

# Cours 3

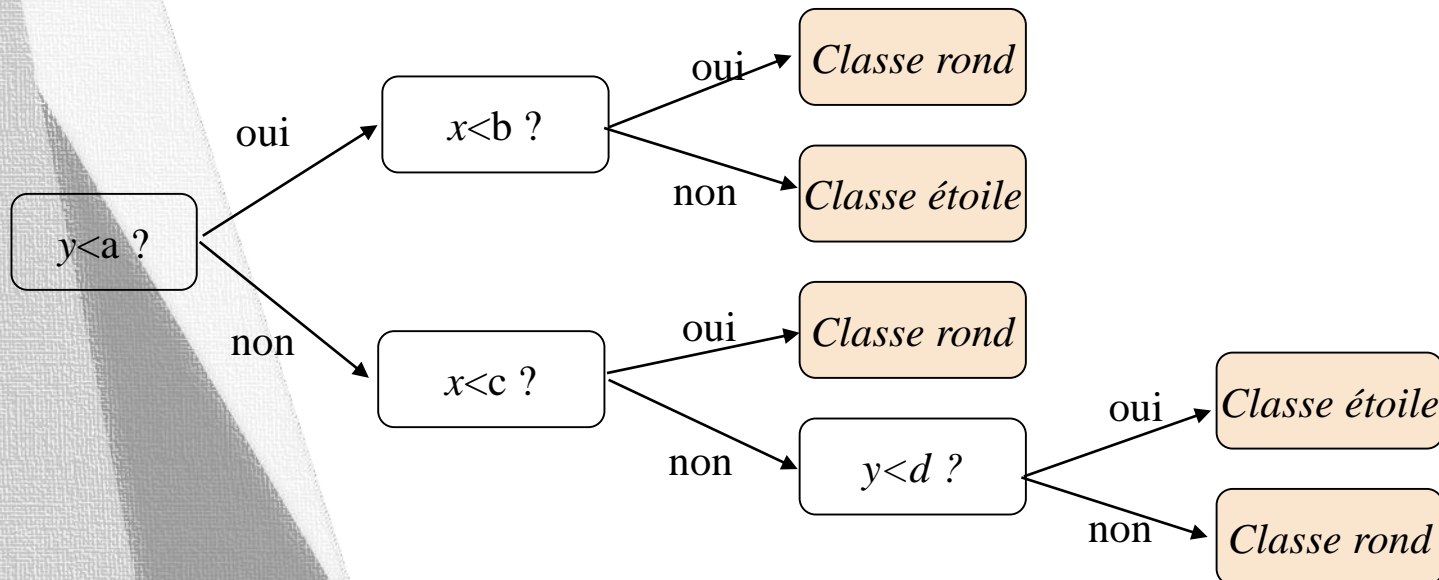
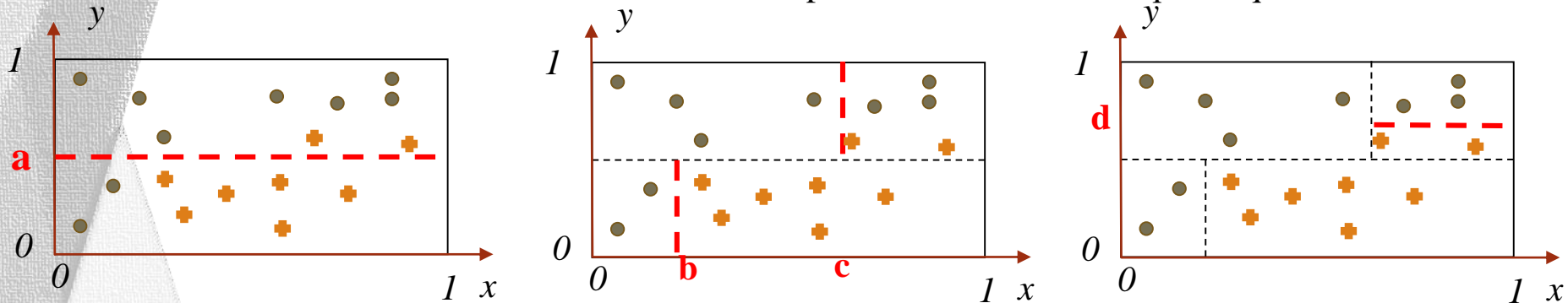
Catherine ACHARD  
Institut des Systèmes Intelligents et de Robotique

catherine.achard@upmc.fr

# Méthode discriminative arbre de décision

## Idée

- Classer avec un ensemble de règles.
- Une suite de décisions permet de partitionner l'espace en régions homogènes
- La difficulté consiste à créer l'arbre à partir de la base d'exemple étiquetée



## But

Trouver l'ordre le plus cohérent des questions qui amènera le plus rapidement à la solution

## Initialisation

tous les exemples sont dans le même nœud  $no$

## Procédure `construit_arbre(no)`

- Si  $no$  est une feuille
  - Affecter une classe à  $no$
- Sinon
  - Choisir la meilleure question et partitionner  $no$  en  $no_1$  et  $no_2$
  - `Construit_arbre( $no_1$ )`
  - `Construit_arbre( $no_2$ )`
- Fin si

Problème 1 : comment le savoir

Problème 2 : quelle classe ?

Problème 3 : comment choisir la meilleure question ?

Toutes les difficultés résident dans la réponse à ces trois problèmes

L'algorithme est très général. Plusieurs solutions proposées en fonction de la réponse à ces problèmes

## Problème 1: comment décider qu'un nœud est une feuille

- Quand tous les exemples du nœud appartiennent à la même classe
- Quand tous les exemples du nœud ont le même vecteur de paramètres
- Quand le nombre d'exemples du nœud est inférieur à un seuil
- Quand une classe est largement majoritaire dans le nœud
- En contrôlant les performances en généralisation sur une base de validation

## Problème 2: Quelle classe attribuer à une feuille

On affecte au nœud la classe majoritaire de ses exemples

## Problème 3: Comment choisir la meilleure question

Plusieurs méthodes. Exemple : on utilise la **théorie de l'information**.

Entropie sur  $X$  conditionnée par  $q$  (quantité d'information qu'il reste sur  $X$  quand on connaît la réponse à la question  $q$ )

$$H(X/q) = - \sum_{u,v} p(X = u, q = v) \log_2(X = u/q = v)$$

$H(X/q)$  : quantité d'information contenue dans  $X$  si on connaît  $q$ .

On va rechercher la question  $q$  qui **minimise** cette quantité d'information restante (on voudrait que  $q$  nous amène toutes les connaissances)



## Problème

- Les arbres peuvent ne pas être équilibrés → temps de parcours dépendant des exemples
- Présence de branches longues, peu représentatives qui nuisent à la généralisation générées par des exemples atypiques. Il faut alors les élaguer ou mettre des critères sur la profondeur, la taille des feuilles,... ou utiliser une base de validation
- Les arbres sont instables → très forte dépendance aux données → variance élevée

## Solution:

→ Construction d'une forêt d'arbre aléatoire grâce au bagging

## Bagging

### But

- Réduire la variance des prédictions

### Principe

- Construire aléatoirement plusieurs sous-ensembles d'apprentissage par tirage avec remise. Chaque sous-ensemble est appelé bootstrap
- Apprendre un arbre sur chaque sous-ensemble
- Fusionner les résultats des classifieurs

## Random forest : arbre décisionnel + bagging

- Construire aléatoirement plusieurs sous-ensembles d'apprentissage
- Construire un arbre sur chaque sous-ensemble : si les données sont de dimension  $n$ , à chaque nœud, tirer aléatoirement  $n' < n$  dimensions pour construire l'arbre. Ceci amène à des arbres moins corrélés
- Fusionner les résultats des arbres par vote majoritaire

Déterminer l'arbre de décision avec les exemples ci-dessous:

	Ciel	Température	Humidité	Vent	Décision
	q1	q2	q3	q4	
ex1	soleil	Chaud	Normale	oui	randonnée
ex2	Nuage	Froid	Haute	non	randonnée
ex3	soleil	Froid	Normale	oui	randonnée
ex4	soleil	Chaud	Normale	non	randonnée
ex5	Nuage	Froid	Normale	non	Pas randonnée
ex6	Nuage	Froid	Haute	oui	Pas randonnée
ex7	soleil	Chaud	Haute	non	Pas randonnée
ex8	soleil	Froid	Haute	oui	Pas randonnée



Il faut estimer l'entropie conditionnée par chaque question

$$H(X/q1) = - \sum_{u,v} p(X = u, q1 = v) \log_2(p(X = u/q1 = v))$$

F

Ciel	soleil	nuage
rand	3	1
Pas rand	2	2
	5	3

	Ciel	Température	Humidité	Vent	Décision
	q1	q2	q3	q4	
ex1	soleil	Chaud	Normale	oui	randonnée
ex2	Nuage	Froid	Haute	non	randonnée
ex3	soleil	Froid	Normale	oui	randonnée
ex4	soleil	Chaud	Normale	non	randonnée
ex5	Nuage	Froid	Normale	non	Pas randonnée
ex6	Nuage	Froid	Haute	oui	Pas randonnée
ex7	soleil	Chaud	Haute	non	Pas randonnée
ex8	soleil	Froid	Haute	oui	Pas randonnée

$$H(X/q1) = \begin{aligned} & -p(\text{rand}, q1 = \text{soleil}) \log_2(p(\text{rand}/q1 = \text{soleil})) && -3/8 \log_2(3/5) \\ & -p(\text{pasrand}, q1 = \text{soleil}) \log_2(p(\text{pasrand}/q1 = \text{soleil})) && -2/8 \log_2(2/5) \\ & -p(\text{rand}, q1 = \text{nuage}) \log_2(p(\text{rand}/q1 = \text{nuage})) && -1/8 \log_2(1/3) \\ & -p(\text{pasrand}, q1 = \text{nuage}) \log_2(p(\text{pasrand}/q1 = \text{nuage})) && -2/8 \log_2(2/3) \end{aligned} =$$

En faisant ainsi pour chaque question, on trouve :

$$H(X / q_1) = 0.95$$

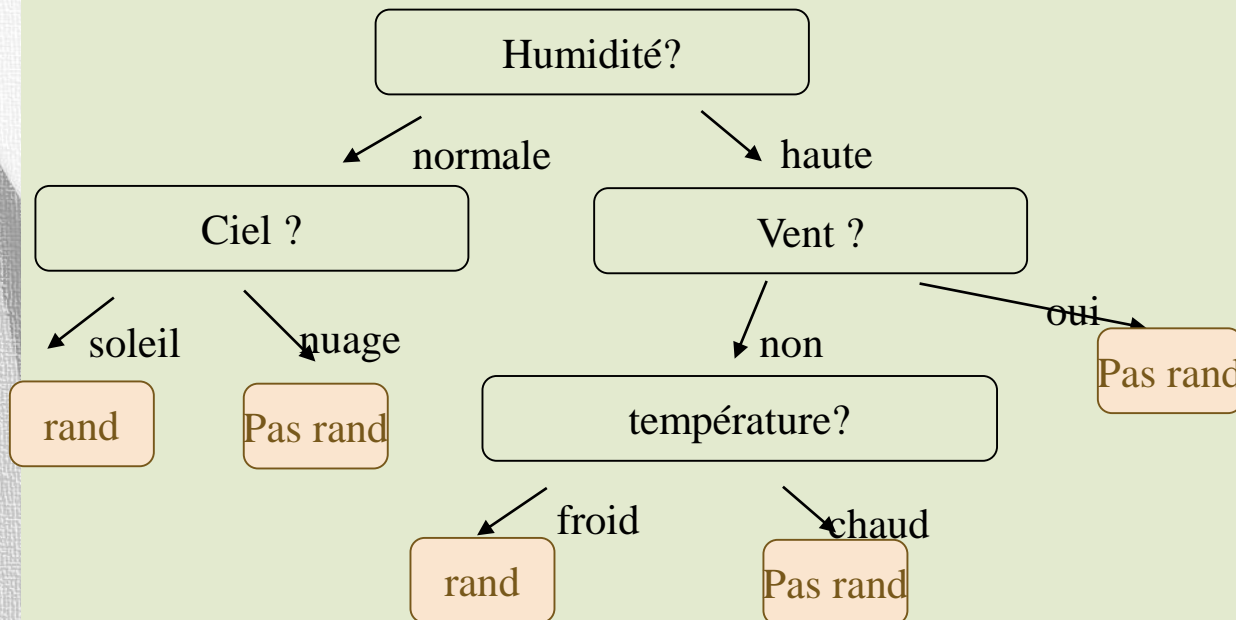
$$H(X / q_2) = 0.93$$

$$H(X / q_3) = 0.80$$

$$H(X / q_4) = 1$$

On choisit donc la question  $q_3$  qui minimise l'entropie: Humidité?

En répétant le processus,



# Méthode générative classification bayésienne

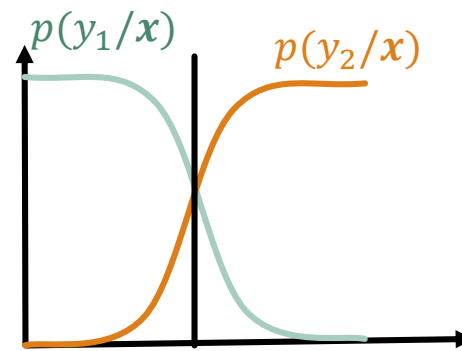
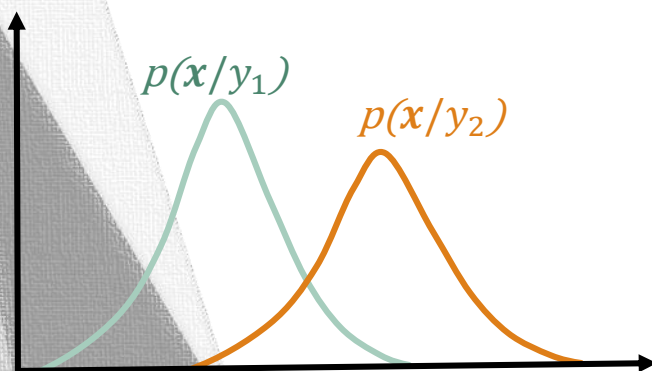
# Méthode générative : classification bayésienne

On dispose d'un exemple  $\mathbf{x}$  que l'on souhaite classer avec une étiquette  $y$ ,  
On calcule les densité de probabilité *a posteriori* de chaque classe:

$$p(y_k|\mathbf{x}) = \frac{p(\mathbf{x}|y_k)P(y_k)}{p(\mathbf{x})}$$

- $P(y_k)$  : probabilité *a priori* (probabilité de chaque classe avant d'observer  $\mathbf{x}$ )
- $p(\mathbf{x}|y_k)$  : vraisemblance des observations
- $p(\mathbf{x})$  : constante de normalisation:  $p(\mathbf{x}) = \sum_k p(\mathbf{x}|y_k) p(y_k)$
- $p(y_k|\mathbf{x})$  : probabilité *a posteriori*

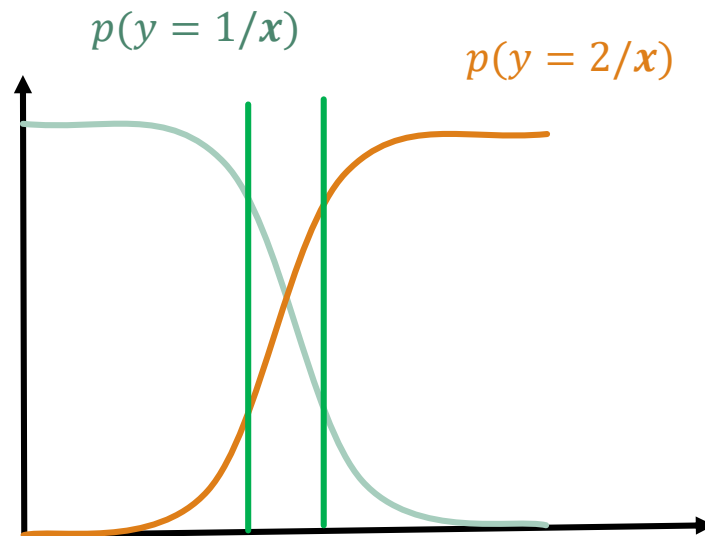
Pour classer, on choisit la classe avec la plus grande probabilité *a posteriori*  $p(y_k|\mathbf{x})$ . **Règle du Maximum A Posteriori (MAP)**



## Décision avec rejet

Comme  $p(y|\mathbf{x})$  est connu, nous pouvons rejeter :

- les exemples tq la valeur maximale de  $p(y|\mathbf{x}) < \text{seuil}$
- les exemples qui ont leur deux plus grandes probabilités a posteriori similaires





**Classification**  $p(y_k | \mathbf{x}) = \frac{p(\mathbf{x} | y_k) P(y_k)}{p(\mathbf{x})}$

Une fois la règle de décision choisie, il reste à estimer

- les vraisemblances des observations  $p(\mathbf{x} | y)$
- les probabilités a priori  $P(y)$

## Estimation des densités a priori

- En l'absence d'informations particulières, les supposer égales :

$$P(y_k) = \frac{1}{K} \quad \text{où } K \text{ est le nombre de classes}$$

- Si l'échantillon d'apprentissage est représentatif, utiliser un estimateur fréquentiel :

$$P(y_k) = \frac{N_k}{N} \quad \text{où } N_k \text{ est le nombre d'exemples de la classe } k$$

et  $N$  le nombre d'exemple total

## Estimation de la vraisemblance des observations

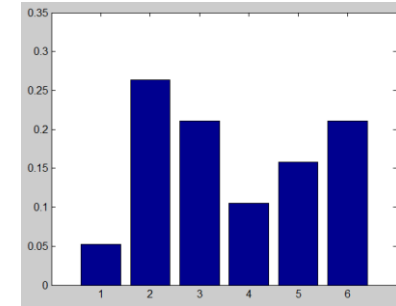
Voir méthode d'estimation de densité de probabilité

# Estimation des densités de probabilité

## Quelques rappels

**Probabilités discrètes** : A est un événement

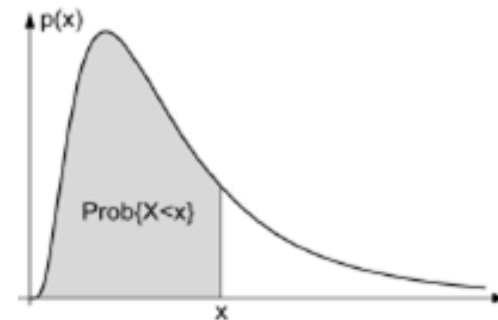
- $0 < P(A) < 1$
- $P(A) + P(B) + P(C) + \dots + P(Z) = 1$
- $P(A,B) = P(A/B) * P(B)$



**Probabilités continues** (densité de probabilité) :

- Elles ne sont pas majorées par 1 (mais l'aire vaut 1)

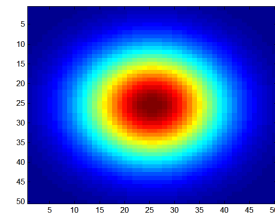
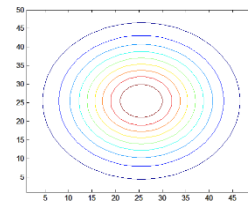
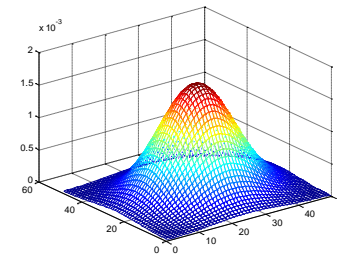
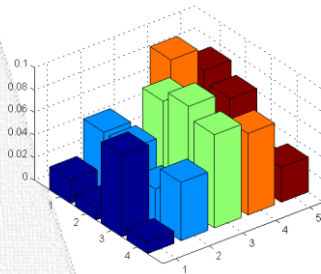
$$\int p(x) dx = 1$$



## Quelques rappels

### Probabilités jointes

Densité de probabilité jointe de  $x$  et  $y$  :  $p(x,y)$



### Indépendance

Si  $x$  et  $y$  sont indépendantes alors

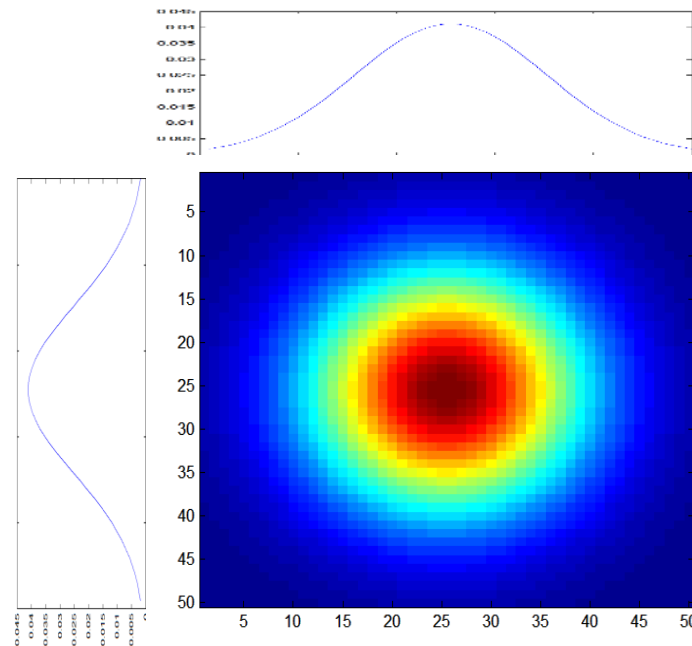
$$p(x/y) = p(x)$$

$$p(x,y) = p(x) * p(y)$$

## Quelques rappels

### Marginalisation

$$p(x) = \int p(x, y) dy \quad \text{ou} \quad p(x) = \sum_y p(x, y)$$





## Quelques rappels

### Probabilités Conditionnelles

$p(x/y_k)$  : probabilité de  $x$  sachant que  $y$  vaut  $y_k$

Cette probabilité conditionnelle peut être estimée à partir des probabilités jointes :

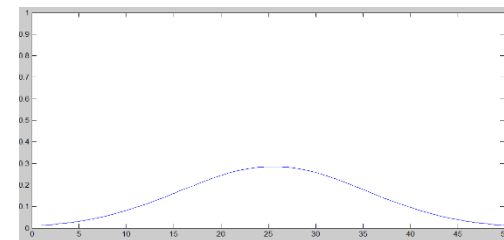
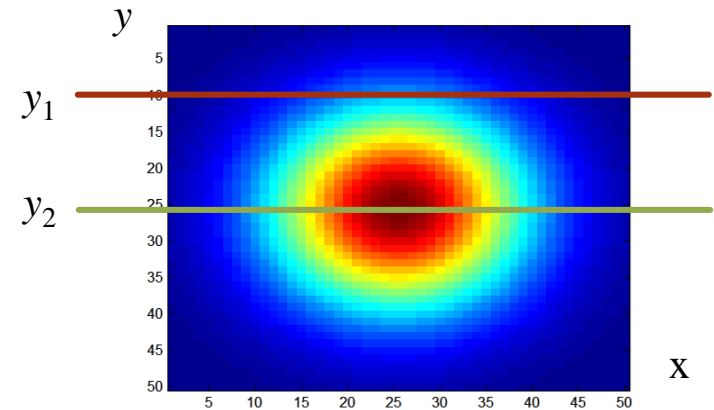
$$p(x/y_k) = \frac{p(x, y_k)}{\int p(x, y_k) dx} = \frac{p(x, y_k)}{p(y_k)}$$

Souvent, on ne spécifie pas la valeur de  $y_k$  et :

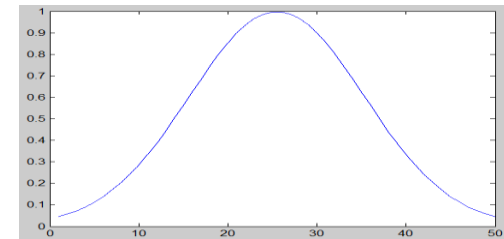
$$p(x/y) = \frac{p(x, y)}{p(y)} \text{ et on retrouve la règle de Bayes}$$

Ceci peut être étendu avec plus de variables :

$$\begin{aligned} p(w, x, y, z) &= p(w, x, y/z) p(z) \\ &= p(w, x/y, z) p(y/z) p(z) \\ &= p(w/x, y, z) p(x/y, z) p(y, z) \end{aligned}$$



$P(x/y_1)$



$P(x/y_2)$

Connaissant un ensemble de  $N$  échantillons  $\{\mathbf{x}_i\}_{i=1,\dots,N}$  de dimension  $n$  générés selon la loi de probabilité  $p(\mathbf{x})$ , comment estimer la densité de probabilité  $p(\mathbf{x})$  ?

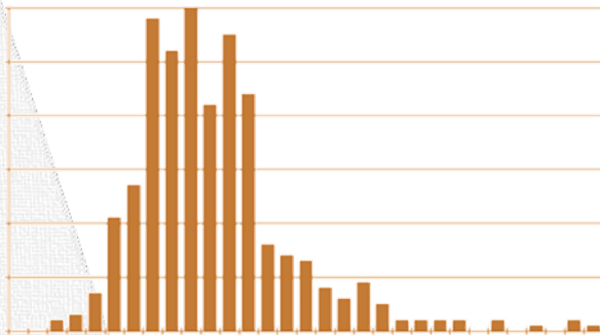
Il existe deux grands types d'approches :

- les méthodes non paramétriques
- les méthodes paramétriques (la loi est fixée *a priori* et on en recherche les paramètres)

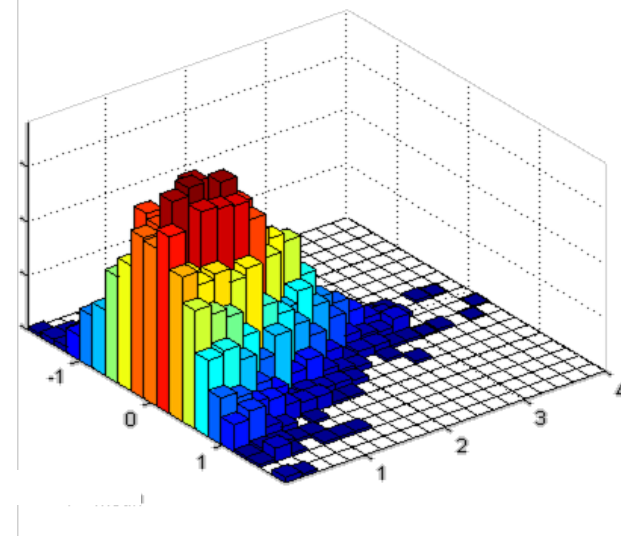
## Histogramme (non paramétrique)

- On divise chaque dimension en cases (bins) de **largeur  $h$**
- On compte le nombre d'échantillons  $x_i$  par case (divisé par N, le nombre d'échantillons)

En 1D



En 2D



Plusieurs problèmes :

- **Problème de l'origine**
- **Problème du choix de  $h$  (discrétisation)**
- **Problème des grandes dimensions**

Si les données sont de dimension 20 et que chaque dimension puisse prendre 5 valeurs, l'histogramme aura en tout  $5^{20}=9.10^{13}$  cases

→ Il faudra une grosse base de donnée pour estimer  $p(\mathbf{x})$

Exemple lié au choix de  $h$

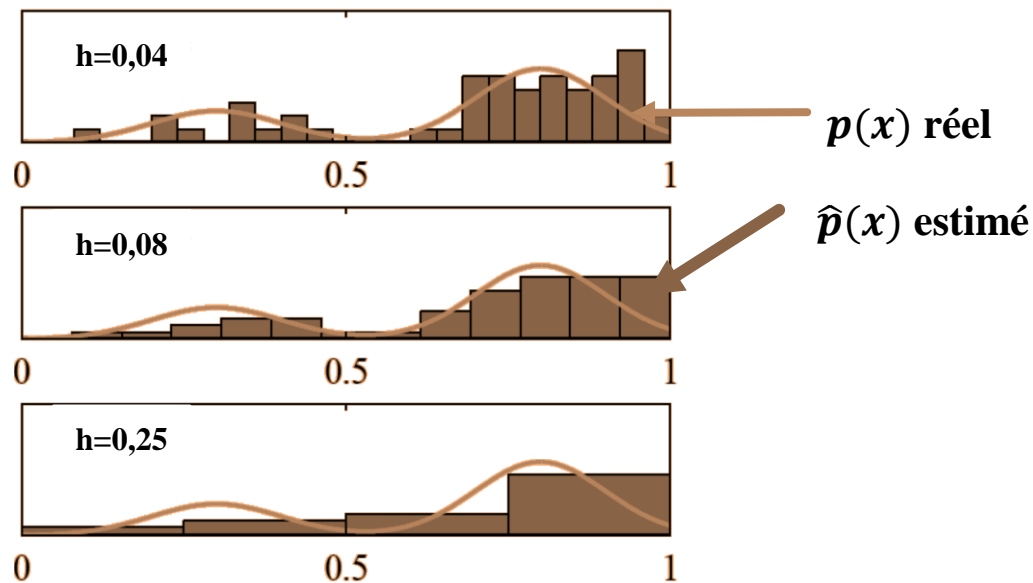


Image issue de Pattern Recognition and machine learning – M. Bishop - 2007



## Estimation par noyau (Kernel density estimation)

Pour remédier au problème de l'origine,

$$\hat{p}(x) = \frac{1}{N} \frac{\text{nombre d'échantillons dans } [x - h, x + h]}{2h}$$

Ceci s'exprime mathématiquement par :

$$\hat{p}(x) = \frac{1}{N} \sum_{i=1}^N \frac{1}{2h} K\left(\frac{x-x_i}{h}\right) \quad \text{avec } K(x) = \begin{cases} 1 & \text{si } |x| < 1 \\ 0 & \text{sinon} \end{cases}$$

Nombre d'échantillons dans un hyper-cube de largeur  $h$  centré en  $x$

**Cette estimation est continue, elle est faite pour tout  $x$**

$K(x)$  peut être un noyau quelconque (généralement une gaussienne).  $K()$  est appelé **fenêtre de Parzen**

Exemple avec un noyau gaussien en 2D :

$$\hat{p}(x) = \frac{1}{N} \sum_{i=1}^N \frac{1}{(2\pi)^{1/2}h} \exp\left(-\frac{\|x - x_i\|^2}{2h^2}\right)$$

Ceci revient à placer une gaussienne autour de chaque point et à sommer leur contribution



## Estimation par noyau (Kernel density estimation)

Estimation de la densité de probabilité sur les mêmes données que pour l'histogramme

- $h$  trop petit  $\rightarrow$  estimation très bruitée
- $h$  trop grand  $\rightarrow$  estimation trop lisse

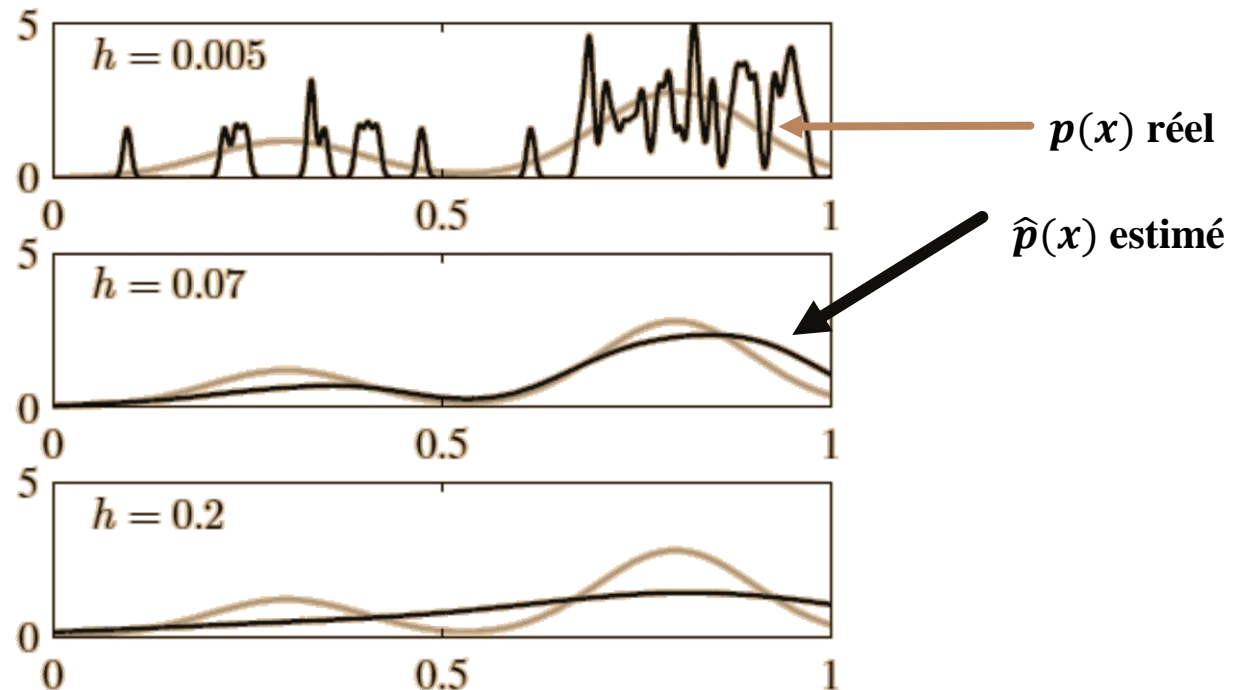


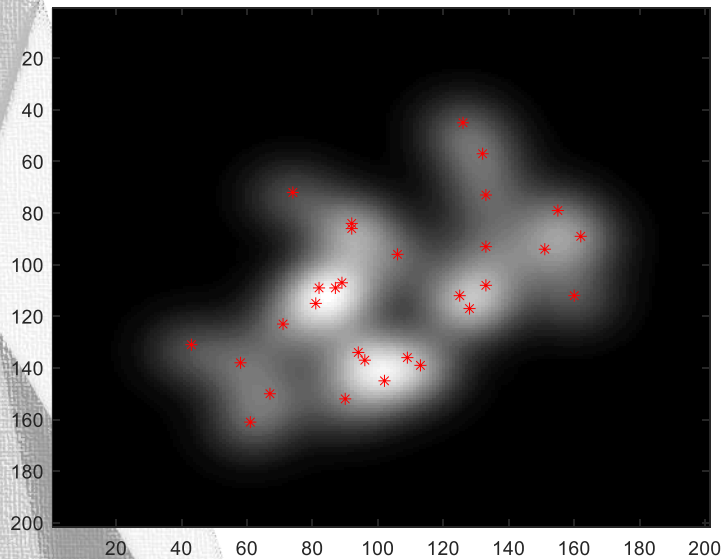
Image issue de Pattern Recognition and machine learning – M. Bishop - 2007

## Estimation par noyau (Kernel density estimation)

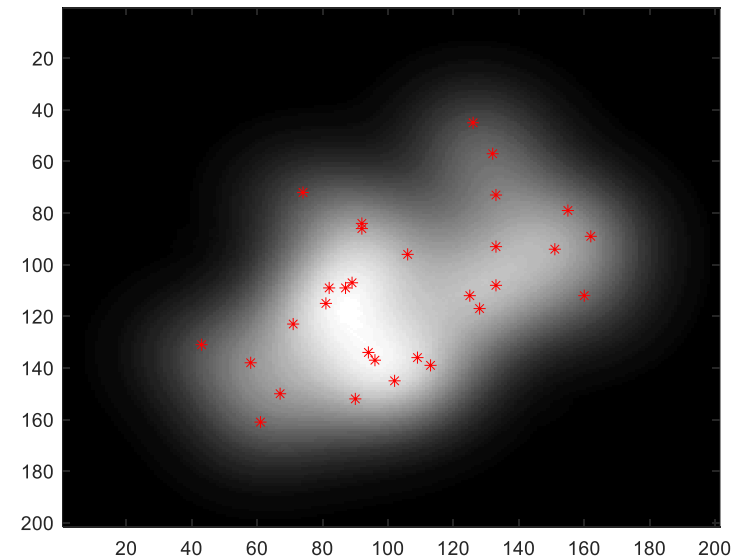
### En 2D

La densité de probabilité

Noyau gaussien,  $h=0.02$



$h=0.05$

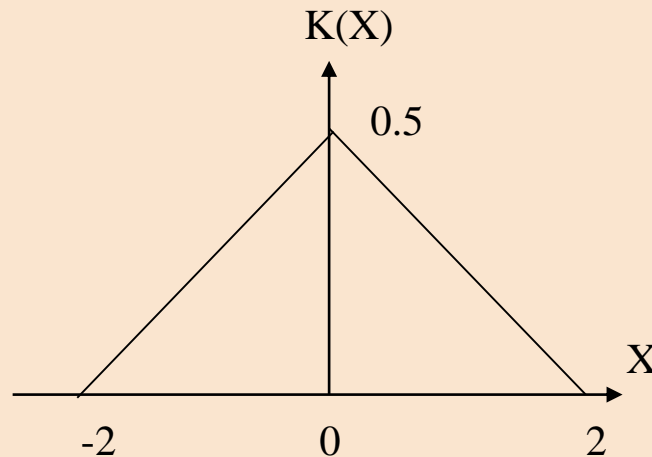


On dispose des notes de mathématiques au bac de 10 élèves pour un certain lycée.

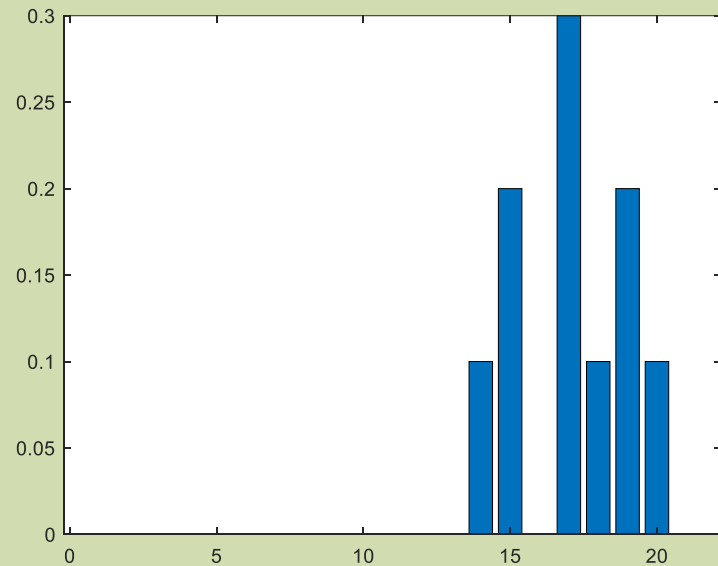
19 16 16 14 18 17 16 18 14 13

Estimer la densité de probabilité des notes avec:

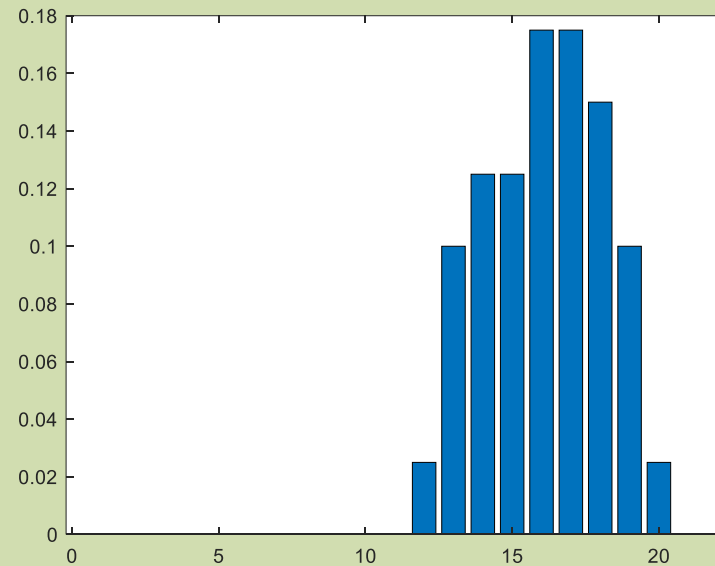
- Un histogramme 1D discrétisé avec un pas de 1
- La méthode des noyaux de Parzen avec le noyau suivant:



Avec l'histogramme



Avec les fenêtres de Parzen



On souhaite comme précédemment estimer la densité de probabilité  $p(\mathbf{x})$  à partir d'une réalisation de  $N$  échantillons,

Le problème est très difficile quand on a un faible nombre d'échantillons de dimension élevée.

La difficulté du problème est réduite si on connaît *a priori* une forme paramétrique de la loi. Dans ce cas, il n'y a plus qu'à estimer les paramètres de la loi.

Ce problème est soluble par **l'estimation du maximum de vraisemblance**.



## Estimation du maximum de vraisemblance

Nous disposons de  $N$  échantillons  $\mathbf{x}_i$  de dimension  $n$  tirés à partir de la loi  $p(\mathbf{x})$   
 $p(\mathbf{x})$  est modélisée par une loi paramétrique  $f(\mathbf{x}, \boldsymbol{\theta})$  de paramètres  $\boldsymbol{\theta}$  que l'on cherche à estimer à partir des échantillons  $\mathbf{x}_i$

Vraisemblance des observations :

$$L(\boldsymbol{\theta}/\mathbf{x}) = \prod_{i=1}^N f(\mathbf{x}_i, \boldsymbol{\theta})$$

Il est souvent plus simple de travailler avec la log-vraisemblance:

$$l(\boldsymbol{\theta}/\mathbf{x}) = \ln \left( \prod_{i=1}^N f(\mathbf{x}_i, \boldsymbol{\theta}) \right) = \sum_{i=1}^N \ln(f(\mathbf{x}_i, \boldsymbol{\theta}))$$

On recherche  $\boldsymbol{\theta}$  tq :  $\hat{\boldsymbol{\theta}} = \max_{\boldsymbol{\theta}} L(\boldsymbol{\theta}/\mathbf{x})$

Si  $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_p)^T$  est un vecteur de dimension  $p$  et que  $\nabla_{\boldsymbol{\theta}} = \left( \frac{\partial}{\partial \theta_1}, \dots, \frac{\partial}{\partial \theta_p} \right)^T$  est l'opérateur gradient, l'estimation des paramètres optimaux  $\boldsymbol{\theta}$  est telle que:

$$\nabla_{\boldsymbol{\theta}} l(\boldsymbol{\theta}/\mathbf{x}) = 0 \text{ ( } p \text{ équations pour } p \text{ inconnues)}$$

## Exercice

Retrouver les paramètres d'une gaussienne 1D avec la règle du maximum de vraisemblance

## Exercice

Retrouver les paramètres d'une gaussienne 1D avec la règle du maximum de vraisemblance

$$L(\theta/x) = \prod_{i=1}^N \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

$$l(\theta/x) = \sum_{i=1}^N -\ln(\sigma\sqrt{2\pi}) - \frac{(x-\mu)^2}{2\sigma^2}$$

$$l(\theta/x) = \sum_{i=1}^N -\ln(\sigma) - \ln(\sqrt{2\pi}) - \sum_{i=1}^N \frac{(x-\mu)^2}{2\sigma^2}$$

$$\frac{\partial l(\theta/x)}{\partial \sigma} = -\sum_{i=1}^N \frac{1}{\sigma} - \sum_{i=1}^N \frac{(x-\mu)^2}{\sigma^3} = 0$$

$$-\frac{N}{\sigma} - \sum_{i=1}^N \frac{(x-\mu)^2}{\sigma^3} = 0$$

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x - \mu)^2$$

$$\frac{\partial l(\theta/x)}{\partial \mu} = \sum_{i=1}^N \frac{2(x-\mu)}{2\sigma^2} = 0$$

$$\sum_{i=1}^N (x - \mu) = 0$$

$$\sum_{i=1}^N x - N\mu = 0$$

$$\mu = \frac{1}{N} \sum_{i=1}^N x$$

## Loi de Bernoulli (estimation paramétrique)

Si  $x$  est une variable binaire, alors

$$\text{Bern}(x = 1) = \mu \quad \text{et} \quad \text{Bern}(x = 0) = 1 - \mu$$

Ou encore

$$\text{Bern}(x) = \mu^x (1 - \mu)^{1-x}$$

On montre que :

$$\mathbb{E}[x] = \mu \quad \text{et} \quad \text{var}[x] = \mu(1 - \mu)$$

Et, avec l'estimation du maximum de vraisemblance:

$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^N x_i$$

Ces résultats peuvent être retrouvés par le calcul

## Loi binomiale (estimation paramétrique)

Supposons que l'on tire  $N$  échantillons binaires selon la loi de Bernoulli. La variable aléatoire  $x$  qui compte le nombre de réalisations de 1 parmi ces  $N$  échantillons suit une loi binomiale de paramètres  $N$  et  $\lambda$ .

$x$  peut donc prendre toutes les valeurs entières de 0 à  $N$  et

$$p(x) = \frac{N!}{(N-x)! x!} \lambda^x (1-\lambda)^{N-x}$$

On montre alors que :

$$\mathbb{E}[x] = N\lambda \quad \text{et} \quad \text{var}[x] = \sqrt{N\lambda(1-\lambda)}$$

Ces résultats peuvent être retrouvés par le calcul



## Loi uniforme (estimation paramétrique)

La variable aléatoire continue  $x$  suit une loi uniforme sur l'intervalle  $[a,b]$  si:

$$p(x) = \frac{1}{b-a}$$

On a alors

$$\mathbb{E}[x] = \frac{b-a}{2} \quad \text{et} \quad \text{var}[x] = \frac{(b-a)^2}{12}$$

## Loi normale mono variable (estimation paramétrique)

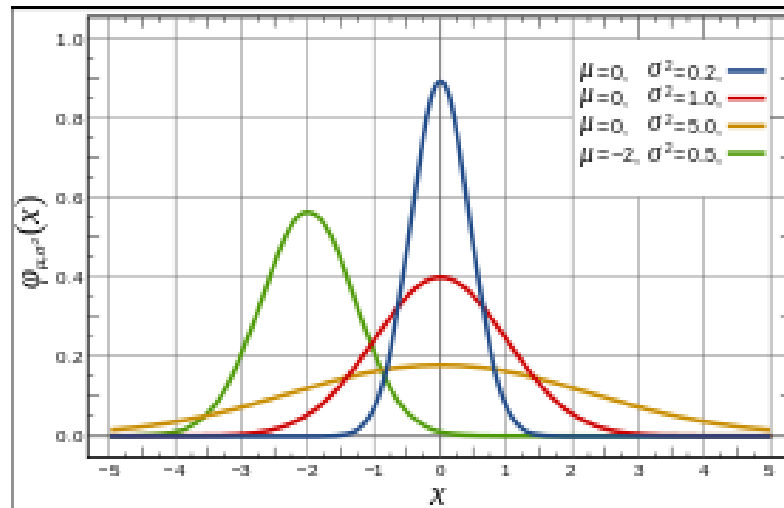
Elle est définie par :

$$p(x) = \mathcal{N}(x/\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Et :  $\mathbb{E}[x] = \mu$  et  $\text{var}[x] = \sigma^2$

L'estimation du maximum de vraisemblance conduit à:

$$\mu = \frac{1}{N} \sum_{i=1}^N x_i \quad \text{et} \quad \sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2$$



# Problème

Beaucoup de fraudes apparaissent dans les animaleries sur les lapins : alors que les gens pensent acheter un lapin nain, on leur vend un lapin normal qui devient vite très imposant. Le problème est que bébé, les deux espèces se ressemblent énormément. Aussi, deux mesures ont été réalisées à partir de prélèvements sanguins sur 20 lapins de chaque espèce. Les résultats sont présentés dans le tableau 1. Ces mesures sont reportées graphiquement figure 1.

Afin de comparer différentes méthodes de reconnaissance des formes, on dispose d'une base de test composée de 5 lapins de chaque espèce (tableau 2)

On considère que les 2 classes sont équiprobables.

1. Réaliser la classification des exemples de test avec l'algorithme des 1PPV. Donner la matrice de confusion

## Base de références

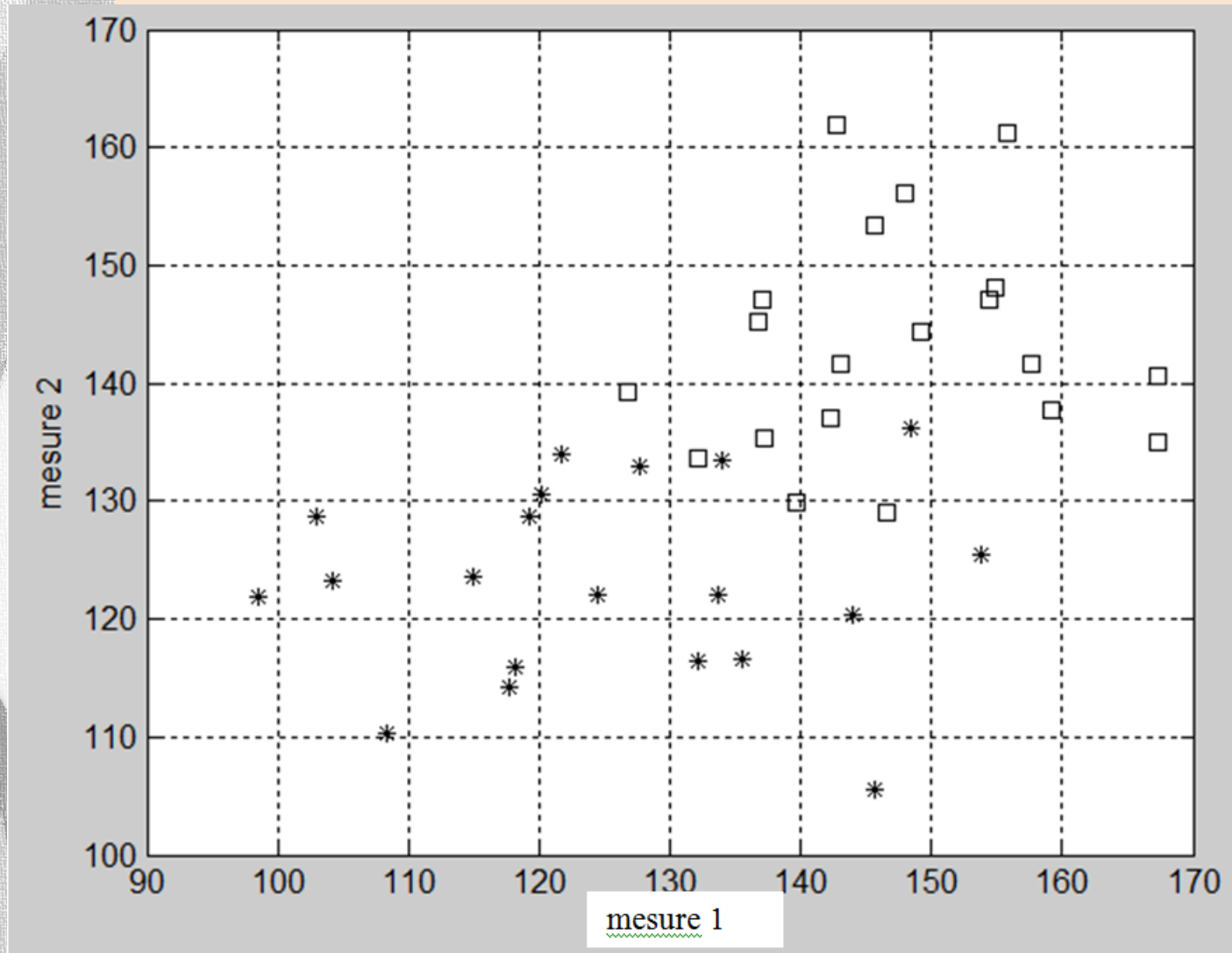
Lapins nains		Lapins normaux	
Mesure 1	Mesure 2	Mesure 1	Mesure 2
153.7	125.4	157.7	141.5
145.6	105.5	142.2	136.9
108.3	110.3	167.3	134.9
124.4	122.0	137.0	147.1
135.5	116.5	145.6	153.3
127.6	132.9	155.8	161.2
133.9	133.4	167.2	140.5
117.7	114.1	143.0	141.6
119.2	128.7	132.0	133.6
121.7	133.9	148.0	156.1
104.2	123.2	126.8	139.2
148.4	136.2	137.2	135.2
120.1	130.6	142.7	161.8
133.7	122.1	154.8	148.0
102.9	128.7	154.3	147.1
98.4	121.9	139.7	129.9
118.1	115.8	149.2	144.3
114.9	123.5	136.7	145.2
143.8	120.3	146.6	129.0
132.1	116.3	159.1	137.7
<b>Moy 125.2</b>	<b>Moy 123.1</b>	<b>Moy 147.1</b>	<b>Moy 143.2</b>
<b>Std 15.6</b>	<b>Std 8.3</b>	<b>Std 11.1</b>	<b>Std 9.5</b>

Tableau 1

## Base de test

Lapins nains		Lapins normaux	
Mesure 1	Mesure 2	Mesure 1	Mesure 2
135	115	137	152
125	125	155	152
155	125	132	142
135	132	142	135
129	139	147	136

Tableau 2





2. On souhaite classer ces exemples avec une classification bayésienne. Pour cela, on modélise la vraisemblance des observations de chaque classe par des histogrammes bi-dimensionnel. Remplir les cases qui nous sont utiles des tableaux ci-dessous.

Réaliser ensuite la classification (en cas d'égalité, l'exemple sera rejeté). Donner la matrice de confusion. En déduire le taux de reconnaissance et le taux de rejet.

Lapins nains							
m1 \ m2	100-110	110-120	120-130	130-140	140-150	150-160	160-170
90-100							
100-110							
110-120							
120-130							
130-140							
140-150							
150-160							
160-170							

Lapins normaux							
m1 \ m2	100-110	110-120	120-130	130-140	140-150	150-160	160-170
90-100							
100-110							
110-120							
120-130							
130-140							
140-150							
150-160							
160-170							

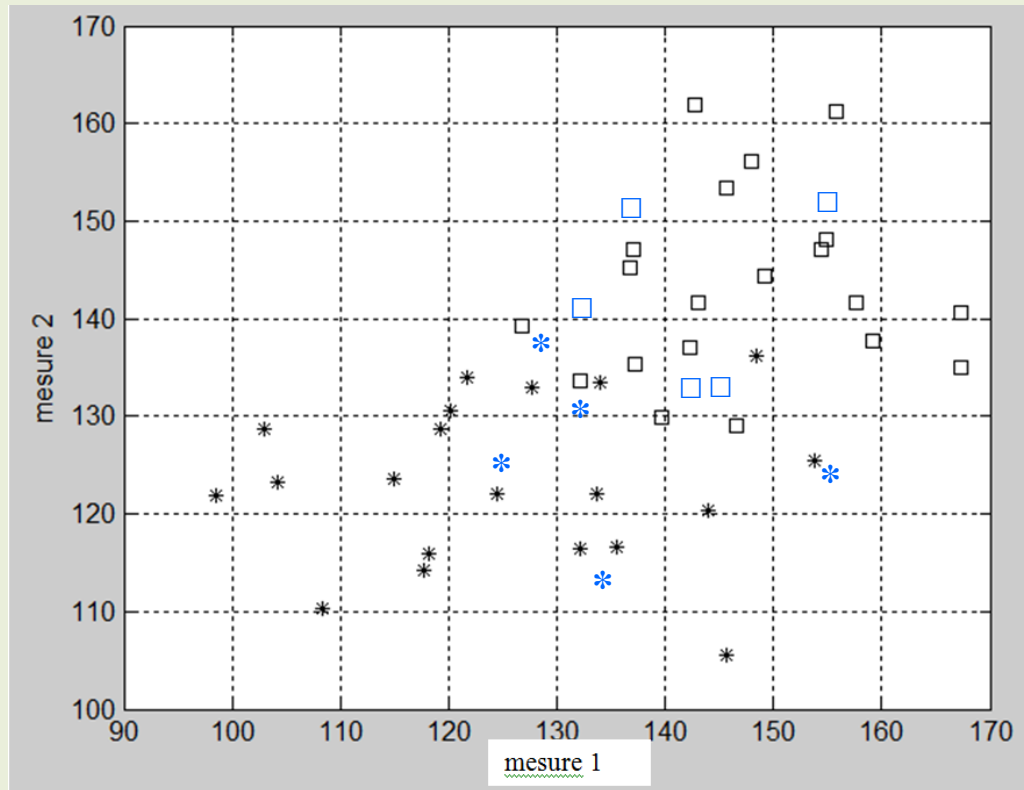
3. Reprendre les mêmes questions avec les nouveaux tableaux. Conclusion.

Lapins nains				
	90-110	110-130	130-150	150-170
90-110				
110-130				
130-150				
150-170				

Lapins normaux				
	90-110	110-130	130-150	150-170
90-110				
110-130				
130-150				
150-170				

4. On décide de modéliser chaque classe **uniquement par sa moyenne**. Réaliser la classification et donner le taux de reconnaissance
5. On décide de modéliser chaque classe par une gaussienne. On suppose pour la matrice de covariance que les deux dimensions sont décorréées. Réaliser la classification et donner le taux de reconnaissance

## Question 1



## Algorithme du 1PPV

	nain	normaux
nain	3	2
normaux	0	5

## Question 2

Lapins nains							
m1 \ m2	100-110	110-120	120-130	130-140	140-150	150-160	160-170
90-100	0	0	1	0	0	0	0
100-110	0	1	2	0	0	0	0
110-120	0	2	2	0	0	0	0
120-130	0	0	1	3	0	0	0
130-140	0	2	1	1	0	0	0
140-150	1	0	1	1	0	0	0
150-160	0	0	1	0	0	0	0
160-170	0	0	0	0	0	0	0

Lapins normaux							
m1 \ m2	100-110	110-120	120-130	130-140	140-150	150-160	160-170
90-100	0	0	0	0	0	0	0
100-110	0	0	0	0	0	0	0
110-120	0	0	0	0	0	0	0
120-130	0	0	0	1	0	0	0
130-140	0	0	1	2	2	0	0
140-150	0	0	1	1	2	2	1
150-160	0	0	0	1	3	0	1
160-170	0	0	0	1	1	0	0

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

Comme les classes sont équiprobables, il suffit de comparer les cases des histogrammes

	nain	normaux
nain	4	1
normaux	0	1

$$\text{Taux de reconnaissance} = \frac{5}{10} = 50\%$$

$$\text{Taux de rejet} = \frac{4}{10} = 40\%$$

## Question 3

Lapins nains				
	90-110	110-130	130-150	150-170
90-110	0	3	0	0
110-130	0	5	3	0
130-150	1	4	2	0
150-170	0	1	0	0

Lapins normaux				
	90-110	110-130	130-150	150-170
90-110	0	0	0	0
110-130	0	0	1	0
130-150	0	2	7	3
150-170	0	0	6	1

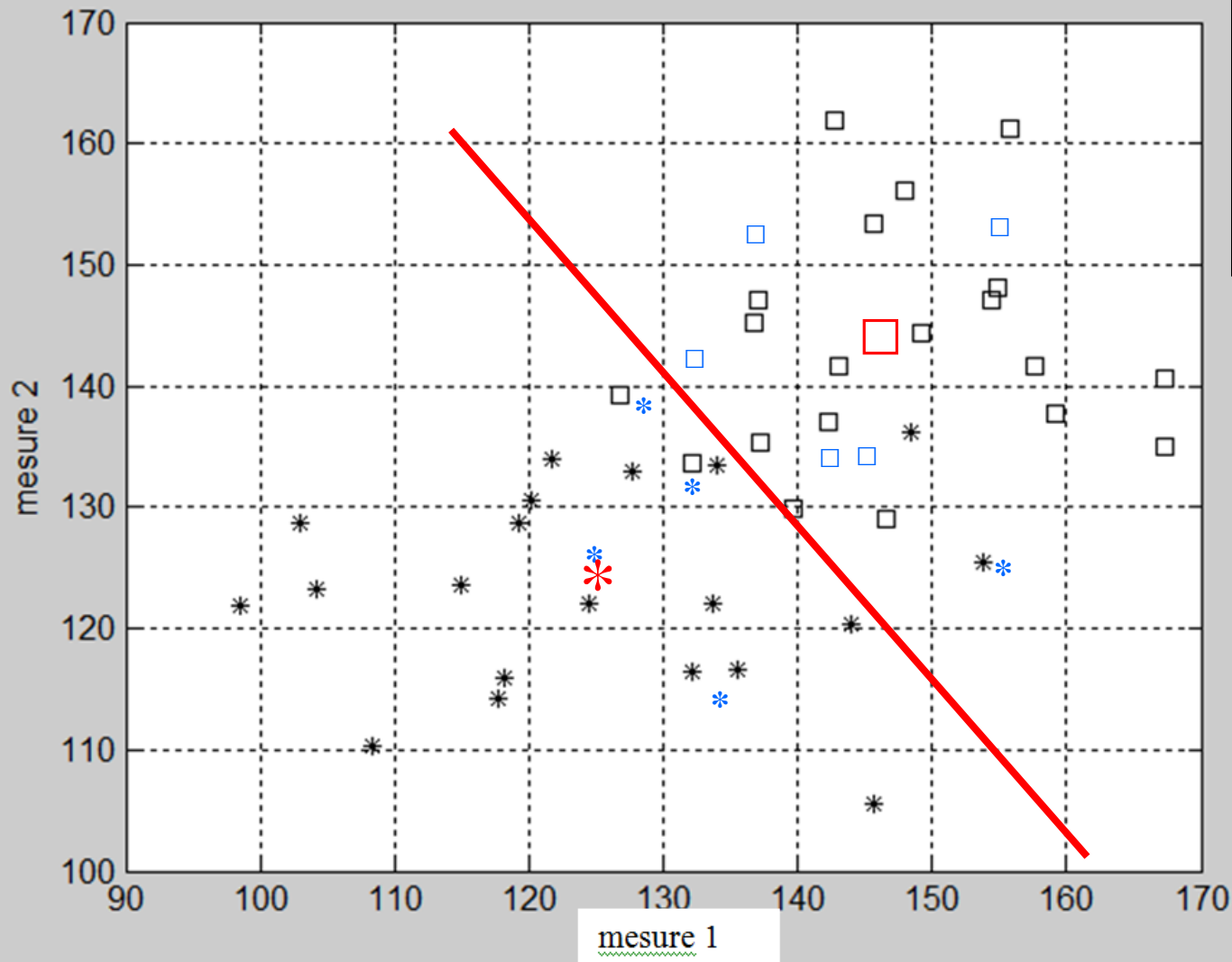
	nain	normaux
nain	4	1
normaux	0	5

$$\text{Taux de reconnaissance} = \frac{9}{10} = 90\%$$

$$\text{Taux de rejet} = \frac{0}{10} = 0\%$$



## Question 4



	nain	Nor maux
nain	4	1
Nor maux	0	5

Taux de reconnaissance

$$= \frac{9}{10} = 90\%$$

## Question 5

*Lapin normaux*

$$\text{Moy} = \begin{pmatrix} 147.1 \\ 143.2 \end{pmatrix} \quad \text{cov} = \begin{pmatrix} 122.4 & 0 \\ 0 & 89.4 \end{pmatrix}$$

$$P(x/\text{normaux}) = \frac{1}{\sqrt{2\pi}\sigma_1} \exp\left(-\frac{(x - \text{moy1})^2}{2 * \sigma_1^2}\right) * \frac{1}{\sqrt{2\pi}\sigma_2} \exp\left(-\frac{(x - \text{moy2})^2}{2 * \sigma_2^2}\right)$$

*Lapin Nains*

$$\text{Moy} = \begin{pmatrix} 125.2 \\ 123.1 \end{pmatrix} \quad \text{cov} = \begin{pmatrix} 244.7 & 0 \\ 0 & 69.7 \end{pmatrix}$$

$$P(x/\text{nains}) = \frac{1}{\sqrt{2\pi}\sigma_1} \exp\left(-\frac{(x - \text{moy1})^2}{2 * \sigma_1^2}\right) * \frac{1}{\sqrt{2\pi}\sigma_2} \exp\left(-\frac{(x - \text{moy2})^2}{2 * \sigma_2^2}\right)$$

		P(x/nain)	P(x/norm)
Nain	ex1	0.0006	0.0000098
	ex2	0.0012	0.0000322
	ex3	0.0002	0.0001854
	ex4	0.0006	0.0004127
	ex5	0.0002	0.0003591
Norm	ex1	0.0000023	0.0006
	ex2	0.0000005	0.0008
	ex3	0.0000847	0.0006
	ex4	0.0002465	0.0009
	ex5	0.0001391	0.0011

	nain	Normaux
nain	4	1
Normaux	0	5

*Taux reconnaissance*

$$T = \frac{9}{10} = 90\%$$