4. Classificadores

No capítulo anterior, discutiu-se a importância do uso de técnicas de redução de dimensionalidade em sistemas de reconhecimento de padrões. Essas técnicas além de diminuírem consideravelmente o número de parâmetros a ser utilizado como entrada por modelos posteriores de classificação, também procuram realçar as principais características discriminatórias dos diversos grupos ou padrões existentes.

Um sistema de reconhecimento de padrões, no entanto, não é constituído apenas por métodos de redução de dimensionalidade. Na verdade, pode-se dizer que um sistema de reconhecimento de padrões, desconsiderando todo o processo de captura e pré-processamento da imagem, é composto basicamente por três etapas. A primeira corresponde à localização e isolamento dos objetos de interesse na imagem. A segunda à extração de características discriminatórias: usualmente esta etapa envolve técnicas de redução de dimensionalidade como as exemplificadas no capítulo anterior. A terceira e última etapa corresponde à classificação propriamente dita dos padrões, onde se tem por objetivo determinar a que grupo ou padrão uma determinada amostra ou nova observação pertence. Esta última etapa é discutida com mais detalhes neste capítulo.

Um classificador ou modelo de classificação consiste, basicamente, em mapear um conjunto de observações no espaço n-dimensional em um conjunto de classes ou grupos. Pode-se encontrar na literatura de reconhecimento de padrões diversos métodos lineares e não-lineares que constrõem este tipo de mapeamento [John92, Gree78, Manl94]. Muitos deles estão associados a algum tipo de medida de proximidade entre uma dada observação e grupos de observações pré-existentes e já definidos.

Alguns modelos de classificação serão discutidos nas seções seguintes deste capítulo. Na seção 4.1 apresenta-se o princípio das distâncias Euclideana e de Mahalanobis e o conceito de classificação com relação ao centróide. A seguir, na seção 4.2, a teoria das redes neurais de Funções Base Radiais (RBF) é abordada, descrevendo-se os dois algoritmos de aprendizado testados neste trabalho. Por fim, a terceira e última seção 4.3 apresenta o classificador de máxima probabilidade e suas regras de discriminação com relação às matrizes de covariância. Nesta última seção, discute-se uma nova abordagem que considera a classificação de padrões através de matrizes de covariâncias "misturadas".

4.1 Distâncias Euclideana e de Mahalanobis

Diversos problemas em análise multivariada de dados podem ser vistos como problemas de cálculo de distâncias entre observações ou amostras, ou entre observações e grupos de observações. A idéia básica se resume em considerar que amostras ou observações pertencentes a um mesmo grupo ou objeto devem se encontrar, no respectivo espaço n-dimensional, próximas umas das outras, possuindo distâncias entre si relativamente pequenas. Existem diversas medidas de distâncias propostas na literatura de análise multivariada de dados. Esta seção descreve as duas mais frequentemente utilizadas em problemas de reconhecimento de padrões: a distância Euclideana e a distância de Mahalanobis.

4.1.1 Distância Euclideana

Considere-se, inicialmente, o caso onde existem k objetos definidos por n variáveis $X_1, X_2, ..., X_n$. Os valores para o objeto i podem ser denotados como $x_{i1}, x_{i2}, ..., x_{in}$ e os

valores para o objeto j como $x_{j1}, x_{j2}, ..., x_{jn}$. Deseja-se, então, medir a distância entre esses dois objetos. Se n for igual a 2, ou seja, se existirem apenas duas variáveis de medida, então os valores dos objetos em questão podem ser plotados conforme a figura 4.1 abaixo.

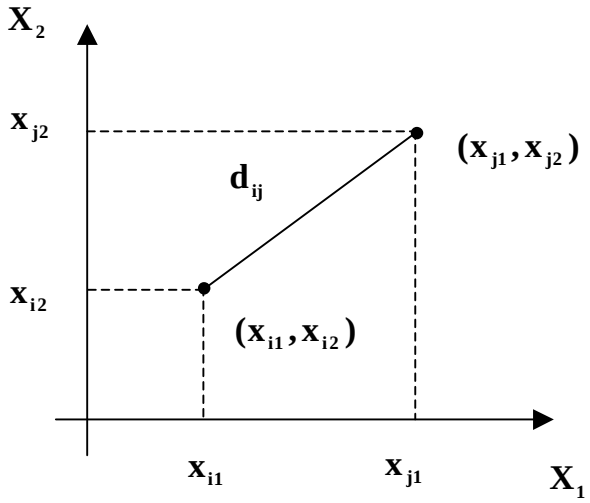


Figura 4.1 – Distância Euclideana entre os objetos i e j para o caso de duas variáveis (**n**=2).

De acordo com o teorema de Pythagoras, a distância Euclideana d_{ij} entre o objeto i e o objeto j é dada pela fórmula

$$d_{ij} = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2} \ .$$

Para o caso n-dimensional de variáveis, a distância Euclideana pode ser calculada como

$$d_{ij} = \sqrt{\overset{n}{} (x_{in} - x_{jn})^2}$$

cuja forma matricial quadrática corresponde a

$$\mathrm{d}_{ij}^2 = (\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j).$$

Pela fórmula acima, pode-se observar que se as variáveis do modelo medirem valores em escalas bem distintas então aquelas que possuírem maior ordem de grandeza dominarão os cálculos de distância. Para resolver esta distorção costuma-se realizar, nestas situações, algum tipo de normalização nas variáveis originais $X_1, X_2, ..., X_n$ fazendo com que tais variáveis exerçam a mesma influência sobre o cálculo das distâncias. Uma normalização usualmente empregada é aquela que transforma as variáveis originais $X_1, X_2, ..., X_n$ em variáveis $Z_1, Z_2, ..., Z_n$ com média igual a 0 e variância igual 1. O cálculo desta transformação pode ser visto na seção 3.1.2 referente ao cálculo das componentes principais, já detalhado no capítulo 3.

Embora a distância Euclideana seja simples de calcular e eficiente em determinadas aplicações, a mesma possui uma forte limitação. O cálculo dessa distância não leva em consideração a correlação entre as variáveis do problema. Isto significa que se duas ou mais variáveis medirem essencialmente o mesmo objeto e, por conseguinte, tiverem uma forte correlação entre si, estas variáveis terão, aproximadamente, a mesma contribuição para as medidas de distância que uma outra variável que seja independente das demais. Desta forma, características expressas redundantemente em mais de uma variável acabam tendo peso maior na classificação, ocasionando, muitas vezes, uma pobre representação discriminatória dos dados. Em termos geométricos, esta limitação impõe que os grupos definidos pelas respectivas observações no espaço n-dimensional sejam descritos por hiperesferas , ou seja, todas as variações de distância são consideradas iguais em todas as direções.

4.1.2 Distância de Mahalanobis

Uma medida de distância entre um objeto i e um objeto j que leva em consideração a correlação entre as variáveis do modelo é a distância de Mahalanobis D_{ij}, cuja fórmula quadrática pode ser escrita como

$$\mathrm{D}_{ij}^2 = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{S}^{-1} (\mathbf{x}_i - \mathbf{x}_j)$$

onde

$$\mathbf{x}_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{in} \end{bmatrix} \quad \text{e} \quad \mathbf{x}_j = \begin{bmatrix} x_{j1} \\ x_{j2} \\ \vdots \\ x_{jn} \end{bmatrix}$$

e $\mathbf{S}$ é a matriz de covariânça das n variáveis.

Esta medida pressupõe que, embora as matrizes de covariância sejam iguais para todos os grupos existentes, as mesmas descrevem hiperelipsóides no espaço n-dimensional e não hiperesferas como na distância Euclideana. Assim, o cálculo da distância irá variar de acordo com a variância e a covariância das variáveis. No caso particular onde $\mathbf{S}$ é igual à matriz identidade e, consequentemente, as variáveis possuem variância 1 e covariância zero, a distância de Mahalanobis se iguala à distância Euclideana.

4.1.3 Classificadores de Distância

Usualmente, tanto a distância Euclideana quanto a de Mahalanobis são utilizadas para medir a separação entre uma observação e os centróides dos grupos definidos por todas as observações.

Supõe-se que $x_{i1}, x_{i2}, ..., x_{in}$ são valores de uma dada amostra i das variáveis $X_1, X_2, ..., X_n$. Estas variáveis possuem vetores média ou centróides $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, ..., \boldsymbol{\mu}_g$ para os g grupos existentes. Pode-se, portanto, medir a distância de Mahalanobis entre a observação i e o centróide de um grupo j através da fórmula

$$\mathrm{D}_{ij}^2 = (\mathbf{x}_i - \boldsymbol{\mu}_j)^T \mathbf{S}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_j)$$

onde $\mathbf{x}_i^T = (x_{i1}, x_{i2}, ..., x_{in})$, $\boldsymbol{\mu}_j^T = [\mu_{j1}, \mu_{j2}, ..., \mu_{jn}]$.

Para um caso geral, quando deseja-se saber a que grupo uma nova observação pertence, pode-se fazer uso destas funções distância para classificar tal observação. Nestas situações, costuma-se comparar a distância entre a nova observação e os centróides $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, ..., \boldsymbol{\mu}_g$ dos g grupos existentes, escolhendo-se aquele grupo cuja distância é mínima com relação à observação.

Este classificador é comumente chamado de classificador de distância com relação ao centróide e um exemplo pode ser visualizado na figura 4.2. Nas figuras 4.2(a) e 4.2(b), $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\mu}_3$ são os centróides das 3 grupos existentes; $\mathbf{x_i}$ é a nova observação a ser classificada; $\mathrm{d}_{i1}, \mathrm{d}_{i2}, \mathrm{d}_{i3}$ e $\mathrm{D}_{i1}, \mathrm{D}_{i2}, \mathrm{D}_{i3}$ são, respectivamente, as funções distância Euclideana e de Mahalanobis da amostra $\mathbf{x_i}$ com relação aos centróides de cada grupo. Pode-se

observar que embora o classificador de distância seja bastante simples de implementar, o mesmo sofre a limitação de não considerar, para o cálculo Euclideano, as formas das classes e, para o cálculo da distância de Mahalanobis, a possibilidade de se ter formas arbitrárias ou distintas para os diversos grupos ou classes.

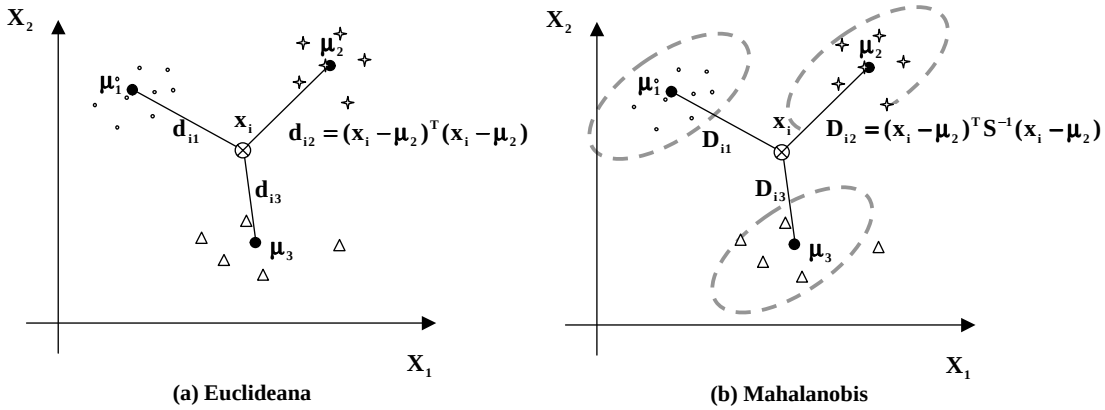


Figura 4.2 – Representação geométrica do classificador de distância para o caso de duas variáveis (**n**=2)

Ainda que as distâncias Euclideana e de Mahalanobis tenham fórmulas matemáticas parecidas, ambas possuem características bastante singulares. Estas características, muitas vezes, podem fazer com que o uso de uma determinada distância seja a melhor solução para alguns problemas e a pior para outros. Assim sendo, esta escolha de qual medida de distância deve-se usar para medir a proximidade de observações em espaços n-dimensionais depende muito da proposta de similariedade em que se está interessado e da natureza dos dados a serem avaliados, como será visto na seção 6.1 do capítulo 6 de análise de resultados dos classificadores de distância.

4.2 Redes Neurais de Funções Base Radiais

As redes neurais são modelos não-lineares de mapeamento de variáveis inspiradas na biologia do sistema nervoso humano. São compostas por diversos elementos, chamados neurônios, altamente interconectados e que atuam de forma paralela. Toda conexão entre dois neurônios possui um valor ou peso que é atualizado de acordo com determinados algoritmos de otimização, conhecidos como algoritmos de aprendizado ou de treinamento.

Existem inúmeros algoritmos de treinamento na literatura de redes neurais, tais como o *back-propagation,* bem como diversas topologias de rede [Hayk94]. Este trabalho irá se concentrar em um tipo específico de redes neurais muito aplicado em problemas de reconhecimento de padrões, as redes neurais de funções base radiais ou *radial basis function networks* (RBFs) [Bish97].

4.2.1 Breve Histórico

Os métodos de funções base radiais foram originalmente empregados por Powell, em 1987, para solucionar problemas de interpolação exata de um conjunto de pontos em um espaço multi-dimensional. A idéia básica desse tipo de interpolação consistia em mapear, exatamente, cada vetor de entrada $\mathbf{x}$ pertencente a um espaço n-dimensional em um vetor de saída $\mathbf{t}$.

Para melhor compreender esta idéia, considere-se, por exemplo, um conjunto de dados K formado por vetores de entrada $\mathbf{x^k}$ e seus respectivos vetores de saída $\mathbf{t^k}$, contendo t_c^k componentes. O problema consiste em encontrar uma função de mapeamento $h(x)$ de tal forma que

$$h_c(\mathbf{x}^k) = t_c^k, \qquad k = 1,...,K.$$

O método proposto por Powell introduz um conjunto de Q funções da forma $\mathbf{\Phi}\left(\left\|\mathbf{x} - \mathbf{x}^k\right\|\right)$, uma para cada ponto de entrada $\mathbf{x^k}$, ou seja, $Q = K$. A função $\mathbf{\Phi(.)}$ é uma função base radial não-linear cuja resposta cresce ou decresce monotonicamente dependendo da distância $\left\|\mathbf{x} - \mathbf{x}^k\right\|$, geralmente Euclideana, entre os vetores $\mathbf{x}$ e $\mathbf{x^k}$. A saída do mapeamento pode ser calculada como a combinação linear das Q funções base radiais, de tal forma que

$$h_c(\mathbf{x}^k) = \underset{q}{\quad} w_{cq}\mathbf{\Phi}(\left\|\mathbf{x} - \mathbf{x}^k\right\|), \qquad q = 1,...,Q \;\; e \;\; Q = K,$$

onde o peso de cada componente é dado por w_{cq}.

Este tipo de mapeamento cria uma função de interpolação que passa exatamente sobre cada ponto $\mathbf{x^k}$ do conjunto de dados utilizado como entrada. No entanto, em diversas situações não é este o comportamento que pretende-se obter. Principalmente em circunstâncias onde tem-se dados ruidosos que fazem com que a função exata de interpolação passe a apresentar uma forma altamente oscilatória. Nestes casos, a função de mapeamento capaz de apresentar a melhor generalização é aquela que contém um comportamento mais suave e que calcula valores médios para os dados ruidosos. Um exemplo simples de uma interpolação exata utilizando funções base radiais pode ser vista na figura 4.3. Os pontos ligados pela linha contínua representam a idéia de uma interpolação gerada por funções base radiais, enquanto a linha tracejada representa uma interpolação com melhor generalização.

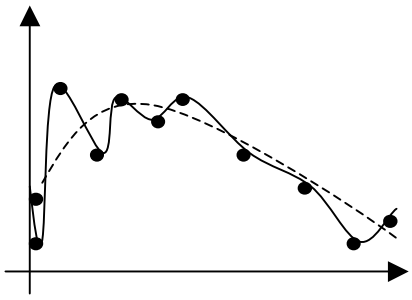


Figura 4.3 – Um exemplo simples de interpolação exata.

Além dos problemas de generalização, a interpolação exata possui uma outra limitação. Como existe uma função base radial para cada ponto pertencente ao conjunto de dados, se existir então uma grande quantidade de pontos a implementação desse mapeamento pode se tornar inviável.

Em 1988, Broomhead e Lowe introduziram algumas modificações significativas nesta abordagem de interpolação exata e modelaram, então, as redes neurais RBF.

4.2.2 Modelagem da Rede RBF

As principais diferenças entre as redes neurais RBF e os modelos de funções base radiais propostos por Powell (seção 4.2.1) estão no fato de a rede neural poder criar uma função de interpolação que apresenta um comportamento menos oscilatório e, consequentemente, uma melhor generalização dos dados. Além disso, a modelagem neural contém uma quantidade de funções base radiais consideravelmente menor.

Nas redes RBF, o número de funções base radiais passa a ser determinado pela complexidade do mapeamento descrito pelos pontos e não pela quantidade dos mesmos. Isto

faz com que o número Q de funções base radiais não precise ser igual ao número K de pontos do conjunto de dados. Na verdade, Q é geralmente muito menor que K ($Q << K$). Por conseguinte, os centros das funções base radiais não mais necessariamente coincidem com todos os dados de entrada, como acontecia antes. Ao contrário, a determinação desses centros passa a fazer parte do treinamento da rede neural, bem como as larguras das funções base radiais que podem ser feitas distintas uma das outras. Ambos os parâmetros são descritos com mais detalhes a seguir.

O cálculo da função de mapeamento da rede neural RBF pode ser definido como

$$y_c(\mathbf{x}) = \sum_{j=1}^{Q} w_{cj} \mathbf{\Phi}_j(\mathbf{x}) \tag{4.2.2.1}$$

onde $\mathbf{\Phi_j}$ é a função base radial de ativação do neurônio j e w_{cj} é o peso ou conexão entre o neurônio j e o neurônio y_c de saída. A função base radial mais utilizada neste tipo de rede é a função gaussiana, que pode ser escrita como

$$\mathbf{\Phi}_j(\mathbf{x}) = exp(-\frac{\left\|\mathbf{x} - \boldsymbol{\mu}_j\right\|^2}{2\sigma_j^2}), \tag{4.2.2.2}$$

onde σ e $\boldsymbol{\mu}$ são, respectivamente, o desvio padrão e o vetor média da função de ativação que define o neurônio j, e $\mathbf{x}$ um vetor n-dimensional de entrada com elementos x_i, onde $i = 1,2,...,n$.

Ao contrário das demais redes neurais que calculam a ativação de um neurônio através do produto escalar entre o vetor de entrada e o vetor de pesos – por exemplo, as redes

perceptron de múltiplas camadas [Hayk94] – as redes RBF calculam essa ativação medindo-se a distância entre o centróide da respectiva função de ativação e o vetor de entrada.

A fórmula 4.2.2.1 pode ser visualizada como um diagrama de rede neural, conforme figura 4.4. Cada função base radial $\mathbf{\Phi_j}$ representa um único neurônio na camada escondida. As linhas que conectam as entradas x_i às funções radiais $\mathbf{\Phi_j}$ representam os elementos μ_{ij} do vetor $\boldsymbol{\mu}_j$. Os pesos w_{cj} são apresentados como linhas que conectam as funções base $\mathbf{\Phi_j}$ aos neurônios de saída y_c. Ainda que não se tenha nenhuma restrição com relação ao número de camadas utilizado pelas redes RBF, normalmente não se considera topologias contendo mais de uma camada escondida de neurônios [Bish97].

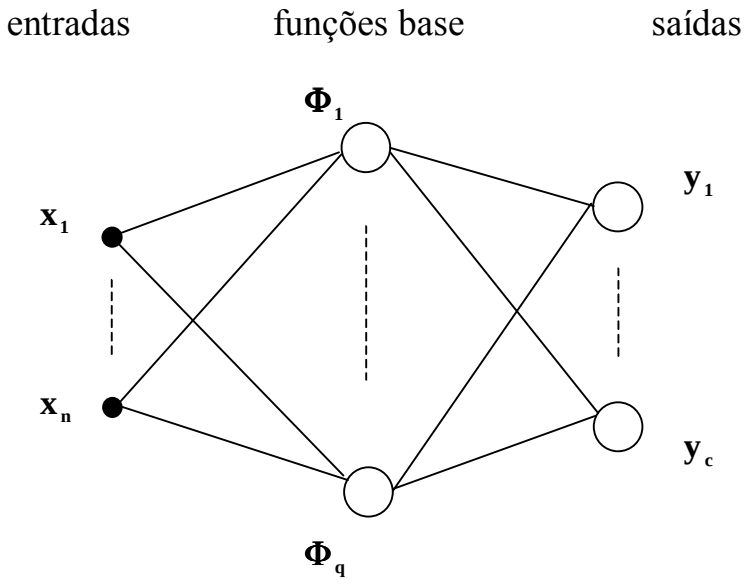


Figura 4.4 – Topologia de uma rede neural RBF

Um aspecto muito importante das redes neurais RBF é a distinção entre os papéis desempenhados pelos pesos ou conexões entre a camada de entrada e a camada escondida

(primeira camada de pesos), e entre a camada escondida e a camada de saída(segunda camada de pesos).

As redes RBF podem ser interpretadas de tal forma que os pesos da primeira camada, cujos valores se referem aos parâmetros das funções base radiais de ativação, possam ser determinados por técnicas não-supervisionadas de treinamento. Isto permite que o treinamento do modelo neural possa ser feito, portanto, em dois estágios. No primeiro estágio, o estágio não-supervisionado, somente o conjunto de dados de entrada $\{\mathbf{x}^k\}$ é utilizado para determinar os parâmetros das funções de ativação - por exemplo, no caso de se utilizar funções gaussianas determina-se os valores $\boldsymbol{\mu}_j$ e σ_j das respectivas funções. Depois, com os pesos da primeira camada fixados, utiliza-se o conjunto de dados de entrada $\{\mathbf{x}^k\}$ e o conjunto correspondente de saída $\{\mathbf{t}^k\}$ para se obter os pesos da segunda camada, através de um treinamento supervisionado.

Existem algumas técnicas não supervisionadas de otimização dos parâmetros das funções de ativação das redes RBF, e duas delas serão vistas na seção 4.2.4. Na próxima seção, admitindo-se que os parâmetros das funções base radiais já tenham sido determinados, é discutido o problema de se encontrar os pesos da segunda camada das redes RBF, através de um treinamento supervisionado.

4.2.3 Treinamento da Rede RBF

Considere-se a função de mapeamento de uma rede neural RBF descrita pela fórmula 4.2.2.1. Admitindo que os parâmetros das funções base radiais estejam fixos, pretende-se otimizar os pesos da segunda camada minimizando-se uma função de erro. Esta função de erro é frequentemente calculada a partir da soma quadrática das diferenças entre a resposta da

rede referente aos dados aplicados na entrada e as respectivas saídas desejadas de tal forma que

$$E(\mathbf{w}) = \frac{1}{2} \sum_k \sum_c \left\{ \sum_j w_{cj} \mathbf{\Phi}_j(\mathbf{x}^k) - t_c^k \right\}^2, \qquad (4.2.3.1)$$

onde t_c^k é o valor desejado ou *target* na saída para o neurônio c quando o vetor de entrada $\mathbf{x^k}$ é apresentado na entrada da rede. Diferenciando-se a expressão 4.2.3.1 com relação w_{cj} e igualando-a a zero, tem-se a equação que descreve o problema dos mínimos quadrados

$$\sum_k \left\{ \sum_j w_{cj} \mathbf{\Phi}_j(\mathbf{x}^k) - t_c^k \right\} \mathbf{\Phi}_j(\mathbf{x}^k) = 0. \qquad (4.2.3.2)$$

Para solucionar a equação 4.2.3.2 é conveniente escrevê-la na forma matricial, como se segue

$$(\mathbf{\Phi}^{\mathbf{T}}\mathbf{\Phi})\mathbf{W}^{\mathbf{T}} = \mathbf{\Phi}^{\mathbf{T}}\mathbf{T}. \qquad (4.2.3.3)$$

A matriz $\mathbf{\Phi}$ tem dimensão K x Q e elementos $\mathbf{\Phi}_j^k \equiv \mathbf{\Phi}_j(x^k)$, $\mathbf{W}$ tem dimensão c x Q e elementos w_{cj}, e $\mathbf{T}$ tem dimensões K x c e elementos t_c^k . A matriz $\mathbf{\Phi}^{\mathbf{T}}\mathbf{\Phi}$ é uma matriz quadrada de dimensão Q x Q. Se esta matriz quadrada for não singular, então pode-se invertê-la e calcular uma solução para a equação 4.2.3.3 da forma

$$\mathbf{W}^{\mathbf{T}} = \mathbf{\Phi}^{\lambda}\mathbf{T}, \qquad (4.2.3.4)$$

onde $\mathbf{\Phi}^{\lambda}$ é uma matriz Q x K conhecida como pseudo-inversa de $\mathbf{\Phi}$ (Golub e Kahan, 1965; Rao e Mitra, 1971) dada por

$$\mathbf{\Phi}^{\lambda} \equiv (\mathbf{\Phi}^{\mathbf{T}}\mathbf{\Phi})^{-1}\mathbf{\Phi}^{\mathbf{T}}. \qquad (4.2.3.5)$$

Embora a matriz $\mathbf{\Phi}$ seja geralmente retangular e, por conseguinte, não inversível, a sua pseudo-inversa contém a propriedade $\mathbf{\Phi}^{\lambda}\mathbf{\Phi} = \mathbf{I}$, onde $\mathbf{I}$ é a matriz identidade. Observa-se, ainda, que se $\mathbf{\Phi}^{\mathbf{T}}\mathbf{\Phi}$ for singular então a equação 4.2.3.3 não tem uma única solução. Nestes casos, define-se a matriz pseudo-inversa como

$$\mathbf{\Phi}^{\lambda} \equiv \lim_{\boldsymbol{\varepsilon}\to 0}\left(\mathbf{\Phi}^{T}\mathbf{\Phi} + \boldsymbol{\varepsilon}\mathbf{I}\right)^{-1}\mathbf{\Phi}^{T}. \qquad (4.2.3.6)$$

Pode-se mostrar [Bish97] que este limite sempre existe e que o mesmo minimiza a função erro dada pela fórmula 4.2.3.1. Na prática, a equação 4.2.3.4 é geralmente solucionada através da técnica linear algébrica de decomposição singular de valores [Stran88], evitando-se os problemas inerentes à inversibilidade da matriz $\mathbf{\Phi}$. Constata-se, assim, que os pesos da segunda camada das redes RBF podem ser encontrados através de procedimentos lineares de inversão de matrizes cujos tempos de processamento são, consequentemente, bastante rápidos.

É importante ressaltar que nada impede de se construir redes RBF cujas funções de ativação dos neurônios da camada de saída sejam não-lineares, e que contenham outro tipo menos usual de função de erro. Nestas situações, a determinação dos pesos da segunda camada passa a ser um problema mais complexo, que envolve técnicas não-lineares de

otimização. No entanto, uma das grandes vantagens das redes RBF está justamente no fato de se poder evitar tal tipo de otimização, fazendo com que o treinamento da rede seja bastante rápido comparado com os demais algoritmos de aprendizado, tais como o *backpropagation* de redes de múltiplas camadas [Hayk94].

Além disso, o processo de treinamento em dois estágios permite que os parâmetros das funções base radiais possam ser escolhidos previamente através apenas do conjunto de dados de entrada, desconsiderando-se qualquer informação relativa aos dados de saída. Este treinamento não-supervisionado sugere que os parâmetros radiais sejam definidos estimando-se a densidade de probabilidade dos dados de entrada. Duas dessas técnicas de otimização dos parâmetros das funções radiais são examinadas nas seções seguintes.

4.2.4 Otimização das Funções Base Radiais

Como dito antecedentemente, o processo de treinamento da rede neural RBF em dois estágios permite que os parâmetros das funções base radiais possam ser escolhidos previamente através apenas do conjunto de dados de entrada $\{\mathbf{x}^k\}$, desconsiderando-se qualquer informação relativa aos dados objetivados na saída $\{\mathbf{t}^k\}$. Este treinamento não-supervisionado sugere que os parâmetros radiais sejam definidos estimando-se a densidade de probabilidade dos dados $\{\mathbf{x}^k\}$.

4.2.4.1 Método Ortogonal dos Mínimos Quadrados

Um método simples de seleção dos parâmetros das funções base radiais de uma rede neural RBF e comumente implementado em *softwares* matemáticos de otimização – por

exemplo, o Matlab – é o método ortogonal dos mínimos quadrados, também chamado de *forward selection* (FS). Este método tem por objetivo fazer com que os centros $\boldsymbol{\mu}_j$ das funções base radiais sejam iguais a um sub-conjunto dos dados de treinamento.

Para um melhor entendimento deste processo, considera-se o seguinte procedimento de seleção das funções base radiais. Admite-se que existe uma rede RBF com um único neurônio inicial na sua camada escondida. Faz-se com que o centro da função base radial deste neurônio seja igual a cada ponto ou dado do conjunto de treinamento. Calcula-se, então, os pesos da segunda camada através da técnica pseudo-inversa analisada na seção 4.2.3, para o conjunto completo dos K pontos de treinamento. O ponto ou dado de entrada que apresentar o menor erro é escolhido como centro da função de ativação do neurônio e passa a não mais fazer parte do conjunto de treinamento. Nos passos seguintes, o algoritmo de treinamento vai aumentando incrementalmente o número de neurônios da camada escondida de tal forma que se Q pontos tiverem sido escolhidos como centros das funções base, $(K-Q)$ vezes a rede neural é treinada para se selecionar o novo centro dado por um dos $(K-Q)$ pontos restantes. Como em cada passo deve-se calcular a solução pseudo-inversa completa dos pesos da segunda camada para cada ponto pertencente ao conjunto de treinamento, esta abordagem é, muitas vezes, computacionalmente intensiva e demorada.

Um procedimento bem mais eficiente e que atinge os mesmos resultados é dado pelo método ortogonal dos mínimos quadrados [Bish97]. Este algoritmo de treinamento contém a mesma idéia descrita acima, de adicionar incrementalmente as funções base radiais e fazer com que seus respectivos centros sejam definidos pelos dados de treinamento disponíveis. No entanto, esta adição é feita de tal forma que a soma ponderada dos vetores definidos pelas funções de ativação dos neurônios na camada escondida seja ortogonal aos padrões definidos como saída para cada ponto do conjunto de treinamento. Com isto, é possível calcular

diretamente que dado de entrada deve ser escolhido como centro da nova função base radial e quais serão os valores dos pesos da segunda camada para que o erro quadrático na saída seja mínimo.

É válido observar que o algoritmo ortogonal dos mínimos quadrados pode treinar uma rede RBF indefinidamente, acrescentando todos os pontos que fazem parte do conjunto de treinamento como centros das inúmeras funções base radiais. Este comportamento faz com que o erro residual com relação aos dados de entrada vá a zero, prejudicando a generalização da rede. Desta forma, comumente define-se um critério de parada para este tipo de treinamento que pode ser determinado quando um certo nível de erro na saída for atingido ou quando um número específico de neurônios for adicionado na camada escondida da rede.

A idéia geométrica deste método de otimização pode ser observada na figura 4.5. Esta figura mostra observações bi-dimensionais que descrevem quatro classes distintas. Pode-se verificar que o espalhamento de cada classe é representado por funções base radiais cujos centros são feitos iguais a um sub-conjunto dos dados de treinamento.

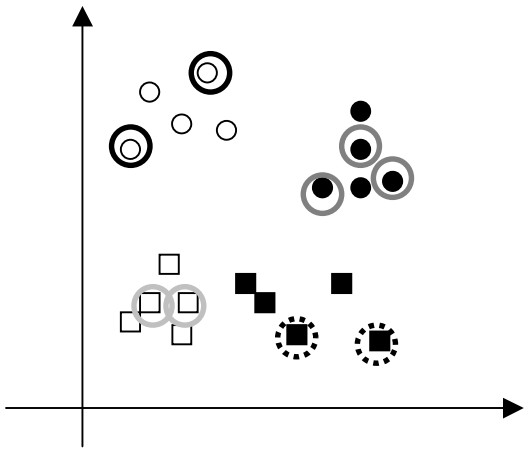


Figura 4.5 – Idéia geométrica do método ortogonal dos mínimos quadrados

Um aspecto importante do método ortogonal dos mínimos quadrados que deve ser destacado é que esta técnica de treinamento otimiza apenas os centros das funções base radiais, desconsiderando os valores σ_j relativos às suas respectivas larguras. Estes parâmetros devem ser escolhidos através de um outro procedimento como, por exemplo, a técnica que é apresentada a seguir.

4.2.4.2 Método de Mistura de Gaussianas

Em termos gerais, a densidade de probabilidade de um conjunto de pontos pode ser representada pela sobre-posição linear de funções base, onde o centro de cada função é definido por um único dado de entrada.

O método de mistura de gaussianas estima a densidade de probabilidade através da combinação linear de um conjunto de funções base, no entanto esse número Q de funções passa a ser um parâmetro do modelo e é, tipicamente, muito menor que o número K de pontos do conjunto de entrada.

Neste método de otimização, as funções base são entendidas como componentes de um modelo de densidade misturada ou *mixture density model* (Titterington et al., 1985; McLachLan and Basford, 1988) e seus respectivos parâmetros são determinados de acordo com técnicas de máxima verossimilhança ou *maximum likelihood* [Bish97]. Pode-se, portanto, escrever tal modelagem na forma de

$$p(\mathbf{x}) = \sum_{j=1}^{Q} P(j)\Phi_j(\mathbf{x})$$

onde $p(\mathbf{x})$ é a estimativa da densidade de probabilidade do conjunto de dados de entrada, $P(j)$ são os coeficientes misturados e $\mathbf{\Phi}_j(\mathbf{x})$ são as funções base gaussianas da rede neural consideradas como densidades componentes.

A densidade conjunta das K observações é o produto das densidades individuais. Esta densidade conjunta é a função de verossimilhança, definida como uma função do vetor de parâmetros desconhecidos, dada neste caso pela fórmula

$$L = \prod_k p(\mathbf{x}^k)$$

que é então maximizada com relação aos coeficientes misturados $P(j)$ e aos parâmetros das funções base. Esta maximização é feita em dois passos. No primeiro passo, calcula-se as derivadas de L em função dos parâmetros ajustáveis do modelo, ou seja, com relação à $P(j)$, $\boldsymbol{\mu}_j$ e σ_j, onde $j = 1,2,...,Q$ funções base. Depois, no segundo passo, iguala-se essas derivadas a zero e obtém-se as fórmulas da solução de máxima verossimilhança, escritas abaixo:

$$\hat{\boldsymbol{\mu}}_j = \frac{\sum_k P(j \mid \mathbf{x}^k)\mathbf{x}^k}{\sum_k P(j \mid \mathbf{x}^k)}$$

$$\hat{\boldsymbol{\sigma}}_j^2 = \frac{1}{n}\frac{\sum_k P(j \mid \mathbf{x}^k)\left\|\mathbf{x}^k - \hat{\boldsymbol{\mu}}_j\right\|^2}{\sum_k P(j \mid \mathbf{x}^k)}$$

$$\hat{P}(j) = \frac{1}{K}\sum_{k=1}^{K} P(j \mid \mathbf{x}^k)$$

onde $P(j \mid \mathbf{x}^k)$ é a probabilidade dos dados $\mathbf{x}^{\mathbf{k}}$ n-dimensionais terem sido gerados pela componente j. A dedução destas fórmulas podem ser encontradas na referência [Bish97].

Existem diversos algoritmos não-lineares de otimização que fazem uso dessas equações para se estimar os parâmetros ajustáveis do modelo - um exemplo é o método estocástico em tempo real ou *sthocastic on-line technique* proposto por Travén em 1991 [Bish97]. Alternativamente, esses parâmetros podem ser encontrados através de um processo iterativo de estimativas que comumente se baseia na técnica de EM(*expectation-maximization*) [Bish97]. Este algoritmo evita as complexidades de uma otimização não-linear e geralmente é implementado em problemas de otimização desta natureza.

Uma vez que o modelo de densidade misturada tenha sido otimizado e os parâmetros das funções base tenham sido definidos e fixados, os coeficientes $P(j)$ são descartados e as funções base são utilizadas para se determinar os pesos da segunda camada da rede RBF. Estes pesos são encontrados pelo procedimento de treinamento supervisionado examinado na seção 4.2.3, onde faz-se uso tanto do conjunto de dados de entrada $\{\mathbf{x}^{\mathbf{k}}\}$ como do correspondente conjunto de dados de saída da rede $\{\mathbf{t}^{\mathbf{k}}\}$.

A idéia geométrica deste método de otimização pode ser observada na figura 4.6. Esta figura é semelhante à figura 4.5 apresentada na seção anterior que mostra observações bi-dimensionais de quatro classes distintas. Pode-se verificar que para este método de otimização o espalhamento de cada classe é representado por funções base radiais cujos centros não são mais necessariamente iguais aos dados de treinamento e cujas respectivas larguras podem diferir de classe para classe.

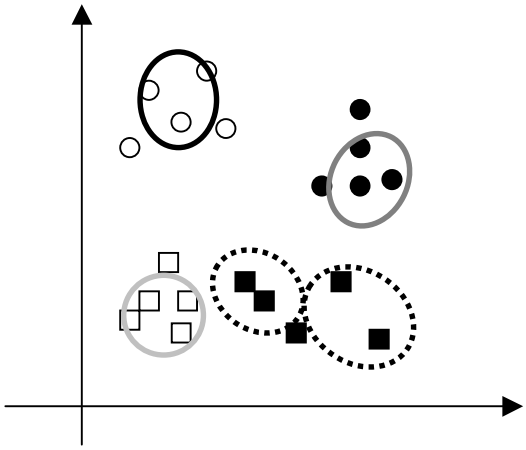


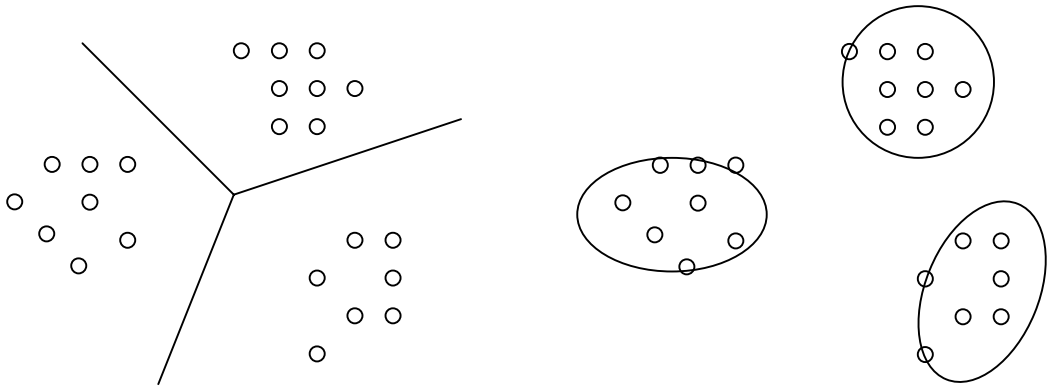
Figura 4.6 – Idéia geométrica do método de mistura de gaussianas

Constata-se que o método em questão é bem mais complexo que o método ortogonal dos mínimos quadrados discutido anteriormente e, por consequência, a mistura de gaussianas pode apresentar um processo de treinamento bem mais demorado. No entanto, o método de mistura de gaussianas permite construir redes RBF que contenham menos neurônios na camada escondida e que apresentem um comportamento bem menos sensível a escolha do conjunto de treinamento que o método ortogonal dos mínimos quadrados. Esta constatação se justifica justamente pelo fato de que ambos os parâmetros $\boldsymbol{\mu}_j$ e σ_j das funções base dos neurônios são otimizados com relação à densidade de probabilidade de todo o conjunto de dados de entrada e não mais com relação a determinadas observações de um subconjunto do mesmo.

4.2.5 Classificador RBF

Uma das aplicações mais comuns da rede neural RBF é em problemas de classificação ou reconhecimento de padrões. Nestes tipos de problemas pode-se ter uma idéia bastante clara da natureza deste tipo de rede.

Supõe-se, então, um exemplo que considera a existência de um conjunto de k observações responsáveis pela caracterização de três grupos ou classes distintas, de acordo com a figura 4.7 abaixo. Uma rede neural *perceptron* de múltiplas camadas é capaz de diferenciar tais grupos fazendo uso de hiperplanos [Bish97], como indicado na figura 4.7(a). Uma maneira alternativa de separar tais grupos é modelar a densidade de probabilidade de cada classe por funções locais centrais, como mostra-se na figura 4.7(b). Esta última representação está relacionada com as redes neurais RBF.



(a) Separação por hiperplanos (b) Separação por funções centrais locais

Figura 4.7 – Exemplo de um conjunto de observações bi-dimensionais que caracterizam três classes distintas

Nas redes RBF, as funções centrais locais são definidas por funções base radiais que representam o espaço original descrito por todos os dados de entrada. Esta representação é local com relação ao espaço original pois, para um dado vetor de entrada, tipicamente somente uns poucos neurônios na camada escondida irão atingir valores significativos de ativação. Na realidade, as ativações das funções base podem ser interpretadas como probabilidades a posteriori definidas pela presença de determinadas características que descrevem o espaço original. Similarmente, os pesos da segunda camada da rede RBF podem

ser interpretados como probabilidades a posteriori definidas por determinadas características que descrevem se tal observação de entrada pertence a uma determinada classe ou grupo. A formulação matemática desta teoria pode ser encontrada com detalhes em [Bish97].

Uma característica de classificação importante das redes RBF é que as regiões do espaço original de entrada que descrevem cada classe ou grupo podem ser representadas de forma arbitrária ou distinta. Em inúmeras situações, uma única função base radial não é capaz de representar corretamente todas as observações pertencentes a uma mesma classe. Nestas ocasiões, a rede RBF pode utilizar mais de uma função base radial para representar tais observações disjuntas, associando-as à mesma classe, como ilustra o exemplo da figura 4.8 abaixo. As regiões de mesmo fundo pertencem à mesma classe e poderiam ser definidas por funções base distintas.

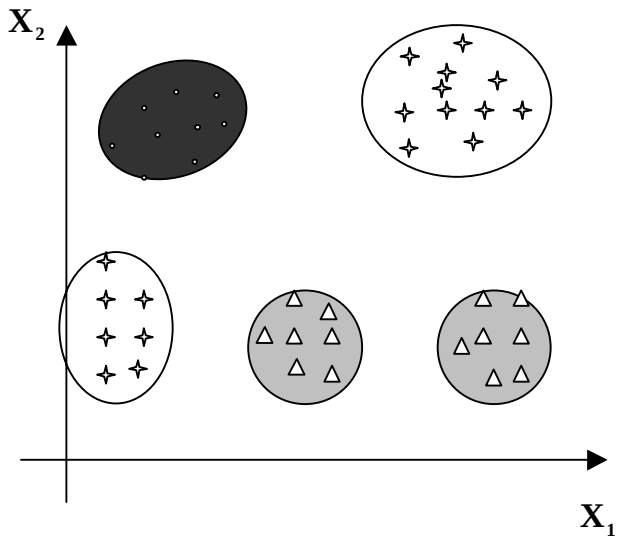


Figura 4.8 – Exemplo de classificação de uma rede neural RBF

4.3 Classificação Multivariada de Máxima Probabilidade

Nos modelos de classificadores apresentados até aqui não se tem como considerar a probabilidade de ocorrência de cada grupo na população de observações. Nestes modelos também não se pode tratar, explicitamente, os custos decorrentes de erros de classificação, isto é, de se atribuir uma determinada observação a um grupo j, por exemplo, quando a mesma pertence a um grupo i.

Esta seção apresenta uma descrição genérica destes conceitos, descrevendo com mais detalhes regras de classificação para populações com comportamento gaussiano que levam em consideração estes aspectos. No final da seção, uma nova abordagem para essas regras de classificação, desenvolvida neste trabalho, é apresentada juntamente com o conceito de matrizes de covariância "misturadas".

4.3.1 Descrição

Para descrever o método de classificação multivariada de máxima probabilidade, comumente chamado de método do mínimo custo esperado para um erro de classificação (*minimum expected cost of misclassification method)* [John92], considere-se $f_i(x)$ como sendo a função densidade de probabilidade dos padrões do grupo π_i, onde $i = 1,2,...,g$.

Assume-se que p_i é a probabilidade a priori de uma observação pertencer à população π_i e $c(j \mid i)$ é o custo de se atribuir uma observação a π_j, onde $j = 1,2,...,g$, quando a mesma pertence a π_i. Se $j = i$ então $c(i \mid i) = 0$.

Seja K_j a região onde as observações $\mathbf{x}$ são classificadas como pertencentes ao grupo π_j e

$$P(j \mid i) = P(\mathbf{x} \in \pi_j \mid \pi_i) = \int_{K_j} f_i(\mathbf{x})dx$$

a probabilidade condicional de se classificar erroneamente uma observação $\mathbf{x}$ do grupo i como pertencente ao grupo j. Pode-se, portanto, definir o custo condicional do erro de classificação (*conditional error cost* - CEC) de uma observação $\mathbf{x}$ do grupo π_1 como

$$\text{CEC}(1) = \sum_{j=2}^{g} P(j \mid 1)c(j \mid 1).$$

Este custo condicional ocorre com probabilidade p_1, que é a probabilidade a priori de uma observação pertencer ao grupo π_1. De forma similar, pode-se obter o custo condicional de erro de classificação para os demais grupos, ou seja, pode-se calcular CEC(2), CEC(3), ..., e CEC(g). Multiplicando-se cada CEC(i) pela respectiva probabilidade a priori e somando-se todos esses valores, obtém-se o custo médio por erro de classificação considerando-se todas os grupos:

$$\text{CEC} = \sum_{i=1}^{g} p_i \sum_{\substack{j=1 \\ j \neq i}}^{g} P(j \mid i)c(j \mid i). \qquad (4.3.1.1)$$

Prova-se [Ande84] que as regiões K_1, K_2, ..., K_g, para as quais o custo médio por erro de classificação dado pela fórmula (4.3.1.1) é mínimo, podem ser definidas atribuindo o padrão $\mathbf{x}$ ao grupo π_j, para a qual o valor

$$\sum_{\substack{i=1 \\ i \neq j}}^{g} p_i f_i(\mathrm{x}) c(j \mid i) \tag{4.3.1.2}$$

é mínimo. Se existirem resultados coincidentes atribui-se $\mathbf{x}$ para qualquer dos grupos empatados.

Quando a natureza do problema não permite distinguir o custo do erro de classificação entre os diversos grupos, pode-se admitir, sem perda de generalização, que todos os custos de erros de classificação são iguais a 1. Nesse caso, utilizando o argumento da fórmula (4.3.1.2), pode-se atribuir a observação $\mathbf{x}$ à população π_j se o valor

$$\sum_{\substack{i=1 \\ i \neq j}}^{g} p_i f_i(\mathbf{x}) \tag{4.3.1.3}$$

for mínimo. Nesta circunstância, a fórmula (4.3.1.3) conterá o valor mínimo quando o termo omitido, $p_j f_j(\mathbf{x})$, for justamente o termo de maior valor. Por conseguinte, quando assume-se que todos os custos de erros de classificação são iguais, pode-se determinar a que grupo uma dada observação pertence através da seguinte regra. Atribui-se $\mathbf{x}$ à população ou grupo π_j se

$$p_j f_j(\mathbf{x}) > p_i f_i(\mathbf{x}), \qquad \text{para todo } i \neq j. \qquad (4.3.1.4)$$

A regra de máxima probabilidade envolve três componentes: as probabilidades a priori p_i, os custos de erro de classificação $c(j|i)$, e as funções de densidade $f_i(\mathbf{x})$ para cada grupo existente. Como consequência, tal regra somente pode ser aplicada quando essas componentes podem ser especificadas ou estimadas.

4.3.2 Classificador Gaussiano

Em muitas aplicações é razoável admitir que os grupos ou classes de observações têm um comportamento governado por uma função de densidade de probabilidade gaussiana. Nestas situações, as funções de densidade de cada grupo $f_i(\mathbf{x})$ podem ser escritas como

$$f_i(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}_i|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right], \qquad i = 1,2,\ldots,g \qquad (4.3.2.1)$$

onde $\mathbf{x}$ é uma observação no espaço n-dimensional, e $\boldsymbol{\mu}_i$ e $\boldsymbol{\Sigma}_i$ são, respectivamente, o vetor média e a matriz de covariância da população i.

Admitindo que os custos de erro de classificação sejam iguais, pode-se escrever a regra (4.3.1.4) examinada na seção anterior como

$$\ln p_j f_j(\mathbf{x}) > \ln p_i f_i(\mathbf{x}), \qquad \text{para todo } i \neq j. \qquad (4.3.2.2)$$

Utilizando a fórmula (4.3.2.1), tem-se que

$$\ln p_j f_j(\mathbf{x}) \quad = \ln p_j - \left(\frac{n}{2}\right)\ln(2\pi) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_k| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1}(\mathbf{x}-\boldsymbol{\mu}_j). \qquad (4.3.2.3)$$

Como o termo $\left(\frac{n}{2}\right)\ln(2\pi)$ é constante para todas as populações, pode-se ignorá-lo para a discriminação das observações. Pode-se, portanto, reescrever a regra (4.3.2.2) para o classificador gaussiano de máxima probabilidade da seguinte forma. Define-se, primeiramente, o valor quadrático de discriminação da i-ésima população como

$$\mathrm{d}_i(\mathbf{x}) = -\frac{1}{2}\ln|\boldsymbol{\Sigma}_i| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i) + \ln p_i, \qquad i = 1,2,...,g \qquad (4.3.2.4)$$

Este valor quadrático $\mathrm{d}_i(\mathbf{x})$ é composto pelo determinante da matriz de covariância da população i ($|\boldsymbol{\Sigma}_i|$), pela probabilidade a priori da observação $\mathbf{x}$ pertencer à população i (p_i), e pelo quadrado da distância entre a observação $\mathbf{x}$ e o vetor média $\boldsymbol{\mu}_i$. Atribui-se, então, uma dada observação $\mathbf{x}$ à população π_j se $\mathrm{d}_j(\mathbf{x})$ for o máximo dos valores quadráticos definidos por (4.3.2.4).

Na prática, tanto os vetores média $\boldsymbol{\mu}_i$ quanto as matrizes de covariância $\boldsymbol{\Sigma}_i$ são desconhecidos. Para estimar estes termos, faz-se uso de um conjunto de treinamento com classificações pré-conhecidas e contendo k_i amostras para cada grupo i. O vetor média $\boldsymbol{\mu}_i$ de cada grupo passa a ser representado pelo vetor média $\bar{\mathbf{x}}_\mathbf{i}$ das respectivas amostras. As matrizes de covariância $\boldsymbol{\Sigma}_i$ podem ser estimadas de algumas maneiras, como se segue.

4.3.2.1 Matrizes de Covariância Distintas

Utilizando $\bar{\mathbf{x}}_\mathbf{i}$ e $\mathbf{S}_\mathbf{i}$ respectivamente como estimadores para $\boldsymbol{\mu}_i$ e $\boldsymbol{\Sigma}_\mathbf{i}$, pode-se escrever o valor quadrático de discriminação da i-ésima população como

$$\mathrm{d}_i(\mathbf{x}) = -\frac{1}{2}\ln|\mathbf{S}_\mathbf{i}| - \frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}}_\mathbf{i})^T\mathbf{S}_\mathbf{i}^{-1}(\mathbf{x}-\bar{\mathbf{x}}_\mathbf{i}) + \ln p_i, \qquad i = 1,2,\ldots,g. \qquad (4.3.2.1.1)$$

Pode-se entender a idéia deste classificador através do exemplo da figura 4.9. Este exemplo considera observações em um espaço bi-dimensional para três grupos distintos. Observa-se neste exemplo que as classes de observações se distribuem diferentemente para cada grupo.

O método descrito apresenta, contudo, dificuldades quando o número de variáveis n do modelo é maior do que o número de amostras k_i do conjunto de treinamento de cada grupo. Situações como estas são comuns na prática. Nestes casos, as matrizes de covariância de cada grupo têm um número de colunas (ou linhas) linearmente independentes – posto da matriz - igual ao tamanho do conjunto de treinamento de cada grupo. Portanto, estas matrizes são não inversíveis e invalidam o emprego da fórmula de discriminação (4.3.2.1.1).

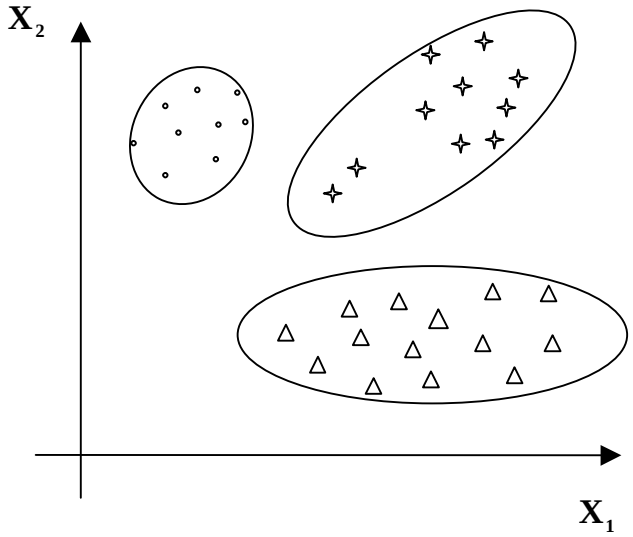


Figura 4.9 – Exemplo de classificação de máxima probabilidade para matrizes de covariância distintas

Além desta limitação algébrica, há ainda o fato que a escolha do conjunto de treinamento influencia significantemente a determinação das matrizes de covariância de cada classe, o que torna esta metodologia muito sensível à escolha desse conjunto.

4.3.2.2 Matrizes de Covariância Iguais

Uma solução frequentemente adotada para o problema de inversibilidade examinado na seção anterior, consiste em admitir que as matrizes de covariância $\mathbf{\Sigma_i}$ sejam iguais para todas as populações, e em utilizar a matriz $\mathbf{S_{pooled}}$ como estimador para as mesmas.

A matriz de covariância $\mathbf{S_{pooled}}$ corresponde à média de todas as matrizes de covariância $\mathbf{S_i}$ de cada grupo, ponderada pelo número de amostras k_i de cada classe. Pode-se calculá-la como

$$\mathbf{S}_{\mathbf{pooled}} = \frac{(k_1 - 1)\mathbf{S_1} + (k_2 - 1)\mathbf{S_2} + ... + (k_g - 1)\mathbf{S_g}}{k_1 + k_2 + ... + k_g - g} \qquad (4.3.2.2.1)$$

onde g é o número de grupos.

Esta aproximação permite escrever o valor quadrático de discriminação da i-ésima população, dado pela fórmula geral (4.3.2.4), como

$$\begin{aligned} \mathrm{d}_i(\mathbf{x}) &= -\frac{1}{2}\ln\left|\mathbf{S}_{\mathbf{pooled}}\right| - \frac{1}{2}(\mathbf{x} - \overline{\mathbf{x}}_{\mathbf{i}})^T \mathbf{S}_{\mathbf{pooled}}^{-1}(\mathbf{x} - \overline{\mathbf{x}}_{\mathbf{i}}) + \ln p_i, \qquad i = 1,2,...,g \\ &= -\frac{1}{2}\ln\left|\mathbf{S}_{\mathbf{pooled}}\right| - \frac{1}{2}\mathbf{x}^T \mathbf{S}_{\mathbf{pooled}}^{-1}\mathbf{x} + \overline{\mathbf{x}}_i^T \mathbf{S}_{\mathbf{pooled}}^{-1}\mathbf{x} - \frac{1}{2}\overline{\mathbf{x}}_i^T \mathbf{S}_{\mathbf{pooled}}^{-1}\overline{\mathbf{x}}_i + \ln p_i, \qquad (4.3.2.2.2) \\ &= \overline{\mathbf{x}}_i^T \mathbf{S}_{\mathbf{pooled}}^{-1}\mathbf{x} - \frac{1}{2}\overline{\mathbf{x}}_i^T \mathbf{S}_{\mathbf{pooled}}^{-1}\overline{\mathbf{x}}_i + \ln p_i \end{aligned}$$

Observa-se que como os dois primeiros termos $-\frac{1}{2}\mathbf{ln}\left|\mathbf{S}_{\mathbf{pooled}}\right| - \frac{1}{2}\mathbf{x}^T \mathbf{S}_{\mathbf{pooled}}^{-1}\mathbf{x}$ são iguais para todos $\mathrm{d}_1(\mathbf{x}), \mathrm{d}_2(\mathbf{x}), ..., \mathrm{d}_g(\mathbf{x})$, os mesmos são ignorados para a discriminação das observações.

A idéia básica deste classificador pode ser compreendida através de um exemplo mostrado na figura 4.10. Este exemplo é idêntico ao da figura 4.9 que considera observações em um espaço bi-dimensional para três grupos distintos. Observa-se que para este tipo de estimativa todas as classes são representadas pela mesma forma média.

Como no cálculo da matriz de covariância $\mathbf{S}_{\mathbf{pooled}}$ - fórmula (4.3.2.2.1) - as amostras de todos os grupos são levadas em consideração, o posto desta matriz é maior do que os das matrizes de covariância de cada grupo $\mathbf{S_i}$. $\mathbf{S}_{\mathbf{pooled}}$ poderá, portanto, ser uma matriz inversível, ainda que as matrizes $\mathbf{S_i}$ não o sejam. A análise multivariada de discriminantes, descrita na seção 3.2 do capítulo 3, é um caso particular da fórmula (4.3.2.2.2) em que se admite que as

matrizes de covariância dos grupos são todas iguais entre si e se utiliza a $\mathbf{S}_{\mathbf{pooled}}$ como estimador [John92].

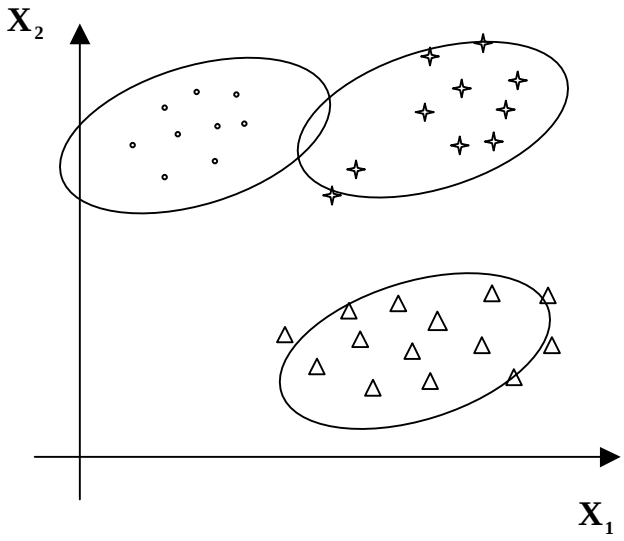


Figura 4.10 – Exemplo de classificação de máxima probabilidade para matrizes de covariância iguais

A este procedimento está implícita a hipótese de que as classes de observações se distribuem igualmente em torno de seus valores médios. O efeito desta simplificação sobre o desempenho do classificador pode ser significativo em muitas aplicações.

4.3.2.3 Matrizes de Covariância Misturadas

Este trabalho propõe um novo estimador para as matrizes de covariância $\mathbf{\Sigma}_i$ de cada classe de observações que não está sujeito às mesmas restrições que $\mathbf{S}_\mathbf{i}$ quanto à inversibilidade e, ao mesmo tempo, representa a distribuição particular de cada grupo.

Este estimador considera que a matriz de covariância de cada grupo $\mathbf{i}$ pode ser representada por uma matriz de covariância "misturada" $\mathbf{Smix}_\mathbf{i}$ de tal forma que

$$\mathbf{Smix_i} = a\mathbf{S_{pooled}} + (1-a)\mathbf{S_i}, \qquad \text{onde } 0 < a < 1. \qquad (4.3.2.3.1)$$

A matriz $\mathbf{Smix_i}$ é uma combinação linear da matriz de covariância $\mathbf{S_{pooled}}$, definida pela fórmula (4.3.2.2.1), e da matriz de covariância $\mathbf{S_i}$ do grupo i, onde $i = 1,2,...,g$ grupos. Este procedimento é similar ao procedimento conhecido como *ridge regression* adotado para resolver o problema de multicolinearidade de estimadores [Judg88].

O estimatidor $\mathbf{Smix_i}$ da matriz de covariância de cada grupo permite escrever o valor quadrático de discriminação da i-ésima população, dado pela fórmula geral (4.3.2.4), como

$$\mathrm{d}_i(\mathbf{x}) = -\frac{1}{2}\ln\left|\mathbf{Smix_i}\right| - \frac{1}{2}(\mathbf{x}-\overline{\mathbf{x}}_\mathbf{i})^{\mathrm{T}}\mathbf{Smix_i}^{-1}(\mathbf{x}-\overline{\mathbf{x}}_\mathrm{i}) + \ln p_i \qquad (4.3.2.3.2)$$

e a regra de classificação das observações $\mathbf{x}$ para matrizes de covariância "misturadas" é idêntica à regra examinada na seção 4.3.2, substituindo a fórmula (4.3.2.4) pela fórmula (4.3.2.3.2).

Chama-se este classificador de Classificador Gaussiano de Máxima Probabilidade para Matrizes de Covariâncias "Misturadas" e será designado neste trabalho daqui por diante como classificador MPM.

A idéia básica introduzida por este classificador pode ser visualizada na figura 4.11.

A figura 4.11 mostra o mesmo exemplo das figuras 4.9 e 4.10 que considera observações em um espaço bi-dimensional para três grupos distintos. As regiões pontilhadas reproduzem as formas definidas pelas matrizes de covariância $\mathbf{S_i}$ e pela matriz $\mathbf{S_{pooled}}$, de

acordo com as figuras 4.9 e 4.10 respectivamente. As regiões de linha contínua mostram as formas consideradas pelas matrizes de covariância $\mathbf{Smix_i}$.

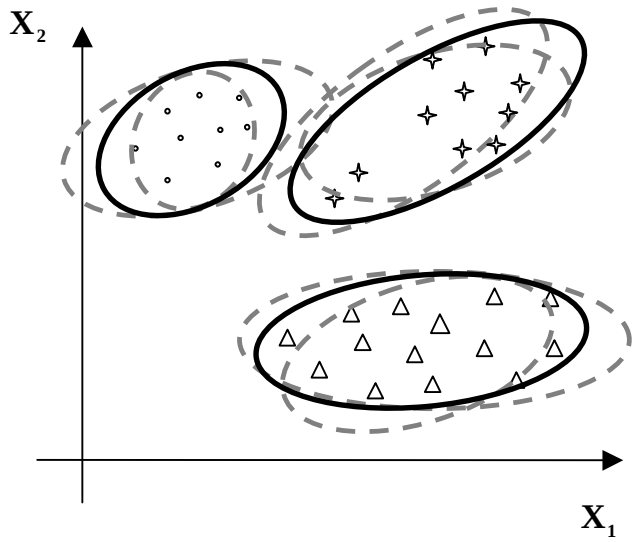


Figura 4.11 – Exemplo de classificação de máxima probabilidade para matrizes de covariância misturadas

O efeito da estimativa proposta na fórmula (4.3.2.3.1) pode ser interpretada da seguinte maneira. Considere que a variabilidade das classes de observações no espaço de características se divide em dois tipos: a variabilidade comum a todas as classes e a variabilidade particular de cada classe. A matriz $\mathbf{S_{pooled}}$ representa melhor a variabilidade comum do que as matrizes $\mathbf{S_i}$, uma vez que é calculada a partir de um conjunto maior de amostras em que este tipo de variabilidade está presente. A variabilidade particular de cada grupo, ao contrário, tende a se diluir em $\mathbf{S_{pooled}}$, já que o número de amostras de cada grupo é em geral muito menor do que o número total de observações utilizadas no cálculo de $\mathbf{S_{pooled}}$. É portanto de se esperar que $\mathbf{S_i}$ represente melhor a variabilidade particular de cada grupo do que $\mathbf{S_{pooled}}$.

Em comparação com $\mathbf{S_i}$ e $\mathbf{S_{pooled}}$, o estimador $\mathbf{Smix_i}$ proposto realça, por um lado, a variabilidade particular pobremente representada em $\mathbf{S_{pooled}}$ e, por outro, a variabilidade comum pobremente representada em $\mathbf{S_i}$, provendo, desta forma, uma melhor representação para as matrizes de covariância de cada grupo. Além disso, é um resultado provado [Magn95] que o determinante da soma de uma matriz positiva definida, no caso a $\mathbf{S_{pooled}}$, com uma matriz positiva semi-definida, no caso $\mathbf{S_i}$, é maior ou igual ao determinante da matriz positiva definida considerada ($\mathbf{S_{pooled}}$). Portanto, se a matriz $\mathbf{S_{pooled}}$ for inversível, a matriz $\mathbf{Smix_i}$ também o é.

Os experimentos descritos no capítulo 5 indicam que o desempenho deste novo classificador é superior aos demais classificadores considerados neste trabalho. Os experimentos mostram também que o desempenho do classificador MPM se altera pouco para o fator de regressão a (vide fórmula 4.3.2.3.1) que foi variado para uma ampla faixa de valores entre 0 e 1.