le cnam

Sujet UE RCP209
Apprentissage, réseaux de neurones et modèles graphiques

Année universitaire 2016–2017

Examen : 29 juin 2017

Responsable : Michel CRUCIANU

Durée : 3h00

Seuls documents autorisés : 2 pages A4 recto-verso, manuscrites.

Les téléphones mobiles et autres équipements communicants (PC, tablette, etc.) doivent être éteints et rangés dans les sacs pendant toute la durée de l'épreuve.

Sujet de 8 pages, celle-ci comprise.

Vérifiez que vous disposez bien de la totalité des pages du sujet en début d'épreuve et signalez tout problème de reprographie le cas échéant.

---

1. Qu'est-ce que la régularisation et quel est son intérêt pour l'apprentissage ? (**2 points**)

Correction :
La régularisation a pour objectif de maîtriser la complexité du modèle afin d'éviter le sur-apprentissage. La régularisation est souvent mise en œuvre par l'ajout, dans la fonction d'erreur à minimiser, d'une pénalité pour la complexité du modèle, mais peut prendre d'autres formes (voir le cours d'introduction).

---

2. Arbres de décision, algorithme Iterative Dichotomiser (IC3) : expliquez le terme « gain d'information sur l'attribut a ». Comment est utilisée cette quantité par l'algorithme ? (**2 points**)

Correction :
Gain d'information sur l'attribut a : $GI(S;a) = H(S) - \sum_i p_i H(S_i)$. C'est la décroissance de l'entropie avant et après le découpage du nœud courant sur l'attribut a. Il permet de choisir dans la construction de l'arbre quel est l'attribut testé dans le nœud courant (c'est celui qui produit un gain d'information maximal).

---

3. Arbres de décision : quel est le critère optimisé par l'algorithme CART en classification ? (**1 point**)

Correction :
C'est l'index (ou l'impureté) de Gini : la vraisemblance qu'un élément du nœud est incorrectement labellisé par un tirage aléatoire qui respecte la loi statistique de la cible estimée dans le nœud. D'autres mesures d'impureté sont aussi utilisées (entropie, erreur de classification).

---

4. Quel est le défaut principal des méthodes de type *bagging* ? Comment on y remédie ? (**1 point**)

Correction :
EPour *bagging* on calcule plusieurs modèles G_i qu'on agrège par la moyenne (en régression) ou par vote majoritaire (en classification). Le problème le plus important est que les estimateurs G_i ne sont pas indépendants car ils sont calculés sur des échantillons qui se recouvrent (tirage avec remise). Pour remédier à cette situation, les forêts aléatoires essayent de réduire la corrélation à l'aide d'une étape

supplémentaire de randomisation.

---

5. On construit un séparateur linéaire selon l'algorithme SVM pour le problème de classification suivant (2 classes en 2 dimensions) :

$$Classe_1 = \{(0,1),(1,0),(1,1)\}$$

$$Classe_2 = \{(0,-1),(-1,0),(-1,-1)\}$$

a) Calculez la marge. b) Quels sont les vecteurs de support ? (**2 points**)

Correction :
a) La marge : $\sqrt{2}$)
b) Vecteurs de support : $\{(0,-1),(-1,0),(0,1),(1,0)\}$

---

6. Donnez la définition d'un noyau défini positif. Quel est l'intérêt de ces noyaux ? (**1 point**)

Correction :
Un noyau $K(x,y)$ est défini positif si quels que soient les vecteurs $\{x_1,...,x_n\}$, la matrice de Gramm associée $[K(x_i,x_j)]_{ij}$ est positive définie. Ils garantissent le fonctionnement de l'astuce des noyaux.

---

7. Apprentissage profond (**2 points**)
   - En quoi les réseaux de neurones convolutifs qui ont remporté le challenge ImageNet 2012 différaient-ils de ceux utilisées dans les années 1980 ?
   - Quels sont les deux facteurs principaux qui ont permis leur succès en 2012 ?

Correction :

Apprentissage profond (**1 point** 0.5+0.5)
   - Pas de différence fondamentale dans l'architecture macroscopique / CNN des années 80 : alternance de couches convolution / pooling puis couches complètement connectées. Simplement des réseaux avec plus de couches et plus de neurones / filtres par couches (60000 vs 60 M paramètres)
   - ImageNet (10^6 images annotées parmi 1000 classes) + GPU

8. Réseaux convolutifs (**2 points**)
   On considère les images de chiffres manuscrits de la figure 1, à laquelle on va appliquer une opération de convolution de filtre : $\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$, suivie d'une opération de non linéarité de type `ReLU`.

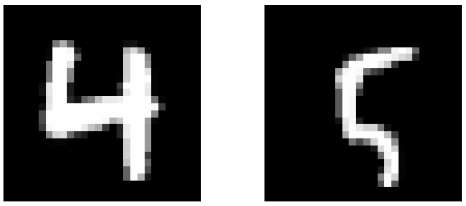


FIGURE 1 – Images de chiffres manuscrits.

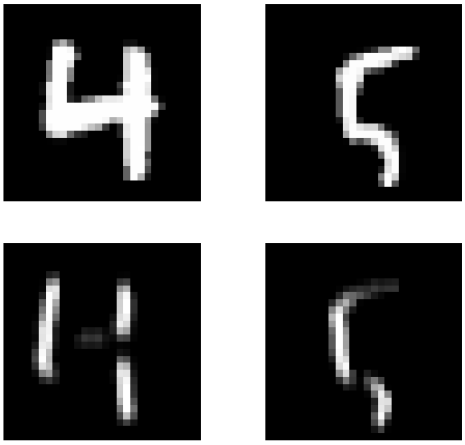
   — Comment s'interprète le filtre et quelle forme va-t-il détecter ? Quel va être l'effet de la non linéarité ?

   — Donner la forme de l'image résultant après convolution et `ReLU`.

Correction :
Réseaux convolutifs (**2 points** 1+1)

   — C'est un filtre de Sobel qui va détecter des contours verticaux. La `ReLU` ne va conserver que les valeur positives, et donc ne conserver que la présence de forts contours (orientés).

   — résultat :



9. Agrégation (pooling) (**2 points**)
   On considère le signal 1D suivant, issu du traitement d'un signal numérique 1D par un filtre de convolution : $c(t) = [0.1; 0.2; 1.0; 0.8; 0.7; 0.6; 0.9; 0.1; 0.2; 0.0]$.
   On va appliquer une étape de max pooling à $c(t)$, avec une taille de région de pooling de 5, et un décalage (stride) de 5.
   - Quelle est la taille du signal après pooling ? Justifier. Écrire le résultat pour le signal $c(t)$, et le signal $c(t-1)$ (on suppose que $c(t) = 0$ pour $t \notin \{0; 9\}$).
   - Quelle est la propriété ici illustrée de l'opération de pooling ? Quel l'intérêt de l'intégrer dans les réseaux convolutifs pour des tâches de classification ?

Correction :
Agrégation (pooling) (**2 points** 1+1)

- La taille du vecteur à la sortie du pooling est de 2, car on a des régions de taille 5 qui ne se superposent pas (car le stride est de 5), et que le signal d'entrée est de taille 10. Signal après pooling dans les deux cas : $p(t) = [1.0; 0.9]$
- Ça illustre la propriété d'invariance aux translations locales de l'opération de max pooling. Ceci est intéressant dans les réseaux convolutifs, en particulier pour des tâches de classification où la position très précise des features est moins importante que leur présence (et le pooling permet de significativement réduire le nombre de paramètres à apprendre).

---

10. Transfert et Fine-tuning (**2 points**)

On souhaite appliquer un réseau convolutif profond pour une tâche de classification d'images de feuilles. On dispose d'une base dont certaines images sont montrées à la figure 2a), le volumes des données annotées étant précisé à la figure 2b).

a) Exemples d'images

Leaf type	Number of images
Lanceolate	568
Oval	554
Acicular	612
Linear	439
Oblong	374
Reniform, kidney-shaped	580
Cordate, heart-shaped	379
Palmate leaf	361

b) Classes et données annotées

FIGURE 2 – Base de données d'images de feuilles avec 8 classes.

Décrire une méthodologie pour entraîner le modèle d'apprentissage profond.

Correction :
Transfert et Fine-tuning (**2 points**)
Pour avoir des bonnes performances, on va utiliser des réseaux convolutifs modernes (AlexNet, VGG, ResNet). Ces réseaux ayant beaucoup de paramètres (au moins plusieurs dizaines de millions), il va être d'impossible d'entraîner ces modèles `from scratch` sur la base ici considérée (qui contient relativement peu d'exemples annotés). On va donc utiliser un réseau pré-entraîné sur ImageNet, supprimer la couche de sortie correspond aux 1000 classes de cette base, et ajouter une couche de transfert vers les 8 classes de la base cible. On peut envisager de fine-tuner, avec un learning rate plus faible pour les paramètres transférés.

---

11. Métrique d'ordonnancement (**3 points**)
On considère un problème de recherche par le contenu dont l'objectif est d'ordonner des documents par ordre de pertinence décroissant par rapport à une requête. Chaque document est annoté comme étant pertinent ($\oplus$) ou non pertinent ($\ominus$) par rapport à la requête.
Deux systèmes de décision sont proposés, et on trie les exemples par ordre décroissant de score du modèle.

Les scores triés pour le premier modèle sont rassemblés dans le tableau suivant, avec les labels $\oplus/\ominus$ correspondant :

S1	1.0	0.8	0.7	0.6	0.5	0.4	0.3	0.25	0.2	0.1
labels	$\oplus$	$\ominus$	$\ominus$	$\ominus$	$\ominus$	$\ominus$	$\ominus$	$\oplus$	$\ominus$	$\ominus$

Les scores triés pour le second modèle sont rassemblés dans le tableau suivant, avec les labels $\oplus/\ominus$ correspondant :

S2	0.9	0.82	0.74	0.61	0.53	0.42	0.33	0.25	0.21	0.17
labels	$\ominus$	$\ominus$	$\ominus$	$\oplus$	$\oplus$	$\ominus$	$\ominus$	$\ominus$	$\ominus$	$\ominus$

On rappelle que la métrique AUC (Area Under Curve) correspond à l'aire sous la courbe ROC, avec : $\Delta_{AUC}(y, y^*) = 1.0 - AUC = \frac{1}{N_+ N_-} \sum_{i \in \oplus} \sum_{j \in \ominus} \frac{(1 - y_{ij})}{2}$, avec :

$$y_{ij} = \begin{cases} +1 & \text{si } d_i \prec_y d_j \text{ (d_i est classé avant d_j dans la liste ordonnée)} \\ -1 & \text{si } d_i \succ_y d_j \text{ (d_i est classé après d_j)} \end{cases}$$

— Rappeler l'interprétation du loss $\Delta_{AUC}(y, y^*)$ défini ci-dessus.

— Calculer la métrique AUC pour les deux systèmes de décision. Quel est le meilleur système selon le critère AUC ?

La métrique AP (Average Precision) correspond à l'aire sous la courbe Précision-Rappel. Les courbes Précision-Rappel pour les deux systèmes S1 et S2 sont montrées dans la figure 3.

— Expliquer pourquoi le système S1 est meilleur que le système S2 au sens de la métrique AP.

— Quel métrique (AP *vs* AUC) vous semble la plus intéressante dans un problème de recherche par le contenu, *e.g.* ordonner des pages web par ordre décroissant de pertinence par rapport à une requête par mots clés ?

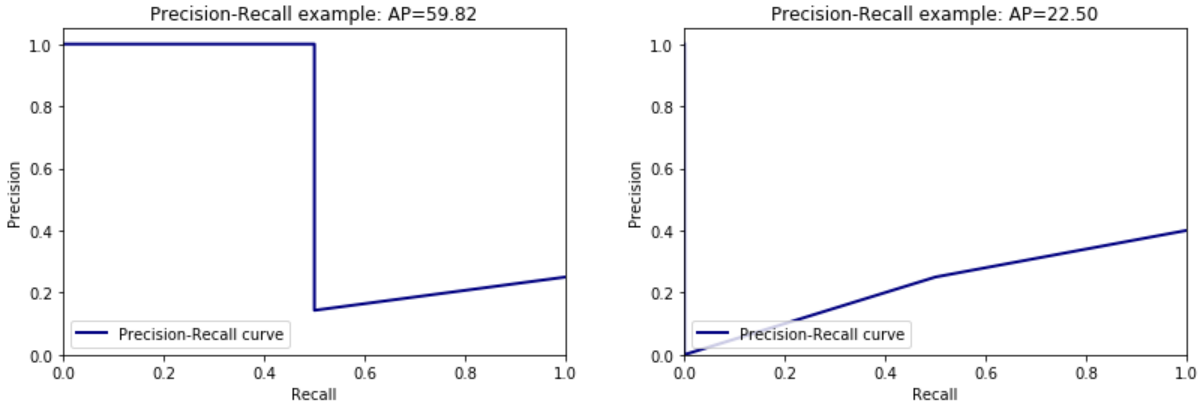
Correction :
Métrique d'ordonnancement (**3 points** 0.5 + 1 +0.5 +1)

— C'est le nombre de paires échangées par rapport au ranking idéal (tous les + devant les -)

— Ici le nombre de paires échangées est de $\frac{6}{16} = \frac{3}{8}$, donc l'AUC est de $1 - \frac{3}{8} =$ 62.5% pour les deux systèmes.

— Avec la métrique AP, le système est meilleur car l'élément au somment de la liste est + : le loss Δ_{AP} pénalise plus les échanges en somment de liste qu'en fin.

— L'AP est plus adapté pour des problèmes de recherche d'informations où ce qui est intéressant est de retrouver les + au somment du classement (et qu'il y a un fort déséquilibre entre le nombre de + N_+ et le nombre - N_-). Dans l'exemple



a) Système de décision S1 b) Système de décision S2

FIGURE 3 – Courbes Précision-Rappel et Average Precision (AP).

choisi, si non retourne les 3 premiers éléments, on n'a aucun exemple pertinent avec le système S2, alors qu'on en a un (classé en plus en premier) avec le système S1.