PMR5251 - Assessment of Mechanical Behavior of Materials using Machine Learning Approach

UNIVERSIDADE DE SÃO PAULO • ESCOLA POLITÉCNICA •

REGRESSÃO LOGÍSTICA

Izabel F. Machado
Larissa Driemeier

SCHEDULE

Date	Contents	Prof.
19/06	Introduction to Machine Learning main concepts	Larissa
26/06	Review mechanical behavior of materials and microstructure characterization	Izabel
03/07	Techniques to evaluate and quantify microstructure features	Izabel
10/07	Neural Networks - theory	Larissa
24/07	Neural Networks – application in structural analyses	Larissa
31/07	Mechanical behavior, Multiscale analysis	Izabel
07/08	Regression	Larissa
14/08	Damage and Failure analysis	Izabel
21/08	Classification	Larissa
28/08 04/09	Seminars	Izabel/Larissa

AULA DE HOJE

- Regressão Logística
- Exemplo de aplicação
- Classificação usando redes neurais

REGRESSÃO LOGÍSTICA

O que é isso?

TÉCNICA DE REGRESSÃO LOGÍSTICA

E se,

- um engenheiro deseja saber se ocorrerá ou não a falha de uma estrutura, em função do estado de tensões, material e geometria?
- um médico deseja investigar se a probabilidade de ataque cardíaco pode ser predita em função de características sanguíneas, sexo, estilo de vida?
- uma operadora de telefonia móvel deseja saber a probabilidade de mudança de plano por parte dos clientes que compõem sua carteira, em função de características como nível de escolaridade, renda, estado civil, número de filhos, tempo de relacionamento dom a operadora?
- um pesquisador quer saber se o passo de seu exoesqueleto é estável, a partir dos movimentos das juntas?
- ...

REGRESSÃO VS CLASSIFICAÇÃO

Um problema de regressão tem um número real como saída.

Por exemplo, podemos usar os dados da tabela para estimar o peso de alguém de acordo com sua altura.

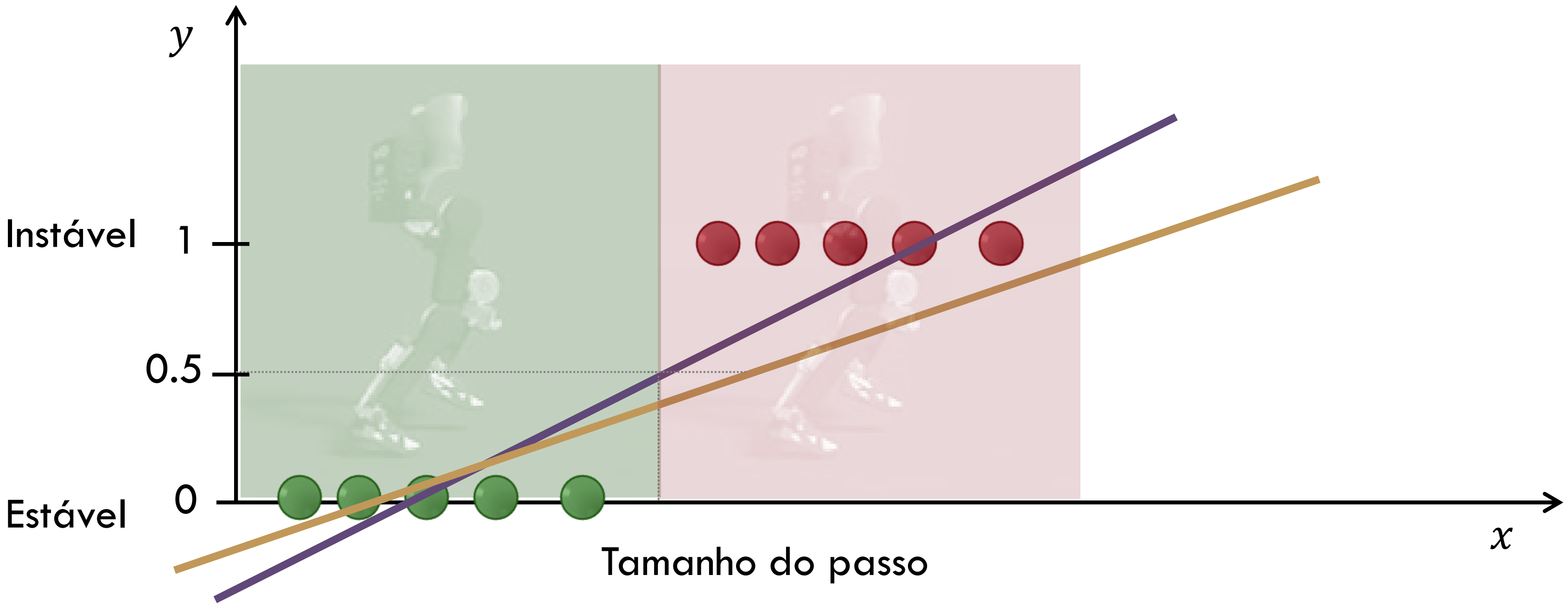
Altura (cm)	Peso(kg)
167,1	51,3
181,7	61,9
176,3	69,4
173,3	64,6
172,2	65,5
174,5	55,9
177,3	64,2
177,8	61,9
172,5	51,0
169,6	54,7
168,9	57,8
171,8	51,8
173,5	57,0
170,5	55,5
173,4	52,7

Um problema de classificação tem um valor discreto como saída.

Por exemplo, "gosta de abacaxi na pizza" e "não gosta de abacaxi na pizza" são opções discretas. Não há meio termo.

Idade	Gosta de abacaxi na pizza
42	1
65	1
50	1
76	1
96	1
50	1
91	0
58	1
25	1
23	1
75	1
46	0
87	0
96	0
45	0
32	1
63	0
21	1
26	1
93	0
68	1
96	0

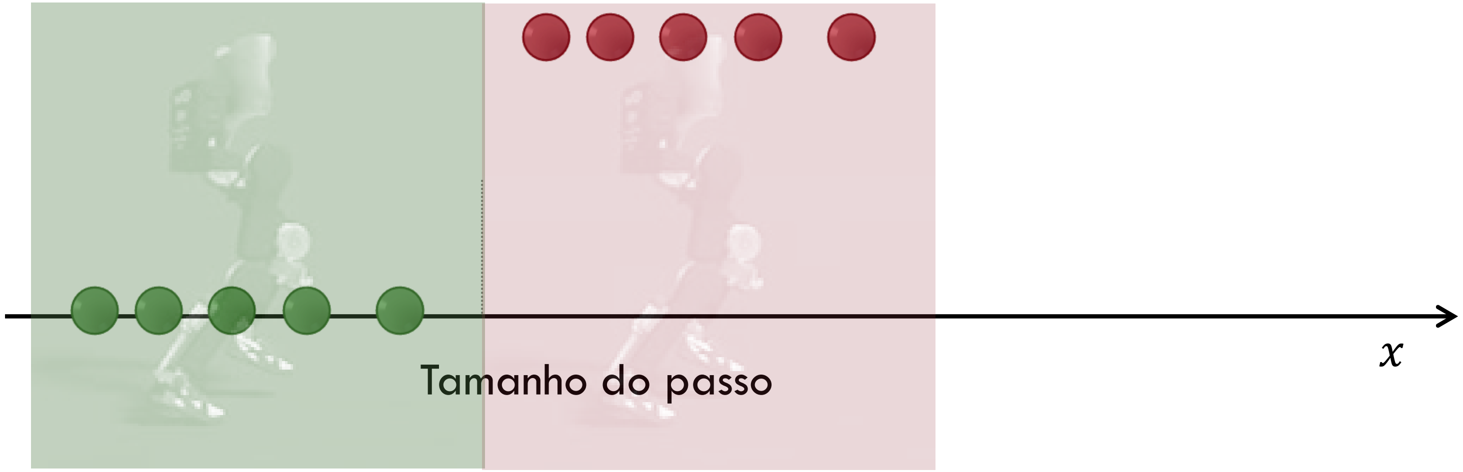
ESTABILIDADE DE MARCHA



O resultado, y_i, assume o valor 1 (em nosso caso, isso representa uma marcha instável) com probabilidade p_i e o valor 0 com probabilidade $1 - p_i$.

É a probabilidade p_i que modelamos em relação às variáveis independentes.

ESTABILIDADE DE MARCHA



$$h(\boldsymbol{x}) = \hat{y} = \boldsymbol{\omega}^T \boldsymbol{x}$$

$$(p_i) = \omega_0 + \omega_1 x_1^{(i)} + \omega_2 x_2^{(i)} + \cdots + \omega_n x_n^{(i)}$$

transformação

Podemos formular o problema da seguinte forma: “**o passo é instável?**” ou, melhor ainda, “**qual a probabilidade do passo ser instável?**”. Teoricamente, **passos instáveis** deveriam ter uma probabilidade de **1.0** (de serem instáveis), ao passo que **passos estáveis** deveriam ter uma probabilidade de **0.0** (de serem instáveis).

Assim, **passos instáveis** pertencem **à classe positiva (SIM, 1,** eles são instáveis)**, e passos estáveis pertencem à classe negativa (NÃO, 0,** eles não são instáveis).

RAZÃO DE CHANCES (ODDS RATIO)

A razão de chances descreve a relação entre a probabilidade de sucesso e de falha. Se um evento ocorre com probabilidade p, a razão de chances deste evento é

$$OR = \frac{p}{1-p}$$

para 1.

	Não compra	Compra	Total
Mulher	106	159	265
Homem	125	121	246

$$OR_M = \frac{159/265}{106/265} = 1.5$$

$$OR_H = \frac{121/265}{125/265} = 0.968$$

Quanto maior OR, melhor é a chance de sucesso.

Esse valor pode variar de 0 a ∞.

$$OR = \omega_0 + \omega_1 x_1^{(i)} + \omega_2 x_2^{(i)} + \cdots + \omega_n x_n^{(i)}$$

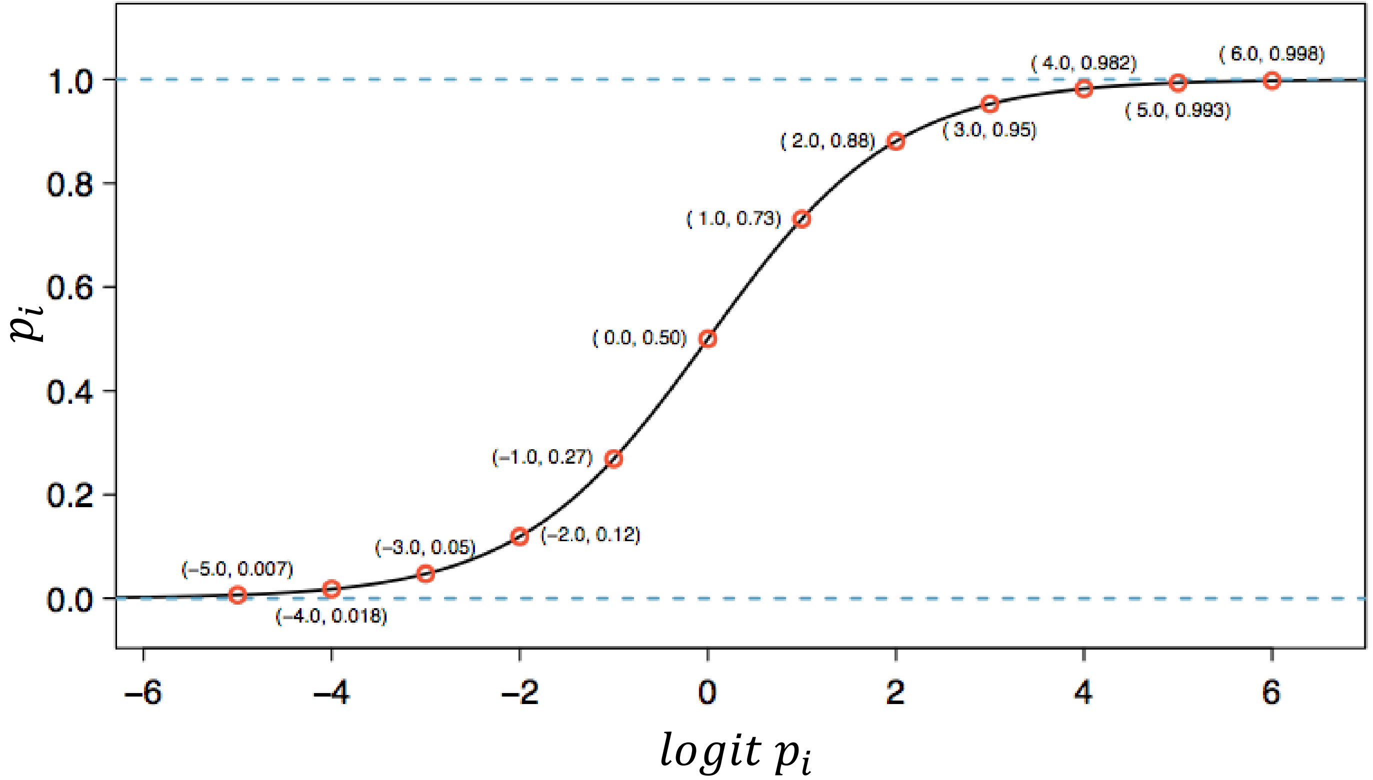
O PROBLEMA DE CLASSIFICAÇÃO

Uma transformação comum para p_i é a transformação de $logit$, que pode ser escrita como

$$logit\ OR_i = \ln\left(\frac{p_i}{1-p_i}\right) = \boldsymbol{\omega}^T\boldsymbol{x}$$

Resolvendo para p_i tal que,

$$p_i = g(\boldsymbol{\omega}^T\boldsymbol{x})$$



O PROBLEMA DE CLASSIFICAÇÃO

De forma completa,

$$p_i = p\left(y = k \middle| \boldsymbol{x}^{(i)}; \boldsymbol{\omega}\right)$$

descreve a probabilidade para do dado i pertencer à classe $k = 1,2, \dots, K$, dado que conhecemos a entrada $\boldsymbol{x}$, parametrizada por $\boldsymbol{\omega}$.

$$p\left(y = k \middle| \boldsymbol{x}^{(i)}; \boldsymbol{\omega}\right) = g\left(\boldsymbol{\omega}^T \boldsymbol{x}^{(i)}\right)$$

Como $g(\boldsymbol{\omega}^T \boldsymbol{x})$ é um modelo de probabilidade, $0 \leq g(\boldsymbol{\omega}^T \boldsymbol{x}) \leq 1$ para qualquer $\boldsymbol{x}$.

CLASSIFICAÇÃO BINÁRIA VS MULTICLASSE

Para problemas de classificação binária ($k = 1{,}2$, e $y = 0{,}1$), aprenderemos um modelo $g(\boldsymbol{x})$ para o qual

$$p(y = 1|\boldsymbol{x}; \boldsymbol{\omega}) \text{ é modelado por } g(\boldsymbol{\omega}^T \boldsymbol{x})$$

$$p(y = 0|\boldsymbol{x}; \boldsymbol{\omega}) \text{ é modelado por } 1 - g(\boldsymbol{\omega}^T \boldsymbol{x})$$

Para problemas multiclasse ($k = 1{,}2, \dots, K$), o classificador retorna uma função com valor vetorial $g(\boldsymbol{x})$, em que

$p(y = 1|\boldsymbol{x}; \boldsymbol{\omega})$ é modelado por $g_1(\boldsymbol{\omega}^T \boldsymbol{x})$

$p(y = 2|\boldsymbol{x}; \boldsymbol{\omega})$ é modelado por $g_2(\boldsymbol{\omega}^T \boldsymbol{x})$

$\vdots$

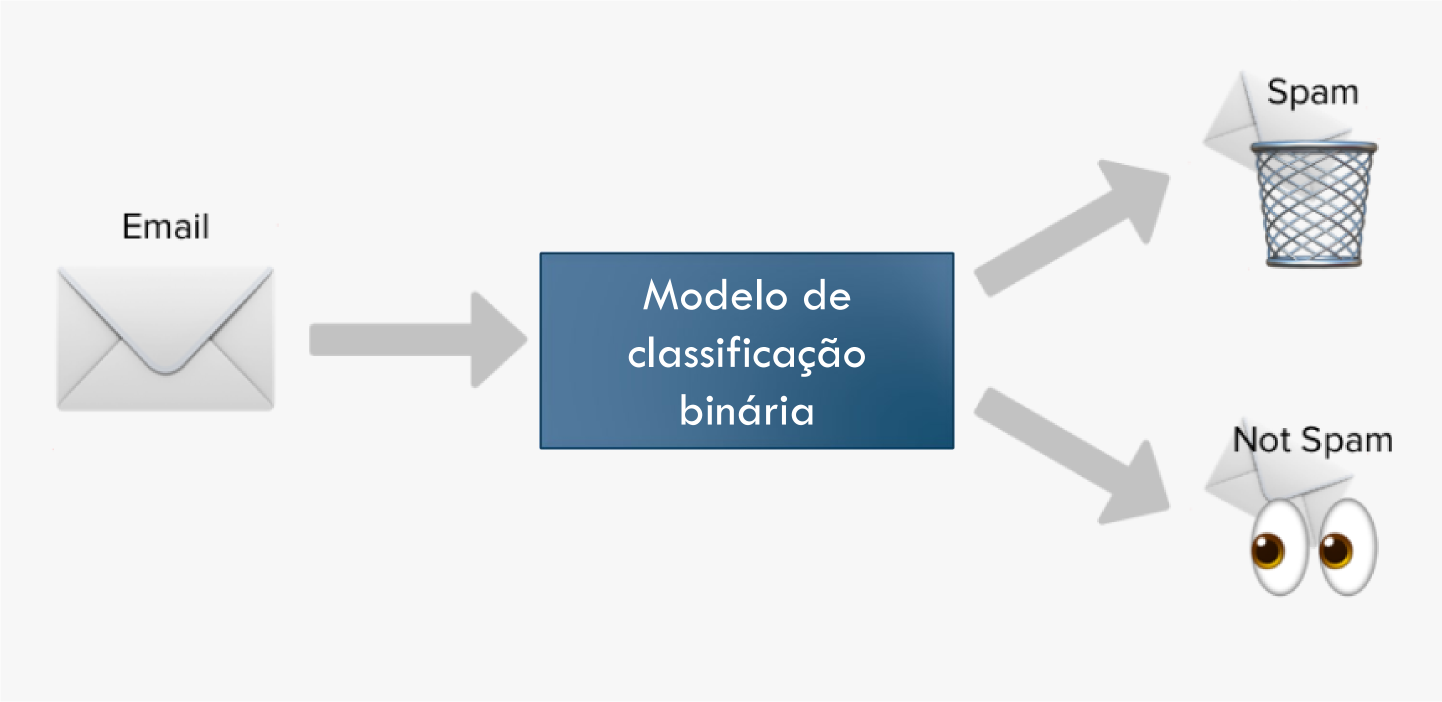
$p(y = K|\boldsymbol{x}; \boldsymbol{\omega})$ é modelado por $g_M(\boldsymbol{\omega}^T \boldsymbol{x})$

$$\boldsymbol{g}(\boldsymbol{x}) = \begin{bmatrix} g_1(\boldsymbol{\omega}^T \boldsymbol{x}) \\ g_2(\boldsymbol{\omega}^T \boldsymbol{x}) \\ \vdots \\ g_K(\boldsymbol{\omega}^T \boldsymbol{x}) \end{bmatrix}$$

cada elemento $g_k(\boldsymbol{\omega}^T \boldsymbol{x})$ de $\boldsymbol{g}(\boldsymbol{x})$ corresponde à probabilidade condicional da classe $p(y = k|\boldsymbol{x}; \boldsymbol{\omega})$

CLASSIFICAÇÃO BINÁRIA

Por enquanto, vamos nos concentrar no problema de classificação binária no qual y pode assumir apenas dois valores, 0 e 1 (a maior parte do que aprendermos também será generalizada para o caso de várias classes).



https://www.pngitem.com/middle/ibiboiT_spam-not-spam-machine-learning-hd-png-download/

RESUMO: REGRESSÃO LOGÍSTICA BINÁRIA

$$y^{(i)} = 0{,}1$$

$$\ln\left(\frac{p_i}{1-p_i}\right) = \omega_0 + \omega_1 x_1^{(i)} + \omega_2 x_2^{(i)} + \cdots + \omega_n x_n^{(i)}$$

$$\ln\left(\frac{p_i}{1-p_i}\right) = \boldsymbol{\omega}^T\boldsymbol{x}$$

$$p_i = g(\boldsymbol{\omega}^T\boldsymbol{x}) = \frac{1}{1+e^{-\boldsymbol{\omega}^T\boldsymbol{x}}}$$

$$e^{\ln\left(\frac{p_i}{1-p_i}\right)} = e^{\boldsymbol{\omega}^T\boldsymbol{x}}$$

$$\frac{p_i}{1-p_i} = e^{\boldsymbol{\omega}^T\boldsymbol{x}}$$

$$p_i = (1-p_i)e^{\boldsymbol{\omega}^T\boldsymbol{x}}$$

$$p_i + p_i e^{\boldsymbol{\omega}^T\boldsymbol{x}} = e^{\boldsymbol{\omega}^T\boldsymbol{x}}$$

$$p_i\left(1+e^{\boldsymbol{\omega}^T\boldsymbol{x}}\right) = e^{\boldsymbol{\omega}^T\boldsymbol{x}}$$

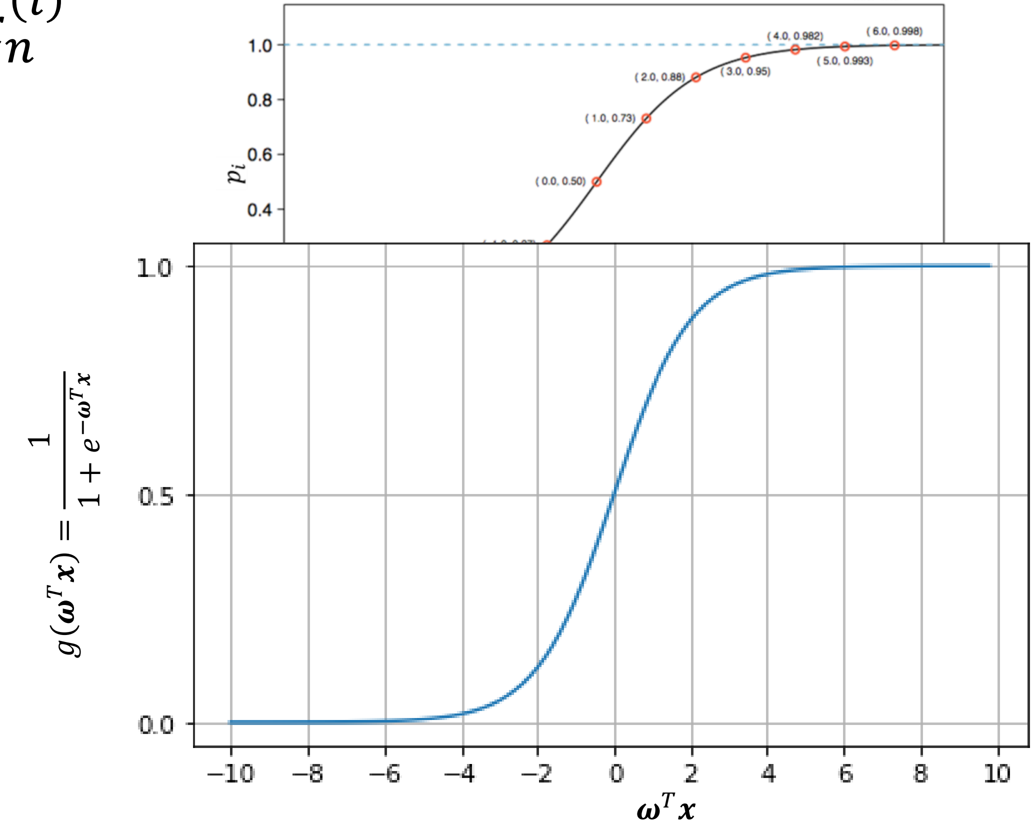
$$p_i = \frac{e^{\boldsymbol{\omega}^T\boldsymbol{x}}}{1+e^{\boldsymbol{\omega}^T\boldsymbol{x}}}\frac{e^{-\boldsymbol{\omega}^T\boldsymbol{x}}}{e^{-\boldsymbol{\omega}^T\boldsymbol{x}}}$$

$$p_i = \frac{1}{1+e^{-\boldsymbol{\omega}^T\boldsymbol{x}}}$$

Temos uma hipótese de como mc
nosso problema.

$$h(\boldsymbol{x}) = g(\boldsymbol{\omega}^T\boldsymbol{x}) = \frac{}{1+e^{-\boldsymbol{\omega}^T\boldsymbol{x}}}$$

$\boldsymbol{\omega}???$

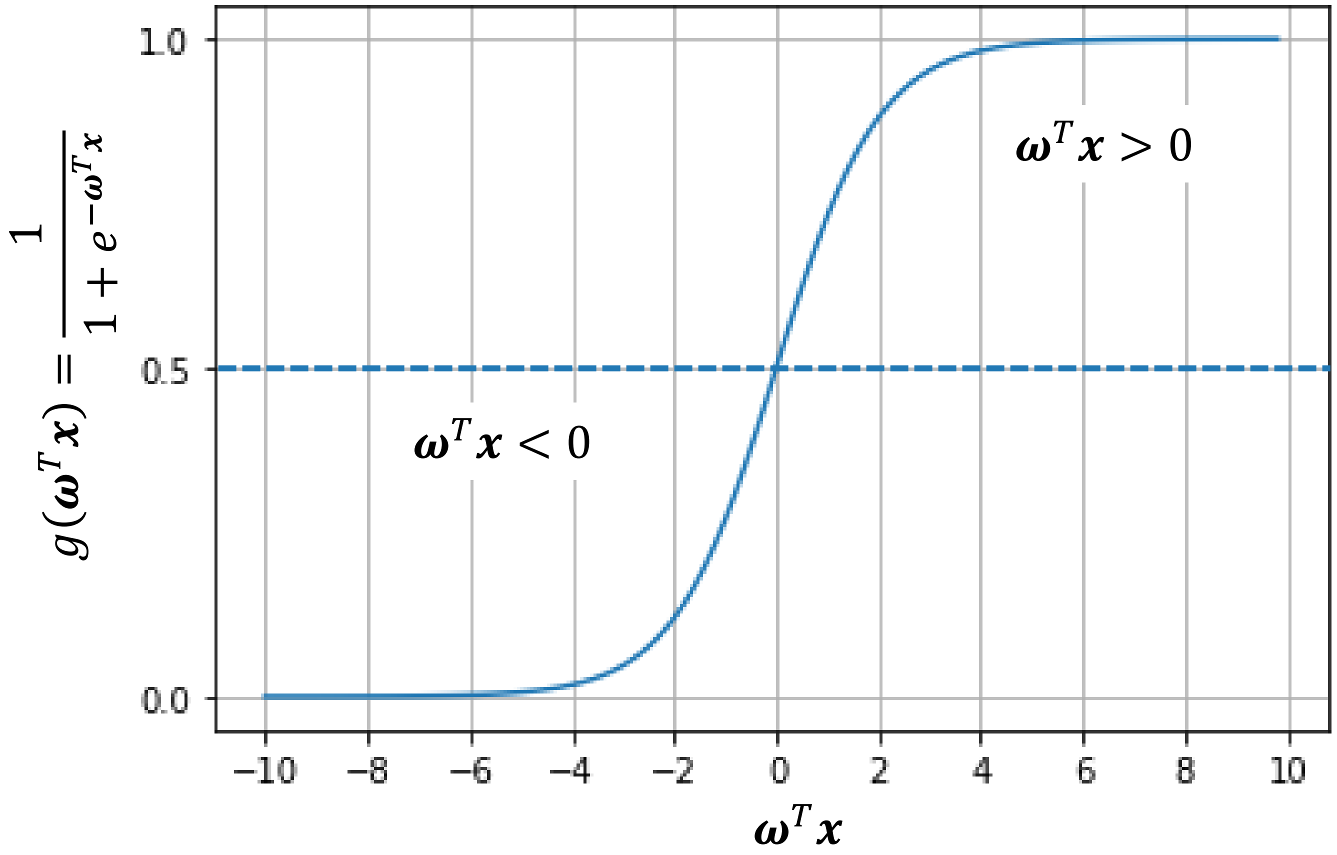


HIPÓTESE h

$$h_\omega(\boldsymbol{x}) = \frac{1}{1+e^{-\boldsymbol{\omega}^T\boldsymbol{x}}} = g(\boldsymbol{\omega}^T\boldsymbol{x})$$

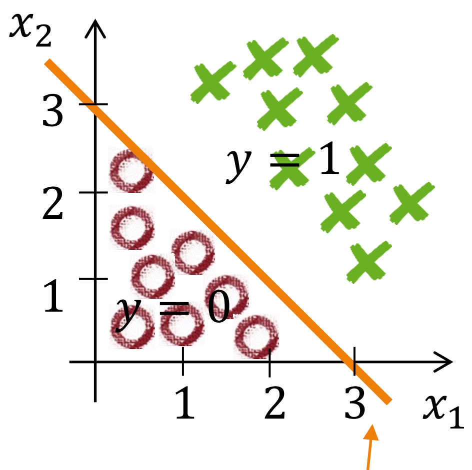
$$h_\omega(\boldsymbol{x}) = \begin{cases} > 0.5 & \boldsymbol{\omega}^T\boldsymbol{x} > 0 \\ < 0.5 & \boldsymbol{\omega}^T\boldsymbol{x} < 0 \end{cases}$$

Se a soma ponderada de entradas for maior que zero, a classe prevista é 1, caso contrário, será 0. Portanto, **o limite de decisão que separa as duas classes pode ser encontrado configurando a soma ponderada das entradas como $\omega^T x = 0$.**



CONTORNO DE DECISÃO

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_0 \\ \omega_1 \\ \omega_2 \end{bmatrix} = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$



Contorno de decisão
É uma propriedade da hipótese e dos parâmetros definidos.

$$h_\omega(\boldsymbol{x}) = g(\boldsymbol{\omega}^T\boldsymbol{x}) = g(\omega_0 + \omega_1 x_1 + \omega_2 x_2)$$

Prevemos $y = 1$ se:

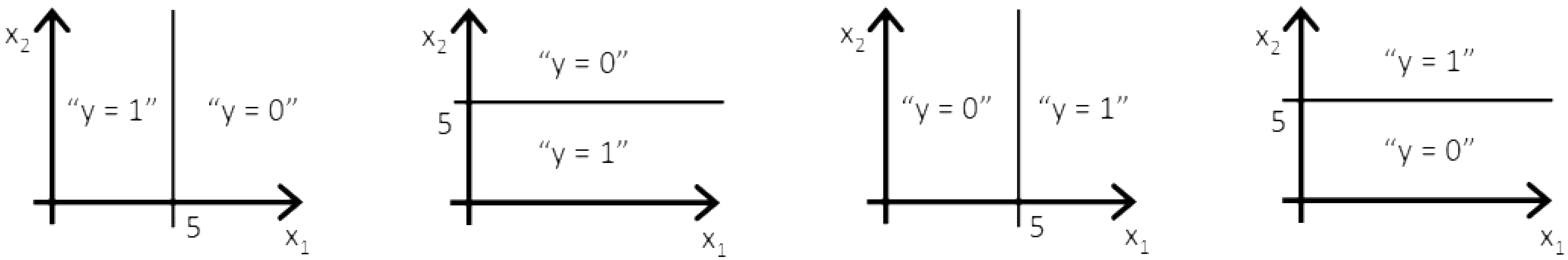
$$-3 + x_1 + x_2 \geq 0$$

$$x_1 + x_2 = 3 \qquad h_\omega(\boldsymbol{x}) = 0.5$$

QUESTÃO PARA VOCÊ PENSAR

Considere a regressão logística com 2 características, $h_\omega(\boldsymbol{x}) = g(\boldsymbol{\omega}^T \boldsymbol{x}) = g(\omega_0 + \omega_1 x_1 + \omega_2 x_2\ )$.

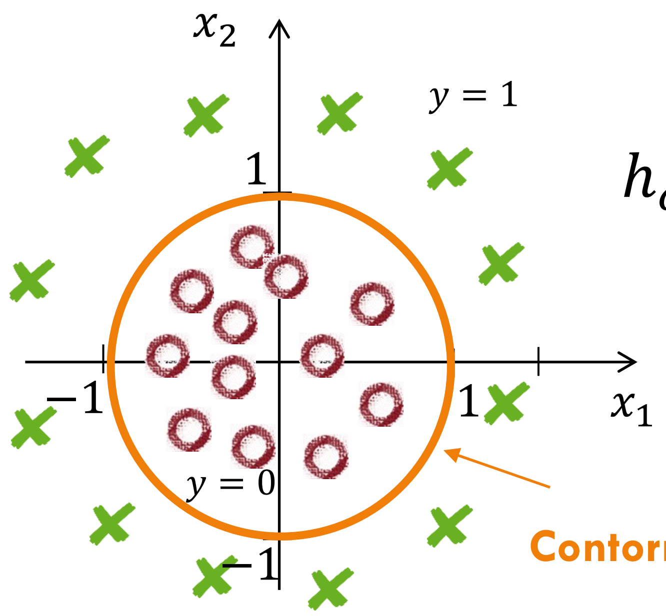
Suponha $\omega_0 = 5, \omega_1 = -1, \omega_2 = 0$, ié, $h_\omega(\boldsymbol{x}) = g(5 - x_1\ )$.



(b) (c) (d)

CONTORNO NÃO LINEAR

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_0 \\ \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$



$$h_\omega(\boldsymbol{x}) = g(\boldsymbol{\omega}^T \boldsymbol{x}) = g(\omega_0 + \omega_1 x_1 + \omega_2 x_2 + \omega_3 x_1^2 + \omega_4 x_2^2)$$

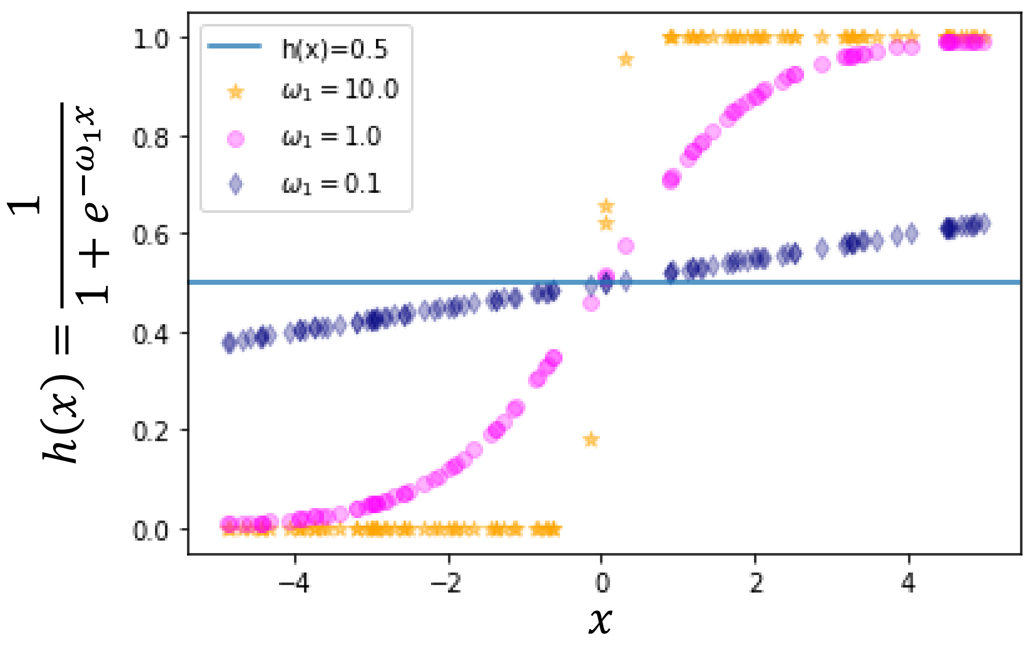
Prevemos $y = 1$ se:

$$-1 + x_1^2 + x_2^2 \geq 0$$

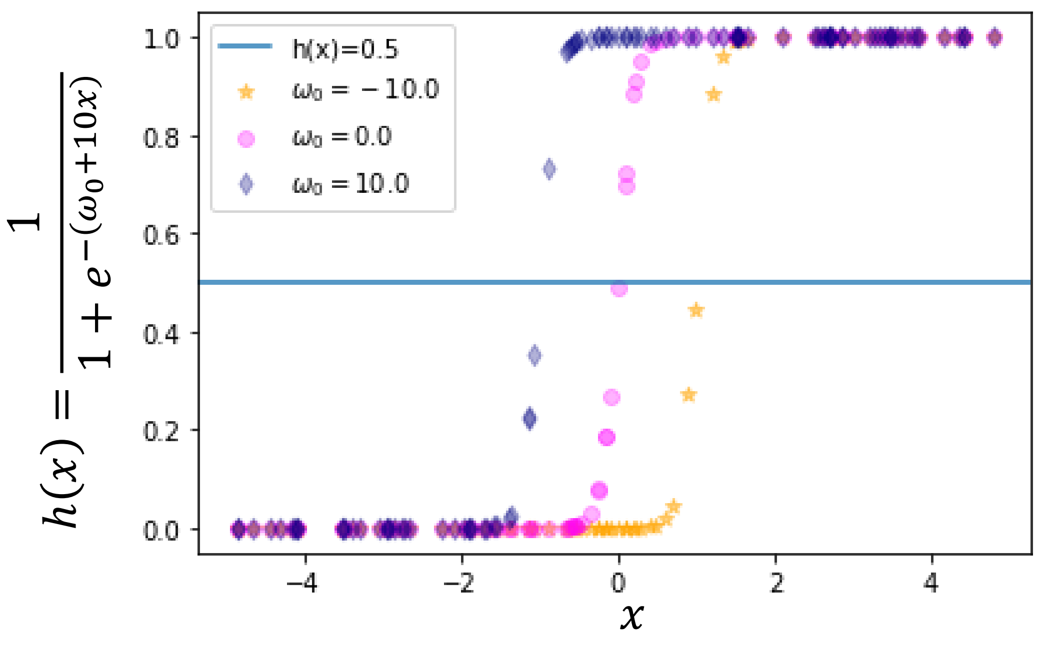
Contorno de decisão $\quad x_1^2 + x_2^2 = 1 \quad h_\omega(\boldsymbol{x}) = 0.5$

ANALISANDO O CASO LINEAR: $\boldsymbol{\omega}^T\boldsymbol{x} = \omega_0 + \omega_1 x$

$\omega_0 = 0$



$\omega_1 = 10$



EXEMPLO

Suponha que a probabilidade de um cliente adquirir uma assinatura de uma revista por mala direta é,

$$prob(evento) = \frac{1}{1 + e^{-(-1.143 + 0.452x_1 + 0.029x_2 - 0.242x_3)}}$$

x_1 é sexo (1 para feminino e 0 para masculino), x_2 é idade e x_3 é estado civil (1 para solteiro e 0 para casado).

Uma pessoa do sexo feminino, com 40 anos de idade e casada, irá adquirir a assinatura da revista?

$$prob(evento) = \frac{1}{1 + e^{-(-1.143 + 0.452 \times 1 + 0.029 \times 40 - 0.242 \times 0)}} = 0.61$$

Sim, irá adquirir a revista.

Exemplo extraído de:
L. P. Favero; P. Belfiore; F. L. da Silva; B. L. Chan. *Análise de dados: modelagem multivariada para tomada de decisões*, Ed. Campus

NOSSO PROBLEMA, ENTÃO É...

Conjunto de m dados treinamento: $\{(\boldsymbol{x}^{(1)}, y^{(1)}), (\boldsymbol{x}^{(2)}, y^{(2)}), \ldots, (\boldsymbol{x}^{(m)}, y^{(m)})\}$
onde

$$\boldsymbol{x}^{(i)} \in \begin{bmatrix} x_0^{(i)} \\ x_1^{(i)} \\ \vdots \\ x_n^{(i)} \end{bmatrix}, x_0^{(i)} = 1, y \in \{0,1\}$$

Como escolho $\boldsymbol{\omega}$????

$$h_\omega(\boldsymbol{x}) = g(\boldsymbol{\omega}^T \boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{\omega}^T \boldsymbol{x}}}$$

ENTENDENDO O CUSTO MATEMATICAMENTE...

Para obter a função custo, a interpretaremos estatisticamente com o método de máxima verossimilhança. Maximizar a função de verossimilhança equivale a encontrar o valor de $\boldsymbol{\omega}$ que torna a observação de $\boldsymbol{y}$ a **mais provável** possível,

$$\widehat{\boldsymbol{\omega}} = \arg\max_{\boldsymbol{\omega}} p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\omega})$$

onde $p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\omega})$ é a densidade de probabilidade de todas as saídas observadas $\boldsymbol{y}$ nos dados de treinamento, dadas todas as entradas $\boldsymbol{x}$ e parâmetros $\boldsymbol{\omega}$. Isso determina matematicamente o que significa **mais provável**.

UM POUCO DA NOMENCLATURA A SER USADA

A probabilidade, de acordo com a definição de distribuição de Bernoulli,

$$p\left(y^{(i)} \middle| \boldsymbol{x}^{(i)}; \boldsymbol{\omega}\right) = \left[h_\omega\left(\boldsymbol{x}^{(i)}\right)\right]^{y^{(i)}} \left[1 - h_\omega\left(\boldsymbol{x}^{(i)}\right)\right]^{1-y^{(i)}}$$

Onde $h_\omega(\boldsymbol{x}^{(i)}) = \frac{1}{1+e^{-\boldsymbol{\omega}^T \boldsymbol{x}^{(i)}}}$

Ou seja,

se a resposta da classificação binária é $y^{(i)} = 1$

$$p\left(y^{(i)} \middle| \boldsymbol{x}^{(i)}; \boldsymbol{\omega}\right) = h_\omega\left(\boldsymbol{x}^{(i)}\right)$$

se a resposta da classificação binária é $y^{(i)} = 0$

$$p\left(y^{(i)} \middle| \boldsymbol{x}^{(i)}; \boldsymbol{\omega}\right) = 1 - h_\omega\left(\boldsymbol{x}^{(i)}\right)$$

CONT...

As m observações são independents e, portanto,

$$p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\omega}) = \prod_{i=1}^{m} p\left(y^{(i)}\middle|\boldsymbol{x}^{(i)};\boldsymbol{\omega}\right) = \prod_{i=1}^{m} h_\omega\left(\boldsymbol{x}^{(i)}\right)^{y^{(i)}}\left[1-h_\omega\left(\boldsymbol{x}^{(i)}\right)\right]^{1-y^{(i)}}$$

Por razões numéricas, geralmente é melhor considerar o logaritmo de $p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\omega})$

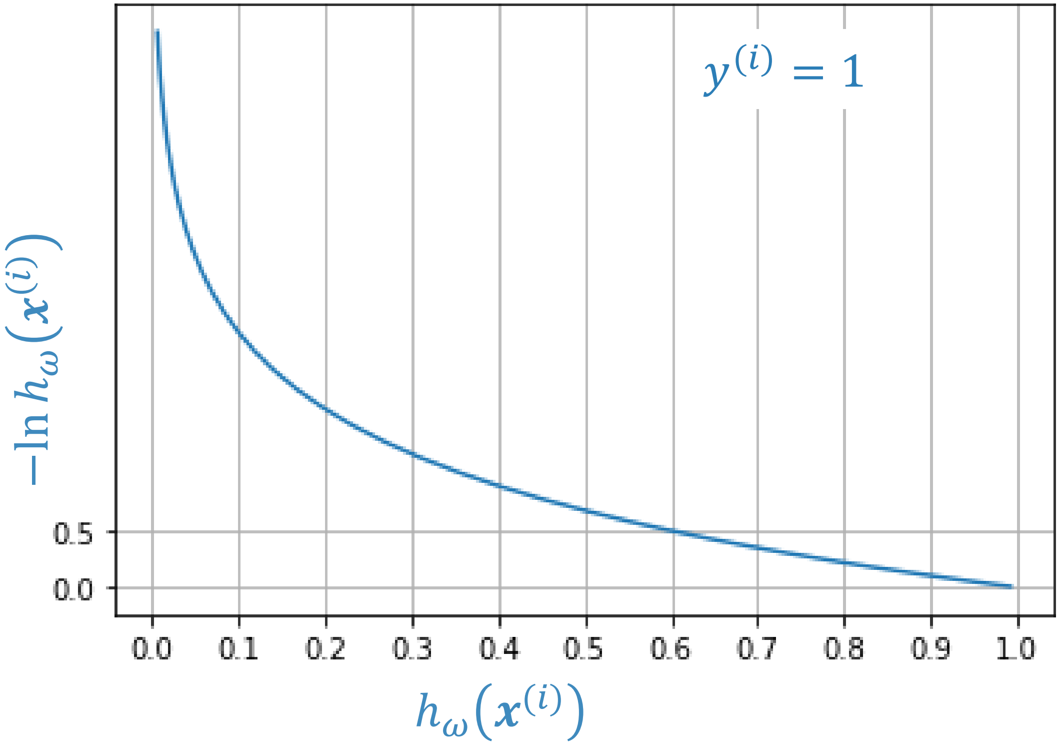
$$\ln p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\omega}) = \sum_{i=1}^{m} y^{(i)} \ln h_\omega\left(\boldsymbol{x}^{(i)}\right) + \left(1-y^{(i)}\right)\ln\left[1-h_\omega\left(\boldsymbol{x}^{(i)}\right)\right]$$

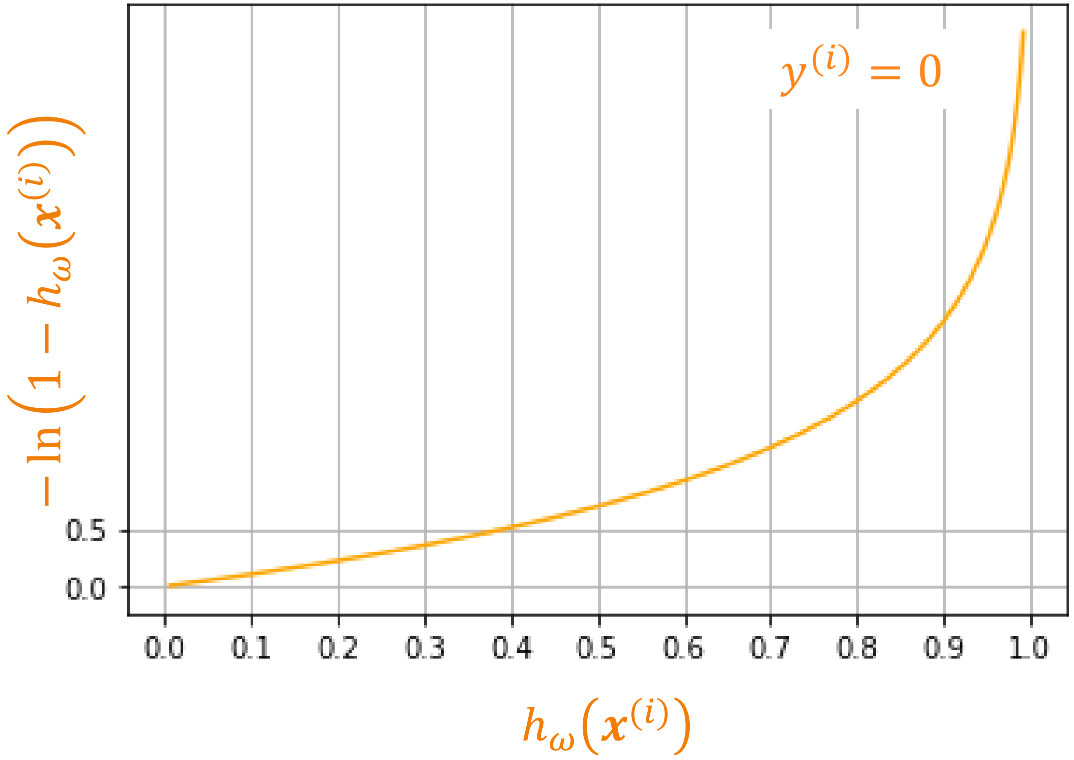
Portanto, achar o valor mais provável equivale a

$$\widehat{\boldsymbol{\omega}} = \arg\max_{\boldsymbol{\omega}} \sum_{i=1}^{m} y^{(i)} \ln h_\omega\left(\boldsymbol{x}^{(i)}\right) + \left(1-y^{(i)}\right)\ln\left[1-h_\omega\left(\boldsymbol{x}^{(i)}\right)\right]$$

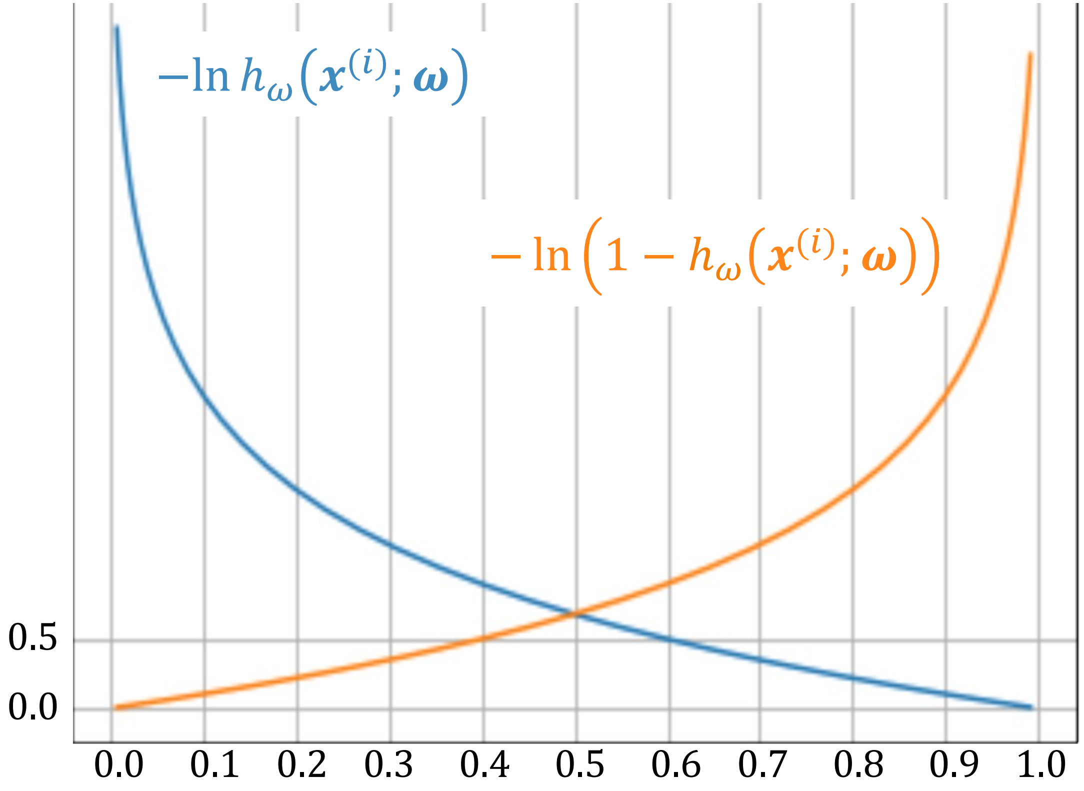
$y^{(i)}$ vale 0 ou 1. Portanto,

$$-\{y^{(i)} \ln h_\omega\left(\boldsymbol{x}^{(i)}\right) + \left(1 - y^{(i)}\right) \ln\left[1 - h_\omega\left(\boldsymbol{x}^{(i)}\right)\right]\} = \begin{cases} -\ln h_\omega\left(\boldsymbol{x}^{(i)}\right) & se\ \ y^{(i)} = 1 \\ -\ln\left(1 - h_\omega\left(\boldsymbol{x}^{(i)}\right)\right) & se\ \ y^{(i)} = 0 \end{cases}$$





OU SEJA... A FUNÇÃO CUSTO



$$L(\boldsymbol{\omega}) = \begin{cases} -\ln h_\omega\left(\boldsymbol{x}^{(i)};\boldsymbol{\omega}\right) & se\ \ y^{(i)} = 1 \\ -\ln\left(1-h_\omega\left(\boldsymbol{x}^{(i)};\boldsymbol{\omega}\right)\right) & se\ \ y^{(i)} = 0 \end{cases}$$

FUNÇÃO PERDA (LOSS FUNCTION)

Fazendo o somatório em todo meu conjunto de dados, tenho:

$$J(\boldsymbol{\omega}) = -\frac{1}{m}\sum_{i=1}^{m}\begin{cases} \ln\left[h_\omega\left(\boldsymbol{x}^{(i)}\right)\right] & se\ \ y^{(i)} = 1 \\ \ln\left[1 - h_\omega\left(\boldsymbol{x}^{(i)}\right)\right] & se\ \ y^{(i)} = 0 \end{cases}$$

entropia cruzada

$$J(\boldsymbol{\omega}) = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)}\ln[h_\omega(\boldsymbol{x}^{(i)})] + \left(1 - y^{(i)}\right)\ln[1 - h_\omega(\boldsymbol{x}^{(i)})]$$

Entropia cruzada como função perda (Cross Entropy Loss funtion). Também é conhecido como **perda de log** (log loss).

GRADIENTE DESCENDENTE

$$J(\boldsymbol{\omega}) = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \ln[h_\omega(\boldsymbol{x}^{(i)})] + \left(1 - y^{(i)}\right) \ln[1 - h_\omega(\boldsymbol{x}^{(i)})]$$

Queremos $\min_{\boldsymbol{\omega}} J(\boldsymbol{\omega})$

Repetir até convergência $\{$

$$\omega_{j+1} := \omega_j - \boxed{\alpha \frac{\partial}{\partial \omega_j} J(\boldsymbol{\omega})}$$

$$\frac{\partial}{\partial \omega_j} J(\boldsymbol{\omega}) = \frac{1}{m}\sum_{i=1}^{m}\left[h_\omega\left(\boldsymbol{x}^{(i)}\right) - y^{(i)}\right] x_j^{(i)}$$

$\}$

$Algoritmo$

Controle convergência plotando gráfico de J pelo número de iterações

$$J(\boldsymbol{\omega}) = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \ln[h_\omega(\boldsymbol{x}^{(i)})] + \left(1 - y^{(i)}\right) \ln[1 - h_\omega(\boldsymbol{x}^{(i)})]$$

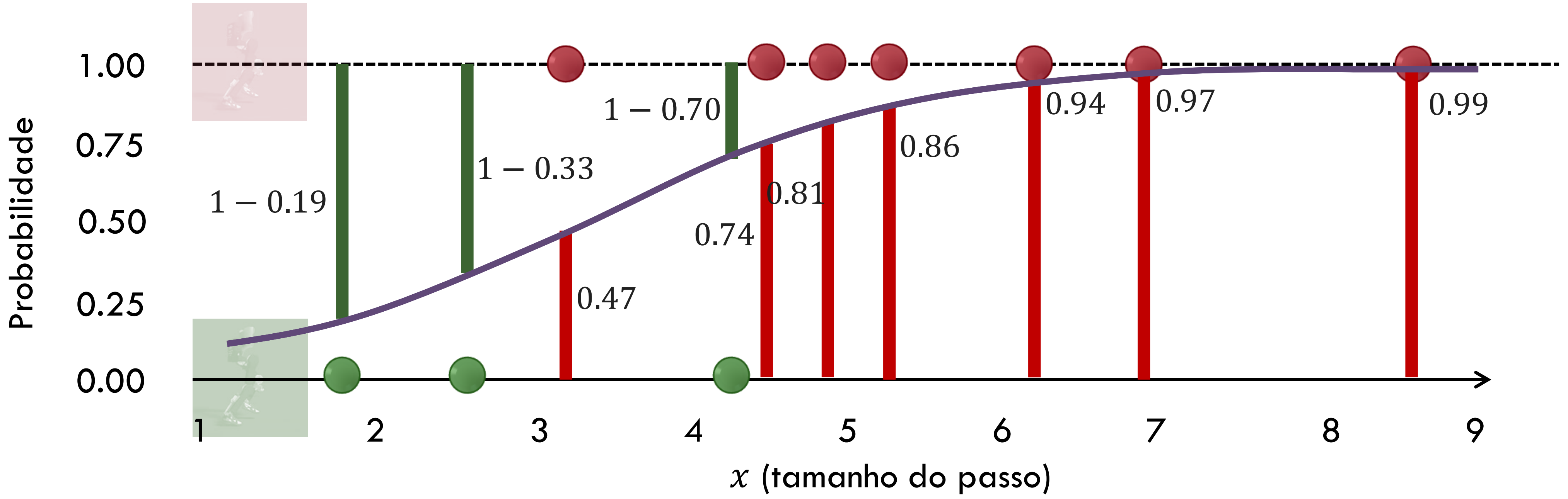
Queremos $\min_{\boldsymbol{\omega}} J(\boldsymbol{\omega})$

Repetir até convergência {

$$\omega_{j+1} := \omega_j - \alpha\frac{1}{m}\sum_{i=1}^{m}\left[h_\omega\left(\boldsymbol{x}^{(i)}\right) - y^{(i)}\right]x_j^{(i)}$$

}

EM NOSSO EXEMPLO

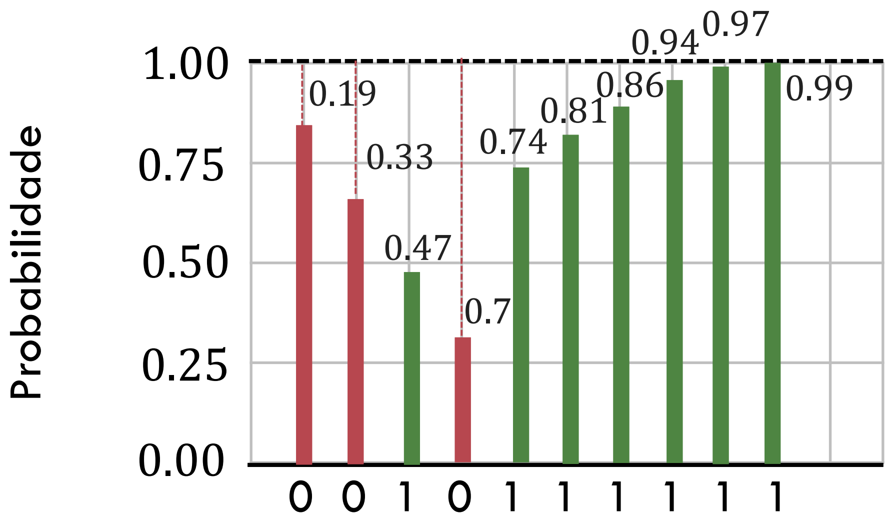


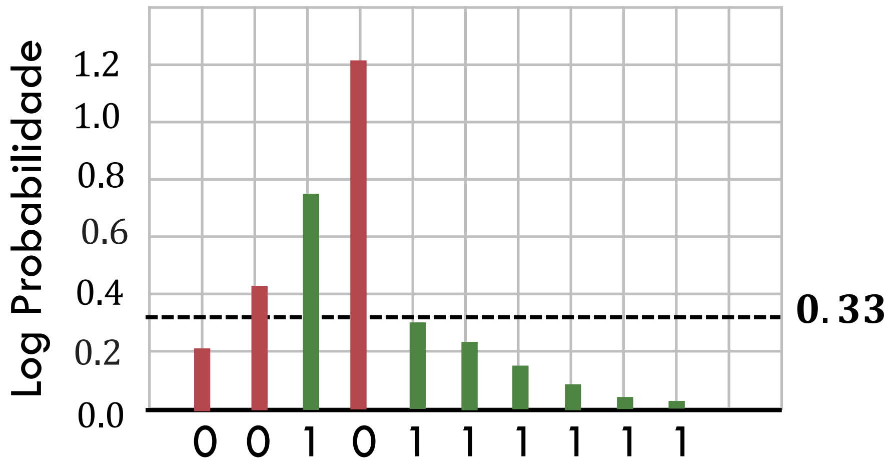
As barras representam as **probabilidades previstas** correspondentes à **verdadeira classe** de cada ponto!

$$J(\boldsymbol{\omega}) = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \ln\left[h_\omega\left(\boldsymbol{x}^{(i)}\right)\right] + \left(1 - y^{(i)}\right) \ln\left[1 - h_\omega\left(\boldsymbol{x}^{(i)}\right)\right]$$

$$J(\boldsymbol{\omega}) = -\frac{1}{10}\left[\begin{matrix}\ln(1-0.19) + \ln(1-0.33) + \ln(0.47) + \ln(1-0.7) + \ln(0.74) + \\ \ln(0.81) + \ln(0.86) + \ln(0.94) + \ln(0.97) + \ln(0.99)\end{matrix}\right] = 0.3329$$

Quão próxima é a distribuição prevista da distribuição verdadeira?
<u>É isso que determina o erro de entropia cruzada</u>.





MÉTRICAS

Matriz de confusão é uma medida de desempenho para o problema de classificação de aprendizado de máquina em que a saída pode ser duas ou mais classes.

Target → Previsão↓	POSITIVO	NEGATIVO
POSITIVO	Verdadeiro Positivo (VP)	Falso Positivo (FP)
NEGATIVO	Falso Negativo (FN)	Verdadeiro Positivo (VP)

Acurácia

$$A = \frac{VP + VN}{VP + VN + FP + FN}$$

performance geral do modelo

Target → Previsão↓	POSITIVO	NEGATIVO
POSITIVO	Verdadeiro Positivo (VP)	Falso Positivo (FP)
NEGATIVO	Falso Negativo (FN)	Verdadeiro Positivo (VP)

$$VPP = \frac{VP}{VP + FP}$$

$$VPN = \frac{VN}{VN + FN}$$

sensibilidade

$$\frac{VP}{VP + FN}$$

especificidade

$$\frac{VN}{VN + FP}$$

Target → Previsão↓	POSITIVO	NEGATIVO
POSITIVO	Verdadeiro Positivo (VP)	Falso Positivo (FP)
NEGATIVO	Falso Negativo (FN)	Verdadeiro Negativo (VN)
	sensibilidade	especificidade
	$\frac{VP}{VP + FN}$	$\frac{VN}{VN + FP}$

$$VPP = \frac{VP}{VP + FP}$$

$$VPN = \frac{VN}{VN + FN}$$

Ou **Revocação (Recall)**, dentre todas as situações de classe positiva como valor esperado, quantas estão corretas

Target → Previsão↓	POSITIVO	NEGATIVO
POSITIVO	Verdadeiro Positivo (VP)	Falso Positivo (FP)
NEGATIVO	Falso Negativo (FN)	Verdadeiro Negativo (VN)

$$VPP = \frac{VP}{VP + FP}$$

Precisão

dentre as classificações positivas que o modelo fez, quantas estão corretas

$$VPN = \frac{VN}{VN + FN}$$

sensibilidade

$$\frac{VP}{VP + FN}$$

especificidade

$$\frac{VN}{VN + FP}$$

Ou **Revocação (Recall)**, dentre todas as situações de classe positiva como valor esperado, quantas estão corretas

Target → / Previsão↓	POSITIVO	NEGATIVO
POSITIVO	Verdadeiro Positivo (VP)	Falso Positivo (FP)
NEGATIVO	Falso Negativo (FN)	Verdadeiro Negativo (VN)

$$VPP = \frac{VP}{VP + FP}$$

$$VPN = \frac{VN}{VN + FN}$$

Precisão

dentre as classificações positivas que o modelo fez, quantas estão corretas

sensibilidade

$$\frac{VP}{VP + FN}$$

especificidade

$$\frac{VN}{VN + FP}$$

Ou **Revocação (Recall)**, dentre todas as situações de classe positiva como valor esperado, quantas estão corretas

F1

$$F1 = \frac{2PR}{P + R}$$

média harmônica entre **precisão e revocação.**

TRADING OFF ENTRE PRECISÃO E RECALL

$$P = \frac{Verdadeiros\ Positivos}{N.de\ positivos\ previstos} = \frac{VP}{VP + \boldsymbol{FP}}$$

$$R = \frac{Verdadeiros\ Positivos}{N.de\ positivos\ reais} = \frac{VP}{VP + \boldsymbol{FN}}$$

$$h_\omega(\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{\omega}^T \boldsymbol{x}}} = g(\boldsymbol{\omega}^T \boldsymbol{x})$$

Prevemos 1 se $h_\omega(\boldsymbol{x}) \geq 0.3$

Prevemos 0 se $h_\omega(\boldsymbol{x}) < 0.3$

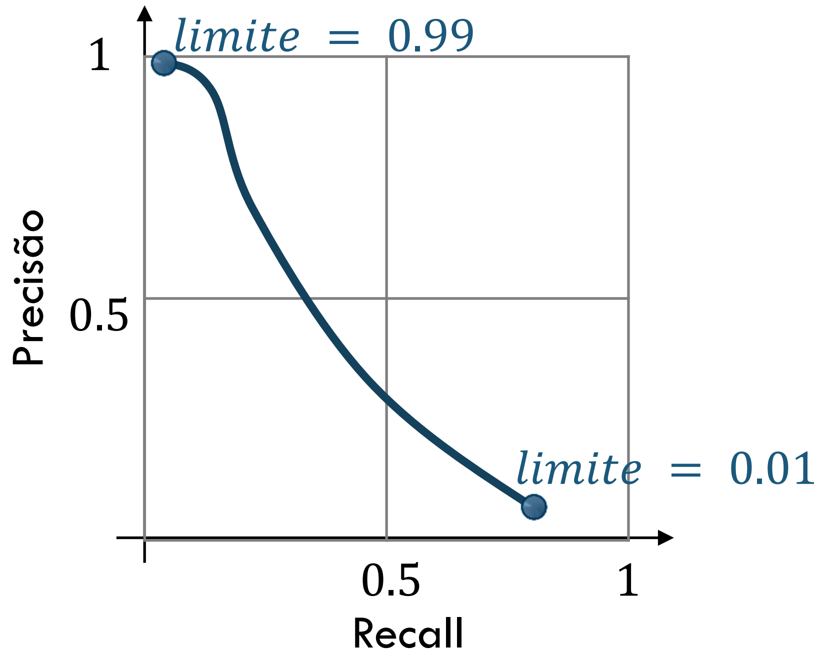
Supondo que queremos prever $y = 1$ somente se tivermos bastante certeza:

Alta precisão, baixo recall

Supondo que queremos prever $y = 0$ somente se tivermos bastante certeza

Alto recall, baixa precisão

De forma general: $h_\omega(\boldsymbol{x}) \geq limite$



F1

	Precisão (P)	Recall (R)
Modelo 01	0.5	0.4
Modelo 02	0.7	0.1
Modelo 03	0.02	1.0

$$M = \frac{P + R}{2}$$

$$F1 = \frac{2PR}{P + R}$$

$$P = 0, R = 0 \rightarrow F1 \rightarrow \infty$$

$$P = 1, R = 1 \rightarrow F1 = 1$$

Exemplo extraído de:
https://www.coursera.org/learn/machine-learning/lecture/CuONQ/trading-off-precision-and-recall

Target → / Previsão↓	POSITIVO	NEGATIVO
POSITIVO	Verdadeiro Positivo (VP)	Falso Positivo (FP)
NEGATIVO	Falso Negativo (FN)	Verdadeiro Negativo (VN)
	sensibilidade	especificidade
	$\frac{VP}{VP+FN}$	$\frac{VN}{VN+FP}$

$$VPP = \frac{VP}{VP+FP}$$

Precisão

dentre as classificações positivas que o modelo fez, quantas estão corretas

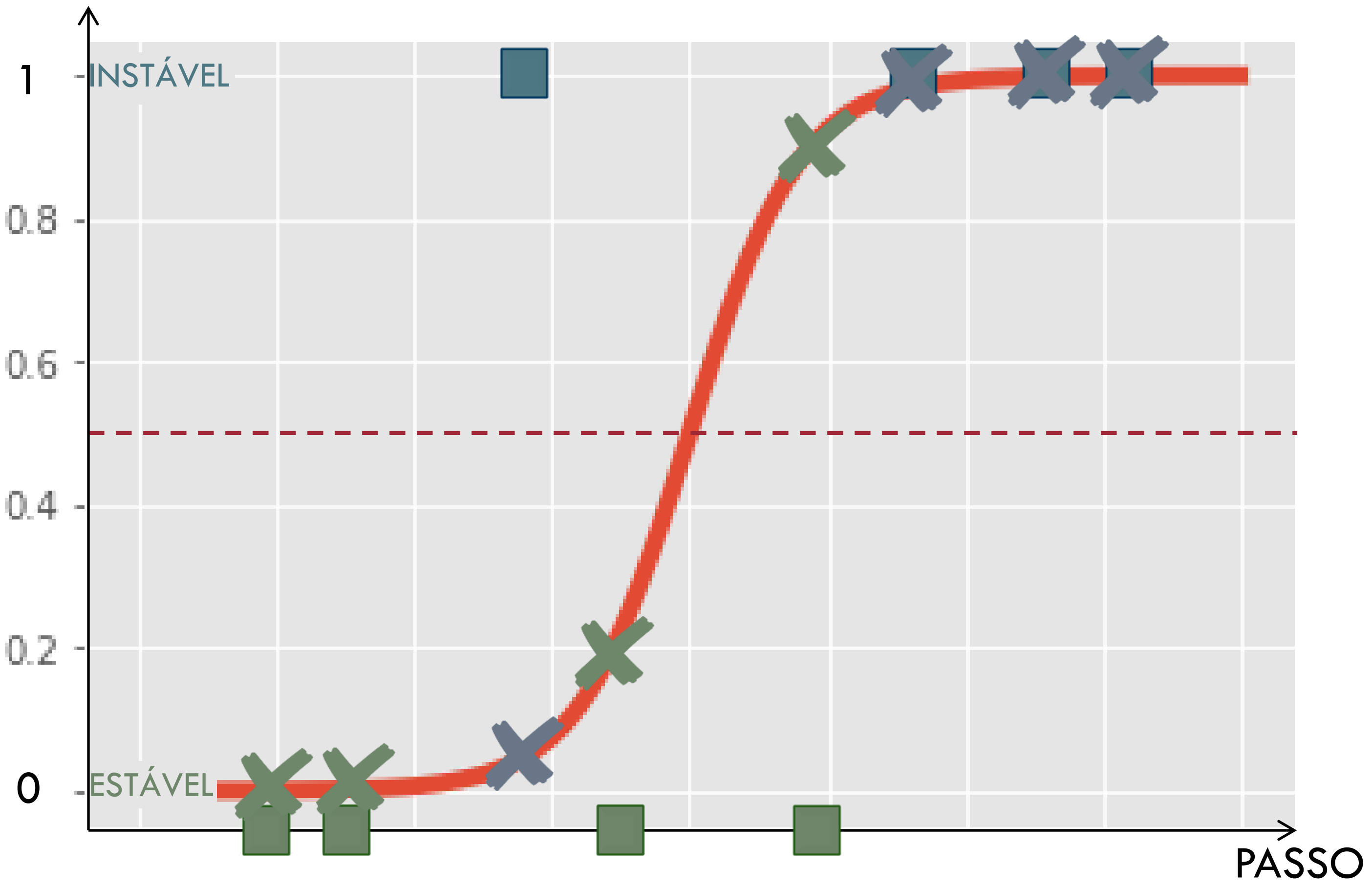
$$VPN = \frac{VN}{VN+FN}$$

Ou **Revocação (Recall)**, dentre todas as situações de classe positiva como valor esperado, quantas estão corretas

Taxa de FP

$$FPR = \frac{FP}{FP+VN}$$

$1 - especificidade =$ dentre todas as situações de classe negativa, quantas estão incorretas



		Target	
	VP	1	0
Predicted	1	3	1
	0	1	3

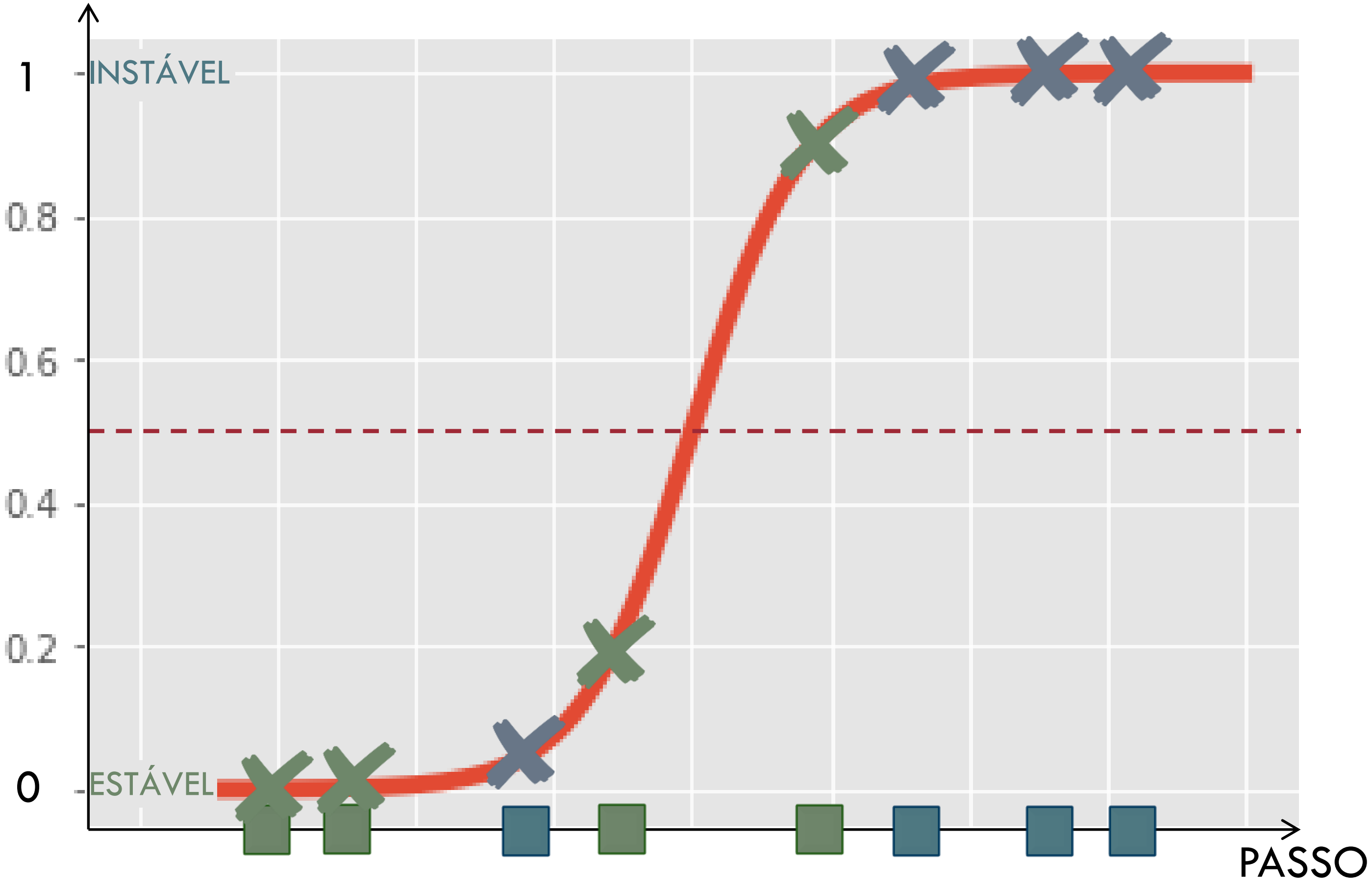
FP

FN VN

$$R = \frac{VP}{VP + FN} = \frac{3}{4} = 0.75$$

$$FPR = \frac{FP}{FP + VN} = \frac{1}{4} = 0.25$$

E se colocarmos o limite de modo a classificarmos SEMPRE como instável?

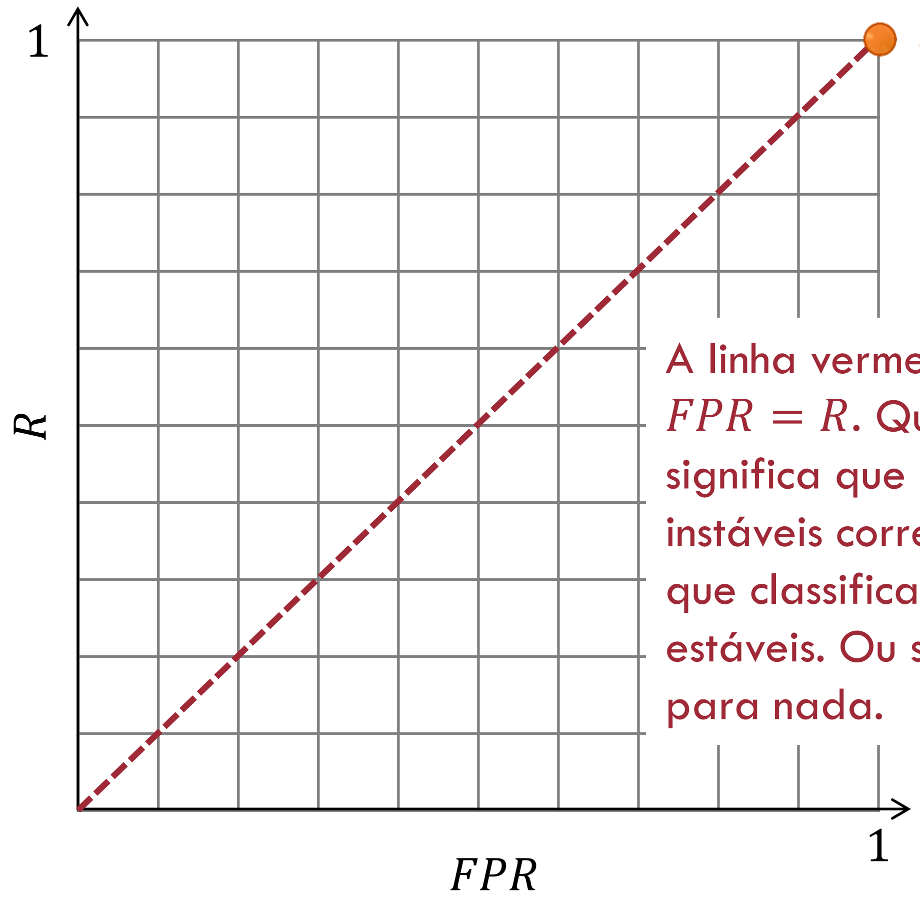


		Target		
	VP	1	0	
Predicted	1	4	4	FP
	0	0	0	
		FN	VN	

$$R = \frac{VP}{VP + FN} = \frac{4}{4 + 0} = 1$$

$$FPR = \frac{FP}{FP + VN} = \frac{4}{4 + 0} = 1$$

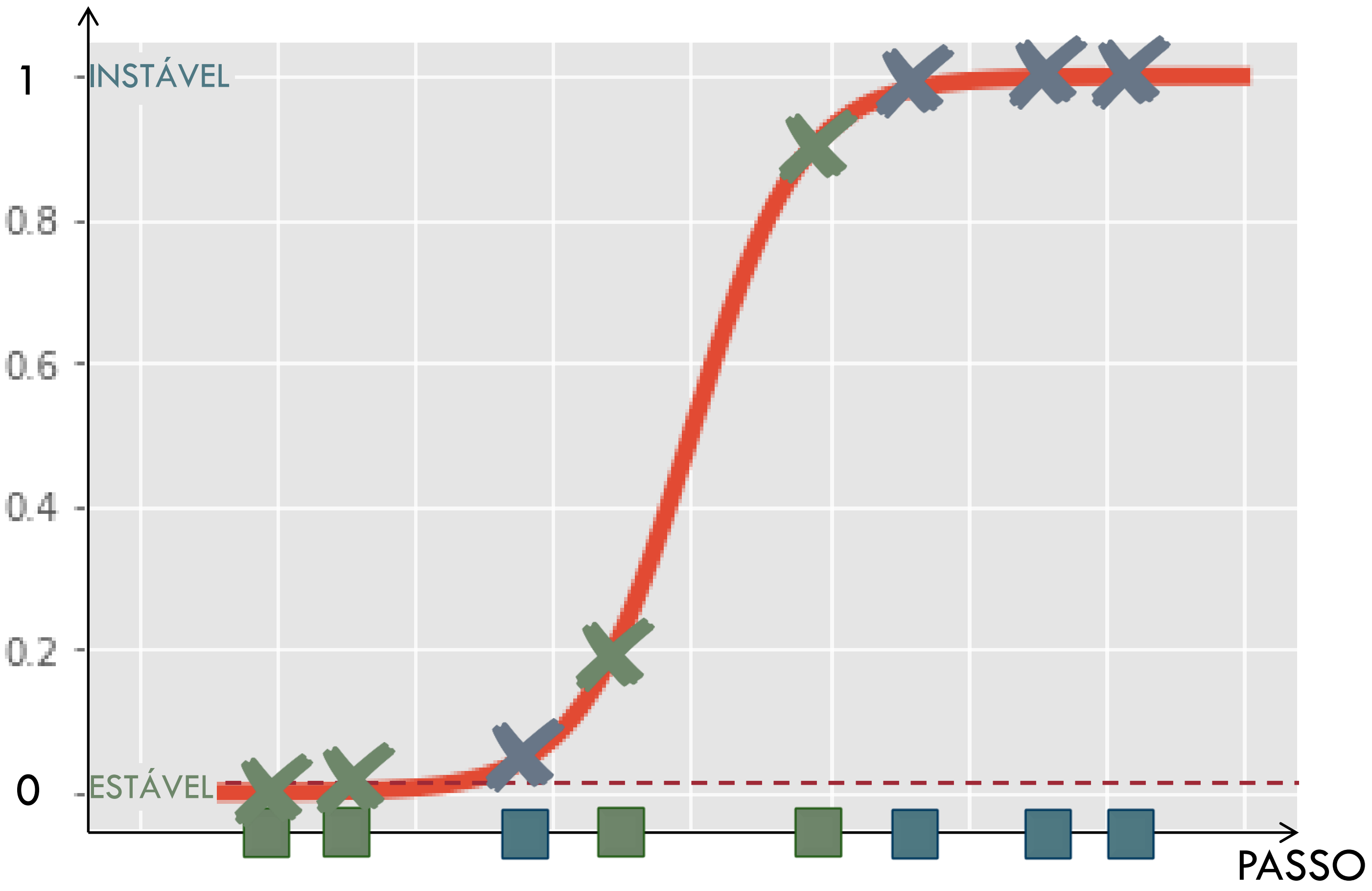
ROC: R vs FPR



No ponto (1,1) significa que, apesar de **classificar corretamente todos os passos instáveis, classificou incorretamente todos os passos estáveis.**

A linha vermelha diagonal significa que $FPR = R$. Qualquer ponto nessa linha significa que a proporção de classificados instáveis corretamente é a mesma proporção que classifica incorretamente os passos estáveis. Ou seja, o classificador não serve para nada.

E se colocarmos o limite de modo que apenas o menor de todos os passos é classificado como estável?



		Target	
		1	0
Predicted	1	4	3
	0	0	1

VP

FP

FN

VN

$$R = \frac{VP}{VP + FN} = \frac{4}{4 + 0} = 1$$

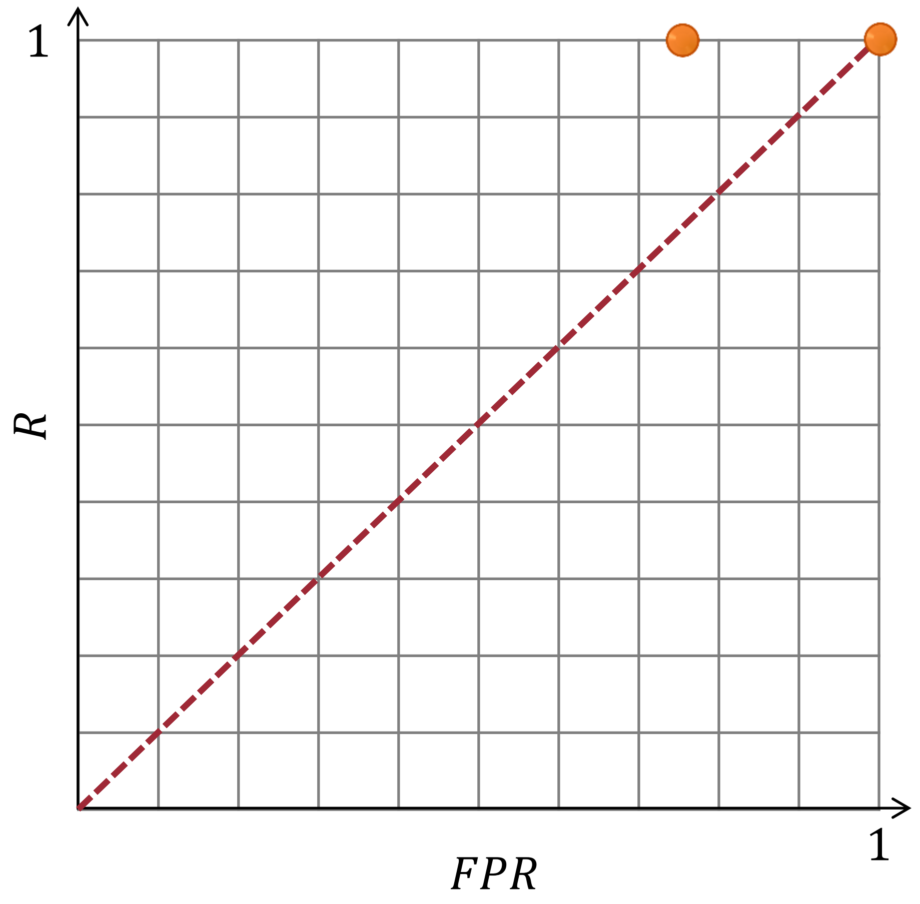
$$FPR = \frac{FP}{FP + VN} = \frac{3}{3 + 1} = 0.75$$

ROC: R vs FPR

O novo ponto (0.75,1) está à esquerda da linha vermelha e, portanto, sabemos que a proporção de corretamente classificados como passos instáveis (VP) **é maior que** a proporção de classificados incorretamente como instáveis (FP).

$$R = \frac{VP}{VP + FN}$$

$$FPR = \frac{FP}{FP + VN}$$



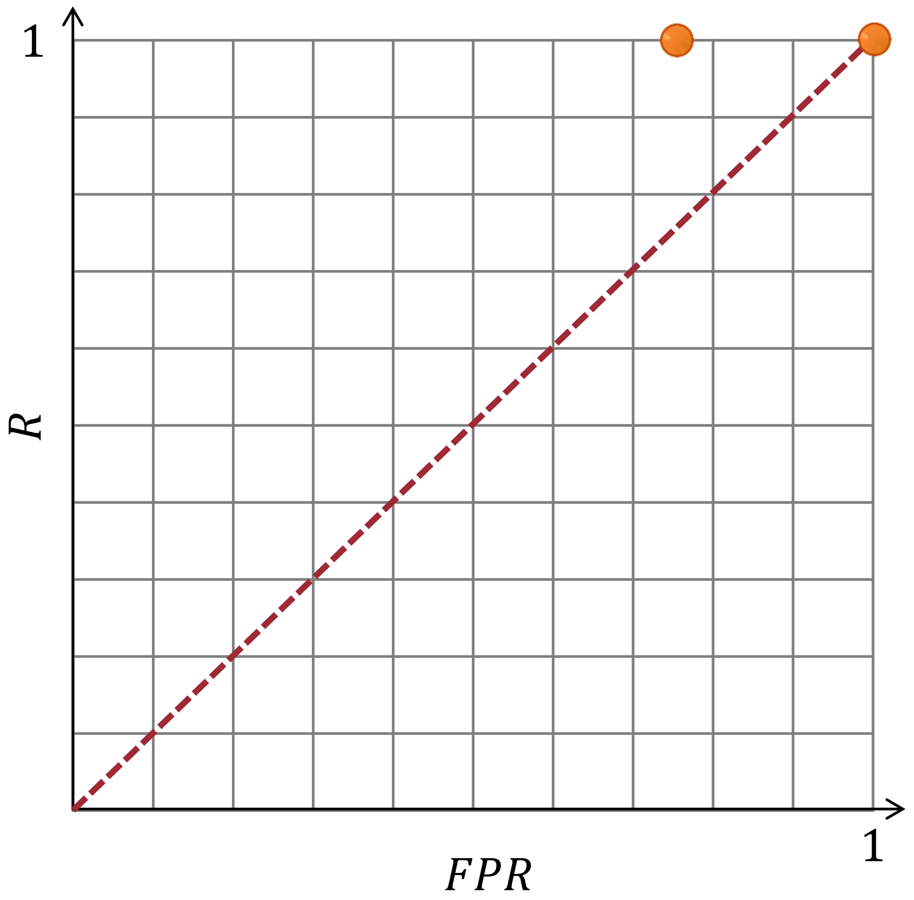
ROC: R vs FPR

Ou seja,

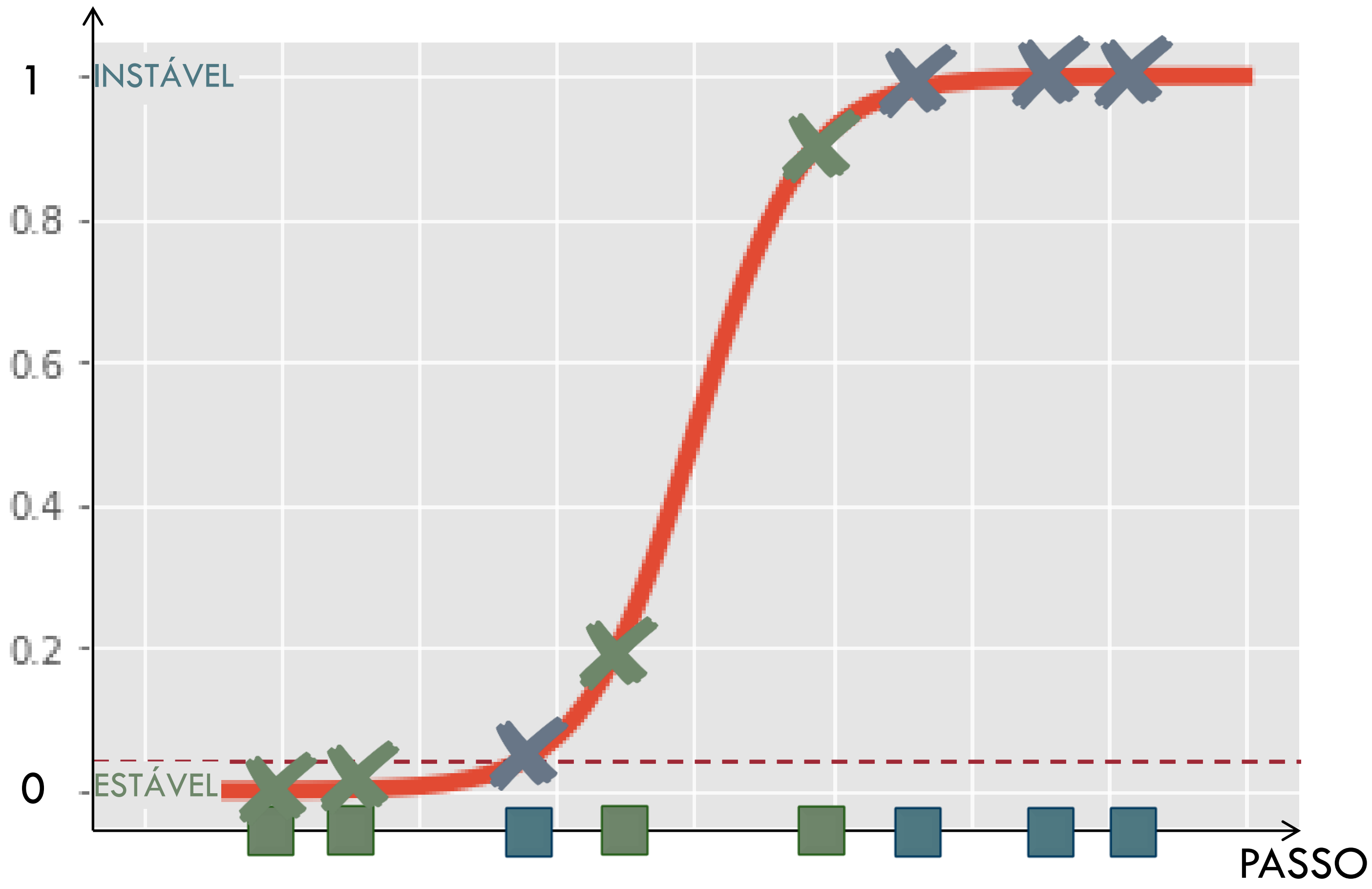
O segundo ponto é melhor que o primeiro...

$$R = \frac{VP}{VP + FN}$$

$$FPR = \frac{FP}{FP + VN}$$



E se colocarmos o limite de modo que os dois menores passos são classificados como estáveis?



			Target	
		VP	1	0
Predicted	1	4	2	FP
	0	0	2	
		FN	VN	

$$R = \frac{VP}{VP + FN} = \frac{4}{4 + 0} = 1$$

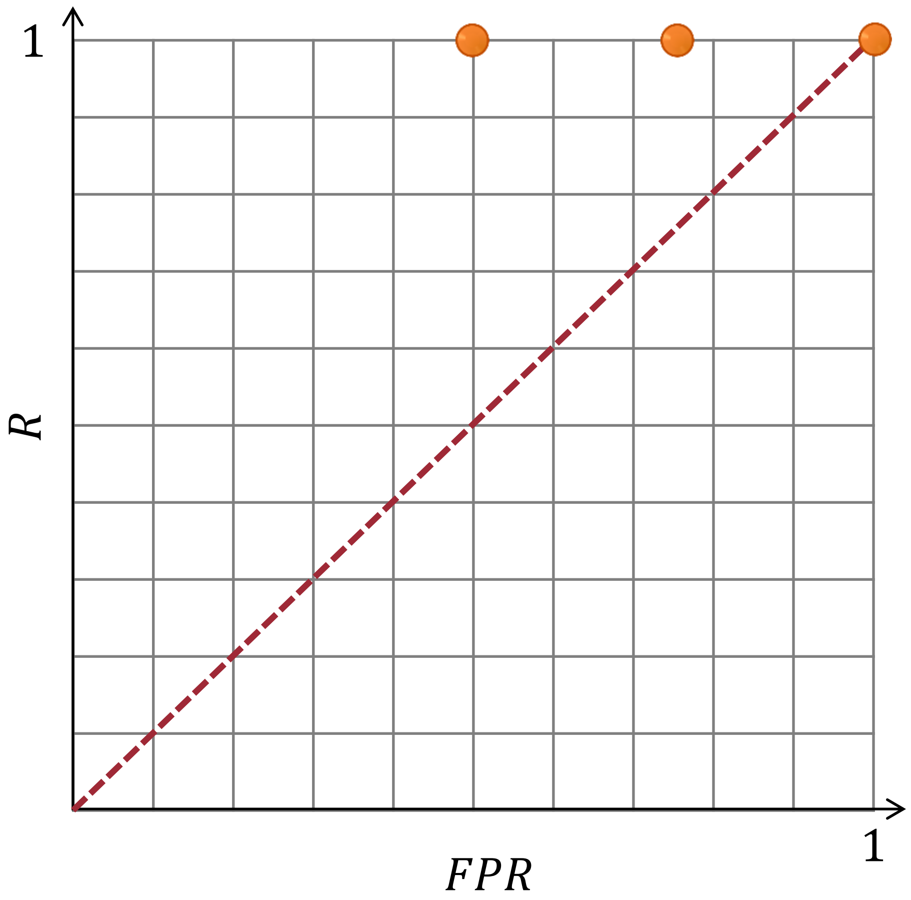
$$FPR = \frac{FP}{FP + VN} = \frac{2}{2 + 2} = 0.5$$

ROC: R vs FPR

O novo ponto (0.5,1) está ainda mais à esquerda da linha vermelha e, portanto, diminuiu a proporção de classificados incorretamente como instáveis (FP).

$$R = \frac{VP}{VP + FN}$$

$$FPR = \frac{FP}{FP + VN}$$



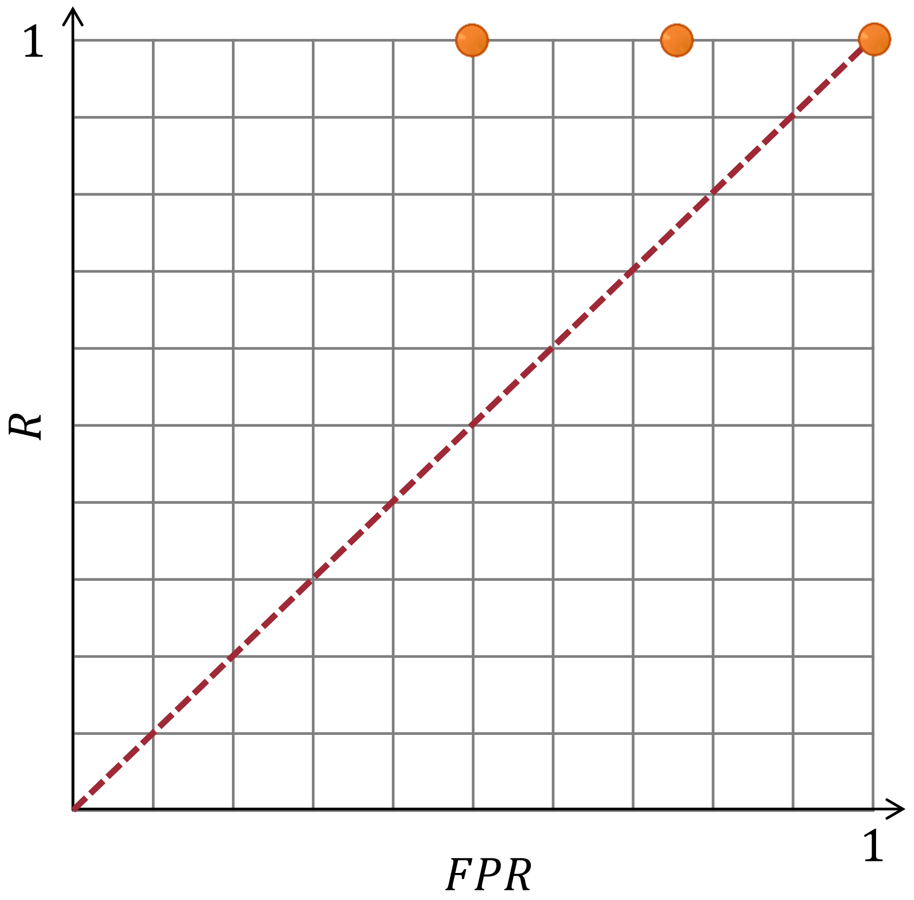
Ou seja,

ROC: R vs FPR

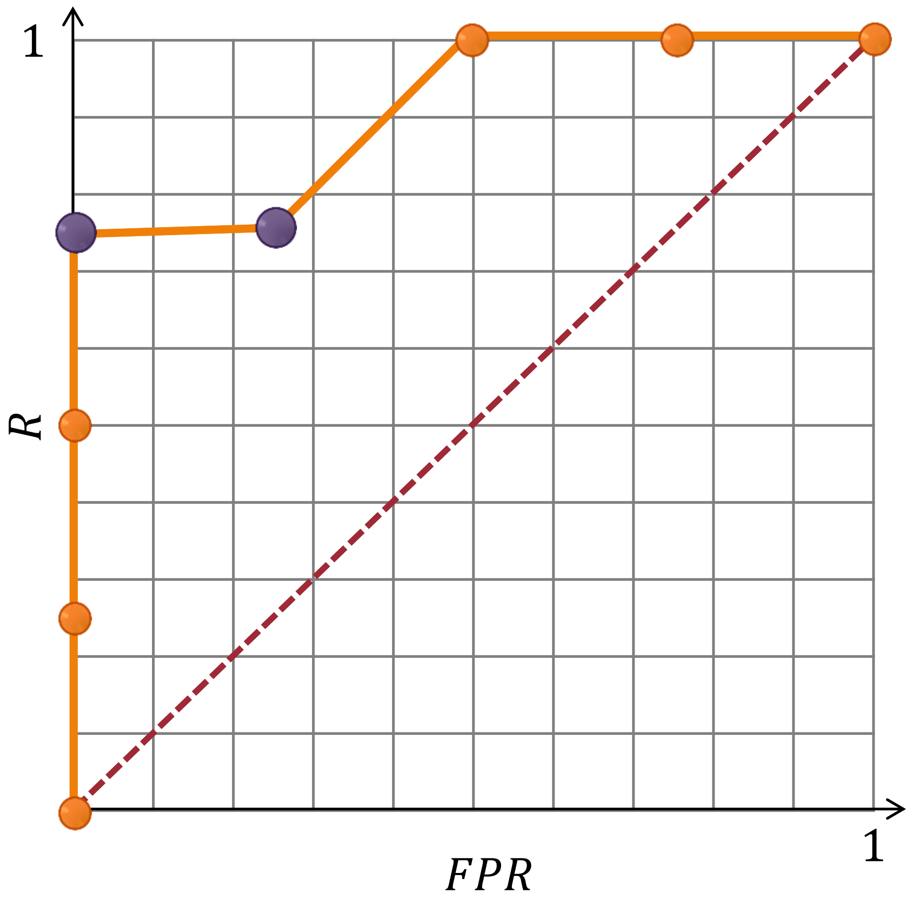
O terceiro ponto é melhor que o segundo...

$$R = \frac{VP}{VP + FN}$$

$$FPR = \frac{FP}{FP + VN}$$

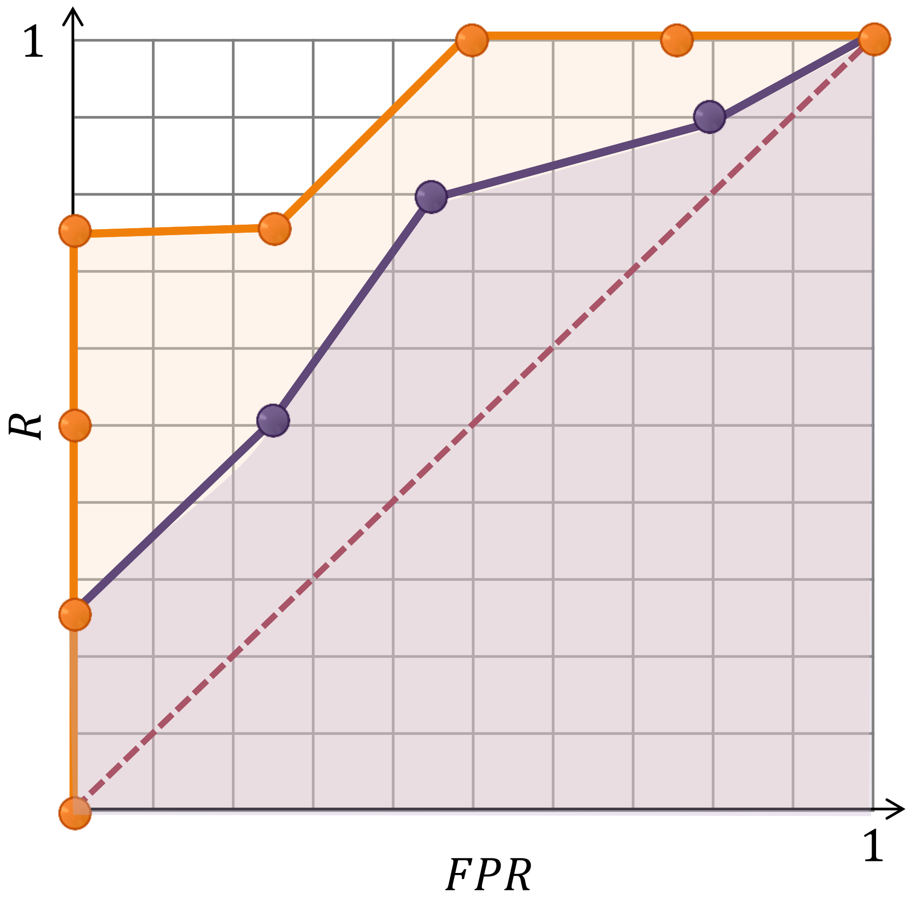


ROC: R vs FPR



Curva ROC resume a matriz de confusão de cada limite escolhido.

AUC



A AUC torna fácil a comparação entre diferentes ROC.

A curva laranja tem AUC maior que a curva roxa, sugerindo que é melhor.

OBSERVAÇÃO

Se as amostras são desbalanceadas (por exemplo, o número de passos estáveis é muito maior que não estáveis) então PRECISÃO pode ser mais útil que FPR.

$$FPR = \frac{FP}{FP + VN}$$

Isso ocorre porque a precisão, definida como,

$$P = \frac{VP}{VP + FP}$$

não inclui o número de VN em seu cálculo e, portanto, não é afetada pelo desbalanceamento.

REGRESSÃO LOGÍSTICA PARA MAIS DE DUAS CLASSES

SOFTMAX

Softmax é uma função (de ativação!) que transforma números/logits em probabilidades. A saída de um Softmax é um vetor com probabilidades de cada resultado possível que somam um para todos os resultados ou classes possíveis.

É uma generalização da função logística para múltiplas dimensões, e usada na regressão logística multinomial.

K: número de classes

$$h_{\omega^{(k)}}(\boldsymbol{x}) = \frac{e^{\boldsymbol{\omega}^{(k)} \cdot \boldsymbol{x}}}{\sum_{i=1}^{K} e^{\boldsymbol{\omega}^{(i)} \cdot \boldsymbol{x}}}$$

REGRESSÃO SOFTMAX

$$h_{\omega^{(k)}}(\boldsymbol{x}) = \frac{e^{\boldsymbol{\omega}^{(k)}\cdot\boldsymbol{x}}}{\sum_{i=1}^{K} e^{\boldsymbol{\omega}^{(i)}\cdot\boldsymbol{x}}}$$

$$h_\omega(\mathbf{x}) = \begin{bmatrix} P(y=1|\mathbf{x};\boldsymbol{\omega}) \\ P(y=2|\mathbf{x};\boldsymbol{\omega}) \\ \vdots \\ P(y=K|\mathbf{x};\boldsymbol{\omega}) \end{bmatrix} = \frac{1}{\sum_{j=1}^{K} \exp(\boldsymbol{\omega}^{(j)\top}\mathbf{x})} \begin{bmatrix} \exp(\boldsymbol{\omega}^{(1)\top}\mathbf{x}) \\ \exp(\boldsymbol{\omega}^{(2)\top}\mathbf{x}) \\ \vdots \\ \exp(\boldsymbol{\omega}^{(K)\top}\mathbf{x}) \end{bmatrix}$$

$$J(\boldsymbol{\omega}) = -\left[\sum_{i=1}^{m}\sum_{k=1}^{K} 1\left\{y^{(i)} = k\right\} \log \frac{\exp(\boldsymbol{\omega}^{(k)\top}\mathbf{x}^{(i)})}{\sum_{j=1}^{K} \exp(\boldsymbol{\omega}^{(j)\top}\mathbf{x}^{(i)})}\right]$$

$1\{\cdot\} \longrightarrow$ Função indicativa

INPUT

SOFTMAX

$$S(y)_i = \frac{\exp(y_i)}{\sum_{j=1}^{n} \exp(y_j)}$$

LOGITS	$p(i)$	CLASSE
3.2	0.775	Coala
1.3	0.116	Gato
0.2	0.039	Cachorro
0.8	0.070	Pintinho

3 1 2 0 3 2 0 1

ONE-VS-ALL (ONE-VS-REST)

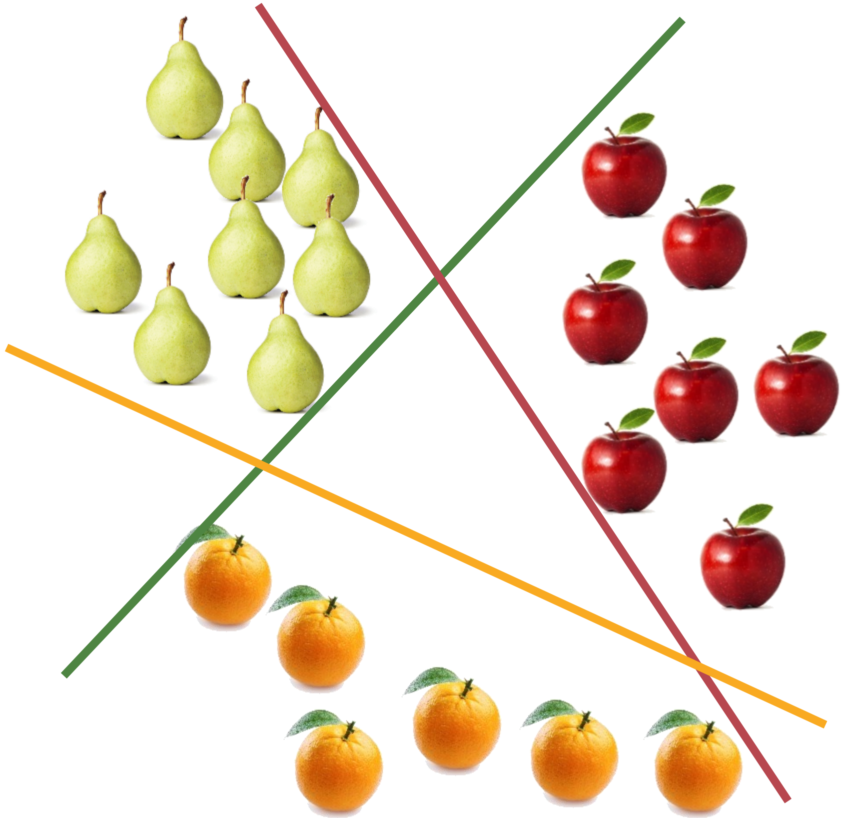
$$y \in \{0,1,2,\dots,K\}$$

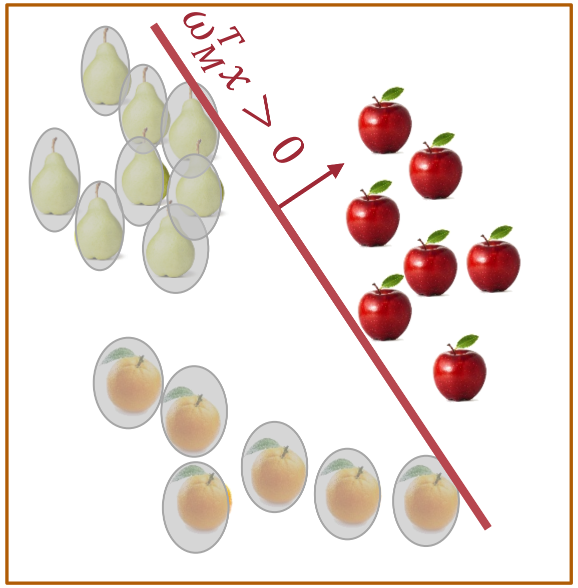
Treinamos K classificadores binário separado para cada classe e executamos todos esses classificadores. Para qualquer novo exemplo $\boldsymbol{x}$ que desejamos prever escolhemos a classe com a pontuação máxima:

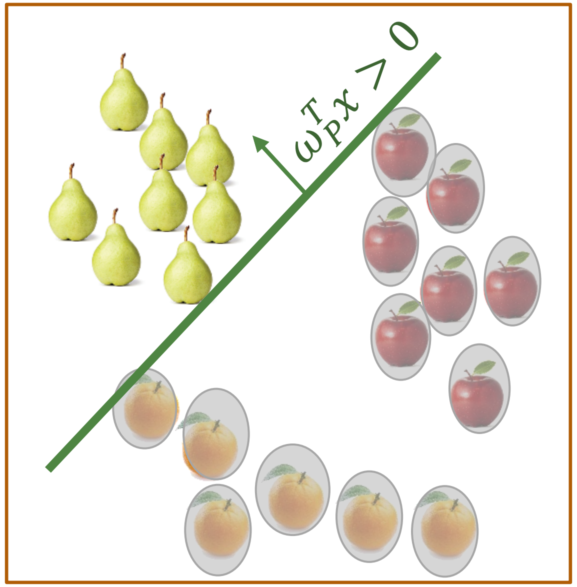
$$\hat{y} = \arg \max_{k \epsilon \{1,2,\dots,K\}} h_\omega^{(k)}(\boldsymbol{x}).$$

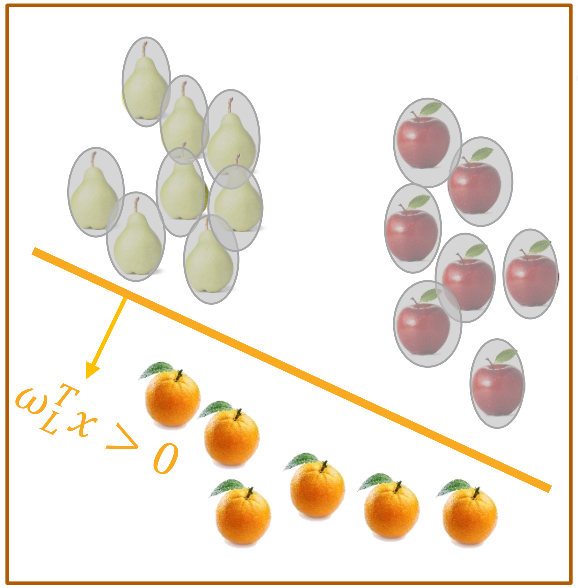
$$h_\omega^{(0)}(x) = P(y=0|x;\omega)$$
$$h_\omega^{(1)}(x) = P(y=1|x;\omega)$$
$$\cdots$$
$$h_\omega^{(C)}(x) = P(y=C|x;\omega)$$

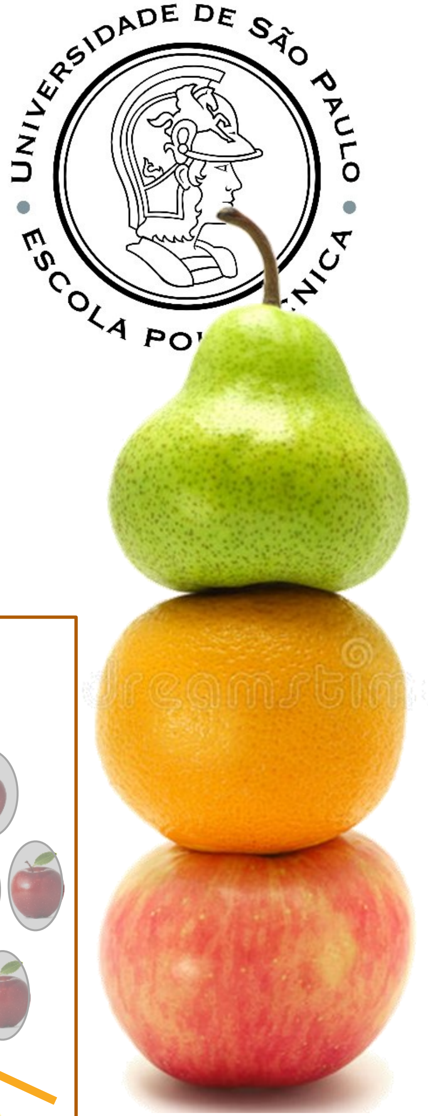
ONE-VS-ALL











ONE-VS-ONE

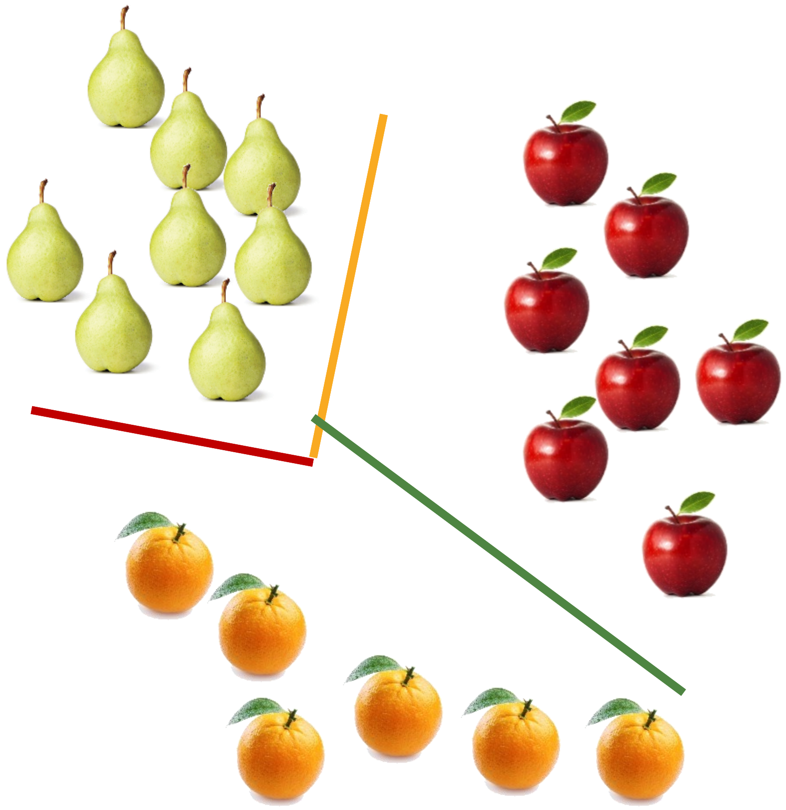
Treinamos

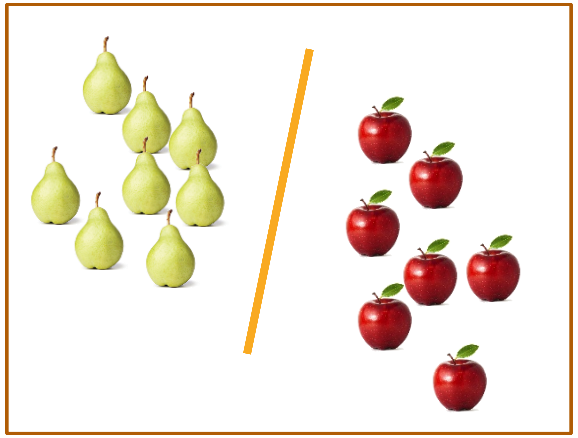
$$\binom{K}{2} = \frac{K(K-1)}{2}$$

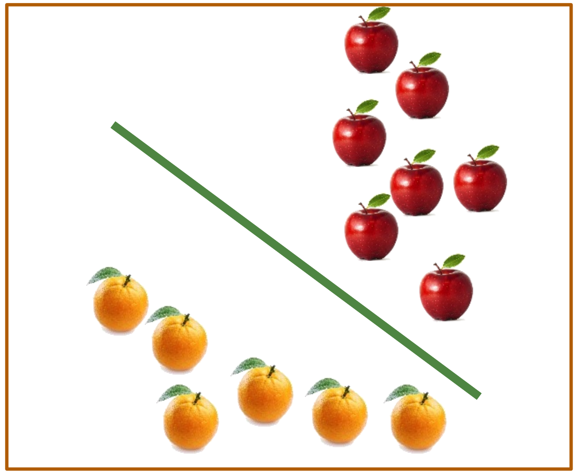
modelos de classificação binária separados. Para qualquer novo exemplo $\boldsymbol{x}$ que desejamos prever escolhemos a classe com a pontuação máxima:

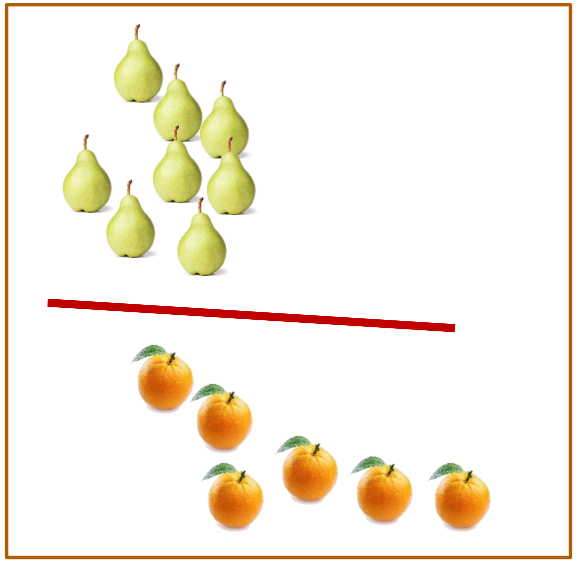
$$\hat{y} = \arg \max_{k \epsilon \{1,2,\ldots,K\}} h_{\omega}^{(k)}(\boldsymbol{x})$$

ONE-VS-ONE









ENOUGH IS ENOUGH!

ACABOU...