Classificação Linear

1. Motivação e aspectos básicos do problema

Exemplo: filtro de spam. 

"Mega-oferta: Vicodin 120mg por apenas US$ 19,99. Na compra de 2 lotes, ganhe um cupom …"

Spam

"Caro Prof.,

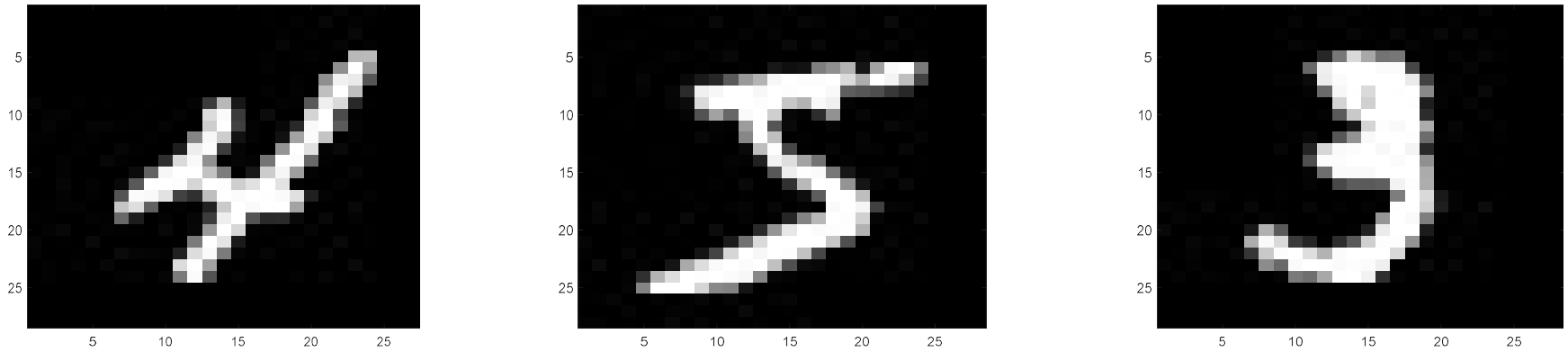
Eu li alguns de seus trabalhos recentes sobre modelos probabilísticos …"

Email pessoal

"Prezado doutor, a receita para o medicamento prescrito foi encaminhada à farmácia sugerida, mas lá há somente lotes de 5 mg, em vez de 12 mg …"

?????

Exemplo: reconhecimento de dígitos escritos à mão – Base de dados MNIST.



Exemplo: classificação de câncer de pele.

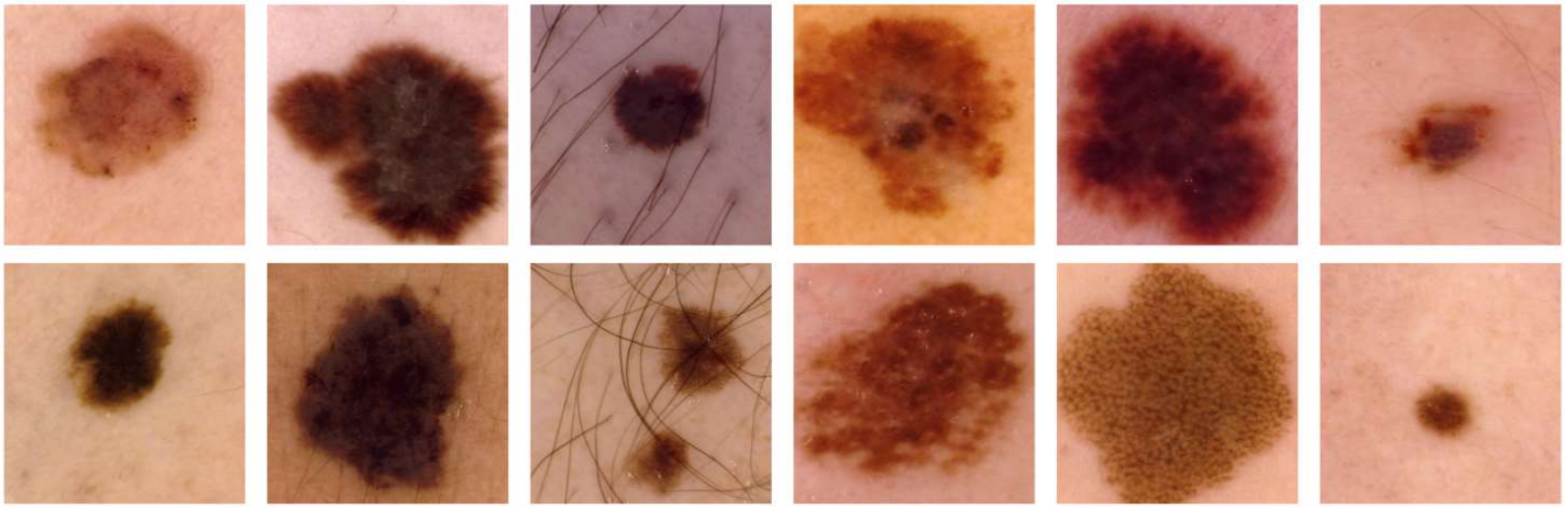


Figura extraída de (FORNACIALI, AVILA, CARVALHO & VALLE, 2014). Na primeira linha, temos exemplos de melanomas, enquanto na segunda linha são exibidas algumas lesões de pele benignas.

Exemplo: categorização de texto.

Desafio: atribuir a cada padrão de entrada o rótulo correspondente a uma das Q classes existentes, $C_k, k = 1, \ldots, Q$, à qual o dado pertence.

Este tipo de desafio caracteriza o problema conhecido como **classificação** (DUDA, HART & STORK, 2001).

No cenário mais comum do problema de classificação, as classes são consideradas como disjuntas, de modo que **cada padrão de entrada deve pertencer a uma – e somente uma – classe**.

- Quando esta hipótese não é válida, tem-se o cenário de classificação multi-rótulo (*multi-label*).

À semelhança do problema de regressão, existe um conjunto de treinamento $\{\mathbf{x}(i); \mathbf{y}(i)\}_{i=0}^{N-1}$ para ser utilizado na construção do **classificador**, em que $\mathbf{x}(i) \in \mathbb{R}^{K\times 1}$ representa o i-ésimo padrão de entrada, caracterizado por K atributos.

1.1. Representação da saída desejada

Em última análise, a saída desejada para o padrão $\mathbf{x}(i)$ deveria ser o rótulo da classe à qual $\mathbf{x}(i)$ pertence. Sendo assim, a saída de um classificador é uma variável categórica (discreta).

Contudo, para podermos realizar o treinamento do modelo, é preciso escolher uma representação numérica para a saída desejada. Neste contexto, algumas opções podem ser adotadas, dependendo do tipo de classificação a ser feita.

Para **classificação binária**:

- Há somente duas classes possíveis, C_1 e C_2. Assim, poderíamos utilizar uma única saída escalar binária para indicar a classe correspondente ao padrão de entrada:

$$y(i) = \begin{cases} 0, & \mathbf{x}(i) \in C_1 \\ 1, & \mathbf{x}(i) \in C_2 \end{cases}$$

- Também é possível adotar $y(i) = -1$ para $\mathbf{x}(i) \in C_1$.

Para o cenário **multi-classe**:

- Uma estratégia bastante utilizada é conhecida como o*ne-hot encoding*, que faz uma representação binária para as variáveis categóricas.
- Neste caso, o classificador produz múltiplas saídas, cada uma representando a possibilidade de o padrão pertencer a uma classe específica.

Exemplo: Quatro classes possíveis – esporte, política, ciências e variedades.

- Esporte – $[1\ \ 0\ \ 0\ \ 0]^T$
- Política – $[0\ \ 1\ \ 0\ \ 0]^T$
- Ciências – $[0\ \ 0\ \ 1\ \ 0]^T$
- Variedades – $[0\ \ 0\ \ 0\ \ 1]^T$

Assim, $\mathbf{y}(i) \in \mathbb{R}^{Q\times 1}$, de maneira que o classificador deve realizar um mapeamento $\mathbb{R}^K \to \mathbb{R}^Q$.

1.2. Funções discriminantes

Uma das maneiras mais usuais de se representar um classificador é por meio de um conjunto de **funções discriminantes** $g_i(\mathbf{x}), i = 1, \ldots, Q$. Cada função discriminante se

concentra em uma das classes existentes e fornece em sua saída um indicativo da possibilidade de o padrão pertencer àquela classe específica.

Classificação: $\mathbf{x}(i)$ é designado para a classe C_k se $g_k(\mathbf{x}) > g_j(\mathbf{x})$ para todo $j \neq k$.

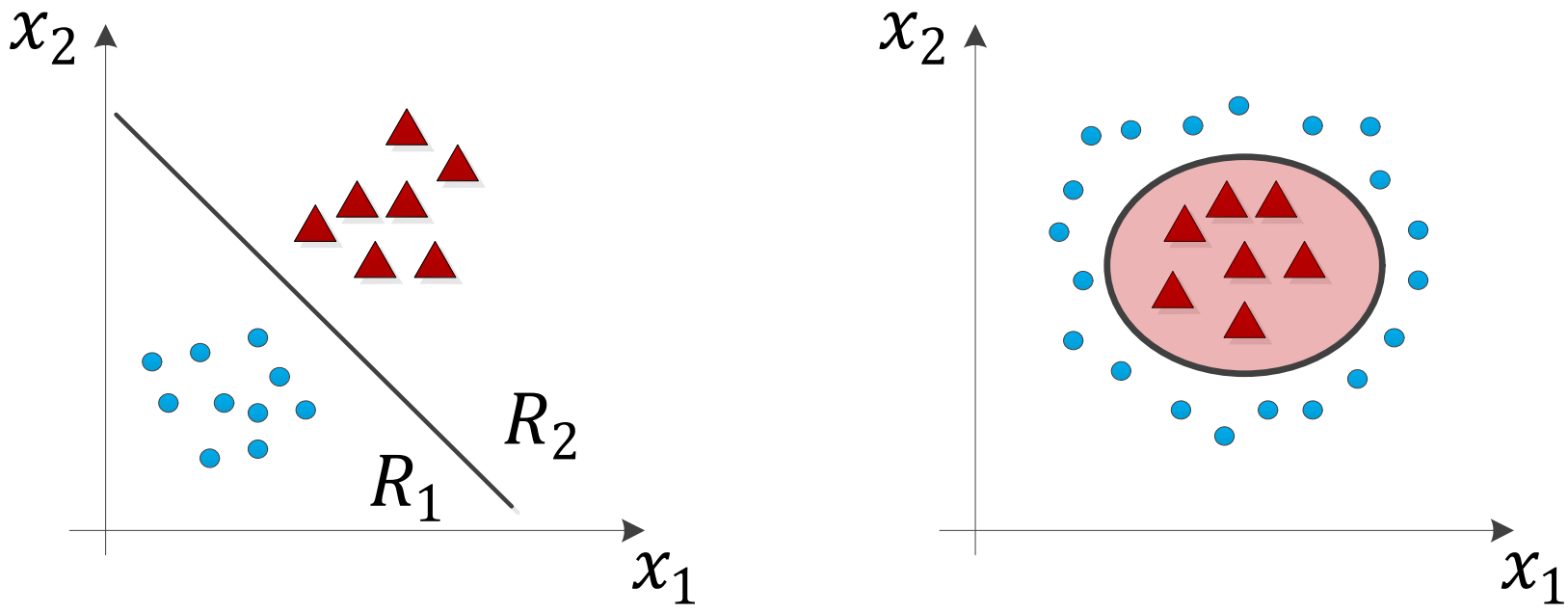
No caso de classificação binária, podemos definir um discriminante único:

$$g(\mathbf{x}) = g_1(\mathbf{x}) - g_2(\mathbf{x}), \tag{1}$$

tal que um padrão será designado à classe C_1 se $g(\mathbf{x}) > 0$, e a classe C_2 é escolhida caso $g(\mathbf{x}) < 0$.

Sendo assim, o espaço de entrada $\mathbb{R}^K$ é dividido em **regiões de decisão** $R_i, i = 1, \dots, Q$, sendo que $R_i = \left\{\mathbf{x} \in \mathbb{R}^K | i = \arg\max_{j=1,\dots,Q} g_j(\mathbf{x})\right\}$, as quais são delimitadas ou separadas pelas **fronteiras de decisão**, que correspondem a superfícies no espaço dos atributos onde ocorre uma indeterminação, ou, analogamente, um empate entre diferentes classes possíveis.

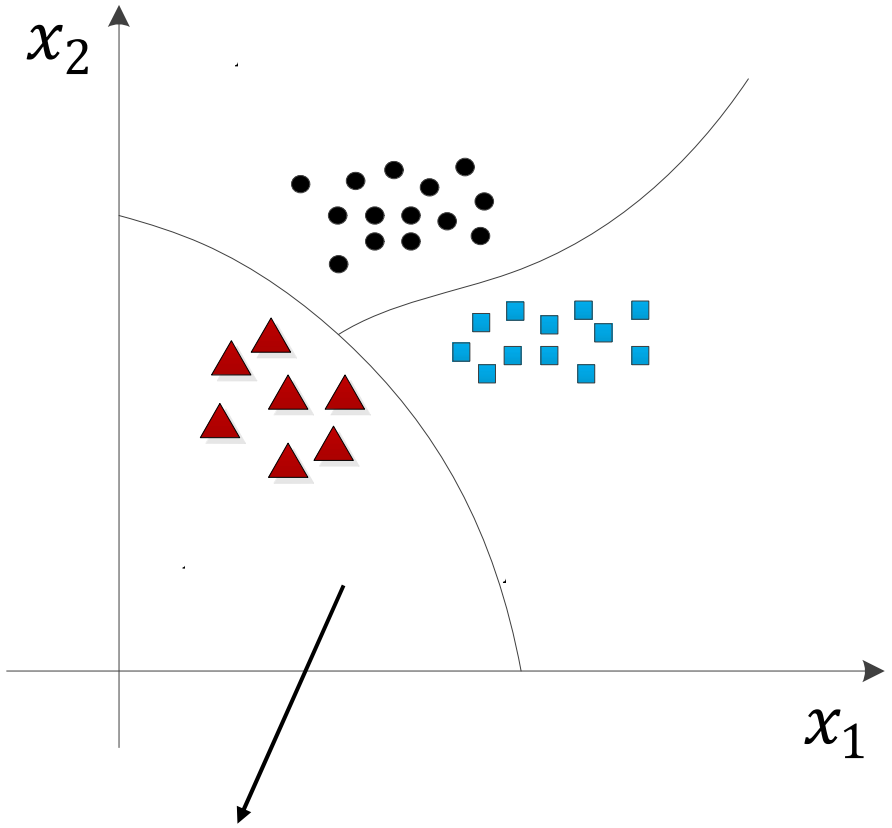
Exemplo: classificação binária.



(a) (b)

Figura. Ilustração das regiões de decisão em um problema de classificação binária. A fronteira de decisão é definida por $g(\mathbf{x}) = 0$, onde $g(\mathbf{x})$ é a função discriminante que descreve o classificador. Em (a), a fronteira é linear, enquanto em (b), ela é não-linear e a região indicada em vermelho corresponde a R_2.

Exemplo: classificação multi-classe.



Fronteira: $g_1(\mathbf{x}) = g_2(\mathbf{x})$ e $g_1(\mathbf{x}) > g_3(\mathbf{x})$

2. Discriminantes lineares

2.1. Classificação binária

Modelo: $g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$

O padrão $\mathbf{x}$ é designado para a classe C_1 se $g(\mathbf{x}) < 0$, e para a classe C_2 se $g(\mathbf{x}) > 0$.

Propriedades:

- Considere dois pontos $\mathbf{x}_a$ e $\mathbf{x}_b$ posicionados sobre a fronteira de decisão. Como $g(\mathbf{x}_a) = g(\mathbf{x}_b) = 0$, então $\mathbf{w}^T(\mathbf{x}_a - \mathbf{x}_b) = 0$, o que significa que o vetor $\mathbf{w}$ é ortogonal a todo vetor que reside no hiperplano da fronteira de decisão; logo, o vetor $\mathbf{w}$ define a orientação da fronteira de decisão.
- Se $\mathbf{x}$ é um ponto pertencente à fronteira de decisão, então $g(\mathbf{x}) = 0$. Assim, a distância normal da origem à fronteira de decisão é dada por:

$$d = \text{projeção}(\mathbf{x}, \mathbf{w}) = \frac{\mathbf{w}^T\mathbf{x}}{\|\mathbf{w}\|} = -\frac{w_0}{\|\mathbf{w}\|}$$

Então, w_0 determina a localização da fronteira de decisão.

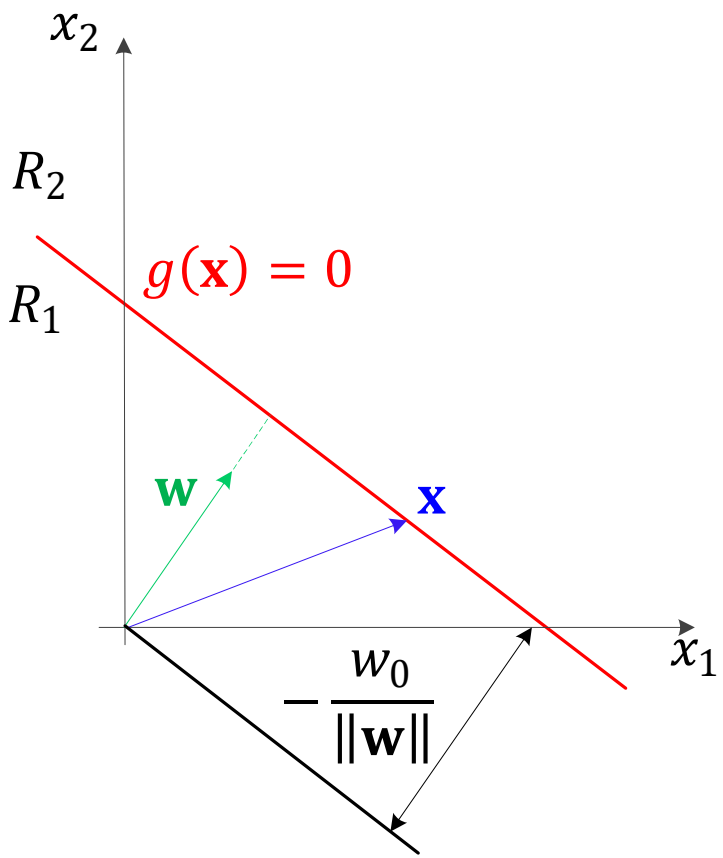


Figura. Ilustração da geometria de um discriminante linear em duas dimensões. A reta indicada em vermelho representa a fronteira de decisão, e é perpendicular a **w**. Sua distância à origem é controlada pelo parâmetro w_0.

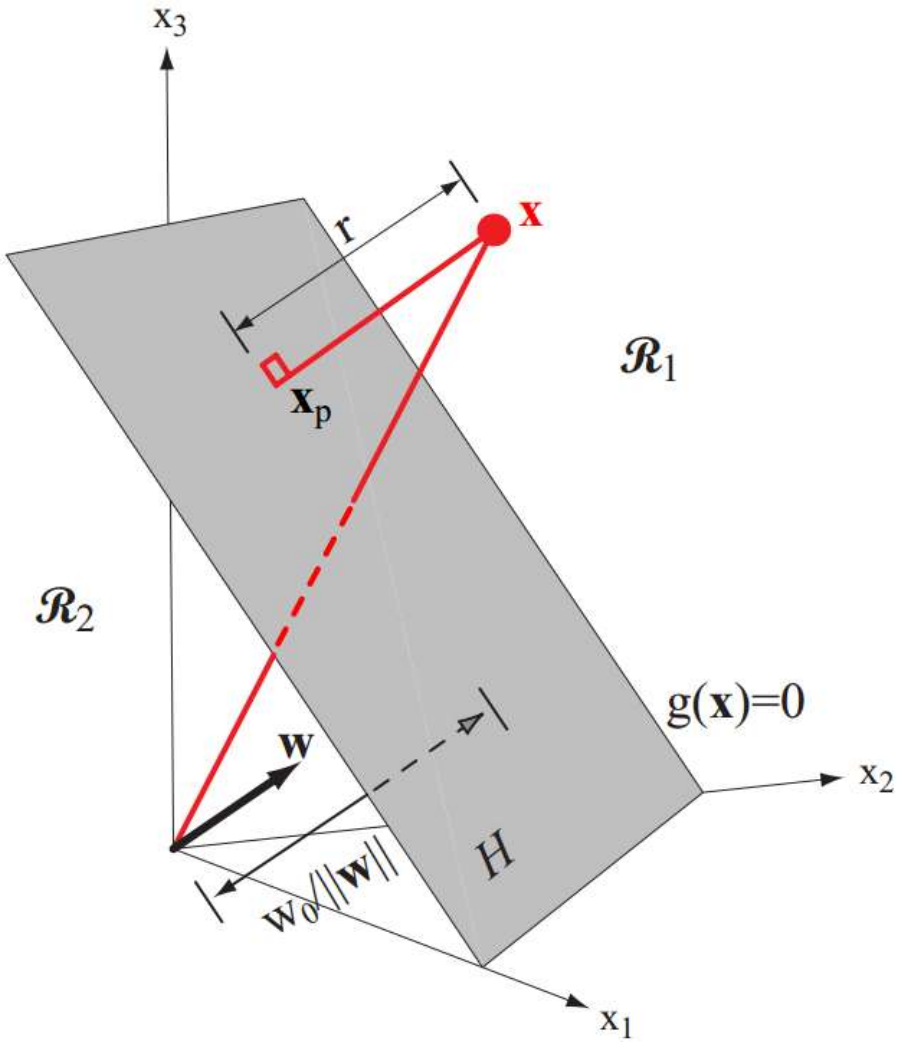


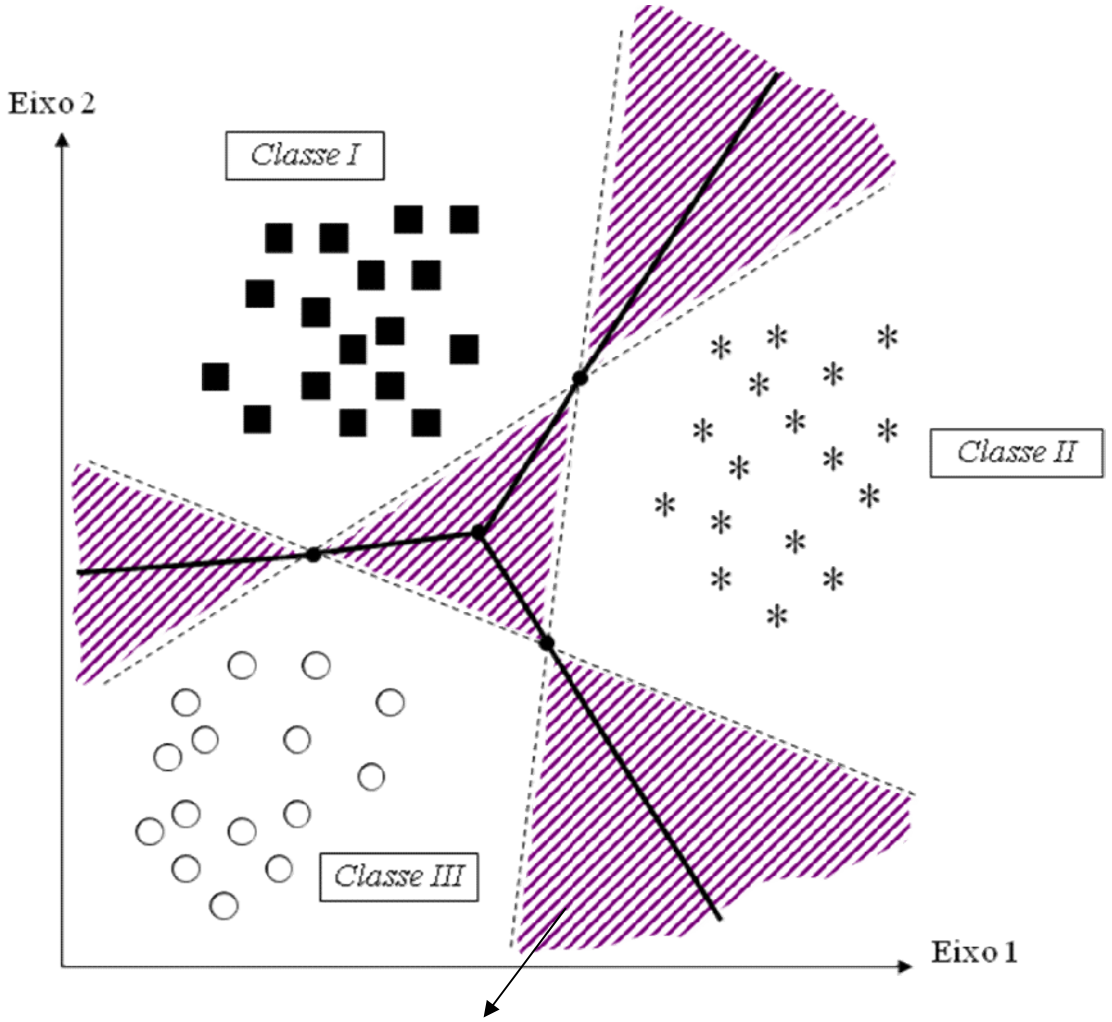
Figura extraída de (DUDA, HART & STORK, 2001). Ilustração da fronteira de decisão e das propriedades geométricas associadas a um discriminante linear no caso de três dimensões. A distância de um ponto **x** arbitrário à fronteira é dada por $r = \frac{g(x)}{\|\mathbf{w}\|}$.

2.2. Abordagens para classificação multi-classe

2.2.1. Um-contra-todos

Nesta estratégia, Q classificadores binários são construídos. A i-ésima função discriminante deve indicar uma saída positiva caso o padrão pertença à classe C_i, e um valor negativo caso o padrão pertença a qualquer outra classe.

Nas regiões hachuradas, ocorrem ambiguidades: mais do que uma classe é indicada pelo conjunto de discriminantes, ou, então, nenhuma.

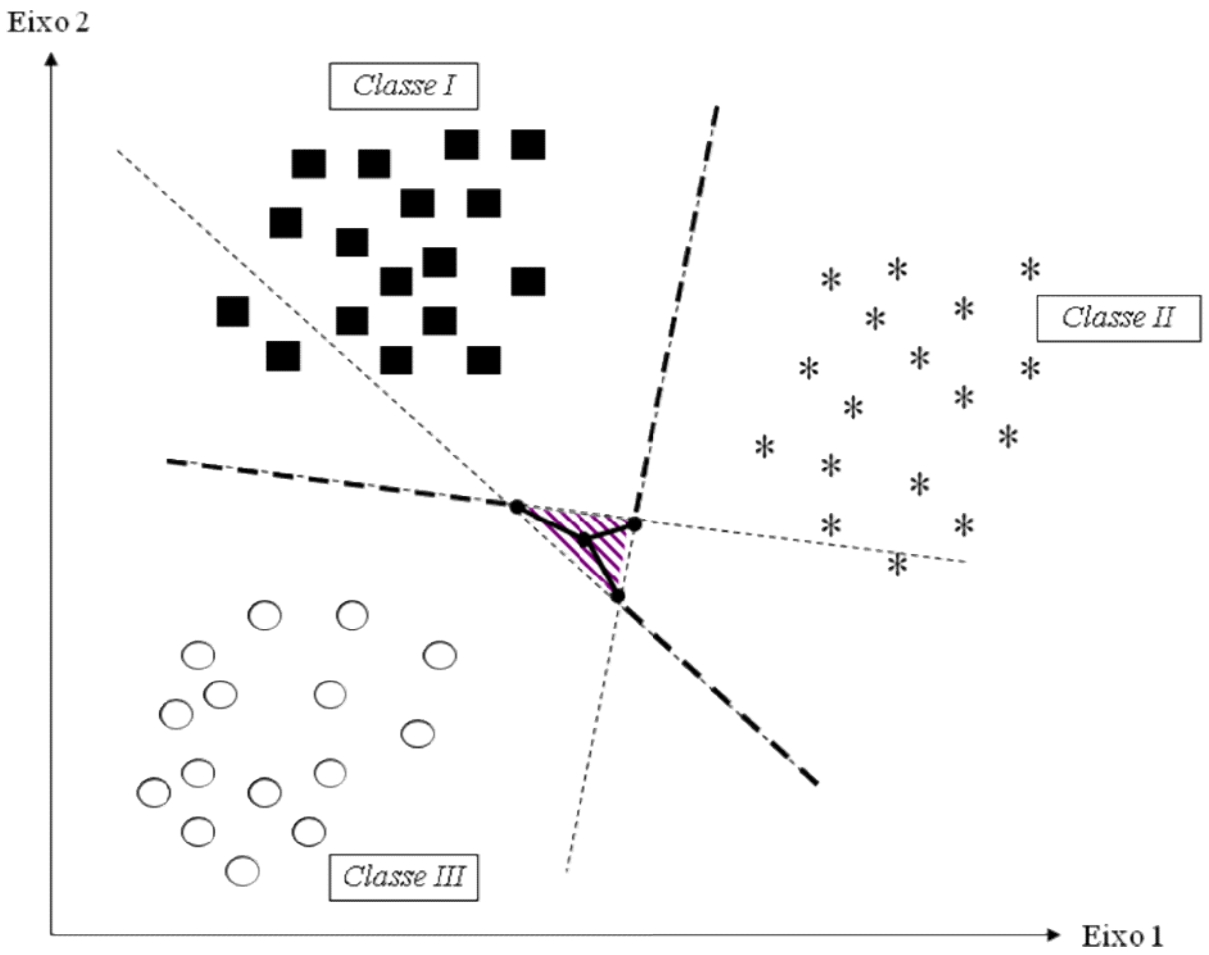


A ambiguidade pode ser resolvida atribuindo cada ponto no espaço à classe associada ao discriminante com máximo valor.

2.2.2. Um-contra-um

Nesta estratégia, treinamos $Q(Q-1)/2$ classificadores binários. Cada classificador é construído para fazer a distinção entre padrões pertencentes a duas classes (e.g., C_1 e C_2). No final, cada padrão é classificado conforme o voto majoritário entre as funções discriminantes.

Também pode gerar ambiguidades em algumas regiões do espaço.



2.2.3. Extensão: múltiplos discriminantes

Outra maneira de abordarmos o problema multi-classe consiste em projetar um classificador dotado de Q funções discriminantes, ou, em outras palavras, um classificador que produza Q saídas para uma entrada $\mathbf{x}$, uma para cada classe. Cada padrão é, então, atribuído à classe cuja saída correspondente assumir o maior valor, i.e., $\mathbf{x} \in C_i$ se $g_i(\mathbf{x}) > g_k(\mathbf{x}), \forall k \neq i$.

Observação: a diferença fundamental desta abordagem para os mecanismos de desambiguação que podem ser aplicados nos casos um-contra-todos e um-contra-um é que as funções discriminantes $g_i(\mathbf{x}), i = 1, \ldots, Q$ são adaptadas de **forma conjunta**, tendo em vista a minimização de um funcional de erro em relação à saída desejada para o classificador, representada de acordo com o *one-hot encoding*.

3. Quadrados mínimos para classificação linear

O arcabouço teórico de regressão linear, discutido no tópico anterior, pode ser aplicado ao ajuste dos parâmetros dos discriminantes lineares envolvidos na tarefa de classificação.

Vetor de saída desejado: $\mathbf{y}(i) \in \mathbb{R}^{Q\times 1}$ – *one-hot encoding*.

Vetor de saída do classificador: $\mathbf{g}(\mathbf{x}(i)) = \left[g_1(\mathbf{x}(i)) \ \ldots \ g_Q(\mathbf{x}(i))\right]^T$, onde cada função discriminante é dada por:

$$g_k(\mathbf{x}(i)) = \mathbf{w}_k{}^T\mathbf{x}(i) + w_0{}^{(k)}, k = 1, \ldots, Q, \tag{2}$$

com $\mathbf{w}_k = \left[w_1{}^{(k)} \ \ldots \ w_K{}^{(k)}\right]^T$.

Solução de quadrados mínimos:

$$\mathbf{W} = (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathbf{Y}, \tag{3}$$

onde

$$\mathbf{W} = \begin{bmatrix} w_0^{(1)} & \cdots & w_0^{(Q)} \\ \vdots & \ddots & \vdots \\ w_K^{(1)} & \cdots & w_K^{(Q)} \end{bmatrix}, \tag{4}$$

$\mathbf{Y} \in \mathbb{R}^{N \times Q}$ é a matriz com as saídas desejadas e $\mathbf{\Phi} \in \mathbb{R}^{N \times (K+1)}$,

$$\mathbf{\Phi} = \begin{bmatrix} \boldsymbol{\phi}(\mathbf{x}(0))^T \\ \vdots \\ \boldsymbol{\phi}(\mathbf{x}(N-1))^T \end{bmatrix},$$

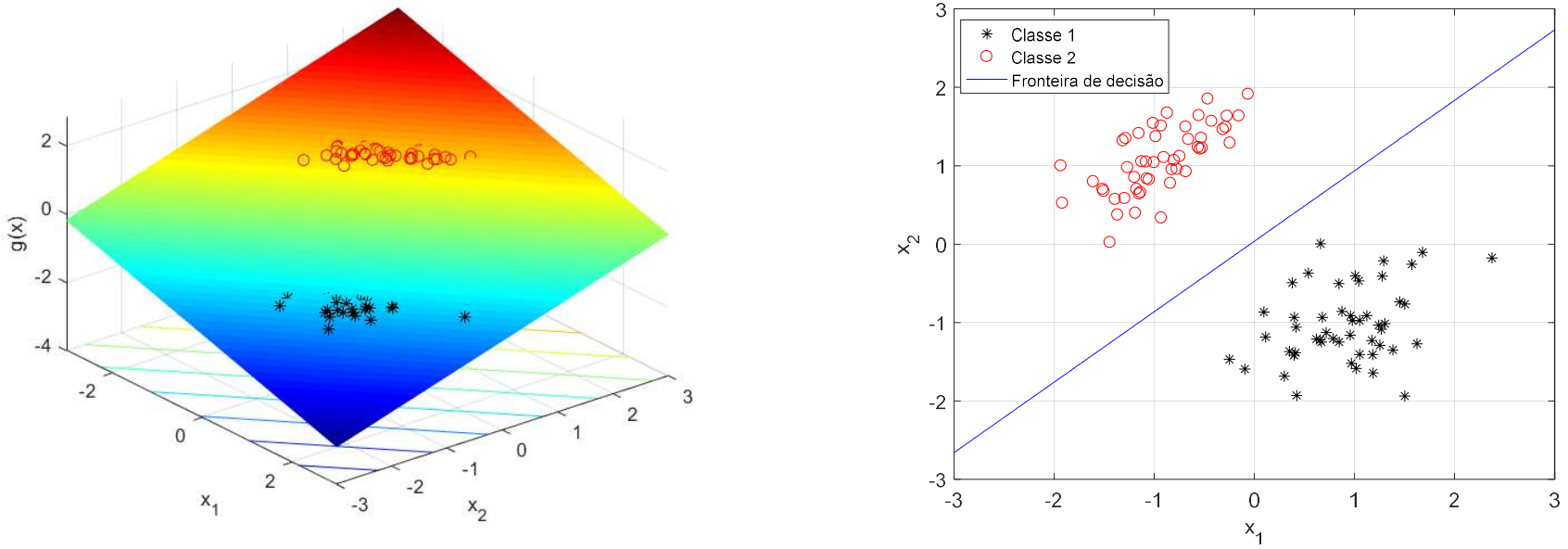
em que $\boldsymbol{\phi}(\mathbf{x}(i)) = [1 \;\; x_1(i) \;\; \cdots \;\; x_K(i)]^T$.

Problemas:

- As saídas produzidas pelo modelo não estão restritas ao intervalo especificado (*e.g.*, [0,1] ou [-1,1]). Então, é possível observarmos valores de saída maiores que o valor máximo (*e.g.*, 1), ou menores que o mínimo (*e.g.*, -1). Como queremos minimizar o erro quadrático médio, os pontos que produzirem saídas muito positivas ou muito negativas terão maior impacto sobre o valor do erro. Por causa disso, esta abordagem de classificação é bastante sensível à presença de *outliers*.

- Além disso, o método de quadrados mínimos tem por trás a hipótese de que a distribuição condicional $p(\mathbf{y}|\mathbf{x})$ é gaussiana; porém, no caso no problema de classificação, dada a natureza discreta da variável de saída, tal hipótese está longe de ser verdadeira.

Exemplo: classificação binária.



(a) (b)

Figura. Em (a), vemos a superfície de decisão associada a um discriminante linear obtido via quadrados mínimos. Em (b), observamos que a fronteira de decisão corretamente separa os padrões das classes C_1 e C_2.

Se, porém, acrescentarmos alguns padrões à classe C_1 bem afastados, a fronteira de decisão acaba sendo afetada devido ao erro na aproximação da saída referente a estes padrões. O critério de quadrados mínimos acaba penalizando padrões de entrada cuja saída é "muito correta".

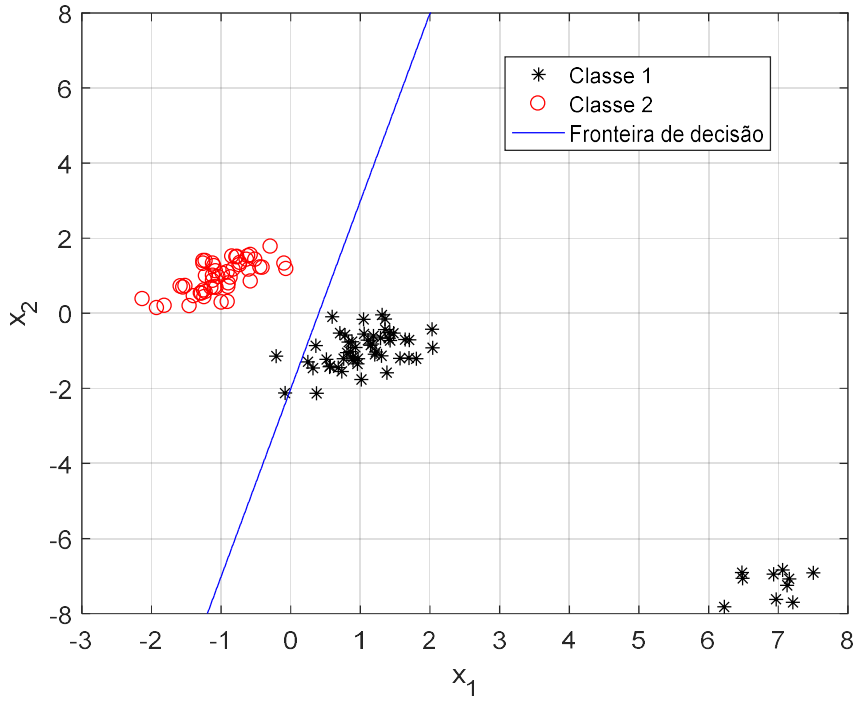


Figura. Fronteira de decisão do discriminante linear obtido via quadrados mínimos. A presença de *outliers* afeta significativamente os parâmetros, pois o critério de erro quadrático acaba se concentrando em reduzir o erro justamente nos pontos mais afastados, embora a classificação em si já estivesse correta.

Exemplo: cenário multi-classe ($Q = 3$).

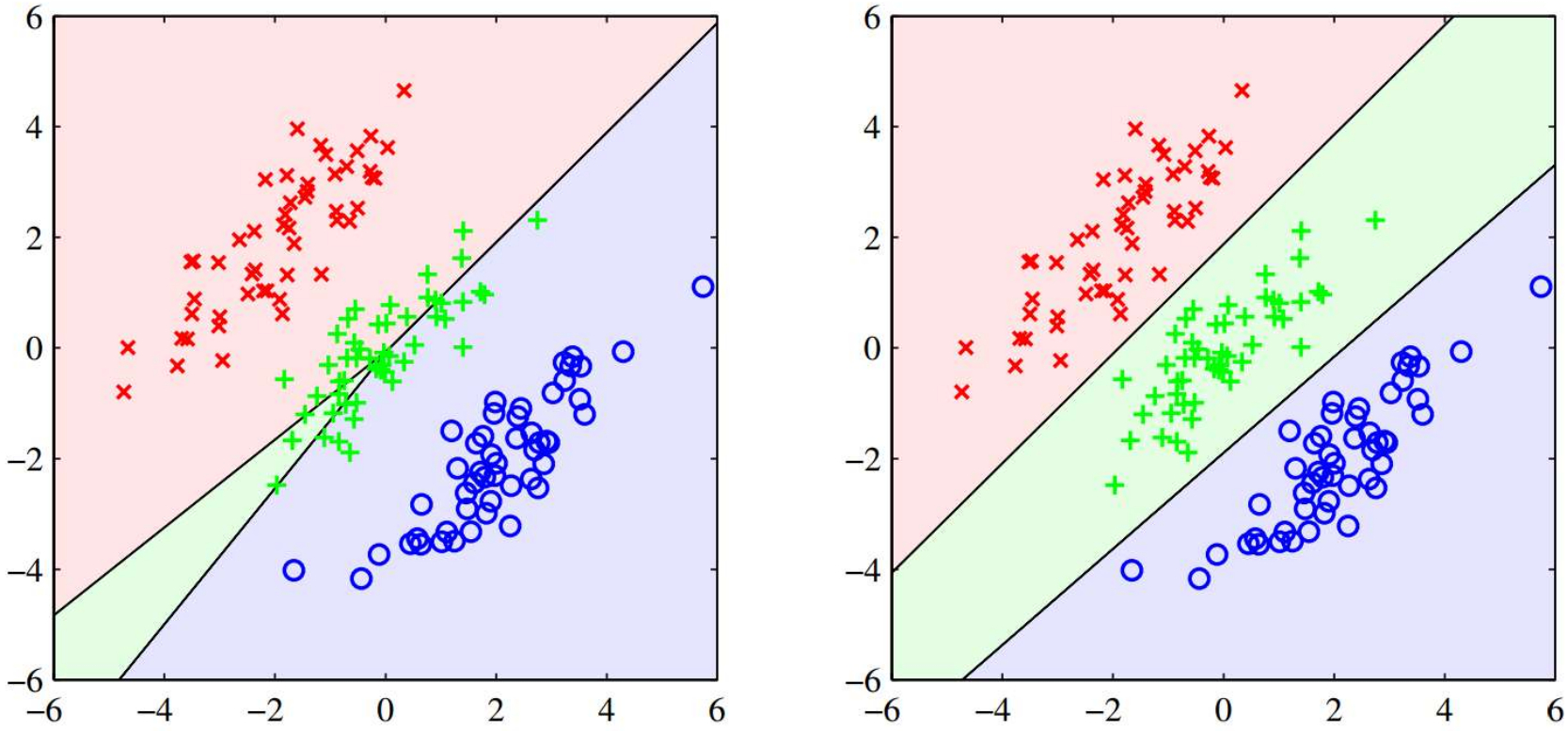


Figura extraída de (BISHOP, 2006). No lado esquerdo, vemos a fronteira de decisão associada a discriminantes lineares ajustados por quadrados mínimos. A região de decisão referente à classe indicada em verde é bastante reduzida, de modo que vários padrões são classificados de forma incorreta. No lado direito, temos outra solução baseada em discriminantes lineares, mas com desempenho bem superior.

4. Teoria bayesiana de decisão

A teoria bayesiana de decisão é uma importante abordagem estatística para o problema de classificação, a qual explora o conhecimento de probabilidades ligadas às classes e aos atributos dos dados, bem como dos custos associados a cada decisão, para realizar a classificação de cada nova amostra.

Definições iniciais:

Considere que uma amostra ou padrão a ser classificado seja descrito por um vetor $\mathbf{x} \in \mathbb{R}^{K\times 1}$ de K atributos. Cada padrão pertence a uma, e somente uma, classe C_i, sendo que existem ao todo Q classes possíveis.

- $P(C_i)$ denota a probabilidade *a priori* associada à classe C_i; em outras palavras, $P(C_i)$ indica a probabilidade de um padrão arbitrário (e desconhecido) pertencer à classe C_i.

Suponha, então, que um padrão **x** seja observado. Diante do conhecimento das características deste padrão, qual deve ser a decisão quanto à classe a que ele pertence?

Um critério simples, mas claramente ineficiente, seria atribuir o padrão **x** à classe mais provável, *i.e.*, à classe cuja probabilidade *a priori* seja máxima. Entretanto, tal decisão ignora por completo as especificidades do padrão **x** sendo avaliado.

Uma opção também intuitiva e, ao mesmo tempo, mais poderosa seria escolher a classe que se mostre a mais provável tendo em vista os atributos específicos do padrão **x**. Ou seja, a decisão é tomada em favor da classe cuja probabilidade *a posteriori* (*i.e.*, já levando em consideração o conhecimento do vetor de atributos **x**) seja máxima. A probabilidade *a posteriori* corresponde à probabilidade condicional $P(C_i|\mathbf{x})$.

Teorema de Bayes:

$$P(C_i|\mathbf{x}) = \frac{P(\mathbf{x}|C_i)P(C_i)}{P(\mathbf{x})}$$

- O termo $P(\mathbf{x}|C_i)$ é denominado verossimilhança (*likelihood*).
- O termo $P(\mathbf{x})$ é usualmente chamado de evidência.

4.1. Máxima probabilidade a posteriori (MAP)

O critério intuitivo sugerido anteriormente é conhecido como o critério da máxima probabilidade *a posteriori* (MAP, do inglês *maximum a posteriori probability*), cuja decisão para o padrão $\mathbf{x}$ é, portanto, pela classe C_k que maximiza $P(C_i|\mathbf{x})$:

$$\text{MAP:} \quad C_k = \arg \max_{C_i, i=1,\ldots,Q} P(C_i|\mathbf{x}) \tag{5}$$

- Supondo um problema de classificação binária ($Q = 2$), a regra de decisão MAP pode ser escrita como:

Decida pela classe C_1 se $P(C_1\|\mathbf{x}) > P(C_2\|\mathbf{x})$; caso contrário, decida por C_2.

- Note que, com base no teorema de Bayes, a solução para (5) é absolutamente equivalente àquela que maximiza o numerador ($P(\mathbf{x}|C_i)P(C_i)$), de modo que:

$$\text{MAP:} \quad C_k = \arg \max_{C_i, i=1,\dots,Q} P(\mathbf{x}|C_i)P(C_i), \tag{6}$$

já que $P(\mathbf{x})$ (denominador) não depende das classes testadas, servindo apenas como fator de escala no critério.

4.2. Máxima verossimilhança (ML)

O decisor de máxima verossimilhança (ML, do inglês *maximum likelihood*) parte da premissa de que não há informação estatística consistente sobre as classes, *i.e.*, sobre $P(C_i)$.

Ideia: o padrão **x** observado é um reflexo da classe (verdadeira) à qual ele pertence. Além disso, existe uma probabilidade de cada classe $C_i, i = 1, \dots, Q$ poder originar um padrão exatamente com os atributos presentes em **x**, a qual é dada por $P(\mathbf{x}|C_i)$.

Então, o critério ML toma a decisão em favor da classe que apresenta o maior valor para a probabilidade $P(\mathbf{x}|C_i)$. Neste sentido, o ML escolhe a classe C_k mais plausível ou mais verossímil em relação ao padrão observado:

$$\text{ML:} \quad C_k = \arg \max_{C_i, i=1,\ldots,Q} P(\mathbf{x}|C_i) \tag{7}$$

- Supondo um problema de classificação binária ($Q = 2$), a regra de decisão ML pode ser escrita como:

Decida pela classe C_1 se $P(\mathbf{x}|C_1) > P(\mathbf{x}|C_2)$; caso contrário, decida por C_2.

Conexão: comparando as expressões associadas aos critérios MAP e ML, indicadas em (6) e (7), é possível perceber que a diferença fundamental reside no fato de o MAP explicitamente incorporar o conhecimento das probabilidades a *priori*. Curiosamente, quando temos um cenário em que as classes são equiprováveis,

$P(C_i) = {}^{1}/_{Q}$ e independe do índice i. Então, maximizar a probabilidade *a posteriori* fornecerá a mesma solução que o ML.

4.3. Generalização: Mínimo risco

Vamos estender agora a concepção do problema de decisão ao permitir que ações sejam tomadas como consequência direta de uma decisão. Ademais, vamos introduzir o conceito de função de perda.

- $\{\alpha_1, \ldots, \alpha_P\}$ denota o conjunto de P ações possíveis.
- A função de perda $l(\alpha_i, C_j)$ indica o custo referente à escolha da ação α_i quando a classe é C_j.

Suponha que um padrão $\mathbf{x}$ seja observado, e que α_i seja a ação que estamos considerando tomar. Se a classe de $\mathbf{x}$ é C_j, então, por definição, isto implica em uma perda $l(\alpha_i, C_j)$. Uma vez que $P(C_j|\mathbf{x})$ indica a probabilidade de a classe referente ao padrão $\mathbf{x}$ ser C_j, então a perda esperada ao tomarmos a ação α_i é dada por:

$$R(\alpha_i|\mathbf{x}) = \sum_{j=1}^{Q} l(\alpha_i, C_j)\, P(C_j|\mathbf{x}). \quad (8)$$

Na terminologia tipicamente utilizada em teoria de decisão, a perda esperada é denominada risco, e $R(\alpha_i|\mathbf{x})$ é o risco condicional.

Neste contexto, uma regra de decisão corresponde à função $\alpha(\mathbf{x})$ que nos diz qual ação deve ser tomada para todas as possíveis observações. Sendo assim, o risco total relacionado a uma regra de decisão é dado por:

$$R = \int R(\alpha(\mathbf{x})|\mathbf{x}) p(\mathbf{x})\, d\mathbf{x}, \quad (9)$$

onde $p(\mathbf{x})$ denota a função densidade de probabilidade do vetor de atributos **x.**

Para minimizar R, temos que escolher uma regra de decisão $\alpha(\mathbf{x})$ que minimize o risco condicional $R(\alpha(\mathbf{x})|\mathbf{x})$.

Ou seja, para minimizar o risco total, deve-se calcular o risco condicional, dado em (8), para $i = 1, \ldots, P$ e selecionar a ação α_i para a qual $R(\alpha_i|\mathbf{x})$ seja mínimo.

4.3.1. Mínima taxa de erro

Em um problema de classificação, usualmente a ação α_i é interpretada como a decisão de que o padrão **x** pertence à classe C_i. Então, se a ação α_i é tomada, mas o padrão pertence à classe C_j, a decisão estará correta somente se $i = j$, e um erro ocorrerá quando $i \neq j$.

Objetivo: minimizar a probabilidade de erro de classificação.

Neste caso, a função de perda é dada pela função *zero-one*:

$$l(\alpha_i, C_j) = \begin{cases} 0, & i = j \\ 1, & i \neq j \end{cases}, \tag{10}$$

com $i, j = 1, \ldots, Q$.

Com isto, o risco condicional é equivalente a:

$$R(\alpha_i|\mathbf{x}) = \sum_{j=1}^{Q} l(\alpha_i, C_j)\, P(C_j|\mathbf{x}) = \sum_{j \neq i} P(C_j|\mathbf{x}), \tag{11}$$

sendo que $\sum_{j \neq i} P(C_j|\mathbf{x})$ é a probabilidade média de erro.

Ora, como $\sum_j P(C_j|\mathbf{x}) = 1$, então podemos escrever que:

$$R(\alpha_i|\mathbf{x}) = 1 - P(C_i|\mathbf{x}). \tag{12}$$

Para minimizar o risco condicional, portanto, devemos escolher a ação α_i, ou, analogamente, a classe C_i, que leve ao mínimo valor de $R(\alpha_i|\mathbf{x})$. Isto equivale a escolher a classe C_i que maximiza $P(C_i|\mathbf{x})$, que é a probabilidade a *posteriori*.

Conclusão: a regra de decisão MAP nos leva a atingir a mínima probabilidade de erro de decisão.

4.4. *Naive Bayes*

Esta estratégia, como o próprio nome sugere, faz a hipótese "ingênua" de que os atributos do padrão $\mathbf{x}$ são condicionalmente independentes. Segundo esta abordagem, a ocorrência de um determinado valor para o atributo x_j é estatisticamente independente de qualquer outro atributo, dada a classe considerada. Com isto, $P(\mathbf{x}|C_i)$ pode ser decomposto da seguinte forma:

$$P(\mathbf{x}|C_i) = \prod_{j=1}^{K} P(x_j|C_i) \tag{13}$$

Vantagem: maior simplicidade na representação da probabilidade (ou distribuição) conjunta condicional do vetor **x**.

5. *Linear discriminant analysis*

A técnica conhecida como *linear discriminant analysis* (LDA), também chamada de *Fisher's linear discriminant*, é, na realidade, **uma ferramenta para redução de dimensionalidade**. Contudo, ao contrário de outras abordagens, como PCA (*Principal Component Analysis*), o **objetivo** da LDA é aprimorar a discriminação entre as classes existentes no espaço de dimensão reduzida.

Primeiramente, vamos considerar o cenário de classificação binária ($Q = 2$), para, depois, estender para o caso de $Q > 2$ classes.

Ideia: projetar linearmente os dados sobre uma direção, definida pelo vetor $\mathbf{w} \in \mathbb{R}^{K\times 1}$, tal que no espaço de dimensão reduzida – a princípio, unidimensional –, a discriminação entre as classes existentes seja maximizada.

$$\hat{y}(i) = \mathbf{w}^T\mathbf{x}(i) \tag{14}$$

Em geral, projetar os dados para espaços de dimensão reduzida leva a uma perda considerável de informação, e classes que são bem separadas no espaço original podem apresentar uma significativa sobreposição no espaço transformado. Entretanto, ajustando os elementos do vetor $\mathbf{w}$, podemos escolher uma direção que favoreça a separação entre as classes.

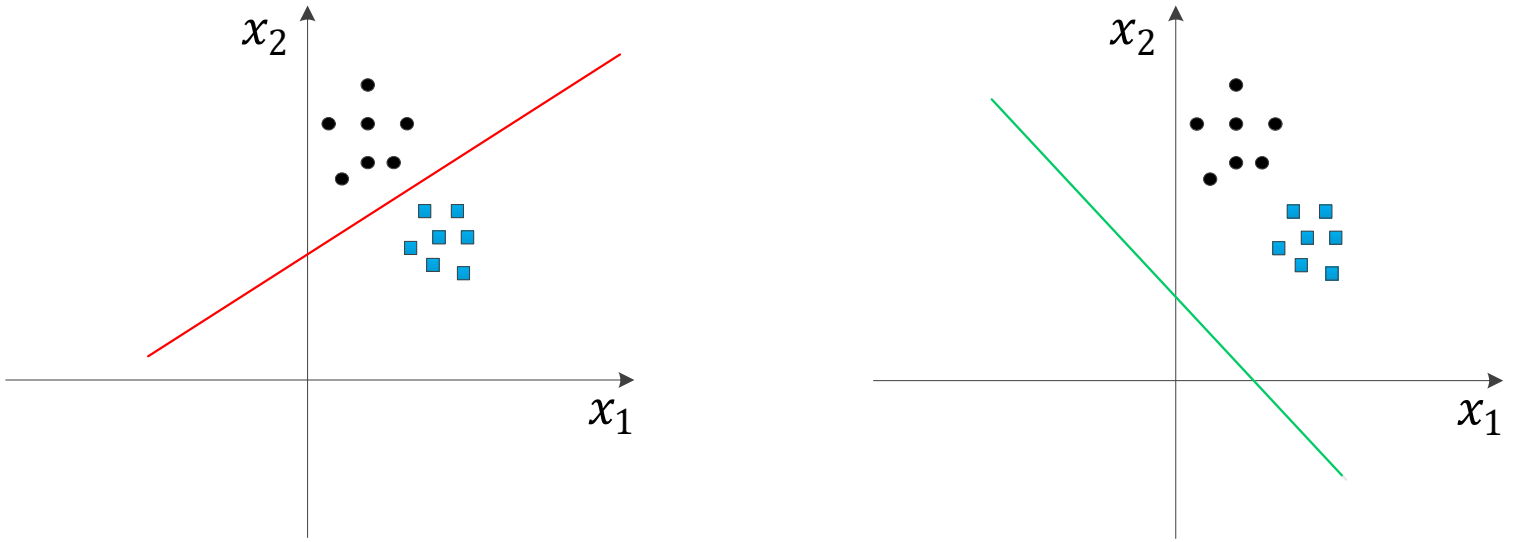


Figura. Exemplos de direções para a projeção dos dados no espaço unidimensional. A direção em (a) levaria a uma grande sobreposição das classes, enquanto a direção em (b) manteria as classes bem afastadas, facilitando a classificação.

Questão: como medir a separação entre as classes?

- Uma medida simples seria a distância entre os pontos médios das classes após a projeção.

Definições:

- $\mathbf{w}$: vetor unitário (i.e., $\|\mathbf{w}\| = 1$) que define a direção sobre a qual a projeção será feita.
- μ_1 e μ_2: médias das projeções das classes 1 e 2.
- $\boldsymbol{\mu}_1$ e $\boldsymbol{\mu}_2$: médias das classes 1 e 2.

Ora,

$$\mu_1 = \frac{1}{N_1}\sum_{i\in C_1}\mathbf{w}^T\mathbf{x}(i) = \mathbf{w}^T\left(\frac{1}{N_1}\sum_{i\in C_1}\mathbf{x}(i)\right) = \mathbf{w}^T\boldsymbol{\mu}_1 \qquad (15)$$

Analogamente, $\mu_2 = \mathbf{w}^T\boldsymbol{\mu}_2$.

Critério inicial: $\max_{\mathbf{w}} J = |\mu_1 - \mu_2|$ (16)

Solução: $\mathbf{w} \propto (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$.

Problema: o critério em (16) não considera a dispersão dos padrões de cada classe, em cada direção, em torno do ponto médio.

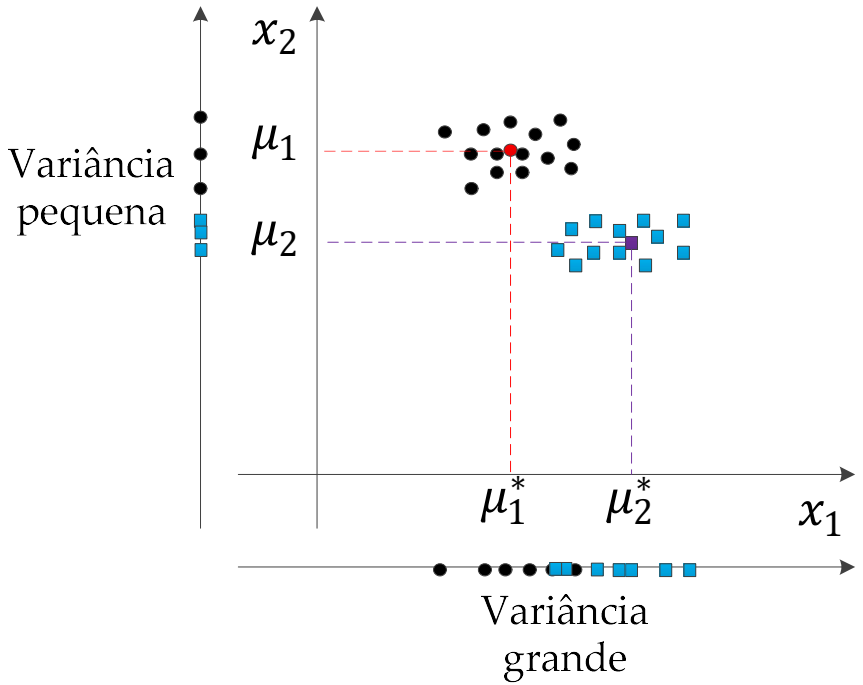


Figura. Usando o critério em (16), a direção horizontal seria preferida em relação à direção vertical, pois $|\mu_1^* - \mu_2^*| > |\mu_1 - \mu_2|$. Entretanto, a projeção na direção horizontal apresenta sobreposição entre as classes, por conta do maior espalhamento das amostras, o que prejudica a classificação.

Visão complementar:

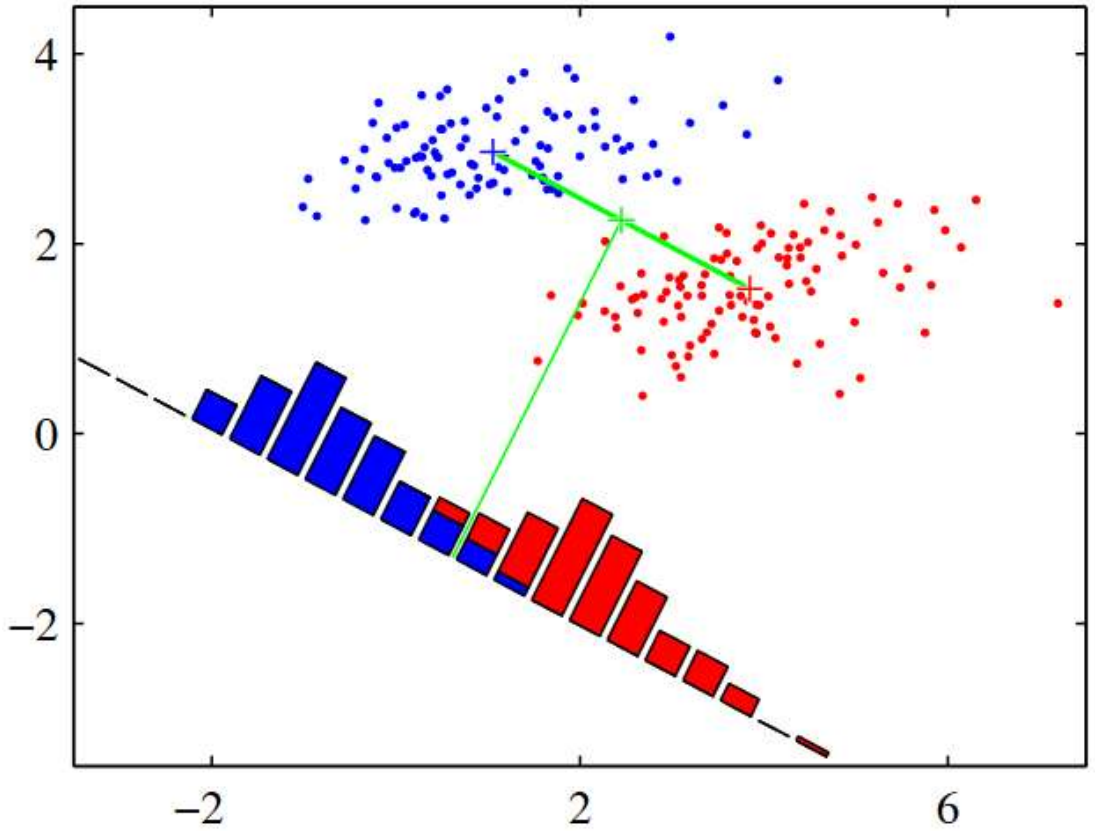


Figura extraída de (BISHOP, 2006). Dados referentes às classes 1 e 2 (azul e vermelho), junto com os histogramas resultantes das projeções sobre a direção $(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$. Observe que há uma sobreposição considerável entre as amostras após a projeção.

Proposta de Fisher: maximizar a separação entre as médias das classes projetadas e, ao mesmo tempo, reduzir a variância dentro de cada classe, de modo a minimizar a sobreposição entre as classes.

Variância intra-classe:

$$s_k^2 = \sum_{i \in C_k} (\hat{y}(i) - \mu_i)^2 \tag{17}$$

Então, podemos definir que a variância total intra-classe é dada por $s_1^2 + s_2^2$.

Variância inter-classe:

$$(\mu_1 - \mu_2)^2 \tag{18}$$

Critério de Fisher: o discriminante linear de Fisher consiste em projetar os dados sobre a direção **w** que maximiza o funcional:

$$J(\mathbf{w}) = \frac{(\mu_1 - \mu_2)^2}{s_1^2 + s_2^2} \tag{19}$$

O objetivo é manter as médias projetadas tão afastadas quanto possível, bem como forçar a redução das variâncias das classes.

Vamos explicitar a dependência de $J(\mathbf{w})$ de (19) em relação ao vetor **w.**

Definindo as matrizes de covariâncias de cada classe para os dados originais:

$$\mathbf{S}_1 = \sum_{i \in C_1} (\mathbf{x}(i) - \boldsymbol{\mu}_1)(\mathbf{x}(i) - \boldsymbol{\mu}_1)^T \tag{20}$$

e

$$\mathbf{S}_2 = \sum_{i \in C_2} (\mathbf{x}(i) - \boldsymbol{\mu}_2)(\mathbf{x}(i) - \boldsymbol{\mu}_2)^T \tag{21}$$

temos que a matriz de covariância intra-classe é dada por:

$$\mathbf{S}_w = \mathbf{S}_1 + \mathbf{S}_2 \tag{22}$$

Vamos, então, relacionar o valor de $s_1^2 + s_2^2$ em (19) com a matriz de covariância intra-classe $\mathbf{S}_w$. O primeiro elemento (s_1^2) é definido como:

$$s_1^2 = \sum_{i \in C_1} (\hat{y}(i) - \mu_1)^2. \tag{23}$$

Como $\hat{y}(i) = \mathbf{w}^T \mathbf{x}(i)$ e $\mu_1 = \mathbf{w}^T \boldsymbol{\mu}_1$, temos que:

$$s_1^2 = \sum_{i \in C_1} (\mathbf{w}^T \mathbf{x}(i) - \mathbf{w}^T \boldsymbol{\mu}_1)^2 = \sum_{i \in C_1} (\mathbf{w}^T [\mathbf{x}(i) - \boldsymbol{\mu}_1])^2 \tag{24}$$

Prosseguindo com o desenvolvimento:

$$s_1^2 = \sum_{i \in C_1} (\mathbf{w}^T [\mathbf{x}(i) - \boldsymbol{\mu}_1])^T (\mathbf{w}^T [\mathbf{x}(i) - \boldsymbol{\mu}_1]) \tag{25}$$

Como $\mathbf{w}^T [\mathbf{x}(i) - \boldsymbol{\mu}_1]$ é um escalar, $\mathbf{w}^T [\mathbf{x}(i) - \boldsymbol{\mu}_1] = [\mathbf{x}(i) - \boldsymbol{\mu}_1]^T \mathbf{w}$. Então,

$$s_1^2 = \sum_{i \in C_1} ([\mathbf{x}(i) - \boldsymbol{\mu}_1]^T \mathbf{w})^T ([\mathbf{x}(i) - \boldsymbol{\mu}_1]^T \mathbf{w}) \tag{26}$$

Aplicando o operador $(\cdot)^T$ no primeiro fator:

$$s_1^2 = \sum_{i \in C_1} \mathbf{w}^T [\mathbf{x}(i) - \boldsymbol{\mu}_1][\mathbf{x}(i) - \boldsymbol{\mu}_1]^T \mathbf{w} \tag{27}$$

Usando (20), obtemos:

$$s_1^2 = \mathbf{w}^T \left(\sum_{i \in C_1} [\mathbf{x}(i) - \boldsymbol{\mu}_1][\mathbf{x}(i) - \boldsymbol{\mu}_1]^T \right) \mathbf{w} = \mathbf{w}^T \mathbf{S}_1 \mathbf{w} \quad (28)$$

Analogamente, $s_2^2 = \mathbf{w}^T \mathbf{S}_2 \mathbf{w}$.

Logo, podemos escrever que:

$$s_1^2 + s_2^2 = \mathbf{w}^T \mathbf{S}_1 \mathbf{w} + \mathbf{w}^T \mathbf{S}_2 \mathbf{w} = \mathbf{w}^T \mathbf{S}_w \mathbf{w}. \quad (29)$$

Considere, agora, a matriz de covariância inter-classe dos dados originais:

$$\mathbf{S}_B = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T. \quad (30)$$

Com isto, podemos reescrever a separação entre as médias projetadas:

$$(\mu_1 - \mu_2)^2 = (\mathbf{w}^T \boldsymbol{\mu}_1 - \mathbf{w}^T \boldsymbol{\mu}_2)^2 = \mathbf{w}^T (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \mathbf{w} = \mathbf{w}^T \mathbf{S}_B \mathbf{w} \quad (31)$$

Combinando (29) com (31), o critério de Fisher de (19) é reescrito como:

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}} \quad (32)$$

Solução:

Derivando (32) com respeito a $\mathbf{w}$ e igualando a zero:

$$\frac{\partial J(\mathbf{w})}{\partial \mathbf{w}} = \frac{\partial \left(\frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}} \right)}{\partial \mathbf{w}} = 0 \tag{33}$$

Explorando a propriedade da derivada do quociente:

$$(\mathbf{w}^T \mathbf{S}_w \mathbf{w}) \frac{\partial \mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\partial \mathbf{w}} - (\mathbf{w}^T \mathbf{S}_B \mathbf{w}) \frac{\partial \mathbf{w}^T \mathbf{S}_w \mathbf{w}}{\partial \mathbf{w}} = 0 \tag{34}$$

Ora, $\frac{\partial \mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\partial \mathbf{w}} = 2\mathbf{S}_B \mathbf{w}$ e $\frac{\partial \mathbf{w}^T \mathbf{S}_w \mathbf{w}}{\partial \mathbf{w}} = 2\mathbf{S}_w \mathbf{w}$.

Então, (34) se reduz a:

$$(\mathbf{w}^T \mathbf{S}_w \mathbf{w}) 2\mathbf{S}_B \mathbf{w} - (\mathbf{w}^T \mathbf{S}_B \mathbf{w}) 2\mathbf{S}_w \mathbf{w} = 0 \tag{35}$$

Dividindo ambos os lados por $(\mathbf{w}^T \mathbf{S}_w \mathbf{w})$:

$$\mathbf{S}_B \mathbf{w} - \frac{(\mathbf{w}^T \mathbf{S}_B \mathbf{w})}{(\mathbf{w}^T \mathbf{S}_w \mathbf{w})} \mathbf{S}_w \mathbf{w} = 0 \tag{36}$$

O termo $\frac{(\mathbf{w}^T \mathbf{S}_B \mathbf{w})}{(\mathbf{w}^T \mathbf{S}_w \mathbf{w})}$ é precisamente o valor de $J(\mathbf{w})$. Então,

$$\mathbf{S}_B \mathbf{w} - J(\mathbf{w}) \mathbf{S}_w \mathbf{w} = 0. \tag{37}$$

Multiplicando por $\mathbf{S}_w{}^{-1}$ e passando o segundo termo para o outro lado, chegamos a:

$$\mathbf{S}_w{}^{-1}\mathbf{S}_B\mathbf{w} = J(\boldsymbol{w})\mathbf{w} \tag{38}$$

A equação (38) caracteriza o problema de obtenção dos autovalores generalizados.

Então:
$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \frac{(\mathbf{w}^T\mathbf{S}_B\mathbf{w})}{(\mathbf{w}^T\mathbf{S}_w\mathbf{w})} \propto \mathbf{S}_w{}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2). \tag{39}$$

Caminho alternativo: a partir da definição de $\mathbf{S_B}$, percebemos que o produto $\mathbf{S_B w}$ é dado por:

$$\mathbf{S}_B\mathbf{w} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T\mathbf{w} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)\epsilon, \tag{40}$$

onde $\epsilon = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T\mathbf{w}$ é um escalar. Ou seja, $\mathbf{S}_B\mathbf{w}$ gera um vetor que está na mesma direção que $(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$.

Substituindo (40) em (37), obtemos:

$$(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)\epsilon = J(\boldsymbol{w})\mathbf{S}_w\mathbf{w} \tag{41}$$

Isolando o termo que depende de $\mathbf{w}$:

$$\mathbf{S}_w\mathbf{w} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)\frac{\epsilon}{J(\boldsymbol{w})} \tag{42}$$

Já que estamos interessados somente na direção de **w**, podemos desprezar o fator de escala $\frac{\epsilon}{J(\boldsymbol{w})}$. Assim, multiplicando ambos os lados por $\mathbf{S}_w{}^{-1}$, obtemos (39).

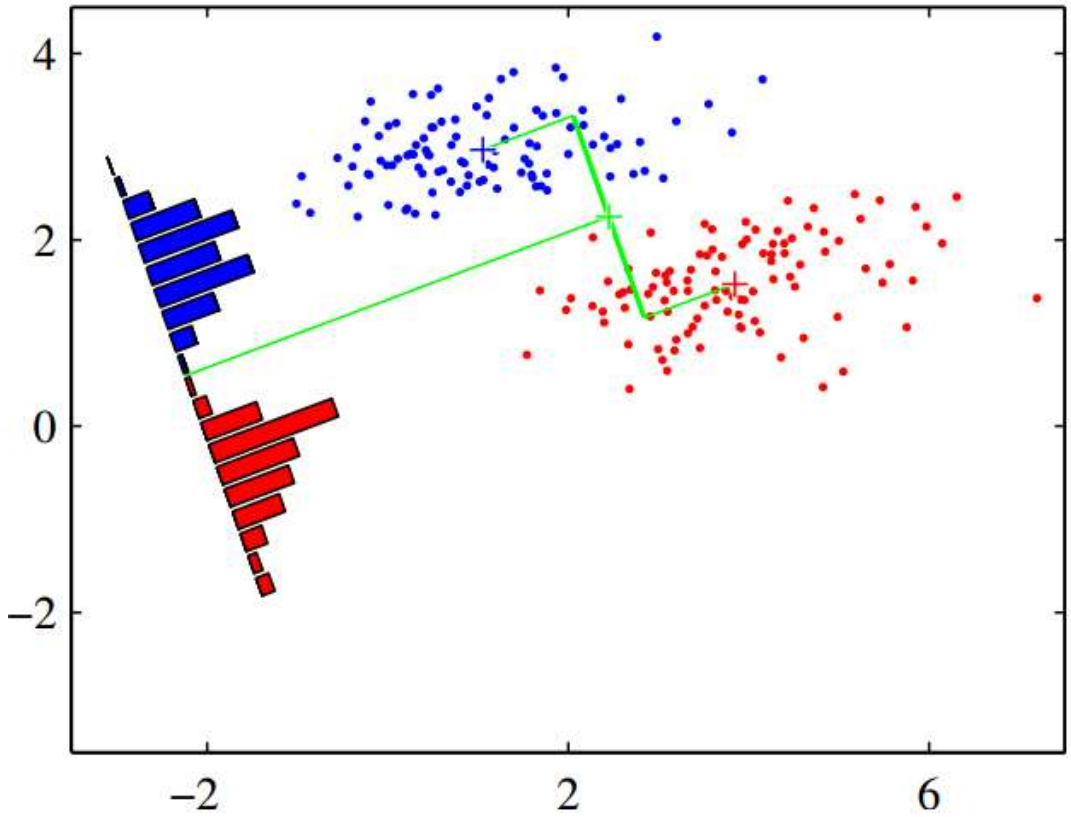


Figura extraída de (BISHOP, 2006). Dados referentes às classes 1 e 2 (azul e vermelho), junto com os histogramas resultantes das projeções sobre a direção determinada pelo discriminante de Fisher. Observe que há uma melhora considerável na separação entre as classes após a projeção, se comparada à solução anterior que desprezava as dispersões.

5.1. Etapa de decisão

Uma vez que a direção de projeção tenha sido determinada, conforme o critério de Fisher, e os padrões tenham sido projetados para o espaço de dimensão reduzida, é necessário decidir a qual classe um padrão novo pertence.

Para isto, aplica-se um limiar (*threshold*) sobre o valor do dado projetado, de tal modo que se $\hat{y}(i) = \mathbf{w}^T\mathbf{x}(i) > -w_0$, então consideramos que o padrão $\mathbf{x}(i)$ pertence à classe C_1. Caso contrário, ele é designado para a classe C_2.

Questão: como definir o valor do *threshold*?

- Heurísticas: $w_0 = \frac{\mu_1+\mu_2}{2}$ ou $w_0 = p_1\mu_1 + p_2\mu_2$, onde p_1 e p_2 representam as probabilidades *a priori* das classes C_1 e C_2.
- Através da teoria bayesiana de decisão: requer a definição das PDFs condicionais e das probabilidades *a priori* das classes.
- Busca unidimensional + validação cruzada.

5.2. Relação com a solução de quadrados mínimos para regressão linear

É possível estabelecer uma relação entre o método baseado em quadrados mínimos e o discriminante de Fisher (BISHOP, 2006):

- Definindo o valor desejado na regressão para amostras da classe C_1 como sendo igual a $y(i) = \frac{N}{N_1}, i \in C_1$, enquanto, para a classe C_2 o valor alvo é $y(i) = -\frac{N}{N_2}, i \in C_2$, a solução de quadrados mínimos é dada por:

$$w_0 = -\mathbf{w}^T \boldsymbol{\mu}, \tag{43}$$

onde $\boldsymbol{\mu}$ denota o vetor média de todo o conjunto de dados, e

$$w \propto \mathbf{S}_w{}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2). \tag{44}$$

5.3. Extensão para múltiplas classes

Agora, consideramos o caso em que a classificação envolve $Q > 2$ classes possíveis. A projeção a ser feita é do espaço K-dimensional para o espaço K'-dimensional. Assumimos que $K > Q$.

Definições:

A matriz de covariância intra-classe pode ser prontamente escrita como:

$$\mathbf{S}_w = \sum_{i=1}^{Q} \mathbf{S}_i, \tag{45}$$

onde, à semelhança do caso anterior,

$$\mathbf{S}_i = \sum_{j \in C_i} (\mathbf{x}(j) - \boldsymbol{\mu}_i)(\mathbf{x}(j) - \boldsymbol{\mu}_i)^T, \tag{46}$$

em que

$$\boldsymbol{\mu}_i = \frac{1}{N_i} \sum_{j \in C_i} \mathbf{x}(j). \tag{47}$$

Porém, a generalização para a matriz de covariância inter-classe não é tão direta.

Seja $\boldsymbol{\mu}$ o vetor média total e $\mathbf{S}_T$ a matriz de covariância total, cujas definições são indicadas abaixo:

$$\boldsymbol{\mu} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}(i) = \frac{1}{N}\sum_{i=1}^{N} N_i \boldsymbol{\mu}_i \tag{48}$$

e

$$\mathbf{S}_T = \sum_{i=1}^{N}(\mathbf{x}(i) - \boldsymbol{\mu})(\mathbf{x}(i) - \boldsymbol{\mu})^T \tag{49}$$

Então, podemos escrever que:

$$\mathbf{S}_T = \sum_{i=1}^{Q}\sum_{j\in C_i}(\mathbf{x}(j) - \boldsymbol{\mu}_i + \boldsymbol{\mu}_i - \boldsymbol{\mu})(\mathbf{x}(j) - \boldsymbol{\mu}_i + \boldsymbol{\mu}_i - \boldsymbol{\mu})^T \tag{50}$$

Separando a expressão em dois termos:

$$\mathbf{S}_T = \sum_{i=1}^{Q}\sum_{j\in C_i}(\mathbf{x}(j) - \boldsymbol{\mu}_i)(\mathbf{x}(j) - \boldsymbol{\mu}_i)^T + \sum_{i=1}^{Q}\sum_{j\in C_i}(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \tag{51}$$

Prosseguindo:

$$\mathbf{S}_T = \mathbf{S}_w + \sum_{i=1}^{Q} N_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \tag{52}$$

É natural definir o segundo termo como sendo a matriz de covariância inter-classe, de modo que a matriz de covariância total seja dada pela soma das componentes inter-classe e intra-classe:

$$\mathbf{S}_B = \sum_{i=1}^{Q} N_i(\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \tag{53}$$

e

$$\mathbf{S}_T = \mathbf{S}_w + \mathbf{S}_B \tag{54}$$

A projeção do espaço K-dimensional para o espaço K'-dimensional é feita através de K' funções discriminantes:

$$\hat{\mathbf{y}}_k(i) = \mathbf{w}_k^T \mathbf{x}(i), \qquad k = 1, \ldots, K' \tag{55}$$

Explorando uma notação matricial, o vetor projetado $\hat{\mathbf{y}}(i)$ pode ser definido como:

$$\hat{\mathbf{y}}(i) = \mathbf{W}^T \mathbf{x}(i), \tag{56}$$

onde $\mathbf{W} \in \mathbb{R}^{K \times K'}$ é a matriz de projeção, sendo que cada coluna de $\mathbf{W}$ corresponde a uma direção de projeção $\mathbf{w}_k$.

Após a projeção, os dados $\hat{\mathbf{y}}(i)$ também possuem matrizes próprias de covariâncias intra e inter-classe:

$$\hat{\mathbf{S}}_w = \sum_{i=1}^{Q} \sum_{j \in C_i} (\hat{\mathbf{y}}(j) - \hat{\boldsymbol{\mu}}_i)(\hat{\mathbf{y}}(j) - \hat{\boldsymbol{\mu}}_i)^T \tag{57}$$

e

$$\hat{\mathbf{S}}_B = \sum_{i=1}^{Q} N_i (\hat{\boldsymbol{\mu}}_i - \hat{\boldsymbol{\mu}})(\hat{\boldsymbol{\mu}}_i - \hat{\boldsymbol{\mu}})^T, \tag{58}$$

onde $\hat{\boldsymbol{\mu}}_i = \frac{1}{N_i} \sum_{j \in C_i} \hat{\mathbf{y}}(j)$ é o vetor média dos dados projetados referentes à classe C_i e $\hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_{i=1}^{Q} N_i \, \hat{\boldsymbol{\mu}}_i$ é o vetor média total dos dados projetados.

De forma semelhante ao caso em que $Q = 2$ classes, é possível mostrar que:

$$\hat{\mathbf{S}}_w = \mathbf{W}^T \mathbf{S}_w \mathbf{W} \tag{59}$$

e

$$\hat{\mathbf{S}}_B = \mathbf{W}^T \mathbf{S}_B \mathbf{W} \tag{60}$$

O discriminante de Fisher busca a matriz de transformação $\mathbf{W}$ que, em algum sentido, maximize a razão da dispersão inter-classe pela dispersão intra-classe. Precisamos, portanto, definir uma medida (escalar) que expresse este objetivo.

Existem várias possíveis escolhas (DUDA, HART & STORK, 2000; BISHOP, 2006):

$$J(\mathbf{W}) = \mathrm{tr}\left(\hat{\mathbf{S}}_w{}^{-1} \hat{\mathbf{S}}_B\right) \tag{61}$$

Ou,

$$J(\mathbf{W}) = \frac{\det\left(\hat{\mathbf{S}}_B\right)}{\det\left(\hat{\mathbf{S}}_w\right)} = \frac{\det\left(\mathbf{W}^T \mathbf{S}_B \mathbf{W}\right)}{\det\left(\mathbf{W}^T \mathbf{S}_w \mathbf{W}\right)} \tag{62}$$

Solução: os pesos ótimos são dados pelos autovetores generalizados associados aos K' maiores autovalores.

Observação: como a matriz $\mathbf{S}_B$ é composta pela soma de Q matrizes de *rank* 1, e como somente $(Q-1)$ destas matrizes são independentes por conta da restrição dada em (48), então $\mathbf{S}_B$ tem, no máximo, *rank* igual a $(Q-1)$, de modo que há, no máximo, $(Q-1)$ autovalores não-nulos. Portanto, a redução de dimensionalidade pode ser feita até a dimensão $(Q-1)$. Ou seja, o máximo valor de K' é $(Q-1)$.

6. Regressão logística

Trata-se de uma abordagem de classificação que também tenta promover a separação das classes com base em fronteiras de decisão lineares. Porém, diferentemente do que ocorre na regressão linear, a saída do modelo é gerada a partir de um mapeamento não-linear (DUDA, HART & STORK, 2001; ALPAYDIN, 2013).

A fim de facilitar o entendimento das características desta técnica, vamos começar a exposição pelo cenário de classificação binária, fazendo, posteriormente, a extensão para o caso multi-classe.

6.1. Classificação binária

Neste caso, o modelo produz uma única saída, $\hat{y}(\mathbf{x})$, por meio do seguinte mapeamento:

$$\hat{y}(\mathbf{x}) = \frac{e^{(w_0+w_1x_1+\cdots+w_Kx_K)}}{1+e^{(w_0+w_1x_1+\cdots+w_Kx_K)}} = \frac{1}{1+e^{-(w_0+w_1x_1+\cdots+w_Kx_K)}} \tag{63}$$

Explorando uma notação vetorial, podemos escrever que:

$$\hat{y}(\mathbf{x}) = \frac{1}{1 + e^{-(\boldsymbol{\phi}(\mathbf{x})^T \mathbf{w})}}, \tag{64}$$

onde $\mathbf{w} = [w_0 \; w_1 \; \cdots \; w_K]^T$ e $\boldsymbol{\phi}(\mathbf{x}) = [1 \; x_1 \; \cdots \; x_K]^T$. A função $g(z) = \frac{1}{1+e^{-z}}$ é denominada função logística (sigmoide) e realiza um mapeamento $\mathbb{R} \rightarrow [0,1]$, como podemos observar na figura a seguir.

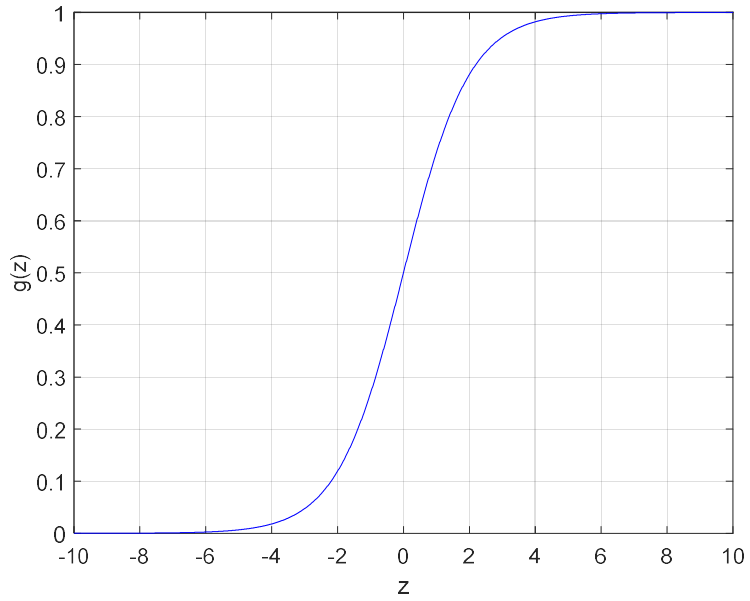


Figura. Gráfico da função logística.

Propriedades:

- Restrição: $0 \leq \hat{y}(\mathbf{x}) \leq 1$.
- A saída $\hat{y}(\mathbf{x})$ representa a probabilidade de $\mathbf{x}$ pertencer à classe positiva (C_2), para a qual a saída desejada é $y = 1$. Ou seja, $\hat{y}(\mathbf{x}) = P(C_2|\mathbf{x}; \mathbf{w})$.
- Logo, $\left(1 - \hat{y}(\mathbf{x})\right) = P(C_1|\mathbf{x}; \mathbf{w})$.
- A fronteira de decisão é determinada quando $P(C_1|\mathbf{x}; \mathbf{w}) = P(C_2|\mathbf{x}; \mathbf{w})$. Isto ocorre quando $P(C_2|\mathbf{x}; \mathbf{w}) = \hat{y}(\mathbf{x}) = g(\boldsymbol{\phi}(\mathbf{x})^T \mathbf{w}) = 0{,}5$. Observando a figura anterior, percebemos que $g(z) = 0{,}5$ quando $z = 0$. Sendo assim, a fronteira de decisão é caracterizada por $\boldsymbol{\phi}(\mathbf{x})^T \mathbf{w} = w_0 + w_1 x_1 + \cdots + w_K x_K = 0$, e corresponde a um hiperplano.

6.1.1. Critério de ajuste

No caso da regressão logística, adotar o critério de erro quadrático médio (ou quadrados mínimos) não constitui uma opção muito apropriada para a adaptação dos parâmetros. A função custo de quadrados mínimos é dada por:

$$J_e(\mathbf{w}) = \frac{1}{N}\sum_{i=0}^{N-1}(y(i) - \hat{y}(i))^2 = \frac{1}{N}\sum_{i=0}^{N-1}(y(i) - g(\boldsymbol{\phi}(\mathbf{x})^T\mathbf{w}))^2 \qquad (65)$$

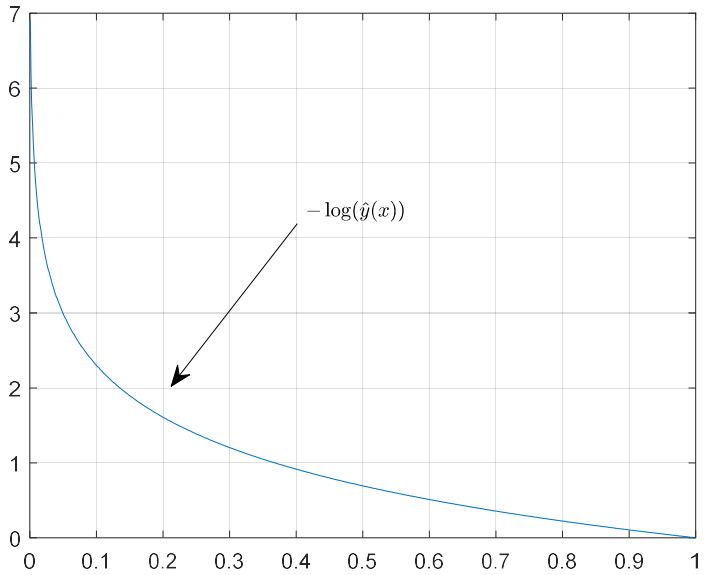
Como $g(\cdot)$ é uma função não-linear, $J_e(\mathbf{w})$ não é uma função convexa, de maneira que sua superfície está sujeita à presença de mínimos locais.

Ideia: adotar uma função custo que se acomode melhor às características do problema de classificação.

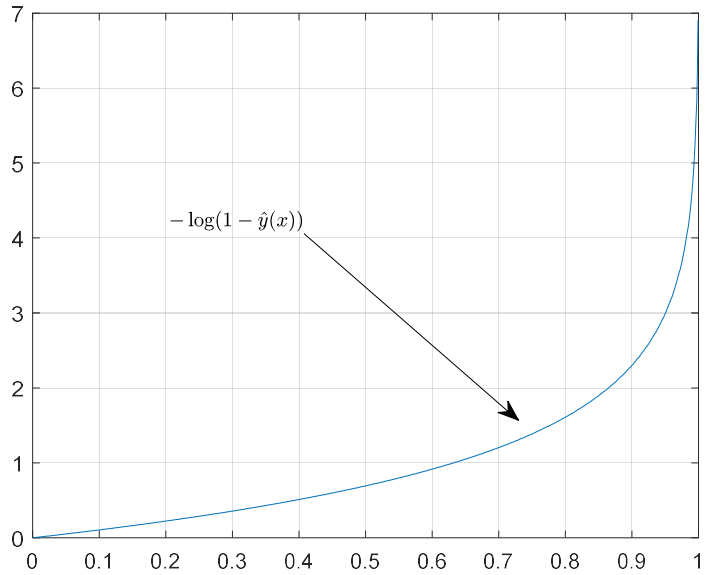
Proposta (intuitiva):

$$\text{Custo}(\hat{y}(\mathbf{x}); y) = \begin{cases} -\log \hat{y}(\mathbf{x}), & \text{se } y = 1 \\ -\log\big(1 - \hat{y}(\mathbf{x})\big), & \text{se y} = 0 \end{cases} \qquad (66)$$

A figura a seguir mostra as duas situações possíveis para o custo. Como podemos observar, a penalização aplicada a cada saída reflete o erro de classificação.



(a) $y = 1$



(b) $y = 0$

Figura. Em (a), o custo é nulo somente se a saída é $\hat{y}(\mathbf{x}) = 1$ (ou seja, se a classe C_2 é corretamente identificada). À medida que $\hat{y}(\mathbf{x}) \to 0$, o custo tende a infinito. Em (b), o custo é nulo quando a saída é $\hat{y}(\mathbf{x}) = 0$ (ou seja, quando a classe C_1 é corretamente identificada). Conforme $\hat{y}(\mathbf{x}) \to 1$, o custo tende a infinito.

Podemos reduzir a definição do custo em (66) a uma expressão única:

$$\text{Custo}(\hat{y}(\mathbf{x}); y) = \underbrace{-y \log\big(\hat{y}(\mathbf{x})\big)}_{\text{Só exerce influência se } y=1} \underbrace{-(1-y) \log\big(1-\hat{y}(\mathbf{x})\big)}_{\text{Penaliza apenas se } y=0}. \tag{67}$$

Com isto, podemos definir a seguinte função custo:

$$J_{\text{CE}}(\mathbf{w}) = -\frac{1}{N} \sum_{i=0}^{N-1} y(i) \log\Big(\hat{y}\big(\mathbf{x}(i)\big)\Big) + (1-y(i)) \log\big(1-\hat{y}(\mathbf{x}(i))\big) \tag{68}$$

Interessantemente, esta função custo também pode ser obtida a partir de uma abordagem mais formal, a saber, baseada no princípio da máxima verossimilhança.

Ora, sabemos que:

$$P(y(i)=0|\mathbf{x}(i); \mathbf{w}) = 1 - \hat{y}(\mathbf{x}(i)) \tag{69}$$

e

$$P(y(i)=1|\mathbf{x}(i); \mathbf{w}) = \hat{y}(\mathbf{x}(i)) \tag{70}$$

Então, podemos escrever que:

$$P(y(i) = y_i|\mathbf{x}(i); \mathbf{w}) = \hat{y}(\mathbf{x}(i))^{y_i}\big(1 - \hat{y}(\mathbf{x}(i))\big)^{(1-y_i)} \tag{71}$$

Partindo da hipótese de que há independência estatística entre os dados observados, a função de verossimilhança pode ser escrita como:

$$\mathcal{L}(\mathbf{w}|\mathbf{X}) = P(\mathbf{y}|\mathbf{X}; \mathbf{w}) = \prod_{i=0}^{N-1} P(y(i) = y_i|\mathbf{x}(i); \mathbf{w}) \tag{72}$$

Usando (71), podemos escrever que:

$$\mathcal{L}(\mathbf{w}|\mathbf{X}) = \prod_{i=0}^{N-1} \hat{y}(\mathbf{x}(i))^{y_i}\big(1 - \hat{y}(\mathbf{x}(i))\big)^{(1-y_i)} \tag{73}$$

Ora, maximizar $\mathcal{L}(\mathbf{w}|\mathbf{X})$ é equivalente a minimizar $-\log \mathcal{L}(\mathbf{w}|\mathbf{X})$. Aplicando, então, o logaritmo e inserindo um fator de escala $\frac{1}{N}$, a função custo a ser otimizada pode ser definida como:

$$J_{\mathrm{CE}}(\mathbf{w}) = -\frac{1}{N}\sum_{i=0}^{N-1} y_i \log \hat{y}(\mathbf{x}(i)) + (1 - y_i)\log\big(1 - \hat{y}(\mathbf{x}(i))\big) \tag{74}$$

Esta expressão está associada ao critério conhecido como **entropia cruzada** (CE, do inglês *cross-entropy*).

- Sob certas condições, minimizar a entropia cruzada pode ser visto como uma tentativa de minimizar a divergência de Kullback-Leibler entre a probabilidade verdadeira (y) e a probabilidade estimada ($\hat{y}(\mathbf{x})$).

6.1.2. Processo de treinamento

A função custo associada à entropia cruzada, cuja expressão é dada em (74), é uma função convexa. Contudo, não é possível obter uma solução em forma fechada para os coeficientes $\mathbf{w}$. Sendo assim, é preciso recorrer a algoritmos iterativos para realizar o treinamento do modelo. À semelhança do tópico anterior, vamos apresentar o algoritmo de gradiente descendente para a minimização da entropia cruzada.

Antes de começarmos a calcular o vetor gradiente de $J_{\mathrm{CE}}(\mathbf{w})$, vamos reescrever a função custo em (74) explorando as seguintes equivalências:

$$\log \hat{y}(\mathbf{x}(i)) = \log\left(\frac{1}{1+e^{-(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w})}}\right) = -\log\left(1+e^{-(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w})}\right) \tag{75}$$

e

$$\begin{aligned}\log\left(1-\hat{y}(\mathbf{x}(i))\right) &= \log\left(1-\frac{1}{1+e^{-(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w})}}\right)\\ &= \log\left(e^{-(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w})}\right) - \log\left(1+e^{-(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w})}\right)\\ &= -\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w} - \log\left(1+e^{-(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w})}\right)\end{aligned} \tag{76}$$

Assim, a nova expressão para a entropia cruzada é dada por:

$$\begin{aligned}J_{\mathrm{CE}}(\mathbf{w}) = -\frac{1}{N}\sum_{i=0}^{N-1} &-y_i \log\left(1+e^{-\left(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w}\right)}\right)\\ &+(1-y_i)\left[-\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w} - \log\left(1+e^{-(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w})}\right)\right]\end{aligned} \tag{77}$$

O termo $-y_i \log\left(1 + e^{-\left(\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}\right)}\right)$ é cancelado com um dos elementos gerados a partir do produto envolvido no segundo termo, de maneira que:

$$J_{\text{CE}}(\mathbf{w}) = -\frac{1}{N} \sum_{i=0}^{N-1} -\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w} + y_i \boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w} - \log\left(1 + e^{-\left(\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}\right)}\right) \quad (78)$$

Ora, $-\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w} = \log e^{-\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}} = -\log e^{\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}}$. Sendo assim,

$$\begin{aligned} -\boldsymbol{\phi}\left(\mathbf{x}(i)\right)^T \mathbf{w} - \log\left(1 + e^{-\left(\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}\right)}\right) &= -\log e^{\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}} - \log\left(1 + e^{-\left(\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}\right)}\right) \\ &= -\log\left(1 + e^{\left(\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}\right)}\right). \end{aligned} \quad (79)$$

Com isto, a entropia cruzada se torna:

$$J_{\text{CE}}(\mathbf{w}) = -\frac{1}{N} \sum_{i=0}^{N-1} y_i \boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w} - \log\left(1 + e^{\left(\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}\right)}\right). \quad (80)$$

Vetor gradiente

$$\frac{\partial [y_i \boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}]}{\partial \mathbf{w}} = y_i \boldsymbol{\phi}(\mathbf{x}(i))^T \quad (81)$$

$$\frac{\partial\left[\log\left(1+e^{\left(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w}\right)}\right)\right]}{\partial\mathbf{w}} = \frac{1}{1+e^{\left(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w}\right)}} e^{\left(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w}\right)}\boldsymbol{\phi}\left(\mathbf{x}(i)\right)^T$$

$$= \frac{1}{1+e^{-\left(\boldsymbol{\phi}(\mathbf{x}(i))^T\mathbf{w}\right)}}\boldsymbol{\phi}\left(\mathbf{x}(i)\right)^T$$

$$= \hat{y}(\mathbf{x}(i))\boldsymbol{\phi}\left(\mathbf{x}(i)\right)^T. \tag{82}$$

Logo, combinando (81) e (82), o vetor gradiente (do tipo vetor linha) pode ser escrito como:

$$\frac{\partial J_{\mathrm{CE}}(\mathbf{w})}{\partial\mathbf{w}} = -\frac{1}{N}\sum_{i=0}^{N-1} y_i\boldsymbol{\phi}\left(\mathbf{x}(i)\right)^T - \hat{y}\left(\mathbf{x}(i)\right)\boldsymbol{\phi}\left(\mathbf{x}(i)\right)^T$$

$$= -\frac{1}{N}\sum_{i=0}^{N-1}\left(y_i - \hat{y}\left(\mathbf{x}(i)\right)\right)\boldsymbol{\phi}\left(\mathbf{x}(i)\right)^T$$

$$= -\frac{1}{N}\mathbf{e}^T\boldsymbol{\Phi}, \tag{83}$$

onde $\mathbf{e} = [e(0)\ \cdots\ e(N-1)]^T$, $e(i) = y_i - \hat{y}\left(\mathbf{x}(i)\right)$ e $\boldsymbol{\Phi} \in \mathbb{R}^{N\times(K+1)}$:

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\phi}(\mathbf{x}(0))^T \\ \vdots \\ \boldsymbol{\phi}(\mathbf{x}(N-1))^T \end{bmatrix} \tag{84}$$

Observações:

- A expressão do vetor gradiente da entropia cruzada, dada em (83), para a regressão logística é idêntica àquela obtida na regressão linear com o critério de quadrados mínimos.
- A regressão logística também pode ser facilmente estendida para incorporar termos polinomiais envolvendo os atributos de entrada (*e.g.*, $x_1{}^2$). Assim, fronteiras de decisão não-lineares podem ser produzidas.
- Assim como discutido no tópico anterior, a regressão logística está sujeita à ocorrência de *overfitting* e/ou *underfitting*. Por isso, técnicas de regularização podem ser empregadas em seu treinamento, assim como validação cruzada e *early stopping*.

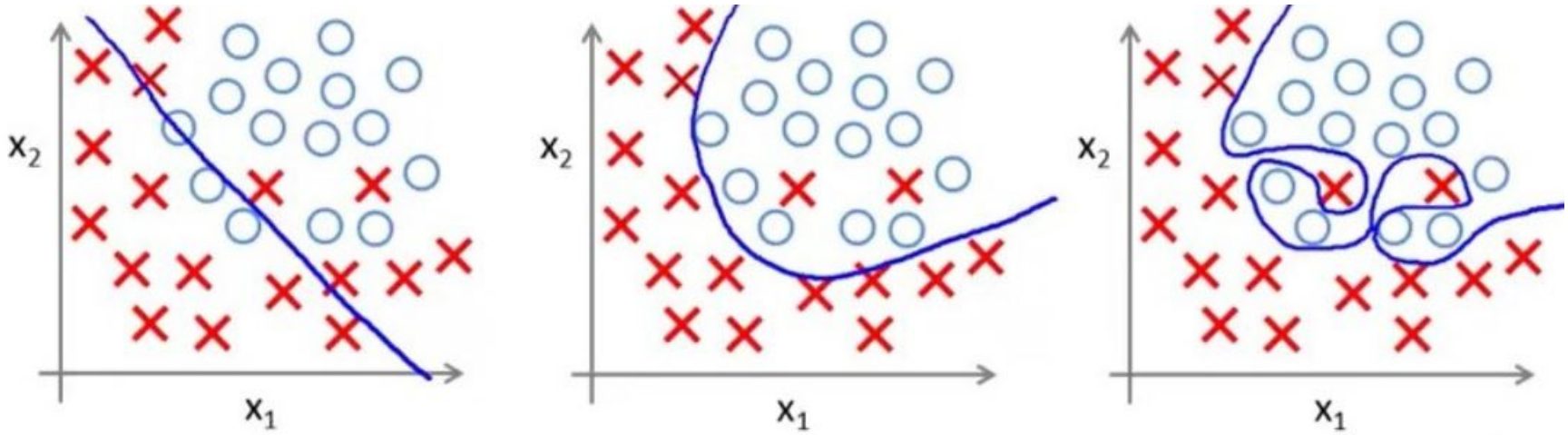


Figura. Ocorrência de *overfitting* com regressão logística. No cenário mais à direita, a flexibilidade excessiva do modelo (explorando um polinômio de ordem elevada) deu origem a contorções na fronteira na tentativa de minimizar o erro de classificação junto aos dados de treinamento. Porém, o modelo ficou mais susceptível a erros de classificação para novos dados.

- Uma alternativa popular ao algoritmo do gradiente descendente para a regressão logística está associada ao algoritmo IRLS (*iteratively reweighted least squares*) (BISHOP, 2006). Nele, informações de derivadas de 2ª ordem da função custo de entropia cruzada, contidas na matriz hessiana $\mathbf{H} \in \mathbb{R}^{(K+1)\times(K+1)}$, são incorporadas:

$$\mathbf{H} = \mathbf{\Phi}^T \mathbf{R} \mathbf{\Phi},$$

onde **R** é uma matriz diagonal com elementos $r_{i,i} = \hat{y}\big(\mathbf{x}(i)\big)\Big(1 - \hat{y}\big(\mathbf{x}(i)\big)\Big)$.

6.2. Cenário multi-classe

No caso em que existem múltiplas classes ($Q > 2$), é possível empregar as abordagens descritas na Seção 2.2, baseadas em classificadores binários. No entanto, uma estratégia mais robusta consiste em montar um modelo que produza Q saídas, em que cada saída representa a probabilidade de cada padrão pertencer a uma classe específica. Isto pode ser feito a partir de uma generalização da regressão logística, explorando a função *softmax*:

$$P(C_k|\mathbf{x}(i)) = \hat{y}_k\big(\mathbf{x}(i)\big) = \frac{e^{\left(\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}_k\right)}}{\sum_j e^{\left(\boldsymbol{\phi}(\mathbf{x}(i))^T \mathbf{w}_j\right)}}, \tag{85}$$

onde $\mathbf{w}_k = \left[w_0^{(k)} \;\; w_1^{(k)} \;\; \cdots \;\; w_K^{(k)}\right]^T$ é o vetor de coeficientes associado à k-ésima saída.

Propriedades:

- $\sum_{k=1}^{Q} \hat{y}_k\big(\mathbf{x}(i)\big) = 1$

- $0 \leq \hat{y}_k(\mathbf{x}(i)) \leq 1$ ➡ Temos, portanto, um vetor $\hat{\mathbf{y}} \in \mathbb{R}^Q$ que atende os requisitos de uma função probabilidade de massa (PMF, do inglês *probability mass function*).

Com o esquema *one-hot encoding* para representar o vetor de saída desejado, $\mathbf{y}$, a tarefa de obter a expressão da função de verossimilhança fica simplificada:

$$\mathcal{L}(\mathbf{w}|\mathbf{X}) = \prod_{i=0}^{N-1} \prod_{k=1}^{Q} P(C_k|\mathbf{x}(i))^{y_{i,k}}, \tag{86}$$

onde $y_{i,k} = \begin{cases} 1, \text{se } \mathbf{x}(i) \in C_k \\ 0, \text{caso contrário} \end{cases}$.

Neste caso, a entropia cruzada é dada pela seguinte expressão:

$$J_{\text{CE}}(\mathbf{W}) = -\sum_{i=0}^{N-1} \sum_{k=1}^{Q} y_{n,k} \log \hat{y}_k(\mathbf{x}(i)). \tag{87}$$

- A derivada de $J_{\text{CE}}(\mathbf{W})$ com respeito a cada vetor de coeficientes de saída $\mathbf{w}_k$ segue uma expressão semelhante àquela obtida na seção anterior:

$$\frac{\partial J_{\mathrm{CE}}(\mathbf{W})}{\partial \mathbf{w}_k} = \sum_{i=0}^{N-1} \left(y_{i,k} - \hat{y}_k\left(\mathbf{x}(i)\right)\right) \boldsymbol{\phi}\left(\mathbf{x}(i)\right)^T \tag{88}$$

Exemplo:

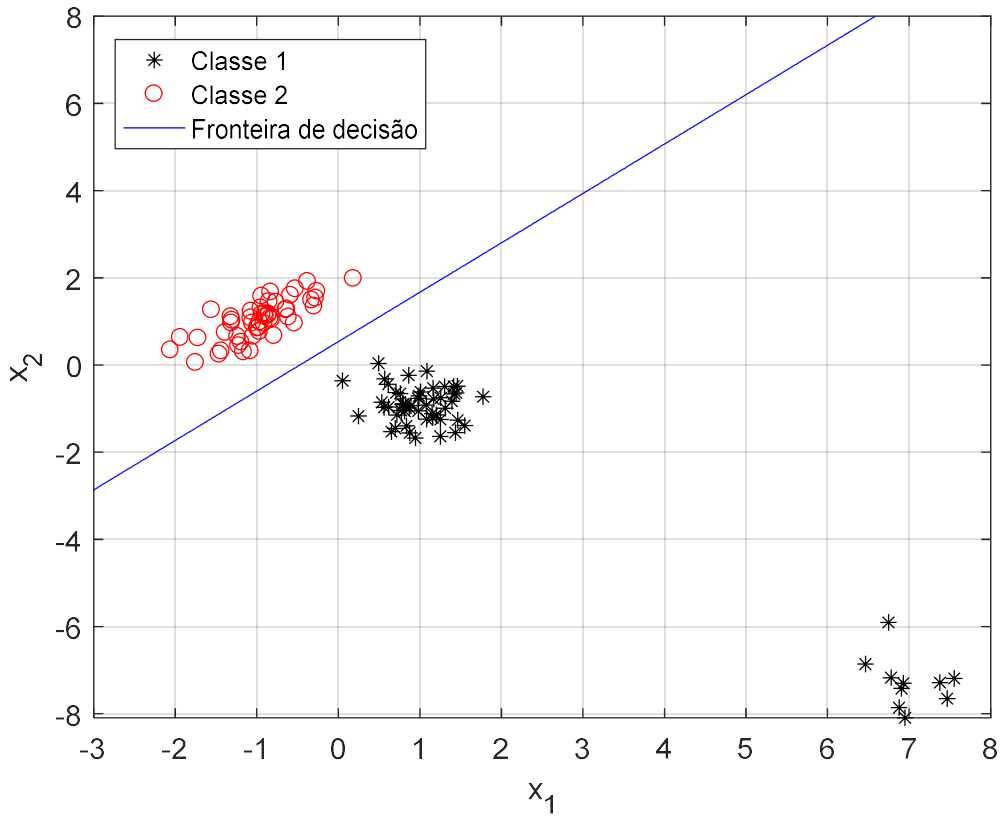


Figura. Fronteira de decisão obtida com regressão logística. O contraste com a solução do discriminante linear via quadrados mínimos é bem evidente: como o critério, agora, avalia o erro de classificação, a presença de *outliers* não afeta significativamente os parâmetros.

Exemplo:

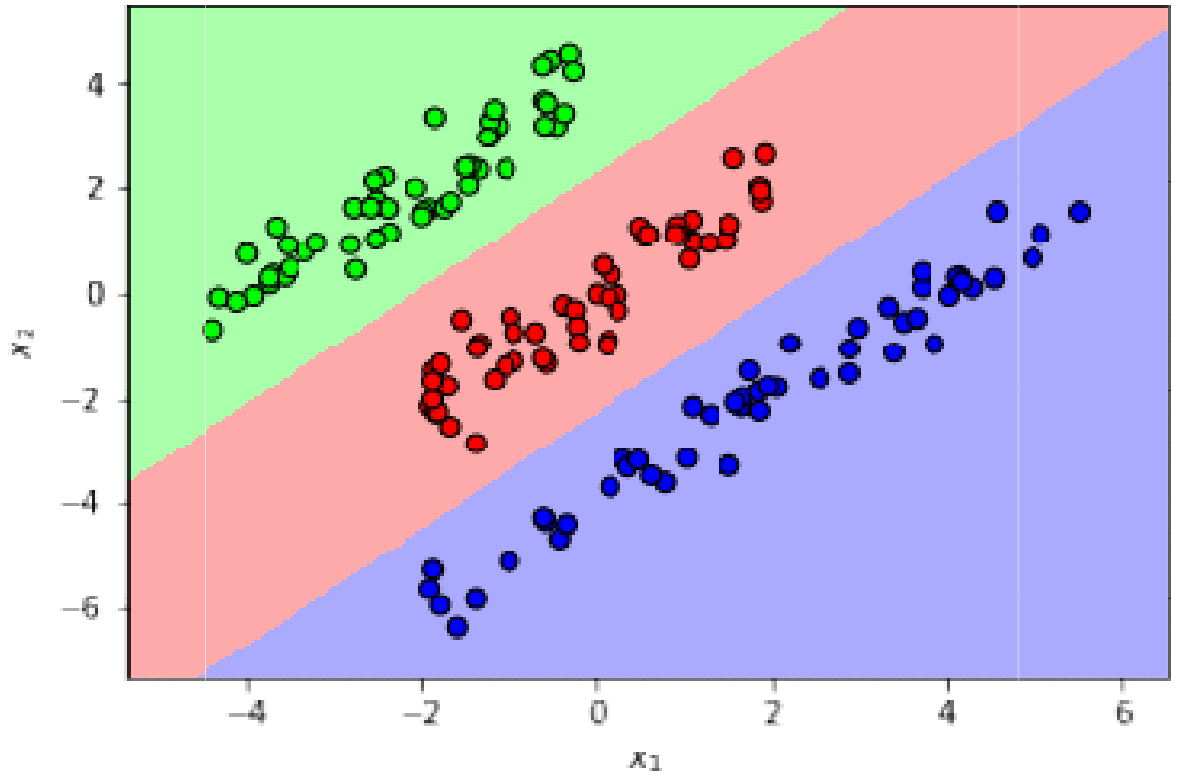


Figura. Fronteira de decisão obtida com regressão logística para um problema com três classes linearmente separáveis.

7. Métricas de avaliação

7.1. Taxa de erro e acurácia

Intuitivamente, a métrica mais direta para se avaliar o desempenho de um classificador é dada pela **taxa de erro**, que corresponde à porcentagem de padrões classificados incorretamente considerando o conjunto de dados disponíveis para teste:

$$p_e\big(\hat{y}(\mathbf{x})\big) = \frac{1}{N}\sum_{i=0}^{N-1}\big(1 - \delta_{y(i),\hat{y}(\mathbf{x}(i))}\big), \tag{89}$$

onde $\delta_{i,j} = \begin{cases} 0, \text{se } i \neq j \\ 1, \text{se } i = j \end{cases}$ é o delta de Kronecker. Note que $p_e\big(\hat{y}(\mathbf{x})\big) \in [0,1]$.

Seu complemento é conhecido como **acurácia**:

$$\text{acc}\big(\hat{y}(\mathbf{x})\big) = 1 - p_e\big(\hat{y}(\mathbf{x})\big) \tag{90}$$

7.2. Matriz de confusão

A matriz de confusão $\mathbf{C} \in \mathbb{R}^{Q \times Q}$ contabiliza o número de classificações corretas e incorretas para cada uma das Q classes existentes: o elemento c_{ij} indica quantos padrões da classe i foram designados à classe j. Em sua diagonal, portanto, encontramos o número de classificações corretas.

A informação apresentada nesta matriz nos permite ver quais classes o algoritmo tem maior dificuldade em classificar.

		Classe estimada	
		+	−
Classe verdadeira	+	Verdadeiro positivo (TP)	Falso negativo (FN)
	−	Falso positivo (FP)	Verdadeiro negativo (TN)

- *True positive* (TP): número de exemplos da classe positiva classificados corretamente.

- *True negative* (TN): número de exemplos da classe negativa identificados corretamente.
- *False positive* (FP): exemplos classificados como positivos (+), mas que na realidade pertencem à classe negativa (−).
- *False negative* (FN): exemplos atribuídos à classe negativa (−), mas que, na verdade, pertencem à classe positiva (+).

Note que:

N_+ = número de padrões pertencentes à classe positiva = $\mathrm{TP} + \mathrm{FN}$

N_- = número de padrões pertencentes à classe negativa = $\mathrm{TN} + \mathrm{FP}$

N = número total de padrões = $\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}$

A partir das informações contidas na matriz de confusão, podemos computar diversas métricas de desempenho (SOKOLOVA & LAPALME; 2009):

- Taxa de falso negativo: proporção de exemplos da classe positiva (+) classificados incorretamente.

$$\text{Taxa de falso negativo} = p_e^+\big(\hat{y}(\mathbf{x})\big) = \frac{\text{FN}}{\text{TP} + \text{FN}} = \frac{\text{FN}}{N_+} \tag{91}$$

- Taxa de falso positivo: proporção de exemplos da classe negativa (−) classificados incorretamente.

$$\text{Taxa de falso positivo} = p_e^-\big(\hat{y}(\mathbf{x})\big) = \frac{\text{FP}}{\text{TN} + \text{FP}} = \frac{\text{FP}}{N_-} \tag{92}$$

- Taxa de erro:

$$p_e\big(\hat{y}(\mathbf{x})\big) = \frac{\text{FP} + \text{FN}}{N} \tag{93}$$

- Acurácia:

$$\text{acc}\big(\hat{y}(\mathbf{x})\big) = \frac{\text{TP} + \text{TN}}{N} \tag{94}$$

7.3. Precisão

Corresponde à proporção de padrões da classe positiva corretamente classificados em relação a todos os exemplos atribuídos à classe positiva.

$$\text{Precisão}\left(\hat{y}(\mathbf{x})\right) = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{95}$$

7.4. Sensibilidade (recall)

Também conhecida como taxa de verdadeiro positivo, a sensibilidade corresponde à proporção de amostras da classe positiva corretamente classificadas.

$$\text{Recall}\left(\hat{y}(\mathbf{x})\right) = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{96}$$

7.5. Especificidade

Também conhecida como taxa de verdadeiros negativos, a especificidade é dada pela proporção de amostras da classe negativa corretamente classificadas.

$$\text{Especificidade}\left(\hat{y}(\mathbf{x})\right) = \frac{\text{TN}}{\text{TN} + \text{FP}} = 1 - p_e^-\left(\hat{y}(\mathbf{x})\right) \tag{97}$$

Observações:

- É possível estender de maneira natural estas métricas para o cenário multi-classe: para isto, basta tomar, uma vez, cada classe $C_k, k = 1, \ldots, Q$ como sendo a classe positiva, enquanto todas as demais classes formam a classe negativa; assim, obtemos os valores das métricas para cada classe.
- A acurácia global pode ser obtida a partir da informação presente na diagonal da matriz de confusão no cenário multi-classe.
- A precisão pode ser vista como uma medida da qualidade de um modelo, enquanto a sensibilidade (recall) dá uma noção de sua completude.
  - Um valor de precisão $= 1$ significa que, para uma determinada classe, cada padrão classificado como sendo pertencente a esta classe realmente pertence a

ela. Entretanto, isso não dá informações a respeito de quantas amostras desta classe foram classificadas de forma incorreta (falso negativo).

- Por outro lado, um valor de recall = 1 indica que todos os exemplos da classe C_k foram classificados como sendo pertencentes a C_k. Porém, isso não traz informações a respeito de quantos padrões associados a outras classes foram classificados como sendo pertencentes a C_k (falso positivo).

7.6. F-medida

Uma vez que precisão e recall costumam ser analisados juntos, existe uma métrica única, denominada **F-medida** (ou F-*score*), aqui denotada por F_m, que combina as duas informações através de uma média harmônica ponderada:

$$F_{\mathrm{m}} = \frac{(m+1) \times \mathrm{recall}\big(\hat{y}(\mathbf{x})\big) \times \mathrm{precisão}\big(\hat{y}(\mathbf{x})\big)}{\mathrm{recall}\big(\hat{y}(\mathbf{x})\big) + m \times \mathrm{precisão}\big(\hat{y}(\mathbf{x})\big)}, \tag{98}$$

onde m é o fator de ponderação.

Quando $m = 1$, a mesma importância é dada para a precisão e para o recall:

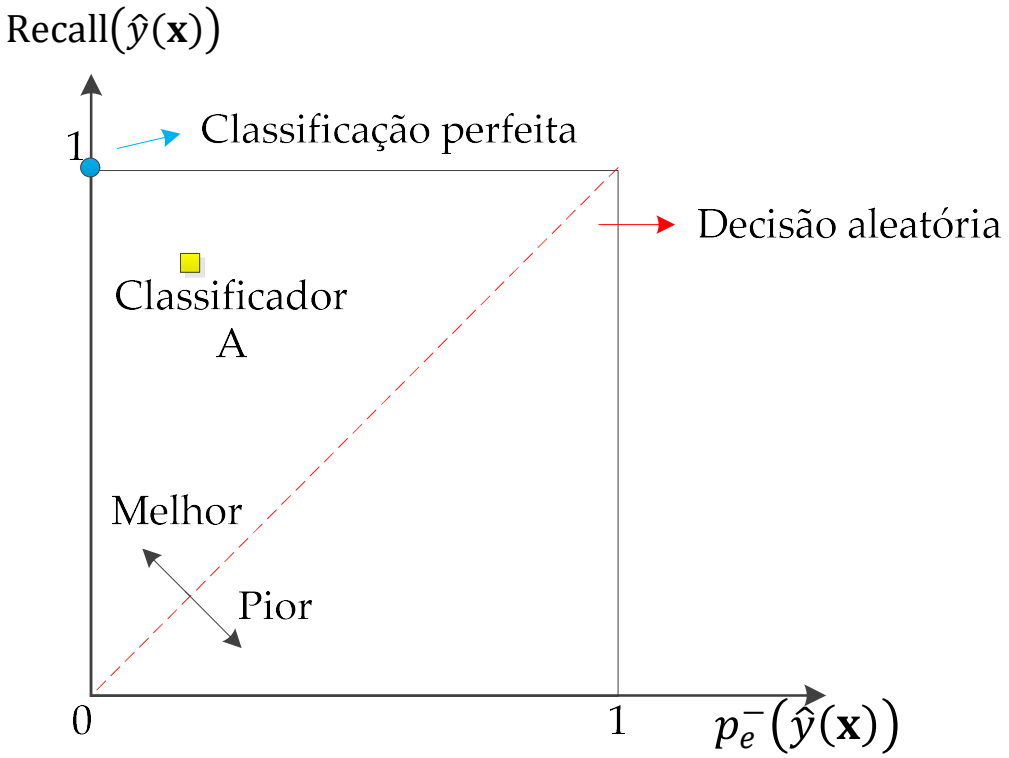
$$F_1 = 2\frac{\text{recall}\big(\hat{y}(\mathbf{x})\big) \times \text{precisão}\big(\hat{y}(\mathbf{x})\big)}{\text{recall}\big(\hat{y}(\mathbf{x})\big) + \text{precisão}\big(\hat{y}(\mathbf{x})\big)} \tag{99}$$

Valores de F_1 próximos de 1 indicam que o classificador obteve bons resultados tanto na precisão quanto no recall.

7.7. *Receiver operating curve* (ROC)

Trata-se de um gráfico em que a taxa de verdadeiro positivo, a qual equivale ao recall, é exibida em função da taxa de falso positivo.

Quanto mais à esquerda e para cima estiver um classificador, melhor será o seu desempenho. A linha diagonal, por sua vez, está associada a um classificador aleatório.



A forma usual de se comparar classificadores consiste em criar uma **curva ROC**. Vários tipos de classificadores produzem uma saída real para cada padrão de entrada. Normalmente, estas saídas são, então, discretizadas para que se tenha a decisão final: por exemplo, se $\hat{y}(\mathbf{x})$ ultrapassa um limiar (*threshold*), ela é mapeada no valor 1 (classe positiva); caso contrário, ela é mapeada no valor 0 (classe

negativa). Sendo assim, ao plotar a taxa de verdadeiro positivo (recall) versus a taxa de falso positivo para diferentes valores de *threshold*, obtemos a curva ROC associada a um classificador.

Por exemplo, considere as curvas ROC exibidas na figura a seguir. Para decidir qual o melhor classificador, podemos tomar como base **a área sob a curva ROC**.

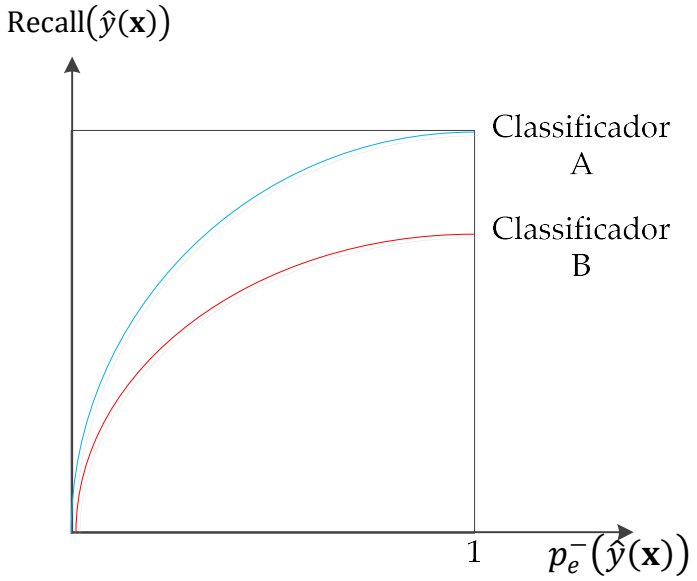


Figura. Curvas ROC referentes a dois classificadores. A área sob a curva ROC é uma medida da qualidade do classificador.

No caso, o classificador A seria o de melhor desempenho.

Vantagens:

- Possibilita a análise de diferentes métricas de desempenho independente do *threshold* escolhido.
- Auxilia o estudo de diferentes *thresholds* para lidar com problemas de desbalanceamento nos dados (*i.e.*, nos quais as classes possuem tamanhos discrepantes).

Desvantagens:

- Apropriada para problemas de classificação binária.
  - No cenário multi-classe, podemos explorar uma estratégia um-contra-todos e utilizar várias curvas ROC.

8. Referências bibliográficas

ALPAYDIN, E. **Introduction to Machine Learning**. MIT Press. 3rd edition. 2014.

BISHOP, C. M. **Pattern Recognition and Machine Learning**. Springer. 2006.

DUDA, R. O., HART, P. E., STORK, D. G. **Pattern Classification**. John Wiley & Sons. 2nd edition, 2001.

FORNACIALI, M., AVILA, S. CARVALHO, M., VALLE, E., Statistical Learning Approach for Robust Melanoma Screening, *SIBGRAPI*, 2014.

HASTIE, T., TIBSHIRANI, R., FRIEDMAN, J. **The Elements of Statistical Learning: Data Mining, Inference and Prediction**. Springer. 2nd edition, 2009.

SOKOLOVA, M., LAPALME, G. A Systematic Analysis of Performance Measures for Classification Tasks. *Information Processing & Management*, vol. 45, no. 4, pp. 427-437, 2009.

https://math.stackexchange.com/questions/477207/derivative-of-cost-function-for-logistic-regression