Tema 10

Outros tópicos em classificação supervisionada

Professora:
Ariane Machado Lima

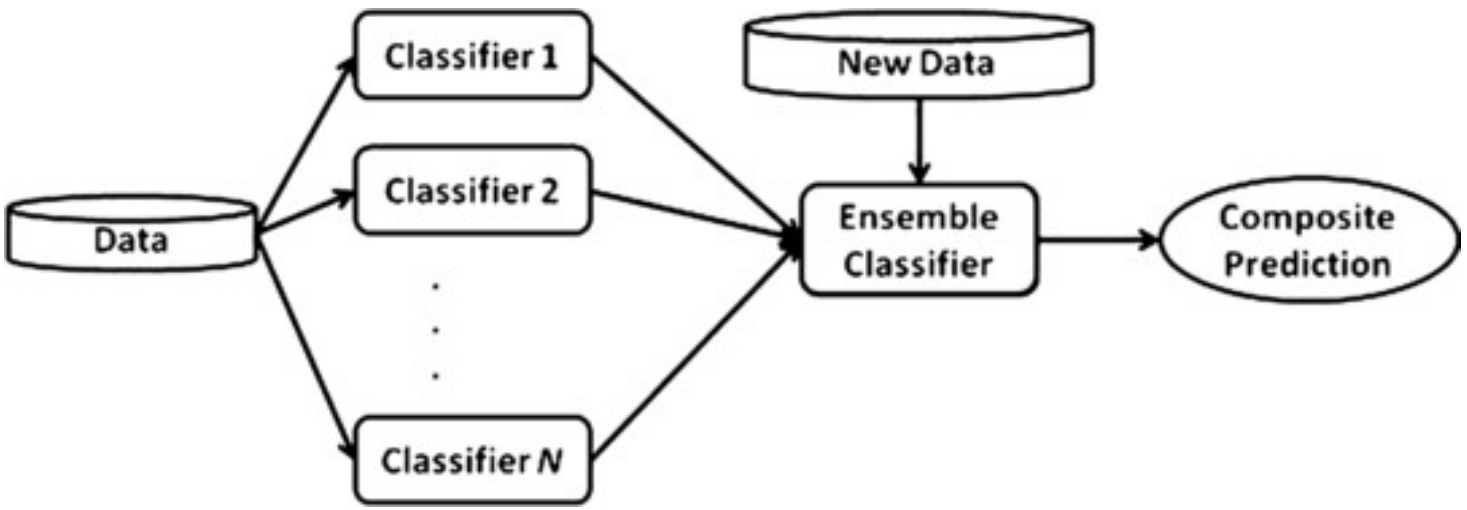
Outros tópicos

- Comitês de classificadores (ensembles)
- Classificação multiclasse
- Classificação multirrótulo

Vídeo 1

Comitê de classificadores

Comitês de classificadores



https://link.springer.com/chapter/10.1007/978-1-4471-4884-5_14

Comitê de classificadores

Vários possíveis motivos para querer combinar classificadores:

- Diversos classificadores baseados em diferentes aspectos do problema (multimodal)
  - Ex: classificação de uma pessoa por voz, face, escrita, genética, etc.

Comitê de classificadores

Vários possíveis motivos para querer combinar classificadores:

- Mais de uma amostra de treinamento, cada uma com diferentes características
  - Juntar tudo resultaria em muitos *missing values*

Comitê de classificadores

Vários possíveis motivos para querer combinar classificadores:

- Diferentes métodos de classificação com diferentes desempenhos em diferentes pontos do espaço do busca
  - Ex: classificação de documentos: textos curtos x textos longos (diferentes métodos melhores para cada um deles)

Comitê de classificadores

Vários possíveis motivos para querer combinar classificadores:

- Vários classificadores fracos formando um forte
  - *Random forests* seguem essa ideia

Comitê de classificadores

Esquema típico de configuração:

- Vários classificadores individuais

- Um combinador

O combinador coordena QUANDO chamar os classificadores e COMO combinar os resultados para dar a classificação final (média, votação, máximo, mínimo, produto, ...)

Comitê de classificadores

Esquema típico de configuração:

- Vários classificadores individuais
  - Diferentes algoritmos indutores ou não
  - Diferentes características ou não
  - Diferentes subamostras ou não
- Um combinador

O combinador coordena QUANDO chamar os classificadores e COMO combinar os resultados para dar a classificação final (média, votação, máximo, mínimo, produto, ...)

Comitê de classificadores

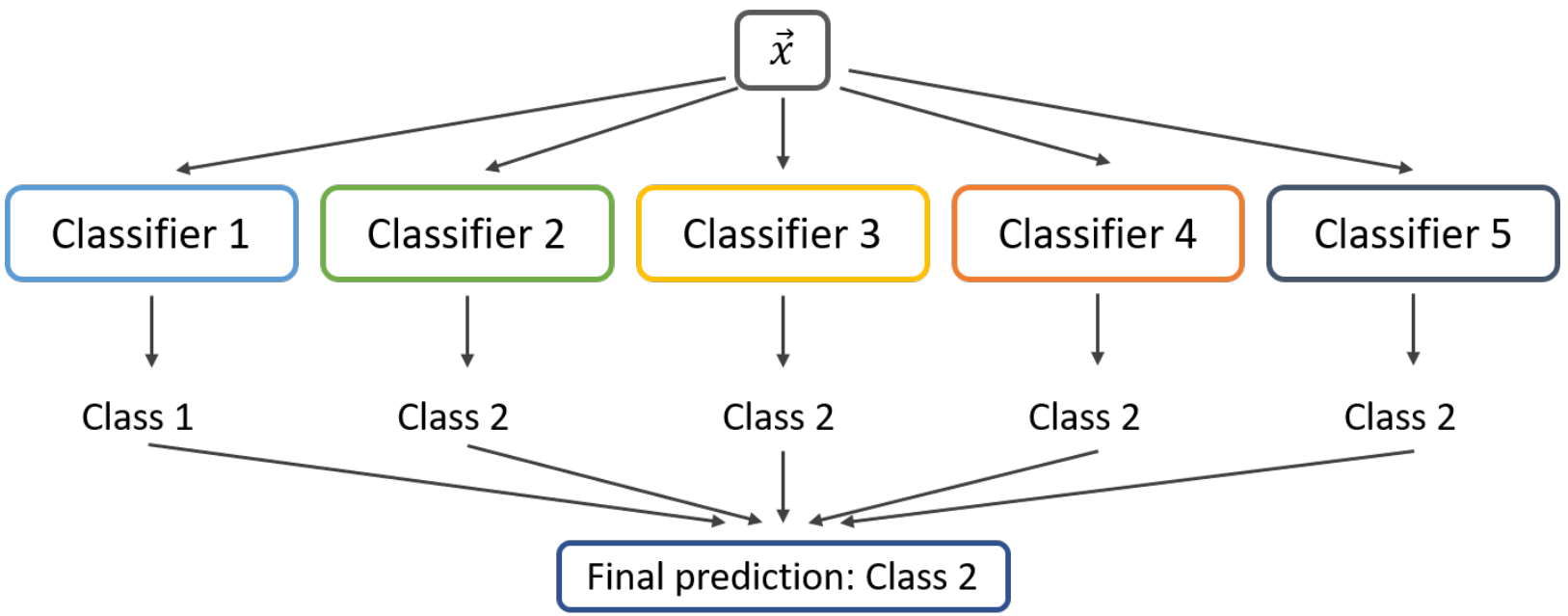
3 grandes arquiteturas:

- Paralela
- Cascata
- Hierárquica

Arquitetura Paralela

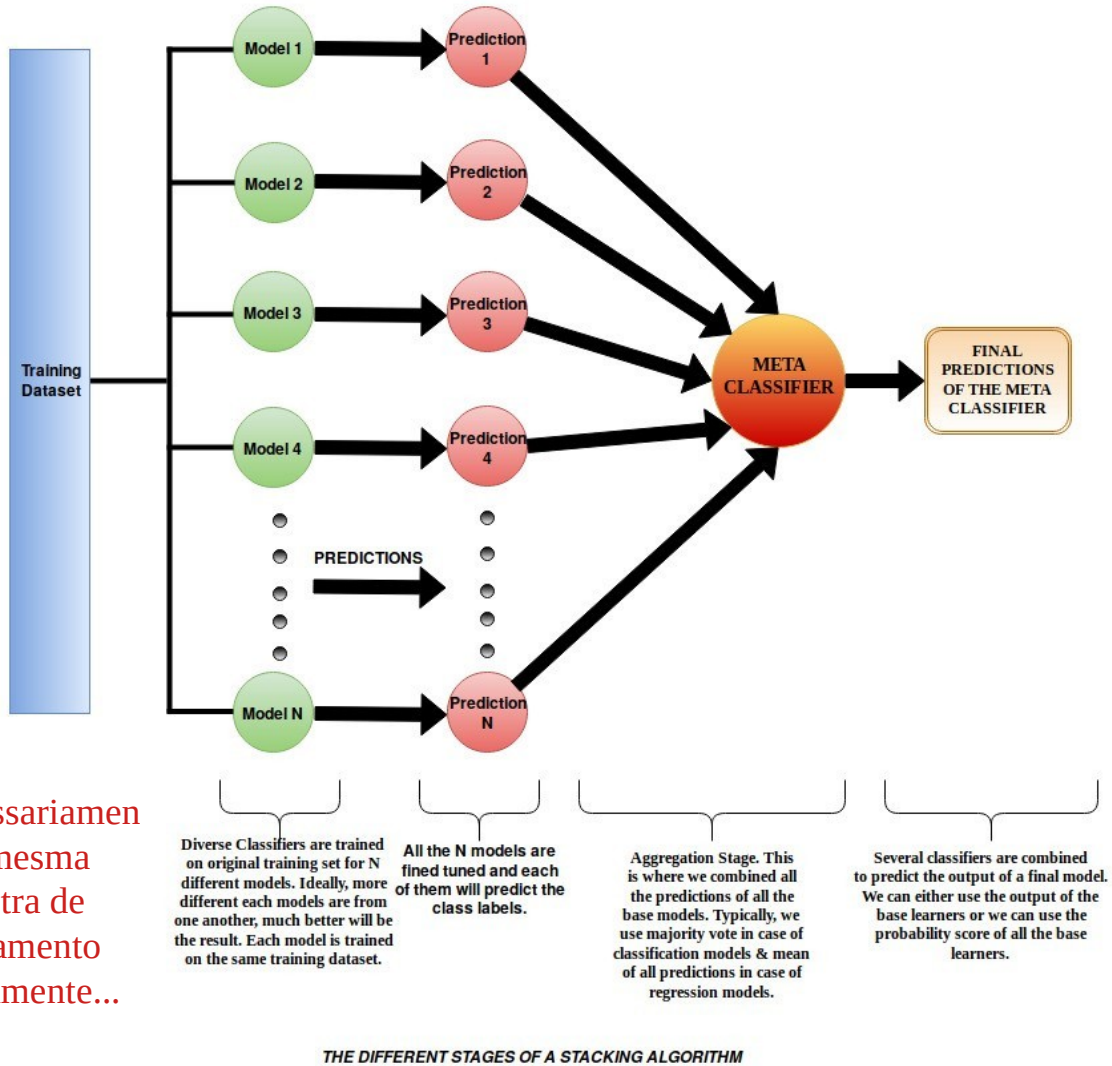
Os classificadores são chamados independentemente, e os resultados de todos são depois combinados

Vantagem: velocidade



https://www.kaggle.com/fengdanye/machine-learning-6-basic-ensemble-learning

Stacking (um tipo de comitê paralelo)



THE DIFFERENT STAGES OF A STACKING ALGORITHM

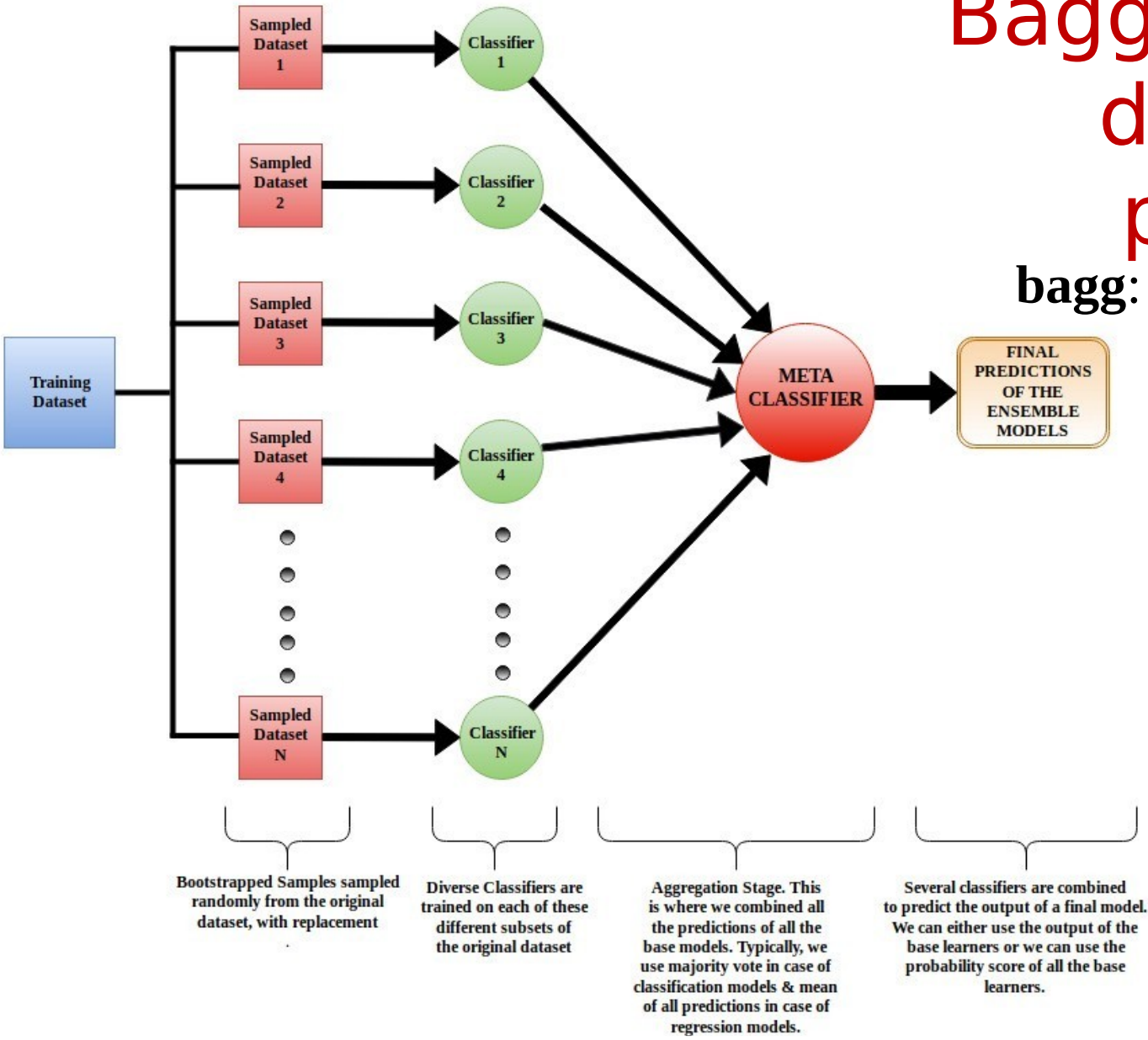
Modelos podem ser heterogêneos (quanto ao algoritmo de indução de classificadores) ou explorar diferentes aspectos do problema

Não necessariamente a mesma amostra de treinamento exatamente...

EACH

Bagging (um tipo de comitê paralelo)

bagg: **b**ootstrap **agg**regation



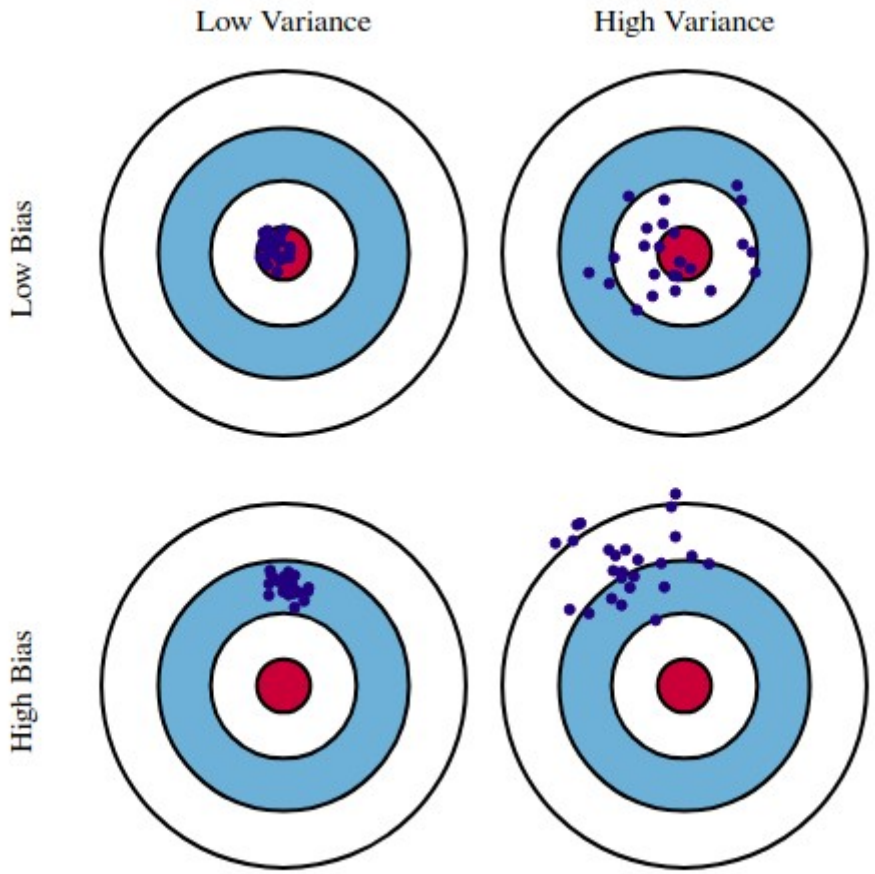
Ideia explorada pelas Random Forests

EACH

https://medium.com/@saugata.paul1010/ensemble-learning-bagging-boosting-stacking-and-cascading-classifiers-in-machine-learning-9c66cb271674

Bagging (um tipo de comitê paralelo)

bagg: **b**ootstrap **agg**regation



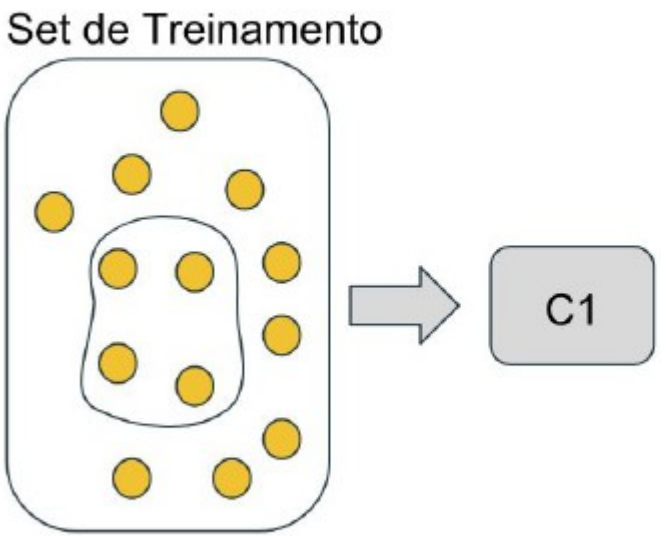
Objetivo: diminuir a variância, obtendo um classificador aggregado (o comitê) com menor erro)

http://scott.fortmann-roe.com/docs/BiasVariance.html

Boosting (paralelo na execução, cascata na construção)

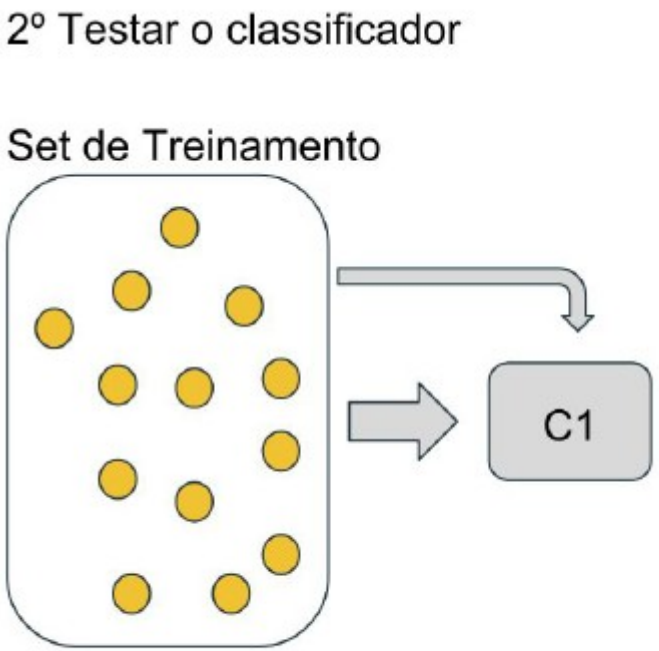
Segue a ideia do bagging, mas para obter classificadores com menor viés, direciona a reamostragem

1º Criar um classificador



Boosting (paralelo na execução, cascata na construção)

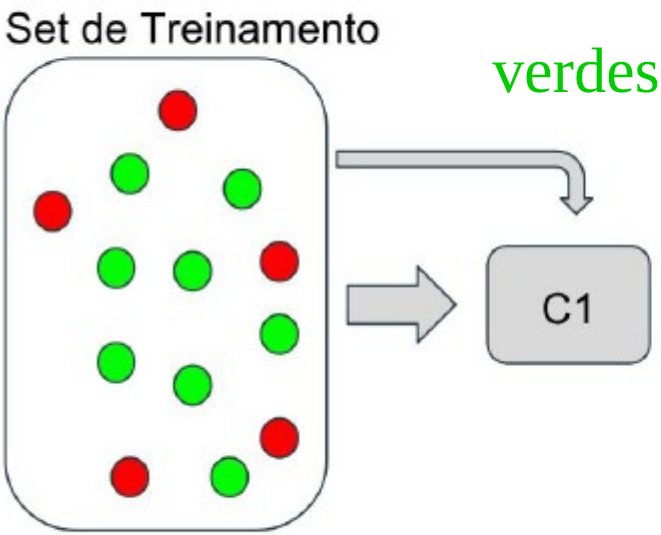
Segue a ideia do bagging, mas para obter classificadores com menor viés, direciona a reamostragem



Boosting (paralelo na execução, cascata na construção)

Segue a ideia do bagging, mas para obter classificadores com menor viés, direciona a reamostragem

2º Testar o classificador



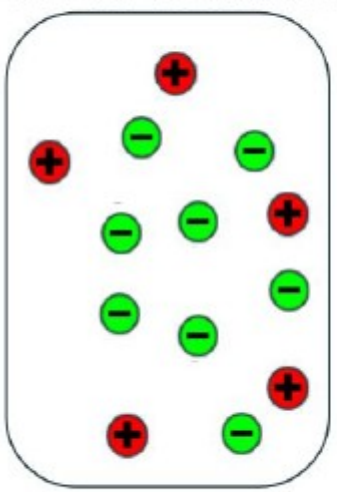
verdes = acertos; vermelhos = erros

Boosting (paralelo na execução, cascata na construção)

Segue a ideia do bagging, mas para obter classificadores com menor viés, direciona a reamostragem

3º Ajustar os pesos do Set

Set de Treinamento



verdes = acertos; vermelhos = erros

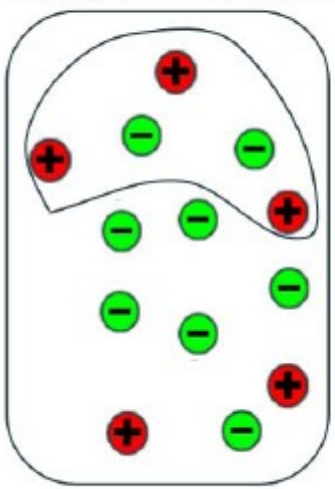
C1

Boosting (paralelo na execução, cascata na construção)

Segue a ideia do bagging, mas para obter classificadores com menor viés, direciona a reamostragem

4º Vamos repetir o processo até obter N classificadores

Set de Treinamento

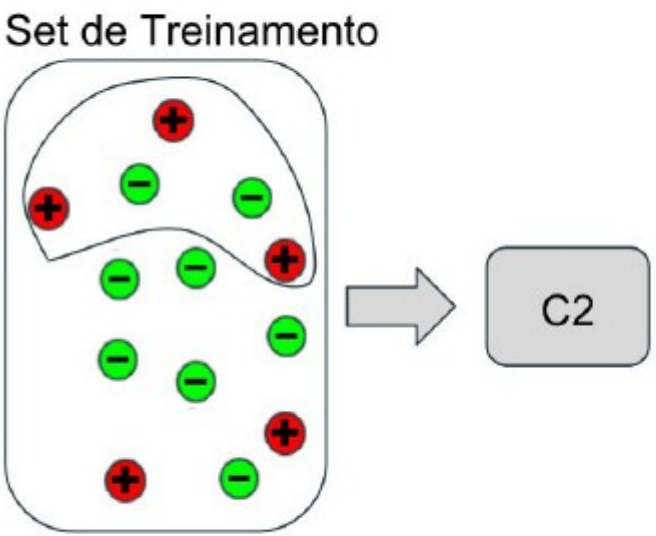


C1

Boosting (paralelo na execução, cascata na construção)

Segue a ideia do bagging, mas para obter classificadores com menor viés, direciona a reamostragem

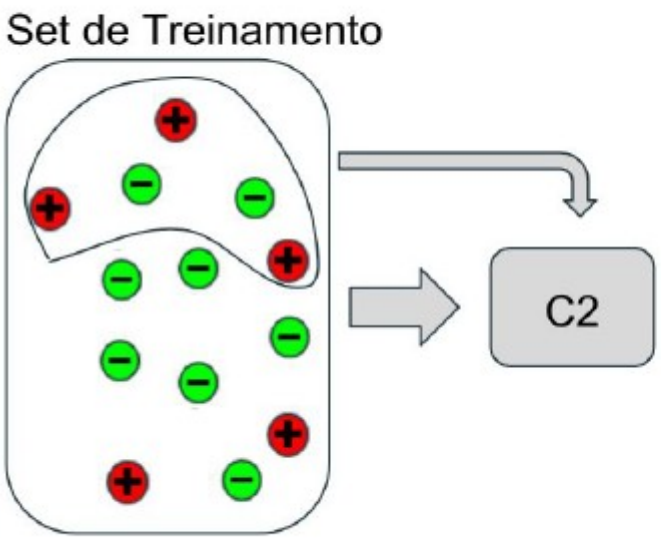
4º Vamos repetir o processo até obter N classificadores



Boosting (paralelo na execução, cascata na construção)

Segue a ideia do bagging, mas para obter classificadores com menor viés, direciona a reamostragem

4º Vamos repetir o processo até obter N classificadores



Boosting (paralelo na execução, cascata na construção)

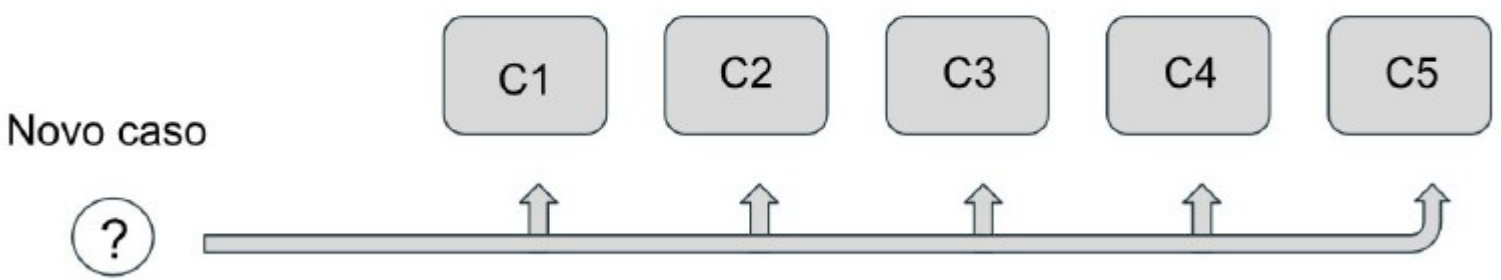
Segue a ideia do bagging, mas para obter classificadores com menor viés, direciona a reamostragem

4º Vamos repetir o processo até obter N classificadores

Boosting (paralelo na execução, cascata na construção)

Segue a ideia do bagging, mas para obter classificadores com menor viés, direciona a reamostragem

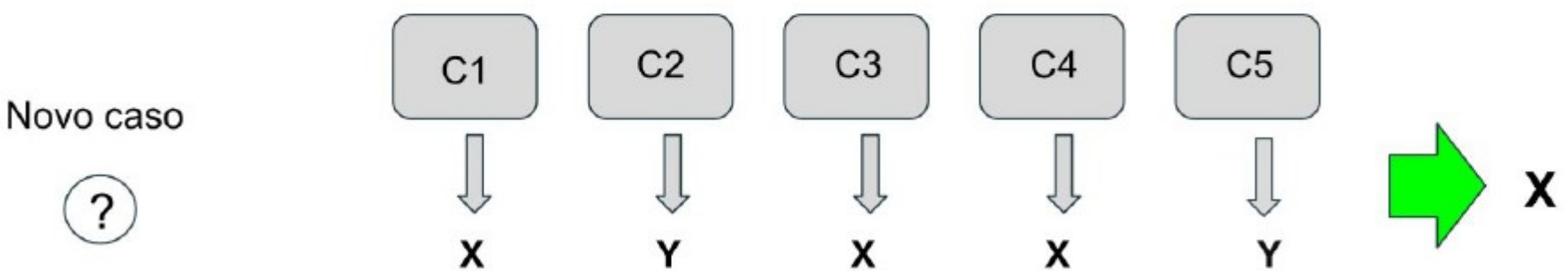
5º Com os N classificadores prontos, podemos começar a analisar casos novos



Boosting (paralelo na execução, cascata na construção)

Segue a ideia do bagging, mas para obter classificadores com menor viés, direciona a reamostragem

5º Com os N classificadores prontos, podemos começar a analisar casos novos



AdaBoost

Segue a ideia do boosting, mas:

- para cada classificador aprendido é calculado um peso Wt (com base em seus erros e acertos)
- o ajuste do peso de cada instância é adaptativo (aumenta ou diminui com base em Wt (multiplicando-se por e^{Wt} ou e^{-Wt})
- Resultado final é uma ponderação dos classificadores (usando Wt)

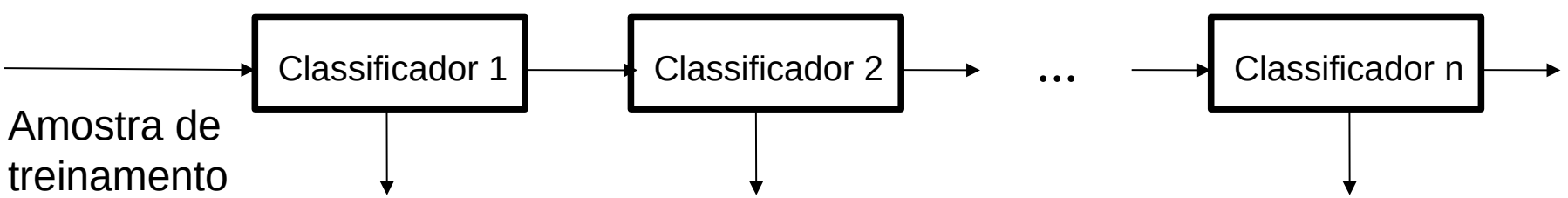
F_t (x) acertou?

F_t (x) = y?	W_t	Novo peso de ***a***	Resultado
Sim	2.3	$e^{-2.3} = 0.1$	Diminuímos o peso ***a***
Sim	0	$e^{0} = 1$	O peso ***a*** continua igual
Não	2.3	$e^{2.3} = 9.98$	Aumentamos o peso ***a***
Não	-2.3	$e^{-2.3} = 0.1$	Diminuímos o peso ***a***

$$\hat{y} = sign\left(\sum_{t=1}^{T} \hat{w}_t f_t(\mathbf{x})\right)$$

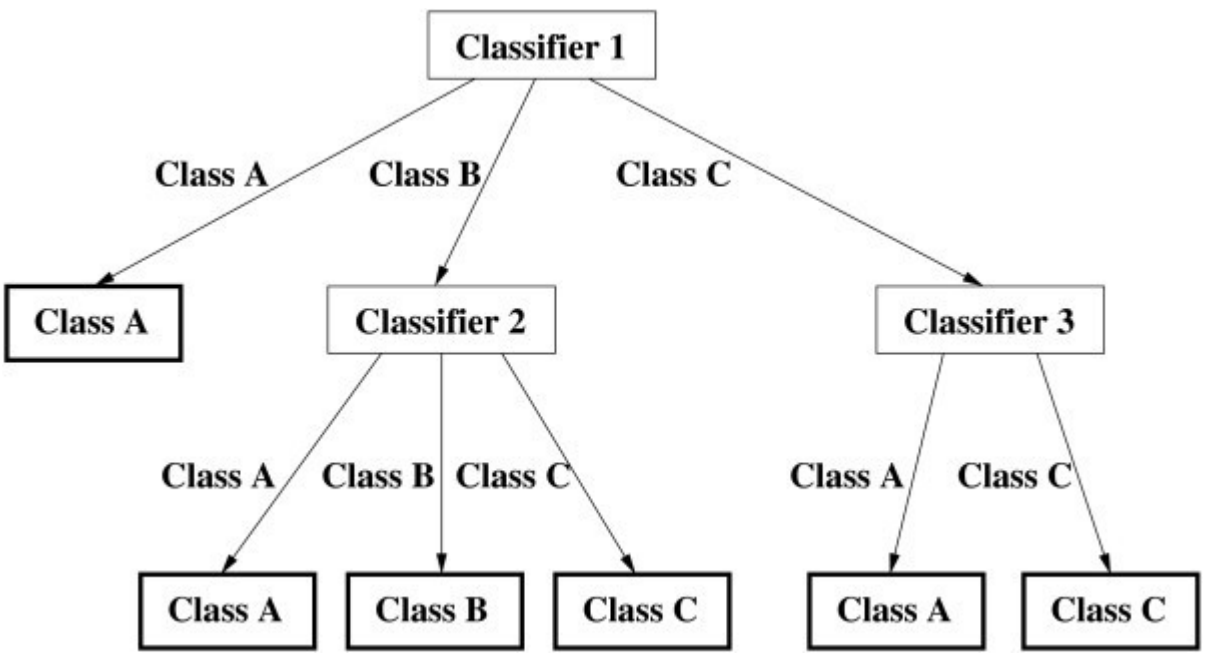
Arquitetura Cascata

- Classificadores são chamados um após o outro, em uma sequência linear
- Processamento sequencial (*pipeline*)
- Cada classificador vai eliminando possibilidades de classes
- Normalmente classificadores mais fracos e mais rápidos são executados primeiro



Arquitetura Hierárquica

- Estrutura semelhante a uma árvore de decisão, onde os nós são os classificadores
- Nem todos os classificadores são chamados
- Os classificadores de um caminho da raiz a uma folha são executados sequencialmente



BMC Bioinformatics 7(1):298

Comitê de classificadores – Arquiteturas Híbridas

- Por exemplo: hierarquia de módulos paralelos e/ou cascata

Fim do vídeo 1

Comitê de classificadores

Vídeo 2

Classificação multiclasse

Classificação Multiclasse

- Classificação multiclasse: dado um elemento, classificá-lo em 1 de M classes

- Muitos métodos de classificação são naturalmente multiclasse
- Outros possuem extensões para tratamento multiclasse
- Outros, mesmo destinados à classificação binária (apenas 2 classes) podem ser utilizados para classificação multiclasse

Classificação Multiclasse

- Classificação multiclasse: dado um elemento, classificá-lo em 1 de M classes

- Muitos métodos de classificação são naturalmente multiclasse
- Outros possuem extensões para tratamento multiclasse
- Outros, mesmo destinados à classificação binária (apenas 2 classes) podem ser utilizados para classificação multiclasse

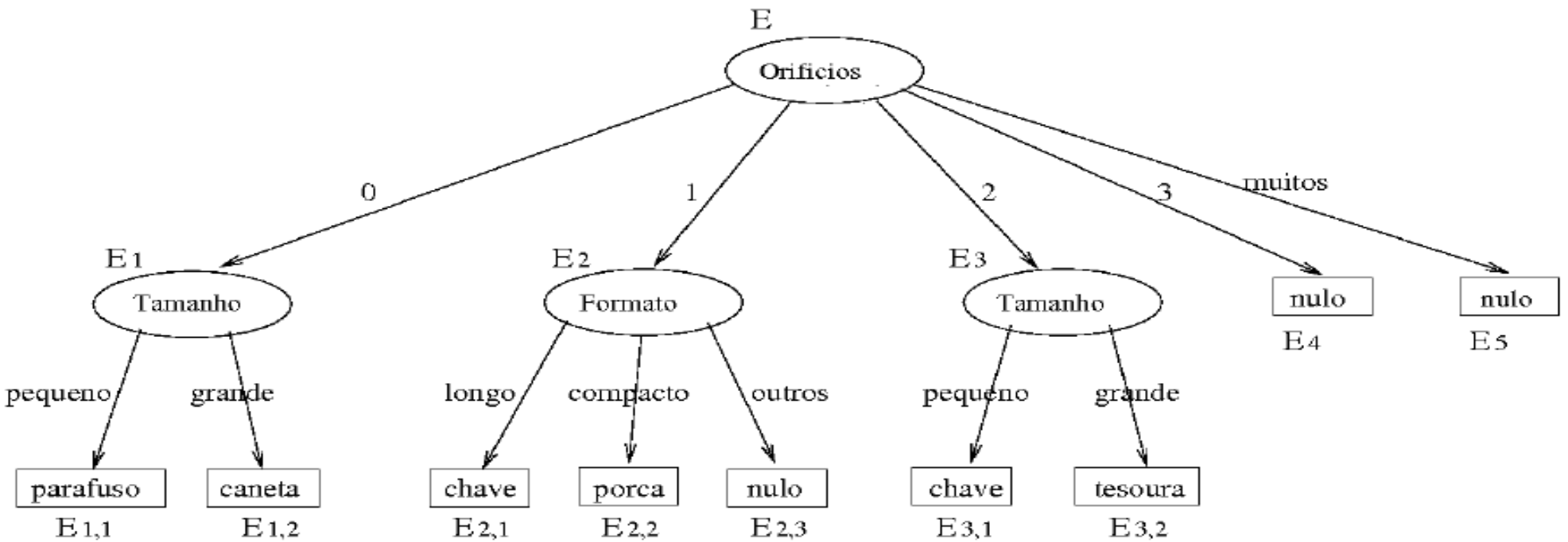
Métodos Multiclasse

- Técnicas de agrupamento (classificação não supervisionada) são aplicadas em classificação multiclasse

- Árvores de decisão / Random Forests
- Naive Bayes
- Redes Neurais
- kNN

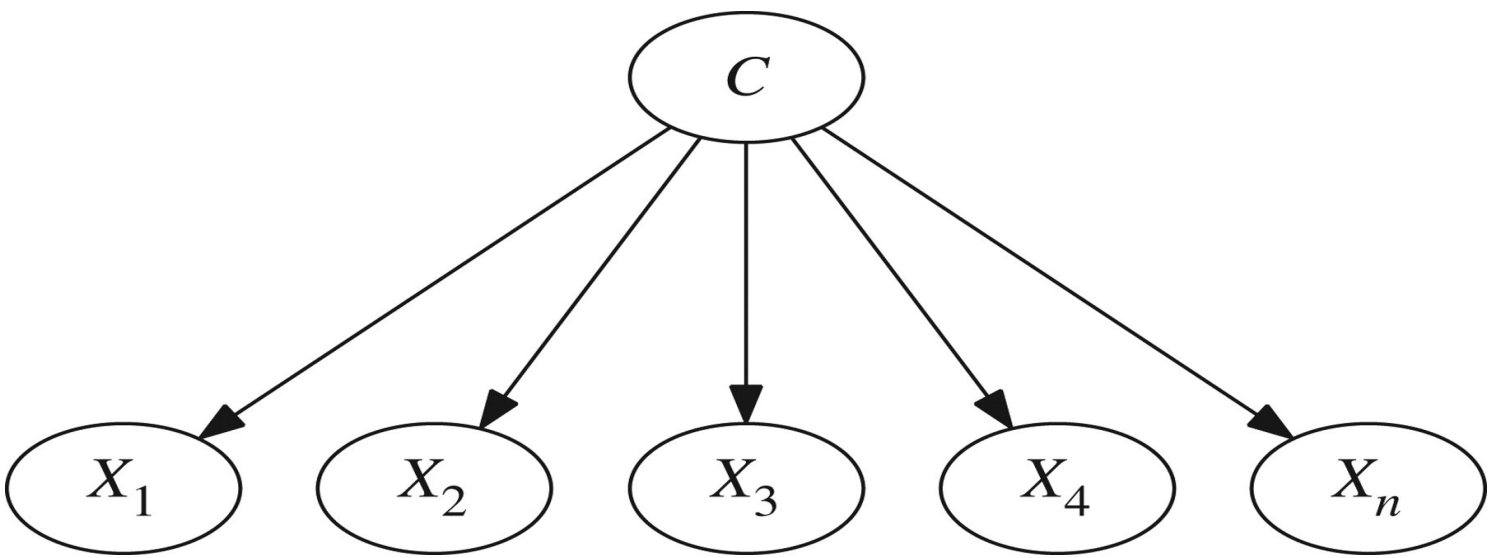
Árvore de Decisão:

Classes: parafuso, caneta, chave, porca, tesoura

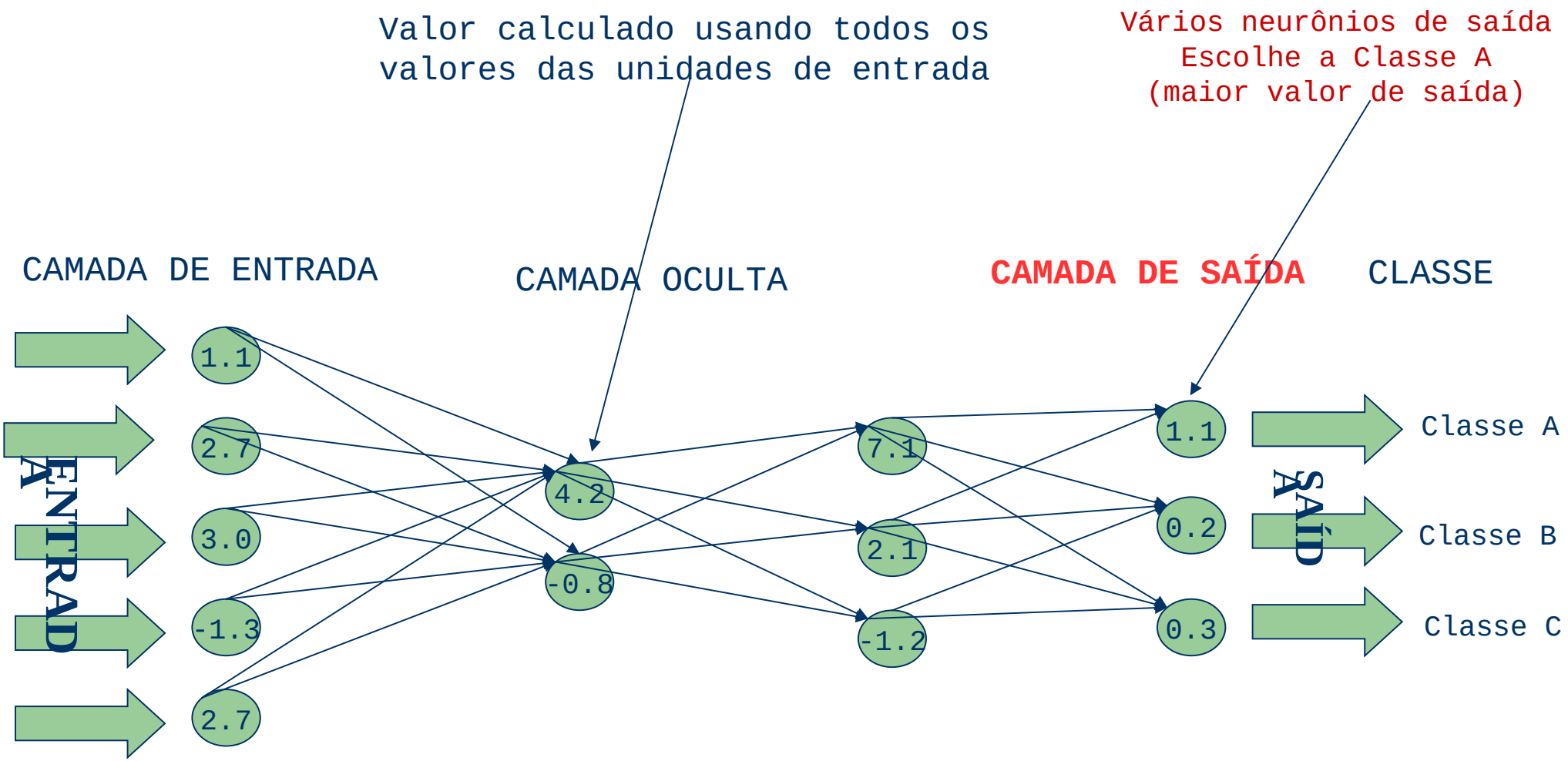


Naive Bayes Classifier

- Nó classe é uma variável que pode assumir múltiplos valores



Redes Neurais MULTICAMADAS



Redes Neurais MULTICAMADAS

- Outras 2 opções:
- Em ambas, os resultados possíveis de cada neurônio de saída é 0 ou 1
  - Codificação um por classe
  - Codificação distribuída

Redes Neurais MULTICAMADAS
Codificação “um por classe”

- M classes, M neurônios de saída (um neurônio por classe)
- Cada neurônio de saída i representa classe i
- Classificação de x para a classe j:
  - Neurônio de saída j = 1
  - Neurônio de saída i = 0 para todo i ≠ j

Redes Neurais MULTICAMADAS
Codificação “um por classe”

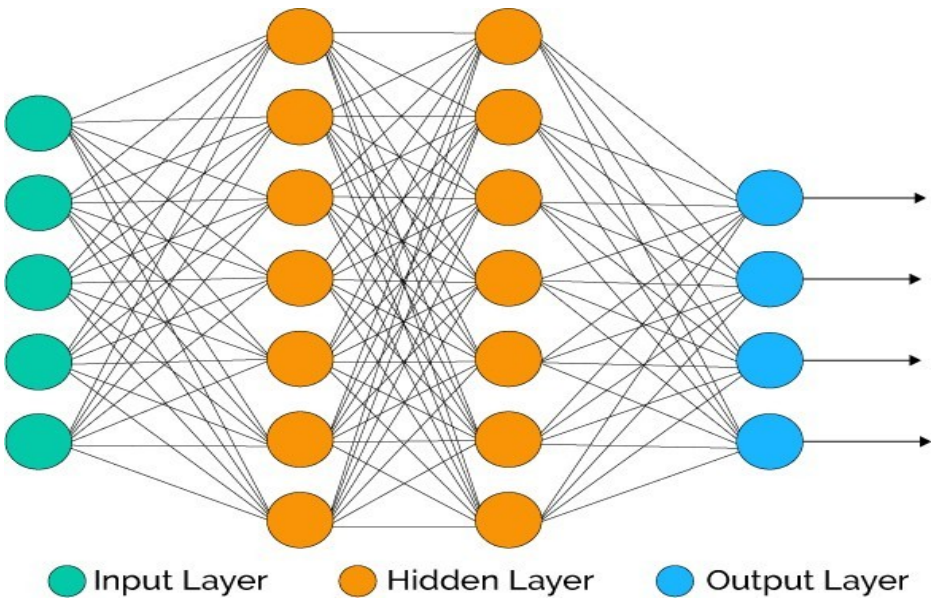
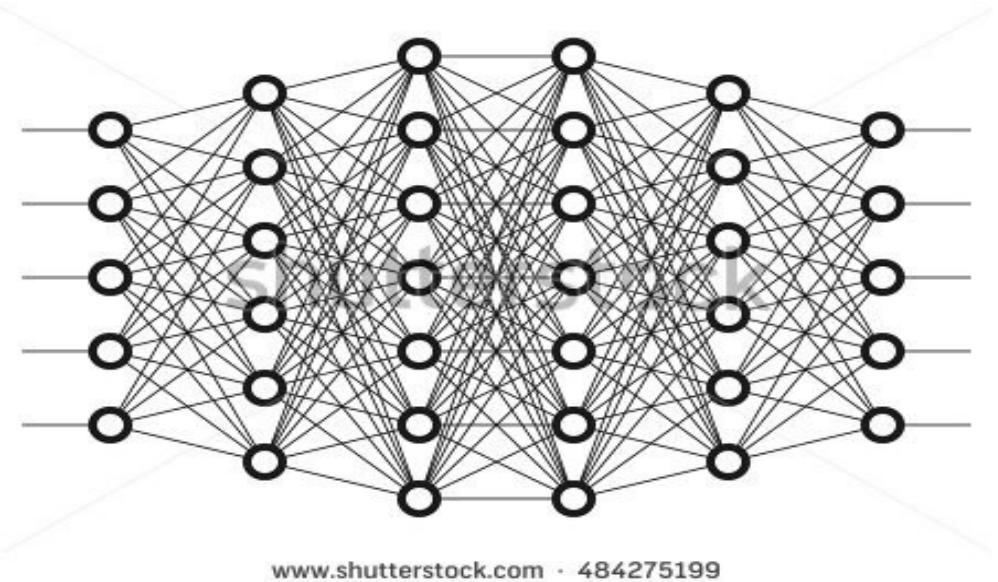


Table 1: One-per-class Coding

Class 1	1000
Class 2	0100
Class 3	0010
Class 4	0001

Redes Neurais MULTICAMADAS
Codificação distribuída

- M classes, S neurônios de saída (S normalmente maior que M)
- Ex: 4 classes e 5 neurônios de saída



Redes Neurais MULTICAMADAS
Codificação distribuída

- M classes, S neurônios de saída (S normalmente maior que M)
- Cada classe i é associada a um string binário Bi de tamanho S (S bits)

Table 2: Distributed coding

Class 1	00000
Class 2	00111
Class 3	11001
Class 4	11110

Redes Neurais MULTICAMADAS
Codificação distribuída

- M classes, S neurônios de saída (S normalmente maior que M)
- Cada classe i é associada a um string binário Bi de tamanho S (S bits)
- Classificação de x:
  - Usa x como entrada na rede e olha os S neurônios de saída (que juntos formam o string $B^x = y_1y_2y_3...y_S$ de seus resultados)
  - Classifica para a classe j cujo Bj é mais “similar” a B^x, ie, j = argmin d(B^x,Bj)
    - d é normalmente a distância de Hamming (nr de posições diferentes – bit a bit)

Redes Neurais MULTICAMADAS
Codificação distribuída

- Neste exemplo, qual a distância entre cada Bj?

Table 2: Distributed coding

Class 1	00000
Class 2	00111
Class 3	11001
Class 4	11110

Redes Neurais MULTICAMADAS
Codificação distribuída

- Neste exemplo, qual a distância entre cada Bj? 3

Table 2: Distributed coding

Class 1	00000
Class 2	00111
Class 3	11001
Class 4	11110

Redes Neurais MULTICAMADAS
Codificação distribuída

- Neste exemplo, qual a distância entre cada Bj? 3
- Como seria classificada uma entrada x cuja saída fosse 11100?

Table 2: Distributed coding

Class 1	00000
Class 2	00111
Class 3	11001
Class 4	11110

Redes Neurais MULTICAMADAS
Codificação distribuída

- Neste exemplo, qual a distância entre cada Bj? 3

- Como seria classificada uma entrada x cuja saída fosse 11100? Classe 4

Table 2: Distributed coding

Class 1	00000
Class 2	00111
Class 3	11001
Class 4	11110

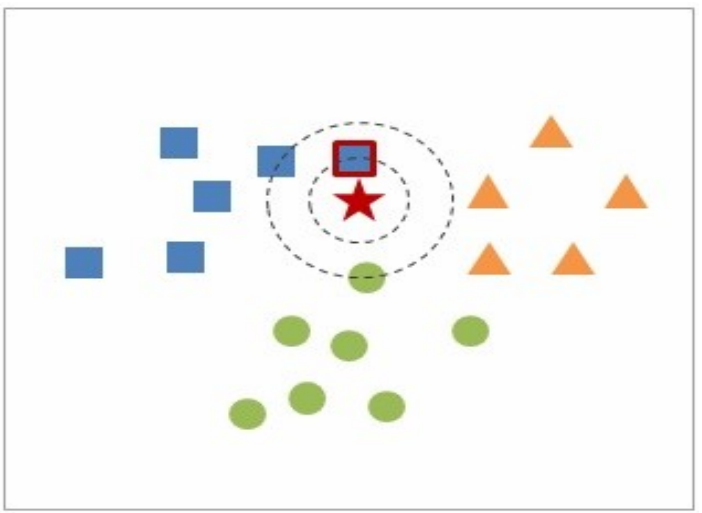
KNN – k nearest neighbors (k vizinhos mais próximos)

Aprendizado eager x lazy

- **Eager**: processo de aprendizado de um modelo ANTES da etapa de classificação
  - Aprendido o classificador, ele está pronto para ser utilizado na fase de aplicação
  - Mesmo que o treinamento demore, isso é feito fora dele estar “em produção”
  - Sua execução pode ser rápida (na classficação)
  - Ex: todos os que vimos até agora
- **Lazy**: a amostra “de treinamento” só é analisada na fase de avaliação
  - Não há treinamento
  - Fase de classificação mais trabalhosa
  - Também chamados de “baseados em instância”
  - Ex: kNN

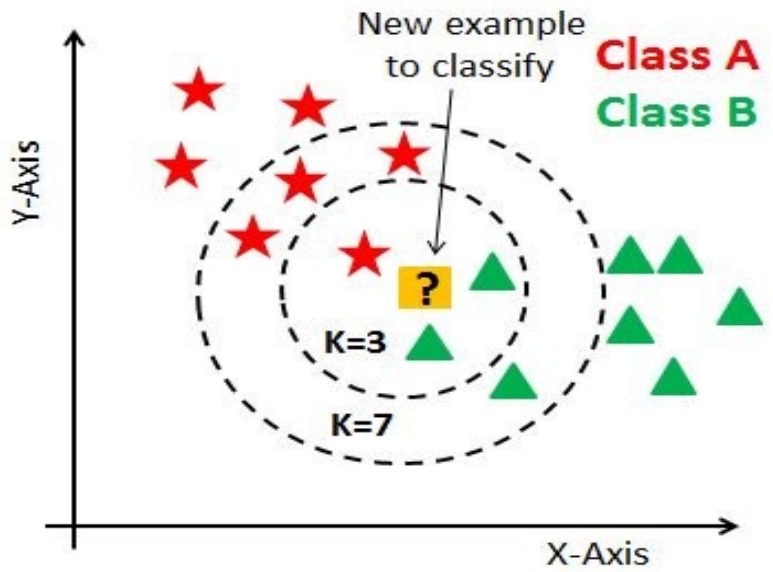
KNN – k nearest neighbors (k vizinhos mais próximos)

- Classificador simples (supervisionado, mas sem treinamento de um modelo)
- Dada uma amostra rotulada de exemplos de M classes
  - Cada novo dado x é classificado para a classe majoritária dos k vizinhos mais próximos de x
  - Opção dos votos serem ponderados pelo inverso da distância do vizinho



KNN – k nearest neighbors (k vizinhos mais próximos)

- Medida de distância deve ser definida (ex: distância euclidiana)
- K é um parâmetro (a ser calibrado)



Classificação Multiclasse

- Classificação multiclasse: dado um elemento, classificá-lo em 1 de M classes

- Muitos métodos de classificação são naturalmente multiclasse
- Outros possuem extensões para tratamento multiclasse
- Outros, mesmo destinados à classificação binária (apenas 2 classes) podem ser utilizados para classificação multiclasse

Extensões para multiclasse

- SVM: formulações para multiclasse
- Parâmetros e restrições adicionais na definição do problema de otimização
- Dependendo da formulação, a solução do problema de otimização a ser resolvido pode ser impraticável para um grande número de classes

Extensões para multiclasse

- SVM: formulações para multiclasse
- Algumas referências:
  - Koby Crammer and Yoram Singer. On the algorithmic implementation of multiclass kernel-based vector machines. Journal of Machine Learning Research , pages 265–292, 2001.
  - J. Weston and C. Watkins. Support vector machines for multiclass pattern recognition. In Proceedings of the Seventh European Symposium On Artificial Neural Networks , 4 1999.
  - Yoonkyung Lee, Yi Lin, and Grace Wahba. Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data. Journal of the American Statistical Association, 99(465):67–81, March 2004.

Classificação Multiclasse

- Classificação multiclasse: dado um elemento, classificá-lo em 1 de M classes

- Muitos métodos de classificação são naturalmente multiclasse
- Outros possuem extensões para tratamento multiclasse
- Outros, mesmo destinados à classificação binária (apenas 2 classes) podem ser utilizados para classificação multiclasse, utilizando técnicas como:
  - One Against One
  - One Against All

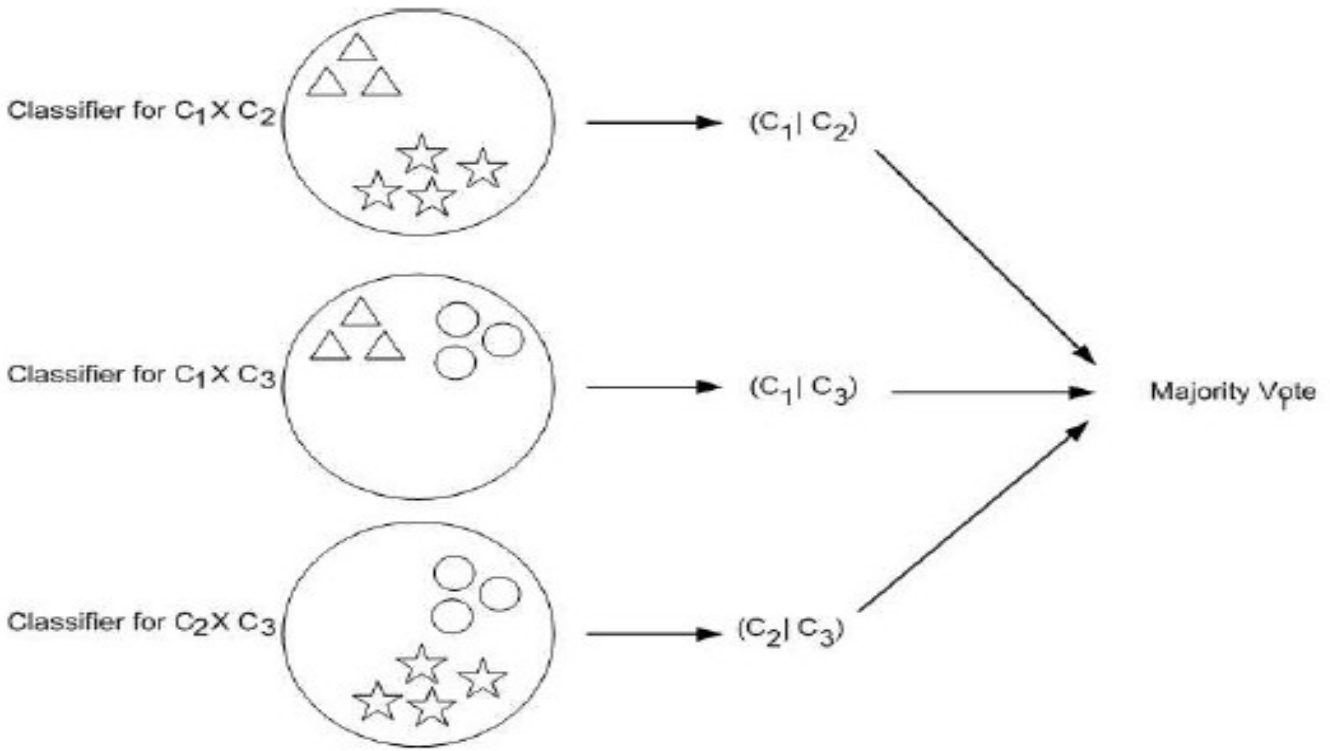
One-Against-One (Um-Contra-Um)

- Um classificador binário para cada par de classes (um contra um)
- Cada classificador “vota” em uma classe
- Ganha a classe mais votada

One-Against-One
(Um-Contra-Um)

Ex: 3 classes:

- Triângulos
- Estrelas
- Círculos



One-Against-One (Um-Contra-Um)

- Um classificador binário para cada par de classes (um contra um)
- Cada classificador “vota” em uma classe
- Ganha a classe mais votada
- Para k classes, temos um comitê de k(k-1)/2 classificadores binários

One-Against-All (Um-Contra-Todos)

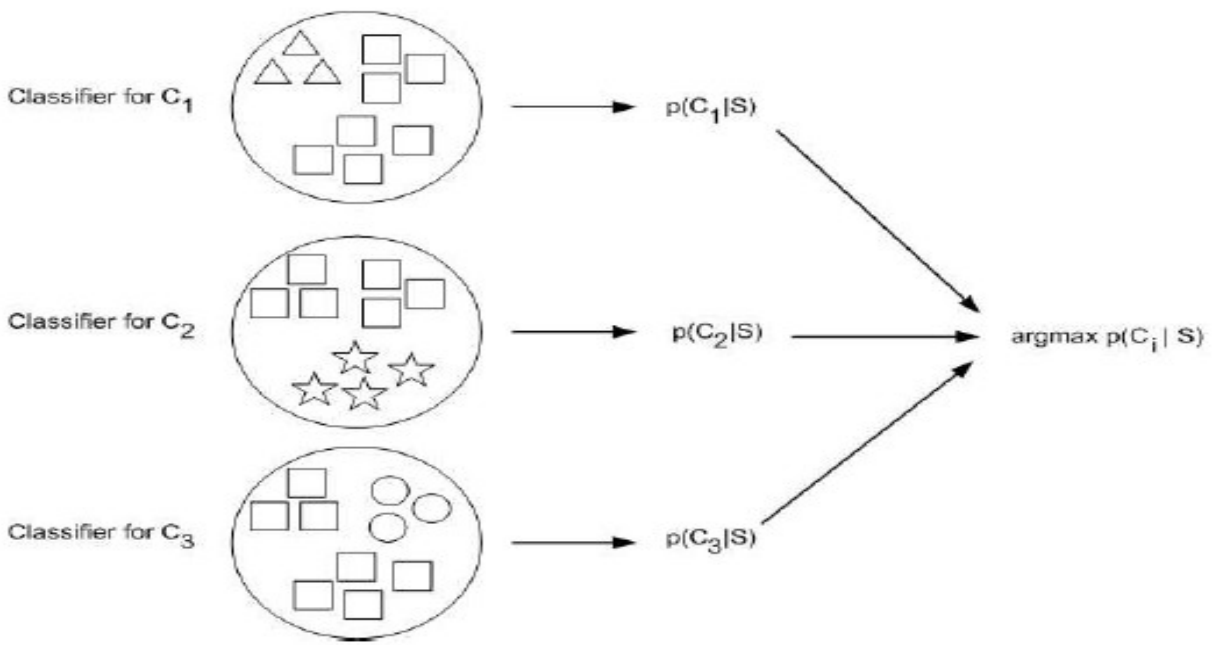
- Um classificador binário para cada classe C_i versus todas as outras juntas (um contra todos os outros)
- Para um dado objeto x, usar cada classificador C_i para obter um “escore” de x (valor, *posteriori* $P(C_i|x)$, etc).
- Ganha a classe C_k cujo classificador C_k atribuir a x o maior “escore”
- Para k classes, temos um comitê de k classificadores binários

One-Against-All
(Um-Contra-Todos)

Ex: 3 classes:

- Triângulos
- Estrelas
- Círculos

Quadrados: o “resto”



Medidas de desempenho de classificadores multiclasse

- Não há mais uma classe positiva e outra negativa
- Como ficam as medidas de desempenho?

Classificadores binários

Acurácia = nr de acertos / nr de instâncias testadas

Erro = 1 - Acurácia

Classificadores binários

Matriz de confusão:

Data class	Classified as *pos*	Classified as *neg*
pos	true positive (*tp*)	false negative (*fn*)
neg	false positive (*fp*)	true negative (*tn*)

$$\begin{bmatrix} tp & fn \\ fp & tn \end{bmatrix}$$

Medidas derivadas:

Measure	Formula	Evaluation focus
Accuracy	$\frac{tp+tn}{tp+fn+fp+tn}$	Overall effectiveness of a classifier
Precision	$\frac{tp}{tp+fp}$	Class agreement of the data labels with the positive labels given by the classifier
Recall (*Sensitivity*)	$\frac{tp}{tp+fn}$	Effectiveness of a classifier to identify positive labels
Fscore	$\frac{(\beta^2+1)tp}{(\beta^2+1)tp+\beta^2 fn+fp}$	Relations between data's positive labels and those given by a classifier
Specificity	$\frac{tn}{fp+tn}$	How effectively a classifier identifies negative labels
AUC	$\frac{1}{2}\left(\frac{tp}{tp+fn}+\frac{tn}{tn+fp}\right)$	Classifier's ability to avoid false classification

Information Processing and Management 45 (2009) 427–437

Classificadores multiclasse

Acurácia absoluta = nr de acertos / nr de instâncias testadas

Erro absoluto= 1 – Acurácia

No entanto, esses valores serão muito baixos

Classificadores multiclasse

Matriz de confusão:

	Anger	Disgust	Fear	Happiness	Neutral	Sadness	Surprise
Anger	**51%**	7%	6%	1%	27%	5%	3%
Disgust	38%	**49%**	2%	2%	5%	2%	2%
Fear	9%	7%	**36%**	2%	12%	13%	21%
Happiness	0%	0%	0%	**86%**	10%	2%	2%
Neutral	1%	0%	2%	2%	**91%**	2%	2%
Sadness	6%	5%	6%	0%	3%	**79%**	1%
Surprise	0%	1%	8%	5%	13%	2%	**71%**

2015 IEEE 28th International Symposium on Computer-Based Medical Systems

Rafael L. Testa*, Antônio H. N. Muniz*, Liseth. U. S. Carpio*, Rodrigo S. Dias †, Cristiana C. A. Rocca †,
Ariane Machado-Lima* and Fatima L. S. Nunes*

Classificadores multiclasse

Medidas derivadas **POR CLASSE**

classe i é a positiva e as demais negativas

Measure	Formula	Evaluation focus
Average Accuracy	$\frac{\sum_{i=1}^{l}\frac{tp_i+tn_i}{tp_i+fn_i+fp_i+tn_i}}{l}$	The average per-class effectiveness of a classifier
Error Rate	$\frac{\sum_{i=1}^{l}\frac{fp_i+fn_i}{tp_i+fn_i+fp_i+tn_i}}{l}$	The average per-class classification error
$Precision_{\mu}$	$\frac{\sum_{i=1}^{l}tp_i}{\sum_{i=1}^{l}(tp_i+fp_i)}$	Agreement of the data class labels with those of a classifiers if calculated from sums of per-text decisions
$Recall_{\mu}$	$\frac{\sum_{i=1}^{l}tp_i}{\sum_{i=1}^{l}(tp_i+fn_i)}$	Effectiveness of a classifier to identify class labels if calculated from sums of per-text decisions
$Fscore_{\mu}$	$\frac{(\beta^2+1)Precision_{\mu}Recall_{\mu}}{\beta^2 Precision_{\mu}+Recall_{\mu}}$	Relations between data's positive labels and those given by a classifier based on sums of per-text decisions
$Precision_M$	$\frac{\sum_{i=1}^{l}\frac{tp_i}{tp_i+fp_i}}{l}$	An average per-class agreement of the data class labels with those of a classifiers
$Recall_M$	$\frac{\sum_{i=1}^{l}\frac{tp_i}{tp_i+fn_i}}{l}$	An average per-class effectiveness of a classifier to identify class labels
$Fscore_M$	$\frac{(\beta^2+1)Precision_M Recall_M}{\beta^2 Precision_M+Recall_M}$	Relations between data's positive labels and those given by a classifier based on a per-class average

Information Processing and Management 45 (2009) 427–437

Classificadores multiclasse

Medidas derivadas **POR CLASSE**

classe i é a positiva e as demais negativas

Measure	Formula	Evaluation focus
Average Accuracy	$\frac{\sum_{i=1}^{l} \frac{tp_i+tn_i}{tp_i+fn_i+fp_i+tn_i}}{l}$	The average per-class effectiveness of a classifier

Average Accuracy

$$\frac{\sum_{i=1}^{l} \frac{tp_i+tn_i}{tp_i+fn_i+fp_i+tn_i}}{l}$$

Acurácia média (média das acurácias de cada classe)

Information Processing and Management 45 (2009) 427–437

Classificadores multiclasse

Medidas derivadas **POR CLASSE**

classe i é a positiva e as demais negativas

Measure	Formula	Evaluation focus
Average Accuracy	$\frac{\sum_{i=1}^{l} \frac{tp_i+tn_i}{tp_i+fn_i+fp_i+tn_i}}{l}$	The average per-class effectiveness of a classifier
Error Rate	$\frac{\sum_{i=1}^{l} \frac{fp_i+fn_i}{tp_i+fn_i+fp_i+tn_i}}{l}$	The average per-class classification error

Error Rate

$$\frac{\sum_{i=1}^{l} \frac{fp_i+fn_i}{tp_i+fn_i+fp_i+tn_i}}{l}$$

Taxa de erro (médio) (média dos erros de cada classe)

Classificadores multiclasse

Medidas derivadas **POR CLASSE**

classe i é a positiva e as demais negativas

Measure	Formula
Average Accuracy	$\frac{\sum_{i=1}^{l}\frac{tp_i}{tp_i+fn_i}}{l}$
Error Rate	$\frac{\sum_{i=1}^{l}\frac{fp_i}{tp_i+fn_i}}{l}$
$Precision_\mu$	$\frac{\sum_{i=1}^{l}tp_i}{\sum_{i=1}^{l}(tp_i+fp_i)}$
$Recall_\mu$	$\frac{\sum_{i=1}^{l}tp_i}{\sum_{i=1}^{l}(tp_i+fn_i)}$
$Fscore_\mu$	$\frac{(\beta^2+1)Precision_\mu Recall_\mu}{\beta^2 Precision_\mu + Recall_\mu}$
$Precision_M$	$\frac{\sum_{i=1}^{l}\frac{tp_i}{tp_i+fp_i}}{l}$
$Recall_M$	$\frac{\sum_{i=1}^{l}\frac{tp_i}{tp_i+fn_i}}{l}$

Precisão/revocação (totais)

Fscore baseado na precisão/revocação (totais)

Classificadores multiclasse

Medidas derivadas **POR CLASSE**

classe i é a positiva e as demais negativas

Measure	Formula
Average Accuracy	$\frac{\sum_{i=1}^{l}\frac{tp_i+}{tp_i+fn_i+}}{l}$
Error Rate	$\frac{\sum_{i=1}^{l}\frac{fp_i+}{tp_i+fn_i+}}{l}$
$Precision_\mu$	$\frac{\sum_{i=1}^{l}tp_i}{\sum_{i=1}^{l}(tp_i+fi}$
$Recall_\mu$	$\frac{\sum_{i=1}^{l}tp_i}{\sum_{i=1}^{l}(tp_i+fi}$
$Fscore_\mu$	$\frac{(\beta^2+1)Precis}{\beta^2 Precision}$
$Precision_M$	$\frac{\sum_{i=1}^{l}\frac{tp_i}{tp_i+fp_i}}{l}$
$Recall_M$	$\frac{\sum_{i=1}^{l}\frac{tp_i}{tp_i+fn_i}}{l}$

$$Precision_M = \frac{\sum_{i=1}^{l}\frac{tp_i}{tp_i+fp_i}}{l}$$

$$Recall_M = \frac{\sum_{i=1}^{l}\frac{tp_i}{tp_i+fn_i}}{l}$$

$$Fscore_M = \frac{(\beta^2+1)Precision_M Recall_M}{\beta^2 Precision_M + Recall_M}$$

Média da precisões/revocações por classe

Fscore baseado na média de precisão/revocação

Fim do vídeo 2

Classificação multiclasse

Vídeo 3

Classificação multirrótulo

Classificação multirrótulo

- Multiclasse
- Classes não mutuamente excludentes
- Cada elemento pode ser atribuído a mais de uma classe
- Exemplos:
  - classificação de músicas com relação ao gênero musical (blues, jazz, samba, ...)
  - Classificação de textos com relação ao assunto (política, religião, esporte, ...)

Principais questões

- Correlação entre rótulos
  - Assume que são independentes
  - Incorporação implícita
  - Incorporação explícita (modelos probabilísticos)
- Alta dimensionalidade
  - Características
  - Rótulos
- Desbalanceamento de labels

Principais abordagens

- Adaptação dos dados / dos problemas
- Adaptação dos métodos de classificação
- Uso de comitês de classificadores

Principais abordagens

- Adaptação dos dados / dos problemas
- Adaptação dos métodos de classificação
- Uso de comitês de classificadores

Adaptação dos dados / dos problemas

- Seleção de um único rótulo (eliminação de rótulos)
- Eliminação de instâncias multirrótulo
- Decomposição de instâncias multirrótulo
- Criação de novos rótulos (*label powerset*)
- Binarização por rótulo (binary relevance)
- Cascata (classifier chain)

Adaptação dos dados / dos problemas

- Dataset original (Do)
  - Número de instâncias No
  - Classe(s): uma variável multivalorada (uma lista de rótulos)
- Problema de classificação inicial: multirrótulo

Dataset original

Instâncias	Classes
1	A
2	B, C
3	B
4	B, C
5	A
6	A, B, C
7	A, B
8	C

Seleção de um único rótulo

(eliminação de rótulos)

- Dataset final (f) x original (o):
  - Número de instâncias: Nf = No
  - Classe(s): uma variável de um único valor
- Classificador: 1 multiclasse

Ex: classificar músicas pelo gênero musical principal

Dataset original

Instâncias	Classes
1	A
2	B, C
3	B
4	B, C
5	A
6	A, B, C
7	A, B
8	C

Dataset final

Instâncias	Classes
1	A
2	B
3	B
4	C
5	A
6	B
7	A
8	C

Eliminação de instâncias multirrótulo

- Dataset final (f) x original (o):
  - Número de instâncias: Nf < No
  - Classe(s): uma variável de um único valor
- Classificador: 1 multiclasse

Ex: classificar vinhos sem misturas

Dataset original

Instâncias	Classes
1	A
2	B, C
3	B
4	B, C
5	A
6	A, B, C
7	A, B
8	C

Dataset final

Instâncias	Classes
1	A
3	B
5	A
8	C

Decomposição de instâncias multirrótulo

- Dataset final (f) x original (o):
  - Número de instâncias: Nf > No
  - Classe(s): uma variável de um único valor
- Classificador: 1 multiclasse

Ex: classificação de sentimentos por expressão facial

Dataset original

Instâncias	Classes
1	A
2	B, C
3	B
4	B, C
5	A
6	A, B, C
7	A, B
8	C

Instâncias com diferentes pesos

Dataset final

Instâncias	Classes
1	A
2a	B
2b	C
3	B
4a	B
4b	C
5	A
6a	A
6b	B
6c	C
7a	A
7b	B
8	C

Criação de novos rótulos

(Label Powerset ou Labelset)

- Dataset final (f) x original (o):
  - Número de instâncias: Nf = No
  - Classe(s): uma variável de um único valor
- Classificador: 1 multiclasse de 2^r classes (r = nr original de rótulos) - na prática multirrótulo

Ex: classificar notícias pelo assunto (ex: política e esporte)

Dataset original

Instâncias	Classes
1	A
2	B, C
3	B
4	B, C
5	A
6	A, B, C
7	A, B
8	C

Dataset final

Instâncias	Classes
1	A = {A}
2	D = {B,C}
3	B = {B}
4	D = {B,C}
5	A = {A}
6	E = {A,B,C}
7	F = {A,B}
8	C = {C}

Criação de novos rótulos

(Label Powerset ou Labelset)

- Dataset final (f) x original (o):
  - Número de instâncias: Nf = No
  - Classe(s): uma variável de um único valor
- Classificador: 1 multiclasse de 2^r classes (r = nr original de rótulos) - na prática multirrótulo

Dataset original

Pruned Labelset:
- novos rótulos apenas para os grupos mais frequentes
- instâncias dos grupos menos frequentes precisarão ir para os demais grupos

Problema: não cobre todas as combinações possíveis

Dataset final

Instâncias	Classes
1	A = {A}
2	D = {B,C}
3	B = {B}
4	D = {B,C}
5	A = {A}
6	E = {A,B,C}
7	F = {A,B}
8	C = {C}

Criação de novos rótulos

COMITÊ de Pruned Labelsets: Vários classificadores PL distintos, cada um treinado com uma porção aleatória da amostra de treinamento (~63%)
Sistema de votação permite identificar combinações mesmo que não presente na amostra de treinamento!

nal de rótulos) -

Pruned Labelset:
- novos rótulos apenas para os grupos mais frequentes
- instâncias dos grupos menos frequentes precisarão ir para os demais grupos

Problema: não cobre todas as combinações possíveis

Dataset final

Instâncias	Classes
1	A = {A}
2	D = {B,C}
3	B = {B}
4	D = {B,C}
5	A = {A}
6	E = {A,B,C}
7	F = {A,B}
8	C = {C}

Binarização por rótulo (Binary Relevance)

- Dataset final (f) x original (o):
  - Número de instâncias: Nf = No
  - Classe(s): r rótulos → r variáveis (uma variável binária por rótulo), só 1 ativa por vez
- Classificador: r classificadores binários **independentes** - na prática um comitê (paralelo) multirrótulo

Ex: classificar filmes pelo gênero (ação, romance, comédia, etc.)

Dataset original

Instâncias	Classes
1	A
2	B, C
3	B
4	B, C
5	A
6	A, B, C
7	A, B
8	C

Dataset final

Instâncias	Cl A	Cl B	Cl C
1	1	0	0
2	0	1	1
3	0	1	0
4	0	1	1
5	1	0	0
6	1	1	1
7	1	1	0
8	0	0	1

Cascata (Classifier Chain)

- Dataset final (f) x original (o):
  - Número de instâncias: Nf = No
  - Classe(s): r rótulos → r variáveis (uma variável binária por rótulo), só 1 ativa por vez
- Classificador: r classificadores binários, cada um contendo as classes anteriores como características adicionais - na prática um comitê (paralelo) multirrótulo

Ex: diagnóstico de doenças

Dataset original

Instâncias	Classes
1	A
2	B, C
3	B
4	B, C
5	A
6	A, B, C
7	A, B
8	C

Dataset final v.1

Instâncias	Cl A
1	1
2	0
3	0
4	0
5	1
6	1
7	1
8	0

Cascata (Classifier Chain)

- Dataset final (f) x original (o):
  - Número de instâncias: Nf = No
  - Classe(s): r rótulos → r variáveis (uma variável binária por rótulo), só 1 ativa por vez
- Classificador: r classificadores binários, cada um contendo as classes anteriores como características adicionais - na prática um comitê (paralelo) multirrótulo

Ex: diagnóstico de doenças

Dataset original

Instâncias	Classes
1	A
2	B, C
3	B
4	B, C
5	A
6	A, B, C
7	A, B
8	C

caract. adicionais

Dataset final v.2

Instâncias	Cl A	Cl B
1	1	0
2	0	1
3	0	1
4	0	1
5	1	0
6	1	1
7	1	1
8	0	0

Cascata (Classifier Chain)

- Dataset final (f) x original (o):
  - Número de instâncias: Nf = No
  - Classe(s): r rótulos → r variáveis (uma variável binária por rótulo), só 1 ativa por vez
- Classificador: r classificadores binários, cada um contendo as classes anteriores como características adicionais - na prática um comitê (paralelo) multirrótulo

Ex: diagnóstico de doenças

Dataset original

Instâncias	Classes
1	A
2	B, C
3	B
4	B, C
5	A
6	A, B, C
7	A, B
8	C

caract. adicionais

Dataset final v.3

Instâncias	Cl A	Cl B	Cl C
1	1	0	0
2	0	1	1
3	0	1	0
4	0	1	1
5	1	0	0
6	1	1	1
7	1	1	0
8	0	0	1

Cascata (Classifier Chain)

- Dataset final (f) x original (o):
  - Número de instâncias: Nf = No
  - Classe(s): r rótulos → r variáveis (uma variável binária por rótulo), só 1 ativa por vez
- Classificador: r classificadores binários, cada um contendo as classes anteriores como características adicionais - na prática um comitê (paralelo) multirrótulo

Ex: diagnóstico de do

Dataset original

Instâncias	Classes
1	A

caract. adicionais

Dataset final v.3

Instâncias	Cl A	Cl B	Cl C
1	1	0	0

Diferentes ordens de tratamento de rótulos → diferentes resultados

Possibilidade: comitê de cascatas utilizando diferentes ordens

Principais abordagens

- Adaptação dos dados / dos problemas
- Adaptação dos métodos de classificação
- Uso de comitês de classificadores

Adaptação dos métodos de classificação

- Baseados em árvores de decisão
  - ML-C4.5 (Multilabel)
  - ADTBoomost.MH (Alternate Decision Tree)
  - ML-Tree
  - LaCova
- Baseados em redes neurais
  - BP-MLL (Multilabel backpropagation)
  - CCA-ELM
- Baseados em SVMs
- Baseado em KNN
- E muitos outros

Principais abordagens

- Adaptação dos dados / dos problemas
- Adaptação dos métodos de classificação
- Uso de comitês de classificadores

Comitê de classificadores

- Comitê de cascatas (ensemble of classifier chains)
- Comitê de Pruned Labelsets
- Comitê de Random k-Labelsets (RakEL): m classificadores multiclasse, labelsets de tamanho k
- HOMER (**H**ierarchy **O**f **M**ultilabel classifi**ER**s) – hierarquia baseada em diferentes subconjuntos de rótulos

Ferramentas

- MULAN
- MEKA
- mldr (pacote do R)
- scikit-learn tem alguma coisa
- scikit-multilearn

Medidas de desempenho para classificação multirrótulo

Notação

- l rótulos (labels)
- **I**(x): função indicadora (1 se x é verdadeiro, 0 c.c.)
- $L^d_i = L^d_i[1], L^d_i[2], \ldots, L^d_i[l]$ são os rótulos (reais) da instância x_i ($L^d_i[j] = 1$ se rótulo j está presente e 0 c.c.)
- $L^c_i = L^c_i[1], L^c_i[2], \ldots, L^c_i[l]$ são os rótulos atribuídos pelo classificador à instância x_i ($L^c_i[j] = 1$ se rótulo j está presente e 0 c.c.)
- Y_i = conjunto de rótulos da instância x_i
- Z_i = conjunto de rótulos atribuídos pelo classificador à instância x_i

Medidas de desempenho para classificação multirrótulo

- Medidas baseadas em instâncias (1/n)
- Medidas baseadas nos rótulos (1/l)

Medidas de desempenho para classificação multirrótulo

- **Medidas baseadas em instâncias (1/n)**
- Medidas baseadas nos rótulos (1/l)

Exact Match Ratio (subset accuracy)

$$\frac{\sum_{i=1}^{n} I(L_i^c = L_i^d)}{n}$$

Taxa de match exato de todos os rótulos
(quanto mais próximo de 1 melhor)

Rótulos dos dados			
ação	romance	aventura	drama
1	1	0	0
0	1	0	1
1	1	1	0
1	0	1	0
0	0	0	1

Rótulos vindos do classificador			
ação	romance	aventura	drama
1	1	0	0
0	0	0	1
1	0	1	0
1	0	0	0
0	1	0	0

EMR = 1/5
= 0,2

Exact Match Ratio (subset accuracy)

$$\frac{\sum_{i=1}^{n} I(L_i^c = L_i^d)}{n}$$

cuidado: no artigo (Sokolova, 2009) está d

Taxa de match exato de todos os rótulos
(quanto mais próximo de 1 melhor)

Rótulos dos dados			
ação	romance	aventura	drama
1	1	0	0
0	1	0	1
1	1	1	0
1	0	1	0
0	0	0	1

Rótulos vindos do classificador			
ação	romance	aventura	drama
1	1	0	0
0	0	0	1
1	0	1	0
1	0	0	0
0	1	0	0

EMR = 1/5
= 0,2

Acurácia

$$\frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|}$$

Valor médio (sobre as instâncias) da proporção de rótulos detectados corretamente dentre todos os rótulos preditos ou reais

Rótulos dos dados			
ação	romance	aventura	drama
1	1	0	0
0	1	0	1
1	1	1	0
1	0	1	0
0	0	0	1

Rótulos vindos do classificador			
ação	romance	aventura	drama
1	1	0	0
0	0	0	1
1	0	1	0
1	0	0	0
0	1	0	0

Acurácia por instância
2/2 = 1
1/2 = 0,5
2/3 = 0,66
1/2 = 0,5
0/2 = 0

Acurácia = (1+0,5+0,66+0,5+0)/5 = 0,532

Hamming Loss

$$\frac{\sum_{i=1}^{n}\sum_{j=1}^{l} I(L_i^c[j] \neq L_i^d[j])}{nl}$$

Erro médio sobre todos os rótulos e todas as instâncias
(quanto mais próximo de 0 melhor)

Rótulos dos dados			
ação	romance	aventura	drama
1	1	0	0
0	1	0	1
1	1	1	0
1	0	1	0
0	0	0	1

Rótulos vindos do classificador			
ação	romance	aventura	drama
1	1	0	0
0	0	0	1
1	0	1	0
1	0	0	0
0	1	0	0

HL = 5/20 = 0,25

Hamming Loss

$$\frac{\sum_{i=1}^{n} \sum_{j=1}^{l} I(L_i^c[j] \neq L_i^d[j])}{nl}$$

Erro médio sobre todos os rótulos e todas as instâncias
(quanto mais próximo de 0 melhor)

Rótulos dos dados			
ação	romance	aventura	drama
1	1	0	0
0	1	0	1
1	1	1	0
1	0	1	0
0	0	0	1

Rótulos vindos do classificador				Distância de Hamming
ação	romance	aventura	drama	
1	1	0	0	0
0	0	0	1	1
1	0	1	0	1
1	0	0	0	1
0	1	0	0	2

HL = 5/20 = 0,25

1-HL = 0,75

Precisão, revocação, medida F

$$Precision = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Z_i|}$$

$$Recall = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Y_i|}$$

$$F\text{-}measure = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Rótulos dos dados			
ação	romance	aventura	drama
1	1	0	0
0	1	0	1
1	1	1	0
1	0	1	0
0	0	0	1

Rótulos vindos do classificador			
ação	romance	aventura	drama
1	1	0	0
0	0	0	1
1	0	1	0
1	0	0	0
0	1	0	0

|Yi|	|Zi|	|Yi ∩ Zi|	|Yi U Zi|	**Precisão**	**Revocação**
2	2	2	2	1	1
2	1	1	2	1	0,5
3	2	2	3	1	0,67
2	1	1	2	1	0,5
1	1	0	2	0	0
			Média:	**0,8**	**0,53**

Medidas baseadas nos rótulos

Macro-averaging

$$Precision_M = \frac{\sum_{i=1}^{l} \frac{tp_i}{tp_i + fp_i}}{l}$$

$$Recall_M = \frac{\sum_{i=1}^{l} \frac{tp_i}{tp_i + fn_i}}{l}$$

Micro-averaging

$$Precision_\mu = \frac{\sum_{i=1}^{l} tp_i}{\sum_{i=1}^{l} (tp_i + fp_i)}$$

$$Recall_\mu = \frac{\sum_{i=1}^{l} tp_i}{\sum_{i=1}^{l} (tp_i + fn_i)}$$

tp_i = nr de instâncias que receberam o rótulo i e que realmente tinham o rótulo i

Classificação multidimensional

- Também chamado multioutput
- Várias variáveis respostas, cada uma delas com mais de um valor possível
- Ex: categorização de filmes por
  - Gênero (drama, ação, comédia, ...)
  - Indicação etária (livre, 12 anos, 14 anos, ...)
  - Lucro
  - ...

Fim do vídeo 3

Classificação multirrótulo

Referências - Comitês

- BISHOP, C. M. **Pattern Recognition and Machine Learning**. Cap 14. Springer, 2006.
- DONG, X. et al. A survey on ensemble learning. **Frontiers in Computer Science**, v. 14 n. 2, p. 241-258, 2020
- JAIN, A. K.; DUIN, R. P. W.; MAO, J. Statistical Pattern Recognition: A Review. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 22 n. 1 p. 4-37, 2000 (seção 6).

Referências - Multiclasse

- ALY, M. Survey on Multiclass Classification Methods. Technical Report. Caltech, 2005.
- MEHRA, N.; GUPTA, S. A Survey on Multiclass Classification Methods. **IJCSIT** 4(4):572-576, 2013
- SILLA Jr, C. N.; KOERICH, A. L.; KAESTNER, C.A.A. A Machine Learning Approach to Automatic Music Genre Classification. **Journal of the Brazilian Computer Society**, v. 14, n.3 p. 7-18, 2008.

Referências - Multilabel

- CLARE, A., KING, R.D.: Knowledge discovery in multi-label phenotype data. In: Proceedings of the 5th European Conference Principles on Data Mining andKnowledge Discovery, PKDD'01, vol. 2168, pp. 42–53. Springer (2001)
- HERRERA, F. et al. **Multilabel Classification**: Problem Analysis, Metrics and Techniques. Springer, 2016.
- SOKOLOVA, M.; LAPALME, G. A systematic analysis of performance measures for classification tasks. **Information Processing and Management** 45:427-437, 2009.