Projet Master 2 SSD

Crowd-Sourcing

Rédigé par

PRALON Nicolas

THIRIET Aurelien

Encadrant : Joseph SALMON

Table des matières

Introduction

En science appliquée et notamment en statistique inférentielle, le recueil de données constitue une étape primordiale, il nous permet d'élaborer des modèles et de construire des estimateurs. Il convient à ce titre d'être attentif à l'observation des données.

Dans un cadre idéal, les modèles construits nécessitent un faible nombre de données dont l'observation peut être réalisée par des professionnels du domaine concerné. Toujours est-il que ces situations sont peu courantes et l'observation, par ces experts, d'un grand nombre de données n'est pas envisageable. C'est à ce titre que le "Crowd-Sourcing" est couramment utilisé, puisqu'un grand nombre d'intervenants permet de construire la base de donnée nécessaire. Toutefois des erreurs d'obervations quant à la nature des données peut-être commises, il est alors essentiel de s'accorder sur une décision.

C'est à ce problème, que ce projet propose de présenter différentes solutions. Nous étudierons le modèle construit par DAWID et SKENE [1].

Chapitre 1

Modélisation du problème

1.1 Modèle Dawid-Skene

Dans ce chapitre, nous établissons le cadre nous permettant de traiter le problème de décision lorsque plusieurs obervations d'une même donnée sont fournies.

Nous définissons alors l'exemple d'un ensemble d'images à labeliser, les experts du domaines décident alors de faire appel à plusieurs participants pour labéliser ces images.
On définit le cadre suivant :

- On dispose d'un esemble d'images $\{1, \cdots, I\}$ et on suppose le vecteur aléatoire $(X_i)_{i\in\{1,\cdots,I\}}$, tel que $\forall i$, X_i à valeur dans $\{1, \cdots, J\}$ correspond à l'objet illustré sur l'image i.
- On dispose egalement d'un ensemble de participants (annotateurs) $\{1, \cdots, K\}$, et on considère $\forall i \in \{1, \cdots, I\}$ le vecteur aléatoire $(Y_i^k)_{k\in\{1,\cdots,K\}}$ la labélisation du participant k à l'image i, Y_i^k à valeur dans $\{0, 1\}^J$ (par exemple {0,...,0,1,0,..,0} si le participant annote une seule fois l'image k comme ce sera le cas dans la partie 2).
- Chaque participant répond indépendamment des autres et chaque image est indépendante des autres, elle ne donne aucune indication sur les autres images . On note l'indépendance 2 à 2, $(Y_i^k)_k \perp\!\!\!\perp$ et $(X_i)_i \perp\!\!\!\perp$

A partir des résultats donnés par chaque participant nous pouvons essayer d'attribuer un label à chaque image. Toutefois nous pouvons nous retrouver dans la situation où le label donné par un participant à une image diffère de celui d'un autre participant, pour diverses raisons ; erreur d'annotation, fatigue, spam. Il convient dans ce cas de trouver une solution à la question : quel label donner à chaque image ?

Afin de simplifier les démarches dans un premier temps, nous allons envisager le cadre suivant ; nous allons considérer, en plus de l'observation des résultats des participants, observer les labels des images.
Ce cadre s'écarte légèrement de celui du problème puisque l'information sur les labels des images n'est pas accessible, cependant nous verrons comment pallier cela dans la partie suivante ; notamment par une garantie théorique d'un algorithme que l'on appelle algorithme EM.
Dans un premier temps nous considérons le cas d'un participant et une image.

Nous émettons l'hypothèse suivante :

Hypothèse 1. *Nous supposons que,* $\forall j, k \in [\![1, J]\!] \times [\![1, K]\!]$, $Y_i^k | X_i = \alpha_i \sim Multinomiale(n, (\pi^k_{\alpha_i,j})_{j\in[\![1,J]\!]})$

Sachant que l'image i a pour label α_i, la réponse du participant k suit une loi multinomiale. Dans le cas où le label de l'image i est connu, le participant k peut aussi y avoir accès, et nous avons Y_i^k, X_i non indépendant.

Chaque paramètre $(\pi^k_{\alpha_i,j})_{j\in[\![1,J]\!]}$ correspond à la probabilité que le participant k attribue le label j à l'image i sachant que le label correct est le label α_i.

On a ainsi :

$$\mathbb{P}_{Y_i^k|X_i=\alpha_i} = \sum_{(n_{i,1}^k,\cdots,n_{i,J}^k)} \underbrace{\frac{(\sum_{j=1}^{J} n_{i,j}^k)!}{\prod_{j=1}^{J}(n_{i,j}^k)!} \prod_{j=1}^{J} \left(\pi_{\alpha_i,j}^k\right)^{n_{i,j}^k} \quad \delta_{(n_{i,1}^k,\cdots,n_{i,J}^k)}}_{densité\ suivant \sum \delta_{(n_{i,1}^k,\cdots,n_{i,J}^k)}}$$

Avec $n_{i,j}^k$ le nombre de fois que le participant k a attribué le label j à l'image i

On a alors que la vraisemblance de $Y_i^k|X_i = \alpha_i$ est proportionnelle à $\prod_{j=1}^{J} \left(\pi_{\alpha_i,j}^k\right)^{n_{i,j}^k}$

Remarque 1. Ici on a une dépendance des paramètres $(\pi_{\alpha_i,j}^k)$ en k,j et α_i (pour le conditionnement), mais l'indice i sert simplement à indiquer qu'on considère l'image i et ne consitue pas un paramètre.
Cela signifie que, sachant que le label de l'image i est α_i et que le label de l'image i' est $\alpha_{i'}$ mais $\alpha_i = \alpha_{i'}$, la probabilité que le participant k attribue le label j à l'image i est la même de celle où il attribue le label j pour l'image i'. En tout état de connaissance, le participant sachant que les deux images ont le même label, la probabilité qu'il annote le label j à l'une, et j à l'autre n'a pas lieu d'être différente. Il s'agit pour le particpant d'une même situation et le choix ne dépend que de lui.

Dans le but de se ramener au cas de plusieurs images et plusieurs participants, on considère ici une image i et tous les participants $(1,\cdots,K)$.

Remarque 2. Par souci de lisibilité on notera $\mathbb{P}(Y_i^k = (n_{i,1}^k,\cdots,n_{i,J}^k)|X_i = \alpha_i)$ par $\mathbb{P}(Y_i^k|X_i = \alpha_i)$.

Pour tous les participants on a dans ce cas :

$$\begin{aligned}\mathbb{P}\left(\bigcap_{k=1}^{K} Y_i^k|X_i = \alpha_i\right) &= \prod_{k=1}^{K} \mathbb{P}\left(Y_i^k|X_i = \alpha_i\right),\ car(Y_i^k)_k \perp\!\!\!\perp \\ &\propto \prod_{k=1}^{K}\prod_{j=1}^{J} \left(\pi_{\alpha_i,j}^k\right)^{n_{i,j}^k}\end{aligned}$$

La vraisemblance du vecteur $(Y_i^k|X_i = \alpha_i)_k$ est proportionnelle à cette valeur.

Ramenons nous à présent au cadre de notre problème et intéressons nous au vecteur $((Y_i^k)_k, X_i)$.
Nous pouvons de ce qui précède déterminer la probabilité $\mathbb{P}\left(\left(\bigcap_k Y_i^k\right)\bigcap(X_i = \alpha_i)\right)$.

On note $\forall(i,j) \in \{1,\cdots,J\}^2, p_j = \mathbb{P}(X_i = j)$.

$$\begin{aligned}\mathbb{P}\left(\left(\bigcap_k Y_i^k\right)\bigcap(X_i = \alpha_i)\right) &= \mathbb{P}\left(\bigcap_{k=1}^{K} Y_i^k|X_i = \alpha_i\right) \times \mathbb{P}(X_i = \alpha_i) \\ &\propto p_{\alpha_i} \times \prod_{k=1}^{K}\prod_{j=1}^{J} \left(\pi_{\alpha_i,j}^k\right)^{n_{i,j}^k}\end{aligned}$$

p_{α_i} désigne la probabilité qu'une image tirée au hasard ait le label α_i.

Plus généralement, en définissant $T_{i,l}$ la variable aléatoire telle que $T_{i,l} = 1$, si le label de l'image i est l et $T_{i,l} = 0$ sinon, on a :

$$p_{\alpha_i} \prod_{k=1}^{K} \prod_{j=1}^{J} \left(\pi^k_{\alpha_i,j}\right)^{n^k_{i,j}} = \prod_{l=1}^{J} \left(p_l \prod_{k=1}^{K} \prod_{j=1}^{J} \left(\pi^k_{l,j}\right)^{n^k_{i,j}} \right)^{T_{i,l}}$$

De plus l'événement $\{X_i = \alpha_i\} = \{T_{i,\alpha_i} = 1\}$, $\forall l \in \{1, \cdots, J\}$ $T_{i,l}$ est alors connu.

Il manque maintenant à se placer dans le cas où on observe les résutats des participants pour toutes les images et les labels de chaque image.

$\forall k, (Y^k_i) \perp\!\!\!\perp X_{i'}, i' \neq i$ et $(Y^k_i)_i \perp\!\!\!\perp$, on a alors $Y^k_i \perp\!\!\!\perp (Y^k_{i'}, X_{i'})$ puis $(Y^k_i, X_i) \perp\!\!\!\perp (Y^k_{i'}, X_{i'})$
On a ainsi la probabilité :

$$\begin{aligned}
\mathbb{P}\left(\left(\bigcap_i \bigcap_k Y^k_i\right) \cap \left(\bigcap_i Xi = \alpha_i\right)\right) &= \mathbb{P}\left(\bigcap_i \left[\left(\bigcap_k Y^k_i\right) \cap (X_i = \alpha_i)\right]\right) \\
&= \prod_i \mathbb{P}\left(\left(\bigcap_k Y^k_i\right) \cap (X_i = \alpha_i)\right) \\
&\propto \prod_{i=1}^{I} \prod_{l=1}^{J} \left(p_l \prod_{k=1}^{K} \prod_{j=1}^{J} \left(\pi^k_{l,j}\right)^{n^k_{i,j}} \right)^{T_{i,l}}
\end{aligned}$$

La vraisemblance est également proportionnelle à cette même expression.

1.2 Résolution du problème

Pour faire un choix quant au label de chaque image, la démarche que l'on adopte est la suivante :

Soit l'image i, on décide que le label de cette image est l si :

$$l = argmax_l(T_{i,l})$$

Les paramètres $T_{i,l}$ n'étant pas connu on souhaite les estimer.
Un estimateur naturel pouvant être pris pour chaque $T_{i,l}$ est l'espérance conditionnelle sachant $(Y^k_i)_k$, $\mathbb{E}[T_{i,l}|(Y^k_i)_k]$.
Il s'agit de la meilleure approximation de $T_{i,l}$ au sens de la norme $\mathscr{L}^2$, à partir de ce que l'on connait $< (Y^k_i)_k >$ (les réponses des participants).

$\mathbb{E}[T_{i,l}|(Y^k_i)_k]$ peut se réécrire sous la forme suivante :

$$\begin{aligned}
\mathbb{E}[T_{i,l}|(Y^k_i)_k] &= \mathbb{E}[0 \times \mathbb{1}_{\{T_{i,l}=0\}} + 1 \times \mathbb{1}_{\{T_{i,l}=1\}}|(Y^k_i)_k] \\
&= \mathbb{E}[\mathbb{1}_{\{T_{i,l}=1\}}|(Y^k_i)_k] \\
&= \mathbb{P}\left(T_{i,l} = 1|(Y^k_i)_k\right)
\end{aligned}$$

En explicitant cette probabilité on a :

$$
\begin{aligned}
\mathbb{P}\left(T_{i,l}=1|(Y_i^k)_k\right) &= \frac{\mathbb{P}\left((T_{i,l}=1)\cap\left(Y_i^k\right)_k\right)}{\mathbb{P}\left(\left(Y_i^k\right)_k\right)} \\
&= \frac{\mathbb{P}\left(\left(Y_i^k\right)_k|\left(T_{i,l}=1\right)\right)\times\mathbb{P}(T_{i,l}=1)}{\sum_q^J \mathbb{P}\left(\left(Y_i^k\right)_k|\left(X_i=q\right)\right)\times\mathbb{P}(X_i=q)} \\
&= \frac{\mathbb{P}\left(\left(Y_i^k\right)_k|\left(X_i=l\right)\right)\times\mathbb{P}(X_i=l)}{\sum_q^J \mathbb{P}\left(\left(Y_i^k\right)_k|\left(X_i=q\right)\right)\times\mathbb{P}(X_i=q)} \\
&= \frac{p_l\prod_k^K\prod_j^J(\pi_{l,j}^k)^{n_{i,j}^k}}{\sum_q^J p_q\prod_k^K\prod_j^J(\pi_{q,j}^k)^{n_{i,j}^k}}, car\ \forall i,l\ \{X_i=l\}=\{T_{i,l}=1\}
\end{aligned}
$$

Remarque 3. On a donc réussi à obtenir une estimation des $T_{i,l}$ dépendant des paramètres $(\pi_{l,j}^k)_{k,l,j}$ et p_l. Cependant ces paramètres n'étant pas connus, il nous faut les estimer et on les estime par maximum de vraisemblance, à partir du modèle défini dans la partie précédente.
Un problème est présent, ceci étant dû au modèle que l'on a défini car, la détermination des estimateurs du maximum de vraisemblance nécessite de bénéficier des observations des labels des images, qui ne sont pas connus, en particulier il nous faut avoir accès aux valeurs des $T_{i,l}$ dans l'expression de la vraisemblance.

Ceci ne nous permet pas, dans ce cas, de donner les estimateurs du maximum de vraisemblance des $(\pi_{l,j}^k)_{k,l,j}$ et p_l.
Toutefois, on voit apparaître une procédure permettant d'estimer les estimateurs du maximum de vraisemblance et l'espérance conditionnelle des $T_{i,l}$, il s'agit dans un premier temps, d'estimer arbitrairement les $T_{i,l}$ afin d'obtenir une estimation des estimateurs du maximum de vraisemblance $(\pi_{l,j}^k)_{k,l,j}$ et p_l.
On peut alors donner une estimation des $\mathbb{E}[T_{i,l}|(Y_i^k)_k]$, et de nouveau obtenir une estimation des estimateurs du maximum de vraisemblance des $(\pi_{l,j}^k)_{k,l,j}$ et p_l. Cette procédure itérative correspond à ce que l'on appelle l'algoritme EM.

Il n'existe pas de preuve de convergence de la suite de paramètres établie par l'algorithme EM vers les estimateurs définis, et le choix de bons paramètres initiaux est de fait primordial.
Comme sus-mentionné en début de partie, nous disposons d'une garantie théorique de l'algoritme EM justifiant la démarche que nous avons introduite ; celle-ci étant que les estimateurs ainsi obtenus ont la particularité de faire croitre, à chaque itération, la vraisemblance des observations.
Toutefois, ce résultat ne prétend pas que l'algorithme EM maximisera la vraisemblance des observations. Ce dernier peut tout à fait produire une suite de paramètres correspondant à un maximum local ; et une augmentation du nombre d'itérations n'apportera pas de solution à ce problème.

Considérerons connu les valeurs de Ti, l, et déterminons à présent les expressions des estimateurs du maximum de vraisemblance.

En effectuant la maximisation du Lagrangien de la fonction de vraisemblance :

$$\left(\left(\pi_{l,j}^k\right)_{l,j,k}, (p_j)_j\right) \longmapsto \prod_i^I \prod_{l=1}^J \left(p_l \prod_{k=1}^K \prod_{j=1}^J \left(\pi_{l,j}^k\right)^{n_{i,j}^k} \right)^{T_{i,l}}$$

Sous la contrainte $\forall l, k \in [\![1, J]\!] \times [\![1, K]\!], \left(\sum_{j=1}^J \pi_{l,j}^k = 1 \right)$

On obtient les estimateurs du maximum de vraisemblance suivants :

$$\widehat{\pi_{l,j}^k} = \frac{\sum_i T_{i,l} n_{i,j}^k}{\sum_j \sum_i T_{i,l} n_{i,j}^k}$$

$$\widehat{p_l} = \frac{\sum_i T_{i,l}}{I}$$

Remarque 4. La preuve alourdissant la lecture, on se contente de donner l'idée générale permettant de retrouver les estimateurs.
Il convient de faire ressortir de la fonction ci-dessus, pour l'estimation des $\pi_{l,j}^k$ (respectivement des p_l), le terme en $\pi_{l,j}^k$ (repectivement en p_l). On obtient alors les termes suivant $\prod_i^I \left(\pi_{l,j}^k\right)^{n_{i,j}^k T_{i,l}} a_i$ et $\prod_i^I (p_l)^{T_{i,l}} b_i$ avec a_i et b_i correspondant au reste du produit.
Les contraintes de maximisation permettent ensuite d'obtenir les dénominateurs dans l'expression des estimateurs.

1.3 Initialisation des $T_{i,l}$

On peut proposer une initialisation simple des $T_{i,l}$ sans la base d'une quelconque information des données, en les estimant par :

$$T_{i,l} = \frac{1}{J}$$

Mais cette initialisation n'est pas optimale pour espérer estimer converger vers les estimateurs du maximum de vraisemblance.

Si les participants annotent suffisament bien les images, on peut donner une meilleure initialisation des $T_{i,l}$ en se basant sur les résultats obtenus des participants, dans ce cas on estime $T_{i,l}$ par :

$$T_{i,l} = \frac{\sum_k n_{i,l}^k}{\sum_k \sum_j n_{i,j}^k}$$

Il s'agit d'estimer $T_{i,l}$ par le nombre de fois où les participants ont annoté l l'image i sur l'ensemble des annotations qu'ils ont données pour l'image i, ici les $n_{i,l}^k$ valent soit 1 soit 0. On estime $T_{i,l}$ par une sorte de "vote majoritaire".

Chapitre 2

Pseudo-Code algoritme EM

Dans cette partie nous allons présenter l'algorithme EM, il s'agit d'une méthode itérative, constituée de deux étapes, à savoir une étape "Expectation" et une étape "Maximization". Cette algorithme nous permet de répondre à notre problèmatique initiale, par la garantie théorique de la croissance de la vraisemblance du modèle défini dans la partie Dawid-Skene.

2.1 L'étape E (Expectation)

La phase E consiste à calculer l'estimation des $T_{i,l}$ par $\mathbb{E}[T_{i,l}|(Y_i^k)_k]$.
Puisque seule son expression est nécessaire pour obtenir l'estimation des estimateurs du maximum de vraisemblance des $(\pi_{l,j}^k)_{k,l,j}$ et p_l. Son calcul est rendu possible par la forme analytique que l'on a pû déterminer précédement à la page 6, il convient dans un premier temps d'estimer les $(\pi_{l,j}^k)_{k,l,j}$ et p_l en ayant utilisé l'initialisation des $T_{i,l}$.

2.2 L'étape M (Maximization)

La phase M consiste à maximiser la vraisemblance du modèle, ce maximum est atteint en les paramètres $\left(\widehat{\pi_{l,j}^k}, \widehat{p_l}\right)$ et ces deux paramètres seront calculés à l'aide des maximas établis à la page 7. Ce sont ces estimateurs qui seront utilisés dans l'itération suivante pour mettre à jour estimation à l'étape E.

2.3 Pseudo-code

Algorithm 1 L'algorithme EM (Dempster et al., 1977).

Entrée(s) : initialisation des $(T_{i,l})_{i,l} = (T_{i,l})_0$, les observations $\left(Y_i^k\right)_{k,i}$, $(n_{i,l}^k)_{k,i,l}$ et $N \in \mathbb{N}$ le nombre d'intération ;

1: **pour** n allant de 1 à N **faire**

2: **ETAPE M :** *Calculer les* $\left(\widehat{\pi_{l,j}^k}\right)_n = \dfrac{\sum\limits_i (T_{i,l})_{n-1}\, n_{i,j}^k}{\sum\limits_j \sum\limits_i (T_{i,l})_{n-1}\, n_{i,j}^k}$, $(\widehat{p_l})_n = \dfrac{\sum\limits_i (T_{i,l})_{n-1}}{I}$

3: **ETAPE E :** *Calculer* $\left(\widehat{T_{i,l}}\right)_n = \dfrac{(\widehat{p_l})_n \prod\limits_k^K \prod\limits_j^J \left(\widehat{\pi_{l,j}^k}\right)_n^{n_{i,j}^k}}{\sum\limits_q^J (\widehat{p_q})_n \prod\limits_k^K \prod\limits_j^J \left(\widehat{\pi_{l,j}^k}\right)_n^{n_{i,j}^k}}$;

4: **fin du pour**

5: **retourner** $max_l \left(\widehat{T_{i,l}}\right)_N$;

Chapitre 3

Application numérique - Labels de CIFAR 10

3.1 CIFAR 10

CIFAR 10 est un ensemble de 60 000 images 32x32 réparties en 10 classes :

0	1	2	3	4	5	6	7	8	9
airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck

Il y a 50 000 images pour le "training" des algorithmes de type réseaux de neurones et 10 000 images de "test". Nous nous intéressons exclusivement au sous-ensemble "test". Celui-ci contient exactement 1 000 images de chaque classe. (On pourrait donc utiliser directement $p_l = 0.1$ plutôt que de l'estimer). CIFAR-10H nous donne accès à 514 200 annotations d'un total de 2571 annotateurs sur nos 10 000 images. Nous avons pris soin d'enlever les annotations "check trials" qui n'entrent pas en compte dans les annotations réelles.

En prenant en compte l'identifiant des annotateurs, le label choisi par chacun sur une image donnée (chaque image a été annoté 50 fois en moyenne $(10000 * 50 = 500000)$), et l'identifiant de l'image correspondante, nous pouvons utiliser l'algorithme de la section précédente.

3.2 Applications

Une simulation du jeu de données est disponible sur notre page GitHub : `https://github.com/Aurelien2021/Projet_Crowd_Sourcing` en suivant les instructions données sur le Beamer.

A noter à la différence du modèle développé dans la première partie, que, ici les participants n'annotent pas l'entièreté des images, mais les images sont annotées par un certain nombre de participants. On a aura dans ce cas que les termes en $(n_{i,j}^k)_{k,i,j}$ peuvent être pour certaines images i nuls pour tout choix de label j du participant k.

Le modèle introduit diffère quelques peu du cadre de notre projet, puisque les participants annotent toutes les images dans le modele défini. Toutefois le modèle peu être utilisé et les estimateurs du maximum de vraisemblance gardent toujours de leur sens dans le cadre du "Crowd-Sourcing".

Remarque 5. L'implémentation de l'algorithme EM fonctionne, cependant nous obtenons pour les valeurs des $T_{i,l}$ de sortie, les mêmes que celles données en initialisation. Ce problème n'a pas encore été résolu, une cellule est disponible si l'on veut tester différentes valeurs (nous obtenons des valeurs $T_{i,l}$ de sortie différentes lorsque l'on initialise aléatoirement les $T_{i,l}$.

Chapitre 4

Bibliographie

[1] Dawid A.P., Skene A.M. (1979). Maximum Likelihood Estimation of Observer Error-rates using the EM Algorithm *Applied Statistics 28 (1) (1979), pp. 20–28.*
Pour télécharger le document : `https://canvas.northwestern.edu/courses/65895/files/4174911/download?verifier=lUyedGR8Yb7I4NwFOtosvJqir8Oc6huVBwBgI5ko&wrap=1`