STT3030 - Cours #10

Arthur Charpentier

Automne 2024

Apprentissage non-supervisé

Deux problèmes standards en apprentissage non-supervisé:

- La mise-en-grappe/le regroupement : Méthodes et techniques afin de créer des groupes de variables *similaires*
- La réduction de dimension: Méthodes et techniques pour projeter les données $\mathbf{X} \in \mathcal{X}$ sur $\mathbf{Z} \in \mathcal{Z}$ où la dimension de $\mathcal{Z}$ est **beaucoup** plus petite que celle de $\mathcal{X}$.

Mise-en-grappe

- Dans la mise-en-grappe, on veut regrouper des observations similaires.
- On forme des groupes à l'aide des variables $\mathbf{X}$.
- On doit penser à une mesure de distance ou similitude.
- On doit aussi penser à combien de groupes on formera.

Pas de mesure de succès

- Un grand défi de l'apprentissage non-supervisé est le manque d'une bonne mesure de succès.
- Comme il n'y a pas de réponse, nous ne pouvons pas utiliser l'erreur de prédiction/classification.
- Il n'y a pas de vrais groupes ni de vraie représentation de petite dimension non plus.
- Un problème est donc qu'il est difficile d'établir si nous avons raisonnablement accompli notre tâche d'apprentissage non-supervisé.
- Le succès du modèle est plus subjectif.
- L'évaluation du succès d'une technique d'apprentissage non-supervisé est un problème encore d'actualité.

Partitionnement en k-groupe

- Dans la mise-en-grappe, il faut déterminer des groupes différents d'observations similaires.
- La première approche de mise-en-grappe que nous voyons est le **k-mean clustering (KMC)**.
- Il s'agit d'une approche relativement simple conceptuellement de mise-en-grappe, qui vise à
- (1) Déterminer des centres de masse (des moyennes) pour des groupes et
- (2) Détermine ensuite le groupe auquel appartient une observation en fonction de la distance entre l'observation et les centres de masse.

Algorithme

Le partitionnement en k-moyenne est fondé sur algorithme itératif qui exécute deux opérations jusqu'à atteindre une stabilité:

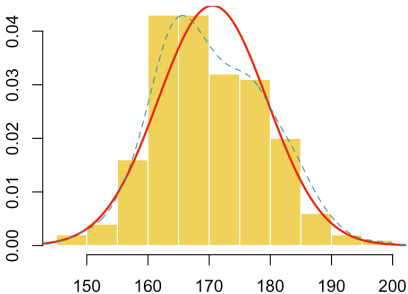
1. Étant donné l'assignement des observations à des groupes, on estime le centre du groupe.
2. Étant données des centres de groupes, on assigne les observations aux groupes.

Après avoir exécuté ces deux opérations quelques fois, les moyennes cessent de bouger et les observations cessent de changer de groupes.

Parametric Statistical Models

Height of students 1. Gaussian model, $f(x) = \phi_{\bar{x},s^2}(x)$

```
> Davis = read.table("http://freakonometrics.
    free.fr/Davis.txt")
> X = Davis$height
> hist(X, proba=TRUE)
> (param = fitdistr(X,"normal")$estimate)
     mean        sd
170.02000  11.97788
> f1 = function(x) dnorm(x,param[1],param[2])
> x = seq(100,210,by=.2)
> lines(x,f1(x),lwd=2)
```



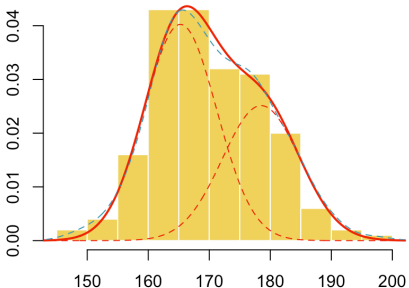

```
> logdf = function(x,p){
 p1 = p[1]
 m1 = p[2]; s1 = p[4]
 m2 = p[3]; s2 = p[5]
 log(p1*dnorm(x,m1,s1)+(1-p1)*dnorm(x,m2,s2))}
```

Parametric Statistical Models

Height of students 2. Mixture of 2 Gaussians,

$$f(x) = p\phi_{\mu_1,\sigma_1^2}(x) + (1-p)\phi_{\mu_2,\sigma_2^2}(x)$$

```
> logL = function(parameter) -sum(
    logdf(X,parameter))
> Amat = matrix(c
    (1,-1,0,0,0,0,0,0,0,0,0,1,0,0,0,
    0,0,0,0,0,0,1), 4, 5)
> bvec = c(0,-1,0,0)
> (param12 = constrOptim(c
    (.5,160,180,10,10), logL, NULL,
    ui = Amat, ci = bvec)$par)
[1]   0.5996263 165.2690084
    178.4991624   5.9447675
      6.3564746
```

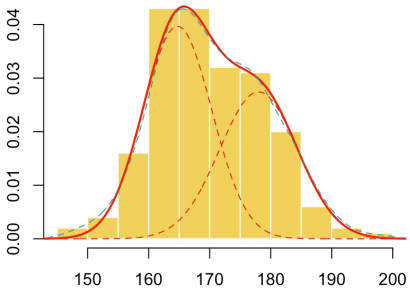


Parametric Statistical Models

Height of students 3. Conditional Gaussian

$$f(x) = \mathbb{P}(F) \cdot \phi_{\bar{x}_F, s_F^2}(x) + \mathbb{P}(M) \cdot \phi_{\bar{x}_M, s_M^2}(x)$$

```
> (pM = mean(Davis$sex=="F"))
[1] 0.56
> (paramF = fitdistr(X[Davis$sex=="F
    "],"normal")$estimate)
      mean         sd
163.74107  11.59183
> (paramM = fitdistr(X[Davis$sex=="M
    "],"normal")$estimate)
      mean         sd
178.011364   6.404001
```



Introduction

- Le modèle de mélange gaussien permet la partionnement en groupes de manière plus incertaine, on fait plus de place au doute et à l'aléatoire.
- Un concept de partionnement en groupes plus probabiliste, plus statistique.
- On considère l'assignement r_i comme une variable aléatoire.
- Au lieu d'avoir des assignments binaires $r = [0, 0, 1]$ on pourrait avoir des probabilités d'assignment $\pi = [p(r_1 = 1), p(r_2 = 1), p(r_3 = 1)]$.

Introduction

- On considère donc aussi la variance des groupes et non seulement leur centres de masse.
- Au lieu de calculer la distance euclidienne et de choisir le centre de masse le plus proche
- On peut calculer une distance probabiliste, qui considère la variabilité des groupes.
- Par exemple, on peut calculer la probabilité qu'une observation appartienne a une distribution normal étant donné sa moyenne (centre du groupe) et sa variance.
- On a ainsi une distance probabiliste en quelque sorte.

Mélange de modèles

- Introduisons tout d'abord ce qu'est un *mélance de modèles*, *Mixture models* en anglais.
- Le concept de mélange de modèles/mélange de distributions est pour bien capturer/modéliser des distributions multimodales.
- On exprime ce genre de modèle comme suit:

$$p(x) = \sum_{j=1}^{k} \pi_j p(x|\theta_j), \tag{1}$$

où $\pi_j = p(r_j = 1)$ est la probabilité que l'observation provienne du groupe j et $p(x|\theta_j)$ est une distribution de probabilité dont les paramètres dépendent du groupe j.

Mélange de modèle Gaussien

- **Gaussian Mixture Model** (GMM)
- Une distribution classique pour ce type de modèle est la loi normale.
- Dans ce contexte, la moyenne μ et la variance Σ dépendent du groupe

$$p(x) = \sum_{j=1}^{k} \pi_j \varphi(x|\mu_j, \Sigma_j), \tag{2}$$

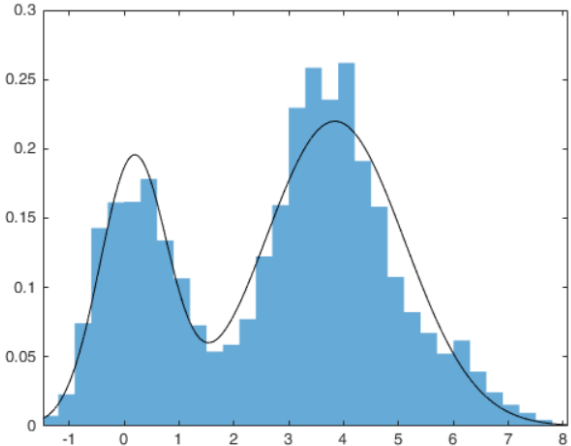
où $\boldsymbol{x} \mapsto \varphi(\boldsymbol{x}|\mu, \Sigma)$ est la densité de la loi normale $\mathcal{N}(\mu, \Sigma)$,

$$\varphi(\boldsymbol{x}|\mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(\boldsymbol{x}-\mu)^\top \Sigma^{-1}(\boldsymbol{x}-\mu)\right], \text{ sur } \mathbb{R}^d.$$

Mélange de modèle Gaussien

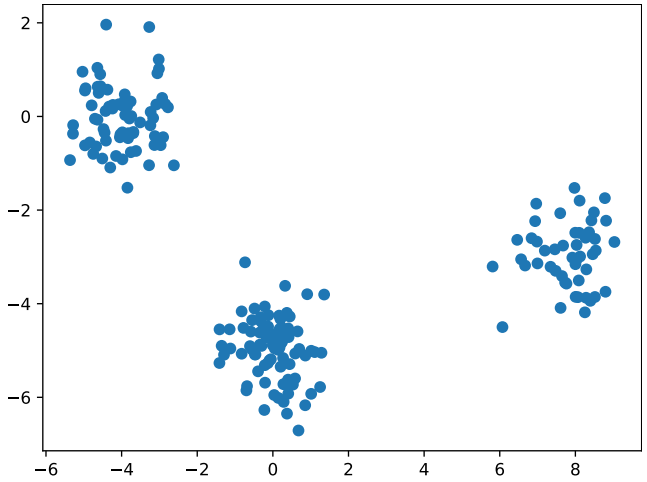
- Permet de capturer des distributions multimodals (plusieurs modes).
- Par exemples des populations hétérogènes; le poids d'animaux variés.
- π est la probabilité/proportion de chaque animal (chat, chien, lapin)
- Le poids de chaque animal est distribué selon un loi normal; par exemple $X_{Chat} \sim \mathcal{N}(\mu_{Chat}, \sigma_{Chat})$

Mélange de modèles Gaussien : 1 dimension



μ contrôle l'emplacement des pics, σ leurs largeurs, et π leurs hauteurs.

Mélange de modèles Gaussien : 2 dimensions



μ contrôle l'emplacement des blobs, σ leurs largeurs, et π la quantité de points.

Mélange de modèles Gaussien

- Toute population non-homogène peut être bien représentée par un mélange de modèles gaussiens.
- En théorie on peut approximer n'importe quelle distribution en ajustant, k, π, μ_j et Σ_j.

Mélange de modèles Gaussien: apprentissage non-supervisé

- L'idée est de capturer ou d'apprendre une distribution de x, $p(x)$ très complexe (multimodal bizarre, par exemple les images MNIST).
- Pour capture les différents modes de la distributon, on définit un ensemble de distributions conditionnelles de la sorte: $p(x|r_j = 1) = \varphi(x|\mu_j, \Sigma_j)$.
- Essentiellement on veut un mode par groupe/amas de point.
- L'ensemble de ces distributions conditionnelles crée ainsi une distribution multimodal pour $p(x)$.

Mélange de modèles

- Soit : $p(x|r_j = 1) = \varphi(x|\mu_j, \Sigma_j)$
- et $p(x, r_j = 1) = p(r_j = 1)p(x|r) = \pi_j\varphi(x|\mu_j, \Sigma_j)$
- alors: $p(x) = \sum_{j=1}^{k} p(r_j = 1)p(x|r_j = 1) = \sum_{j=1}^{k} \pi_j(x|\mu_j, \Sigma_j)$.
- En général on apelle tout type de distribution de la forme: $\sum_{j=1}^{k} p(r_j = 1)p(x|r_j = 1)$ un mélange de modèle; comme chaque composant intégre à 1, a mesure que les probabilités sommes a 1 aussi, le $p(x)$ résultant est bel et bien une distribution de probabilité et on peut définir $p(x|r_j = 1)$ comme on le veut.

Mélange de modèles Gaussien: Définition formelle

- On note $0 \leq \pi_j = p(r = j) \leq 1$ avec $\sum_{i=j}^{k} \pi_j = 1$, donc π est le vecteur de probabilité qu'une variable proviennent de la distribution $p(x|r_j = 1)$.
- Ceci définit un modèle *génératif*, une hypothèse de comment les données sont générées.
- Nous devons maintenant nous demander comment estimer les paramètre d'un tel modèles: π, μ, Σ

Mélange de modèles Gaussien: Définition formelle

Une valeur importante est la probabilité d'assignment étant donnés x, c'est-à-dire $\gamma(j) = p(r_j = 1|x)$ (probabilité a posteriori pour les bayesiens dans la salle)
Par la formule de Bayes:

$$\gamma(j) = p(r_j = 1|x) = \frac{p(r_i = 1, x)}{p(x)} \tag{3}$$

$$= \frac{\pi_j \varphi(x|\mu_j, \Sigma_j))}{\sum_{l=1}^{k} \pi_l \varphi(x|\mu_l, \Sigma_l)} \tag{4}$$

Observez que $\sum_j \gamma(j) = 1$.

Maximisation de vraissemblance: rapelle

- Il est standard d'apprendre un modèle probabiliste par maximisation de la vraisemblance

$$\mathcal{L}(\theta) = p(\mathbf{X}|\theta) = \prod_{i=1}^{n} p_\theta(x_i) \tag{5}$$

$$\Rightarrow \log(\mathcal{L}(\theta)) = \sum_{i=1}^{n} \log p_\theta(x_i) \tag{6}$$

On utilise $\sum_{i=1}^{n} \log p_\theta(x_i)$ comme une fonction objective (en fonction de θ). On peut la maximiser (en prennant la dérivé etc..)

Maximisation de vraissemblance: rapelle

Si $x \sim \mathcal{N}(\mu, \sigma)$ alors:

$$\sum_{i=1}^{n} \log p_\theta(x_i) = \sum_{i=1}^{n} \log \left(\frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-1}{2} \left(\frac{x_i - \mu}{\sigma} \right)^2 \right) \tag{7}$$

$$= \sum_{i=1}^{n} \log \left(\frac{1}{\sigma\sqrt{2\pi}} \right) + \sum_{i=1}^{n} \frac{-1}{2} \left(\frac{x_i - \mu}{\sigma} \right)^2 \tag{8}$$

$$\propto \sum_{i=1}^{n} (x_i - \mu)^2 + \text{ constante} \tag{9}$$

(Si la distribution est normal, maximiser la vraissemble et minimser l'EQM est la même chose... cf. cours de régression)

Mélange de modèles Gaussien: Apprentissage

- La vraisemblance d'un jeu de donnés $\mathbf{X}$ en fonction de nos paramètres π_js, μ_j's et Σ_j's est:

$$p(\mathbf{X}|\pi,\mu,\Sigma) = \prod_{i=1}^{n}\sum_{j=1}^{k}\pi_j\varphi(x_i|\mu_j,\Sigma_j) \tag{10}$$

$$\Rightarrow \ln p(\mathbf{X}|\pi,\mu,\Sigma) = \sum_{i=1}^{n}\ln\sum_{j=1}^{k}\pi_j\varphi(x_i|\mu_j,\Sigma_j) \tag{11}$$

$$\tag{12}$$

Malheureusement, on ne peut calculer la dérivée de cette vraisemblance, conjointement par rapport a π_js, μ_j's et Σ_j's. (à cause du mélange!)

Mélange de modèles Gaussien: Apprentissage

- La vraisemblance est une fonction objective comme les autres.
- Malheureusement, on ne peut pas maximiser la vraisemblance, conjointement par rapport a π_js, μ_j's et Σ_j's.
- Tout comme on ne pouvais pas maximiser la fonction objective:

$$Obj = \sum_{i=1}^{n} \sum_{j=1}^{k} r_{i,j} \sqrt{(\mathbf{x}_i - \mu_j)^2}, \tag{13}$$

en fonction de $r_{i,j}$ et μ_j en simultané pour le partitionnement en k-moyennes.

Mélange de modèles Gaussien: Apprentissage

- Par contre, si les paramètres μ_j's et Σ_j's sont connues, nous pouvons facilement maximiser la fonction pas rapport à π_js.
- Pareillement, si lesπ_js sont connus, nous pouvons facilement déterminer les valeurs des μ_j's et Σ_j's qui maximise la fonction objective.
- Tout comme le partitionnement en k-moyenne!
- On peut en quelque sorte maximiser la fonction objective en alternance.

Partitionnement en k-moyenne: Apprentissage

- L'apprentissage de mélange de modèles Gaussien ressemble *énormément* à l'apprentissage de partitionnement en k-moyenne.
- Rappel:

1. Étant donné des groupes d'observations, on estime la moyenne du groupe en calculant la moyenne empirique des observations dans le groupe.
2. Étant données des moyennes de groupe, on assigne les observations aux groupes en choisissant la moyenne la plus proche.

Mélange de modèles Gaussien: Apprentissage

- La différence principale est que l'on obtient maintenant un assignement probabiliste où $\gamma(i,j) = p(r_j = 1|x_i)$ représente la probabilité que l'observation i appartienne au groupe j.
- On ajuste donc l'algorithme précédent pour tenir compte de cette assignment probabiliste.

1. Étant donné des probabilités d'assignement, on estime des moyennes pondérées par les probabilités d'assignement (on estime aussi la variance).
2. Étant données les moyennes et variances de groupe, on estime les probabilités d'assignement γ.

Algorithme: partitionnement en k-moyennes

Tout d'abord on initialise les soit les centres μ ou les assignement r (ou les deux)

1.

$$r_{i,j} = \begin{cases} 1 \text{ si } j = \underset{l}{\text{argmin}} ||\mathbf{x_i} - \mu_l||^2 \\ 0 \text{ sinon} \end{cases}$$

2.

$$\mu_j = \frac{\sum_{i=1}^{n} r_{i,j} \mathbf{x}_i}{\sum_{i=1}^{n} r_{i,j}}$$

On peut donc en quelque sorte minimiser la fonction objective en alternance.

Mélange de modèles Gaussien: Apprentissage

Tout d'abord on initialise les paramètres μ_j, Σ_j et π.

1. Évaluer les probabilités d'assignement: $\gamma(i,j) = \dfrac{\pi_j \varphi(x_i|\mu_j, \Sigma_j))}{\sum_{l=1}^{k} \pi_l \varphi(x_j|\mu_l, \Sigma_l)}$
2. Estimé les moyennes $\mu_j = \dfrac{1}{n_j} \sum_{i=1}^{n} \gamma(i,j) x_i$, et les variances:

$$\Sigma_j = \frac{1}{n_j} \sum_{i=1}^{n} \gamma(i,j)(x_i - \mu_j)(x_i - \mu_j)^\top \text{ et } \pi_j = \frac{n_j}{n}$$

Répéter jusqu'à stabilité (en croisant les doigts car le problème a plusieurs optimums - problème des modèles non-identifiables)

où $n_j = \sum_{i=1}^{n} \gamma(i,j)$ l'espérance du nombre d'observations aux groupes j.

Mélange de modèles Gaussien: Apprentissage

- Cet algorithme s'appelle EM pour *Expectation-Maximisation* et on peut démontrer que celui-ci maximise tranquillement la log-vraisemblance:

$$\ln p(\mathbf{X}|\pi, \mu, \Sigma) = \sum_{i=1}^{n} \ln \sum_{j=1}^{k} \pi_j N(x_i|\mu_j, \Sigma_j) \tag{14}$$

$$\tag{15}$$

Un peu hors des attentes du cours, mais il est possible de lire le chapitre 8 et ensuite 9.3 (Bishop and Nasrabadi (2006)).

Mélange de modèles Gaussien: Apprentissage

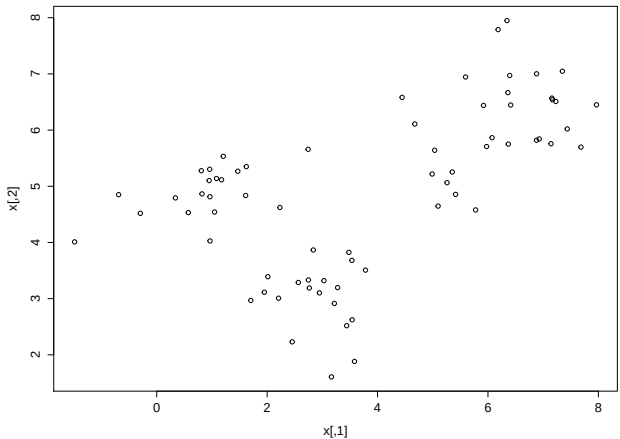


Figure 1: Initialisation aléatoire des paramètres

Mélange de modèles Gaussien: Apprentissage

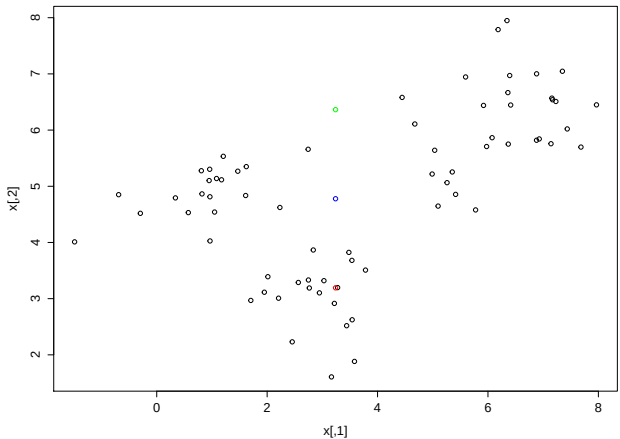


Figure 2: 1er itération

Mélange de modèles Gaussien: Apprentissage

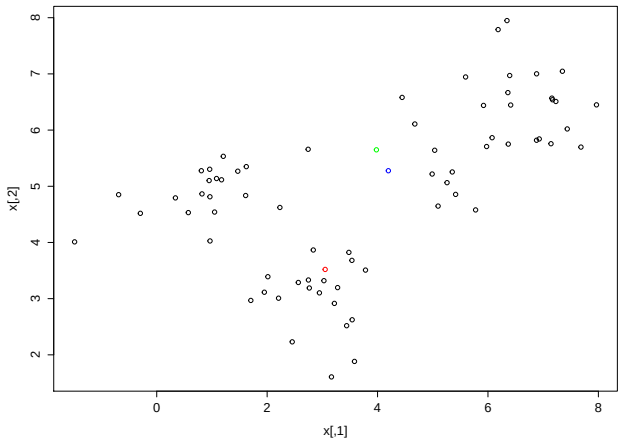


Figure 3: 5 itérations

Mélange de modèles Gaussien: Apprentissage

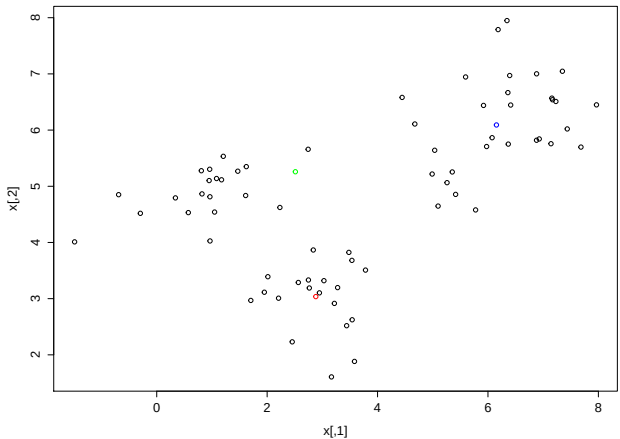


Figure 4: 10 itérations

Mélange de modèles Gaussien: Apprentissage

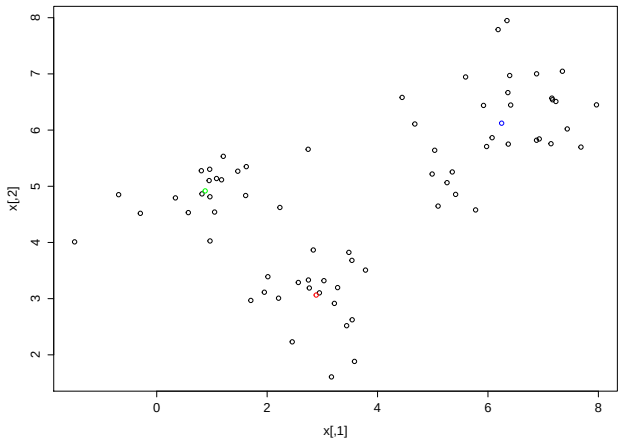


Figure 5: 50 itérations

Mélange de modèles Gaussien: Apprentissage

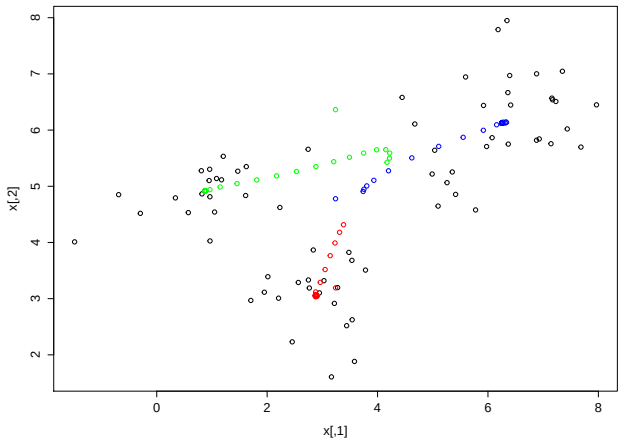


Figure 6: Position de la moyenne à travers 50 itérations.

Mélange de modèles Gaussien: Apprentissage

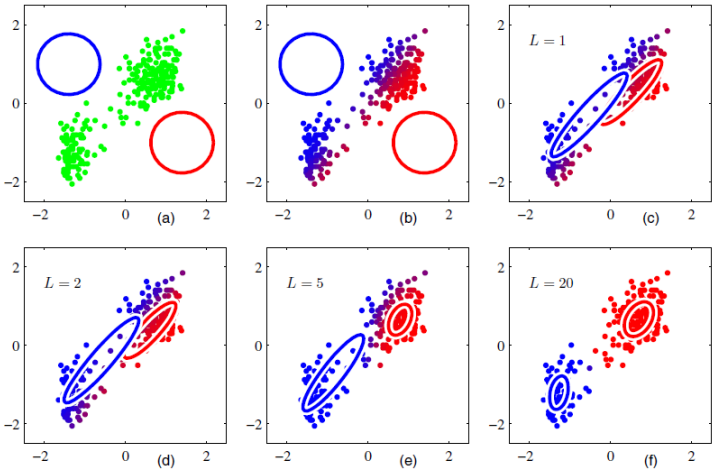


Figure 7: Extrait de Bishop and Nasrabadi (2006)

Mélange de modèle Gaussien: Problème

- La vraisemblance n'est pas toujours une bonne fonction objective.
- Exemple si $\mu_j = x_i$ pour un i quelconque, alors la vraisemblance limite sera infini si $\Sigma_j \to 0$:

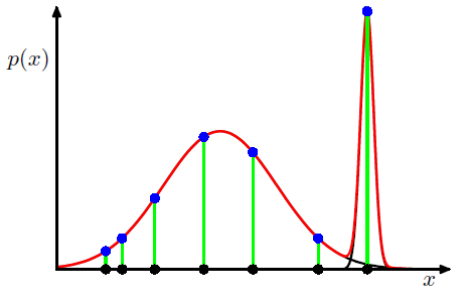


Figure 8: Extrait de Bishop and Nasrabadi (2006)

Mélange de modèle Gaussien: Problème

- La vraisemblance ne fait qu'augmenter si on augmente le nombre de groupes k.
- Le Akaike information criterion (AIC) et Bayesian information criterion (BIC) suggèrent de choisir k à l'aide d'une vraisemblance pénalisée:
- $\text{AIC} = 2p - 2\ln(\mathcal{L})$
- $\text{BIC} = \ln(p) - 2\ln(\mathcal{L})$

où p est le nombre de paramètres et $\ln(\mathcal{L})$ la log-vraisemblance.

- On peut donc calculer l'AIC ou le BIC et déterminer le modèle idéal par celui ayant la plus petite valeur.

Relation entre le GMM et le KMC

- On dit que l'assignement du KMC, $r_i = [0, 1, 0]$ est tranché (hard) alors que celui du MMG est flexible (soft) $\gamma(i) = [0.3, 0.5, 0.2]$.
- Pour KMC, on assigne un point au groupe qui semble être le meilleur choix, alors que pour GMM, on détermine explicitement la probabilité que le point provienne de chacun de ces choix (vraisemblance).
- Le partitionnement en k-moyenne est un sous-cas du mélange de modèles gaussien.

Relation entre le GMM et le KMC

- Le GMM est une généralisation du KMC.
- On *récupère* le KMC à partir du GMM en posant $\Sigma = \varepsilon I$.
- Et ensuite en prenant la limite $\varepsilon \to 0$

Modèle à variables latentes (PMLR Chapitre 8)

- Le GMM fait partie d'une grande classe de modèles à variables latentes.
- Ces modèles sont souvent représentés par des graphiques, ici:

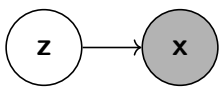


Figure 9: Représentation graphique de modèle à variable latente $p(\mathbf{x}, \mathbf{z}) = p(\mathbf{z})p(\mathbf{x}\|\mathbf{z})$.

Les arrêtes représentent des liens de corrélation.

Représentation graphique

- Chaîne de Markov

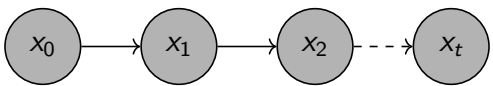


Figure 10: Le graphique d'une chaîne de Markov est une chaîne.

Modèle à variables latentes

- On présuppose l'existence d'une variable latente, cachée, non observée (ici z ou le groupe r).
- On suppose que celle-ci influence directement ce que nous observons x.
- Pour bien capturer la distribution de x, $p(x)$, nous modélisons activement z à l'aide de $p(x, z) = p(z)p(x|z)$.

Modèle à variables latentes

- Le GMM est un modèle simple de cette large classe de modèle, une bonne intro à ces idées.
- $z = r \sim \text{Multi}(n = 1, \pi)$
- $p(x|z) = p(x|r_j = 1) = \varphi(x|\mu_j, \Sigma_j)$

L'algorithme EM

L'algorithme suivant, est un algorithme EM conçu pour les MMG:

1. Initialiser les paramètres μ_j, Σ_j et π.
2. Évaluer les probabilités d'assignement: $\gamma(i,j) = \dfrac{\pi_j N(x_i|\mu_j, \Sigma_j))}{\sum_{l=1}^{k} \pi_l N(x_j|\mu_l, \Sigma_l)}$
3. Estimer les moyennes $\mu_j = \frac{1}{n_j} \sum_{i=1}^{n} \gamma(i,j) x_i$, variance:
   $\Sigma_j = \frac{1}{n_j} \sum_{i=1}^{n} \gamma(i,j)(x_i - \mu_j)(x_i - \mu_j)^\top$ et $\pi_j = \dfrac{n_j}{n}$

Répéter jusqu'à stabilité (en croisant les doigts car le problème a plusieurs optimums - problème des modèles non-identifiables)
Par contre, sur le principe EM peut s'appliquer à tout modèle à variables latentes.
voir 9.4 (Bishop and Nasrabadi (2006)) pour plus de détails.

Pour le prochain cours:

- Lecture: 8 (Bishop and Nasrabadi (2006)), 9.2 (Bishop and Nasrabadi (2006)), 9.3 (Bishop and Nasrabadi (2006))
- Préparation: 12 (Bishop and Nasrabadi (2006)), 12.2 (James (2013))
- Projet

Références

Bishop, C. M. and Nasrabadi, N. M. (2006). *Pattern recognition and machine learning*, volume 4. Springer.

Everitt, B. and Hand, D. (1981). *Finite mixture distributions, Chapman and Hall*. London.

James, G. (2013). *An introduction to statistical learning*. springer.

McLachlan, G. (2000). *Finite mixture models*. Wiley-interscience publication.