou 0) em relação ao padrão. A probabilidade de ocorrer um dos eventos é dada por:

$$p_i = P(x_i = 1|\omega_1)$$

$$q_i = P(x_i = 1|\omega_2)$$

Se o valor $p_i > q_i$ é esperado que a i -ésima observação tenha resposta sim com mais frequência em ω_1 que em ω_2 . Deste modo assumindo que há independência condicional, pode-se escrever $p(\mathbf{x}|\omega_i)$ como o produto das probabilidades dos componentes de $\mathbf{x}$ (BISHOP, 2006; DUDA et al, 2000; FRIEDMAN et al, 2008; LANGLEY; JOHN, 1995; MARQUES, 2005).

Sendo um conjunto de dados com as características anteriores, uma forma particular de escrever a classe condicional de probabilidades é dada por:

$$p(\mathbf{x}|\omega_1) = \prod_{i=1}^d p_i^{x_i} (1 - p_i)^{1-x_i}$$

e

$$p(\mathbf{x}|\omega_2) = \prod_{i=1}^d q_i^{x_i} (1 - q_i)^{1-x_i}$$

Desta maneira a função de razão de verossimilhança é dada por:

$$\frac{p(\mathbf{x}|\omega_1)}{p(\mathbf{x}|\omega_2)} = \prod_{i=1}^d \left(\frac{p_i}{q_i}\right)^{x_i} \left(\frac{1-p_i}{1-q_i}\right)^{1-x_i}$$

Aplicando a função logaritmo na razão de verossimilhança obtêm-se a função discriminante dada por:

$$g(\mathbf{x}) = \sum_{i=1}^d \left[x_i \ln \frac{p_i}{q_i} + (1 - x_i) \ln \frac{1-p_i}{1-q_i} \right] + \ln \frac{P(\omega_1)}{P(\omega_2)}$$

por meio desta função discriminante pode ser definido se o vetor $\mathbf{x}$ pertence a classe ω_1 ou ω_2 , levando-se em consideração o valor obtido em $g(\mathbf{x})$.

Este caso foi o caso particular em que para a situação há somente duas classes, quando há mais de duas classes possíveis para a classificação, ou seja, $\omega_1, \omega_2, \dots, \omega_c$, pode-se verificar de modo análogo que, as probabilidades para as classes são dadas por:

$$\begin{aligned} p_{1i} &= P(x_i = 1|\omega_1) \\ p_{2i} &= P(x_i = 1|\omega_2) \\ p_{3i} &= P(x_i = 1|\omega_3) \\ &\vdots \\ p_{ci} &= P(x_i = 1|\omega_c) \end{aligned}$$

em que p_{ci} representa a probabilidade de ocorre 1 na c -ésima classe e na i -ésima característica do vetor $\mathbf{x}$.

Deste modo a função discriminante para a classificação será dada por:

$$\begin{aligned} g_{ci}(\mathbf{x}) &= \prod_{i=1}^d p_{ci}^{x_i} (1 - p_{ci})^{1-x_i} P(\omega_i) \\ &= \sum_{i=1}^d (x_i \ln(p_{ci}) + (1 - x_i) \ln(1 - p_{ci})) + P(\omega_i) \end{aligned}$$

em que g_{ci} representa a função discriminante para a i -ésima classe.

O caso anterior, para dados binários, é uma particularidade de dados multinomiais em que as características observadas assumem valores (0 e 1), para os casos em que não ocorre essa peculiaridade, como por exemplo a avaliação de um produto que pode ser feita com mais de duas respostas possíveis, como um questionário sobre a qualidade do produto que tenha três opções, ruim, regular e ótimo. Outro exemplo seria a quantidade de cores ou a estrutura de um determinado objeto (Bishop, 2006; DUDA et al, 2000; LANGLEY; JOHN, 1995). Para esse caso os estimadores de máxima verossimilhança para a probabilidade de cada observação é dado por:

$$\hat{p}_i = \frac{x_i}{N}$$

em que x_i representa o número de observações encontradas para a i -ésima característica de interesse e N representa o total de observações.

Os cálculos para obtenção desses estimadores podem ser vistas no Apêndice A.

2.1.3.4 Função discriminante para distribuição Normal

Como visto anteriormente a estrutura do classificador bayesiano é determinada pela densidade condicional $p(\mathbf{x}|\omega_i)$, nos casos em que é utilizada a distribuição normal de probabilidade há a necessidade de dividi-la entre univariada e multivariada.

A função densidade de probabilidade para a normal univariada é dada por:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left[-\frac{1}{2} \left(\frac{x - \mu}{\sigma} \right)^2 \right]$$

com esperança de X dada por:

$$\mu \equiv E[X] = \int_{-\infty}^{\infty} x f(x) dx$$

e com variância dada por:

$$\sigma^2 \equiv E[(X - \mu)^2] = \int_{-\infty}^{\infty} (x - \mu)^2 f(x) dx$$

Deste modo a função discriminante para esta distribuição pode ser escrita como:

$$g_i(x) = \ln f(x|\omega_i) + \ln P(\omega_i)$$

sendo assim

$$g_i(x) = -\frac{1}{2} \ln 2\pi - \ln \sigma_i - \frac{(x - \mu)^2}{2\sigma_i^2} + \ln P(\omega_i)$$

Como o valor de $-0,5 \ln 2\pi$ é uma constante, ou seja, não contribui para a discriminação entre as classes pode ser descartado da função discriminante (DUDA et al, 2000; MARQUES, 2005).

Assim a função utilizada será:

$$g_i(x) = -\ln \sigma_i - \frac{(x - \mu)^2}{2\sigma_i^2} + \ln P(\omega_i)$$

Utilizando o critério de classificação de MAP, uma observação será classificada como pertencente a classe i se esta possuir o maior valor de $g_i(x)$

2.1.3.5 Função discriminante para normal multivariada

Dentre todas as funções que são estudadas a função que necessita de um estudo mais aprofundado é a função densidade de probabilidade normal multivariada, que deve-se

à sua tratabilidade analítica. A densidade da normal multivariada também é um modelo apropriado para uma importante situação, em casos que os vetores de características $\mathbf{x}$ para uma determinada classe ω_i são valores contínuos (DUDA et al, 2000; FRIEDMAN et al, 2008; MARQUES, 2005).

A normal multivariada pode ser escrita como $f(\mathbf{x}) \sim N_P(\boldsymbol{\mu}, \Sigma)$, sendo que $\boldsymbol{\mu}$ representa o vetor de médias e Σ a matriz de covariância. A sua função densidade de probabilidade é dada por:

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left[-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^t \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right]$$

em que $\mathbf{x}$ representa o vetor com d componentes, $|\Sigma|$ representa o determinante da matriz de covariância, Σ^{-1} representa a sua inversa e $(\mathbf{x} - \boldsymbol{\mu})^t$ representa a transposta de $\mathbf{x} - \boldsymbol{\mu}$.

A esperança para a normal multivariada dada por:

$$\boldsymbol{\mu} \equiv E[\mathbf{X}] = \int_{-\infty}^{\infty} \mathbf{x} f(\mathbf{x}) d\mathbf{x}$$

e variância dada por:

$$\Sigma \equiv E[(\mathbf{X} - \boldsymbol{\mu})(\mathbf{X} - \boldsymbol{\mu})^t] = \int_{-\infty}^{\infty} (\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^t f(\mathbf{x}) d\mathbf{x}$$

Os valores esperados para o vetor e para a matriz são os dos seus componentes, deste modo se x_i é o i -ésimo elemento de $\mathbf{x}$, então μ_i é o i -ésimo elemento de $\boldsymbol{\mu}$ e σ_{ij} é o ij -ésimo elemento de Σ , então:

$$\mu_i = E[X_i]$$

e variância dada por:

$$\sigma_{ij} = E[(X_i - \mu_i)(X_j - \mu_j)]$$

A matriz de covariância é sempre simétrica e positiva definida e possui $\frac{n \times (n+1)}{2}$ parâmetros para serem estimados. A diagonal dos seus elementos σ_{ii} são as variâncias dos elementos x_i e os elementos σ_{ij} são as covariâncias de x_i e x_j . Se todos os valores fora da diagonal, ou seja, os valores das covariâncias forem 0 então $f(\mathbf{x})$ é reduzido ao produto das densidades da normal para os componentes de $\mathbf{x}$ (BISHOP, 2006; DUDA et al, 2000).

A função discriminante para normal multivariada é dada por:

$$g_i(\mathbf{x}) = -\frac{d}{2} \ln 2\pi - \frac{1}{2} \ln |\Sigma_i| - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_i)^t \Sigma_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) + \ln \mathbf{P}(\omega_i)$$

Para alguns casos especiais a função discriminante é modificada (DUDA et al, 2000; MARQUES, 2005). Os casos especiais são:

- 1) Caso 1: $\Sigma_i = \sigma^2 \mathbf{I}$: Ocorre quando as classes são independentes e possuem a mesma variância;
- 2) Caso 2: $\Sigma_i = \Sigma$: Ocorre quando a matriz Σ é não estruturada e a mesma para todas as classes;
- 3) Caso 3: Σ_i =arbitrário: Ocorre quando a matriz Σ é não estruturada e cada diferente para as classes.

2.1.3.5.1 Caso 1: $\Sigma_i = \sigma^2 \mathbf{I}$

Este caso ocorre quando as observações das c classes são independentes e possuem a mesma variância σ^2 . Neste caso a matriz de covariância é diagonal, representada apenas por σ^2 vezes a matriz identidade $\mathbf{I}$ de dimensão c , que pode ser visualizado na matriz a seguir. Isto ocorre quando as amostras pertencem à grupos de tamanho igual e aos mesmos grupos hiperesféricos, ou seja, o grupo para i -ésima classe está centrado em torno do vetor de média $\boldsymbol{\mu}_i$ (DUDA et al, 2000; FRIEDMAN et al, 2008; MARQUES, 2005).

$$\begin{pmatrix} \sigma^2 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma^2 \end{pmatrix}$$

O cálculo do determinate e da inversa de Σ_i , neste caso específico se torna fácil sendo os valores obtidos por:

$$|\Sigma_i| = \sigma^{2d}$$

e

$$\Sigma_i^{-1} = \frac{1}{\sigma^2} \mathbf{I}$$

Desde que ambos os termos $|\Sigma_i|$ e $(d/2) \ln 2\pi$ sejam independentes de i , eles são considerados constantes e podem ser ignorados (DUDA et al, 2000; FRIEDMAN et al, 2008; JAIN et al, 2000; MARQUES, 2005). Deste modo a função discriminantes é dada por:

$$g_i(\mathbf{x}) = -\frac{\|\mathbf{x} - \boldsymbol{\mu}_i\|^2}{2\sigma^2} + \ln P(\omega_i)$$

em que $\|\cdot\|$ representa distância Euclidiana, dada por:

$$\|\mathbf{x} - \boldsymbol{\mu}_i\|^2 = (\mathbf{x} - \boldsymbol{\mu}_i)^t(\mathbf{x} - \boldsymbol{\mu}_i)$$

Para os casos em que as probabilidades a priori não são semelhantes, a equação anterior mostra que a distância $\|\mathbf{x} - \boldsymbol{\mu}_i\|^2$ é normalizada pela variância σ^2 e pela adição do logaritmo de $P(\omega_i)$, ou seja, se o valor de $\mathbf{x}$ é próximo de dois valores do vetor de médias, a decisão será baseada em prol da probabilidade a priori da categoria mais provável (DUDA et al, 2000).

Para a criação da função discriminante para as classe não é necessário o cálculo da distância Euclidiana (DUDA et al, 2000; FRIEDMAN et al, 2008). Utilizando a expansão da forma quadrática $(\mathbf{x} - \boldsymbol{\mu}_i)^t(\mathbf{x} - \boldsymbol{\mu}_i)$ a função discriminante será:

$$g_i(\mathbf{x}) = -\frac{1}{2\sigma^2}[\mathbf{x}^t\mathbf{x} - 2\boldsymbol{\mu}_i^t\mathbf{x} + \boldsymbol{\mu}_i^t\boldsymbol{\mu}_i] + \ln P(\omega_i)$$

Pode-se verificar que o termo $\mathbf{x}^t\mathbf{x}$ será igual para todos i , tornando-se uma constante que pode ser desprezada. Assim pode-se obter a função equivalente a função discriminante linear que pode ser escrita como:

$$g_i(\mathbf{x}) = \mathbf{w}_i^t\mathbf{x} + \omega_{i0}$$

em que

$$\mathbf{w}_i = \frac{1}{\sigma^2}\boldsymbol{\mu}_i$$

e

$$\omega_{i0} = \frac{-1}{\sigma^2}\boldsymbol{\mu}_i^t\boldsymbol{\mu}_i + \ln P(\omega_i)$$

ω_{i0} é chamado de viés de correção no sentido da i -ésima observação.

O classificador que utiliza a função linear discriminante é conhecido como classificador linear. Esta superfície de decisão é definida pelas equações lineares $g_i(\mathbf{x}) = g_j(\mathbf{x})$, para as duas categorias com maior probabilidade a posteriori (DUDA et al, 2000). Quando o objetivo é realizar a classificação entre duas das c classes possíveis tem-se:

$$\mathbf{w}^t(\mathbf{x} - \mathbf{x}_0) = 0$$

em que

$$\mathbf{w} = \boldsymbol{\mu}_i - \boldsymbol{\mu}_j$$

e

$$\mathbf{x}_0 = \frac{1}{2}(\boldsymbol{\mu}_i + \boldsymbol{\mu}_j) - \frac{\sigma^2}{\|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|} \ln \frac{P(\omega_i)}{P(\omega_j)}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j).$$

Esta equação define um hiperplano que passa pelo ponto x_0 e é ortogonal ao vetor $\mathbf{w}$. Desde que $\mathbf{w} = \boldsymbol{\mu}_i - \boldsymbol{\mu}_j$, o hiperplano que separa $\mathcal{R}_i$ e $\mathcal{R}_j$ é ortogonal a linha que une as médias.

Se $P(\omega_i) = P(\omega_j)$ o segundo termo de $\mathbf{x}_0$ não é necessário, desde modo o ponto $\mathbf{x}_0$ está entre as médias, e o hiperplano é perpendicular a linha que passa entre as médias. Se $P(\omega_i) \neq P(\omega_j)$ o ponto $\mathbf{x}_0$ afasta-se da média mais provável. Se a diferença entre as variâncias for pequena em relação ao quadrado da distância $\|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|$, então a decisão não será sensível para os valores exatos da probabilidade a priori (DUDA et al, 2000; FRIEDMAN et al, 2008; MARQUES, 2005).

Se as probabilidades a priori $P(\omega_i)$ forem as mesmas para todas as c classes, então o $\ln P(\omega_i)$ será um termo constante igual para todos portanto pode ser eliminado. Quando ocorre este caso a função discriminante para classificação torna-se muito simples.

A decisão é feita em relação a medida da distância Euclidiana $\|\mathbf{x} - \boldsymbol{\mu}_i\|$, esta é calculada para cada uma das c classes e o vetor $\mathbf{x}$ é classificado em relação à sua proximidade da média. Essa classificação é conhecida como o classificador de distância mínima. Se para cada vetor médio for considerado um modelo para as classificações de padrões, este será um modelo de correspondência (BISHOP, 2006; DUDA et al, 2000; FRIEDMAN et al, 2008; JAIN et al, 2000; MARQUES, 2005).

2.1.3.5.2 Caso 2: $\Sigma_i = \Sigma$

Este caso ocorre quando a matriz de covariância não é estruturada, porém é idêntica para todas as classe, que pode ser visualizada na matriz a seguir, e corresponde a situação em que cada amostra pertence ao hiperplano dos grupos com o mesmo tamanho e formato. Os grupos formados estão centrados em torno do vetor de médias $\boldsymbol{\mu}_i$. Desde que ambos os termos, $|\Sigma_i|$ e $(d/2) \ln 2\pi$ sejam independentes de i , estes podem ser ignorados por serem constantes (DUDA et al, 2000; MARQUES, 2005).

$$\begin{pmatrix} \sigma_{11} & \dots & \sigma_{ij} \\ \vdots & \ddots & \vdots \\ \sigma_{ji} & \dots & \sigma_{ii} \end{pmatrix}$$

Deste modo a função discriminante é dada por:

$$g_i(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^t \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_i) + \ln P(\omega_i)$$

Se as probabilidades a priori $P(\omega_i)$ forem iguais para todas as c classes, então o logaritmo de $P(\omega_i)$ pode ser ignorado, deste modo a função discriminante torna-se mais simples de ser calculada. Para poder fazer a classificação do vetor $\mathbf{x}$ é calculada a distância de Mahalanobis $(\mathbf{x} - \boldsymbol{\mu}_i)^t \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)$ para cada $\mathbf{x}$ correspondente a cada um dos vetores das c médias, e realizada a classificação de $\mathbf{x}$ para a classe mais próxima da média. Assim como visto anteriormente para diferentes probabilidades a priori a decisão de classificação será em favor da priori mais provável (DUDA et al, 2000; MARQUES, 2005)

A expansão da forma quadrática $(\mathbf{x} - \boldsymbol{\mu}_i)^t \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)$ resulta em uma soma dos termos quadráticos $\mathbf{x}^t \Sigma^{-1} \mathbf{x}$ com independência dos i . Realizando este desenvolvimento a função novamente será uma função discriminante linear, dada por:

$$g_i(\mathbf{x}) = \mathbf{w}_i^t \mathbf{x} + \omega_{i0}$$

em que

$$\mathbf{w}_i = \Sigma^{-1} \boldsymbol{\mu}_i$$

e

$$\omega_{i0} = -\frac{1}{2} \boldsymbol{\mu}_i^t \Sigma^{-1} \boldsymbol{\mu}_i + \ln P(\omega_i)$$

Desde que as funções discriminante sejam lineares as fronteiras de decisão serão novamente hiperplanos (DUDA et al, 2000). Se $\mathcal{R}_i$ e $\mathcal{R}_j$ estão de modo sequencial, a decisão entre as duas classes será dada por:

$$\mathbf{w}^t(\mathbf{x} - \mathbf{x}_0) = 0$$

em que

$$\mathbf{w} = \Sigma^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)$$

e

$$\mathbf{x}_0 = \frac{1}{2}(\boldsymbol{\mu}_i + \boldsymbol{\mu}_j) - \frac{\ln[P(\omega_i)/P(\omega_j)]}{(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^t \Sigma^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)$$

Desde que $\mathbf{w} = \Sigma^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)$, o hiperplano que separa as regiões $\mathcal{R}_i$ e $\mathcal{R}_j$, geralmente não será ortogonal em relação a linha entre as médias. Porém, este valor cruza com a linha no ponto $\mathbf{x}_0$ com a mesma distância entre as médias e as probabilidades a priori são iguais, caso as probabilidades a priori não forem iguais então o hiperplano será deslocado para a média mais provável (BISHOP, 2006; DUDA et al, 2000; FRIEDMAN et al, 2008; MARQUES, 2005).

2.1.3.5.3 Caso 3: Σ_i =arbitrária

Este caso ocorre quando a matriz de covariância não é estruturada e é diferente para cada uma das classes, ou seja, para este tipo de caso o único termo que pode ser retirado é $(d/2) \ln 2\pi$, que semelhante para todas as funções discriminates, deste modo pode-se escrever a função discriminante como:

$$g_i(\mathbf{x}) = \mathbf{x}^t W_i \mathbf{x} + \mathbf{w}_i^t \mathbf{x} + \omega_{i0}$$

em que

$$W_i = -\frac{1}{2}\Sigma_i^{-1}$$

$$\mathbf{w}_i = \Sigma_i^{-1}\boldsymbol{\mu}_i$$

e

$$\omega_{i0} = -\frac{1}{2}\boldsymbol{\mu}_i^t \Sigma_i^{-1} \boldsymbol{\mu}_i - \frac{1}{2} \ln |\Sigma_i| + \ln P(\omega_i)$$

As decisões são hiperquadráticas, e podem assumir qualquer forma: um hiperplano, pares de hiperplanos, hiperesferas, hiperelipsóides, hiperparabolóides, quadráticas e hiperbolóides de vários tipos (DUDA et al, 2000).

A extensão destes resultados para mais de duas classes é simples mas é necessário que duas das c classes sejam responsáveis por qualquer segmento da fronteiras de decisão. Caso as distribuições sejam mais complexas, as regiões de decisão podem ser ainda mais complexas (DUDA et al, 2000; MARQUES, 2005)

2.1.3.6 Avaliação da classificação

Feita a classificação é necessária uma avaliação da mesma, ou seja, verificar a sua eficiência, para encontrar tal eficiência são utilizados métodos que utilizam as informações das classificações comparando-as com as suas verdadeiras classes, ou seja, são medidos os erros de classificação (DUDA et al, 2000; MARQUES, 2005). Os métodos que podem ser utilizados para verificar a eficiência da classificação são:

- 1) Índice Kappa;
- 2) Probabilidade de erro Geral.

Para cada uma das classificações possíveis é necessário realizar um tipo de método diferente para o calculo em relação aos erros. A seguir serão explicados como se calculam os valores para os métodos.

2.1.3.6.1 Índice Kappa

O índice Kappa, ou coeficiente kappa de Cohen, é uma medida de concordância usada em escalas nominais que fornece uma ideia do quanto as observações se afastam daquelas esperadas, fruto do acaso, indicando assim o quão legítimas as interpretações são (LOBÃO et al, 2005; WITTEN; FRANK, 2005)

Para o cálculo deste índice é utilizado a seguinte fórmula:

$$\kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)}$$

sendo que, $\Pr(a)$ corresponde a probabilidade de classificar as observações de modo correto e $\Pr(e)$ corresponde a probabilidade hipotética de ocorrer a classificação, ou seja, a classificação

esperada, para se calcular as probabilidades referentes a cada classe são utilizados os dados observados. Se todas as classificações forem feitas de modo correto o valor do índice kappa, κ , será $\kappa = 1$ e se todos as observações forem classificadas na mesma classe, então o valor de κ será $\kappa = 0$, desta maneira que o valor de κ sempre será $0 \leq \kappa \leq 1$.

Para facilitar a visualização de como foram feitas as classificações é criada uma matriz de confusão, sendo que a sua i -ésima linha representa a classificação realizada e a i -ésima coluna representa a classe verdadeira, assim as classificações corretas são encontradas na diagonal, desta maneira esta matriz é dada por:

$$\begin{pmatrix} y_{11} & y_{12} & \cdots & y_{1c} \\ y_{21} & y_{22} & \cdots & y_{2c} \\ \vdots & \vdots & \ddots & \vdots \\ y_{c1} & y_{c2} & \cdots & y_{cc} \end{pmatrix}.$$

Para encontrar os valores das probabilidades $\text{Pr}(a)$, como dito anteriormente, é utilizada é a soma da diagonal principal da matriz e é feita a divisão pelo total de observações a serem classificadas que é dada por:

$$\text{Pr}(a) = \frac{\sum_{i=1}^c y_{ii}}{n}$$

Com a matriz de confusão pode-se elaborar uma tabela com as classificações em relação às classes originais e com as totais de classificações, esta tabela pode ser utilizada para facilitar o modo como obter as probabilidade de classificação, a Tabela 1 mostra estas classificações:

Tabela 1 - Classificações em relação as classes verdadeiras

	Verdadeiro				
$\hat{\omega}$	1	2	...	c	Total
1	y_{11}	y_{12}	...	y_{1c}	$y_{1.}$
2	y_{21}	y_{22}	...	y_{2c}	$y_{2.}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
c	y_{c1}	y_{c2}	...	y_{cc}	$x_c.$
Total	$y_{.1}$	$y_{.2}$	...	$y_{.c}$	$y_{..}$

em que y_{ij} representa o total de classificações feitas de $\hat{\omega}_i$ em ω_j , $y_{.i}$ representa o total de classificações realizadas com $\hat{\omega}_i$, $y_{.i}$ representa o total de classificações em ω_i e $y_{..}$ representa o total de observações classificadas.

Para encontrarmos o valor de $\Pr(e)$ pode-se utilizar

$$\Pr(e) = \frac{\sum_{i=1}^c y_{i.} y_{.i}}{y_{..}^2}$$

para facilitar a compreensão destes cálculos será utilizado o exemplo 1 a seguir.

Seja uma matriz de confusão dada por:

$$\begin{pmatrix} 20 & 5 \\ 10 & 15 \end{pmatrix}$$

e a tabela auxiliar dada pela Tabela 2

Tabela 2 - Classificações em relação as classes verdadeiras

	Verdadeiro		
$\hat{\omega}$	1	2	Total
1	20	5	25
2	10	15	25
Total	30	20	50

para este caso pode-se observar que,

$$\Pr(a) = \frac{20 + 15}{50} = 0,7$$

e

$$\Pr(e) = \frac{25 \times 30}{50^2} + \frac{25 \times 20}{50^2} = 0,5$$

assim o valor do índice kappa será:

$$\kappa = \frac{0,7 - 0,5}{1 - 0,5} = 0,4$$

Com os valores de κ obtidos é necessário interpretá-los. A sua interpretação pode ser vista na Tabela 2.

Tabela 3 - Interpretação do índice Kappa

Valor de kappa	Interpretação
0,00	Péssima
0,00-0,20	Ruim
0,21-0,40	Razoável
0,41-0,60	Boa
0,61-0,80	Ótima
0,81-1,00	Excelente

2.1.3.6.2 Probabilidade de erro geral

Outro método para verificação do erro de classificação é a probabilidade de erro geral, para encontrar este é utilizada a seguinte fórmula (MARQUES, 2005).

$$P_e(\mathbf{x}) = 1 - \sum_{i=1}^c P_{ii}P(\omega_i)$$

Sendo que $P_{ii} = P\{\hat{\omega} = \omega_i | \omega = \omega_i\}$, que corresponde a probabilidade de ocorrer a classificação correta para a classe observada, essa probabilidade pode ser obtida utilizando os valores totais

de cada linha da Tabela 1 e com os valores da diagonal, deste modo pode-se escrever a P_{ii} como : $\frac{y_{ii}}{y_{i1}}$. o valor $P(\omega_i)$ representa a probabilidade a priori para a i -ésima classe.

Utilizando o exemplo 1 pode-se observar que $P_{11} = \frac{20}{25} = 0,8$ e $P_{22} = \frac{15}{25} = 0,6$, assumindo que para este caso como ambas as classes possuem a mesma chance de ocorrer o valor $P(\omega_i)$ será $1/2$, assim a probabilidade de erro geral será:

$$1 - 0,8 \times 1/2 + 0,6 \times 1/2 = 1 - 0,4 + 0,3 = 0,3$$

Para os casos em que os valores de $P(\omega_i) = P(\omega_j)$ a probabilidade de erro pode ser encontrada utilizando somente os valores da diagonal da matriz de confusão, ou seja, $1 - Pr(a)$

Para este caso o interesse será o oposto do índice kappa, ou seja, há o interesse no menor valor possível sendo que no índice kappa o interesse está em obter o maior índice.

2.1.4 Problemas de classificação

Ao utilizar a técnica de reconhecimento de padrões para realizar as classificações podem ocorrer alguns problemas, que podem dificultar a implementação do algoritmo de classificação ou até mesmo torná-lo inviável. Estes tipos de problemas podem ocorrer independentemente do tipo de aprendizado utilizado (BISHOP, 2006; DUDA et al, 2000; JAIN et al, 2000; MORAIS, 2010).

Como para a classificação de padrões há o interesse em otimizar os processos de classificações e conseqüentemente melhorar o seu desempenho é necessário verificar qual ou quais, são os tipos de problemas que podem surgir. Há três problemas que podem ocorrer em relação aos dados:

- 1) Sobre-ajuste (*overfitting*) relacionado com o número de parâmetros livres do classificador;
- 2) Sobre-treinamento (*overtraining*) relacionado com o número de iterações de treinamento;
- 3) Problema da dimensionalidade (*curse of dimensionality*) relacionado com a dimensão do espaço de características.

Para o desenvolvimento deste trabalho foi observado o problema da dimensionalidade. Este problema está relacionado com número de características utilizadas para repre-

sentar os padrões de uma determinada classe, ou seja, a dimensão do espaço das características d . O problema da dimensionalidade está diretamente relacionado com o número de elementos utilizados no aprendizado para que um classificador tenha um bom desempenho. Desde modo o desempenho do classificador está relacionado com a sua complexidade e a qualidade do conjunto utilizado para o aprendizado (MORAIS, 2010).

Tais problemas podem afetar os sistemas de reconhecimento de padrões por causa da sua relação com seleção de características, ou seja, em casos que se tenham uma grande quantidade de informações pode tornar a classificação muito demorada, que para tomadas de decisão pode se tornar crucial, como nos exemplos citados na seção 2.1.2, em que há a necessidade de detectar um incêndio ou um avião no radar (MARQUES, 2005; MORAIS 2010; JAIN et al, 2000).

Para os casos em que ocorrem os problemas de dimensionalidade pode-se utilizar de técnicas para solucionar esses problemas, pode-se realizar uma extração de características, utilizando a técnica de seleção de variáveis. Deste modo a seleção de variáveis é muito relevante, não importando qual o classificador ou tipo de aprendizado é utilizado (DUDA, 2000; CAMPOS, 2001; JAIN et al, 2000; MORAIS, 2010).

2.1.5 Seleção de Variáveis

Quando há um grande número de características pode-se obter classificadores que realizem uma classificação precisa, porém, pode ocorrer um aumento na complexidade do algoritmo. Porém se em um conjunto de dados com número de características pequeno pode-se obter classificadores que não realizem as classificações de modo preciso, caso que não é de interesse para as classificações. De um modo geral, deseja-se criar uma decisão de classificação que realize boas classificações utilizando o menor número possível de características. A classificação mais simples possível, como visto anteriormente, ocorre quando há apenas uma característica para ser avaliada e quando se utiliza todas as d características possíveis a classificação será a mais complexa de ser realizada (CAMPOS, 2001; DEMÉTRIO; ZOCCHI, 2010; JAIN et al, 2000; FARAWAY, 2002; MARQUES, 2005; MORAIS, 2010).

Deste modo a seleção das características está baseada em:

- 1) Um classificador com o maior número possível de variáveis, de tal forma que a predição

seja a melhor possível;

- 2) Um classificador com o menor número possível de variáveis necessárias para uma boa classificação, pois a obtenção de informações a respeito de um número grande de variáveis, muitas vezes, aumenta o custo.

Para realizar a seleção de variáveis, primeiramente é necessário criar um modelo de regressão com os dados de interesse, para a seleção, geralmente, são escolhidos os modelos com menor soma de quadrados de resíduos ($SQRes$), ou de forma equivalente com maior coeficiente de correlação (R^2). Para realizar esta seleção das características pode ser utilizado dois tipos procedimentos (DEMÉTRIO; ZOCCHI, 2010). São eles:

- 1) Métodos de seleção de subconjuntos das variáveis;
- 2) Métodos de seleção das variáveis passo a passo - incluem ou excluem variáveis sequencialmente.

Os métodos de seleção de variáveis passo a passo, podem ser;

- 1) Método do passo atrás (backward);
- 2) Método do passo a frente (forward);
- 3) Método do passo a frente passo atrás (stepwise).

2.1.5.1 Método do passo atrás (*backward*)

Este método consiste em realizar um ajuste inicialmente utilizando o modelo completo e em seguida eliminar as variáveis uma a uma, com:

- 1) Menor correlação parcial com a resposta $\mathbf{Y}$;
- 2) Menor diminuição no R^2 ;
- 3) Menor diminuição significativa no teste F parcial ou no teste t parcial;

Utilizando algum critério de parada, os passos para realizar essa seleção utilizando o teste F parcial são:

- 1) Passo 1: Ajustar o modelo completo com d variáveis e obter a Soma de quadrados de resíduos do modelo completo ($SQRes_c$) com $n - d$ graus de liberdade;
- 2) Passo 2: Para cada uma das d variáveis do modelo completo do Passo 1, considerar o modelo reduzido, com a retirada de uma variável e calcular a soma de quadrados do modelo reduzido ($SQRes_r$) com $n - d + 1$ graus de liberdade e $F = \frac{SQRes_r - SQRes_c}{QMRes_c}$;
- 3) Passo 3: Obter $F_{\text{mín}}$;
- 4) Passo 4: Comparar $F_{\text{mín}}$ com F_{sai} (percentil da tabela de F com 1 e $n - d$ graus de liberdade a um nível de significância $\alpha = 0,10$);
 - i) se $F_{\text{mín}} > F_{\text{sai}}$, não eliminar nenhuma variável e parar o processo, ficando o modelo completo com d variáveis;
 - ii) se $F_{\text{mín}} < F_{\text{sai}}$, eliminar a variável com $F_{\text{mín}}$ e voltar ao Passo 1 utilizando o novo modelo como o modelo completo com $d = d - 1$ variáveis.

2.1.5.2 Método do passo a frente (*forward*)

O método de seleção *forward* consiste em incluir inicialmente no modelo a variável com maior coeficiente de correlação simples com a variável resposta e a seguir as demais variáveis, uma a uma, com:

- 1) maior correlação parcial com a resposta Y ;
- 2) maior aumento no R^2 ;
- 3) maior aumento significativo no teste F parcial ou no teste t parcial

Utilizando algum critério de parada, os passos para realizar essa seleção utilizando o teste F parcial são:

- 1) Passo 1: Ajustar o modelo reduzido com m variáveis e obter $SQRes_r$ com $n - m$ graus de liberdade;
- 2) Passo 2: Para cada uma das variáveis que não pertencerem ao modelo utilizado no Passo 1, considerar o modelo completo, com a adição de uma variável extra e calcular $SQRes_c$ com $n - m - 1$ graus de liberdade e $F = \frac{SQRes_r - SQRes_c}{QMRes_c}$

- 3) Passo 3: Obter $F_{\text{máx}}$
- 4) Passo 4: Comparar $F_{\text{máx}}$ com F_{en} (percentil da tabela de F com 1 e $n - m - 1$ graus de liberdade a um nível de significância $\alpha = 0,10$)
 - i) se $F_{\text{máx}} > F_{\text{en}}$, deve-se incluir a variável com $F_{\text{máx}}$ e em seguida realizar o Passo 1 utilizando o novo modelo como o modelo reduzido com $m = m + 1$ variáveis;
 - ii) se $F_{\text{máx}} < F_{\text{en}}$, não incluir a variável com o $F_{\text{máx}}$ e parar o processo, ficando assim com o modelo reduzido com m variáveis.

2.1.5.3 Método do passo a frente passo atrás (*stepwise*)

Esse método consiste na mistura dos dois anteriores. Para realizar este método devem-se, baseados no teste F parcial, serem seguidos os seguintes passos:

- 1) Passo 1: Ajustar o modelo reduzido com m variáveis e obter $SQRes_r$ com $n - m$ graus de liberdade.
- 2) Passo 2: Para cada uma das variáveis que não pertencerem ao modelo utilizado no Passo 1, considerar o modelo completo, com a adição de uma variável extra e calcular $SQRes_c$ com $n - m - 1$ graus de liberdade e $F = \frac{SQRes_r - SQRes_c}{QMRes_c}$
- 3) Passo 3: Obter $F_{\text{máx}}$
- 4) Passo 4: Comparar $F_{\text{máx}}$ com F_{en} (percentil da tabela de F com 1 e $n - m - 1$ graus de liberdade a um nível de significância $\alpha = 0,10$)
 - i) se $F_{\text{máx}} > F_{\text{en}}$, incluir a variável com $F_{\text{máx}}$ realizar o Passo 5 utilizando o novo modelo como o modelo completo com $d = m + 1$ variáveis;
 - ii) se $F_{\text{máx}} < F_{\text{en}}$, não incluir a variável com $F_{\text{máx}}$ realizar o Passo 5 utilizando o novo modelo como o modelo completo com $d = m$ variáveis.
- 5) Passo 5: Ajustar o modelo completo com d variáveis e obter $SQRes_c$ com $n - d$ graus de liberdade.

- 6) Passo 6: Para cada uma das d variáveis do modelo completo do Passo 4, considerar o modelo reduzido, com a retirada de uma variável e calcular $SQRes_r$ com $n - d + 1$ graus de liberdade e $F = \frac{SQRes_r - SQResc}{QMResc}$
- 7) Passo 7: Obter $F_{\min}$
- 8) Passo 8: Comparar $F_{\min}$ com F_{sai} (percentil da tabela de F com 1 e $n - l$ graus de liberdade a um nível de significância $\alpha = 0,10$)
 - i) se $F_{\min} > F_{\text{sai}}$, não eliminar nenhuma variável e voltar ao Passo 1 utilizando o novo modelo como o modelo reduzido com $m = d$ variáveis e parar o processo se no Passo 4 nenhuma variável for incluída;
 - ii) se $F_{\min} < F_{\text{sai}}$, eliminar a variável com $F_{\min}$ e voltar ao Passo 1 com novo modelo reduzido com $m = d - 1$ variáveis.

2.2 Material e Métodos

Foram utilizados três conjuntos de dados: 1) referentes a marcadores moleculares, desenvolvido pelo programa de melhoramento de milho do Departamento de Genética - ESALQ/USP, pelo professor Dr. Cláudio Lopes de Souza Jr. 2) dados de soja obtidos no repositório UCI Machine Learning Repository e 3) dados referentes a embalagens de peças de carro de uma autopeças, fornecidos pela professora Dra. Juliana Garcia Cespedes.

Para realizar as classificações dos dados foi utilizado um aprendizado supervisionado, com a método de classificação bayesiana, as implementações dos programas foram de realizadas de dois modos: 1) Utilizando métodos implementados nos *software WEKA* e *R*, por meio de programas implementados passo a passo que podem ser vistos no Apêndice B. Para o conjunto de dados das embalagens de peças de carro foi realizada a técnica de seleção de características.

A seguir apresenta-se uma pequena descrição sobre os conjuntos de dados utilizados.

2.2.1 Marcadores Moleculares

Os estudos de divergência genética e relações filogenéticas entre espécies têm merecido uma atenção cada vez mais maior, principalmente com o advento dos marcadores moleculares, nesses estudos, os pesquisadores têm o interesse de agrupar indivíduos semelhantes, de forma que as maiores diferenças ocorram entre os grupos formados e menores diferenças dentro dos grupos (DUARTE et al, 1999; MEYER, 2002).

Os resultados obtidos com o emprego dos marcadores moleculares fornecem uma matriz de valores binários, em que as colunas da matriz representam os diferentes genótipos e as linhas representam as diferentes características a serem comparadas. No caso dos dominantes, cada linha corresponde à identificação de uma marca (ou banda) numa região específica do DNA. A presença da marca é identificada pelo número 1 e a ausência pelo número 0 (MEYER, 2002).

Pode ser obtida a seguinte matriz:

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

em que p e n indicam o número de marcas e de genótipos respectivamente. Quando dois genótipos são comparados podem ocorrer as seguintes situações, para cada marca p :

- 1) ambos possuem a mesma marca (código 1 e 1);
- 2) um genótipo possui a característica e o outro não (código 1 e 0 ou 0 e 1);
- 3) ambos não possuem tal característica (código 0 e 0)

Para a realização da análise foram utilizadas 18 linhagens desenvolvidas pelo programa de melhoramento de milho do Departamento de Genética-ESALQ/USP, pelo professor Dr. Cláudio Lopes de Souza Jr. As linhagens são provenientes das populações dos milhos BR-105 e BR-106, desenvolvidas pelo Centro Nacional de Milho e Sorgo (Embrapa Milho e Sorgo). A origem das linhagens é apresentada pela Tabela 4, sendo que 8 linhagens originam-se da população BR-105 e 10 da população BR-106.

Tabela 4 - Origem das 18 linhagens

Origem	Linhagens (código de uso)
BR-105	1 2 3 4 5 6 7 8
BR-106	9 10 11 12 13 14 15 16 17 18

2.2.2 Dados de soja

Outro conjunto de dados para a realização das análises foi um conjunto de dados de soja, este conjunto pertence ao site <<http://archive.ics.uci.edu/ml/>>, o repositório da Universidade da Califórnia, Irvine (UCI), neste encontram-se diversos tipos conjunto de dados para a realização das técnicas de reconhecimento de padrões desde dados referentes a jogos, como pôquer e jogo da velha, ao de reconhecimento de imagens.

Os dados de soja que foram utilizados para a análise correspondem a 19 classes de danos à soja, entre elas: podridão de carvão, podridão radicular de *Rhizoctonia*, podridão parda da haste, Mancha Marrom, pústula bacteriana, sementes de coloração púrpura, mancha foliar, lesão 2-4-d e lesão de herbicida, foram observadas 35 características entre elas: data, o estado de plantas, precipitação, temperatura, granizo, área danificada, severidade, germinação, crescimento das plantas, folhas, tamanho de mancha foliar, caule, cancro da lesão, corpos de frutificação, deterioração externa, manchas de frutas, sementes, desenvolvimento de fungos, tamanho das sementes tamanho, murchamento, raízes, sendo que estas podem ser do tipo discreto ou contínuo, totalizando 307 observações. Para a obtenção destes dados basta entrar no site <<http://archive.ics.uci.edu/ml/machine-learning-databases/soybean/soybean-large.data>>.

2.2.3 Dados de peças de carros

O reconhecimento de padrões pode ser muito útil para otimizar problemas específicos, para o conjunto de dados das peças de automóvel há o interesse de classificar as peças utilizando as suas embalagens, ou seja, ao observar uma embalagem e colocar a mesma em sua verdadeira classe. Quando este processo de classificação for bem sucedido há poucas peças clas-

sificadas e modo errado e também pode-se “ganhar” tempo no momento de armazená-las, pois com o algoritmo implementado corretamente a quantidade de embalagens em setores errados será reduzida.

Os dados utilizados para esta análise foram cedidos pela professora Dra. Juliana Garcia Cespedes. Os dados correspondem a embalagens de peças de automóveis e estão divididas em 9 classes e o vetor de característica possui dimensão 16, sendo elas: código da fábrica, código de barras, formato da embalagem, composição, luminosidade, formato da peça, cor da predominante da peça, cor predominante na embalagem, número de cores na embalagem, largura (cm), comprimento (cm), altura (cm), fabricante da peça, originalidade, marca do carro e índice de vendas no mês, totalizando 88 observações.

Para a classificação foram descartadas inicialmente 9 características: código da fábrica, código de barras, luminosidade, formato da peça, cor da predominante da peça, fabricante da peça, originalidade, marca do carro e índice de vendas no mês, pois estas não seriam de utilidade, sendo que luminosidade, formato da peça e cor da predominante da peça só podem ser observadas para um tipo de composição da embalagem. Para as 7 características restantes a análise foi realizada por dois métodos: 1) sem seleção de característica e 2) com seleção de característica.

Como os dados analisados eram de tipos discretos e contínuos foi necessário, para a implementação manual, realizar algumas transformações em relação aos dados discretos, criando vetores de (0 e 1). Essas transformações são referentes às características: formato da embalagem, composição, cor predominante na embalagem, número de cores na embalagem.

As transformações realizadas para essas características foram feitas da seguinte forma:

- 1) Formato da embalagem: São possíveis 3 formatos de embalagem, retangular, quadrada e irregular. Desta maneira foram criados 3 vetores para “transformar” estas informações. Para a embalagem quadrada foi criado o vetor (0 0 1), para a embalagem retangular foi criado o vetor (1 0 0) e para a embalagem irregular foi criado o vetor (0 1 0);
- 2) Composição: São possíveis 2 tipos de composição, papelão ou plástico. Desta maneira foram criados 2 vetores para “transformar” estas informações. Para a composição de papelão foi criado o vetor (1 0) e para composição de plástico foi criado o vetor (0 1);

- 3) Cor predominante: São possíveis 8 cores predominantes, vermelho, preto, amarelo, azul, branco, verde, cinza e laranja e foram criados os seguintes vetores respectivamente (1 0 0 0 0 0 0), (0 1 0 0 0 0 0), (0 0 1 0 0 0 0), (0 0 0 1 0 0 0), (0 0 0 0 1 0 0), (0 0 0 0 0 1 0), (0 0 0 0 0 0 1);
- 4) Número de cores na embalagem: São possíveis 7 valores para o número de cores, 2, 3, 4, 5, 6, 8, 9 e foram criados os seguintes vetores, respectivamente (1 0 0 0 0 0 0), (0 1 0 0 0 0 0), (0 0 1 0 0 0 0), (0 0 0 1 0 0 0), (0 0 0 0 1 0 0), (0 0 0 0 0 1 0), (0 0 0 0 0 0 1).

2.2.4 *Software*

Para a realização deste trabalho foram utilizados dois *software*: *Weka* e *R*. Ambos *software* são de domínio público, ou seja, podem ser obtidos gratuitamente.

O *software Weka* (*Waikato Environment for Knowledge Analysis*) foi criado em 1993, na Universidade de Wakato, Nova Zelândia, surgiu pela necessidade de uma abordagem unificada para permitir um fácil acesso para pesquisadores da área de reconhecimento de padrões (HALL et al, 2009).

A análise dos conjuntos de dados foi realizada por meio do *Weka* versão 3.7, com o classificador bayesiano e os comandos *naiveBayes* e *naiveBayesMultinomial* e o *software R* - versão 2.11.1 utilizando-se: 1) a biblioteca *e1071*, a qual é específica para a classificação de padrões e 2) um programa implementado com códigos próprios que, a partir deste momento, será designado por PRP (Programa de Reconhecimento de Padrões), os códigos estão disponíveis no Apêndice B. Para selecionar as variáveis, quando necessário, foi utilizada a biblioteca *leaps*

2.3 Exemplo

Para uma melhor compreensão do método de reconhecimento de padrões será apresentado um exemplo com o conjunto de dados Íris, disponível no banco de dados de ambos os *software*. Este conjunto de dados corresponde a um conjunto de 150 observações divididos em 3 classes: setosa, versicolor e virginica. Para cada classe foram medidas 4 características: comprimento da sépala (*Sepal.Length*), largura da sépala (*Sepal.Width*), comprimento da pétala (*Petal.Length*) e largura da pétala (*Petal.Width*).

Para a fase de aprendizado considerado um conjunto com 66% do total dos dados disponíveis. Após a análise com o classificador Naive Bayes do *software Weka* foi obtida a seguinte matriz de confusão:

$$\begin{pmatrix} 15 & 0 & 0 \\ 0 & 18 & 1 \\ 0 & 2 & 15 \end{pmatrix}$$

a i -ésima linha representa a classificação realizada e i -ésima coluna representa a classe verdadeira, desta maneira verifica-se, por meio da diagonal principal, que 48 observações foram classificadas corretamente, sendo 15 para a classe setosa, 18 para a classe versicolor e 15 para a classe virginica, foram classificadas 3 observações erroneamente, sendo que uma observação da classe versicolor foi considerada como virginica e duas observações da classe virginica foram consideradas como versicolor.

Para este caso o índice kappa obtido foi 0,9113, que representa uma excelente classificação e a probabilidade de erro de classificação foi de 0,0567, portanto um valor muito baixo.

Por meio da biblioteca e1071 implementada no *software R* matriz de confusão obtida foi:

$$\begin{pmatrix} 19 & 0 & 0 \\ 0 & 14 & 0 \\ 0 & 3 & 15 \end{pmatrix}$$

desta maneira verifica-se, por meio da diagonal principal, que 48 observações foram classificadas corretamente, sendo 19 para a classe setosa, 14 para a classe versicolor e 15 para a classe virginica, foram classificadas 3 observações erroneamente, sendo que três observações da classe virginica foram consideradas como versicolor.

Para este caso o índice kappa obtido foi 0,9117, que representa uma excelente classificação e a probabilidade de erro de classificação foi de 0,0555, portanto um valor muito baixo.

Considerando-se que os componentes do vetor de característica são independentes

e que cada classe possui a mesma matriz de covariância $\sigma^2\mathbf{I}$ (caso 1) foi obtida a seguinte matriz de confusão utilizando-se o programa PRP.

$$\begin{pmatrix} 17 & 0 & 0 \\ 0 & 15 & 0 \\ 0 & 6 & 13 \end{pmatrix}$$

desta maneira verifica-se, por meio da diagonal principal, que 45 observações foram classificadas corretamente, sendo 17 para a classe setosa, 15 para a classe versicolor e 13 para a classe virginica, foram classificadas 6 observações erroneamente, sendo que seis observações da classe virginica foram consideradas como versicolor.

Para este caso o índice kappa obtido foi 0,8251, que representa uma excelente classificação e a probabilidade de erro de classificação foi de 0,1176, portanto um valor baixo.

Considerando-se que existe dependência entre os componentes do vetor de característica e que cada classe possui a mesma matriz de covariância Σ (caso 2) foi obtida a seguinte matriz de confusão utilizando-se o programa PRP.

$$\begin{pmatrix} 17 & 0 & 0 \\ 0 & 12 & 3 \\ 0 & 4 & 15 \end{pmatrix}$$

desta maneira verifica-se, por meio da diagonal principal, que 44 observações foram classificadas corretamente, sendo 17 para a classe setosa, 12 para a classe versicolor e 14 para a classe virginica, foram classificadas 7 observações erroneamente, sendo que três observações da classe versicolor foi considerada como virginica e quatro observações da classe virginica foram consideradas como versicolor.

Para este caso o índice kappa obtido foi 0,7936, que representa uma ótima classificação e a probabilidade de erro de classificação foi de 0,1372, portanto um valor baixo.

Considerando-se que existe dependência entre os componentes do vetor de característica e que cada classe possui a sua própria matriz de covariância Σ_i , $i = 1, 2, 3$ (caso 3) foi obtida a seguinte matriz de confusão utilizando-se o programa PRP.

$$\begin{pmatrix} 17 & 0 & 0 \\ 0 & 15 & 0 \\ 0 & 13 & 6 \end{pmatrix}$$

desta maneira verifica-se, por meio da diagonal principal, que 38 observações foram classificadas corretamente, sendo 17 para a classe setosa, 15 para a classe versicolor e 6 para a classe virginica, foram classificadas 13 observações erroneamente, sendo seis observações da classe virginica foram consideradas como versicolor.

Para este caso o índice kappa obtido foi 0,6271, que representa uma ótima classificação e a probabilidade de erro de classificação foi de 0,2549, portanto um valor razoável.

Pode-se verificar pelos resultados anteriores que para todos os métodos utilizados não ocorreu erro para a classe setosa, porém os resultados das classificações foram diferentes, sendo que foram obtidas três classificações como excelente e duas classificações como ótima, os valores da probabilidade de erro aumentaram de acordo como método utilizado, deste modo para este caso não é recomendado utilizar uma Σ não estruturada, ou seja, os melhores resultados obtidos ocorreram ao assumir independência entre as características.

3 RESULTADOS E DISCUSSÃO

Para a realização das análises, como mencionado anteriormente, foram utilizados três conjuntos de dados. Um conjunto é referente a marcadores moleculares dominantes, o outro conjunto é referente a embalagens de peças de carro de uma autopeça e outro conjunto é referente a dados de soja adquiridos no repositório UCI.

3.1 Análise de marcadores moleculares

O vetor de características para cada indivíduo, nesse caso cada linhagem, é composto por 162 características binárias (0,1), portanto, todas discretas e com 18 observações, sendo 8 da linhagem BR-105, classe 1, e 10 da linhagem BR-106, classe 2. As classificações para estes dados foram feitas por meio de conjuntos de treino de tamanhos diferentes, com 20%, 40%, 60%, 80% do tamanho total de observações, com o objetivo de classificar um determinado indivíduo como pertencente a classe 1, BR-105, ou a classe 2, BR-106.

Resultados no *R*

Utilizando o programa PRP desenvolvido no *software R* foram obtidas as seguintes matrizes de confusão para os diferentes tamanhos de conjunto de treino:

$$\begin{pmatrix} 0 & 7 \\ 0 & 8 \end{pmatrix}, \begin{pmatrix} 4 & 1 \\ 1 & 5 \end{pmatrix}, \begin{pmatrix} 3 & 1 \\ 0 & 4 \end{pmatrix}, \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix},$$

desta maneira verifica-se, por meio das diagonais principais, que:

- 1) Para conjunto de treino de 20% foram realizadas 8 classificações corretas e 7 erradas, todas observações foram classificadas como pertencente à classe 2;
- 2) Para conjunto de treino de 40% foram realizadas 9 classificações corretas e 2 erradas, foi obtida uma classificação errada para cada classe;
- 3) Para conjunto de treino de 60% foram realizadas 7 classificações corretas e 1;
- 4) Para conjunto de treino de 80% foram realizadas 4 classificações corretas e nenhuma errada, neste caso ocorreu a perfeição do classificador.

Os valores para a probabilidade de de erro geral e para o índice kappa podem ser vistos, respectivamente nas Tabelas 5 e 6.

Tabela 5 - Probabilidade de erro geral para o tamanho do treino pelo programa PRP

Tamanho do treino	erro
20%	0,5000
40%	0,1833
60%	0,1250
80%	0,0000

Tabela 6 - Índice kappa para o tamanho do treino no pelo programa PRP

Tamanho do treino	índice kappa	Avaliação
20%	0,0000	Péssima
40%	0,6333	Ótima
60%	0,7500	Ótima
80%	1,0000	Excelente

Resultados no *WEKA*

O resultado da análise realizada no *software WEKA* foram as seguintes matrizes de confusão para os diferentes tamanhos de conjunto de treino

$$\begin{pmatrix} 5 & 1 \\ 2 & 5 \end{pmatrix}, \begin{pmatrix} 4 & 1 \\ 2 & 4 \end{pmatrix}, \begin{pmatrix} 3 & 0 \\ 1 & 3 \end{pmatrix}, \begin{pmatrix} 2 & 0 \\ 1 & 1 \end{pmatrix},$$

desta maneira verifica-se, por meio das diagonais principais, que:

- 1) Para conjunto de treino de 20% foram realizadas 10 classificações corretas e 3 erradas foi obtida 1 classificação errada para a classe 1 e 2 classificações erradas para a classe 2;
- 2) Para conjunto de treino de 40% foram realizadas 8 classificações corretas e 3 erradas foi obtida 1 classificação errada para a classe 1 e 2 classificações erradas para a classe 2;

- 3) Para conjunto de treino de 60% foram realizadas 6 classificações corretas e 1 errada foi obtida 1 classificação errada para a classe 2;
- 4) Para conjunto de treino de 80% foram realizadas 3 classificações corretas 1 errada foi obtida 1 classificação errada para a classe 2.

Os valores para a probabilidade de de erro geral e para o índice kappa podem ser vistos, respectivamente nas Tabelas 7 e 8.

Tabela 7 - Probabilidade de erro geral para o tamanho do treino no *WEKA*

Tamanho do treino	erro
20%	0,2261
40%	0,2667
60%	0,1667
80%	0,2500

Tabela 8 - Índice kappa para o tamanho do treino no *WEKA*

Tamanho do treino	índice kappa	Avaliação
20%	0,5714	Boa
40%	0,4590	Boa
60%	0,7200	Ótima
80%	0,5000	Boa

Avaliação da classificação

Para avaliar as classificações realizadas nas tabelas foram utilizados dois métodos: a probabilidade geral de erro e o índice kappa.

Pelas Tabela 9 e 10 são apresentados todos os erros e os índices kappa obtidos para as classificações feitas em ambos os *software*.

Tabela 9 - Probabilidade de erro em relação ao tamanho do treino

Tamanho do treino	WEKA	PRP
20%	0,2261	0,5000
40%	0,2667	0,1833
60%	0,1667	0,1250
80%	0,2500	0,0000

Tabela 10 - Índice kappa em relação ao tamanho do treino

Tamanho do treino	WEKA	Avaliação	PRP	Avaliação
20%	0,5714	Boa	0,000	Péssima
40%	0,4590	Boa	0,6333	Ótima
60%	0,7200	Ótima	0,7500	Ótima
80%	0,5000	Boa	1,0000	Excelente

Nos gráficos apresentados nas Figura 6 e Figura 7 podem-se verificar o comportamento do erro e do índice kappa em relação ao tamanho do treino utilizado.

Figura 6 - Valores da probabilidade para os diferentes tamanhos de conjunto de treino

Figura 7 - Valores do índice kappa para os diferentes tamanhos de conjunto de treino

Pelas Tabelas 9 e 10 e pelas Figuras 6 e 7 pode-se observar que:

- 1) A probabilidade de erro utilizando o *WEKA* pouco se altera com os diferentes tamanhos de treino assim como o índice kappa, porém obteve um menor erro, 0,1667, e um maior valor do índice kappa, 0,72, consequentemente, uma melhor classificação, para o treino com tamanho de 60%.
- 2) A probabilidade de erro utilizando o *R* diminui de acordo com o tamanho do treino, sendo que para o treino de 80% o valor do erro foi de 0, e o índice kappa, 1, deste modo pode-se verificar que não ocorreram classificações erradas.
- 3) Para as análises com conjunto de treino de tamanho 60% e 80% a quantidade de observações para a validação foi pequena, sendo 8 e 4 respectivamente, deste modo, cada erro de classificação tem uma grande representatividade. Este fato é visto ao se olhar para a matriz de confusão para o treino de 80%, utilizando o *WEKA*, em que somente uma observação foi classificada erroneamente, porém, o erro foi maior que para uma treino de 60%, que também, teve apenas uma classificação errada.
- 4) As diferenças entre as classificações em ambos os *software* ocorre pelo fato do método como são elaborados os conjuntos de treino, pois os conjuntos podem ser compostos de observações diferentes, assim pode ocorrer de em um conjunto de treino ter “melhores” observações que o outro.

3.2 Análise para dados de soja

O vetor de características para cada indivíduo é composto por 35 características, sendo que são de variáveis contínuas e discretas, as observações pertencem ao conjunto de 19 classes, com um total de 307 observações. Para a análise desse conjunto de dados foi elaborado um conjunto de treino com 80% do total das observações, os resultados obtidos foram:

Resultados no *R*

A análise foi realizada no *software R* utilizando-se a biblioteca e1071 sendo obtidas a seguinte matriz de matriz de confusão:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 11 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 4 \end{pmatrix}$$

pode-se verificar que, somando-se os elementos da matriz de confusão que não pertencem a diagonal principal que foram obtidas 17 classificações erradas, somando os elementos da diagonal principal da matriz de confusão que foram obtidos 45 classificações corretas, sendo que não foram obtidas classificações para a classe 1, para a classe 6 e classe 14. O caso de não ocorrência de classificações em uma determinada classe é devido a utilização de todos os elementos da classe na fase de de treino.

Resultados no *WEKA*

A matriz de confusão obtida utilizando-se o *software WEKA* é dada por:

$$\begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

pode-se verificar que, somando-se os elementos da matriz de confusão que não pertencem a diagonal principal que foram obtidas 5 classificações erradas, somando os elementos da diagonal principal da matriz de confusão que foram obtidos 56 classificações corretas, sendo que não foi obtida nenhuma classificação para a classe 18.

Avaliação da classificação

Para realizar a avaliação das classificações foram utilizados os dois métodos propostos, a probabilidade geral de erro e o índice kappa.

Pode-se verificar pelas Tabela 11 e 12, os erros e os índices kappa obtidos para as classificações feitas em ambos os *software*.

Pela Tabela 11 pode-se observar que o *software WEKA* obteve um menor erro de classificação, e consequentemente foi mais eficaz. Pela Tabela 12 é observado que o *WEKA* obteve o melhor resultado, sendo que o *WEKA* teve uma excelente classificação, enquanto, o *R* obteve uma ótima classificação.

Tabela 12 - Índice kappa

Tabela 11 - Probabilidade de erro

<i>WEKA</i>	<i>R</i>
0,1027	0,2742

<i>WEKA</i>	Classificação	<i>R</i>	Classificação
0,9088	Excelente	0,7019	Ótima

3.3 Análise das peças de Carro

O vetor de características para cada indivíduo é composto por 7 características, sendo que são de 3 variáveis contínuas e 4 discretas, as observações pertencem ao conjunto de 9 classes, com um total de 88 observações. Para a análise desse conjunto de dados foi elaborado um conjunto de treino com 72% do total das observações. A análise foi realizada com o programa PRP e a biblioteca e1071 do *software R* e no *software WEKA* e foram realizados dois métodos para observar as observações, sem e com seleção de características.

Resultados no *R*

Para realizar o programa PRP no *software R* foram necessárias as seguintes considerações, para a parte multivariada:

1) Caso 1: $\Sigma = \sigma^2 I$

2) Caso 2: $\Sigma_i = \Sigma$

Resultados sem seleção de características

As classificações realizadas utilizando o programa PRP para o primeiro caso pode ser vista na matriz de confusão a seguir:

$$\begin{pmatrix} 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix}$$

pode-se verificar que, somando-se os elementos da matriz de confusão que não pertencem a diagonal principal que foram obtidas 13 classificações erradas, somando os elementos da diagonal principal da matriz de confusão que foram obtidos 11 classificações corretas.

As classificações realizadas por meio do programa PRP para o segundo caso pode ser vista na matriz de confusão a seguir:

$$\begin{pmatrix} 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix}$$

pode-se verificar que, somando-se os elementos da matriz de confusão que não pertencem a diagonal principal que foram obtidas 9 classificações erradas, somando os elementos da diagonal principal da matriz de confusão que foram obtidos 15 classificações corretas.

As classificações realizadas por meio da biblioteca e1071 do *R* pode ser vista pela matriz de confusão a seguir:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix}$$

pode-se verificar que, somando-se os elementos da matriz de confusão que não pertencem a

diagonal principal que foram obtidas 7 classificações erradas, somando os elementos da diagonal principal da matriz de confusão que foram obtidos 18 classificações corretas, sendo que não foram obtidas classificações para as classes 1 e 6.

Resultados com seleção de características

Para esta análise foi realizada a seleção de características por meio da biblioteca *leaps*, foram selecionadas 4 características são elas: formato da embalagem, cor predominante, largura e altura da caixa.

As classificações realizadas por meio do programa PRP para o primeiro caso podem ser vista pela matriz de confusão a seguir:

$$\begin{pmatrix} 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix}$$

pode-se verificar que, somando-se os elementos da matriz de confusão que não pertencem a diagonal principal que foram obtidas 11 classificações erradas, somando os elementos da diagonal principal da matriz de confusão que foram obtidos 13 classificações corretas.

As classificações realizadas por meio do programa PRP para o segundo caso

podem ser vista pela matriz de confusão a seguir:

$$\begin{pmatrix} 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix}$$

pode-se verificar que, somando-se os elementos da matriz de confusão que não pertencem a diagonal principal que foram obtidas 9 classificações erradas, somando os elementos da diagonal principal da matriz de confusão que foram obtidos 15 classificações corretas.

As classificações realizadas por meio da biblioteca do *R* podem ser vista pela matriz de confusão a seguir:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{pmatrix}$$

pode-se verificar que, somando-se os elementos da matriz de confusão que não pertencem a diagonal principal que foram obtidas 8 classificações erradas, somando os elementos da diagonal principal da matriz de confusão que foram obtidos 17 classificações corretas, sendo que não foi realizada classificação para a classe 6.

Resultados no *WEKA*

Resultados sem seleção de características

As classificações realizadas podem ser vista pela matriz de confusão a seguir:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 \end{pmatrix}$$

pode-se verificar que, somando-se os elementos da matriz de confusão que não pertencem a diagonal principal que foram obtidas 5 classificações erradas, somando os elementos da diagonal principal da matriz de confusão que foram obtidos 20 classificações corretas, sendo que não foi realizada classificação para a classe 6.

Resultados com seleção de características

As classificações realizadas podem ser vista pela matriz de confusão a seguir:

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 4 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 \end{pmatrix}$$

pode-se verificar que, somando-se os elementos da matriz de confusão que não pertencem a

diagonal principal que foram obtidas 6 classificações erradas, somando os elementos da diagonal principal da matriz de confusão que foram obtidos 19 classificações corretas.

Avaliação da classificação

Para realizar a avaliação das classificações realizadas foram utilizados dois métodos, a probabilidade geral de erro e o índice kappa. Pode-se verificar Pelas Tabelas 13 e 14 o desempenho em relação as classificações com e sem o processo de seleção de características.

Tabela 13 - Probabilidade de erro

Método	Sem Seleção	Com seleção
PRP com o Caso 1	0,54167	0,5000
PRP com o Caso 2	0,29167	0,3750
Biblioteca do <i>R</i>	0,2800	0,3200
<i>Weka</i>	0,2541	0,3310

Tabela 14 - Valores e classificação do índice kappa

Método	Sem Seleção	Classificação	Com seleção	Classificação
PRP com Caso 1	0,3918	Razoável	0,4418	Boa
PRP com Caso 2	0,6725	Ótima	0,5814	Boa
Biblioteca do <i>R</i>	0,6747	Ótima	0,6255	Ótima
<i>Weka</i>	0,7681	Ótima	0,7212	Ótima

Pelas Tabelas 13 e 14 e pela Figura 4 pode-se observar que:

- 1) Para este caso não foi utilizado o Caso 3 para normal multivariada, em que cada classe possui uma matriz de covariância, pois algumas classes possuíam um número reduzido de observações, como por exemplo filtro de óleo com apenas 4 observações, e neste caso não foi possível encontrar uma inversa única.

- 2) Os valores utilizados para o conjunto de treino e consequentemente para validação foram diferentes, pois consistia de utilizar um tamanho de conjunto de treino de 72%, deste modo as observações para o trino podem ser 63 ou 64.
- 3) As probabilidades de erro para a análise sem realizar a seleção de características obtiveram um valor alto, sendo que para o Caso 1 o erro superou os 50%, o valor do índice kappa foi de 0,3918, o que indica uma classificação razoável enquanto para os demais métodos as classificações foram consideradas ótimas. Este resultado mostra, para este caso em particular, que utilizar uma matriz $\sigma^2 \mathbf{I}$ não é uma boa alternativa.
- 4) As probabilidades de erro para a análise com a seleção de características obtiveram um valor maior que em relação às análises sem seleção, exceto para o caso 1, sendo que não foi observada nenhuma probabilidade de erro inferior à 30%, as classificações para o índice kappa foram alteradas, sendo que há duas classificações ótimas e duas boas.
- 5) Pode-se observar que ao realizar a seleção de variável os resultados obtidos sofreram uma alteração negativa, ou seja, ocorreram mais erros, isto pode ocorrer pelo fato de se desconsiderar alguma informação importante para a classificação. Pode-se observar que entre os métodos de classificação utilizados os que obtiveram os melhores resultados foram o programa PRP com o caso 2, a biblioteca *R* e o programa no itaWEKA, para a classificação sem seleção de característica, para a classificação com a seleção de caraterística os melhores resultados foram obtidos por meio da biblioteca do *R* e do *software WEKA*.

4 CONCLUSÕES

Após avaliar os resultados para as classificações feitas pode-se concluir que o método de reconhecimento de padrões, utilizando um aprendizado supervisionado com a técnica de classificador bayesiano, gerou bons resultados, independentemente dos procedimentos e dos *software* utilizados.

Pode-se observar que o *software WEKA* é de fácil manuseio, tornando a análise mais simples de ser realizada, este possui bons resultados e o tempo para a obtenção destes, nesse caso, foi pequeno, que é um dos objetivos dos reconhecimentos de padrões.

Pode-se observar que *software R*, quando utiliza-se os comandos já implementados o manuseio é fácil, e são obtidos bons resultados e de modo rápido. Neste *software* há a possibilidade de se elaborar os algoritmos de decisão, pode-se criar classificações mais elaborados, como os casos em que são utilizados dados multivariados.

Pode-se concluir que o método de reconhecimento é um bom método para classificações de dados, pois obteve bons resultados de modo rápido, mostrando assim ser uma técnica eficiente para resolução de problemas.

É de interesse posteriormente realizar um estudo mais detalhado sobre o assunto de reconhecimento de padrões, utilizando outros tipo de classificadores ou sobre a outra técnica de aprendizado, o não supervisionado.

REFERÊNCIAS

- CAMPOS, T. E. **Técnicas de seleção de características com aplicações em reconhecimento de faces**. 159p. 2001. Dissertação (Mestrado em Ciências da Computação) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2001.
- BISHOP, C. M. **Pattern recognition and machine learning**. 2nd ed. New York: Springer Science+Business Media, 2006. 738p.
- CHANG, SHAO-TUNG; LU, HSIAO-CHUAN. The maximum likelihood estimates os multinomial parameters subject to stochastic order with equality, **Journal of National Taiwan University: Mathematics, Science and Technology**, Taiwan, v.44, p.1-9, 1999.
- DEMÉTRIO, C. G. B.; ZOCCHI, S. S. Modelos de regressão. In:_____. **Curso de modelos lineares**. Piracicaba: Escola Superior de Agronomia “Luís de Queiroz”, Universidade de São Paulo, 2010. 204p.
- DUARTE, M. C.; SANTOS, J. B.; MELO, L. C. Comparison of similarity coefficients based on rapd markers in common bean. **Genetics an Molecular Biology**, Ribeirão Preto, v.22, n.3, p.427-432, 1999
- DUDA, R. O.; HART, P. E.; STORK, D. G. **Pattern classification**. 2nd ed. New York: John Wiley & Sons, 2000. 654p.
- FARAWAY, J.J. **Linear models with R**. Boca Raton: Chapman & Hall/CRC, 2005. 242p.
- FERREIRA, M.E; GRATTAPAGLIA, D. **Introdução ao uso de marcadores moleculares em análise genética**. Brasília: EMBRAPA, 1996. 220p.
- FRIEDMAN, J.; HASTIE, T.; TIBSHIRANI, R. **The elements of statistical learning: data mining, inference and prediction**. 2nd ed. New York: Springer, 2008. 745p.
- GADI, M. F. A. **Uma comparação de métodos de classificação aplicados à detecção de fraude em cartões de crédito**. Dissertação (Mestrado em ciências da computação) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2008.
- HALL, M.; Frank, E.; HOLMES, G.; PFAHRINGER, B.; REUTEMANN, P.; Witten, I. H. The WEKA data mining software: an update. **SIGKDD Explorations**, New York, V. 11, p.10-18, 2009.
- JAIN, A. K.; DUIN, R. P. W.; MAO, J., Statistical pattern recognition: a review. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, New York, v.22, n.1, pp. 4-37, 2000.
- JIANG, L.; ZHANG, H.; CAI, Z.; SU, J. One dependence augmented naive Bayes. **Advanced Data Mining and Applications, Proceedings**, New York, v. 3584, p. 186-194, 2005.
- LANGLEY, P.; JOHN, G. H. Estimating continuous distributions in bayesian classifiers. In: CONFERENCE ON UNCERTAINTY IN ARTIFICIAL INTELLIGENCE 11., 1995, San Mateo. **Proceedings...** San Mateo: Morgan Kaufmann Publishers, 1995.
- LOBÃO, J. S. B.; FRANÇA-ROCHA, W. J. S.; SILVA, A. B. Aplicação dos índices kappa & pabak na validação da classificação automática de imagem de satélite em Feira de Santana-BA. In: SIMPÓSIO

BRASILEIRO DE SENSORIAMENTO REMOTO, 12., 2005, Goiânia . **Anais...** São José dos Campos: INPE, 2005. p. 1207-1214.

MARQUES, J. S. **Reconhecimento de padrões: métodos estatísticos neuronais**. 2 ed. Lisboa: IST Press, 2005. 284p.

MITCHELL, T. M. **Machine learning**. New York: McGraw-Hill Science/Engineering/Math, 1997. 423p.

MORAIS, E. C. **Reconhecimento de padrões e redes neurais artificiais em predição de estruturas secundárias de proteínas**. Tese (Doutorado Engenharia de Sistema e Computação) - Instituto Alberto Luiz Coimbra de Pós-graduação e Pesquisa de Engenharia - Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2010.

MEYER, A. S. **Comparação de coeficientes de similaridade usados em análises de agrupamentos com dados de marcadores moleculares dominantes**. 106p. Dissertação (Mestrado em Agronomia) - Escola Superior de Agronomia "Luís de Queiroz", Universidade de São Paulo, Piracicaba, 2002.

PÉREZ, A.; LARRANAGA, P.; INZA, I. Bayesian classifiers based on kernel density estimation: flexible classifiers. **International Journal of Approximate Reasoning**, New York, v. 50, n. 2, p. 341-362, 2009.

R. Development Core Team. R Foundation for Statistical Computing. **R: A language and environment for statistical computing**, Vienna, 2010. Disponível em <<http://www.r-project.org/>>. Acesso em: 25 mar. 2010

RENCER, A. C. **Multivariate statistical inference and applications**. New York: John Wiley & Sons, 1998. 566p.

RENNIE, J. D. M.; SHIH, L.; TEEVAN, J.; KARGER, D. R. Tackling the poor assumptions of naive Bayes text classifiers. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING 20., 2003, Washington DC. **Proceedings**, New York: New York: Springer Science+Business Media, 2003.

SUN, DA-WEN. **Infrared spectroscopy for food quality analysis and control**. New York: Elsevier, 2009. 414p.

TANAKA, E. Theoretical aspects of syntactic pattern recognition. **Pattern Recognition**, Ezmsford, v. 28, n. 7, p. 1053-1061, 1995.

UNIVERSITY of CALIFORNIA - IRVINE. Center for Machine Learning and Intelligent Systems. **Machine learning repository**: banco de dados para análise de reconhecimento de padrões da UCI. Disponível em <<http://archive.ics.uci.edu/ml/index.html>>. Acesso em: 16 set. 2010

WEBB, G. I.; BOUGHTON, J. R.; WANG, Z. Not so naive Bayes: aggregating one-dependence estimators. **Machine Learning**, Boston, v. 58, n. 1, p. 5-24, 2010.

WEKA. Development University of Waikato. The University of Waikato Hamilton, New Zeland. **WEKA**: Waikato environment for knowledge analysis, Hamilton, New Zeland. Disponível em <<http://www.cs.waikato.ac.nz/ml/weka/>>. Acesso em: 25 jun. 2010.

WITTEN, I. H.; FRANK, E. **Data mining pratical machine learning tools and techniques.**
2nd ed. San Francisco: Elsevier, 2005. 525p.

APÊNDICES

APÊNDICE A: Estimadores de máxima verossimilhança

Normal Multivariada

Dizemos que o vetor X de dimensão $d \times 1$ possui distribuição normal multivariada com vetor de médias μ , $d \times 1$ e matriz de covariância Σ , $d \times d$ e denotamos por $X \sim N_d(\mu, \Sigma)$ se sua função densidade de probabilidade é dada por:

$$f(x) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left[-\frac{1}{2} (x - \mu)^t \Sigma^{-1} (x - \mu) \right]$$

em que $|\Sigma|$ e Σ^{-1} representam o determinante e a inversa da matriz Σ respectivamente.

Estimação dos parâmetros

Para a execução dos algoritmos de classificação, a obtenção dos estimadores é fundamental, a seguir apresenta-se a obtenção dos estimadores de máxima verossimilhança.

Caso 1: $\Sigma = \sigma^2 \mathbf{I}$

Para achar os estimadores de máxima verossimilhança deve-se, primeiramente, construir a função de verossimilhança, $L(\mu, \Sigma)$.

$$\begin{aligned} L(\mu, \Sigma) &= \prod_{i=1}^n f(x_i) \\ &= \prod_{i=1}^n \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left[-\frac{1}{2} (x_i - \mu)^t \Sigma^{-1} (x_i - \mu) \right] \\ &= \frac{1}{(2\pi)^{nd/2} |\Sigma|^{n/2}} \exp \left[-\frac{1}{2} \sum_{i=1}^n (x_i - \mu)^t \Sigma^{-1} (x_i - \mu) \right] \end{aligned}$$

Algumas operações com matrizes são necessárias para a obtenção dos estimadores de máxima verossimilhança para o caso da distribuição normal multivariada (RENCHE, 1998).

$$(1) \quad A + B = B + A$$

$$(2) \quad (A \pm B)' = A' \pm B'$$

$$(3) \quad (x \pm y)' = x' \pm y'$$

$$(4) \quad \text{tr}(A + B) = \text{tr}(A) + \text{tr}(B);$$

$$(5) \quad \text{tr}(AB) = \text{tr}(BA);$$

(6) Se $A = (a'_1, \dots, a'_n)'$, então $\text{tr}(A'A) = \sum_{i=1}^n a'_i a_i$

(7) Se $A = (a_{(1)}, \dots, a_{(n)})$ então $\text{tr}(A'A) = \sum_{i=1}^n a'_{(i)} a_{(i)}$

$$\begin{aligned}
 L(\mu, \sigma^2) &= \prod_{i=1}^n f(x_i) \\
 &= \prod_{i=1}^n \frac{1}{(2\pi)^{d/2} |\sigma^2 \mathbf{I}|^{1/2}} \exp \left[-\frac{1}{2} (x_i - \mu)^t \frac{1}{\sigma^2} \mathbf{I} (x_i - \mu) \right] \\
 &= \prod_{i=1}^n \frac{1}{(2\pi)^{d/2} |\sigma^2 \mathbf{I}|^{1/2}} \exp \left[-\frac{1}{2} (x_i - \mu)^t \frac{1}{\sigma^2} \mathbf{I} (x_i - \mu) \right] \\
 &= \frac{1}{(2\pi)^{nd/2} |\sigma^2 \mathbf{I}|^{n/2}} \exp \left[-\frac{1}{2} \sum_{i=1}^n (x_i - \mu)^t \frac{1}{\sigma^2} \mathbf{I} (x_i - \mu) \right] \\
 &= \frac{1}{(2\pi)^{nd/2} \sigma^{2dn/2}} \exp \left[-\frac{1}{2} \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu)^t \mathbf{I} (x_i - \mu) \right] \\
 &= \frac{1}{(2\pi)^{nd/2} \sigma^{2dn/2}} \exp \left[-\frac{1}{2} \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu)^t (x_i - \mu) \right]
 \end{aligned}$$

Por outro lado:

$$\begin{aligned}
 \sum_{i=1}^n (x_i - \mu)^t (x_i - \mu) &= \sum_{i=1}^n (x_i - \bar{x} + \bar{x} - \mu)^t (x_i - \bar{x} + \bar{x} - \mu) \\
 &= \sum_{i=1}^n [(x_i - \bar{x})^t + (\bar{x} - \mu)^t] [(x_i - \bar{x}) + (\bar{x} - \mu)] \\
 &= \sum_{i=1}^n [(x_i - \bar{x})^t (x_i - \bar{x}) + (x_i - \bar{x})^t (\bar{x} - \mu) + (\bar{x} - \mu)^t (x_i - \bar{x}) + (\bar{x} - \mu)^t (\bar{x} - \mu)] \\
 &= \sum_{i=1}^n (x_i - \bar{x})^t (x_i - \bar{x}) + \sum_{i=1}^n (x_i - \bar{x})^t (\bar{x} - \mu) + \sum_{i=1}^n (\bar{x} - \mu)^t (x_i - \bar{x}) + \sum_{i=1}^n (\bar{x} - \mu)^t (\bar{x} - \mu) \\
 &= \sum_{i=1}^n (x_i - \bar{x})^t (x_i - \bar{x}) + \sum_{i=1}^n (\bar{x} - \mu)^t (\bar{x} - \mu) + (\bar{x} - \mu)^t \sum_{i=1}^n (x_i - \bar{x}) + (\bar{x} - \mu) \sum_{i=1}^n (x_i - \bar{x})^t
 \end{aligned}$$

Como $\sum_{i=1}^n (x_i - \bar{x}) = 0$, pode-se escrever a expressão acima como:

$$\sum_{i=1}^n (x_i - \bar{x})^t (x_i - \bar{x}) + n(\bar{x} - \mu)^t (\bar{x} - \mu)$$

Fazendo $H = \sum_{i=1}^n (x_i - \bar{x})^t (x_i - \bar{x})$ tem-se:

$$\sum_{i=1}^n (x_i - \mu)^t (x_i - \mu) = H + n(\bar{x} - \mu)^t (\bar{x} - \mu)$$

Então,

$$L(\mu, \sigma^2) = \frac{1}{(2\pi)^{nd/2} \sigma^{2dn/2}} \exp \left[-\frac{1}{2\sigma^2} (H + n(\bar{x} - \mu)^t (\bar{x} - \mu)) \right]$$

Dessa forma tem-se que o logaritmo da função de verossimilhança é dado por:

$$\begin{aligned} \ln L(\mu, \sigma^2) &= \ln \left[\frac{1}{(2\pi)^{nd/2} \sigma^{2dn/2}} \exp \left[-\frac{1}{2\sigma^2} (H + n(\bar{x} - \mu)^t (\bar{x} - \mu)) \right] \right] \\ &= -\frac{nd}{2} \ln(2\pi) - \frac{dn}{2} \ln \sigma^2 + \left[-\frac{1}{2\sigma^2} (H + n(\bar{x} - \mu)^t (\bar{x} - \mu)) \right] \\ &= -\frac{nd}{2} \ln(2\pi) - \frac{dn}{2} \ln \sigma^2 - \frac{1}{2\sigma^2} (H) - \frac{1}{2\sigma^2} n(\bar{x} - \mu)^t (\bar{x} - \mu) \end{aligned}$$

Sendo assim temos o seguinte sistema de equações:

$$\begin{cases} \frac{\partial \ln L(\mu, \sigma^2)}{\partial \mu} = 0 & \text{(I)} \Rightarrow \begin{cases} \frac{n}{\sigma^2} (\bar{x} - \mu) = 0 \Rightarrow \hat{\mu} = \bar{x} & \text{(I)} \end{cases} \\ \frac{\partial \ln L(\mu, \sigma^2)}{\partial \sigma^2} = 0 & \text{(II)} \Rightarrow \begin{cases} \frac{nd}{2} \ln 2\pi - \frac{nd}{2} \ln \sigma^2 - \frac{1}{2\sigma^2} (H) = 0 & \text{(II)} \end{cases} \end{cases}$$

Substituindo o resultado obtido em (I) em (II) tem-se que:

$$-\frac{nd}{2} \ln 2\pi - \frac{nd}{2} \ln \sigma^2 - \frac{1}{2\sigma^2} (H) = 0$$

Desenvolvendo (II)

$$\begin{aligned} \frac{\partial \ln L(\mu, \sigma^2)}{\partial \sigma^2} &= -0 - \frac{nd}{2\sigma^2} + \frac{1}{2\sigma^4} (H) = 0 \\ \frac{nd}{2\sigma^2} &= \frac{1}{2\sigma^4} (H) \\ \frac{2\sigma^4 nd}{2\sigma^2} &= (H) \\ \hat{\sigma}^2 &= \frac{H}{nd} \end{aligned}$$

Caso 2: Σ é uma matriz não estruturada

Como $(x_i - \mu)' \Sigma^{-1} (x_i - \mu)$ é um escalar, então tem-se:

$$\begin{aligned} (x_i - \mu)' \Sigma^{-1} (x_i - \mu) &= \text{tr}(x_i - \mu)' \Sigma^{-1} (x_i - \mu) \\ \sum_{i=1}^n (x_i - \mu)' \Sigma^{-1} (x_i - \mu) &= \sum_{i=1}^n \text{tr}(x_i - \mu)' \Sigma^{-1} (x_i - \mu) \\ &= \text{tr} \left[\Sigma^{-1} \sum_{i=1}^n (x_i - \mu)(x_i - \mu)' \right] \end{aligned}$$

Por outro lado:

$$\begin{aligned} \sum_{i=1}^n (x_i - \mu)(x_i - \mu)' &= \sum_{i=1}^n (x_i - \bar{x} + \bar{x} - \mu)(x_i - \bar{x} + \bar{x} - \mu)' \\ &= \sum_{i=1}^n [(x_i - \bar{x}) + (\bar{x} - \mu)] [(x_i - \bar{x})' + (\bar{x} - \mu)'] \\ &= \sum_{i=1}^n [(x_i - \bar{x})(x_i - \bar{x})' + (x_i - \bar{x})(\bar{x} - \mu)' + (\bar{x} - \mu)(x_i - \bar{x})' + (\bar{x} - \mu)(\bar{x} - \mu)'] \\ &= \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})' + \sum_{i=1}^n (x_i - \bar{x})(\bar{x} - \mu)' + \sum_{i=1}^n (\bar{x} - \mu)(x_i - \bar{x})' + \sum_{i=1}^n (\bar{x} - \mu)(\bar{x} - \mu)' \\ &= \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})' + \sum_{i=1}^n (\bar{x} - \mu)(\bar{x} - \mu)' + (\bar{x} - \mu)' \sum_{i=1}^n (x_i - \bar{x}) + (\bar{x} - \mu) \sum_{i=1}^n (x_i - \bar{x})' \end{aligned}$$

Como $\sum_{i=1}^n (x_i - \bar{x}) = 0$, pode-se escrever a expressão acima como:

$$\sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})' + n(\bar{x} - \mu)(\bar{x} - \mu)'$$

Fazendo $W = \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})'$ tem-se:

$$\sum_{i=1}^n (x_i - \mu)(x_i - \mu)' = W + n(\bar{x} - \mu)(\bar{x} - \mu)'$$

$$L(\mu, \Sigma) = \frac{1}{(2\pi)^{nd/2} |\Sigma|^{n/2}} e^{-\frac{1}{2} \text{tr}[\Sigma^{-1}(W + n(\bar{x} - \mu)(\bar{x} - \mu)')]} \quad \square$$

Desta forma o logaritmo da função de verossimilhança é dado por:

$$\begin{aligned}
 \ln L(\mu, \Sigma) &= -\frac{nd}{2} \ln 2\pi - \frac{n}{2} \ln |\Sigma| - \frac{1}{2} [\text{tr} \Sigma^{-1} [\mathbf{W} + n(\bar{x} - \mu)(\bar{x} - \mu)']] \\
 &= -\frac{nd}{2} \ln 2\pi - \frac{n}{2} \ln |\Sigma| - \frac{1}{2} \text{tr} (\Sigma^{-1} \mathbf{W}) - \frac{n}{2} \Sigma^{-1} (\bar{x} - \mu)' (\bar{x} - \mu) \\
 &= -\frac{nd}{2} \ln 2\pi - \frac{n}{2} \ln |\Sigma| - \frac{1}{2} \text{tr} (\Sigma^{-1} \mathbf{W}) - \frac{n}{2} (\bar{x} - \mu)' \Sigma^{-1} (\bar{x} - \mu)
 \end{aligned}$$

Para encontrar o estimador de máxima verossimilhança para μ e Σ derivamos $\ln L(\mu, \Sigma)$ da equação anterior e iguala-se a 0. Para isto será utilizado as seguintes propriedades de derivação matricial:

- 1) Sejam $a = (a_1, \dots, a_p)'$, $x = (x_1, \dots, x_p)$ e $u = a'x = x'a$. Então $\frac{\partial(a'x)}{\partial x} = \frac{\partial(x'a)}{\partial x} = a$
- 2) Seja $u = x'Ax$, em que A é uma matriz simétrica $p \times p$. Então $\frac{\partial(x'Ax)}{\partial x} = 2Ax$
- 3) Seja $\text{tr}(AB)$ em que A é uma matriz simétrica e B é uma matriz. Então $\frac{\partial \text{tr}(AB)}{\partial A} = B + B' - \text{diag}(B)$
- 4) Seja $\ln |A|$, em que A é simétrica e não singular $\frac{\partial \ln |A|}{\partial A} = 2A^{-1} - \text{diag}(A^{-1})$

Sendo assim tem-se o seguinte sistema de equações:

$$\begin{cases} \frac{\partial \ln L(\mu, \Sigma)}{\partial \mu} = 0 & \text{(I)} \Rightarrow \begin{cases} \frac{n}{\Sigma^{-1}} (\bar{x} - \mu) = 0 \Rightarrow \hat{\mu} = \bar{x} & \text{(I)} \end{cases} \\ \frac{\partial \ln L(\mu, \Sigma)}{\partial \Sigma^{-1}} = 0 & \text{(II)} \Rightarrow \begin{cases} \frac{n}{2} \ln |\Sigma^{-1}| - \frac{1}{2} \text{tr}(\Sigma^{-1} \mathbf{W}) = 0 & \text{(II)} \end{cases} \end{cases}$$

Desenvolvendo (II)

$$\frac{\partial \ln L(\mu, \Sigma)}{\partial \Sigma^{-1}} = -O + n\Sigma - \frac{n}{2}\text{diag}(\Sigma) - W + \frac{1}{2}\text{diag}(W) = O$$

$$n\Sigma - \frac{n}{2}\text{diag}(\Sigma) = W + \frac{1}{2}\text{diag}(W)$$

$$n\left(\Sigma - \frac{1}{2}\text{diag}(\Sigma)\right) = W - \frac{1}{2}\text{diag}(W)$$

$$\left(\Sigma - \frac{1}{2}\text{diag}(\Sigma)\right) = \frac{1}{n}\left(W - \frac{1}{2}\text{diag}(W)\right)$$

$$\left(\Sigma - \frac{1}{2}\text{diag}(\Sigma)\right) = \left(\frac{W}{n} - \frac{1}{2n}\text{diag}(W)\right)$$

$$\left(\Sigma - \frac{1}{2}\text{diag}(\Sigma)\right) = \left(\frac{W}{n} - \frac{1}{2}\text{diag}\left(\frac{W}{n}\right)\right)$$

Pode-se observar que o estimador para $\hat{\Sigma} = \frac{W}{n}$. Este estimador é viciado, o estimador não viciado é $\hat{S} = \frac{W}{n-1}$, utilizado para estimação amostral.

Um exemplo numérico

Foram feitas simulações de dados provenientes da distribuição normal para os dois casos, em ambos os casos foram simulados duas amostras, uma de 50 elementos e outra de 400 elementos. Ambas amostras tinham o mesmo vetor de médias, $\mu = (4, 5, 2)^t$, para o primeiro caso $\sigma^2 I$, foi utilizado $\sigma^2 = 1,5$ desta maneira a matriz de covariância é:

$$\Sigma = \begin{pmatrix} 1,5 & 0 & 0 \\ 0 & 1,5 & 0 \\ 0 & 0 & 1,5 \end{pmatrix}$$

Para o segundo caso a matriz de covariância utilizada foi:

$$\Sigma = \begin{pmatrix} 2 & 0,90 & 0,50 \\ 0,90 & 1 & 0,45 \\ 0,50 & 0,45 & 3 \end{pmatrix}$$

Após a simulação e utilizando os estimadores encontrados, os resultados obtidos foram:

- 1) Para o primeiro caso com $n=50$ o vetor de médias encontrado foi $(4,000, 5,012, 2,089)^t$ com $\sigma^2 = 1,4616$ e a matriz de covariância

$$\hat{S} = \begin{pmatrix} 1,4616 & 0 & 0 \\ 0 & 1,4616 & 0 \\ 0 & 0 & 1,4616 \end{pmatrix}$$

- 2) Para o primeiro caso com $n=400$ o vetor de médias encontrado foi $(4,045, 4,993, 1,988)^t$ com $\sigma^2 = 1,5156$ e a matriz de covariância

$$\hat{S} = \begin{pmatrix} 1,5156 & 0 & 0 \\ 0 & 1,5156 & 0 \\ 0 & 0 & 1,5156 \end{pmatrix}$$

- 3) Para o segundo caso com $n=50$ o vetor de médias encontrado foi $(4,000, 5,012, 2,089)^t$ e a matriz de covariância

$$\hat{S} = \begin{pmatrix} 2,1262 & 1,1319 & 0,9639 \\ 1,1319 & 1,1761 & 0,6111 \\ 0,9639 & 0,6111 & 2,9952 \end{pmatrix}$$

- 4) Para o segundo caso com $n=400$ o vetor de médias encontrado foi $(4,045, 4,993, 1,988)^t$ e a matriz de covariância

$$\hat{S} = \begin{pmatrix} 2,0515 & 0,9426 & 0,4247 \\ 0,9426 & 1,0039 & 0,4164 \\ 0,4247 & 0,4164 & 2,6323 \end{pmatrix}$$

Distribuição Multinomial

Dizemos que X possui distribuição multinomial se sua função de probabilidade é dada por:

$$p(x) = \frac{N!}{x_1!x_2!\dots x_n!} p_1^{x_1} p_2^{x_2} \dots p_n^{x_n}$$

sendo que $\sum_{i=1}^n p_i = 1$ e $\sum_{i=1}^n x_i = N$

Estimação dos parâmetros

Para encontrar os estimadores de máxima verossimilhança deve-se, obter o logaritmo de função de verossimilhança, $l(\mathbf{p}(\mathbf{x}))$.

$$l(\mathbf{p}(\mathbf{x})) = \log n! - \log \sum_{i=1}^m x_i! + \sum_{i=1}^m x_i \log p_i$$

Para encontrar os estimadores de máxima verossimilhança é necessário fazer as derivadas em relação aos p'_i s, porém este tipo de derivação possui restrição e para resolver um novo parâmetro, o multiplicador de lagrange (CHANG; LU, 1999). Desta maneira a equação fica:

$$f(p, \lambda) = l(p_1, \dots, p_m) + \lambda \left(1 - \sum_{i=1}^m p_i \right)$$

Com base nesta nova equação deve-se fazer as derivadas em relação aos parâmetros p'_i s e em relação à λ .

$$\begin{cases} \frac{\partial f(p_i, \lambda)}{\partial p_i} = 0 & \text{em que } i = 1, \dots, m \\ \frac{\partial f(p_i, \lambda)}{\partial \lambda} = 0 \end{cases}$$

Assim as equação de derivadas ficam:

$$\begin{aligned}\frac{\partial f(p, \lambda)}{\partial p_1} &= \frac{x_1}{p_1} - \lambda \\ \frac{\partial f(p, \lambda)}{\partial p_2} &= \frac{x_2}{p_2} - \lambda \\ \frac{\partial f(p, \lambda)}{\partial p_3} &= \frac{x_3}{p_3} - \lambda \\ \frac{\partial f(p, \lambda)}{\partial p_4} &= \frac{x_4}{p_4} - \lambda \\ &\vdots \\ \frac{\partial f(p, \lambda)}{\partial \lambda} &= 1 - \sum_{i=1}^m p_i\end{aligned}$$

Assim

$$\begin{aligned}\frac{x_1}{p_1} - \lambda &= 0 \\ p_1 &= \frac{x_1}{\lambda}\end{aligned}$$

e

$$\sum_{i=1}^m p_i = 1$$

Deste modo pode-se dizer que

$$\hat{p}_i = \frac{x_i}{\lambda}$$

então

$$\begin{aligned}\sum_{i=1}^m \hat{p}_i &= 1 \\ \frac{x_1}{\lambda} + \frac{x_2}{\lambda} + \frac{x_3}{\lambda} + \dots + \frac{x_m}{\lambda} &= 1 \\ \frac{\sum_{i=1}^m x_i}{\lambda} &= 1 \\ \hat{\lambda} &= \sum_{i=1}^m x_i\end{aligned}$$

sendo assim

$$\begin{aligned}\hat{p}_i &= \frac{x_i}{\lambda} \\ \hat{p}_i &= \frac{x_i}{\sum_{i=1}^m x_i} \\ \hat{p}_i &= \frac{x_i}{N}\end{aligned}$$

Um exemplo numérico

Foram simuladas quatro amostras de tamanhos diferentes porém com a mesma probabilidade. Os tamanhos das amostras foram $n_1 = 50$, $n_2 = 100$, $n_3 = 150$ e $n_4 = 200$. E as probabilidades: $p_1 = 0,25$, $p_2 = 0,4$, $p_3 = 0,15$ e $p_4 = 0,2$.

Utilizando os estimadores calculados foram obtidas as seguintes probabilidades:

- 1) Os valores obtidos para as probabilidades da amostra de tamanho 50 foram: $\hat{p}_1 = 0,2800$, $\hat{p}_2 = 0,4200$, $\hat{p}_3 = 0,1200$ e $\hat{p}_4 = 0,1800$
- 2) Os valores obtidos para as probabilidades da amostra de tamanho 100 foram: $\hat{p}_1 = 0,1900$, $\hat{p}_2 = 0,4100$, $\hat{p}_3 = 0,2100$ e $\hat{p}_4 = 0,1900$
- 3) Os valores obtidos para as probabilidades da amostra de tamanho 150 foram: $\hat{p}_1 = 0,2666$, $\hat{p}_2 = 0,4333$, $\hat{p}_3 = 0,1333$ e $\hat{p}_4 = 0,1666$
- 4) Os valores obtidos para as probabilidades da amostra de tamanho 200 foram: $\hat{p}_1 = 0,2300$, $\hat{p}_2 = 0,4100$, $\hat{p}_3 = 0,1300$ e $\hat{p}_4 = 0,2300$

APÊNDICE B Programação desenvolvida no *R*

CONJUNTO DE DADOS ÍRIS

```

library(e1071); data(iris); set.seed(133)
ob<-sample(length(iris[,1]),.66*length(iris[,1]))
test<-iris[ob,]; val<-iris[-ob,]; m<-matrix(0,4,3)
classe.true<-matrix(0,length(iris[,5]),1)
for(i in 1:length(iris[,5])){
  if(iris[i,5]=="setosa")           classe.true[i]=1
  if(iris[i,5]=="versicolor")      classe.true[i]=2
  if(iris[i,5]=="virginica")        classe.true[i]=3
}
for(i in 1:4){; m[i,<-tapply(test[,i],test[,5],mean)
}
S<-cov(test[,1:4]); SS1<-sum(diag(S))
S1<-cov(test[1:33,1:4]); S2<-cov(test[34:68,1:4]); S3<-cov(test[69:99,1:4])
p1w<-0;p2w<-0;p3w<-0;G<-0;t<-length(val[,1]); x1<-matrix(0,length(val[,1]),4)
##### Sigma I #####
for(i in 1:t){
  x1[,<-c(val[,1],val[,2],val[,3],val[,4])
  p1w[i]<- -(t(x1[i,]-m[,1]))%*(x1[i,]-(m[,1])))/(2*SS1)
  p2w[i]<- -(t(x1[i,]-m[,2]))%*(x1[i,]-(m[,2])))/(2*SS1)
  p3w[i]<- -(t(x1[i,]-m[,3]))%*(x1[i,]-(m[,3])))/(2*SS1)
  G<-t(rbind( p1w,p2w,p3w))
}
classifica<-matrix(0,t,2); x<-1:3
for(i in 1:t){
  ii<-order(G[i,],x); classifica[i,]<-rbind(G[i,],x)[,ii][,3]
}
compara<-cbind(classifica[,2],classe.true[-ob]); erro<-compara[,1]-compara[,2]

```

```

mat_conf_1<-matrix(0,3,3); mat_conf11 = matrix(0,3,3)
tc<-c(17,15,19); tc=cumsum(tc); ntc<-length(tc); resh1=table(compara[1:tc[1]])
mat_conf_1[as.numeric(names(resh1))]=table(compara[1:tc[1]])
mat_conf_1; mat_conf_1[,1]; mat_conf11[1,]=mat_conf_1[,1]; mat_conf11
for(i in 1:(ntc-1)){
mat_conf_1<-matrix(0,3,3)
resh1<-table(compara[(tc[i]+1):tc[i+1]])
mat_conf_1[as.numeric(names(resh1))]=table(compara[(tc[i]+1):tc[i+1]])
mat_conf_1; mat_conf11[i+1,]=mat_conf_1[,1]; mat_conf11
}
mat_conf11;ERR1<-sum(diag(mat_conf11))/sum(mat_conf11);ERR0_Total1<-1-(ERR1)
##### Sigma Comum #####
p11w<-0; p12w<-0; p13w<-0; G1<-0; x1<-matrix(0,t,4)
for(i in 1:t){
x1[,]<-c(val[,1],val[,2],val[,3],val[,4])
p11w[i]<- -(1/2)*t(x1[i,]-m[,1])%*%solve(S)%*%(x1[i,]-m[,1])
p12w[i]<- -(1/2)*t(x1[i,]-m[,2])%*%solve(S)%*%(x1[i,]-m[,2])
p13w[i]<- -(1/2)*t(x1[i,]-m[,3])%*%solve(S)%*%(x1[i,]-m[,3])
G1<-t(rbind(p11w,p12w,p13w))
}
classifica2<-matrix(0,t,2); x<-1:3
for(i in 1:t){
ii2<-order(G1[i,],x); classifica2[i,]<-rbind(G1[i,],x)[,ii2][,3]
}
compara2<-cbind(classifica2[,2],classe.true[-ob]); erro2<-compara2[,1]-compara2[,2]
mat_conf_2<-matrix(0,3,3); mat_conf22 = matrix(0,3,3); resh2=table(compara2[1:tc[1]])
mat_conf_2[as.numeric(names(resh2))]=table(compara2[1:tc[1]])
mat_conf_2; mat_conf_2[,1]; mat_conf22[1,]=mat_conf_2[,1]; mat_conf22
for(i in 1:(ntc-1)){
mat_conf_2<-matrix(0,3,3)

```

```

resh2<-table(compara2[(tc[i]+1):tc[i+1]])
mat_conf_2[as.numeric(names(resh2))]=table(compara2[(tc[i]+1):tc[i+1]])
mat_conf_2; mat_conf22[i+1,]=mat_conf_2[,1]; mat_conf22
};ERR2<-sum(diag(mat_conf22)/sum(mat_conf22) );ERR0_Total2<-1-(ERR2)
##### Sigma arbitrária #####
p111w<-0; p112w<-0; p113w<-0; G3<-0; x1<-matrix(0,t,4)
for(i in 1:t){
x1[,]<-c(val[,1],val[,2],val[,3],val[,4])
p111w[i]<- -((1/2)*t(x1[i,])%%solve(S1)%%(x1[i,]))+(t(solve(S1)%%
m[,1])%%x1[i,])-((1/2)*t(m[,1])%%solve(S1)%%m[,1]-(1/2)*log(det(S1)))
p112w[i]<- -((1/2)*t(x1[i,])%%solve(S2)%%(x1[i,]))+(t(solve(S2)%%
m[,2])%%x1[i,])-((1/2)*t(m[,2])%%solve(S2)%%m[,2]-(1/2)*log(det(S2)))
p113w[i]<- -((1/2)*t(x1[i,])%%solve(S3)%%(x1[i,]))+(t(solve(S3)%%
m[,3])%%x1[i,])-((1/2)*t(m[,3])%%solve(S3)%%m[,3]-(1/2)*log(det(S3)))
G3<-t(rbind(p111w,p112w,p113w))
}; classifica3<-matrix(0,t,2); x<-1:3; for(i in 1:t){
ii3<-order(G3[i,],x); classifica3[i,]<-rbind(G3[i,],x)[,ii3][,3]
}
compara3<-cbind(classifica3[,2],classe.true[-ob]); mat_conf_3<-matrix(0,3,3)
mat_conf33 = matrix(0,3,3); resh3=table(compara[1:tc[1]])
mat_conf_3[as.numeric(names(resh3))]=table(compara[1:tc[1]])
for(i in 1:(ntc-1)){
mat_conf_3<-matrix(0,3,3); resh3<-table(compara[(tc[i]+1):tc[i+1]])
mat_conf_3[as.numeric(names(resh3))]=table(compara3[(tc[i]+1):tc[i+1]])
mat_conf_3; mat_conf33[i+1,]=mat_conf_3[,1]; mat_conf33
}
ERR3<-sum(diag(mat_conf33))/sum(mat_conf33); ERR0_Total3<-1-(ERR3);

```


CONJUNTO DE DADOS DE MARCADORES MOLECULARES

```

rm(list=ls(all=TRUE))
library(RODBC)

dl<- odbcConnectExcel("RAPD.xls"); mat<-sqlFetch(dl,"Plan3"); attach(mat)
N=262; p<-NULL; q<-NULL; gx<-NULL; px<-NULL; pxw<-NULL; qxw<-NULL; err2<-NULL
err1<-NULL; razao<-NULL; err1<-NULL; p1xw<-NULL; q1xw<-NULL; p1x<-NULL; m<-NULL
for( i in 1:N){
p[i]<- (mat[i,1]+mat[i,2]+mat[i,5]+mat[i,6]+mat[i,8]+mat[i,9])/5,00001
q[i]<- (mat[i,9]+mat[i,18]+mat[i,14]+mat[i,19]+mat[i,10]+mat[i,12])/5,00001
for(j in 1:19){
gx[j]<-sum(mat[,j]*log(p/q)+(1-mat[,j])*(log((1-p)/(1-q))))
p1xw[j]<-log(prod((p[i]^mat[,j])*(1-p[i])^(1-mat[,j]))))
q1xw[j]<-log(prod((q^mat[,j])*(1-q)^(1-mat[,j]))))
pxw[j]<-prod((p^mat[1:N,j])*(1-p)^(1-mat[1:N,j]))
qxw[j]<-prod((q^mat[1:N,j])*(1-q)^(1-mat[1:N,j]))
px[j]<-((pxw[j])+(qxw[j])); err1[j]<- 1-(qxw[j]/px[j])
err2[j]<- 1-(pxw[j]/px[j]); m[i]<-mat[i,1]
}
}; p1xw[j]<-log(prod((p^mat[,j])*(1-p)^(1-mat[,j]))); p1xw
p1xw[j]<-(prod((p^mat[,j])*(1-p)^(1-mat[,j]))); p1xw
q1xw[j]<-(prod((q^mat[,j])*(1-q)^(1-mat[,j]))); q1xw

```

CONJUNTO DE DADOS REFERENTES AS EMBALAGENS DE PEÇAS DE AUTOMÓVEIS SEM SELEÇÃO DE VARIÁVEL

```
rm(list=ls(all=TRUE))

library(RODBC)

dl<-odbcConnectExcel("dados Jú1.xls"); mat<-sqlFetch(dl,"Sheet1"); attach(mat)

##### Classe verdadeira #####

classe.true<-matrix(0,88,1)

for(i in 1:88){
  if(mat[i,17]=="filtro combustivel") classe.true[i]=1
  if(mat[i,17]=="vela")               classe.true[i]=2
  if(mat[i,17]=="palheta")            classe.true[i]=3
  if(mat[i,17]=="filtro ar")          classe.true[i]=4
  if(mat[i,17]=="filtro oleo")        classe.true[i]=5
  if(mat[i,17]=="ponta homocinetica") classe.true[i]=6
  if(mat[i,17]=="bomba agua")         classe.true[i]=7
  if(mat[i,17]=="cabo vela")          classe.true[i]=8
  if(mat[i,17]=="correia dentada")    classe.true[i]=9
}

##### Treino #####

fc1<-mat[1:8,3:14]; V1<-mat[12:20,3:14]; P1<-mat[25:35,3:14];
fa1<-mat[39:45,3:14]; fo1<-mat[49:51,3:14]; ph1<-mat[53:60,3:14]
ba1<-mat[64:67,3:14]; cv1<-mat[70:78,3:14]; cd1<-mat[83:87,3:14]
fc11<-mat[1:8,10:12]; V11<-mat[12:20,10:12]; P11<-mat[25:35,10:12]
fa11<-mat[39:45,10:12]; fo11<-mat[49:51,10:12]; ph11<-mat[53:60,10:12]
ba11<-mat[64:67,10:12]; cv11<-mat[70:78,10:12]
cd11<-mat[83:87,10:12]; MM<-rbind(fc1,V1,P1,fa1,fo1,ph1,ba1,cv1,cd1)

##### Transformação da Matriz de treino #####

matcor1=matrix(0,64,3); matcor2=matrix(0,64,2); matcor3=matrix(0,64,8)
matcor4=matrix(0,64,7); matcor5=matrix(0,64,7)

for(i in 1:88){
```

```

if(MM[i,1]=="retangular") matcor1[i,]=c(1,0,0)
if(MM[i,1]=="irregular") matcor1[i,]=c(0,1,0)
if(MM[i,1]=="quadrada") matcor1[i,]=c(0,0,1)
}; for(i in 1:88){
if(MM[i,2]=="papelao") matcor2[i,]=c(1,0)
if(MM[i,2]=="plastico") matcor2[i,]=c(0,1)
}; for(i in 1:88){
if(MM[i,6]=="vermelho") matcor3[i,]=c(1,0,0,0,0,0,0,0)
if(MM[i,6]=="preto") matcor3[i,]=c(0,1,0,0,0,0,0,0)
if(MM[i,6]=="amarelo") matcor3[i,]=c(0,0,1,0,0,0,0,0)
if(MM[i,6]=="azul") matcor3[i,]=c(0,0,0,1,0,0,0,0)
if(MM[i,6]=="branco") matcor3[i,]=c(0,0,0,0,1,0,0,0)
if(MM[i,6]=="verde") matcor3[i,]=c(0,0,0,0,0,1,0,0)
if(MM[i,6]=="cinza") matcor3[i,]=c(0,0,0,0,0,0,1,0)
if(MM[i,6]=="laranja") matcor3[i,]=c(0,0,0,0,0,0,0,1)
}; for(i in 1:88){
if(MM[i,7]==2) matcor4[i,]=c(1,0,0,0,0,0,0,0)
if(MM[i,7]==3) matcor4[i,]=c(0,1,0,0,0,0,0,0)
if(MM[i,7]==4) matcor4[i,]=c(0,0,1,0,0,0,0,0)
if(MM[i,7]==5) matcor4[i,]=c(0,0,0,1,0,0,0,0)
if(MM[i,7]==6) matcor4[i,]=c(0,0,0,0,1,0,0,0)
if(MM[i,7]==8) matcor4[i,]=c(0,0,0,0,0,1,0,0)
if(MM[i,7]==9) matcor4[i,]=c(0,0,0,0,0,0,1,0)
}; TC<-matcor1; TM<-matcor2; CP<-matcor3; QC<-matcor4; MM1<-cbind(TC,TM,CP,QC)
##### Estimaco #####
P1<-MM1[1:8,]; P2<-MM1[9:16,]; P3<-MM1[17:27,]; P4<-MM1[28:34,]; P5<-MM1[35:37,]
P6<-MM1[38:46,]; P7<-MM1[47:51,]; P8<-MM1[52:60,]; P9<-MM1[60:64,]
### Estimaco das variaveis #####
P1p<-0; P2p<-0; P3p<-0; P4p<-0; P5p<-0; P6p<-0; P7p<-0; P8p<-0; P9p<-0
for(j in 1:20){

```

```

P1p[j]<-sum(P1[,j])/8
P2p[j]<-sum(P2[,j])/8
P3p[j]<-sum(P3[,j])/11
P4p[j]<-sum(P4[,j])/7
P5p[j]<-sum(P5[,j])/3
P6p[j]<-sum(P6[,j])/9
P7p[j]<-sum(P7[,j])/5
P8p[j]<-sum(P8[,j])/9
P9p[j]<-sum(P9[,j])/5
}

#### Programa #####

##### Parte multinomial #####

##### Transformação da Matriz original #####

matcor11=matrix(0,88,3); matcor21=matrix(0,88,2); matcor31=matrix(0,88,8)
matcor41=matrix(0,88,7); matcor51=matrix(0,88,7)
for(i in 1:88){
  if(mat[i,3]=="retangular") matcor11[i,]=c(1,0,0)
  if(mat[i,3]=="irregular") matcor11[i,]=c(0,1,0)
  if(mat[i,3]=="quadrada") matcor11[i,]=c(0,0,1)
}; for(i in 1:88){
  if(mat[i,4]=="papelao") matcor21[i,]=c(1,0)
  if(mat[i,4]=="plastico") matcor21[i,]=c(0,1)
}; for(i in 1:88){
  if(mat[i,8]=="vermelho") matcor31[i,]=c(1,0,0,0,0,0,0,0)
  if(mat[i,8]=="preto") matcor31[i,]=c(0,1,0,0,0,0,0,0)
  if(mat[i,8]=="amarelo") matcor31[i,]=c(0,0,1,0,0,0,0,0)
  if(mat[i,8]=="azul") matcor31[i,]=c(0,0,0,1,0,0,0,0)
  if(mat[i,8]=="branco") matcor31[i,]=c(0,0,0,0,1,0,0,0)
  if(mat[i,8]=="verde") matcor31[i,]=c(0,0,0,0,0,1,0,0)
  if(mat[i,8]=="cinza") matcor31[i,]=c(0,0,0,0,0,0,1,0)
}

```

```

if(mat[i,8]=="laranja") matcor31[i,]=c(0,0,0,0,0,0,0,1)
}; for(i in 1:88){
if(mat[i,9]==2) matcor41[i,]=c(1,0,0,0,0,0,0,0)
if(mat[i,9]==3) matcor41[i,]=c(0,1,0,0,0,0,0,0)
if(mat[i,9]==4) matcor41[i,]=c(0,0,1,0,0,0,0,0)
if(mat[i,9]==5) matcor41[i,]=c(0,0,0,1,0,0,0,0)
if(mat[i,9]==6) matcor41[i,]=c(0,0,0,0,1,0,0,0)
if(mat[i,9]==8) matcor41[i,]=c(0,0,0,0,0,1,0,0)
if(mat[i,9]==9) matcor41[i,]=c(0,0,0,0,0,0,1,0)
}; TC1<-matcor11; TM1<-matcor21; CP1<-matcor31; QC1<-matcor41
Mat1<-cbind(TC1, TM1, CP1, QC1)
#### Parte Normal Multivariada #####
n<-c(3); I<- matrix(0,nrow=n,ncol=n); I[row(I)==col(I)] <- 1
M1<-apply(fc11,2,mean); S1<-cov(fc11); M2<-apply(V11,2,mean); S2<-cov(V11)
M3<-apply(P11,2,mean); S3<-cov(P11); M4<-apply(fa11,2,mean); S4<-cov(fa11)
M5<-apply(fo11,2,mean); S5<-cov(fo11); M6<-apply(ph11,2,mean); S6<-cov(ph11)
M7<-apply(ba11,2,mean); S7<-cov(ba11); M8<-apply(cv11,2,mean); S8<-cov(cv11)
M9<-apply(cd11,2,mean); S9<-cov(cd11); S<-cov(MM[8:10]); SS1<-sum(diag(S))
#### Classificação com os dois tipos de distribuição Sigma*I####
p1w<-0; p2w<-0; p3w<-0; p4w<-0; p5w<-0; p6w<-0; p7w<-0; p8w<-0; p9w<-0; G<-0
x1<-matrix(c(rep(0)),88,3)
for(i in 1:88){
x1[,]<-c(mat[,10],mat[,11],mat[,12])
p1w[i]<-log(prod(P1p^Mat1[i,]))-(t(x1[i,]-M1)%*(x1[i,]-M1))/(2*SS1)
p2w[i]<-log(prod(P2p^Mat1[i,]))-(t(x1[i,]-M2)%*(x1[i,]-M2))/(2*SS1)
p3w[i]<-log(prod(P3p^Mat1[i,]))-(t(x1[i,]-M3)%*(x1[i,]-M3))/(2*SS1)
p4w[i]<-log(prod(P4p^Mat1[i,]))-(t(x1[i,]-M4)%*(x1[i,]-M4))/(2*SS1)
p5w[i]<-log(prod(P5p^Mat1[i,]))-(t(x1[i,]-M5)%*(x1[i,]-M5))/(2*SS1)
p6w[i]<-log(prod(P6p^Mat1[i,]))-(t(x1[i,]-M6)%*(x1[i,]-M6))/(2*SS1)
p7w[i]<-log(prod(P7p^Mat1[i,]))-(t(x1[i,]-M7)%*(x1[i,]-M7))/(2*SS1)

```

```

p8w[i]<-log(prod(P8p^Mat1[i,]))-(t(x1[i,]-M8)%*(x1[i,]-(M8)))/(2*SS1)
p9w[i]<-log(prod(P9p^Mat1[i,]))-(t(x1[i,]-M9)%*(x1[i,]-(M9)))/(2*SS1)
G<-t(rbind( p1w,p2w,p3w,p4w,p5w,p6w,p7w,p8w,p9w))
}

#### Classificação com os dois tipos de distribuição Sigma comum ####
p11w<-0; p12w<-0; p13w<-0; p14w<-0
p15w<-0; p16w<-0; p17w<-0; p18w<-0; p19w<-0; G1<-0
x1<-matrix(c(rep(0)),88,3)
for(i in 1:88){
  x1[,]<-c(mat[,10],mat[,11],mat[,12])
  p11w[i]<-log(prod(P1p^Mat1[i,]))-(1/2)*t(x1[i,]-M1)%*%solve(S)%*(x1[i,]-M1)
  p12w[i]<-log(prod(P2p^Mat1[i,]))-(1/2)*t(x1[i,]-M2)%*%solve(S)%*(x1[i,]-M2)
  p13w[i]<-log(prod(P3p^Mat1[i,]))-(1/2)*t(x1[i,]-M3)%*%solve(S)%*(x1[i,]-M3)
  p14w[i]<-log(prod(P4p^Mat1[i,]))-(1/2)*t(x1[i,]-M4)%*%solve(S)%*(x1[i,]-M4)
  p15w[i]<-log(prod(P5p^Mat1[i,]))-(1/2)*t(x1[i,]-M5)%*%solve(S)%*(x1[i,]-M5)
  p16w[i]<-log(prod(P6p^Mat1[i,]))-(1/2)*t(x1[i,]-M6)%*%solve(S)%*(x1[i,]-M6)
  p17w[i]<-log(prod(P7p^Mat1[i,]))-(1/2)*t(x1[i,]-M7)%*%solve(S)%*(x1[i,]-M7)
  p18w[i]<-log(prod(P8p^Mat1[i,]))-(1/2)*t(x1[i,]-M8)%*%solve(S)%*(x1[i,]-M8)
  p19w[i]<-log(prod(P9p^Mat1[i,]))-(1/2)*t(x1[i,]-M9)%*%solve(S)%*(x1[i,]-M9)
  G1<-t(rbind( p11w,p12w,p13w,p14w,p15w,p16w,p17w,p18w,p19w))
}; classifica<-matrix(0,88,2); x<-1:9
for(i in 1:88){
  ii<-order(G[i,],x); classifica[i,]<-rbind(G[i,],x)[,ii][,9]
}; compara<-cbind(classifica[,2],classe.true); erro<-compara[,1]-compara[,2]
classifica1<-matrix(0,88,2); x<-1:9
for(i in 1:88){
  ii1<-order(G1[i,],x); classifica1[i,]<-rbind(G1[i,],x)[,ii1][,9]
}; compara1<-cbind(classifica1[,2],classe.true);
erro1<-compara1[,1]-compara1[,2]

```

CONJUNTO DE DADOS REFERENTES AS EMBALAGENS DE PEÇAS DE AUTOMÓVEIS COM SELEÇÃO DE VARIÁVEL

```
rm(list=ls(all=TRUE))
library(RODBC)
library(rgl)
library(car)
dl<-odbcConnectExcel("dados Jú1.xls"); mat<-sqlFetch(dl,"Sheet1")
attach(mat)
##### Classe verdadeira #####
classe.true<-matrix(0,88,1)
for(i in 1:88){
  if(mat[i,17]=="filtro combustivel") classe.true[i]=1
  if(mat[i,17]=="vela") classe.true[i]=2
  if(mat[i,17]=="palheta") classe.true[i]=3
  if(mat[i,17]=="filtro ar") classe.true[i]=4
  if(mat[i,17]=="filtro oleo") classe.true[i]=5
  if(mat[i,17]=="ponta homocinetica") classe.true[i]=6
  if(mat[i,17]=="bomba agua") classe.true[i]=7
  if(mat[i,17]=="cabo vela") classe.true[i]=8
  if(mat[i,17]=="correia dentada") classe.true[i]=9
}
##### Treino #####
fc1<-mat[1:8,3:14]; V1<-mat[12:20,3:14]; P1<-mat[25:35,3:14]
fa1<-mat[39:45,3:14]; fo1<-mat[49:51,3:14]; ph1<-mat[53:60,3:14]
ba1<-mat[64:67,3:14]; cv1<-mat[70:78,3:14]; cd1<-mat[83:87,3:14]
fc11<-cbind(mat[1:8,10],mat[1:8,12]);V11<-cbind(mat[12:20,10]mat[12:20,12])
P11<-cbind(mat[25:35,10],mat[25:35,12]);fa11<-cbind(mat[39:45,10],mat[39:45,12])
fo11<-cbind(mat[49:51,10],mat[49:51,12]);ph11<-cbind(mat[53:60,10],mat[53:60,12])
ba11<-cbind(mat[64:67,10],mat[64:67,12]); cv11<-cbind(mat[70:78,10],mat[70:78,12])
cd11<-cbind(mat[83:87,10],mat[83:87,12])
```

```

MM<-rbind(fc1,V1,P1,fa1,fo1,ph1,ba1,cv1,cd1)
tr<-c(1:8,12:20,25:35,39:45,49:51,53:60,64:67,70:78,83:87)
##### Transformação da Matriz de treino #####
matcor1=matrix(0,64,3); matcor2=matrix(0,64,2); matcor3=matrix(0,64,8)
matcor4=matrix(0,64,7); matcor5=matrix(0,64,7)
for(i in 1:88){
  if(MM[i,1]=="retangular") matcor1[i,]=c(1,0,0)
  if(MM[i,1]=="irregular") matcor1[i,]=c(0,1,0)
  if(MM[i,1]=="quadrada") matcor1[i,]=c(0,0,1)
}
for(i in 1:88){
  if(MM[i,6]=="vermelho") matcor3[i,]=c(1,0,0,0,0,0,0,0)
  if(MM[i,6]=="preto") matcor3[i,]=c(0,1,0,0,0,0,0,0)
  if(MM[i,6]=="amarelo") matcor3[i,]=c(0,0,1,0,0,0,0,0)
  if(MM[i,6]=="azul") matcor3[i,]=c(0,0,0,1,0,0,0,0)
  if(MM[i,6]=="branco") matcor3[i,]=c(0,0,0,0,1,0,0,0)
  if(MM[i,6]=="verde") matcor3[i,]=c(0,0,0,0,0,1,0,0)
  if(MM[i,6]=="cinza") matcor3[i,]=c(0,0,0,0,0,0,1,0)
  if(MM[i,6]=="laranja") matcor3[i,]=c(0,0,0,0,0,0,0,1)
}
TC<-matcor1; CP<-matcor3; MM1<-cbind(TC,CP)
##### Estimação #####
P1<-MM1[1:8,]; P2<-MM1[9:16,]; P3<-MM1[17:27,]; P4<-MM1[28:34,]
P5<-MM1[35:37,]; P6<-MM1[38:46,]; P7<-MM1[47:51,]; P8<-MM1[52:60,]
P9<-MM1[60:64,]; Ptt<-rbind(P1,P2,P3,P4,P5,P6,P7,P8,P9)
### Estimação das variaveis #####
P1p<-0; P2p<-0; P3p<-0; P4p<-0
P5p<-0; P6p<-0; P7p<-0; P8p<-0; P9p<-0
for(j in 1:20){
  P1p[j]<-sum(P1[,j])/8

```



```

P2p[j]<-sum(P2[,j])/8
P3p[j]<-sum(P3[,j])/11
P4p[j]<-sum(P4[,j])/7
P5p[j]<-sum(P5[,j])/3
P6p[j]<-sum(P6[,j])/9
P7p[j]<-sum(P7[,j])/5
P8p[j]<-sum(P8[,j])/9
P9p[j]<-sum(P9[,j])/5
}

#### Programa ####

##### Parte multinomial #####

##### Transformação da Matriz original #####

matcor11=matrix(0,88,3); matcor21=matrix(0,88,2); matcor31=matrix(0,88,8)
matcor41=matrix(0,88,7); matcor51=matrix(0,88,7)
for(i in 1:88){
  if(mat[i,3]=="retangular") matcor11[i,]=c(1,0,0)
  if(mat[i,3]=="irregular") matcor11[i,]=c(0,1,0)
  if(mat[i,3]=="quadrada") matcor11[i,]=c(0,0,1)
}

for(i in 1:88){
  if(mat[i,8]=="vermelho") matcor31[i,]=c(1,0,0,0,0,0,0,0)
  if(mat[i,8]=="preto") matcor31[i,]=c(0,1,0,0,0,0,0,0)
  if(mat[i,8]=="amarelo") matcor31[i,]=c(0,0,1,0,0,0,0,0)
  if(mat[i,8]=="azul") matcor31[i,]=c(0,0,0,1,0,0,0,0)
  if(mat[i,8]=="branco") matcor31[i,]=c(0,0,0,0,1,0,0,0)
  if(mat[i,8]=="verde") matcor31[i,]=c(0,0,0,0,0,1,0,0)
  if(mat[i,8]=="cinza") matcor31[i,]=c(0,0,0,0,0,0,1,0)
  if(mat[i,8]=="laranja") matcor31[i,]=c(0,0,0,0,0,0,0,1)
}

TC1<-matcor11; CP1<-matcor31; Mat1<-cbind(TC1,CP1); vali<-mat[-tr,]

```

```

T1<-MM1[1:8,]; T2<-MM1[9:16,]; T3<-MM1[17:27,]; T4<-MM1[28:34,]
T5<-MM1[35:37,]; T6<-MM1[38:46,]; T7<-MM1[47:51,]; T8<-MM1[52:60,]
T9<-MM1[60:64,]; n<-c(3); I<- matrix(0,nrow=n,ncol=n)
I[row(I)==col(I)] <- 1
M1<-apply(fc11,2,mean);S1<-cov(fc11);M2<-apply(V11,2,mean); S2<-cov(V11)
M3<-apply(P11,2,mean);S3<-cov(P11); M4<-apply(fa11,2,mean);S4<-cov(fa11)
M5<-apply(fo11,2,mean);S5<-cov(fo11);M6<-apply(ph11,2,mean);S6<-cov(ph11)
M7<-apply(ba11,2,mean);S7<-cov(ba11);M8<-apply(cv11,2,mean);S8<-cov(cv11)
M9<-apply(cd11,2,mean);S9<-cov(cd11); S<-cov(cbind(MM[8],MM[10]))
SS1<-sum(diag(S))

#### Classificação com os dois tipos de distribuição Sigma*I####
p1w<-0; p2w<-0; p3w<-0; p4w<-0; p5w<-0; p6w<-0
p7w<-0; p8w<-0; p9w<-0; Gg<-0; x1<-matrix(c(rep(0)),88,2)
for(i in 1:88){
  x1[,]<-c(mat[,10],mat[,12])
  p1w[i]<-log(prod(P1p^Mat1[i,]))-(t(x1[i,]-M1)%*(x1[i,]-(M1)))/(2*SS1)
  p2w[i]<-log(prod(P2p^Mat1[i,]))-(t(x1[i,]-M2)%*(x1[i,]-(M2)))/(2*SS1)
  p3w[i]<-log(prod(P3p^Mat1[i,]))-(t(x1[i,]-M3)%*(x1[i,]-(M3)))/(2*SS1)
  p4w[i]<-log(prod(P4p^Mat1[i,]))-(t(x1[i,]-M4)%*(x1[i,]-(M4)))/(2*SS1)
  p5w[i]<-log(prod(P5p^Mat1[i,]))-(t(x1[i,]-M5)%*(x1[i,]-(M5)))/(2*SS1)
  p6w[i]<-log(prod(P6p^Mat1[i,]))-(t(x1[i,]-M6)%*(x1[i,]-(M6)))/(2*SS1)
  p7w[i]<-log(prod(P7p^Mat1[i,]))-(t(x1[i,]-M7)%*(x1[i,]-(M7)))/(2*SS1)
  p8w[i]<-log(prod(P8p^Mat1[i,]))-(t(x1[i,]-M8)%*(x1[i,]-(M8)))/(2*SS1)
  p9w[i]<-log(prod(P9p^Mat1[i,]))-(t(x1[i,]-M9)%*(x1[i,]-(M9)))/(2*SS1)
  Gg<-t(rbind( p1w,p2w,p3w,p4w,p5w,p6w,p7w,p8w,p9w))
}

#### Classificação com os dois tipos de distribuição Sigma comum ####
p11w<-0; p12w<-0; p13w<-0; p14w<-0; p15w<-0; p16w<-0; p17w<-0
p18w<-0; p19w<-0; Gg1<-0; x1<-matrix(c(rep(0)),88,2)
for(i in 1:88){

```

```

x1[,]<-c(mat[,10],mat[,12])
p11w[i]<-log(prod(P1p^Mat1[i,]))-(1/2)*t(x1[i,]-M1)%*%solve(S)%*(x1[i,]-M1)
p12w[i]<-log(prod(P2p^Mat1[i,]))-(1/2)*t(x1[i,]-M2)%*%solve(S)%*(x1[i,]-M2)
p13w[i]<-log(prod(P3p^Mat1[i,]))-(1/2)*t(x1[i,]-M3)%*%solve(S)%*(x1[i,]-M3)
p14w[i]<-log(prod(P4p^Mat1[i,]))-(1/2)*t(x1[i,]-M4)%*%solve(S)%*(x1[i,]-M4)
p15w[i]<-log(prod(P5p^Mat1[i,]))-(1/2)*t(x1[i,]-M5)%*%solve(S)%*(x1[i,]-M5)
p16w[i]<-log(prod(P6p^Mat1[i,]))-(1/2)*t(x1[i,]-M6)%*%solve(S)%*(x1[i,]-M6)
p17w[i]<-log(prod(P7p^Mat1[i,]))-(1/2)*t(x1[i,]-M7)%*%solve(S)%*(x1[i,]-M7)
p18w[i]<-log(prod(P8p^Mat1[i,]))-(1/2)*t(x1[i,]-M8)%*%solve(S)%*(x1[i,]-M8)
p19w[i]<-log(prod(P9p^Mat1[i,]))-(1/2)*t(x1[i,]-M9)%*%solve(S)%*(x1[i,]-M9)
Gg1<-t(rbind( p11w,p12w,p13w,p14w,p15w,p16w,p17w,p18w,p19w))
}
G<-Gg[-tr,]; classifica<-matrix(0,length(G[,1]),2); x<-1:9
for(i in 1:length(G[,1])){
ii<-order(G[i,],x)
classifica[i,]<-rbind(G[i,],x)[,ii][,9]
}
compara<-cbind(classifica[,2],classe.true[-tr,])
erro<-compara[,1]-compara[,2]; mat_conf_1<-matrix(0,9,9)
mat_conf11 = matrix(0,9,9); tc<-c(3,4,3,3,1,3,2,3,2); tc=cumsum(tc)
ntc<-length(tc); resh1=table(compara[1:tc[1]]); resh1
mat_conf_1[as.numeric(names(resh1))]=table(compara[1:tc[1]])
mat_conf_1; mat_conf_1[,1]; mat_conf11[1,]=mat_conf_1[,1]; mat_conf11
for(i in 1:(ntc-1)){
mat_conf_1<-matrix(0,9,9)
resh1<-table(compara[(tc[i]+1):tc[i+1]])
mat_conf_1[as.numeric(names(resh1))]=table(compara[(tc[i]+1):tc[i+1]])
mat_conf_1
mat_conf11[i+1,]=mat_conf_1[,1]
mat_conf11

```

```

}

mat_conf11; Pw1<-NULL; ERF1<-NULL; ERR1<-NULL
for( i in 1:9){
Pw1[i]<-(sum(mat_conf11[i,])/sum(mat_conf11))
ERR1[i]<-(mat_conf11[i,i])/sum(mat_conf11[i,])
ERF1[i]<-ERR1[i]*Pw1[i]
ERRO_Total1<-1-sum(ERF1)
}

ERRO_Total1; G1<-Gg1[-tr,]; classifical<-matrix(0,length(G1[,1]),2); x<-1:9
for(i in 1:length(G1[,1])){
ii1<-order(G1[i,],x)
classifical[i,]<-rbind(G1[i,],x)[,ii1][,9]
}

compara1<-cbind(classifical[,2],classe.true[-tr,])
erro1<-compara1[,1]-compara1[,2]; erro; mat_conf<-matrix(0,9,9)
mat_conf1 = matrix(0,9,9); tc<-c(3,4,3,3,1,3,2,3,2); tc=cumsum(tc)
ntc<-length(tc); resh=table(compara1[1:tc[1]]); resh
mat_conf[as.numeric(names(resh))]=table(compara1[1:tc[1]])
mat_conf; mat_conf[,1]; mat_conf1[1,]=mat_conf[,1]; mat_conf1
for(i in 1:(ntc-1)){
mat_conf<-matrix(0,9,9)
resh<-table(compara1[(tc[i]+1):tc[i+1]])
mat_conf[as.numeric(names(resh))]=table(compara1[(tc[i]+1):tc[i+1]])
mat_conf
mat_conf1[i+1,]=mat_conf[,1]
mat_conf1
}

mat_conf1; Pw2<-NULL; ERF2<-NULL; ERR2<-NULL
for( i in 1:9){
Pw2[i]<-(sum(mat_conf1[i,])/sum(mat_conf1))

```

108

```
ERR2[i] <- (mat_conf1[i, i]) / sum(mat_conf1[i, ])
ERF2[i] <- ERR2[i] * Pw2[i]
ERRO_Total2 <- 1 - sum(ERF2)
}
mat_conf11; mat_conf1; ERRO_Total1; ERRO_Total2
```