

# **RECONNAISSANCE DE FORMES**

# Reconnaissance de formes

*(Pattern recognition)*

Principe : identifier des formes dans des images.

Le mot « forme » est à prendre au sens général et correspond à :

- des motifs parfaitement connus
- des formes géométriques
- des lettres, des chiffres
- des visages
- ...

# Applications

- Lecture de plaque d'immatriculation
- Reconnaissance d'écriture manuscrite
- Indexation d'images
- Détection de visage, de sourires, de panneaux...
- ...

# Reconnaissance de formes par classification de caractéristiques

# Classification de caractéristiques

- Reconnaître une forme, c'est prendre une décision sur la base de caractéristiques (ou descripteurs) identifiés.  
Exemples de caractéristiques : luminosité, couleur, taille, forme, orientation du gradient, ...

# Classification de caractéristiques

- Reconnaître une forme, c'est prendre une décision sur la base de caractéristiques (ou descripteurs) identifiés.  
Exemples de caractéristiques : luminosité, couleur, taille, forme, orientation du gradient, ...
- Les caractéristiques sont des nombres, que l'on peut regrouper dans un vecteur de taille  $N$  et représenter dans un espace de dimension  $N$ .

# Classification de caractéristiques

- Reconnaître une forme, c'est prendre une décision sur la base de caractéristiques (ou descripteurs) identifiés.  
Exemples de caractéristiques : luminosité, couleur, taille, forme, orientation du gradient, ...
- Les caractéristiques sont des nombres, que l'on peut regrouper dans un vecteur de taille  $N$  et représenter dans un espace de dimension  $N$ .
- La décision est effectuée en effectuant une classification des caractéristiques dans cet espace.

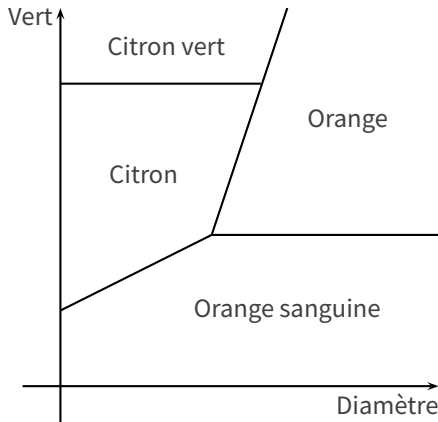
## Classification de caractéristiques





# Classification de caractéristiques

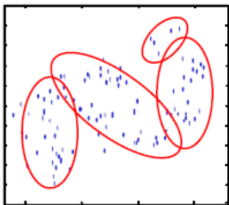
Deux caractéristiques : diamètre et vert ( $N=2$ )



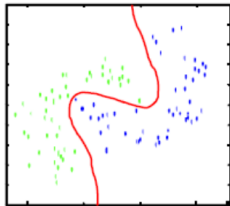
# Mise en place du système de reconnaissance

- 1 Prétraitements éventuels (segmentation, filtrage...)
- 2 Définition de zones dans l'image (découpage en imagerie, fenêtre glissante, détection...)
- 3 Détermination des caractéristiques (*features*) de ces zones
- 4 Classification de ces caractéristiques à l'aide d'un classifieur (*classifier*) avec apprentissage (*learning*) préalable le cas échéant
- 5 Évaluation des performances avec une base de test

## Apprentissage supervisé ou non ?



**Apprentissage non supervisé :**  
faire des groupes à partir de  
données non étiquetées (clus-  
tering)

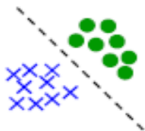


**Apprentissage supervisé :**  
trouver les fonctions séparant  
les classes (classification)

# Choix des caractéristiques

Un « bon » vecteur de caractéristiques est représentatif et discriminant :

- des échantillons issus de la même classe ont des caractéristiques similaires;
- des échantillons de classes différentes ont des caractéristiques différentes.



« bons » vecteurs



« mauvais » vecteur

# Choix des caractéristiques

Les performances des classifieurs se dégradent quand la dimension de l'espace des paramètres devient grand : un grand nombre de caractéristiques ne permet donc pas forcément d'obtenir une bonne classification.

C'est la *malédiction de la dimensionnalité* (*curse of dimensionality / Hughes phenomenon*).

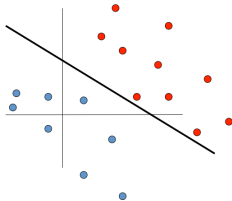
⇒ utilisation d'un nombre restreint de caractéristiques pertinentes.

# Choix du classifieur

- Distance minimale
- k-moyennes (*k-means*)
- Machine à vecteurs de support (*support vector machine*)
- Forêt d'arbres décisionnels (*random forest*)
- Réseaux de neurones (*neural network*)
- ...

## Distance minimale

Les classes sont séparées par un hyperplan.



La frontière de décision entre deux classes  $i$  et  $j$  correspond à la perpendiculaire au segment reliant les moyennes de chaque classe, calculée lors de la phase d'apprentissage.

Les caractéristiques sont affectées à la classe dont la moyenne est la plus proche.

# Distance minimale

## Avantage

- Méthode extrêmement simple

## Inconvénient

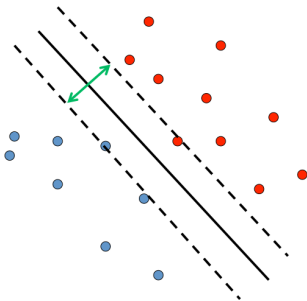
- Non applicable dans le cas où les données ne sont pas linéairement séparables



# Machine à vecteurs de support

(*SVM : support vector machine*)

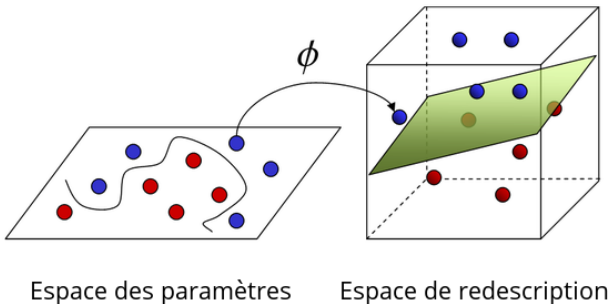
Les SVM sont un type de classifieur linéaire qui cherche l'hyperplan maximisant la marge entre les classes → optimisation.



Les éléments les plus proches de l'hyperplan sont appelés *vecteurs supports*.

# Machine à vecteurs de support

Si les données ne sont pas linéairement séparables  $\Rightarrow$  transformation de l'espace des paramètres en un espace de plus grande dimension pour trouver une séparation linéaire dans ce nouvel espace (méthodes à noyaux).



# Machine à vecteurs de support

## Avantages

- Nombreuses implantations disponibles
- Méthodes à noyaux très puissantes et flexibles
- Coût calculatoire réduit
- Peut être très performant, même sur un nombre restreint d'échantillons

## Inconvénients

- Extension à plus de deux classes difficile (demande d'utiliser plusieurs SVM à deux classes)
- Difficulté de trouver un noyau optimal dans le cas non linéaire
- La phase d'apprentissage est coûteuse en temps de calcul

# Réseaux de neurones

*(neural network)*

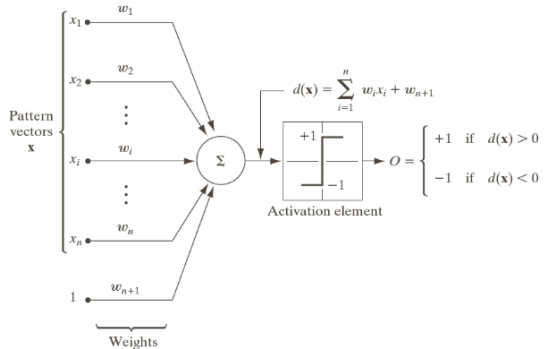
Les neurones sont des éléments de calcul non linéaires en réseau.

# Réseaux de neurones

(neural network)

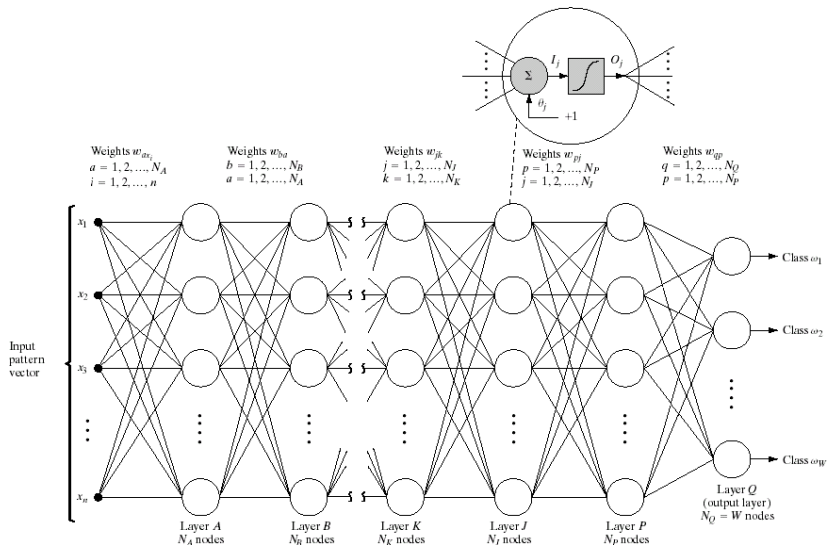
Les neurones sont des éléments de calcul non linéaires en réseau.

Perceptron :  
réseau de neurones à  
deux classes.



[Gonzalez & Woods]

# Réseaux de neurones multicouche



# Réseaux de neurones multicouche

- Le nombre de neurones dans la première couche est (généralement) égal à la dimension du vecteur de caractéristiques.
- Le nombre de neurones dans la dernière couche est égal au nombre de classes.
- La classe du vecteur correspond au numéro de la sortie ayant la valeur la plus grande.
- Chaque neurone est un perceptron, à la différence que sa fonction d'activation ne donne pas une valeur binaire (très souvent ReLU :  $\max(0, x)$ ).

# Réseaux de neurones convolutifs

(*convolutional neural network*)

Les méthodes de *deep learning* ont explosé ces dernières années du fait de plusieurs facteurs :

- apparition des réseaux de neurones convolutifs (cas particuliers des réseaux de neurones multicouche) qui permettent de traiter de grandes images ;
- augmentation des puissances de calcul disponibles ;
- existence de très grandes bases de données permettant l'apprentissage et le test (par exemple, ImageNet regroupe 14 millions d'images et 22 000 classes).



## Conclusion

# Conclusion

Mise en place d'un système de reconnaissance de forme :

- prétraitement éventuel (segmentation, filtrage...)
- choix et calcul des caractéristiques pertinentes
- choix du classifieur
- entraînement & optimisation du classificateur (estimation de ses paramètres)
- évaluation des performances du classifieur