

# Classificação Multirrótulo

1001513 – Aprendizado de Máquina 2  
Turma A – 2023/2  
Prof. Murilo Naldi



[naldi@ufscar.br](mailto:naldi@ufscar.br)



# Agradecimentos

- Pessoas que colaboraram com a produção deste material: Diego Silva e Ricardo Cerri
- Intel IA Academy

# Classificação binária vs. multiclasse

Um problema de classificação **binária** possui **um atributo alvo**

- Ele só pode conter **dois valores** (geralmente, verdadeiro ou falso)

Um problema de classificação **multiclasse** possui **um atributo alvo**

- Podemos atribuir  **$n_c$  valores** a esse atributo ( $n_c > 2$ )



# Classificação **binária** vs. **multiclasse**

$atr_1$	...	$atr_m$	classe
	...		0
	...		1
	...		1
	...		0

$atr_1$	...	$atr_m$	classe
	...		0
	...		3
	...		0
	...		5



# Classificação **binária** vs. **multiclasse**

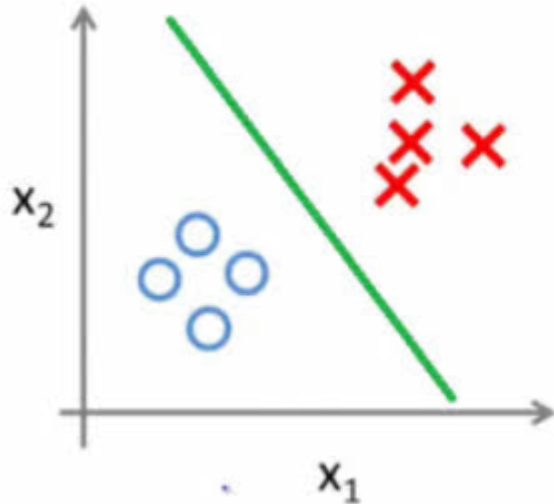
$atr_1$	...	$atr_m$	classe
	...		gato
	...		rato
	...		rato
	...		gato

$atr_1$	...	$atr_m$	classe
	...		rato
	...		gato
	...		cachorro
	...		cachorro



# Classificação **binária** vs. **multiclasse**

Classificação Binária



Classificação Multiclasse

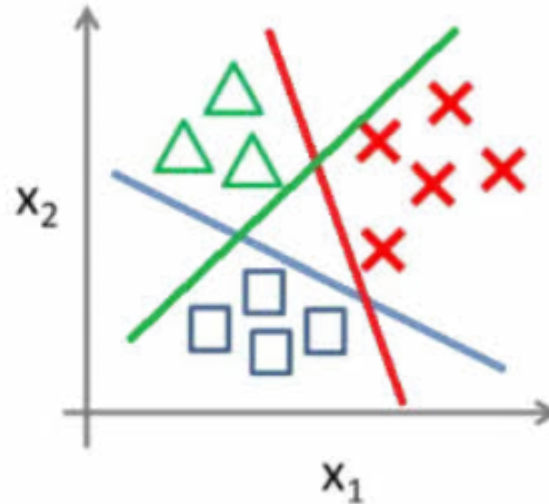



Photo via utkuufuk.com

# Classificação binária vs. multiclasse

Nem todo algoritmo lida com classificação multiclasse

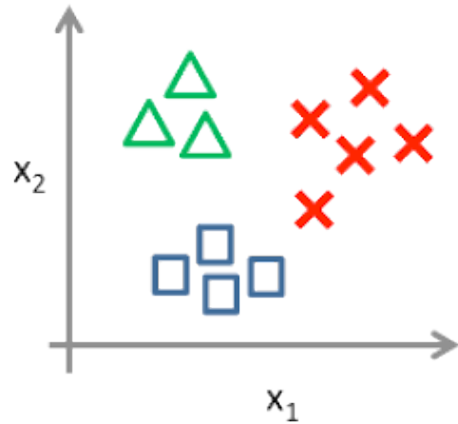
- SVM é um exemplo

E agora, José?

- One-vs-all
  - One-vs-One (All-vs-all)
  - Codificadores (ex. ECOC)
- 

# Classificação binária vs. multiclasse

One-vs-all (one-vs-rest):



Class 1: **Green**

Class 2: **Blue**

Class 3: **Red**

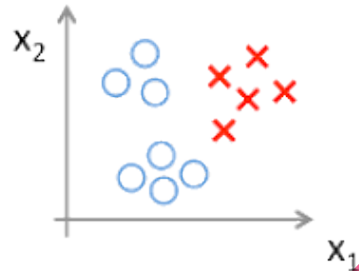
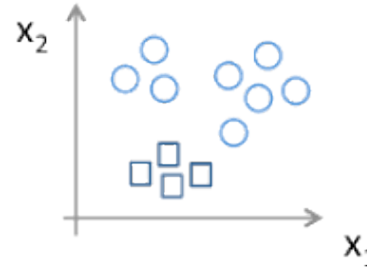
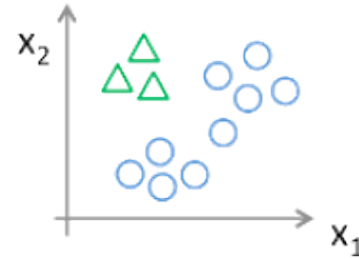


Photo via cc.gatech.edu



# Classificação binária vs. multiclasse

## Visualizing One-vs-all

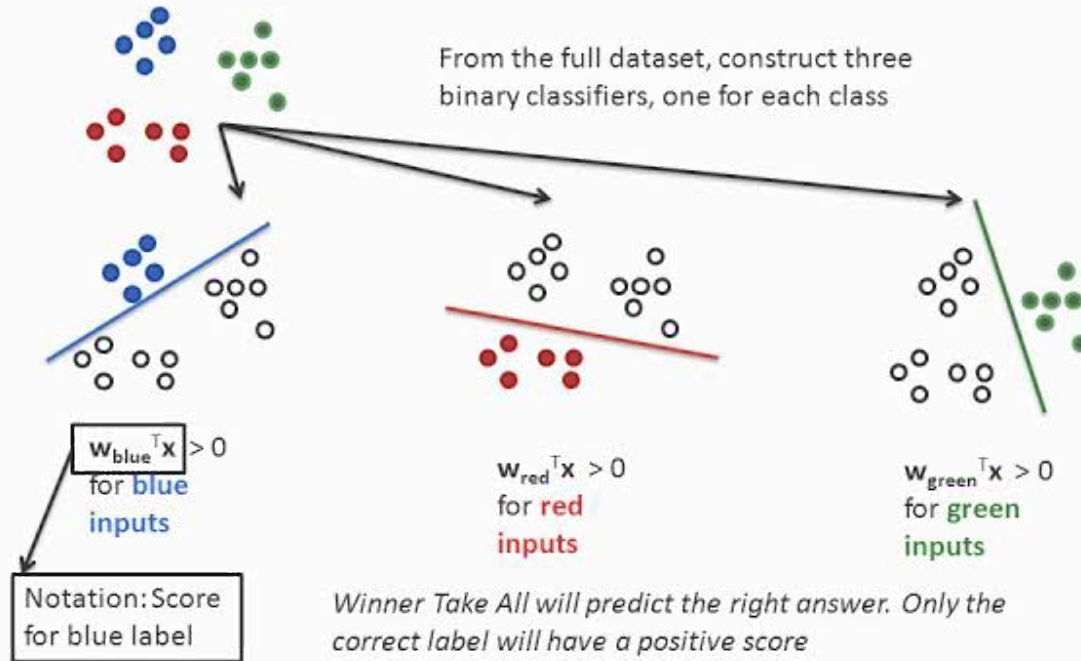


Photo via SlidePlayer.com

# Classificação binária vs. multiclasse

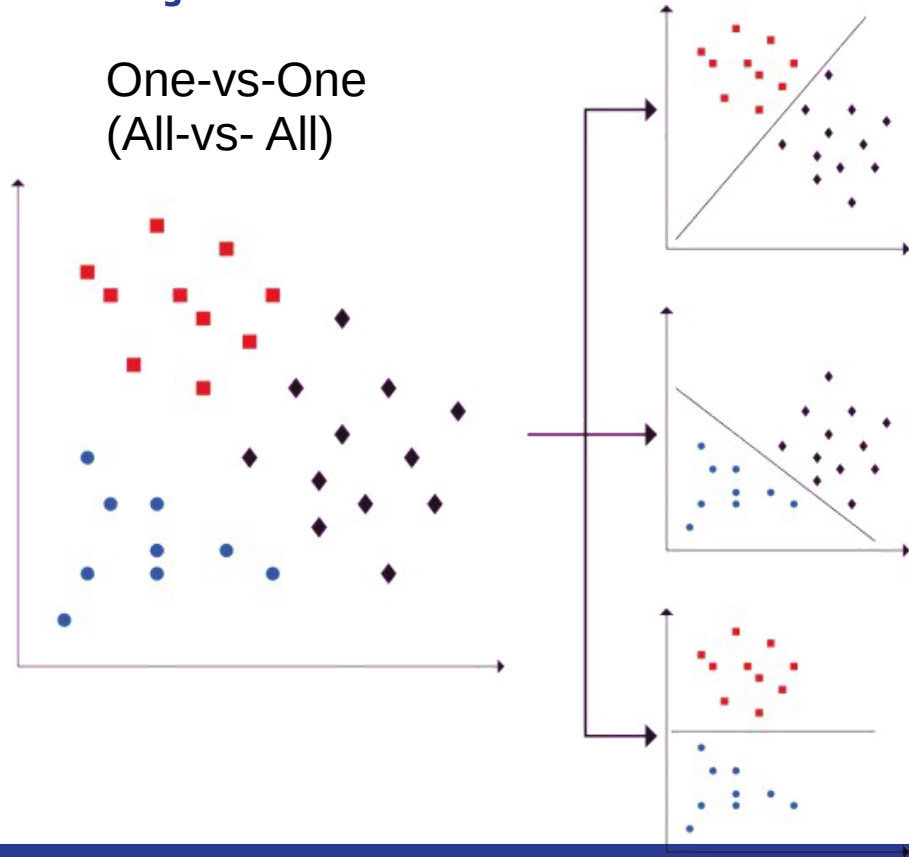


Photo via ScienceDirect.com

# Classificação binária vs. multiclasse

## *Error Correcting Output Codes (ECOC)*

- A base do ECOC é decompor um problema multiclasse em um número maior de problemas binários.
- Cada classificador é treinado em um problema de duas metaclasses
  - onde cada meta-classe consiste em algumas combinações das classes originais
- Duas fases:
  - *Enconding e deconding*



# Classificação binária vs. multiclasse

## *Error Correcting Output Codes (ECOC)*

- *Encoding* gera uma matriz de decomposição
  - sequência de bits representando cada classe, onde cada bit identifica a associação da classe a um classificador
- *Decoding* faz cada classificador binário lançar um voto para uma das duas metaclasses usadas no treinamento
  - Resultado é colocado em um vetor e comparado com as classes na matriz de decomposição



# Outros tipos de problemas

## Classificação **multirrótulo**

- Quando um mesmo exemplo pode ter mais de um rótulo associado a ele ao mesmo tempo

## Regressão **multiobjetivo**

- Similar à classificação multirrótulo, mas para regressão



# Outros tipos de problemas

## Classificação **hierárquica**

- A saída está relacionada a uma posição em uma taxonomia

## Classificação **multi-saída**

- Vários problemas multi-classe



# Classificação multirrótulo

## Exemplos práticos

- Uma música (ou um filme) pode ter mais de um gênero
  - Além de outras tags



- Uma foto pode conter vários objetos
- Uma proteína pode ter mais de uma função
- Um artigo científico pode ser multidisciplinar

# Classificação multirrótulo

## Transformação do **problema**

- Mudamos a “cara” do problema
  - Mudando os rótulos para **monorrótulo**/simples-rótulo
  - Transformando os rótulos (ex: criar novas classes)
  - Removendo exemplos
  - Removendo rótulos
  - Decomposição (aditiva e multiplicativa)
  - ...
- “Abordagem independente de algoritmo”





# Classificação multirrótulo - transformação dos rótulos

## Relevância binária (BR)

- Se temos  $k$  possíveis rótulos para cada exemplo, **criamos  $k$  classificadores binários**
- Cada um desses classificadores é treinado para prever se cada exemplo possui ou não um rótulo específico associado a ele



# Classificação multirrótulo - transformação dos rótulos

Atributos	Rótulo 1	Rótulo 2	Rótulo 3	Rótulo 4	Rótulo 5
ex1	0	0	1	0	1
ex2	1	0	1	1	0
...					



# Classificação multirrótulo - transformação dos rótulos

Atributos	Rótulo 1	Rótulo 2	Rótulo 3	Rótulo 4	Rótulo 5
ex1	0	0	1	0	1
ex2	1	0	1	1	0
...					



Atributos	Rótulo 1
ex1	0
ex2	1
...	

Atributos	Rótulo 2
ex1	0
ex2	0
...	

...

Atributos	Rótulo 5
ex1	1
ex2	0
...	

# Classificação multirrótulo - transformação dos rótulos

## Relevância binária (BR)

- Costuma causar **desbalanceamento**
- Considera **independência** entre os rótulos
  - Rótulos podem ser dependentes
  - Como considerar isso?



# Classificação multirrótulo - transformação dos rótulos

## Encadeamento de Classificadores (CC)

- Tenta resolver o problema da (possível) dependência entre as classes, mas...



# Classificação multirrótulo - transformação dos rótulos

Atributos	Rótulo 1	Rótulo 2	Rótulo 3	Rótulo 4	Rótulo 5
ex1	0	0	1	0	1
ex2	1	0	1	1	0
...					



# Classificação multirrótulo - transformação dos rótulos

Atributos	Rótulo 1	Rótulo 2	Rótulo 3	Rótulo 4	Rótulo 5
ex1	0	0	1	0	1
ex2	1	0	1	1	0
...					



Atributos	Rótulo 1
ex1	0
ex2	1
...	



# Classificação multirrótulo - transformação dos rótulos

Atributos	Rótulo 1	Rótulo 2	Rótulo 3	Rótulo 4	Rótulo 5
ex1	0	0	1	0	1
ex2	1	0	1	1	0
...					



Atributos	Rótulo 1
ex1	
ex2	
...	



Atributos	Rótulo 1	Rótulo 2
ex1	0	
ex2	1	
...		



...

**E qual é a relação de dependência entre os rótulos??**

modelo1

modelo2





# Classificação multirrótulo - transformação dos rótulos

## **Label Powerset (LP)**

- Pode ser uma boa opção quando temos poucos rótulos
  - e/ou poucas combinações entre eles
- Número alto de rótulos aumenta muito o número de classes possíveis, tornando o problema difícil



# Classificação multirrótulo - transformação dos rótulos

Atributos	Rótulo 1	Rótulo 2	Rótulo 3	Rótulo 4	Rótulo 5
ex1	0	0	1	0	1
ex2	1	0	1	1	0
ex3	1	0	0	1	0
ex4	0	0	0	1	1
...					



# Classificação multirrótulo - transformação dos rótulos

Atributos	Rótulo 1	Rótulo 2	Rótulo 3	Rótulo 4	Rótulo 5	Classe
ex1	0	0	1	0	1	5
ex2	1	0	1	1	0	22
ex3	1	0	0	1	0	18
ex4	0	0	0	1	1	3
...						



# Dica

Esses (e os próximos) algoritmos estão implementados no scikit-multilearn



<http://scikit.ml/api/skmultilearn.html>



# Classificação multirrótulo - adaptação de algoritmos

## Algoritmos adaptados

- A ideia não é mais transformar os rótulos, mas sim usar um **classificador específico para o cenário multirrótulo**
- Usualmente, adaptações simples de algoritmos “clássicos”
- DT, SVM, kNN...



# Classificação multirrótulo - adaptação de algoritmos

## ML kNN

- Calcula as probabilidades *a priori* dos rótulos
- Encontra os  $k$  vizinhos mais próximos
- Usando os rótulos desses vizinhos e as *a priori*
  - Aplica regra de Bayes
  - Maximum a Posteriori (MAP)



# Classificação multirrótulo - adaptação de algoritmos

## **RNAs**

- Mais fácil desenhando :)



# Classificação multirrótulo - baseados em *ensemble*

## **RAKEL** (RANdom k-LabELsets)

- “Meio-termo” entre BR e LP
- Escolhe *k-labelsets* (subconjunto de *k* rótulos) aleatórios
- Para cada *labelset*, executa um LP
- Com a regra da média, determina um *score* para cada rótulo
  - Que é positivo ou negativo depende de um *threshold* definido pelo usuário





# Classificação multirrótulo - avaliação

Não podemos utilizar as medidas de avaliação estudadas até agora

- Mas podemos usar algumas parecidas ou adaptadas



# Classificação multirrótulo - avaliação

## Hamming loss

$$HammingLoss(\hat{f}, \mathbf{X}) = \frac{1}{n} \sum_{i=1}^n \frac{a(\mathbf{y}_i, \mathbf{z}_i)}{k}$$

- Sendo  $a(\mathbf{y}_i, \mathbf{z}_i)$  a distância de Hamming entre os vetores verdadeiro e predito (soma do XOR).
- Quanto menor, melhor
  - Zero ocorre quando a classificação é perfeita.



# Classificação multirrótulo - avaliação

$$\text{acurácia}(\hat{f}, \mathbf{X}) = \frac{1}{n} \sum_{i=1}^n \frac{\mathbf{y}_i \text{AND} \mathbf{z}_i}{\mathbf{y}_i \text{OR} \mathbf{z}_i}$$

$$\text{precisão}(\hat{f}, \mathbf{X}) = \frac{1}{n} \sum_{i=1}^n \frac{\mathbf{y}_i \text{AND} \mathbf{z}_i}{\text{sum}(\mathbf{z}_i)}$$

$$\text{revocação}(\hat{f}, \mathbf{X}) = \frac{1}{n} \sum_{i=1}^n \frac{\mathbf{y}_i \text{AND} \mathbf{z}_i}{\text{sum}(\mathbf{y}_i)}$$



# Por hoje é só



Conjunto de rótulos da imagem:

- praia
- mar
- sol
- cadeira
- montanha
- queria
- muito