Apprentissage Statistique

Laurent Miclet

IRISA (Lannion)
France

EMINES 2023

1. L'apprentissage bayésien : principes

2. Approche paramétrique
   - Apprentissage au maximum de vraisemblance
   - La classification bayésienne naïve
   - L'algorithme EM

3. Approche non-paramétrique
   - Généralités
   - Les fenêtres de Parzen
   - Les k-plus proches voisins

4. Apprentissage par Estimation-Maximisation.
   - Plus généralement
   - Retour sur l'exemple
   - Apprentissage des paramètres de distributions multigaussiennes

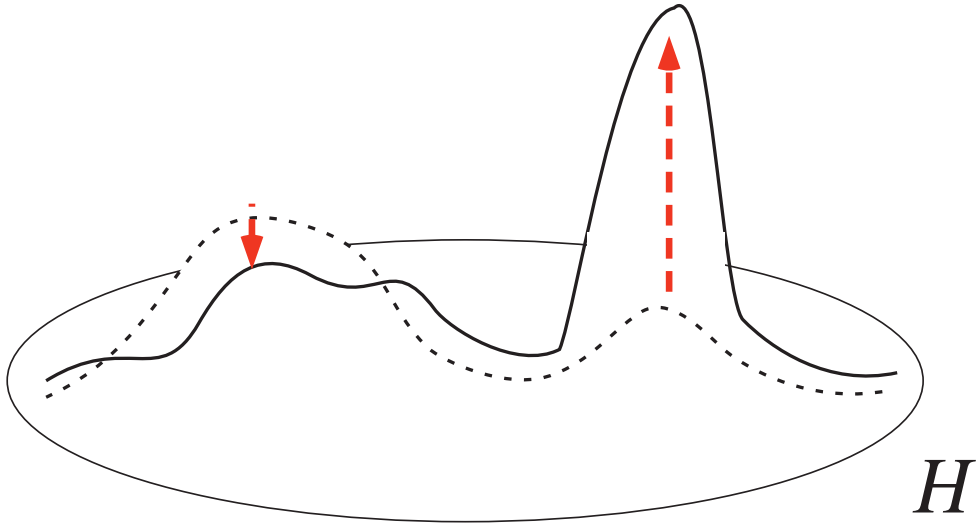
Plan

Le principe inductif bayésien.

Choisir l'hypothèse la plus probable étant donné $\mathcal{S}$

- On suppose qu'il est possible de définir une distribution de probabilité sur les hypothèses.
- La connaissance du domaine préalable à l'apprentissage s'exprime sous la forme d'une distribution de probabilité *a priori* sur les hypothèses.
- L'échantillon d'apprentissage est alors considéré comme une information modifiant la distribution de probabilité sur $\mathcal{H}$.
- On peut choisir l'hypothèse la plus probable *a posteriori* : *Maximum A Posteriori* (*MAP*).
- On peut aussi parfois élire la moyenne des hypothèses pondérée par leur probabilité *a posteriori*

Le principe inductif bayésien.



L'espace des hypothèses $\mathcal{H}$ est supposé muni d'une densité de probabilités *a priori*. L'apprentissage consiste à modifier cette densité en fonction des exemples d'apprentissage.

Exemple.

Notre bonbon surprise favori existe en deux arômes : cerise (miam) et citron (beurk).
Le fabricant enveloppe chaque bonbon dans le même papier opaque, sans se soucier de son arôme. Les bonbons sont vendus dans de très grands sacs, dont il existe cinq types, tout aussi impossibles à distinguer de l'extérieur :

h_1 : 100 % cerise ;

h_2 : 75 % cerise + 25 % citron ;

h_3 : 50 % cerise + 50 % citron ;

h_4 : 25 % cerise + 75 % citron ;

h_5 : 100 % citron.

La variable aléatoire H (pour hypothèse) qui décrit le type d'un nouveau sac prend les valeurs h_1 à h_5.

Exemple.

Quand on déballe et qu'on goûte les bonbons, les données sont dévoilées : $D_1, D_2, \cdots, D_N$, où D_i est une variable aléatoire dont les valeurs possibles sont cerise et citron. La tâche de l'agent consiste à prédire l'arôme du bonbon suivant.
Dans l'apprentissage bayésien, l'agent se contente de calculer la probabilité de chaque hypothèse, connaissant les données, et il effectue des prédictions sur cette base. Autrement dit, il prédit en utilisant toutes les hypothèses, pondérées par leur probabilité, au lieu d'utiliser simplement la seule « meilleure » hypothèse.
De cette manière, l'apprentissage se réduit à un calcul probabiliste.

Exemple

Notons D l'ensemble des données et d les valeurs observées. Les quantités impliquées dans l'approche bayésienne sont la probabilité a priori de l'hypothèse $P(h_i)$ et la vraisemblance (likelihood) des données $P(d|h_i)$ selon chaque hypothèse. La probabilité de chaque hypothèse est obtenue par la règle de Bayes :

$$P(h_i|d) = \frac{P(d|h_i)P(h_i)}{P(d)} = \alpha P(d|h_i)P(h_i)$$

Pour effectuer une prédiction sur un objet inconnu X, on a donc

$$P(X|d) = \sum_i P(X|h_i)P(h_i|d)$$

où chaque hypothèse définit une distribution de probabilités sur X.

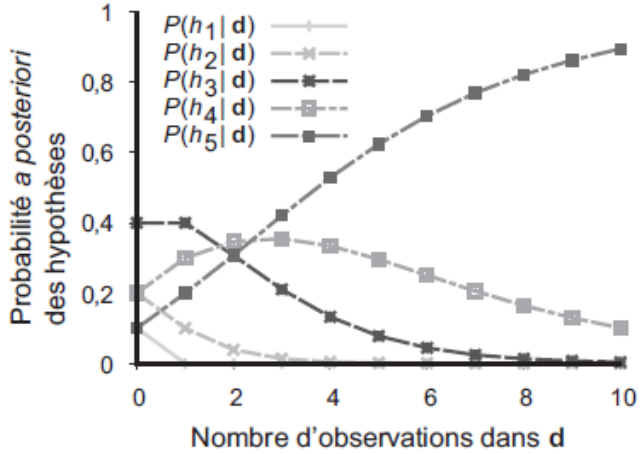
Exemple

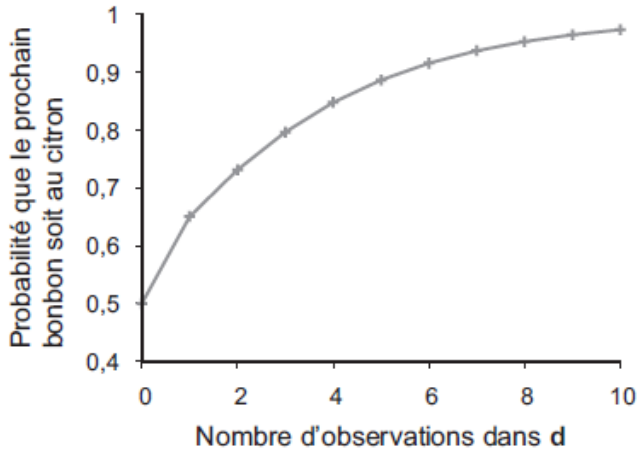
Cette équation exprime que les prédictions sont des moyennes pondérées des prédictions de chaque hypothèse individuelle, avec la pondération $P(h_i|d)$ proportionnelle à la probabilité a priori de h_i et à son adéquation aux données.
Supposons pour le moment que, dans l'exemple des bonbons, la distribution a priori sur $h_1, \cdots, h_5$ est $[0,1 \quad 0,2 \quad 0,4 \quad 0,2 \quad 0,1]$ C'est ce qu'annonce le fabricant.
La probabilité des données se calcule en supposant que les observations sont distribuées de manière indépendante et identique (i.i.d.), de sorte que

$$P(d|h_i) = \sum_j P(d_j|h_i)$$

Exemple

Imaginons que le sac ne contienne en réalité que des bonbons au citron (h_5) et que les 10 premiers bonbons extraits soient tous au citron.
Par exemple, $P(d|h_3)$ vaut $0,5^{10}$. citron.





La première figure montre comment les probabilités a posteriori des cinq hypothèses changent à mesure que la séquence des 10 bonbons est observée.
La seconde figure montre la probabilité prédite que le bonbon suivant soit au citron. Comme on pouvait s'y attendre, elle tend vers 1.

Apprentissage bayésien de classes dans $\mathbb{R}^d$
Statistical Pattern Recognition

Notations

- Espace de représentation : $\mathbb{R}^d$. Les données sont des vecteurs numériques de taille d
- Classes : $\mathcal{C} = \{\omega_i \mid i = 1, C\}$
- Ensemble d'apprentissage supervisé $\mathcal{S}$ de taille m, composé de m_i points $(\boldsymbol{x}_i\ ,\ \omega_i)$ par classe ω_i.

Problème à résoudre

Attribuer une classe parmi C à un point quelconque $\boldsymbol{x}$ de $\mathbb{R}^d$, à partir de la seule connaissance de l'ensemble d'apprentissage.

La formule de Bayes

Hypothèses probabilistes

- $p(\boldsymbol{x} \mid \omega_i)$ densité de probabilité de la classe ω_i au point $\boldsymbol{x}$,
- $P(\omega_i)$ probabilité a priori de la classe ω_i

On a :

$$p(\boldsymbol{x}) = \sum_{i=1}^{i=C} P(\omega_i) p(\boldsymbol{x} \mid \omega_i) \qquad \text{et} \qquad \sum_{i=1}^{i=C} P(\omega_i) = 1$$

Formule de Bayes

$$P(\omega_i \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid \omega_i) P(\omega_i)}{p(\boldsymbol{x})}$$

$P(\omega_i \mid \boldsymbol{x})$ est la probabilité a posteriori que $\boldsymbol{x}$ appartienne à ω_i.

Les surfaces séparatrices

On appelle surface séparatrice entre ω_i et ω_j le lieu des points où les probabilités *a posteriori* d'appartenir à ω_i et à ω_j sont égales.
La surface séparatrice entre les classes ω_i et ω_j a pour équation :

$$P(\omega_i \mid \boldsymbol{x}) = P(\omega_j \mid \boldsymbol{x})$$

$$\frac{p(\boldsymbol{x} \mid \omega_i)P(\omega_i)}{p(\boldsymbol{x})} = \frac{p(\boldsymbol{x} \mid \omega_j)P(\omega_j)}{p(\boldsymbol{x})}$$

$$p(\boldsymbol{x} \mid \omega_i)P(\omega_i) = p(\boldsymbol{x} \mid \omega_j)P(\omega_j)$$

La règle de classification bayésienne.

La règle de classification bayésienne ou règle MAP h^*

Maximum a Posteriori : la règle attribue au point $\boldsymbol{x}$ la classe ω^* qui a la plus forte probabilité *a posteriori* d'avoir engendré $\boldsymbol{x}$:

$$h^* \text{ choisit la classe } \omega^* = \underset{i}{\text{ArgMax}}(P(\omega_i \mid \boldsymbol{x}))$$

Cette règle est optimale

Parmi toutes les règles de classification possibles, elle est celle qui a la plus petite probabilité d'erreur.

$$err(h^*) = \min_h \left[\int_{\mathbb{R}^d} P^h_{err}(\boldsymbol{x}) d\boldsymbol{x} \right]$$

$err(h^*)$ est l'erreur bayésienne de classification.

▸ Démonstration

Comment approcher la règle optimale

Le problème de l'apprentissage d'une règle de classification serait donc résolu si l'on connaissait les $P(\omega_i)$ et les $p(\boldsymbol{x} \mid \omega_i)$.

$P(\omega_i)$: Les probabilités *a priori* des classes peuvent être soit supposées égales, soit estimées à partir des fréquences d'apparition dans l'ensemble d'apprentissage.

$p(\boldsymbol{x} \mid \omega_i)$: Pour chaque classe, on se trouve devant un problème d'estimation de densité de probabilité à partir d'un nombre fini d'observations.

L'estimation des probabilités *a priori* des classes

- Soit, en l'absence d'information particulière, on les suppose égales et on prend l'estimateur : $\widehat{P(\omega_i)} = \frac{1}{C}$.
- Soit on suppose l'échantillon d'apprentissage représentatif et les estimer par les fréquences d'apparition de chaque classe dans cet ensemble : $\widehat{P(\omega_i)} = \frac{m_i}{m}$.
- Soit on utilise des connaissances a priori.

Il est important de bien traiter le problème des classes déséquilibrés.

Estimation d'une densité de probabilité.

Deux techniques :

- les méthodes paramétriques : on suppose que les $p(\boldsymbol{x} \mid \omega_i)$ possèdent une certaine forme analytique.
  Si on les suppose gaussiennes, il suffit d'estimer la moyenne et la covariance de chaque distribution.
  La probabilité d'appartenance d'un point $\boldsymbol{x}$ à une classe se calcule alors directement à partir des coordonnées de $\boldsymbol{x}$.
- les méthodes non-paramétriques : on estime localement les densités $p(\boldsymbol{x} \mid \omega_i)$ au point $\boldsymbol{x}$ en observant l'ensemble d'apprentissage autour de ce point.
  Ces méthodes sont implémentées par la technique des fenêtres de Parzen (noyaux) ou l'algorithme des K-plus proches voisins.

Recherche empirique d'une règle de classification.

- On peut aussi supposer que les classes se séparent « naturellement » par des surfaces séparatrices d'un certain type (par exemple, des hyperplans) et calculer les équations de ceux-ci en utilisant les données d'apprentissage.
- Dans ce cas, c'est à partir des surfaces séparatrices que l'on définit l'ensemble H des règles de classification que l'on explore.
- On peut aussi utiliser un réseau connexionniste ou un arbre de décision pour définir une règle de classification.
- Quelque soit la méthode utilisée pour trouver une règle de classification, il faudra estimer la qualité de l'apprentissage, (au plus simple, par un ensemble de test).

Plan

Plan

Apprentissage au maximum de vraisemblance de classes supposées gaussiennes.

Notons $E[p]$ l'espérance mathématique de la variable aléatoire p.
La moyenne d'une densité de probabilité p dans $\mathbb{R}^d$ est un vecteur de dimension d dont la j^{eme} composante vaut :

$$\mu(j) = E[\boldsymbol{x}_j] = \int_{\mathbb{R}^d} \boldsymbol{x}_j p(\boldsymbol{x}_j) d\boldsymbol{x}$$

L'élément courant de sa matrice de covariance s'écrit :

$$Q(j,k) = E[(\boldsymbol{x}_j - \mu(j))(\boldsymbol{x}_k - \mu(k))]$$

Autrement dit : $m = E[\boldsymbol{x}]$ et $Q = E[(\boldsymbol{x} - \mu)(\boldsymbol{x} - \mu)^T]$

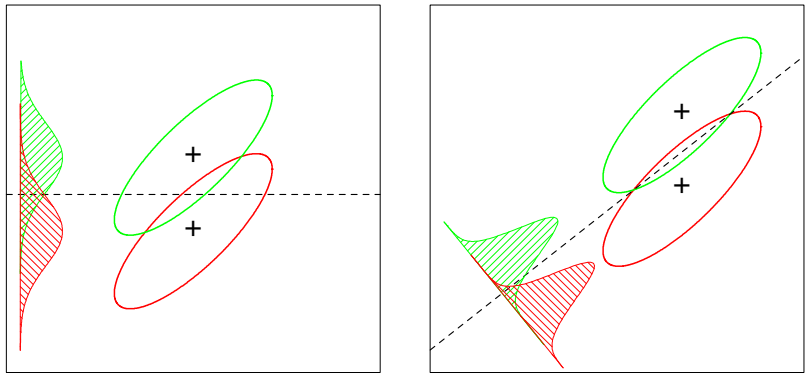


Figure 4.9: *Although the line joining the centroids defines the direction of greatest centroid spread, the projected data overlap because of the covariance (left panel). The discriminant direction minimizes this overlap for Gaussian data (right panel).*

Une distribution de probabilité gaussienne est définie par son vecteur moyenne $\boldsymbol{\mu}$ et sa matrice de covariance Q.
Quand $d = 1$, Q se ramène à un scalaire σ^2 (la variance).
Pour chaque classe :

$$p(\boldsymbol{x} \mid \omega_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{1}{2}\frac{(\boldsymbol{x} - \boldsymbol{\mu}_i)^2}{\sigma_i^2}\right)$$

Pour d quelconque :

$$p(\boldsymbol{x} \mid \omega_i) = \frac{\mid Q_i \mid^{-1/2}}{2\pi^{d/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_i)^T Q_i^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_i)\right)$$

Une estimation au maximum de vraisemblance maximise la probabilité d'observer les données d'apprentissage.
Pour la classe w_i, on possède m_i points d'apprentissage, notés $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_j ..., \boldsymbol{x}_{m_i}\}$.
Il est démontré que les estimations au maximum de vraisemblance de la moyenne $\boldsymbol{\mu}_i$ et de la matrice de covariance Q_i se calculent par :

$$\widehat{\boldsymbol{\mu}_i} = \frac{1}{m_i} \sum_{l=1}^{l=m_i} \boldsymbol{x}_l$$

$$\widehat{Q_i} = \frac{1}{m_i} \sum_{l=1}^{l=m_i} (\boldsymbol{x}_l - \widehat{\boldsymbol{\mu}_i})(\boldsymbol{x}_l - \widehat{\boldsymbol{\mu}_i})^T$$

Apprentissage bayésien de classes gaussiennes : surfaces séparatrices.

Lieu des points où les probabilités d'appartenir aux deux classes ω_i et ω_j sont égales :

$$\frac{\mid Q_i \mid^{-1/2}}{2\pi^{d/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^T Q_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)\right)$$

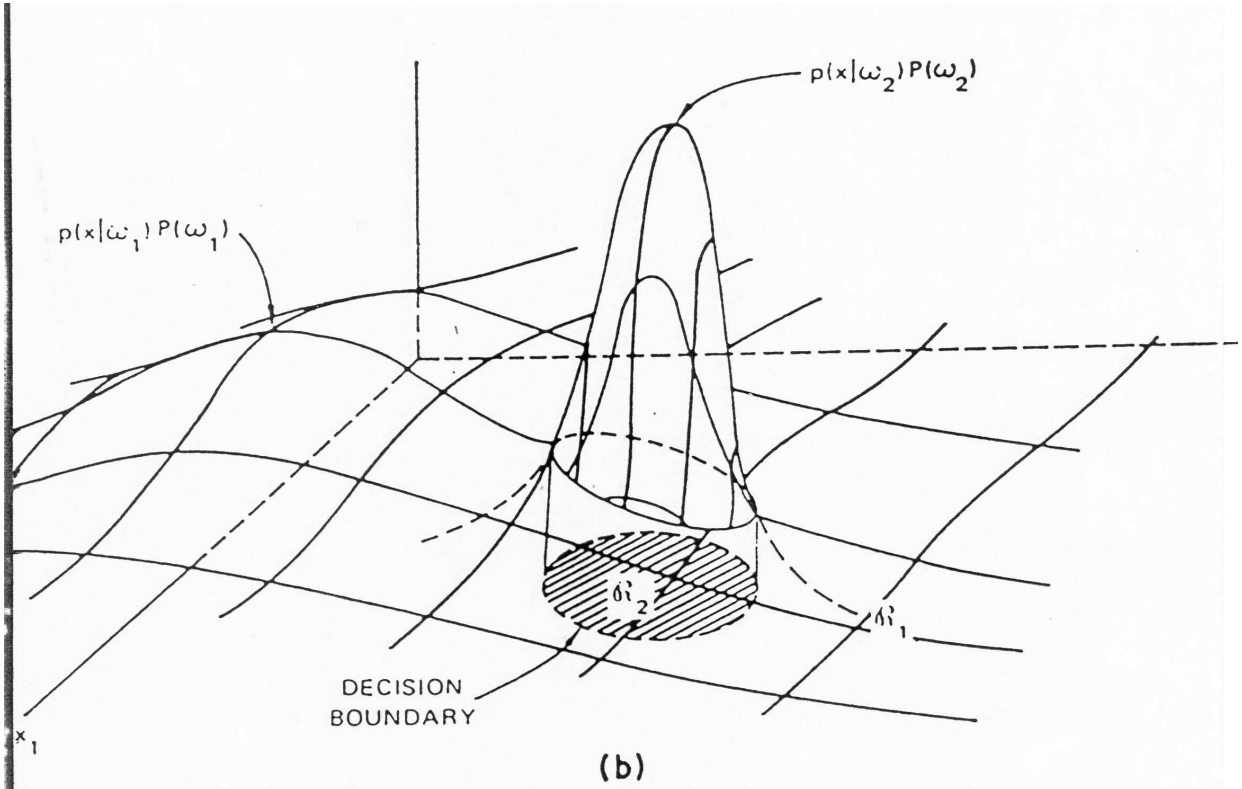
$$= \frac{\mid Q_j \mid^{-1/2}}{2\pi^{d/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_j)^T Q_j^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_j)\right)$$

Après simplification on obtient une forme quadratique :

$$\boldsymbol{x}^T \Phi \boldsymbol{x} + \boldsymbol{x}^T \phi + \alpha = 0$$

La matrice Φ, le vecteur ϕ et α ne dépendent que de $\boldsymbol{\mu}_i$, $\boldsymbol{\mu}_j$, Q_i, Q_j.

Séparatrice à deux dimensions



(b)

Examples of decision boundaries and decision regions.

Plan

Un cas simple : la classification bayésienne naïve

On suppose ici que chaque classe possède une matrice de covariance diagonale. Cette hypothèse revient à dire que les attributs sont statistiquement décorrélés.
Dans cette simplification, la probabilité d'observer $\boldsymbol{x}^T = (x_1, \ldots, x_d)$ pour un point de n'importe quelle classe ω_i est la probabilité d'observer l'attribut x_1 pour cette classe, multipliée par celle d'observer l'attribut x_2 pour cette classe, etc. Donc, par hypothèse :

$$\omega^* = \underset{i \in \{1,\ldots,C\}}{\text{ArgMax}} \; P(\omega_i) \prod_{i=1}^{d} p(x_i \mid \omega_i)$$

Chaque valeur $p(x_i \mid \omega_i)$ s'estime par comptage dans un intervalle (histogramme monodimensionnel).

Simplification des notations

La règle de classification bayésienne ou règle MAP h^* attribue au point $\boldsymbol{x}$ la classe ω^* de plus forte probabilité *a posteriori* d'avoir engendré $\boldsymbol{x}$:

$$h^* \text{ choisit la classe } \omega^* = \underset{i}{\text{ArgMax}}(P(\omega_i \mid \boldsymbol{x}))$$

Dans la suite, on omettra l'indice correspondant au numéro de classe puisque le problème est le même pour chaque classe.

Plan

Un cas plus général : la modélisation par un mélange de gaussiennes

Mélange de K gaussiennes :

$$\sum_{i=1,K} k_i \frac{\mid Q_i \mid^{-1/2}}{(2\pi)^{d/2}} exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^T Q_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)\right\} \text{ avec } \sum_{i=1,K} k_i = 1$$

On apprend pour chaque classe tous les paramètres :

- la moyenne de chaque gaussienne
- la covariance de chaque gaussienne
- les valeurs de mélange

par l'algorithme d'optimisation *EM*.

Plan

Plan

Apprentissage bayésien non paramétrique.

- Soit un point $\boldsymbol{x}$ dont on cherche la classe.
- On estime en $\boldsymbol{x}$ la densité de probabilité de chaque classe, puis on applique la règle de classification bayésienne.
- Pour chaque classe ω, on a le même problème : on possède m points d'apprentissage de $\mathbb{R}^d$ obtenus par tirages indépendants selon une densité $p(\omega \mid \boldsymbol{x})$.
- Comment estimer $p(\boldsymbol{x} \mid \omega)$ au point $\boldsymbol{x}$ à partir de cet ensemble d'apprentissage ?

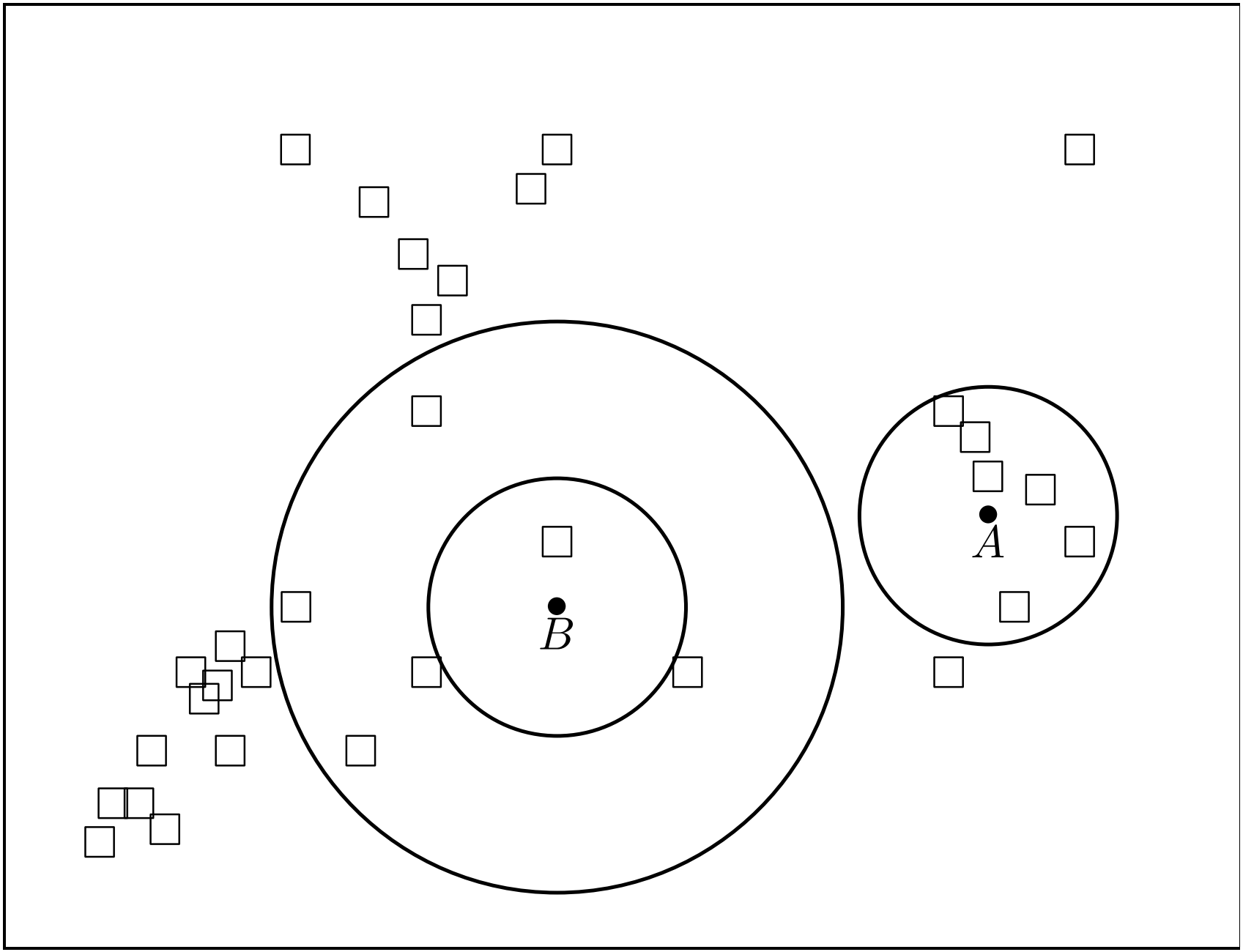
La technique

- On définit autour de $\boldsymbol{x}$ une région R_m de volume V_m et on compte le nombre k_m de points de l'échantillon d'apprentissage qui sont inclus dans cette région.
- Estimateur de $p(\boldsymbol{x} \mid \omega)$ pour un échantillon de taille m :

$$\widehat{p_m}(\boldsymbol{x} \mid \omega) = \frac{k_m/m}{V_m}$$

où V_m est le volume de la région R_m considérée.

- Quand m augmente cet estimateur converge vers $p(\boldsymbol{x} \mid \omega)$ si :
  - $\lim V_m = 0$
  - $\lim k_m = \infty$
  - $\lim(k_m/m) = 0$


A
B

Explication

- Probabilité P_m que $\boldsymbol{x}$ tombe dans la région R_m : $P_m = \int_{R_m} p(\boldsymbol{x} \mid \omega) dx$
- m échantillons sont tirés i.i.d. selon $p(\boldsymbol{x} \mid \omega)$. Probabilité que k_m d'entre eux tombent dans R_m :

$$\begin{pmatrix} m \\ k_m \end{pmatrix} P_m^{k_m} (1 - P_m)^{1-k_m}$$

- On tire de cette distribution que l'espérance de k_m vaut mP_m, donc que $\frac{k_m}{m}$ est un estimateur de P_m.
- Si V_m est assez petit pour que $p(\boldsymbol{x} \mid \omega)$ y soit constant, on a :

$$P_m = \int_{R_m} p(\boldsymbol{x} \mid \omega) dx \simeq p(\boldsymbol{x} \mid \omega)\ V_m \quad \text{donc} \quad p(\boldsymbol{x} \mid \omega) \simeq \frac{k_m/m}{V_m}$$

Apprentissage bayésien non paramétrique.

Il y a deux solutions :

- Soit calculer V_m à partir d'une région R_m de forme fixée. En général, on paramètre le calcul par m :

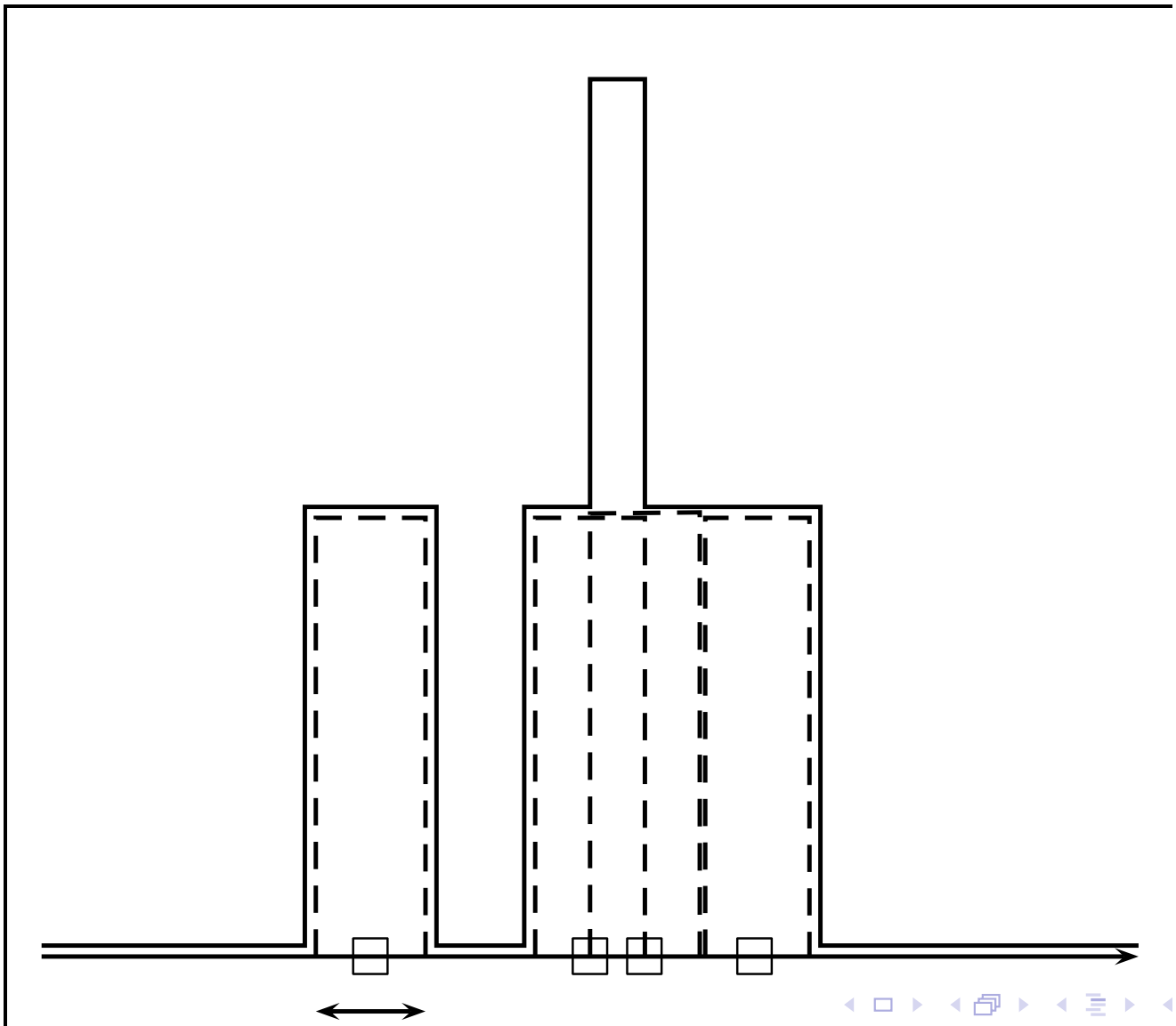
$$V_m = V_0 / f(m)$$

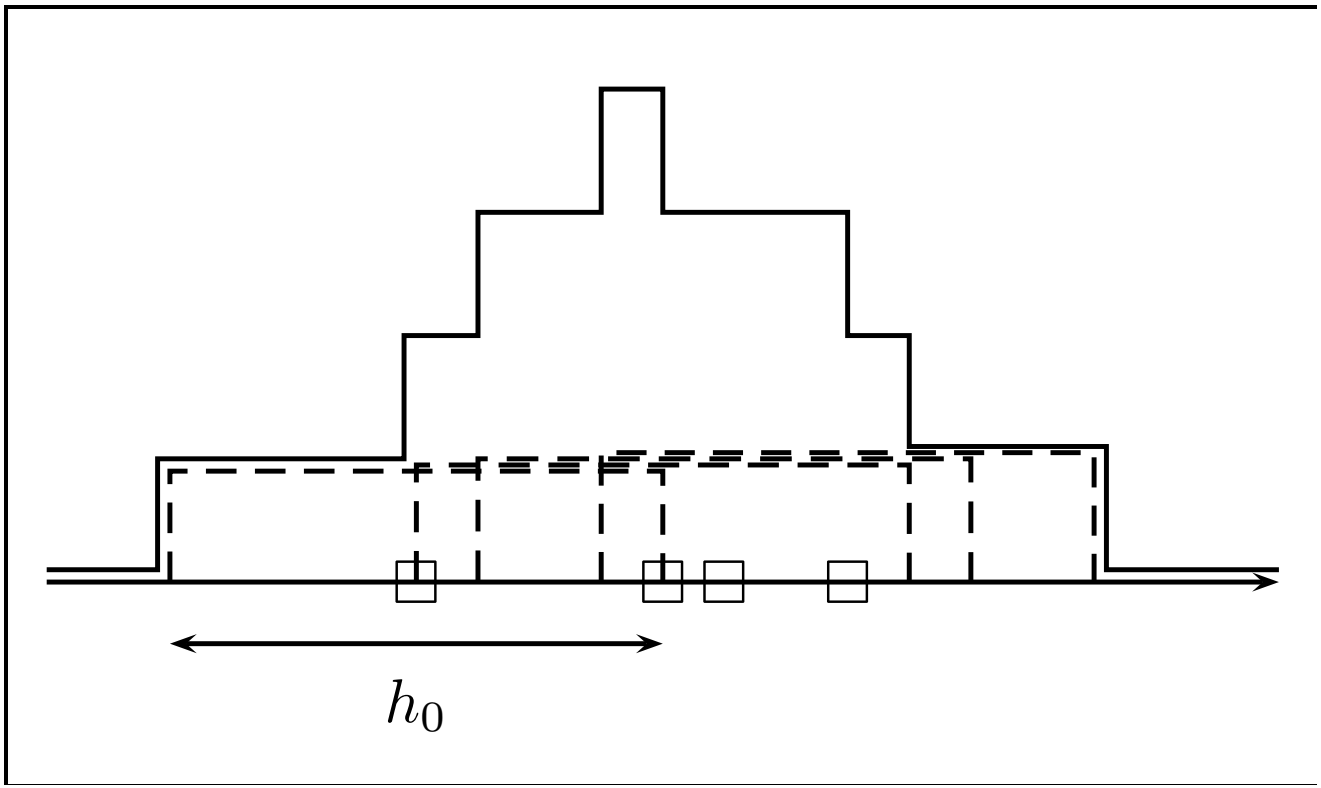
avec f une fonction croissante de m.
Cette méthode s'appelle les Fenêtres de Parzen.

- Soit fixer le volume pour qu'il contienne exactement k_m points de l'ensemble d'apprentissage.
C'est la méthode des K-plus proches voisins, ou K-ppv.

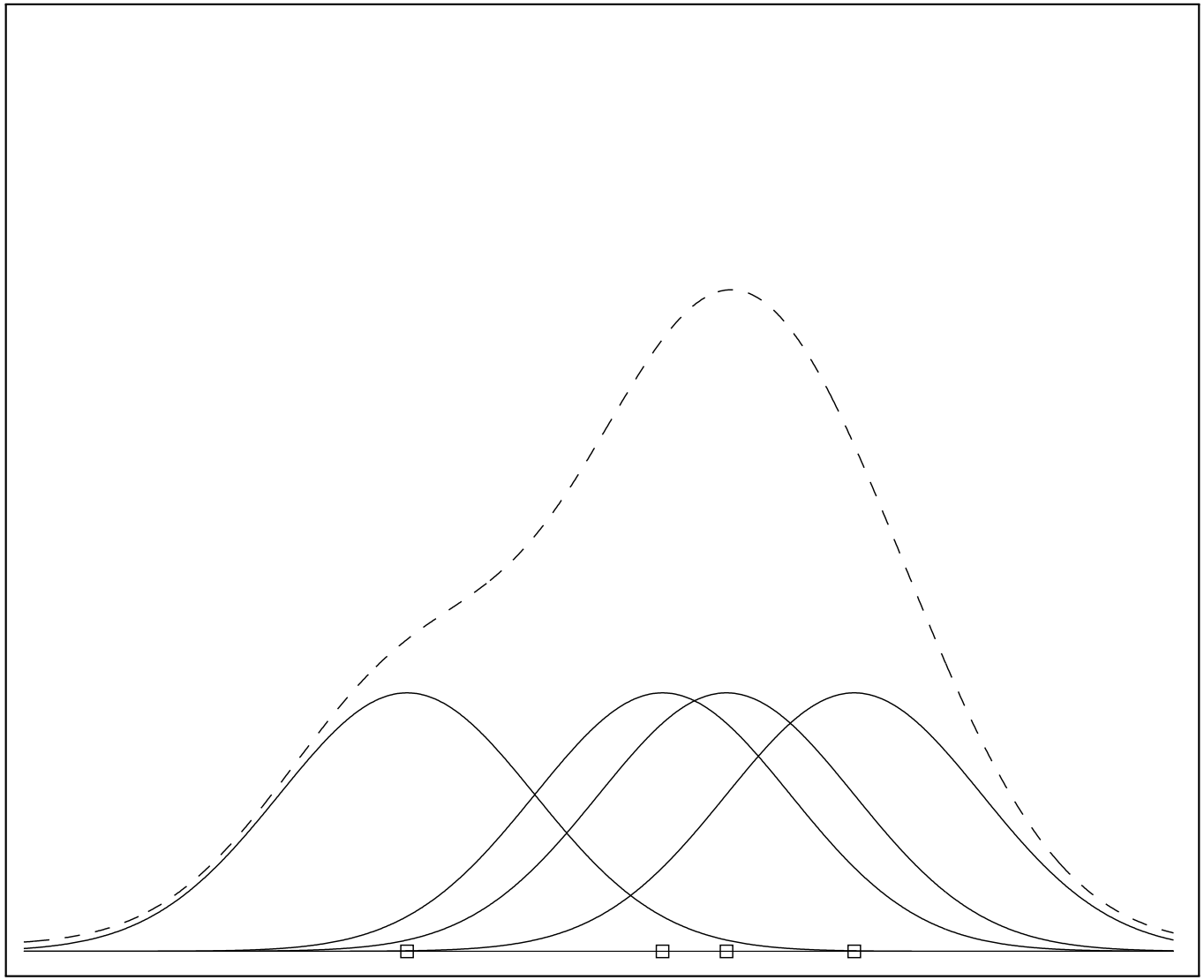
Plan

Fenêtres de Parzen (noyaux rectangles)




h_0

Fenêtres de Parzen (noyaux gaussiens)



Fenêtres de Parzen : les noyaux

La technique se décrit plus généralement par le calcul :

$$\widehat{p_m} = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{\lambda V_m} \kappa(\boldsymbol{x}, \boldsymbol{x}_i)$$

- La fonction $\kappa(\boldsymbol{x}, \boldsymbol{x}_i)$ est centrée en $\boldsymbol{x}_i$ et décroît quand $\boldsymbol{x}$ s'éloigne de $\boldsymbol{x}_i$.
- Elle a une intégrale finie : le volume V_m.
- λ est un facteur de normalisation.

Par exemple κ peut être un rectangle de largeur variable (à surface constante), ou une gaussienne de variance variable (à intégrale constante).

Fenêtres de Parzen : problèmes de calcul

$$\widehat{p_m} = \frac{1}{m}\frac{1}{\lambda V_m}\sum_{i=1}^{m}\kappa(\boldsymbol{x}, \boldsymbol{x}_i)$$

Pour éviter de calculer la somme sur m termes, on utilise une *fonction noyau* pour κ, c'est à dire telle qu'il existe dans un espace de dimension n une fonction Φ avec :

$$\kappa(\boldsymbol{x}, \boldsymbol{y}) = \langle \Phi(\boldsymbol{x}), \Phi(\boldsymbol{y}) \rangle$$

Par exemple, pour $n = 4$ et $d = 2$, avec $\kappa(\boldsymbol{x}, \boldsymbol{y}) = \langle \boldsymbol{x}, \boldsymbol{y} \rangle^3$:

$$\Phi(\boldsymbol{x}) \;=\; (x_1^3 \;,\; \sqrt{3}\; x_1^2 x_2 \;,\; \sqrt{3}\; x_1 x_2^2 \;,\; x_2^3)^\top$$

On peut vérifier que

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle^3 \;= (x_1 y_1 + x_2 y_2)^3 = \langle \Phi(\boldsymbol{x}) \,,\, \Phi(\boldsymbol{y}) \rangle$$

Fenêtres de Parzen : les fonctions noyaux

$$\begin{aligned}
\widehat{p_m} &= \frac{1}{m}\frac{1}{\lambda V_m}\sum_{i=1}^{m}\kappa(\boldsymbol{x},\boldsymbol{x}_i) \\
&= \frac{1}{m}\frac{1}{\lambda V_m}\sum_{i=1}^{m}\langle\Phi(\boldsymbol{x}),\Phi(\boldsymbol{x}_i)\rangle \\
&= \frac{1}{m}\frac{1}{\lambda V_m}\langle\Phi(\boldsymbol{x}),\sum_{i=1}^{m}\Phi(\boldsymbol{x}_i)\rangle
\end{aligned}$$

On précalcule $\sum_{i=1}^{m}\Phi(\boldsymbol{x}_i)$ et il reste à calculer un produit scalaire en dimension n.

Fenêtres de Parzen : noyau gaussien.

- L'exemple $\Phi(\boldsymbol{x}) = (x_1^3\ ,\ \sqrt{3}\ x_1^2 x_2\ ,\ x_1 x_2^2\ ,\ \sqrt{3}\ x_1 x_2^2\ ,\ x_2^3)^\top$ est un *noyau polynômial* particulier dans une famille très générale, que l'on peut définir pour tout d et tout n' (ici, $d = 2$ et $n' = 3$).
- On peut démontrer qu'une fonction gaussienne est une aussi fonction noyau, mais que la fonction Φ correspondante est de dimension infinie (en effet, $e^x = \sum_{i=1}^{\infty} \frac{1}{i!} x^i$).
- On peut l'approcher d'aussi près que l'on veut par un polynôme, d'autant mieux que n' (et n, qui en dépend) augmentent.
- On peut donc réaliser de manière approchée le calcul précédent pour une fenêtre de Parzen gaussienne.

Plan

Plan

L'algorithme *Estimation-maximisation* (EM) est une procédure générale pour apprendre la valeur de paramètres cachés de certains processus probabilistes.
Supposons que nous disposions de deux pièces de monnaie truquées A et B, avec des probabilités respectives p_A et p_B de tomber sur *Pile* et $1 - p_A$ et $1 - p_B$ de tomber sur *Face*. Nous connaissons les valeurs p_A et p_B. Nous demandons à un huissier de procéder en secret aux opérations suivantes :

- tirer un nombre μ au hasard entre 0 et 1,
- répéter N fois les manipulations suivantes :
  - choisir la pièce A avec la probabilité μ ou la pièce B avec la probabilité $(1 - \mu)$
  - lancer la pièce choisie
  - enregistrer le résultat du lancer : *Pile* ou *Face*

Une fois l'affaire terminée, l'huissier nous communique la séquence des N valeurs *Pile* ou *Face* qu'il a notées. Notre problème est alors le suivant : estimer la valeur du paramètre caché μ à partir de la suite $\mathcal{O} = (O_1, \ldots O_N)$ de ces observations. Chaque observation O_i vaut donc *Pile* ou *Face*.
Il est clair que si nous connaissions quelle pièce a été utilisée à chaque

lancer, nous pourrions estimer μ par la valeur :

$$\frac{N_A}{N} = \frac{\textit{Nombre de fois que la pièce A a été lancée}}{N}$$

Mais ce n'est bien sûr pas le cas. Nous pouvons cependant utiliser une technique itérative, *l'algorithme EM*[1].

Initialisation

Donner une valeur arbitraire μ_0 strictement comprise entre 0 et 1.

Etape p

- Estimation

  On utilise les observations $\mathcal{O}$ pour calculer une estimation de N_A.

  On note :

  - μ_p la valeur courante de μ
  - $P_A(O_i)$ la probabilité de l'évènement : "A est sortie au tirage i"
  - $E_p(A \mid \mathcal{O})$ l'estimation du nombre total de tirages de la pièce A

  On a :

$$E_p(A \mid \mathcal{O}) = \sum_{i=1}^{i=N} \frac{\mu_p P_A(O_i)}{\mu_p P_A(O_i) + (1-\mu_p) P_B(O_i)}$$

1. E pour Estimation (ou en anglais *Expectation*), M pour Maximisation.

- Maximisation
  On calcule une nouvelle estimation de μ :

$$\mu_{p+1} = \frac{E_p(\mathcal{A} \mid \mathcal{O})}{N}$$

Test d'arrêt

$$\mu_{p+1} \approx \mu_p$$

Pour résoudre la première partie, il faut savoir calculer $E_p(A \mid \mathcal{O})$ mais on ne connaît pas les valeurs $P_A(O_i)$. Cependant, on peut estimer la valeur cherchée par une *statistique suffisante*.
Pour notre exemple, on constate que l'ordre dans lesquels les O_i sont apparus dans l'ensemble des observations $\mathcal{O}$ n'a pas d'importance : le nombre P d'observations *Pile* dans $\mathcal{O}$ suffit. On dit qu'il s'agit d'une *statistique suffisante* pour estimer le paramètre caché μ. Cette observation

permet de calculer explicitement $E_p(\mathcal{A} \mid \mathcal{O})$ de la façon suivante :

$$
\begin{aligned}
E_p(A \mid \mathcal{O}) &= \sum_{i=1}^{P} \frac{\mu_p P_A(Pile)}{\mu_p P_A(Pile) + (1-\mu_p) P_B(Pile)} \\
&\quad + \sum_{i=1}^{F} \frac{\mu_p P_A(Face)}{\mu_p P_A(Face) + (1-\mu_p) P_B(Face)} \\
&= \sum_{i=1}^{P} \frac{\mu_p p_A}{\mu_p p_A + (1-\mu_p) p_B} + \sum_{i=1}^{F} \frac{\mu_p p_A}{\mu_p p_A + (1-\mu_p) p_B} \\
&= P \frac{\mu_p p_A}{\mu_p p_A + (1-\mu_p) p_B} + F \frac{\mu_p p_A}{\mu_p p_A + (1-\mu_p) p_B}
\end{aligned}
$$

Plan

On démontre que l'algorithme EM décrit en 1 fait croître la vraisemblance de
Λ_p vis-à-vis des données. Par conséquent, il converge vers un optimum local.

Algorithme 1 **Estimation-Maximisation**

Définir une statistique suffisante pour estimer l'ensemble Λ des paramètres cachés

Initialiser Λ

$p \leftarrow 1$

tant que $\Lambda_p \neq \Lambda_{p+1}$ **faire**

ESTIMATION

Utiliser les observations pour calculer la statistique suffisante de Λ_p

MAXIMISATION

Calculer Λ_{p+1} comme une estimation au maximum de vraisemblance de Λ à partir des résultats de l'étape p d'estimation

$p \leftarrow p + 1$

fin tant que

Plan

Prenons $p_A = 0.2$ et $p_B = 0.6$. L'huissier effectue $N = 100$ tirages et fournit une série $\mathcal{O}$ d'observations comportant $P = 50$ *Pile* et $F = 50$ *Face*. Quelle est la valeur de μ que produit l'algorithme EM ?
La récurrence de base est la suivante :

$$\mu_{p+1} = \frac{1}{P+F}\left(P\frac{\mu_p p_A}{\mu_p p_A + (1-\mu_p)p_B} + F\frac{\mu_p(1-p_A)}{\mu_p(1-p_A) + (1-\mu_p)(1-p_B)}\right)$$

D'où, pour deux initialisations de μ :

μ_0	0.800	0.100
μ_1	0.730	0.109
μ_2	0.659	0.118
μ_3	0.593	0.127
μ_4	0.536	0.135
μ_5	0.488	0.144
μ_{10}	0.351	0.183
μ_{20}	0.273	0.228
μ_{30}	0.256	0.243
μ_{40}	0.252	0.248
μ_{50}	0.250	0.249
μ_{60}	0.250	0.250

Il se trouve que notre exemple, on peut en réalité estimer directement μ car la proportion de *Pile* est en effet une estimation de :

$$\mu\, p_A + (1 - \mu\, p_B)$$

Dans notre application numérique :

$$1/2 = \mu\, 0.2 + (1 - \mu)\, 0.6$$

d'où :

$$\mu = 0.25$$

Il faut en effet que l'huissier ait tiré en moyenne 4 fois plus souvent la pièce B que la pièce A pour avoir rétabli l'équilibre entre le nombre de *Pile* et celui de *Face*.

Plan

On dispose d'une collection $X = \{x_1, ..., x_N\}$ d'exemples, qui sont des vecteurs de $\mathbb{R}^d$. On fait l'hypothèse que ces vecteurs sont des tirages aléatoires d'un *mélange* de k distributions gaussiennes $\mathcal{N}_1$,...$\mathcal{N}_k$. On cherche à estimer les paramètres de chaque distribution, ainsi que la façon dont elle sont mélangées.
Le tirage d'un exemple peut se décrire ainsi : d'abors, on choisit aléatoirement une des k distributions. Ensuite, on tire l'exemple selon cette distribution. Par conséquent, pour définir le mélange, il suffit de se donner les k valeurs qui sont les probabilités que l'une des k distributions soit tirée. Un exemple x_i doit donc se décrire de manière plus complète par

$$y_i = (x_i, z_{i1}, ..., z_{ik})$$

où :

- x_i est observable
- pour $j = 1, k$, z_{ij} est non observable. z_{ij} vaut 1 si x_i a été généré par $\mathcal{N}_i$

Pour simplifier, nous prendrons $d = 1$. Nous supposerons aussi que les k distributions, de moyennes μ_1, ..., μ_k, ont la même variance σ. La méthode se généralise sans trop de difficultés à d quelconque et à des matrices de covariances différentes pour chaque gaussienne.

L'algorithme *EM* s'applique maintenant comme suit. Il a pour résultat le vecteur $h = (\mu_1, ..., \mu_k)$ et les estimations des valeurs z_{i1}, z_{ik}. Ces dernières quantités sont donc les probabilités avec lesquelles on tire les gaussiennes $\mathcal{N}_j$, pour $j = 1, k$.

Algorithme 2 — **Mélange de k gaussiennes**

Initialiser aléatoirement $h = (\mu_1, ..., \mu_k)$

tant que le processus n'a pas convergé **faire**

Estimation

Calculer les estimations $E(z_{i1})$ de z_{i1}, ... $E(z_{ik})$ de z_{ik} en supposant que l'hypothèse courante sur h est la bonne :

pour j=1,k **faire**

$$E(z_{ij}) = \frac{p(x = x_i \mid \mu = \mu_j)}{\sum_{n=1}^{k} p(x = x_i \mid \mu = \mu_n)}$$

Donc :

$$E(z_{ij}) \leftarrow \frac{e^{-\frac{1}{2\sigma^2}(x_i - \mu_j)^2}}{\sum_{n=1}^{k} e^{-\frac{1}{2\sigma^2}(x_i - \mu_n)^2}}$$

fin pour

Maximisation

Calculer une nouvelle estimation de $h = (\mu_1, ..., \mu_k)$ en supposant que z_{ij} est égale à $E(z_{ij})$