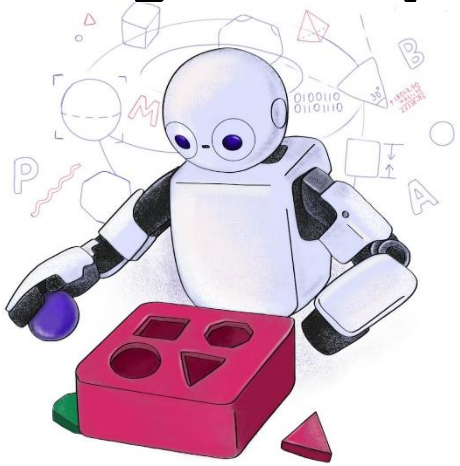
T320 - Introdução ao Aprendizado de Máquina II: *Classificação (Parte II)*



Inatel

Felipe Augusto Pereira de Figueiredo
felipe.figueiredo@inatel.br

Recapitulando

- Anteriormente, vimos exemplos de uso de algoritmos de ***classificação***.
  - Detecção de spams.
  - Análise de sentimentos.
  - Reconhecimento de dígitos.
- Definimos o problema da classificação e concluímos que ele também é um problema de ***aprendizado supervisionado***.
- Aprendemos que as classes são separadas através de ***funções discriminantes*** e que o desafio é encontrar uma função adequada e os pesos correspondentes.
- A partir da aula de hoje, começamos a discutir como encontrar os pesos.

Classificação linear

- Como vimos, o objetivo da ***classificação*** é usar as características (i.e., atributos) de, por exemplo, um objeto para identificar a qual classe ele pertence.
- Um ***classificador linear*** atinge esse objetivo tomando uma decisão de classificação com base no valor de uma ***combinação linear*** dos ***atributos***, ou seja, na saída de uma ***função discriminante linear***.
- A saída de um ***classificador linear*** é dada por

$$y = h_{\boldsymbol{a}}(\boldsymbol{x}) = f(g(\boldsymbol{x})) = f(a_0 + \cdots + a_K x_K) = f\left(\sum_{k=0}^{K} a_k x_k\right) = f(\boldsymbol{a}^T \boldsymbol{x}),$$

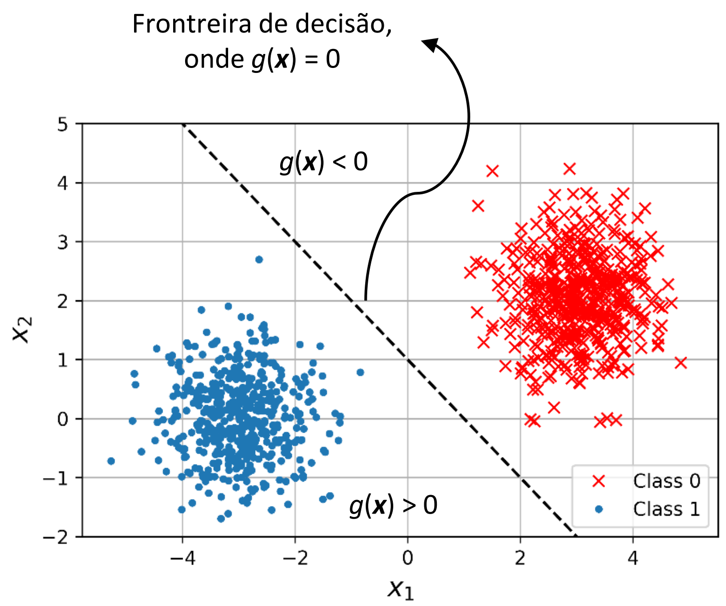
onde $h_{\boldsymbol{a}}(\boldsymbol{x})$ é conhecida como ***função hipótese de classificação***, $\boldsymbol{x} = [1, x_1, \ldots, x_K]^T$ e $f(.)$ é uma ***função de limiar de decisão***.

  - ***Função de limiar de decisão*** é uma função que converte a saída da ***função discriminante linear***, $\boldsymbol{a}^T \boldsymbol{x}$ (produto escalar), na saída desejada, ou seja, na classe $C_q,\ q = 1, \ldots, Q$, do objeto.
  - Ela é apenas uma formalização matemática para os ***if***s e ***else***s que usamos para definir as classes.

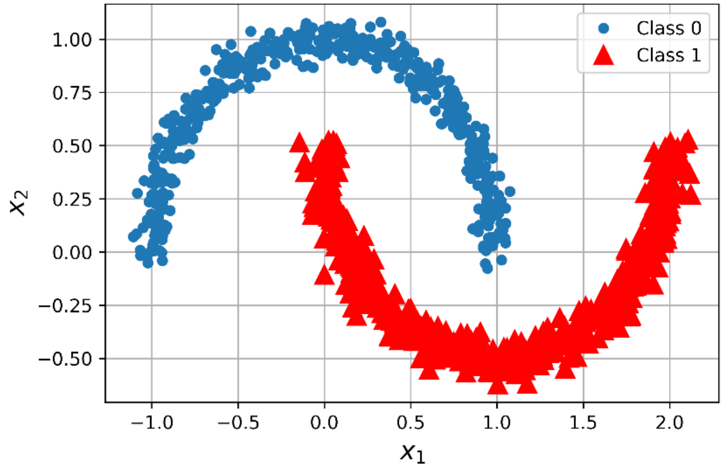
- Originalmente, as ***funções discriminantes*** são formadas por equações de hiperplanos: $\sum_{k=0}^{K} a_k x_k$.

Classificação linear

- Dado um ***conjunto de treinamento***, a tarefa do ***classificador*** é a de ***aprender*** uma ***função hipótese de classificação***, $h_a(\boldsymbol{x})$, que recebe um exemplo (e.g., x_1 e x_2) e retorna a classe do exemplo.
- ***Classificadores binários*** têm como saída o valor ***0*** caso o exemplo pertença à classe C_1 (***classe negativa***) ou ***1*** caso ele pertença à classe C_2 (***classe positiva***).
- Para que um ***classificador linear*** funcione corretamente, as duas classes devem ser ***linearmente separáveis***.
- Isso significa que as classes devem ser ***suficientemente separadas*** umas das outras para garantir que a ***superfície de decisão*** consista de um ***hiperplano***.
- Classes que podem ser separadas por um ***hiperplano*** são chamadas de ***linearmente separáveis***.
- Na primeira figura, a ***fronteira de decisão*** é definida por uma ***função discriminante*** que é uma **reta**: $g(\boldsymbol{x}) = 1 - x_1 - x_2$.
- Na segunda figura, devido à proximidade das classes, não existe um ***hiperplano*** que as separe.



Classes linearmente separáveis.



Classes não-linearmente separáveis.

Classificação não-linear

- Originalmente, ***classificação linear*** é usada quando as classes podem ser separadas por ***superfícies de decisão lineares***.
- Ou seja, as ***funções discriminantes*** são ***hiperplanos***: $\sum_{k=0}^{K} a_k x_k$.
- Mas e se não pudermos separar as classes com um ***hiperplano***, ou seja, se elas não forem ***linearmente separáveis***?
- Nestes casos, usamos ***funções discriminantes não-lineares***, como, por exemplo, ***polinômios***:
  - $g(\boldsymbol{x}) = (x_1 - a)^2 + (x_2 - b)^2 - r^2$, Círculo centrado em (a, b) e com raio r.
  - $g(\boldsymbol{x}) = \frac{(x_1-a)^2}{c^2} + \frac{(x_2-b)^2}{d^2} - 1$, Elipse centrada em (a, b), com largura $2c$ e altura $2d$.
  - $g(\boldsymbol{x}) = (x_1 - a)(x_2 - b) - c$, Hipérbole retangular com eixos paralelos às suas assíntotas.
- Portanto, quando usamos uma ***função discriminante não-linear***, convertemos ***classificadores lineares*** em ***classificadores não-lineares*** através de uma ***transformação dos atributos***.
  - $g(\boldsymbol{x}) = (x_1 - a)^2 + (x_2 - b)^2 - r^2 = x_1^2 - 2ax_1 + x_2^2 - 2bx_2 + (a^2 + b^2 - r^2)$
- Esta transformação pode ser vista também como uma mudança do ***espaço de entrada***, o que normalmente leva ao ***aumento das dimensões de entrada*** ou ***mudança dos eixos***.

Limiar de decisão rígido

- Para o exemplo ao lado, podemos definir a ***função hipótese de classificação*** como

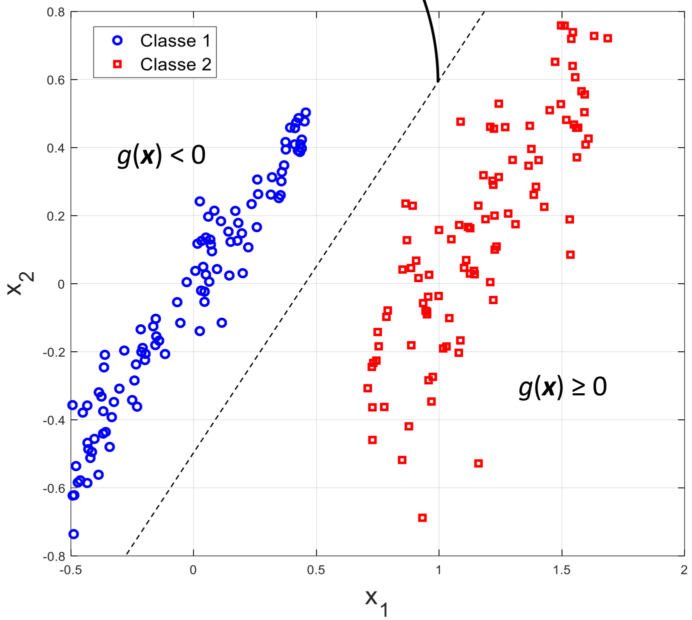
$$y = h_{\boldsymbol{a}}(\boldsymbol{x}) = \begin{cases} 0, & g(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{a} < 0 \text{ (Classe 1)} \\ 1, & g(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{a} \geq 0 \text{ (Classe 2)} \end{cases}$$

- Percebam que a saída da ***função hipótese*** é ***binária***, ou seja, temos apenas 2 possíveis valores, 0 ou 1.
- O mapeamento entre o valor da função discriminante, $g(\boldsymbol{x})$, e a saída 0 ou 1 é feito através da ***função de limiar de decisão***, $f(g(\boldsymbol{x}))$.
- Uma ***função de limiar de decisão*** que faça o mapeamento do valor de $g(\boldsymbol{x})$ em apenas 2 valores é chamada de ***função de limiar de decisão rígido***.
- A ***função de limiar de decisão rígido*** é mostrada na figura ao lado e é definda como

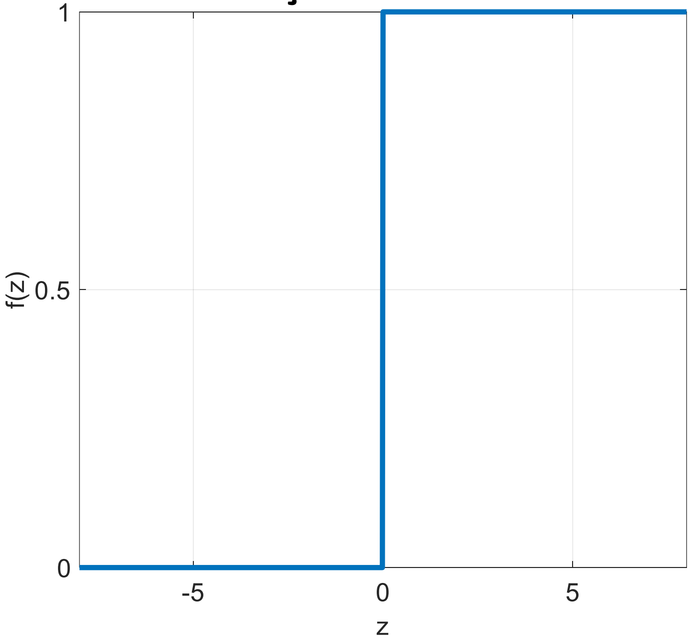
Conhecida também como ***função heaviside.***

$$f(z) = \begin{cases} 0, & z < 0 \\ 1, & z > 0 \\ \text{Indeterminado}, & z = 0 \end{cases}$$

Frontreira de decisão, onde $g(\boldsymbol{x}) = 0$

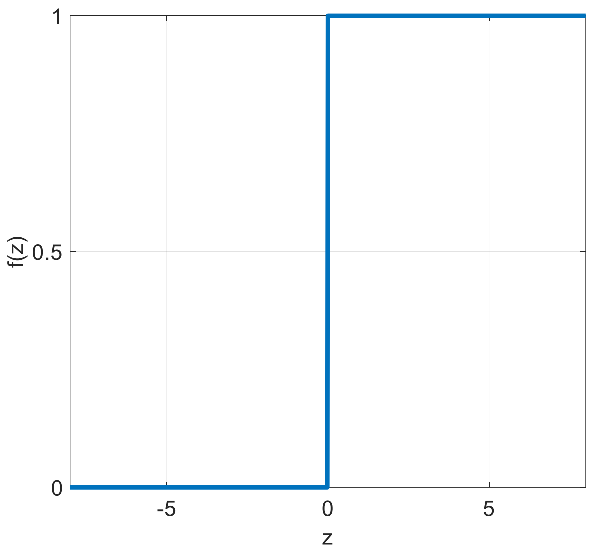


Função heaviside



Classificação com limiar de decisão rígido

- Agora que a ***função hipótese***, $h_{\boldsymbol{a}}(\boldsymbol{x})$, tem uma forma matemática bem definida, precisamos pensar em como encontrar os pesos, $\boldsymbol{a}$.
- Nós queremos encontrá-los de tal forma que ***o erro de classificação seja minimizado***, ou seja, que os exemplos sejam atribuídos às suas respectivas classes.
- No caso da ***regressão linear***, nós fizemos isso de duas maneiras:
  i. de forma fechada (através da ***equação normal***) fazendo a derivada parcial com relação aos pesos igual a zero e resolvendo a equação para os pesos;
  ii. e através do algoritmo do ***gradiente descendente***.
- Entretanto, com a ***função de limiar rígido***, nenhuma das duas abordagens é possível devido a ***derivada*** de $f(g(\boldsymbol{x}))$ ser igual a zero em todos os pontos exceto em $g(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{a} = 0$, onde ela é indeterminada.
- **Portanto, o que podemos fazer?**



Classificação com limiar de decisão rígido

- Uma possível abordagem para o problema da aprendizagem quando utilizamos um ***limiar de decisão rígido*** é utilizar uma ***regra intuitiva*** de atualização dos ***pesos*** que converge para uma solução ***dado que exista uma função discriminante adequada e que as classes não se sobreponham***.
- A atualização dos ***pesos*** é dada pela seguinte equação

Os pesos são atualizados a cada novo exemplo, ou seja, amostra a amostra.

$$\boldsymbol{a} = \boldsymbol{a} + \alpha\left(y(i) - h_{\boldsymbol{a}}\left(\boldsymbol{x}(i)\right)\right)\boldsymbol{x}(i), \forall \mathrm{i},$$

a qual é essencialmente idêntica à regra de atualização para a ***regressão linear*** quando utilizamos o ***gradiente descendente estocástico***, α onde é o passo de aprendizagem.

- Esta regra é chamada de ***regra de aprendizagem do perceptron***, por razões que discutiremos em breve.
- Essa regra de aprendizagem é aplicada a um exemplo por vez, escolhendo exemplos aleatóriamente, assim como fizemos com o ***gradiente descendente estocástico***.
- Como estamos considerando classificadores com valores de saída iguais a 0 ou 1, o comportamento da regra de atualização será diferente do comportamento para a regressão linear, como veremos a seguir.

Classificação com limiar de decisão rígido

- Observem a equação de atualizaçao dos pesos

$$\boldsymbol{a} = \boldsymbol{a} + \alpha\left(y(i) - h_{\boldsymbol{a}}(\boldsymbol{x}(i))\right)\boldsymbol{x}(i).$$

- Ambos, o valor esperado, y, e a saída da ***função hipótese***, $h_{\boldsymbol{a}}(\boldsymbol{x})$, assumem os valores 0 ou 1, portanto, existem 3 possibilidades:
  - Se a saída estiver correta, i.e., $y = h_{\boldsymbol{a}}(\boldsymbol{x})$, então os pesos não são atualizados.
  - Se $y = 1$, mas $h_{\boldsymbol{a}}(\boldsymbol{x}) = 0$, então o peso a_k tem seu valor ***aumentado*** quando o valor de x_k é positivo e ***diminuído*** quando o valor de x_k é negativo.
    - Isso faz sentido pois nós queremos aumentar o valor do produto escalar $\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{a}$, ou seja, $g(\boldsymbol{x})$, de tal forma que $h_{\boldsymbol{a}}(\boldsymbol{x})$ tenha como saída o valor 1.
  - Se $y = 0$, mas $h_{\boldsymbol{a}}(\boldsymbol{x}) = 1$, então o peso a_k tem seu valor ***diminuido*** quando o valor de x_k é positivo e ***aumentado*** quando o valor de x_k é negativo.
    - Isso faz sentido pois nós queremos diminuir o valor do produto escalar $\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{a}$, ou seja, $g(\boldsymbol{x})$, de tal forma que $h_{\boldsymbol{a}}(\boldsymbol{x})$ tenha como saída o valor 0.

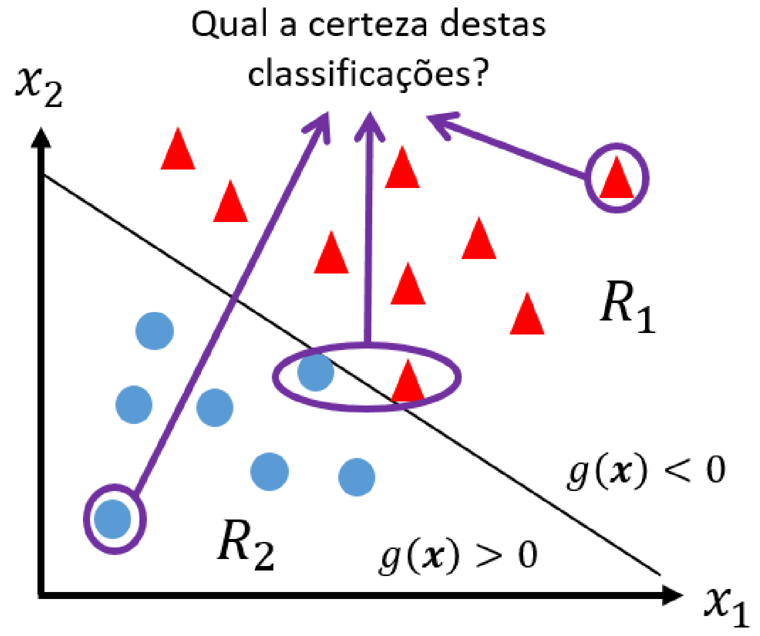
Classificação com limiar de decisão rígido

- A ***regra de aprendizagem do perceptron*** converge para um ***separador perfeito*** quando:
  - As classes são ***suficientemente separadas*** umas das outras, ou seja, não se sobrepõem.
  - Existe uma ***função discriminante adequada para o problema***, mesmo que não seja um ***hiperplano***.
- ***Separador perfeito:*** com erro de classificação igual a zero, ou seja, todos os exemplos são perfeitamente classificados.
- Porém, na prática essa situação não é muito comum.
- Nesse caso, a ***regra de aprendizagem do perceptron*** falha em convergir para uma solução perfeita.
- Em geral, essa regra não converge para uma solução estável para valores fixos do ***passo de aprendizagem***, α, mas se α decresce de acordo com as iterações, então a regra tem uma chance de convergir para uma solução de erro mínimo quando os exemplos são apresentados de forma aleatória.
- Podemos também usar o ***early-stop*** e utilizar os ***pesos*** que resultaram no menor erro de validação.

[Exemplo: classificador_linear_com_limiar_rigido.ipynb]

Classificação com limiar de decisão rígido

- Outro problema com classificadores que usam ***limiar de decisão rígido*** é a falta de informação sobre a confiança do classificador quanto a um resultado.
- No exemplo ao lado, dois exemplos estão bem próximos da ***fronteira de decisão*** enquanto outros dois estão bem distantes dela.
- O classificador com ***limiar rígido***, faria uma previsão completamente confiante pelo valor 1 para os dois pontos azuis e 0 para os dois triângulos vermelhos, mesmo eles tendo valores bem diferentes de $g(\boldsymbol{x})$.
- Em muitas situações, nós precisamos de previsões mais graduadas, que indiquem incertezas quanto à classificação.

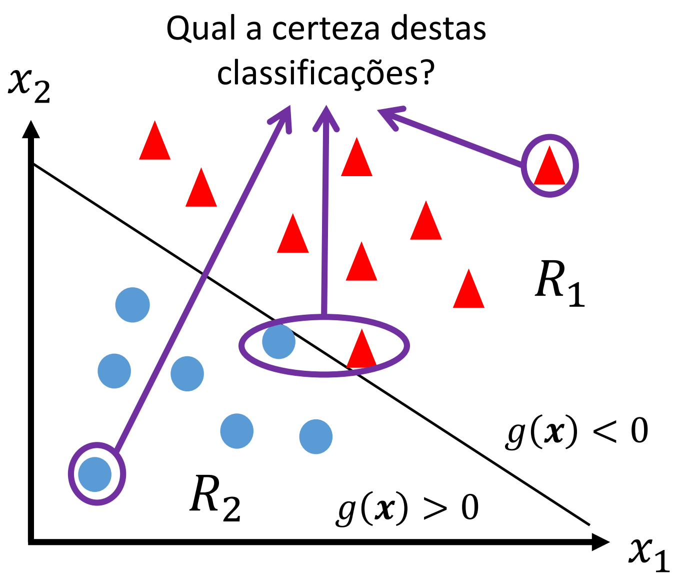


- Os pontos distantes da ***fronteira de decisão*** têm valores absolutos de $g(\boldsymbol{x})$ bem maiores do que os dos pontos próximos, os quais têm valores de $g(\boldsymbol{x})$ muito próximos de 0.
- Ou seja, a confiança deveria ser maior para pontos distantes da fronteira.
- Porém, isso não é refletido na saída do classificador com limiar rígido.

Tarefas

- **Quiz**: “*T320 - Quiz - Classificação (Parte II)*” que se encontra no MS Teams.
- **Exercício Prático**: **Laboratório #2**.
  - Pode ser baixado do MS Teams ou do GitHub.
  - Pode ser respondido através do link acima (na nuvem) ou localmente.
  - Instruções para resolução e entrega dos laboratórios.
  - **Atividades podem ser feitas em grupo, mas as entregas devem ser individuais.**

Obrigado!


Qual a certeza destas classificações?
x_2
R_1
$g(\boldsymbol{x}) < 0$
R_2
$g(\boldsymbol{x}) > 0$
x_1