

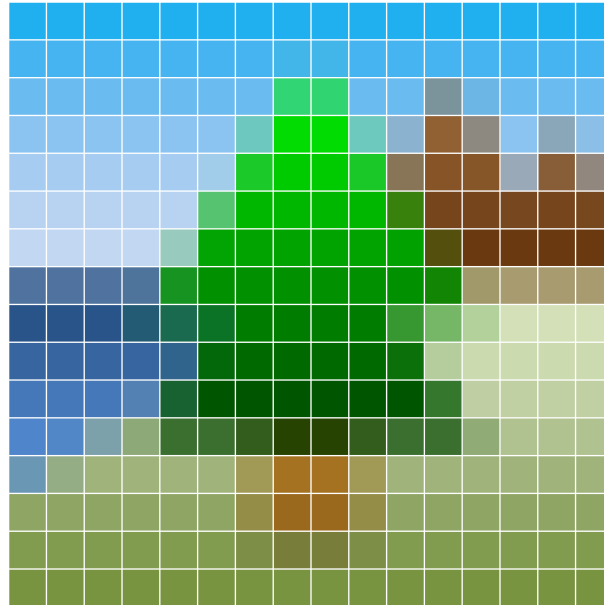
Aula 01 – Classificando imagens

Prof. João Fernando Mari

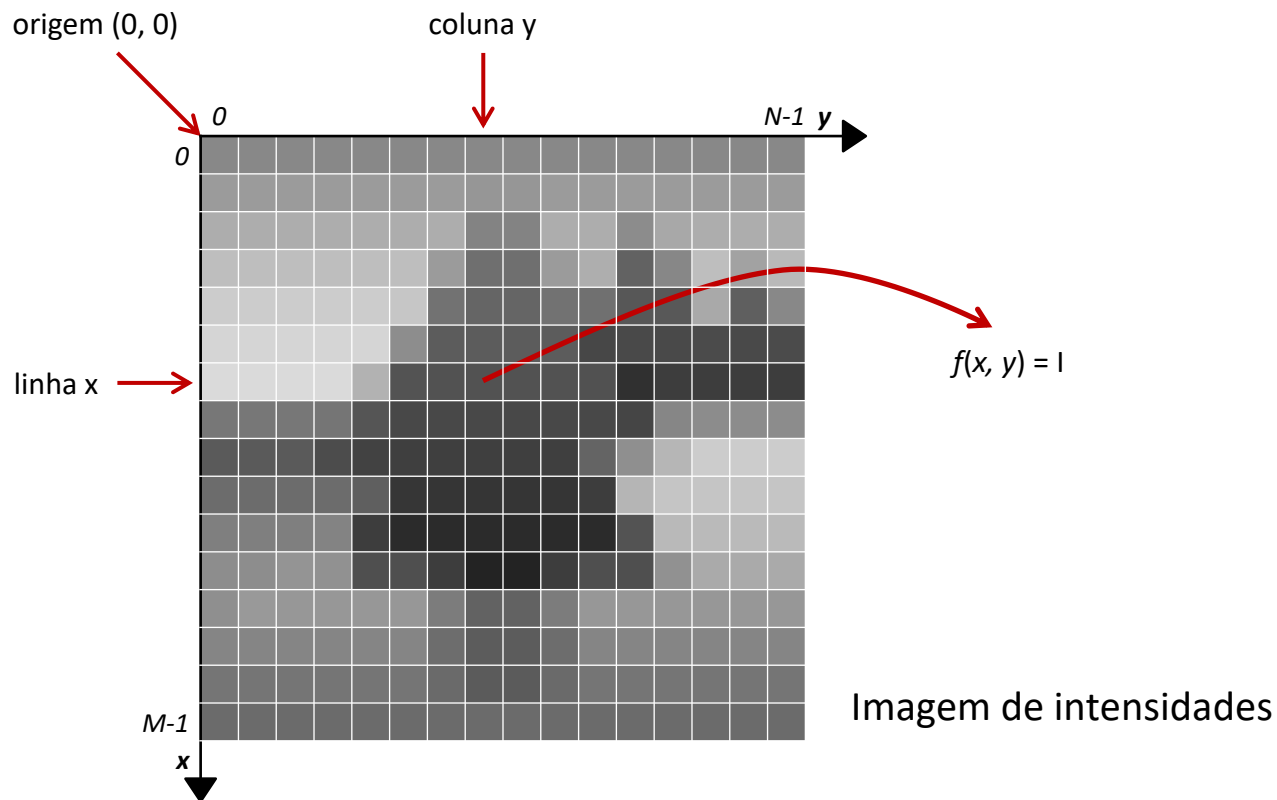
joaofmari.github.io

joaof.mari@ufv.br

- Imagens digitais
- Um problema de classificação
- Pipelines de classificação
- Modelos de aprendizado
- Validação cruzada
- Avaliação dos resultados

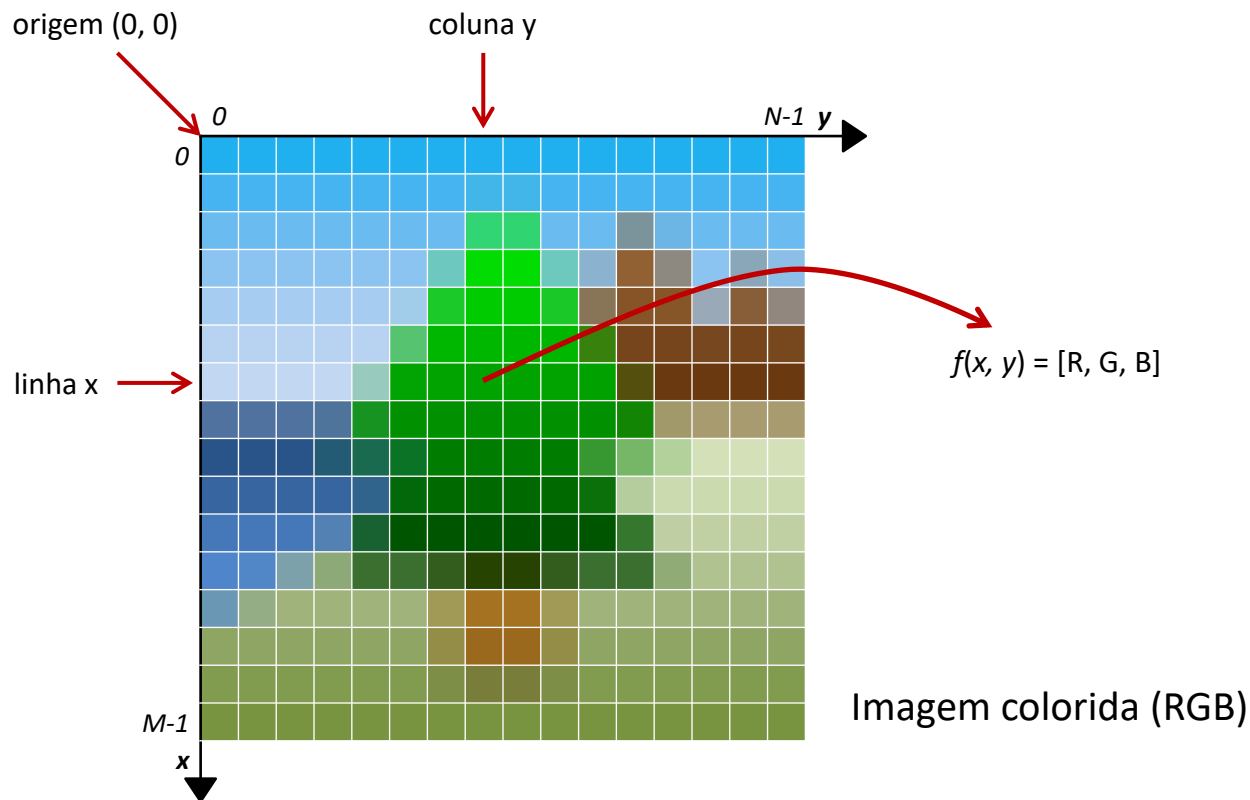


Imagens digitais



M linhas
N colunas
 $M \times N$ pixels

Imagens digitais



M linhas
N colunas
 $M \times N$ pixels

Imagem de intensidades (níveis de cinza):

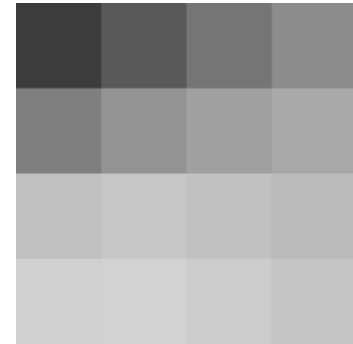
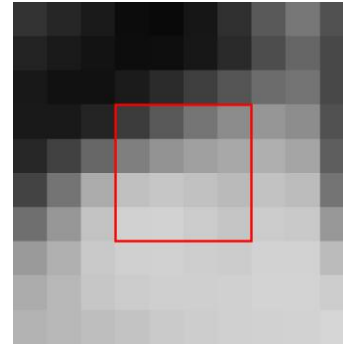
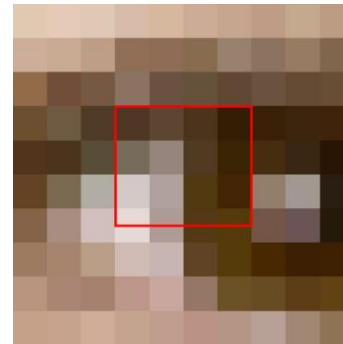
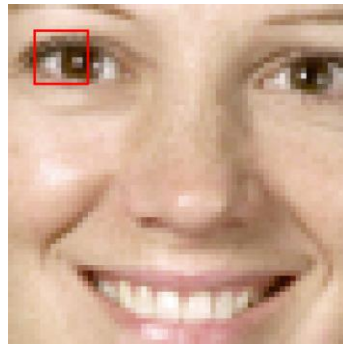
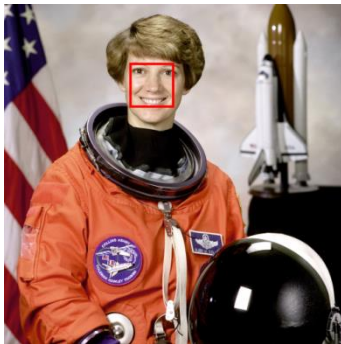
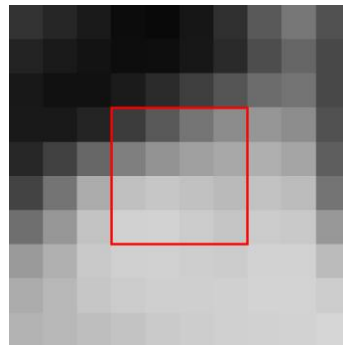


Imagem colorida (RGB):



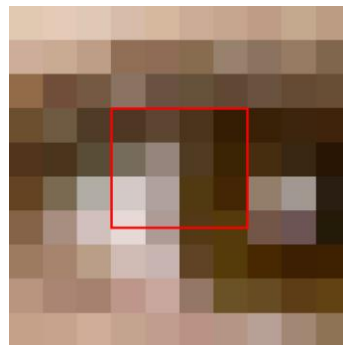
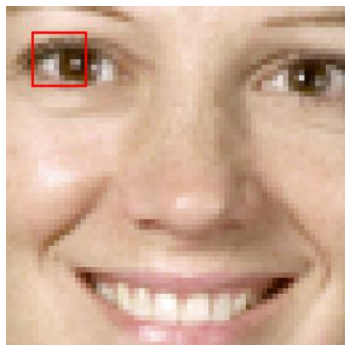
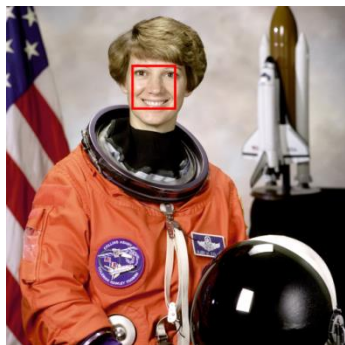
Imagens digitais

Imagem de intensidades (níveis de cinza):



60	89	117	140
127	147	160	168
192	198	193	186
209	210	204	197

Imagem colorida (RGB):



78	92	75	51
56	70	52	30
36	49	28	3
118	149	80	59
108	133	58	36
91	124	33	3
211	176	81	69
202	161	57	38
200	158	17	4
231	174	83	85
218	155	57	58
214	150	21	11



vermelho – R (red)



verde – G (green)



azul – B (blue)





vermelho – R (red)

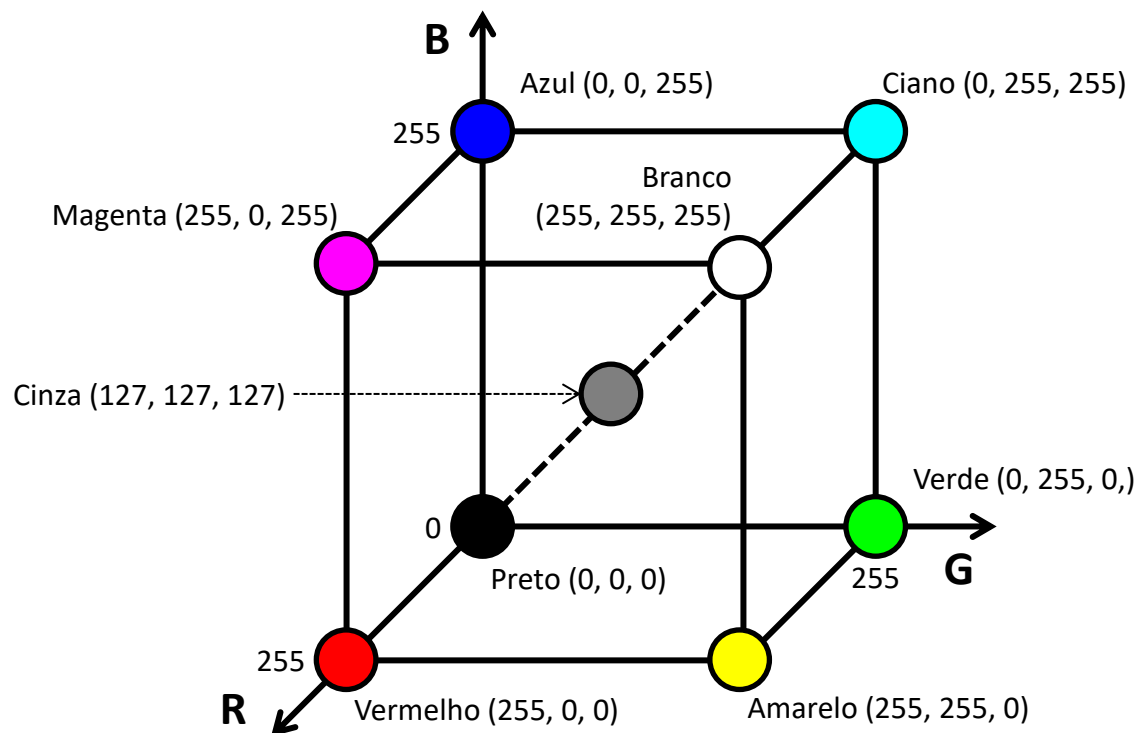


verde – G (green)



azul – B (blue)





UM PROBLEMA DE CLASSIFICAÇÃO

Um problema de classificação

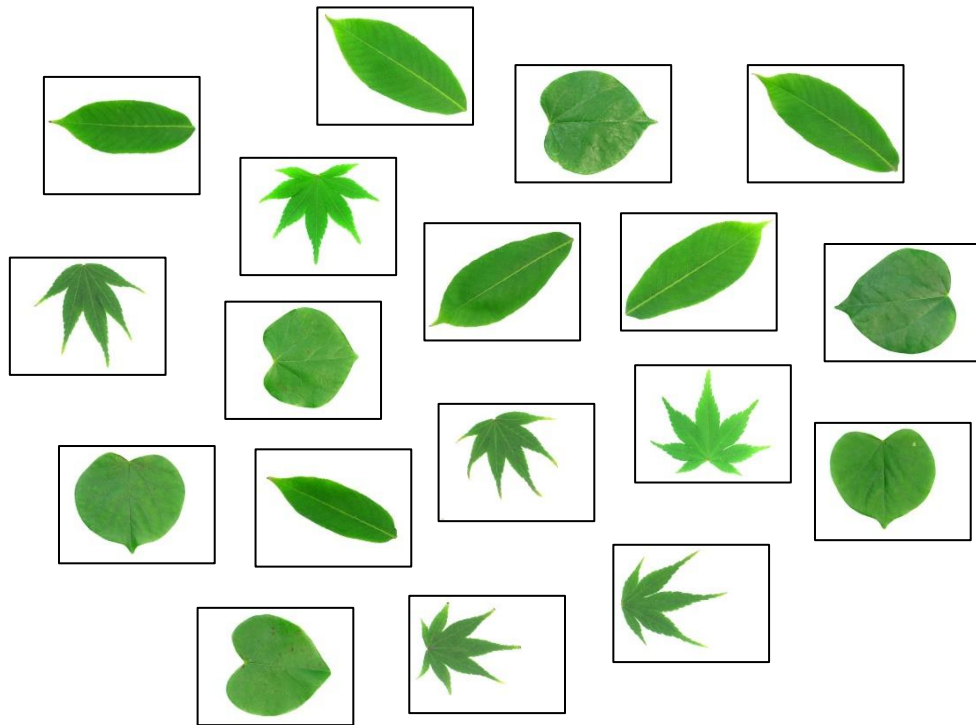
- Aprender a classificar três tipos (classes) de folhas a partir de imagens.

- Flavia leaf dataset:

- <http://flavia.sourceforge.net/>
- 1.907 imagens
- 33 classes

- Seleccionamos 3 classes:

- *aesculus chinensis*
- *acer palmatum*
- *cercis chinensis*



Um problema de classificação

- Aprender a classificar três tipos (classes) de folhas a partir de imagens.

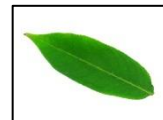
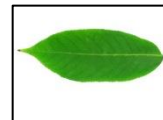
- Extração de características:

- Selecionar características das imagens que podem ser usadas para distinguir entre as classes.

- Características podem ser:

- Formas
- Cores
- Texturas
- Histograma de gradientes (HoG)
- *Bag of Visual Words*
- *Fisher Vectors*
- ...

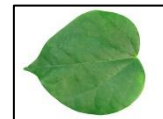
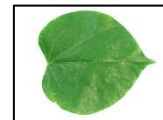
aesculus chinensis



acer palmatum



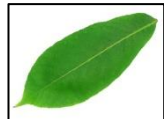
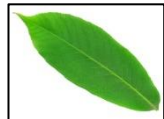
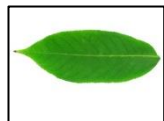
cercis chinensis



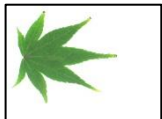
Um problema de classificação

- Algumas características de forma: Área:

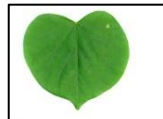
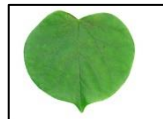
aesculus chinensis



acer palmatum



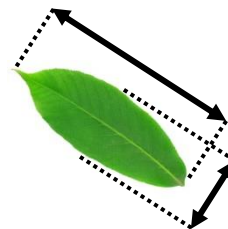
cercis chinensis



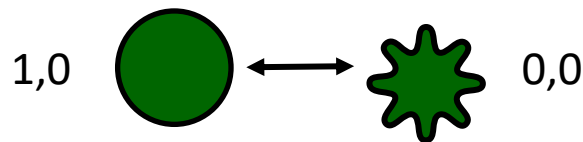
Área:



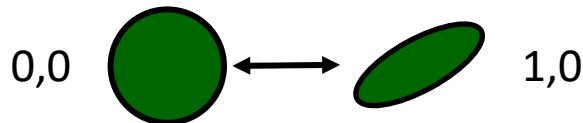
Eixos:



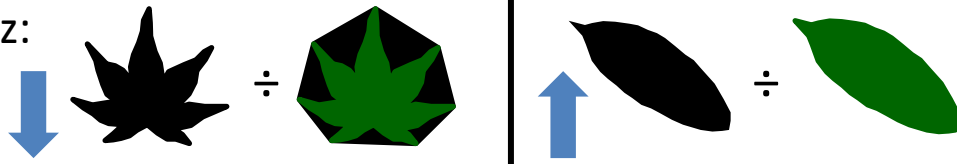
Circularidade:



Excentricidade:



Solidez:



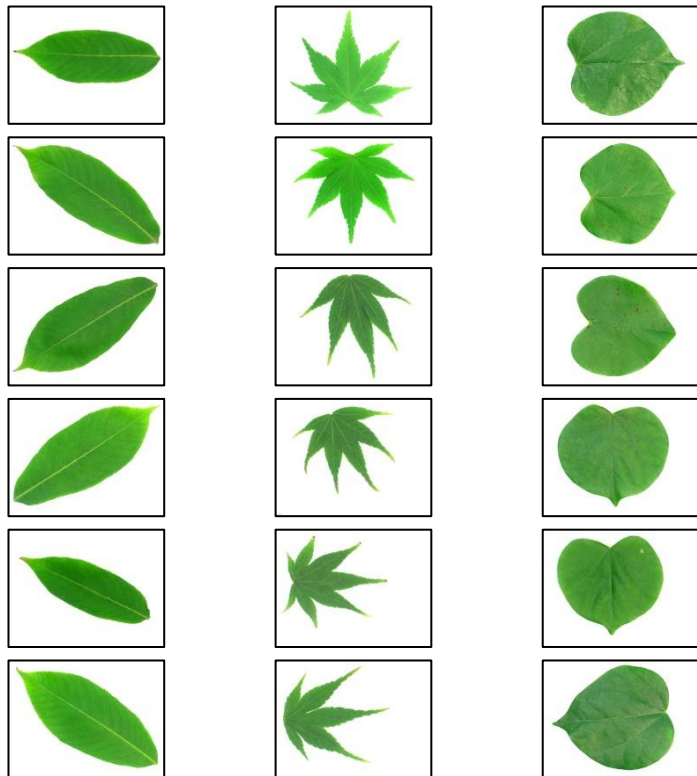
Um problema de classificação

• Algumas características de forma:

aesculus chinensis

acer palmatum

cercis chinensis



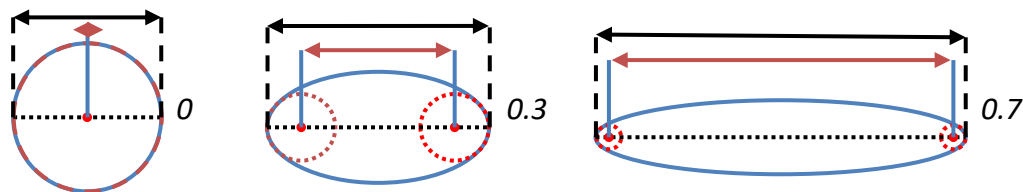
- **Area:** the number of pixels of the shape scaled by pixel-area.

- **Axis:** the length of the major and minor axis of the ellipse with the same normalized second central moments as the region.

- **Roundness:** a function of the perimeter and the area of the region

$$- \text{roundness} = \frac{4 \times \pi \times \text{area}}{\text{perimeter}^2}$$

- **Eccentricity:** the ratio of the focal distance over the major axis length of the ellipse with the same second-moments.



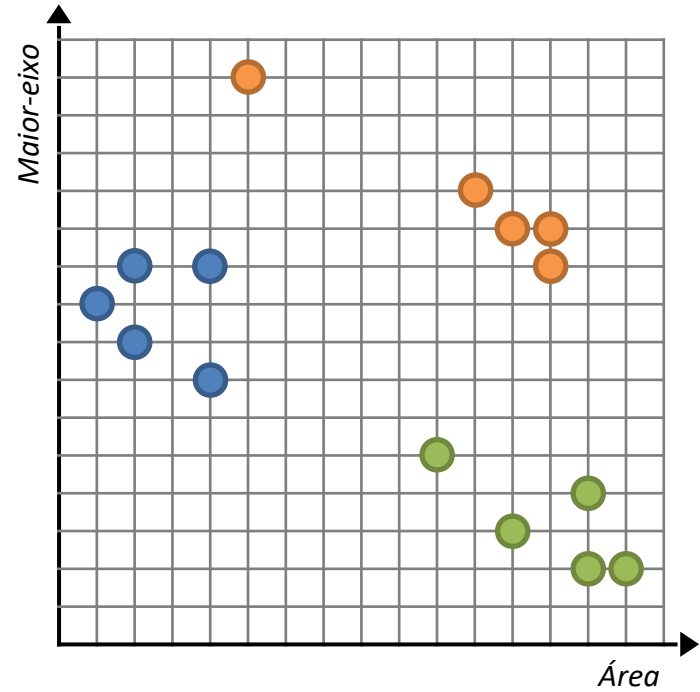
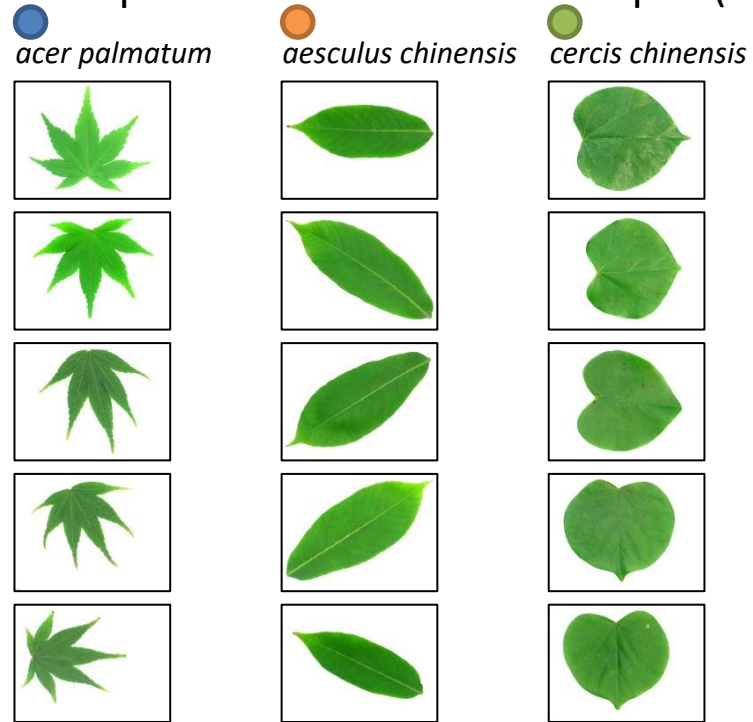
- **Solidity:** the ratio of pixels in the region to pixels of the convex hull image.

- **Convex hull:** the smallest convex polygon that encloses the region.

<https://scikit-image.org/docs/stable/api/skimage.measure.html#skimage.measure.regionprops>

Um problema de classificação

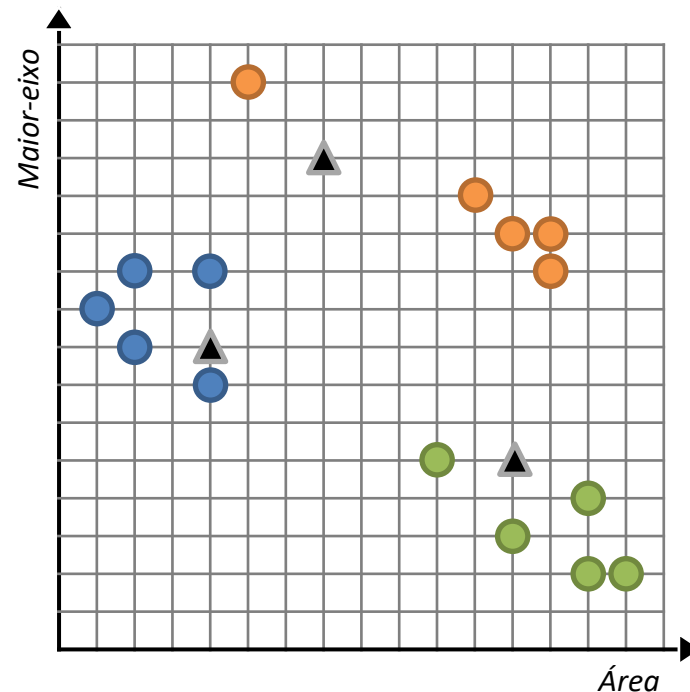
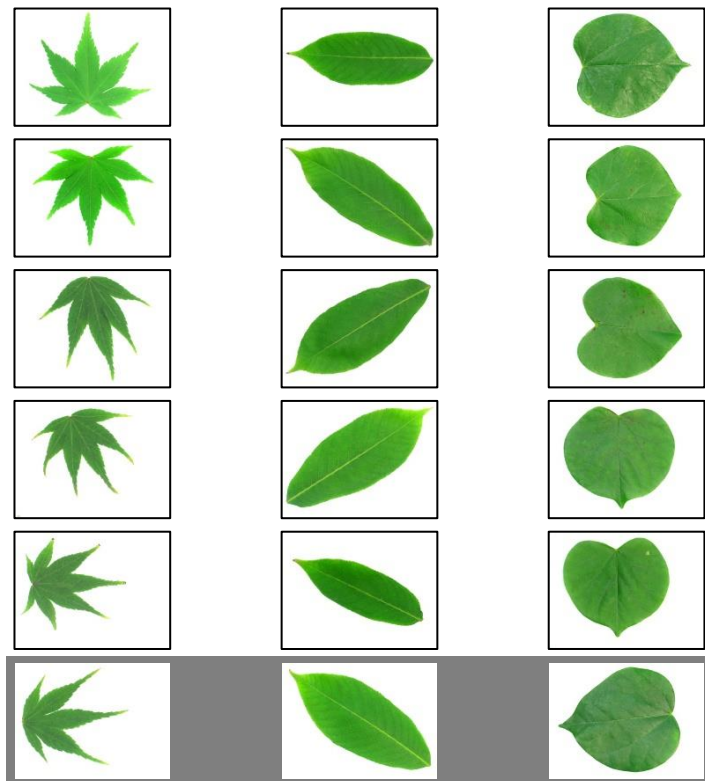
- Aprender a classificar três tipos (classes) de folhas a partir de imagens.



Um problema de classificação




- Aprender a classificar três tipos (classes) de folhas a partir de imagens.

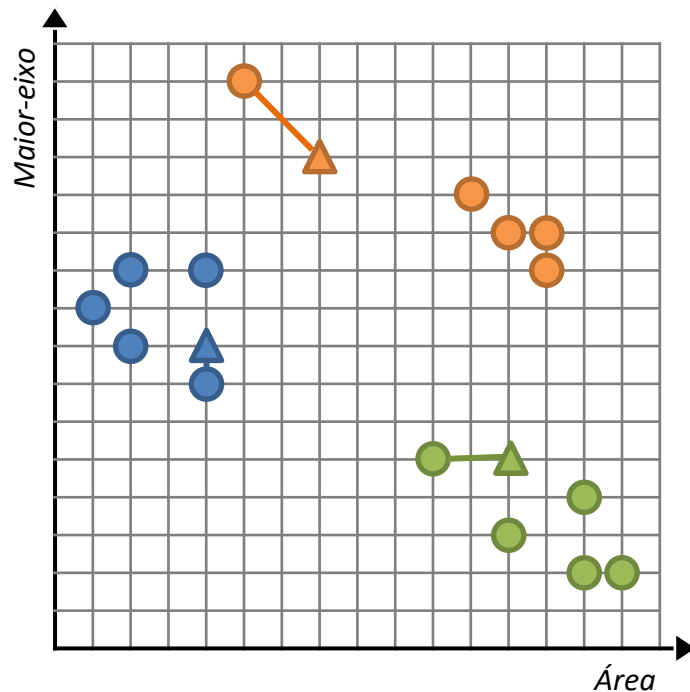
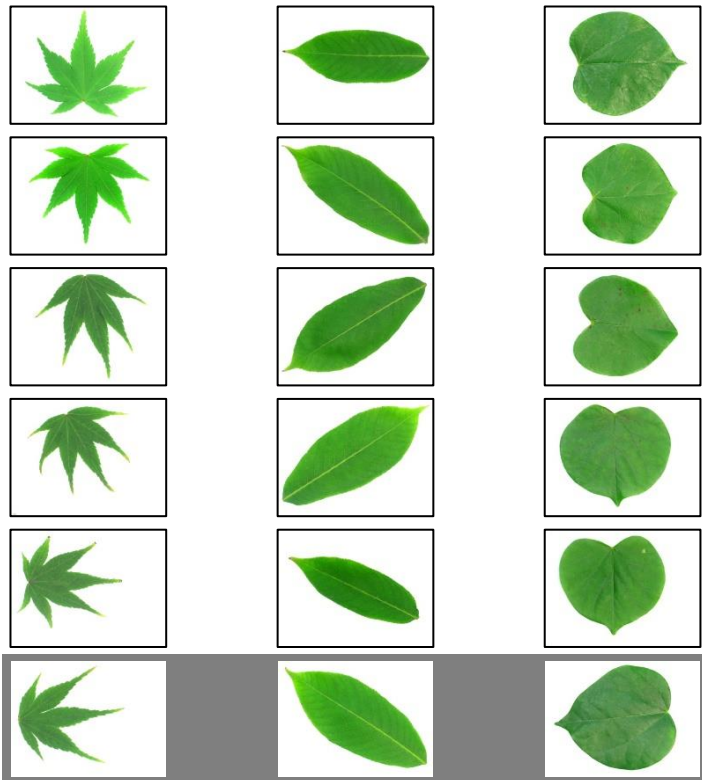
● *acer palmatum*
● *aesculus chinensis*
● *cercis chinensis*



Método do vizinho mais próximo




- Aprender a classificar três tipos (classes) de folhas a partir de imagens.

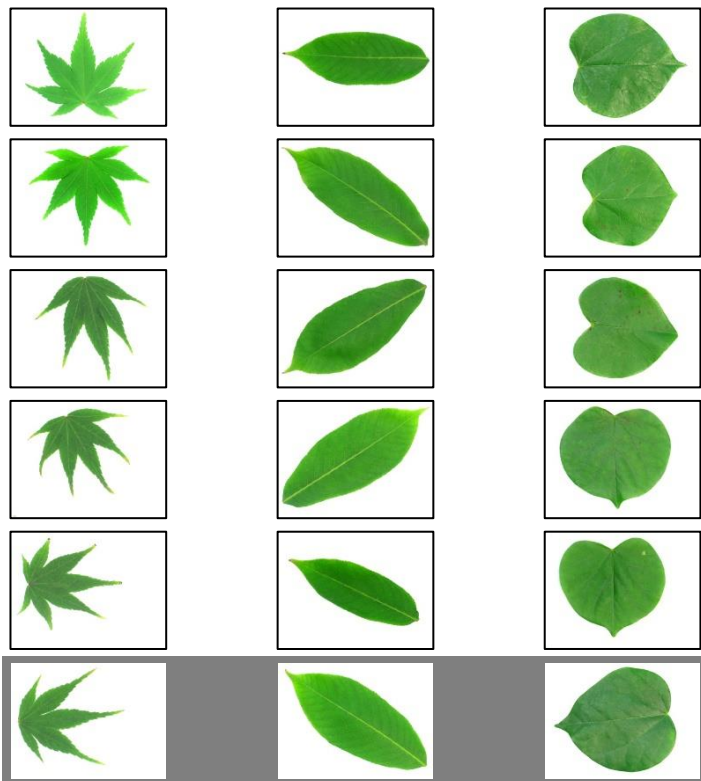
 *acer palmatum*
 *aesculus chinensis*
 *cercis chinensis*



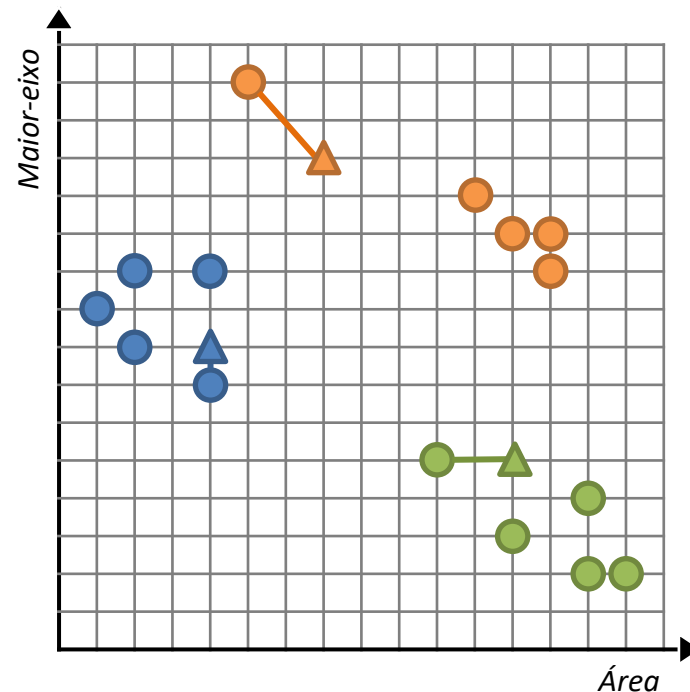
K-vizinhos mais próximos – K-NN

- Aprender a classificar três tipos (classes) de folhas a partir de imagens.

 *acer palmatum*
 *aesculus chinensis*
 *cercis chinensis*






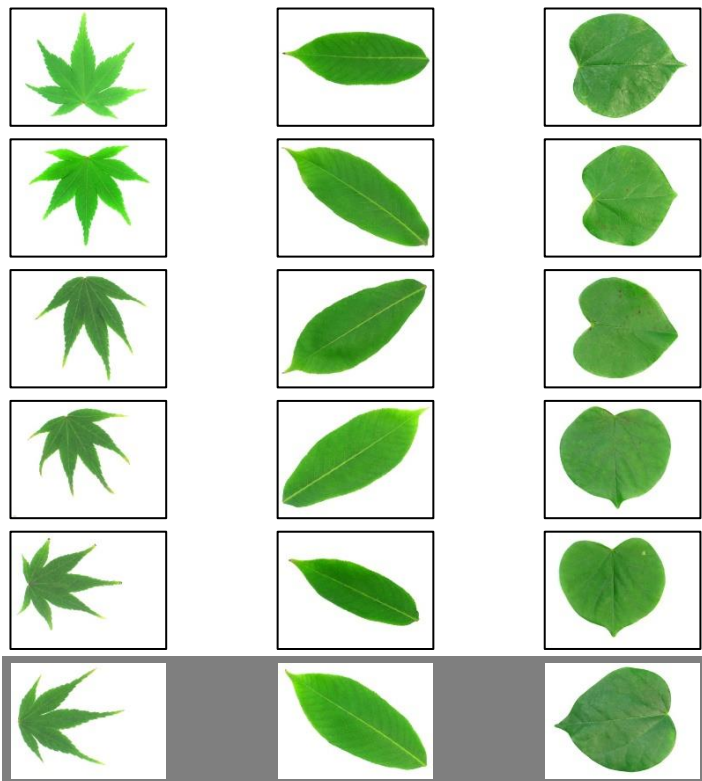
$k = 1$



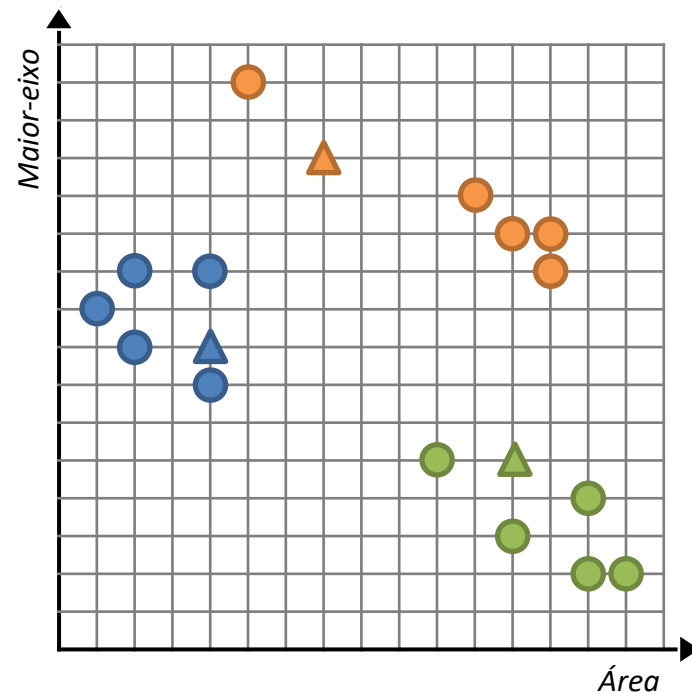
K-vizinhos mais próximos – K-NN

- Aprender a classificar três tipos (classes) de folhas a partir de imagens.

 *acer palmatum*
 *aesculus chinensis*
 *cercis chinensis*






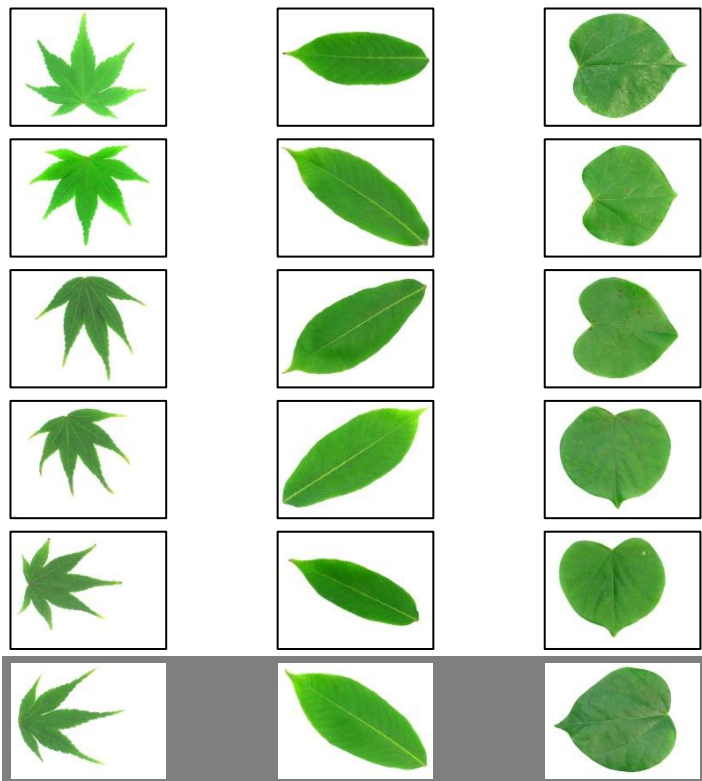
$k = 3$



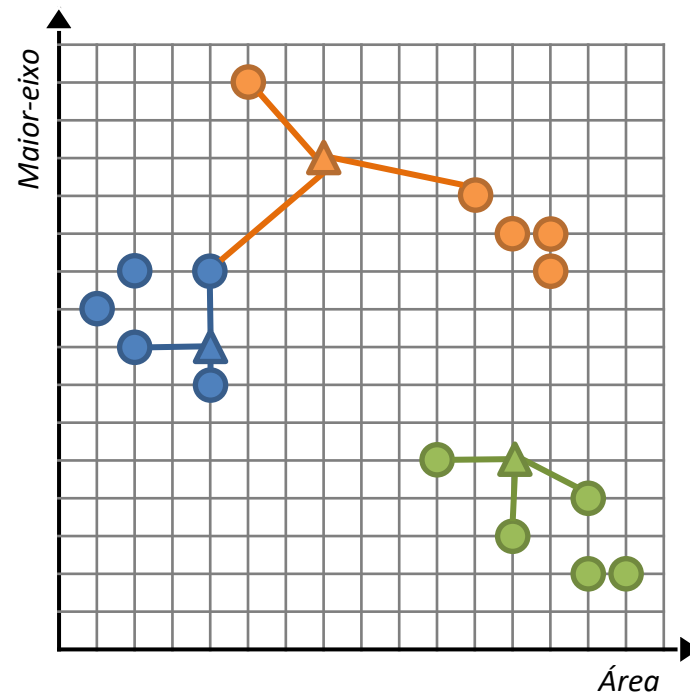
K-vizinhos mais próximos – K-NN

- Aprender a classificar três tipos (classes) de folhas a partir de imagens.

 *acer palmatum*
 *aesculus chinensis*
 *cercis chinensis*



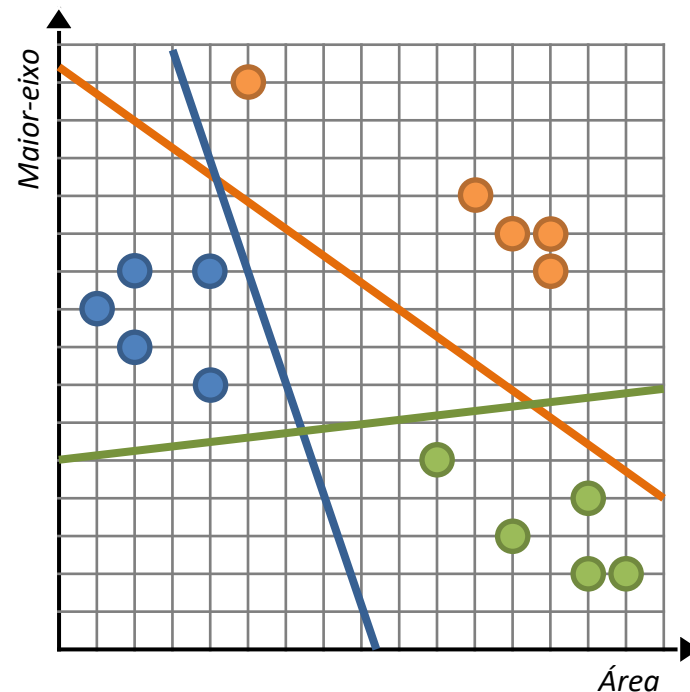
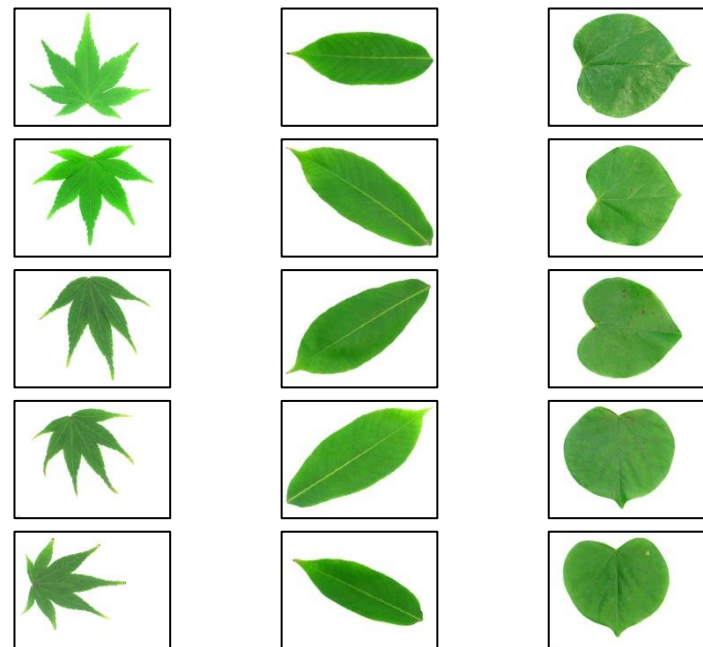
$k = 3$



Funções lineares (Perceptrons)

- Aprender a classificar três tipos (classes) de folhas a partir de imagens.

● *acer palmatum*
● *aesculus chinensis*
● *cercis chinensis*



Funções lineares (Perceptrons)

- Aprender a classificar três tipos (classes) de folhas a partir de imagens.



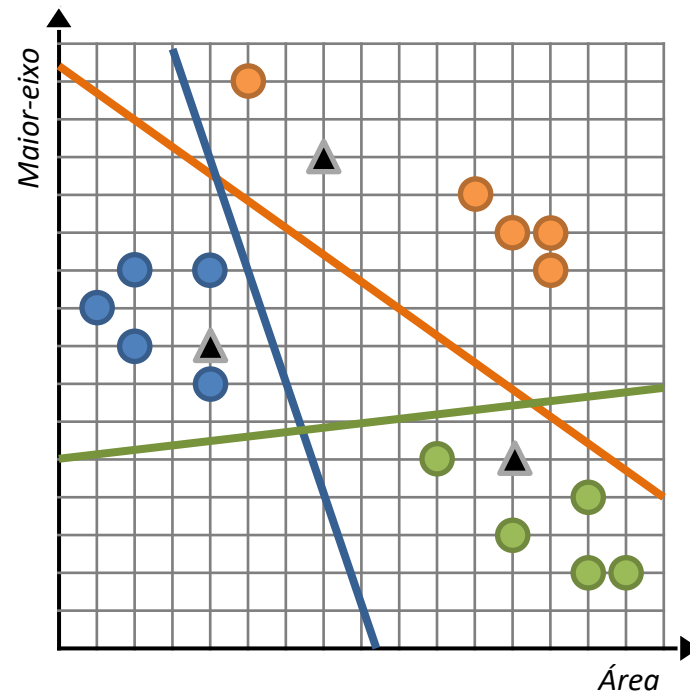
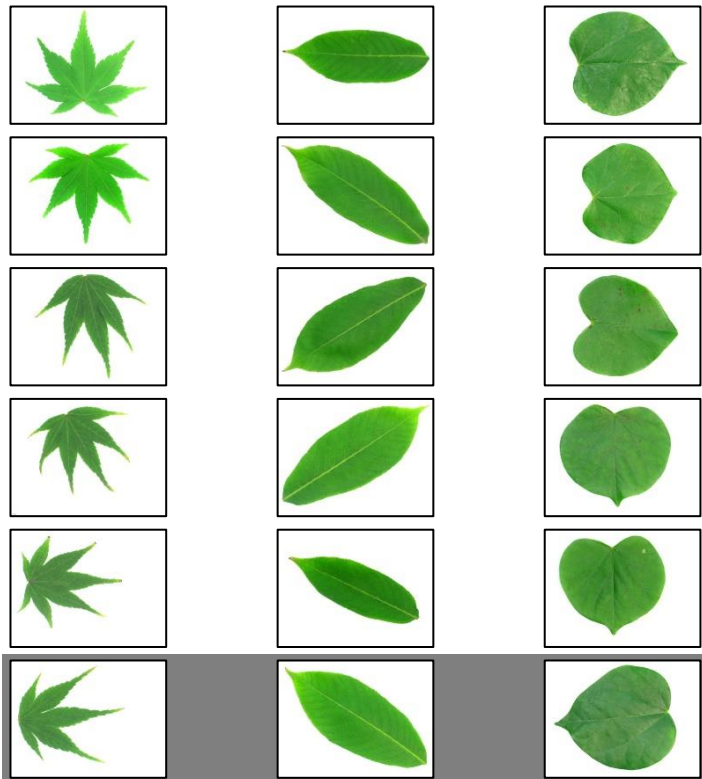
acer palmatum



aesculus chinensis



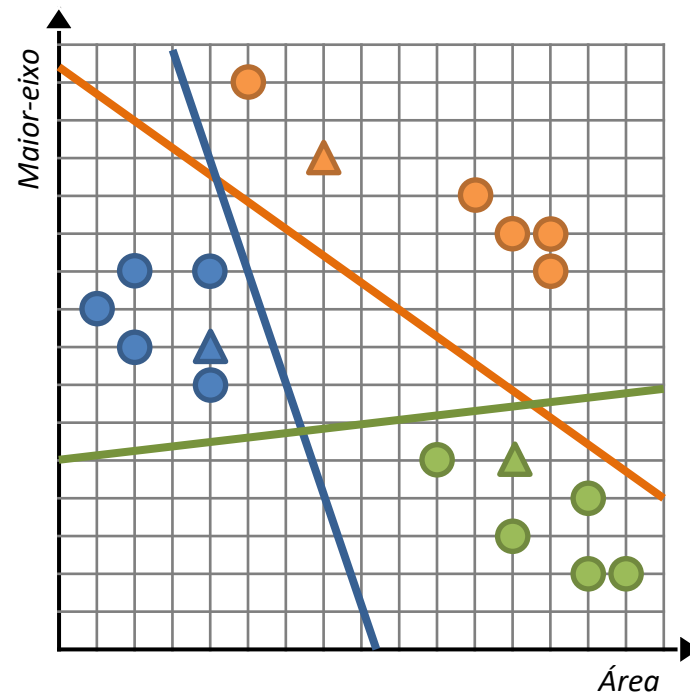
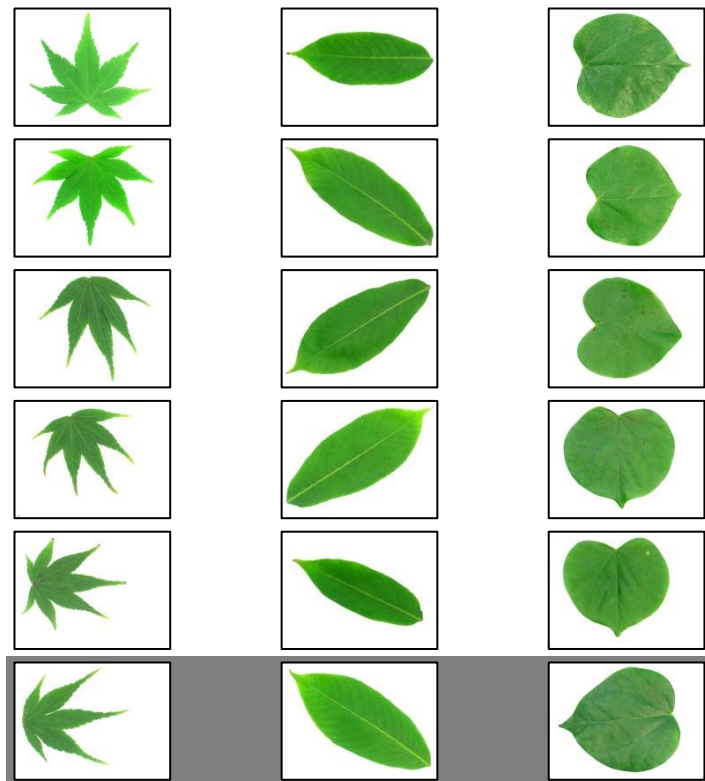
cercis chinensis



Funções lineares (Perceptrons)

- Aprender a classificar três tipos (classes) de folhas a partir de imagens.

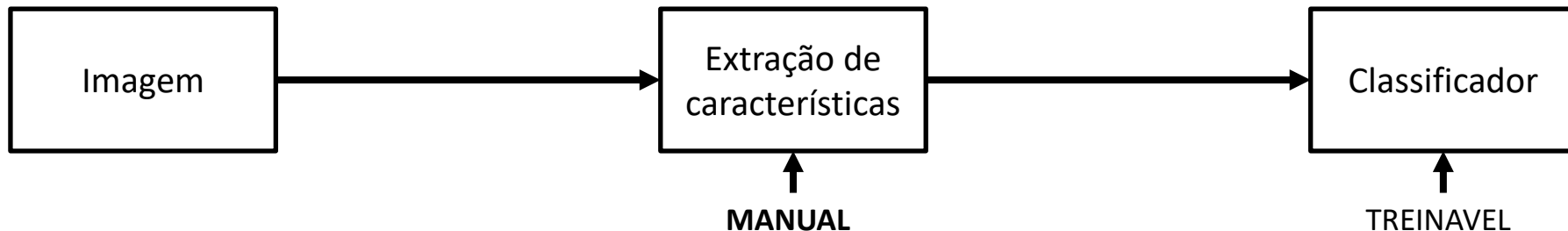
● *acer palmatum*
● *aesculus chinensis*
● *cercis chinensis*



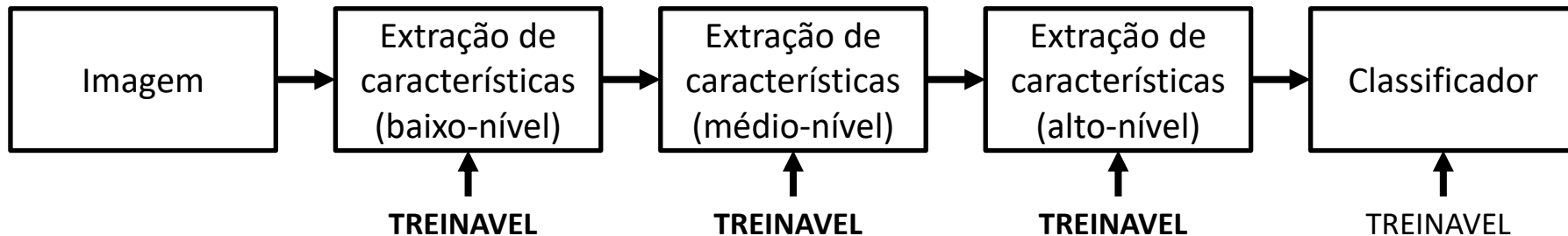
PIPELINES DE CLASSIFICAÇÃO

Pipelines de classificação

Pipeline clássico de classificação de imagens



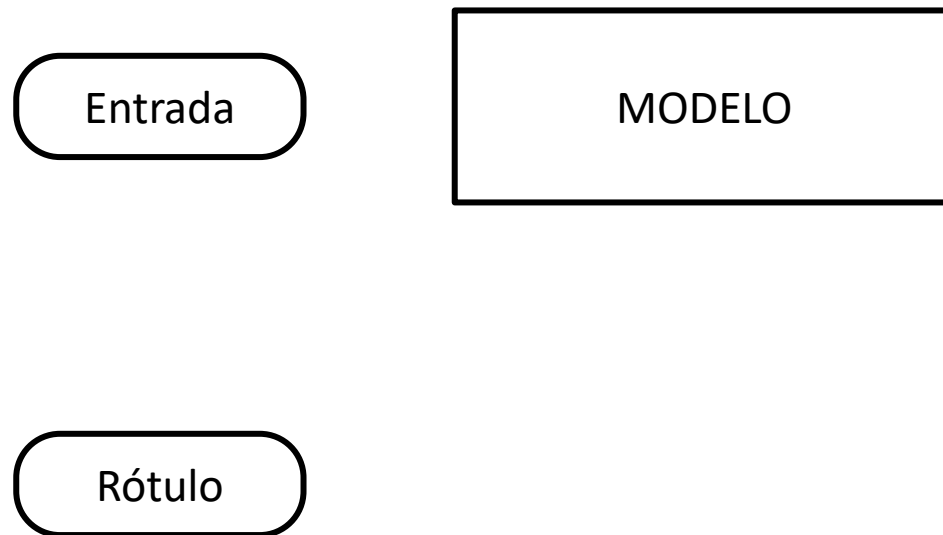
Deep Learning

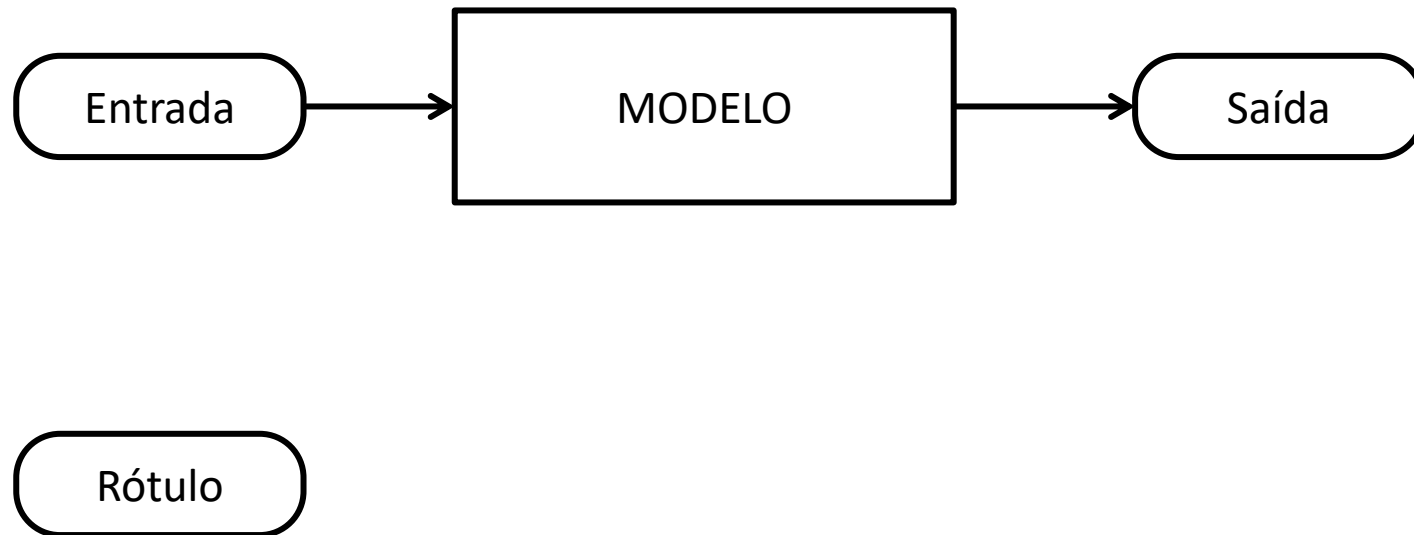


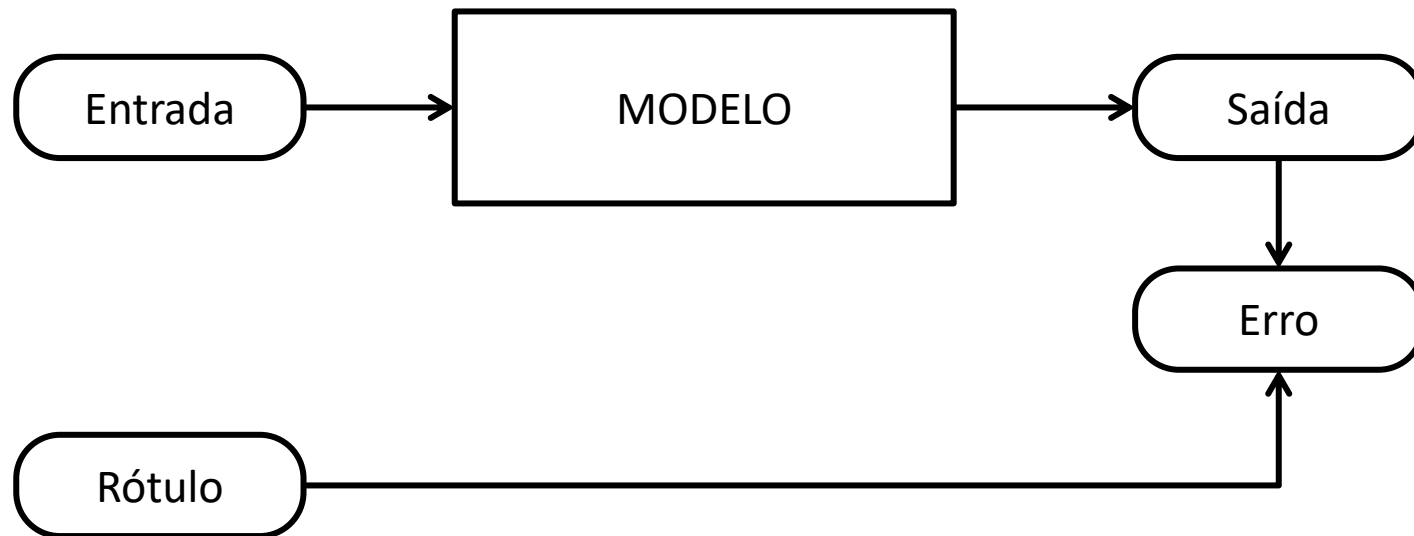
MODELOS DE APRENDIZADO

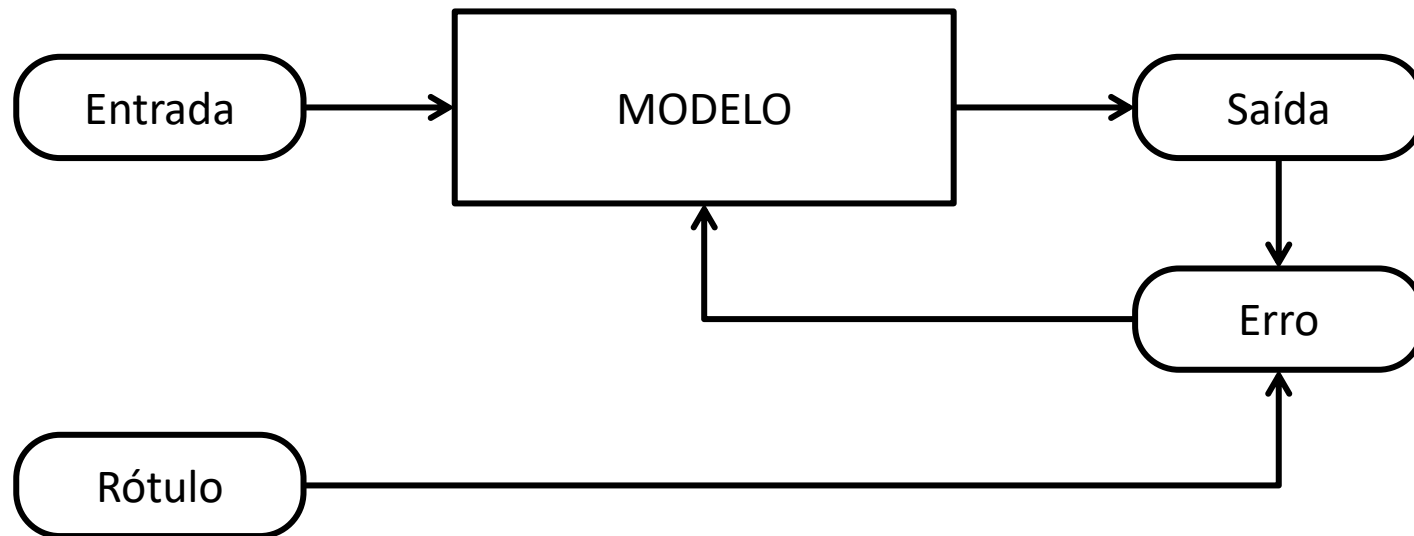
Modelos de aprendizado

- Aprendizado supervisionado
- Aprendizado não supervisionado
- Aprendizado por reforço
- Aprendizado semi-supervisionado
- Aprendizado auto-supervisionado

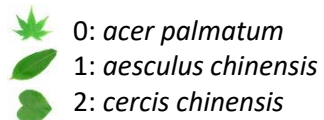
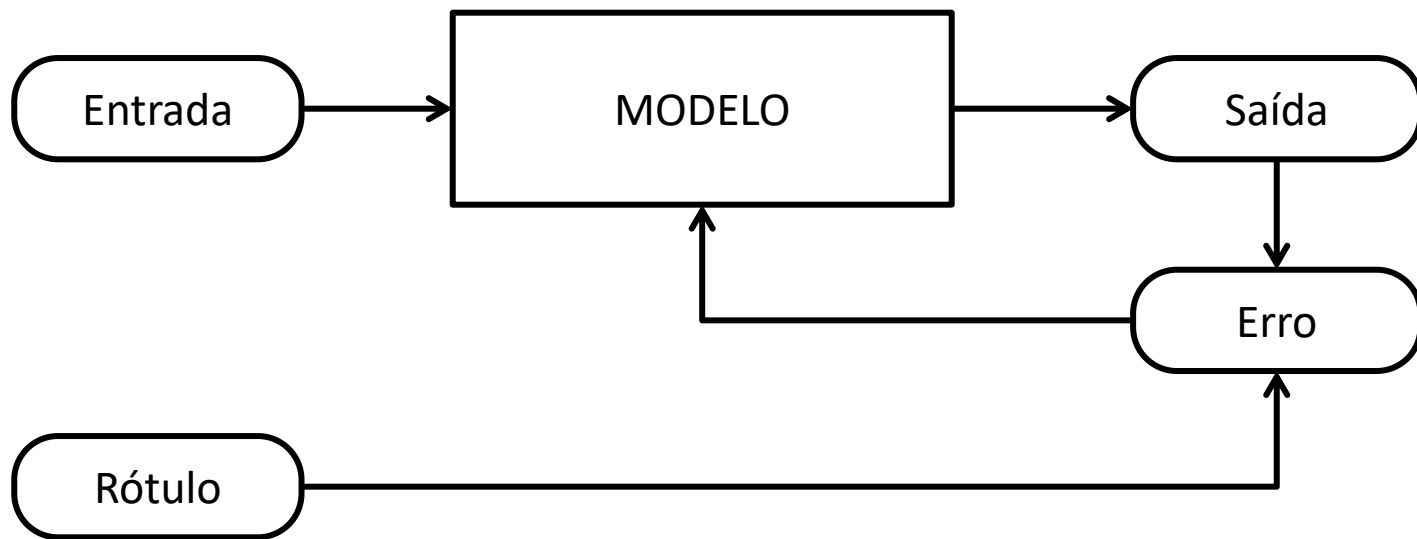
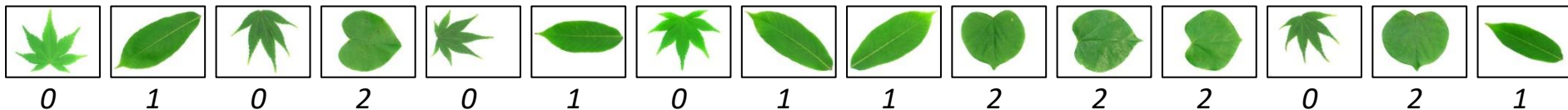




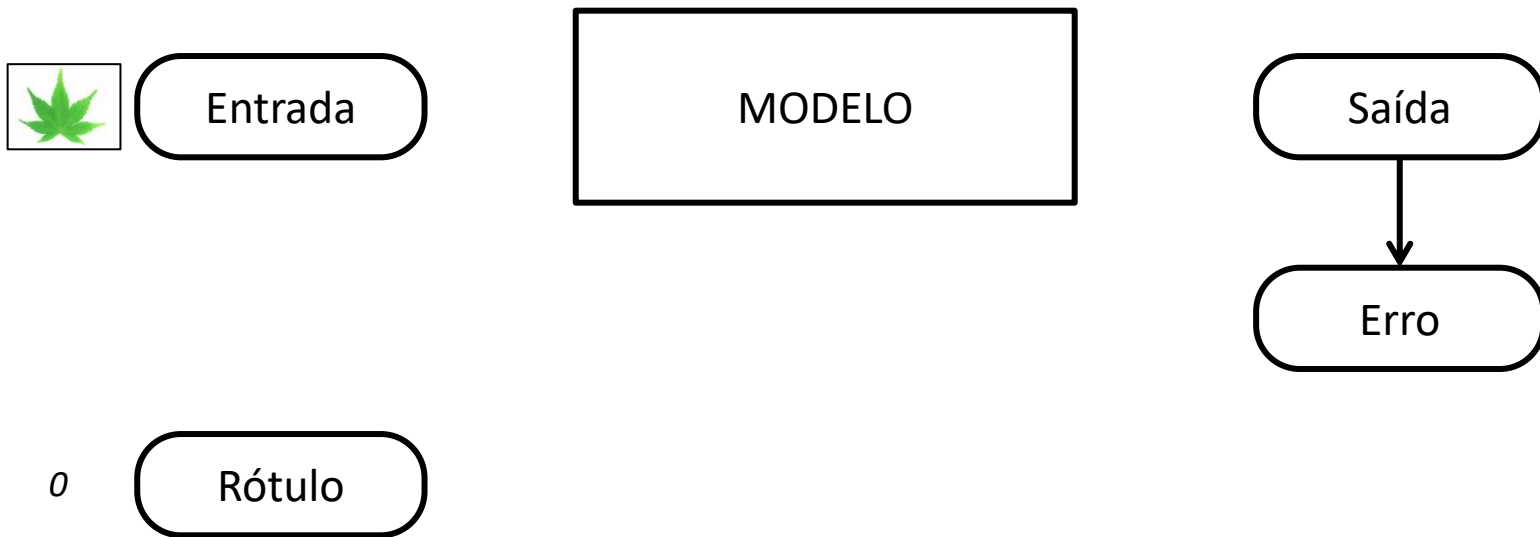
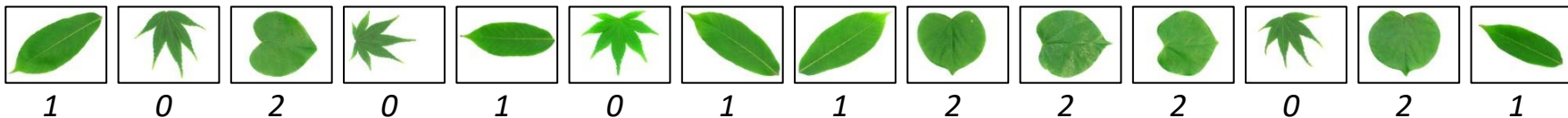







Exemplo: Aprendizado supervisionado

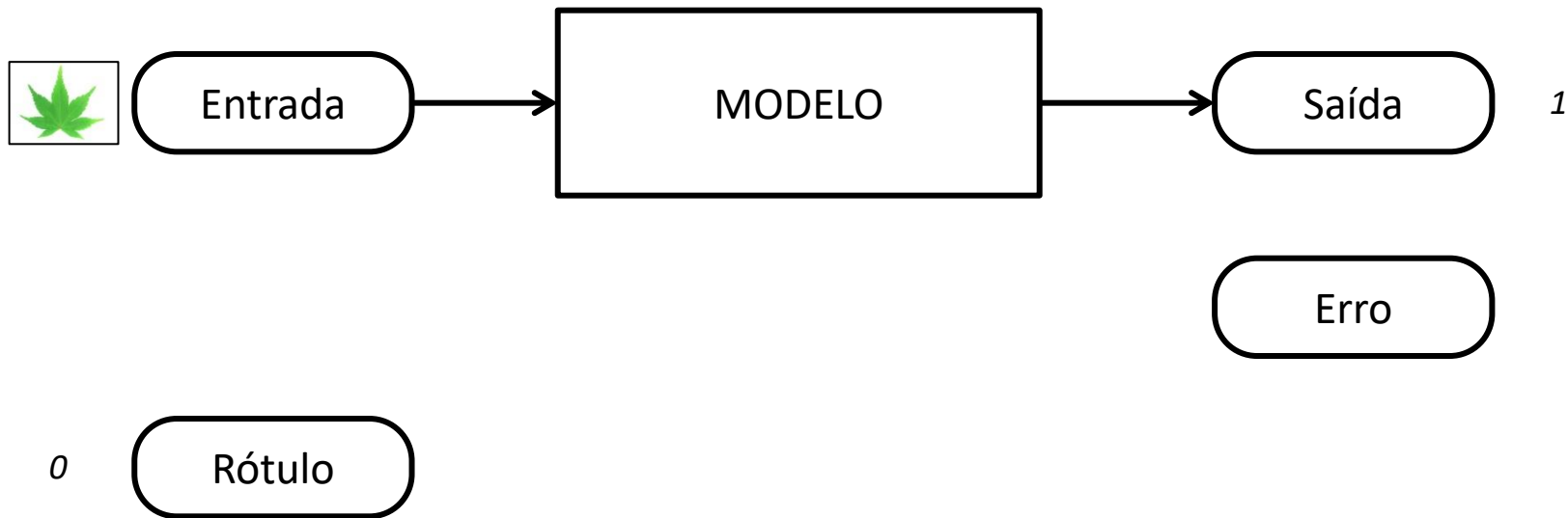
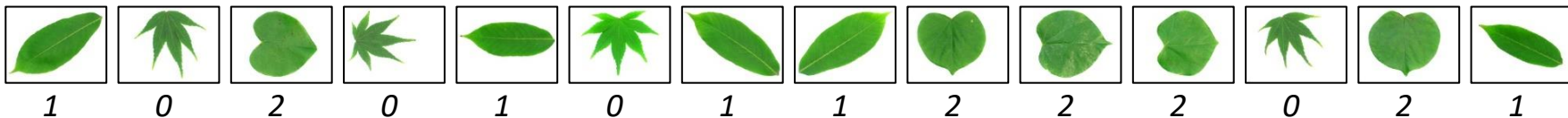





Exemplo: Aprendizado supervisionado



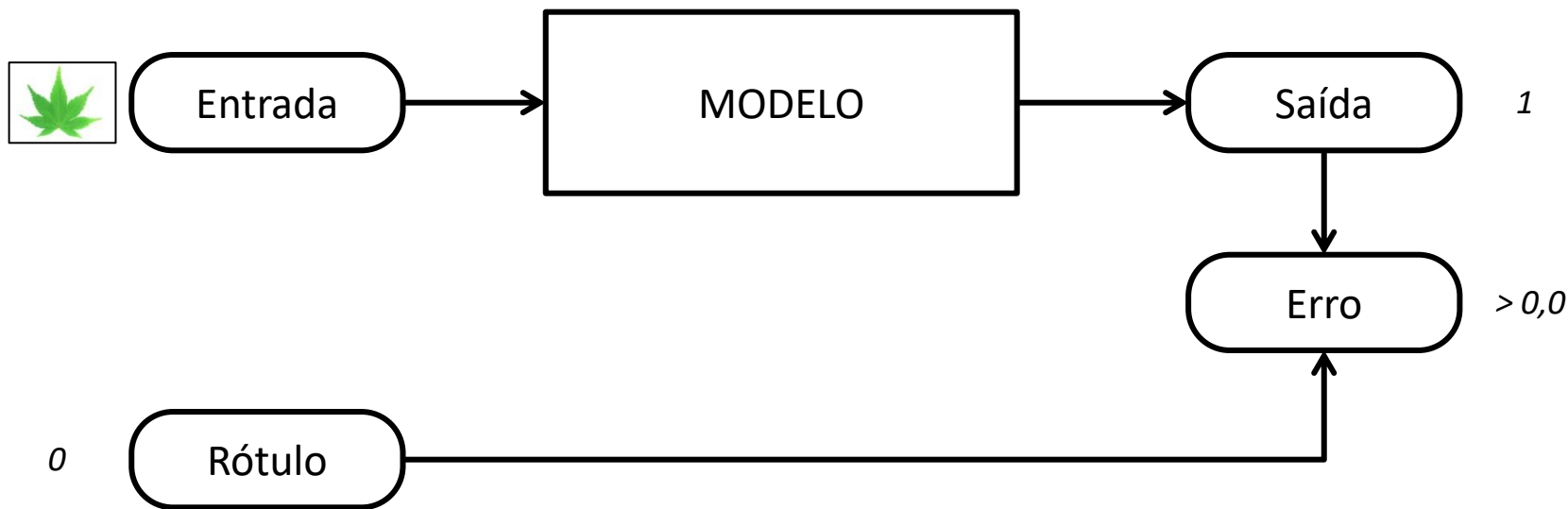
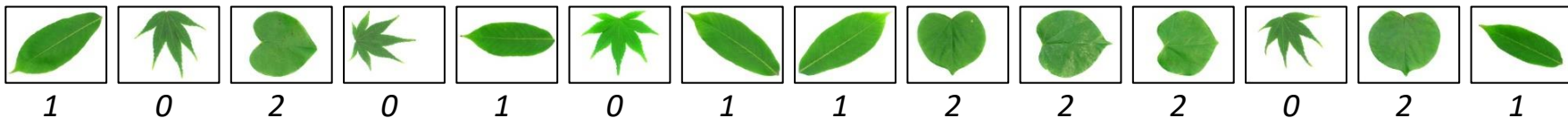
 0: *acer palmatum*
 1: *aesculus chinensis*
 2: *cercis chinensis*




Exemplo: Aprendizado supervisionado



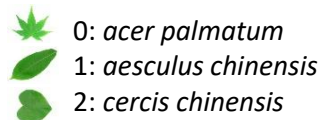
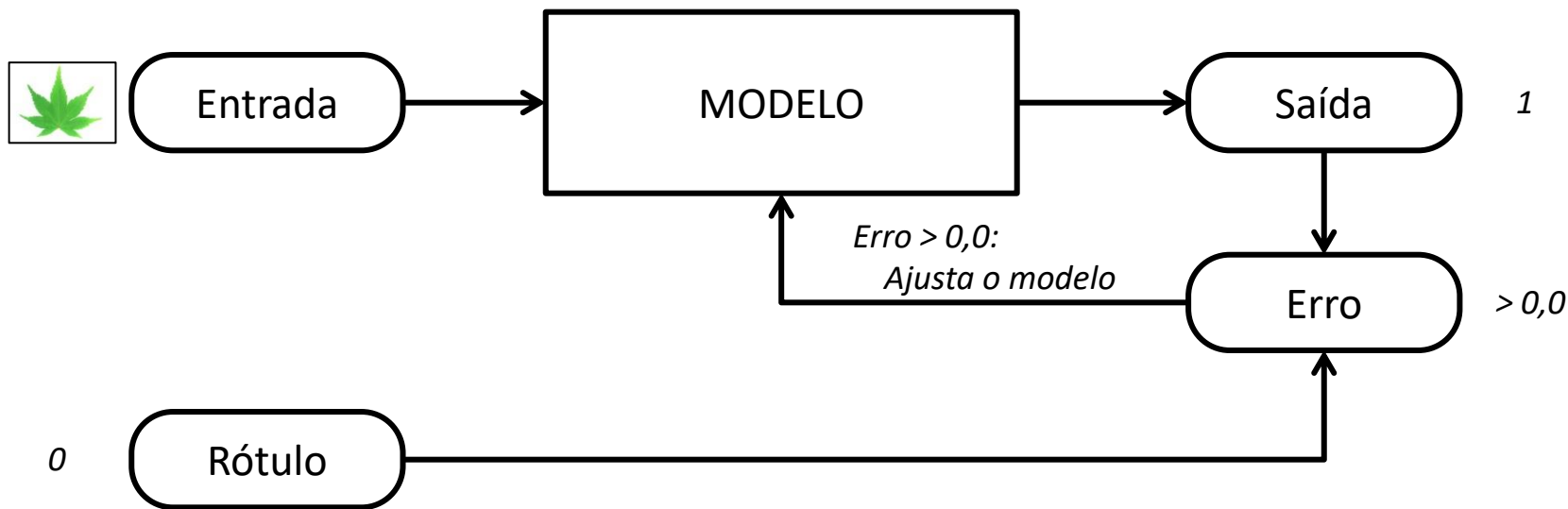
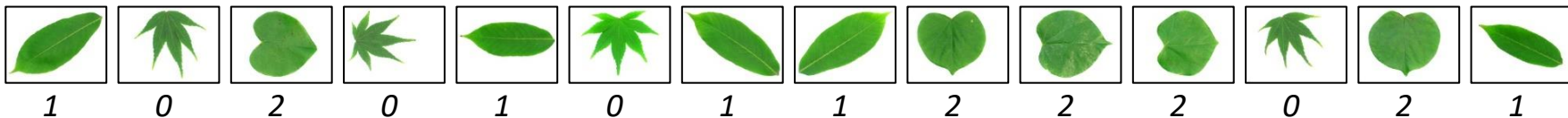
 0: *acer palmatum*
 1: *aesculus chinensis*
 2: *cercis chinensis*

Exemplo: Aprendizado supervisionado



 0: *acer palmatum*
 1: *aesculus chinensis*
 2: *cercis chinensis*

Exemplo: Aprendizado supervisionado



Exemplo: Aprendizado supervisionado



0



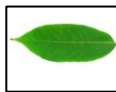
0



2



0



1



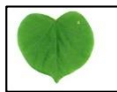
0



1



1



2



2



2



0



2



1



Entrada

MODELO

Saída

Erro

1

Rótulo



0: *acer palmatum*

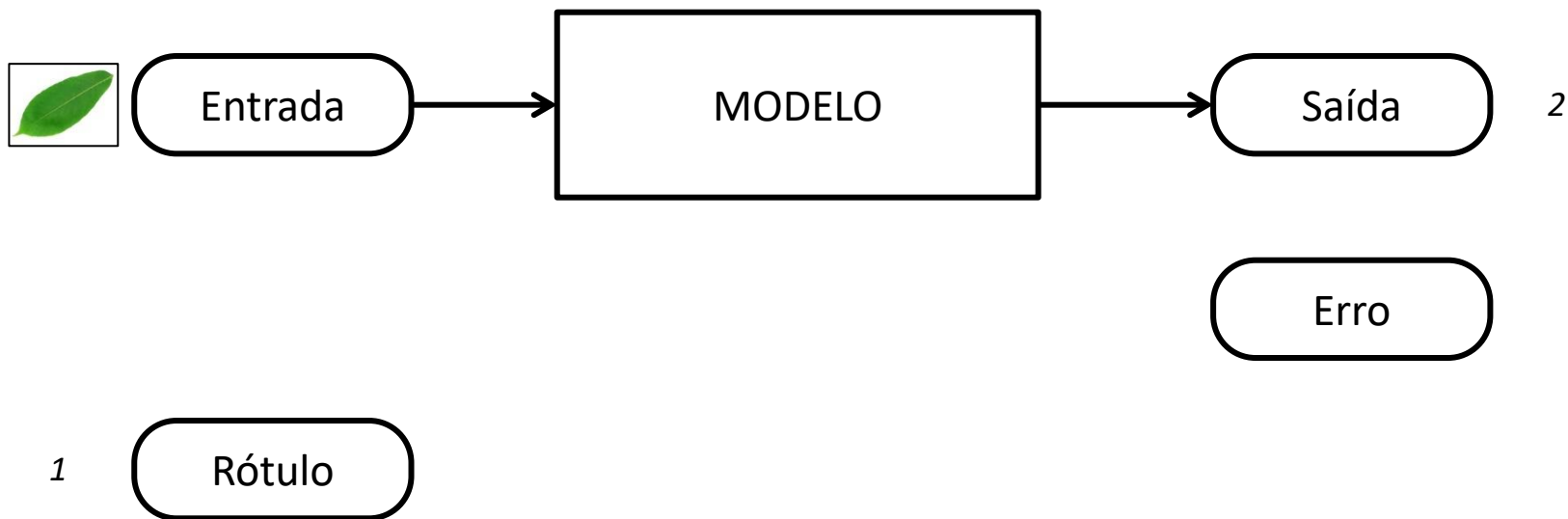
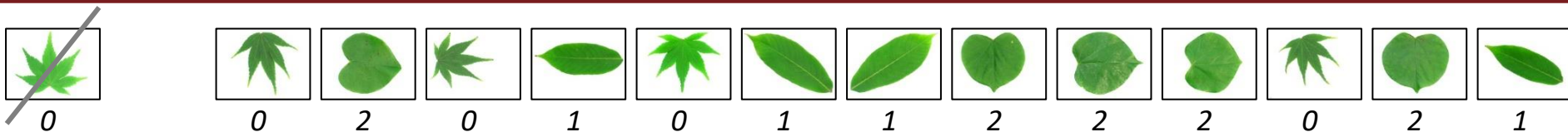





1: *aesculus chinensis*



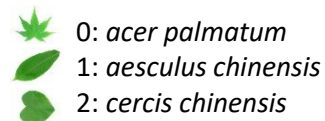
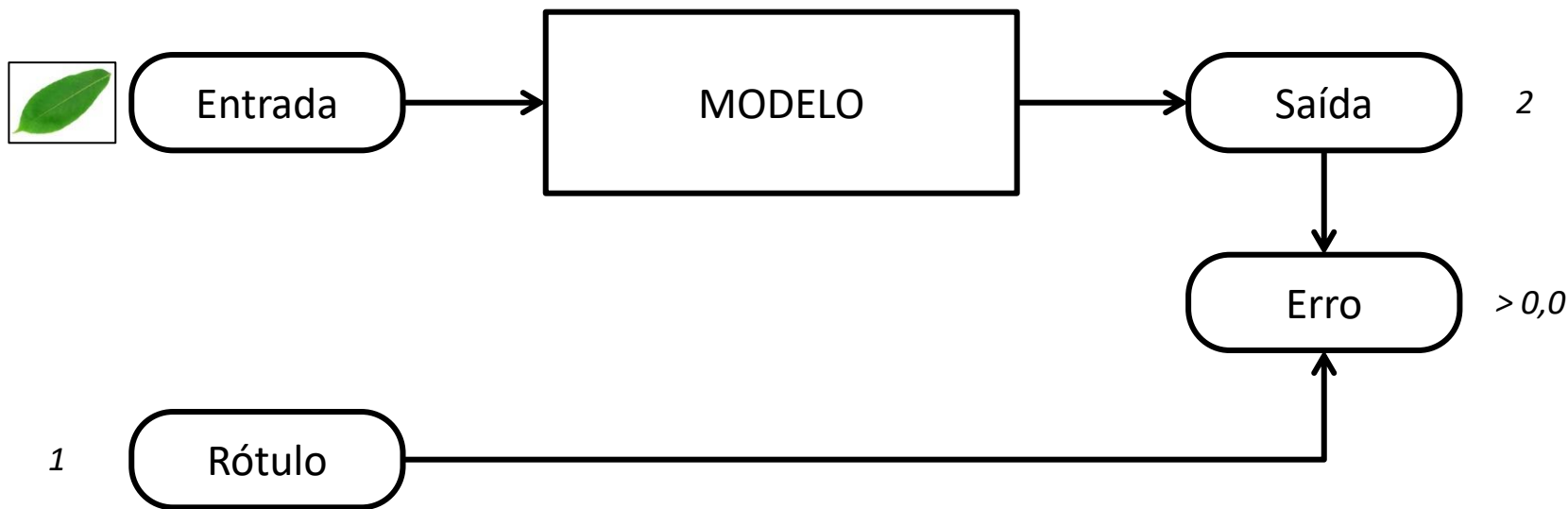
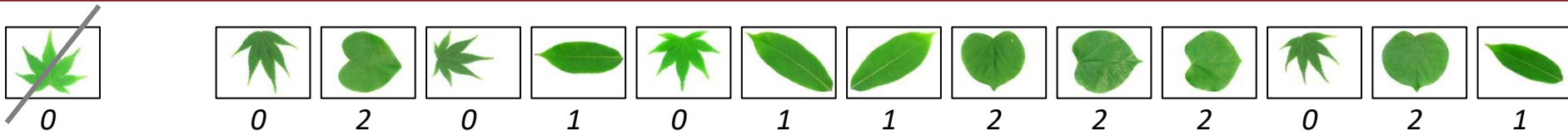
2: *cercis chinensis*

Exemplo: Aprendizado supervisionado

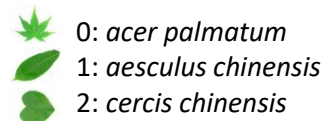
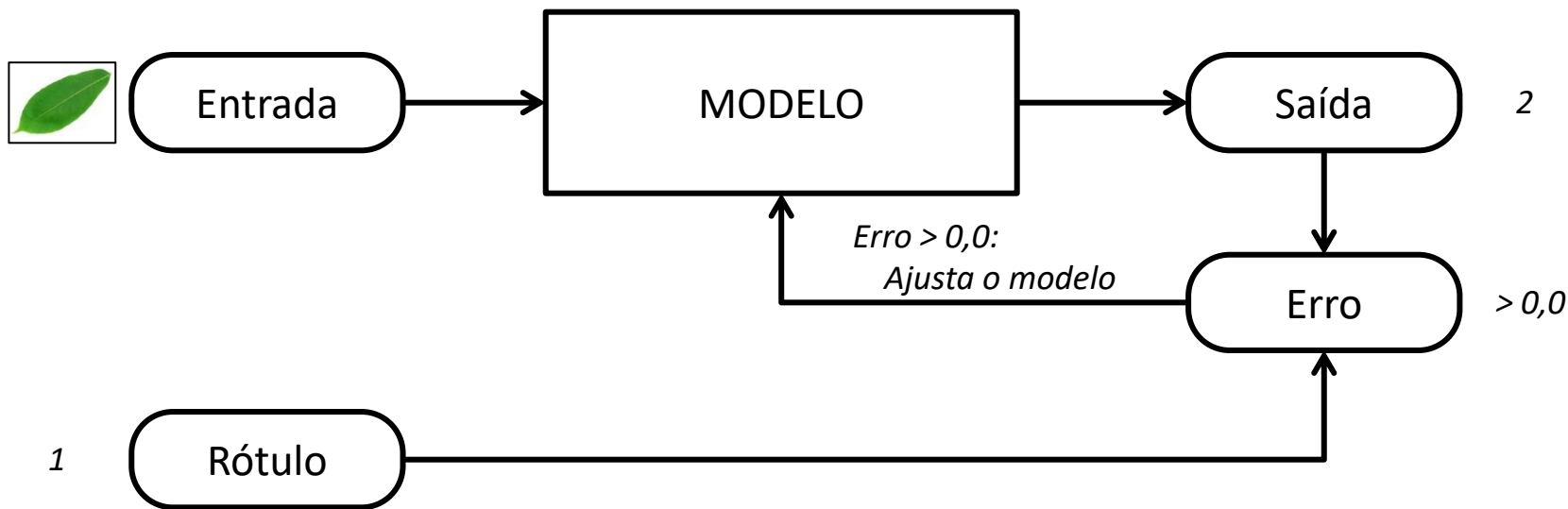
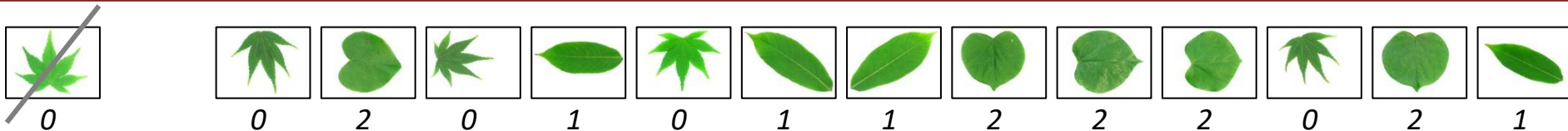


 0: *acer palmatum*
 1: *aesculus chinensis*
 2: *cercis chinensis*

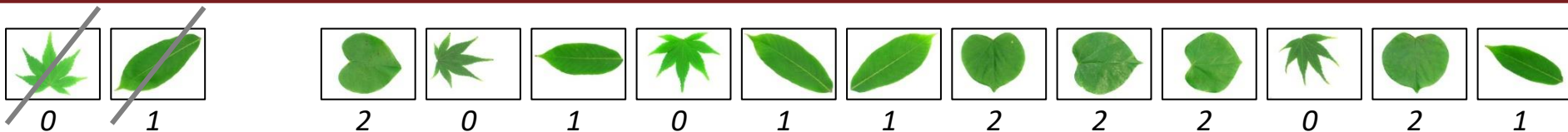
Exemplo: Aprendizado supervisionado



Exemplo: Aprendizado supervisionado



Exemplo: Aprendizado supervisionado



Entrada

MODELO

Saída

Erro

0

Rótulo

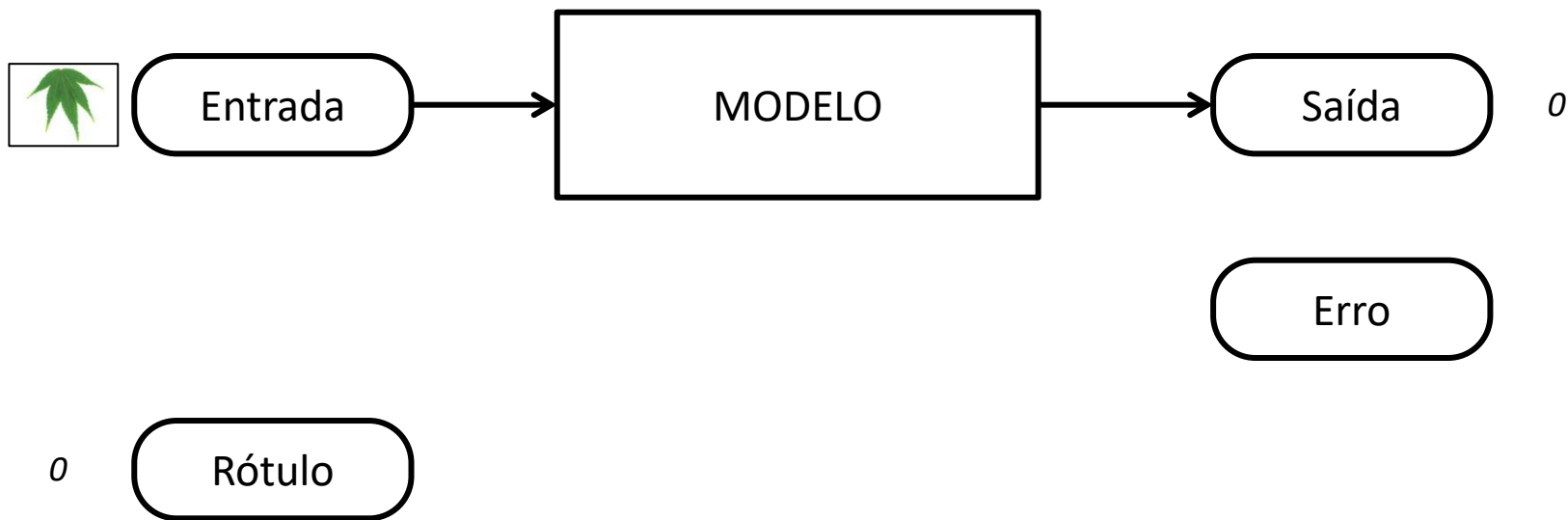
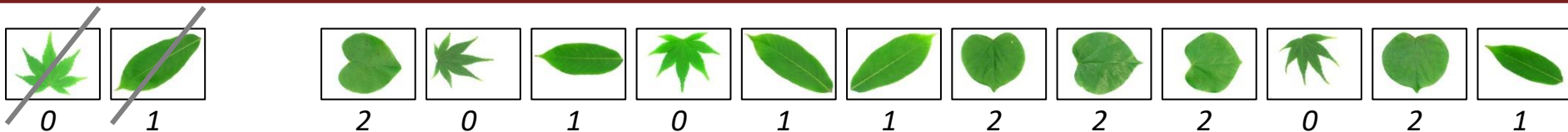





0: *acer palmatum*

1: *aesculus chinensis*

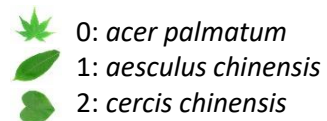
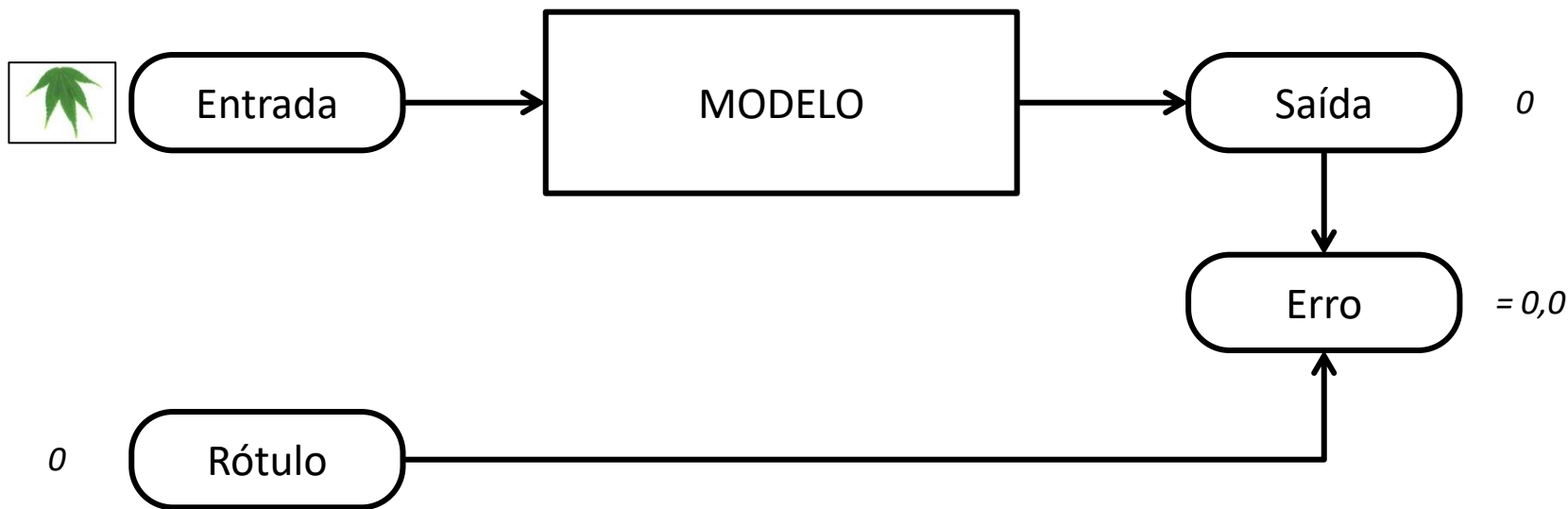
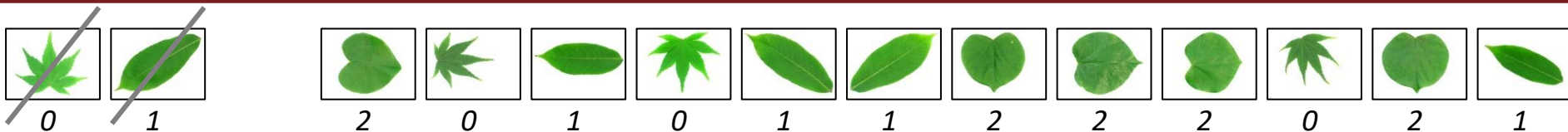
2: *cercis chinensis*

Exemplo: Aprendizado supervisionado

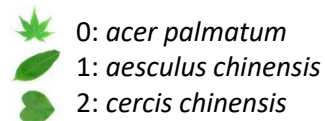
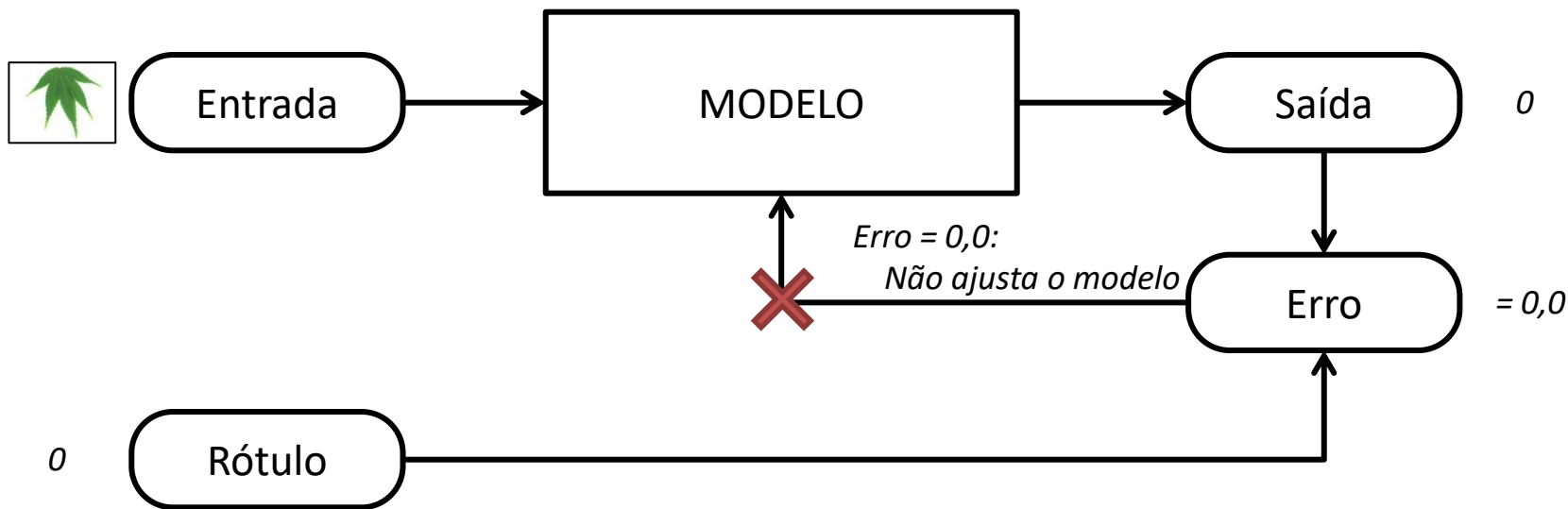
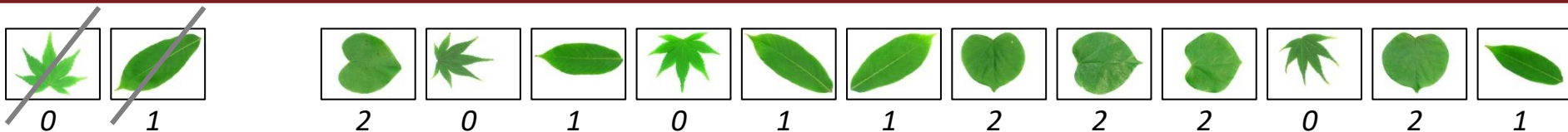


 0: *acer palmatum*
 1: *aesculus chinensis*
 2: *cercis chinensis*

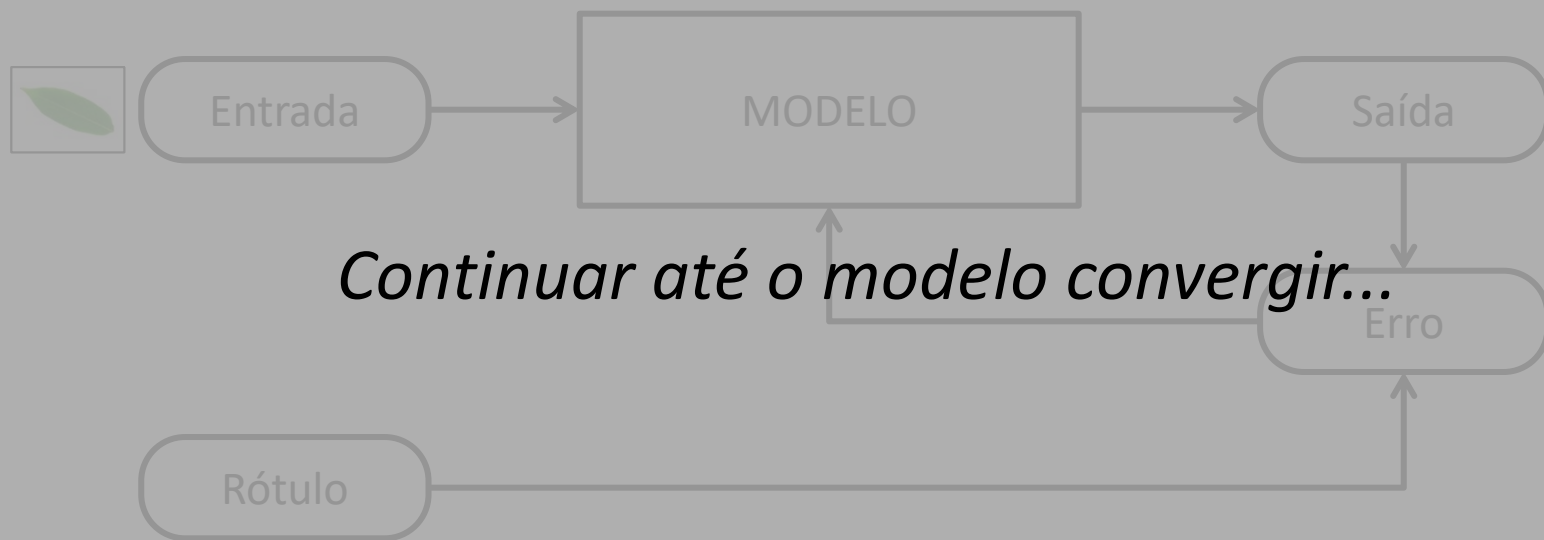
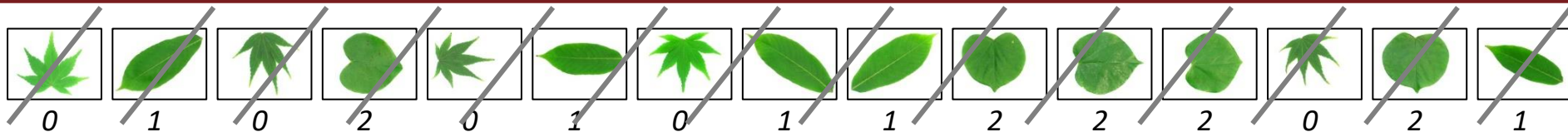
Exemplo: Aprendizado supervisionado






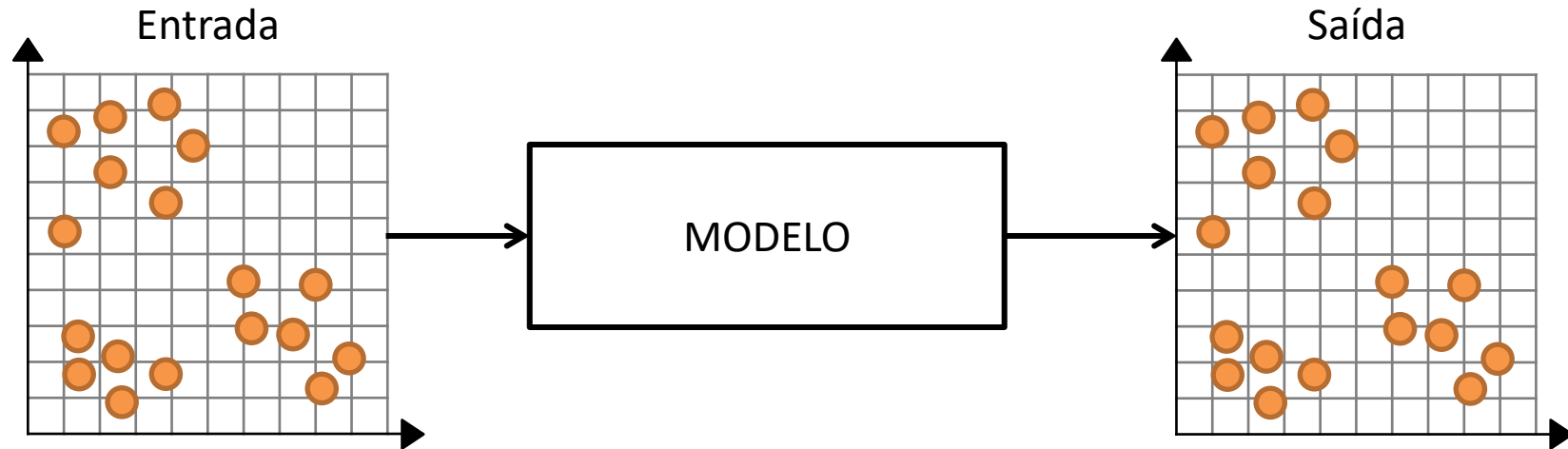
Exemplo: Aprendizado supervisionado

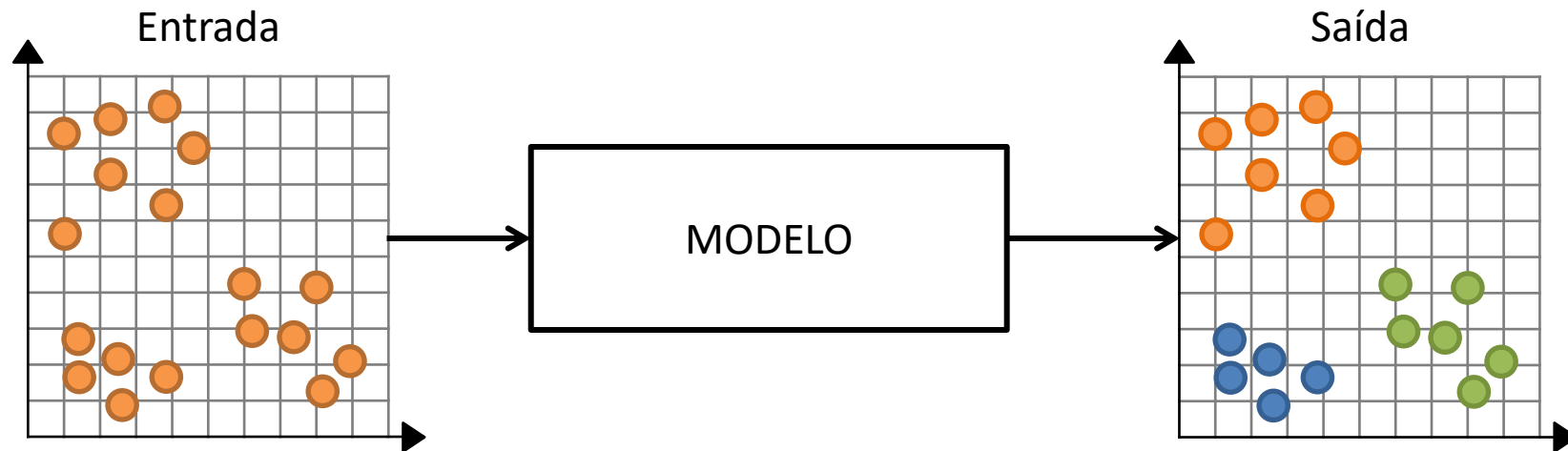


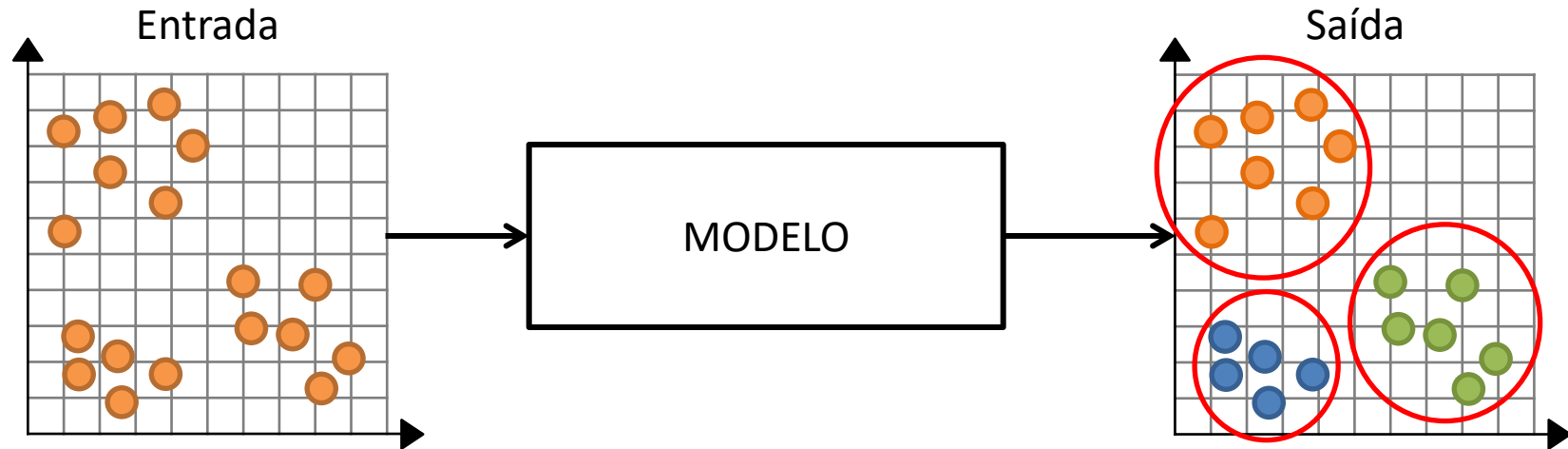
Exemplo: Aprendizado supervisionado



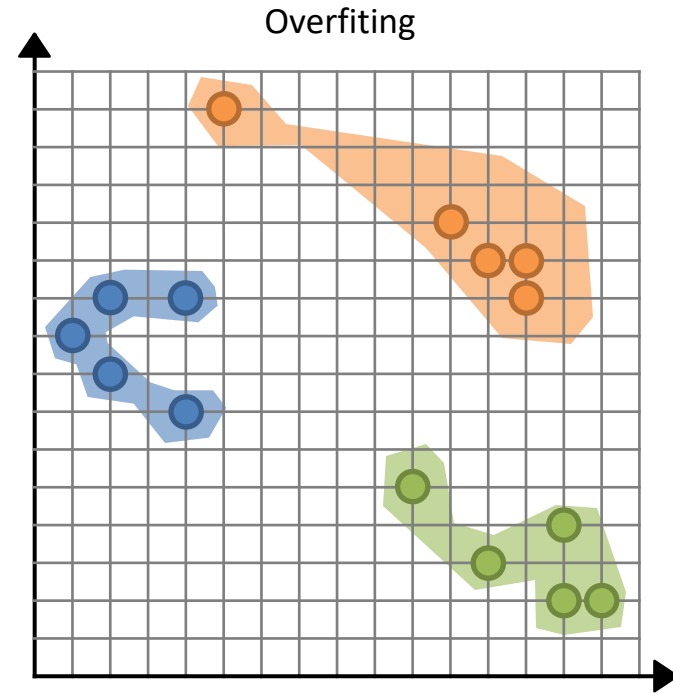
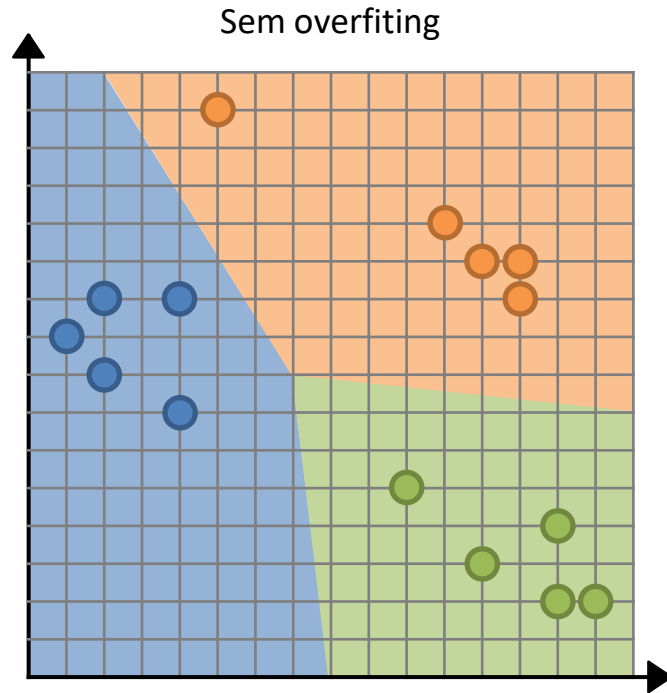
-  0: acer palmatum
-  1: aesculus chinensis
-  2: cercis chinensis

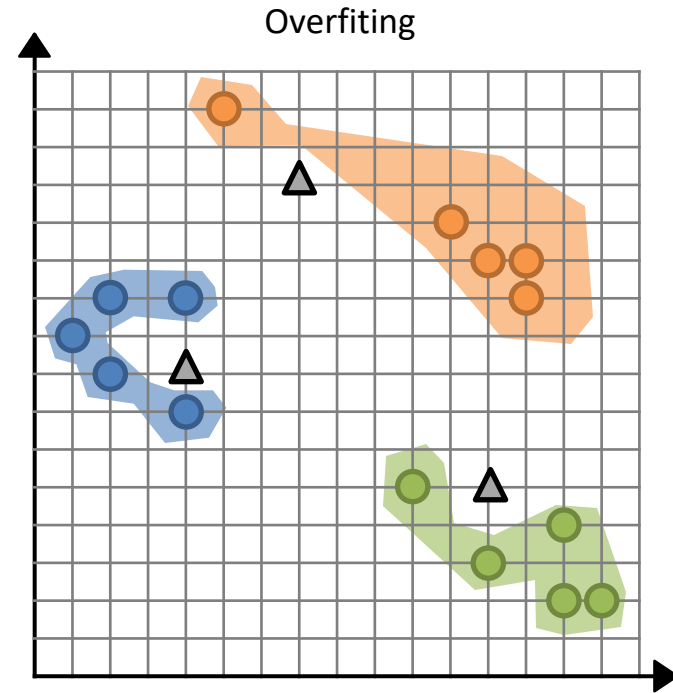
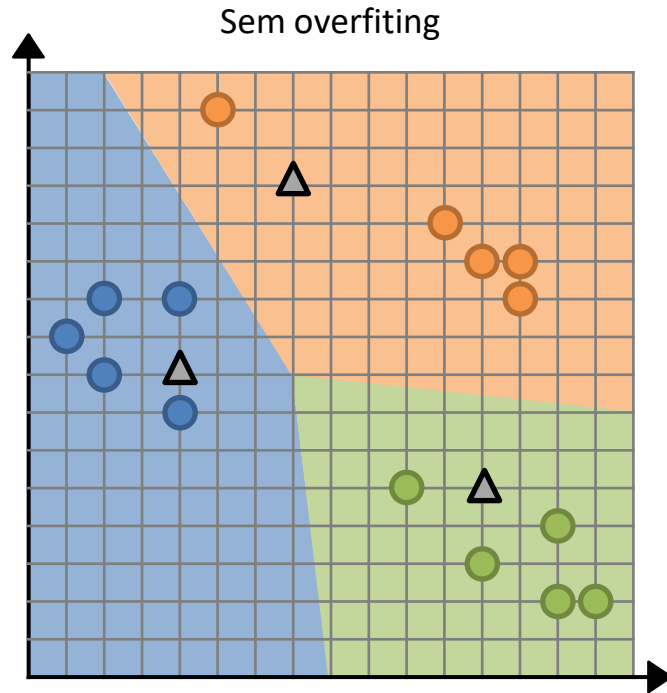


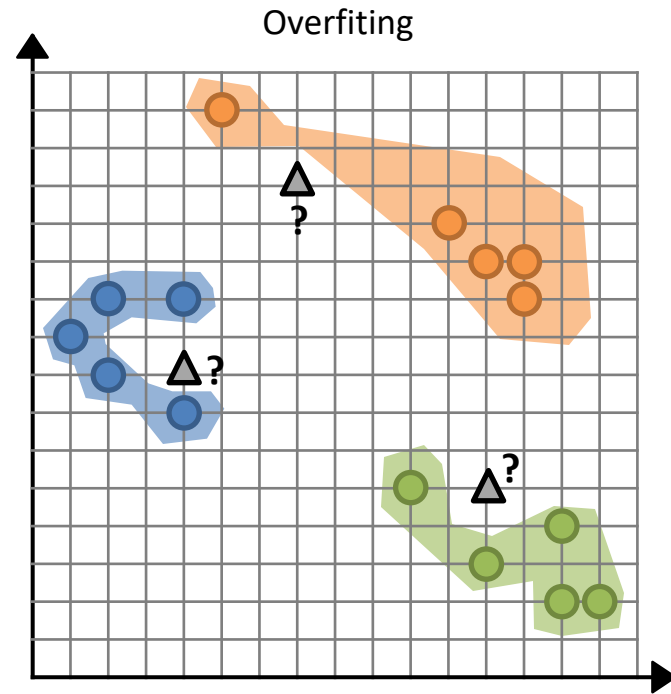
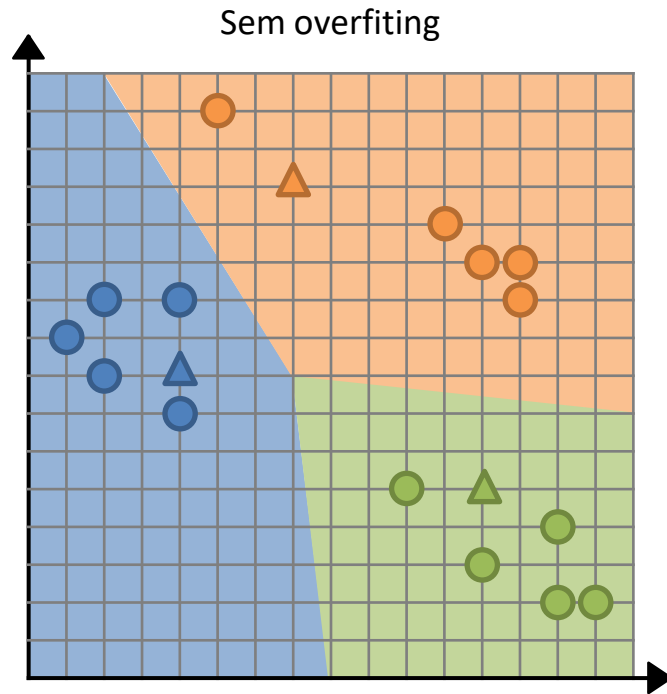


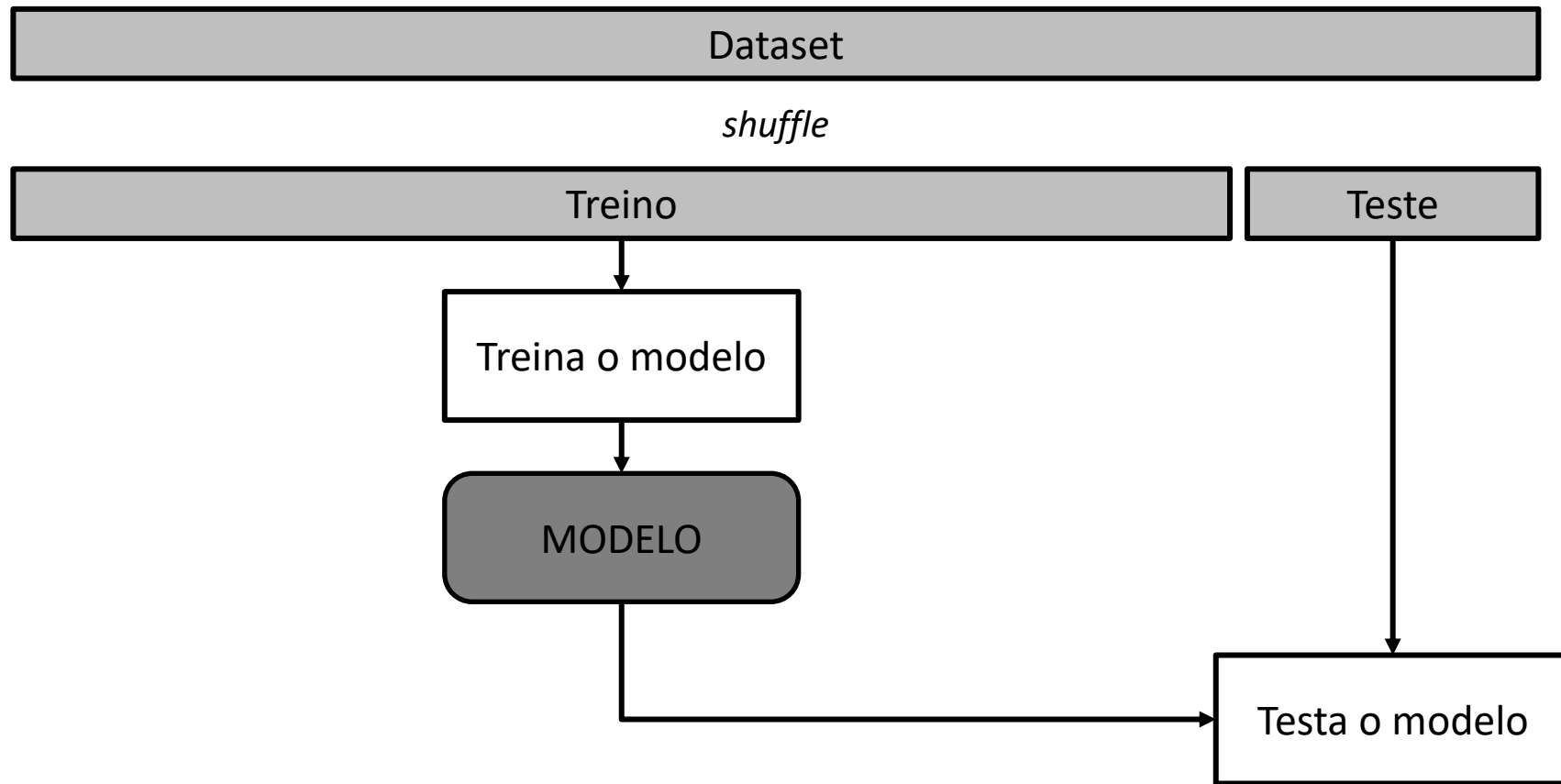


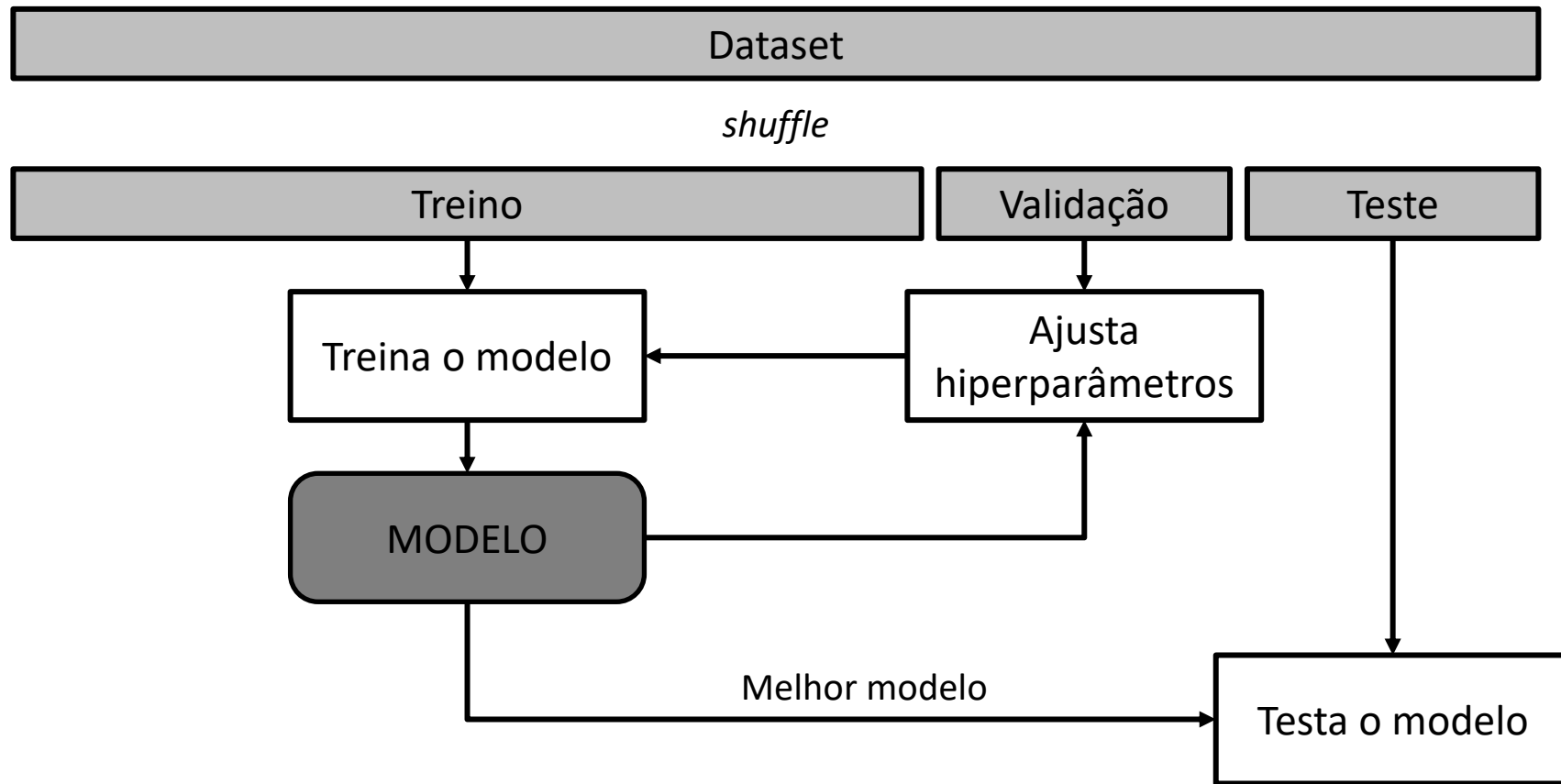
VALIDAÇÃO CRUZADA











Validação cruzada hold-out

Dataset

Imagem	excentricidade	área	classe
0	0,1	250	0
1	0,6	450	2
2	0,3	350	0
3	0,2	550	1
4	0,5	800	1
5	0,2	100	0
6	0,7	200	2
7	0,7	750	1
8	0,4	400	0
9	0,8	150	2
10	0,9	300	2
11	0,8	700	1
12	0,4	150	
13	0,4	300	
14	0,2	200	
15	0,7	250	

shuffle

Shuffled dataset

Imagem	excentricidade	área	classe
13	0,5	300	0
11	0,8	700	1
8	0,4	400	0
9	0,8	150	2
2	0,3	350	0
15	0,7	250	2
4	0,5	800	1
7	0,7	750	1
10	0,9	300	2
12	0,4	150	0
3	0,6	550	1
6	0,6	200	2
0	0,4	700	1
1	0,8	400	2
5	0,3	100	0
14	0,2	200	0

split

Treino

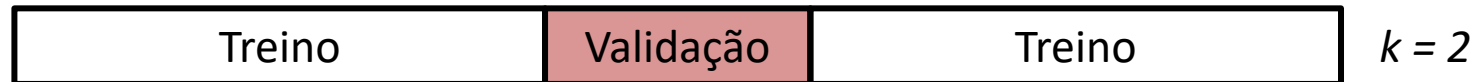
Imagem	excentricidade	área	classe
13	0,5	300	0
11	0,8	700	1
8	0,4	400	0
9	0,8	150	2
2	0,3	350	0
15	0,7	250	2
4	0,5	800	1
7	0,7	750	1
10	0,9	300	2
12	0,4	150	0
3	0,6	550	1
6	0,6	200	2

Teste

Imagem	excentricidade	área	classe
0	0,4	700	1
1	0,8	400	2
5	0,3	100	0
14	0,2	200	0



shuffle



NORMALIZAÇÃO

Normalização

- Transformada Normal de Características (Standard Scaler)

Treinamento (X_{train})

Imagem	excentricidade	área
13	0,50	300
11	0,80	700
8	0,40	400
9	0,80	150
2	0,30	350
15	0,70	250
4	0,50	800
7	0,70	750
10	0,90	300
12	0,40	150
3	0,60	550
6	0,60	200
Média:	0.60	408.33
Desv. Pad.:	0.1859	234.35

Treinamento normalizado (X'_{train})

Imagem	excentricidade	área
13		
11		
8		
9		
2		
15		
4		
7		
10		
12		
3		
6		
Média:		
Desv. Pad.:		

Teste (X_{test})

Imagem	excentricidade	área
0	0,4	700
1	0,8	400
5	0,3	100
14	0,2	200

Teste normalizado (X'_{test})

Imagem	excentricidade	área
0		
1		
5		
14		
Média:		
Desv. Pad.:		

Normalização

- Transformada Normal de Características (Standard Scaler)

Treinamento (X_{train})

Imagem	excentricidade	área
13	0,50	300
11	0,80	700
8	0,40	400
9	0,80	150
2	0,30	350
15	0,70	250
4	0,50	800
7	0,70	750
10	0,90	300
12	0,40	150
3	0,60	550
6	0,60	200
Média:	0.60	408.33
Desv. Pad.:	0.1859	234.35

Treinamento normalizado (X'_{train})

Imagem	excentricidade	área
13		
11		
8		
9		
2		
15		
4		
7		
10		
12		
3		
6		
Média:		
Desv. Pad.:		

Teste (X_{test})

Imagem	excentricidade	área
0	0,4	700
1	0,8	400
5	0,3	100
14	0,2	200

Teste normalizado (X'_{test})

Imagem	excentricidade	área
0		
1		
5		
14		
Média:		
Desv. Pad.:		

$$X'_{train} = \frac{X_{train} - \text{mean}(X_{train})}{\text{std}(X_{train})}$$

$$X'_{test} = \frac{X_{test} - \text{mean}(X_{train})}{\text{std}(X_{train})}$$

Normalização

- Transformada Normal de Características (Standard Scaler)

Treinamento (X_{train})

Imagem	excentricidade	área
13	0,50	300
11	0,80	700
8	0,40	400
9	0,80	150
2	0,30	350
15	0,70	250
4	0,50	800
7	0,70	750
10	0,90	300
12	0,40	150
3	0,60	550
6	0,60	200
Média:	0.60	408.33
Desv. Pad.:	0.1859	234.35

Treinamento normalizado (X'_{train})

Imagem	excentricidade	área
13	-0.5380	-0.4622
11	1.0760	1.2445
8	-1.0760	-0.0355
9	1.0760	-1.1022
2	-1.6140	-0.2489
15	0.5380	-0.6756
4	-0.5380	1.6712
7	0.5380	1.4578
10	1.6140	-0.4622
12	-1.0760	-1.1022
3	0.0000	0.6044
6	0.0000	-0.8889
Média:	0.00	1.00
Desv. Pad.:	0.00	1.00

Teste (X_{test})

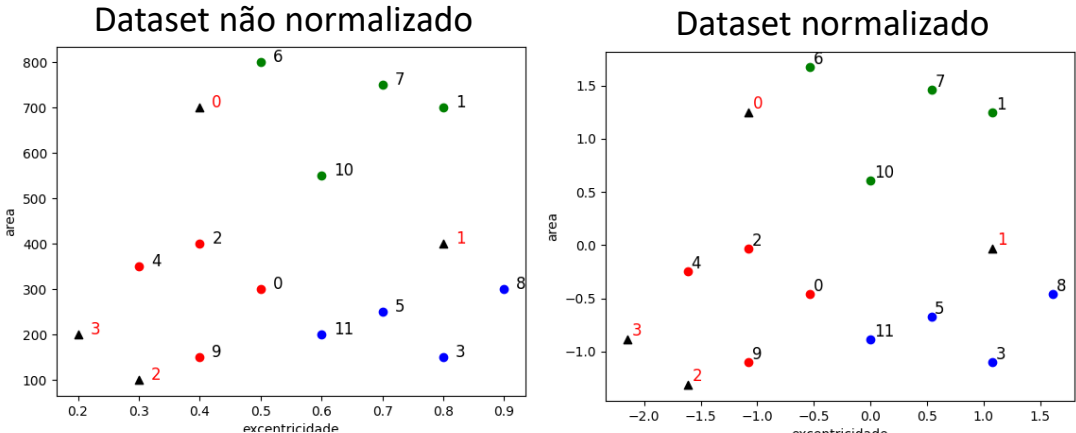
Imagem	excentricidade	área
0	0,4	700
1	0,8	400
5	0,3	100
14	0,2	200

Teste normalizado (X'_{test})

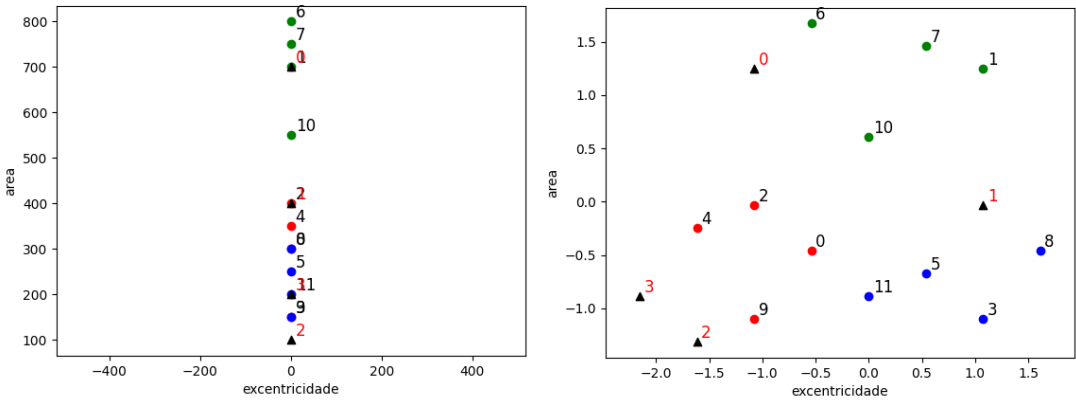
Imagem	excentricidade	área
0	-1.0760	1.244528
1	1.0760	-0.035558
5	-1.6140	-1.315644
14	-2.1521	-0.888949
Média:	-0.9415	-0.2489
Desv. Pad.:	1.4150	1.1289

$$X'_{train} = \frac{X_{train} - \text{mean}(X_{train})}{\text{std}(X_{train})} \quad X'_{test} = \frac{X_{test} - \text{mean}(X_{train})}{\text{std}(X_{train})}$$

Aspect ratio ajustado



Aspect ratio real



AVALIAÇÃO DOS RESULTADOS

Matriz de confusão

- **Verdadeiro positivo (TP):**
 - Objetos da classe C_1 classificados como C_1 .
- **Verdadeiro negativo (TN):**
 - Objetos de outras classes (C_2 e C_3) classificados como não sendo C_1 .
- **Falso positivo (FP) (erro tipo I):**
 - Objetos classificados como C_1 mas pertencem a outras classes (C_2 ou C_3).
- **Falso negativo (FN) (erro tipo II):**
 - Objetos da classe C_1 classificados como outras classes (C_2 ou C_3).

		Classificação			
		Classe C1	Classe C2	Classe C3	Soma
Classe real	Classe C1	5	3	0	8
	Classe C2	2	3	1	6
	Classe C3	0	2	11	13
	Soma	7	8	12	

Matriz de confusão

- **Verdadeiro positivo (TP):**
 - Objetos da classe C_1 classificados como C_1 .
- **Verdadeiro negativo (TN):**
 - Objetos de outras classes (C_2 e C_3) classificados como não sendo C_1 .
- **Falso positivo (FP) (erro tipo I):**
 - Objetos classificados como C_1 mas pertencem a outras classes (C_2 ou C_3).
- **Falso negativo (FN) (erro tipo II):**
 - Objetos da classe C_1 classificados como outras classes (C_2 ou C_3).

Classe C1		Classificação			
		Classe C1		Outras	
Classe real	Classe C1	5	TP	3	FN
	Outras	2	FP	17	TN

		Classificação			
		Classe C1	Classe C2	Classe C3	Soma
Classe real	Classe C1	5	3	0	8
	Classe C2	2	3	1	6
	Classe C3	0	2	11	13
	Soma	7	8	12	

Matriz de confusão

- **Verdadeiro positivo (TP):**
 - Objetos da classe C_1 classificados como C_1 .
- **Verdadeiro negativo (TN):**
 - Objetos de outras classes (C_2 e C_3) classificados como não sendo C_1 .
- **Falso positivo (FP) (erro tipo I):**
 - Objetos classificados como C_1 mas pertencem a outras classes (C_2 ou C_3).
- **Falso negativo (FN) (erro tipo II):**
 - Objetos da classe C_1 classificados como outras classes (C_2 ou C_3).

Classe C1		Classificação	
		Classe C1	Outras
Classe real	Classe C1	5 TP	3 FN
	Outras	2 FP	17 TN

		Classificação			
		Classe C1	Classe C2	Classe C3	Soma
Classe real	Classe C1	5	3	0	8
	Classe C2	2	3	1	6
	Classe C3	0	2	11	13
	Soma	7	8	12	

Matriz de confusão

- **Verdadeiro positivo (TP):**
 - Objetos da classe C_1 classificados como C_1 .
- **Verdadeiro negativo (TN):**
 - Objetos de outras classes (C_2 e C_3) classificados como não sendo C_1 .
- **Falso positivo (FP) (erro tipo I):**
 - Objetos classificados como C_1 mas pertencem a outras classes (C_2 ou C_3).
- **Falso negativo (FN) (erro tipo II):**
 - Objetos da classe C_1 classificados como outras classes (C_2 ou C_3).

		Classificação			
		Classe C1	Classe C2	Classe C3	Soma
Classe real	Classe C1	5	3	0	8
	Classe C2	2	3	1	6
	Classe C3	0	2	11	13
	Soma	7	8	12	

Classe C1		Classificação			
		Classe C1		Outras	
Classe real	Classe C1	5	TP	3	FN
	Outras	2	FP	17	TN

Classe C2		Classificação			
		Classe C2		Outras	
Classe real	Classe C2	3	TP	3	FN
	Outras	5	FP	16	TN

Matriz de confusão

- **Verdadeiro positivo (TP):**
 - Objetos da classe C_1 classificados como C_1 .
- **Verdadeiro negativo (TN):**
 - Objetos de outras classes (C_2 e C_3) classificados como não sendo C_1 .
- **Falso positivo (FP) (erro tipo I):**
 - Objetos classificados como C_1 mas pertencem a outras classes (C_2 ou C_3).
- **Falso negativo (FN) (erro tipo II):**
 - Objetos da classe C_1 classificados como outras classes (C_2 ou C_3).

		Classificação			
		Classe C1	Classe C2	Classe C3	Soma
Classe real	Classe C1	5	3	0	8
	Classe C2	2	3	1	6
	Classe C3	0	2	11	13
	Soma	7	8	12	

Classe C1		Classificação			
		Classe C1		Outras	
Classe real	Classe C1	5	TP	3	FN
	Outras	2	FP	17	TN

Classe C2		Classificação			
		Classe C2		Outras	
Classe real	Classe C2	3	TP	3	FN
	Outras	5	FP	16	TN

Classe C3		Classificação			
		Classe C3		Outras	
Classe real	Classe C3	11	TP	2	FN
	Outras	1	FP	13	TN

Acurácia, precisão, sensibilidade e F1-score

- Acurácia (*Accuracy*):
 - $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
- Precisão (*Precision*):
 - $Precision = \frac{TP}{TP+FP}$
- Sensitividade (*Recall*):
 - $Recall = \frac{TP}{TP+FN}$
- Índice-F1 (*F1-score*):
 - $F1 - score = \frac{2 \times TP}{2 \times TP + FP + FN}$
- Suporte (*Support*):
 - $Support = TP + FN$

Acurácia, precisão, sensibilidade e F1-score

- Acurácia (*Accuracy*):
 - $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
 - Precisão (*Precision*):
 - $Precision = \frac{TP}{TP+FP}$
 - Sensitividade (*Recall*):
 - $Recall = \frac{TP}{TP+FN}$
 - Índice-F1 (*F1-score*):
 - $F1 - score = \frac{2 \times TP}{2 \times TP + FP + FN}$
 - Suporte (*Support*):
 - $Support = TP + FN$
- *Quão próxima a classificação está do valor verdadeiro.*
 - *A capacidade do classificador de não rotular uma amostra negativa como positiva.*
 - *A capacidade do classificador de encontrar todas as amostras positivas.*
 - *A média harmônica ponderada da precisão e recuperação.*
 - *O número de ocorrências de cada classe real (verdadeira).*

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html

Acurácia, precisão, sensibilidade e F1-score

- Acurácia (*Accuracy*):
 - $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
- Precisão (*Precision*):
 - $Precision = \frac{TP}{TP+FP}$
- Sensitividade (*Recall*):
 - $Recall = \frac{TP}{TP+FN}$
- Índice-F1 (*F1-score*):
 - $F1 - score = \frac{2 \times TP}{2 \times TP + FP + FN}$
- Suporte (*Support*):
 - $Support = TP + FN$

Classes	TP	TN	FP	FN	Acurácia	Precisão	Sensitividade	Índice-F1	Suporte
C1	5	17	2	3					
C2	3	16	5	3					
C3	11	13	1	2					
MÉDIA									
DESV. PAD									

Acurácia, precisão, sensibilidade e F1-score

- Acurácia (*Accuracy*):
 - $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
- Precisão (*Precision*):
 - $Precision = \frac{TP}{TP+FP}$
- Sensitividade (*Recall*):
 - $Recall = \frac{TP}{TP+FN}$
- Índice-F1 (*F1-score*):
 - $F1 - score = \frac{2 \times TP}{2 \times TP + FP + FN}$
- Suporte (*Support*):
 - $Support = TP + FN$

Classes	TP	TN	FP	FN	Acurácia	Precisão	Sensitividade	Índice-F1	Suporte
C1	5	17	2	3	0.8148	0.7143	0.6250	0.6667	8
C2	3	16	5	3	0.7037	0.3750	0.5000	0.4286	6
C3	11	13	1	2	0.8889	0.9167	0.8462	0.8800	13
MÉDIA									
DESV. PAD									

Acurácia, precisão, sensibilidade e F1-score

- Acurácia (*Accuracy*):
 - $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
- Precisão (*Precision*):
 - $Precision = \frac{TP}{TP+FP}$
- Sensitividade (*Recall*):
 - $Recall = \frac{TP}{TP+FN}$
- Índice-F1 (*F1-score*):
 - $F1 - score = \frac{2 \times TP}{2 \times TP + FP + FN}$
- Suporte (*Support*):
 - $Support = TP + FN$

Classes	TP	TN	FP	FN	Acurácia	Precisão	Sensitividade	Índice-F1	Suporte
C1	5	17	2	3	0.8148	0.7143	0.6250	0.6667	8
C2	3	16	5	3	0.7037	0.3750	0.5000	0.4286	6
C3	11	13	1	2	0.8889	0.9167	0.8462	0.8800	13
MÉDIA					0.8025	0.6687	0.6571	0.6584	
DESV. PAD					0.0761	0.2235	0.1431	0.1844	

Acurácia Vs. Precisão

- Acurácia (Accuracy):

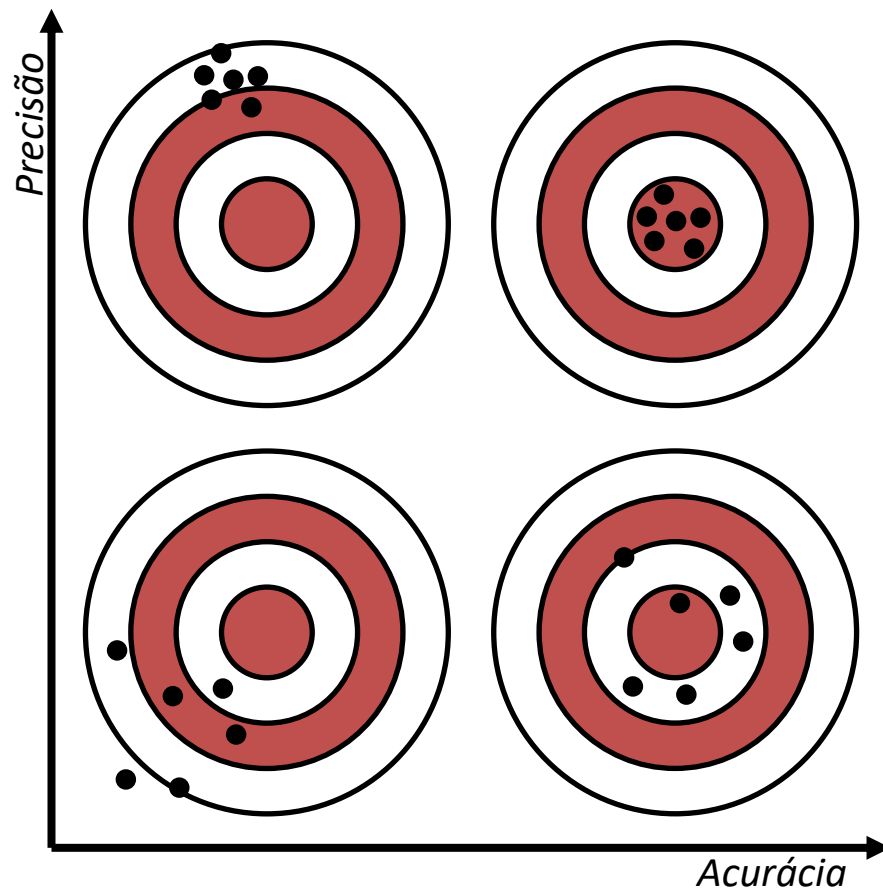
- $$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

- O quão próximo as predições estão do valor real.*

- Precisão (Precision):

- $$Precision = \frac{TP}{TP+FP}$$

- O grau de variação entre diferentes predições.*



- GONZALEZ, R.C.; WOODS, R.E.; **Processamento Digital de Imagens**. 3ª edição. Editora Pearson, 2009.
- COSTA, L. DA F.; CESAR-JR., R. M. **Shape analysis and classification: theory and practice**. CRC Press, 2000. Capítulo 8.
- Yann LeCun', Alfredo Canziani. **Yann LeCun's Deep Learning Course at CDS - SPRING 2021**
 - <https://cds.nyu.edu/deep-learning/>
- **scikit-learn – User Guide**.
 - https://scikit-learn.org/stable/user_guide.html

FIM