Apprentissage et reconnaissance (GIF-4101 / GIF-7005)
Département de génie électrique et de génie informatique
Automne 2016

UNIVERSITÉ LAVAL

EXAMEN FINAL

Instructions : – Une feuille aide-mémoire recto verso manuscrite est permise ;
– Durée de l'examen : 2 h 50.

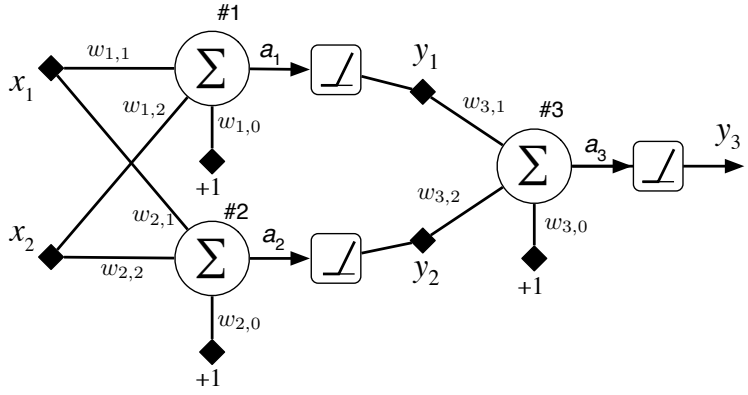
Pondération : – Cet examen compte pour $35\,\%$ de la note finale ;
– La note est saturée à $100\,\%$ si le total des points avec bonus excède cette valeur.

Question 1 (20 points sur 100)

Soit un réseau de neurones avec comme fonction de transfert la fonction ReLU (*rectified linear unit*) :

$$\mathrm{f}_{\mathrm{ReLU}}(a) = \max(0,a) = \begin{cases} 0 & \text{si } a < 0 \\ a & \text{autrement} \end{cases} .$$

(8) (a) Supposons que l'on veut entraîner le réseau suivant à trois neurones avec activation ReLU, pour résoudre le problème du OU exclusif (XOR).



Données du XOR :

$$\mathbf{x}^1 = [0\ 0]^T \quad r^1 = 0$$
$$\mathbf{x}^2 = [0\ 1]^T \quad r^2 = 1$$
$$\mathbf{x}^3 = [1\ 0]^T \quad r^3 = 1$$
$$\mathbf{x}^4 = [1\ 1]^T \quad r^4 = 0$$

Déterminez les poids et biais des trois neurones du réseau permettant d'obtenir une erreur de classement nulle sur ce problème.

(12) (b) Supposons maintenant que l'on fait un apprentissage en ligne (mise à jour pour une instance à la fois) selon l'erreur quadratique moyenne, par une rétropropagation des erreurs. Développez les équations pour mettre à jour les poids sur la couche de sortie de neurones utilisant la fonction d'activation ReLU.

Conseil : Ne vous formalisez pas de la discontinuité de la fonction ReLU pour $a = 0$, vous pouvez formuler la fonction de la dérivée en deux morceaux, comme fait plus haut dans l'énoncé.

Question 2 (17 points sur 100)

Soit le jeu de données $\mathcal{X} = \{\mathbf{x}^t, r^t\}_{t=1}^4$ présenté ici-bas.

$$\mathbf{x^1} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathbf{x^2} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \mathbf{x^3} = \begin{bmatrix} 0{,}8 \\ 1 \end{bmatrix}, \quad \mathbf{x^4} = \begin{bmatrix} 1{,}5 \\ 1{,}25 \end{bmatrix},$$
$$r^1 = -1, \qquad r^2 = -1, \qquad r^3 = 1, \qquad r^4 = 1.$$

Supposons que l'on veut classer ces données avec un classifieur de type Séparateur à vastes marges (SVM) utilisant un noyau linéaire ($K(\mathbf{x},\mathbf{x}') = \langle \mathbf{x},\mathbf{x}' \rangle = (\mathbf{x}')^T\mathbf{x}$), sans marge floue.

(7) (a) Tracez dans votre cahier de réponse les données du jeu $\mathcal{X}$, l'hyperplan séparateur que l'on obtient avec un SVM sur ces données, les marges géométriques maximales correspondantes et encerclez les données agissant comme vecteurs de support.

(10) (b) Donnez les valeurs correspondant aux poids $\mathbf{w}$ et biais w_0 du discriminant linéaire maximisant les marges géométriques tracées en (a).

Question 3 (15 points sur 100)

Soit les réseaux de neurones profonds de type autoencodeur, tel que présenté en classe.

(5) (a) La première phase de l'entraînement d'un autoencodeur est dite non-supervisée. Expliquez en quoi consiste cette phase d'entraînement, en précisant les éléments suivants :
- — Topologie du réseau, en précisant ce à quoi correspond la partie encodeur et la partie décodeur ;
- — Données utilisées pour l'entraînement, incluant valeurs cibles en sortie ;
- — Critère d'erreur utilisé pour l'entraînement ;
- — Fonctionnement de l'algorithme d'entraînement (procédure d'apprentissage et poids modifiés à chaque étape).

(5) (b) La deuxième phase de l'entraînement d'un autoencodeur est dite de raffinement (*fine-tuning)*. Expliquez en quoi consiste cette deuxième phase, en précisant les éléments suivants :
- — Topologie du réseau, en référant aux parties présentées à la question précédente ;
- — Données utilisées pour l'entraînement, incluant valeurs cibles en sortie ;
- — Critère d'erreur utilisé pour l'entraînement ;
- — Fonctionnement de l'algorithme d'entraînement (procédure d'application et poids modifiés à chaque étape).

(5) (c) Une force des réseaux profonds est la capacité d'apprendre une représentation à partir des données. Expliquez à quoi correspond cette représentation dans un autoencodeur.

Également, il est courant d'effectuer des transferts de représentation en apprentissage profond. De quelle façon peut-on faire cela à l'aide d'un autoencodeur ? Expliquez cela en termes concrets, en disant comment on peut transférer la représentation d'un autoencodeur apprise sur un certain problème vers un autre problème distinct, mais relaté, en présentant les différentes étapes nécessaires pour y arriver.

Question 4 (48 points sur 100)

Répondez aussi brièvement et clairement que possible aux questions suivantes.

(3) (a) Expliquez en quoi diffère l'objectif principal d'un apprentissage de modèles discriminatifs de classement relativement à l'apprentissage de modèles génératifs.

(3) (b) Expliquez comment on peut interpréter la fonction sigmoïde dans un contexte de classement binaire avec densités normales.

(3) (c) Expliquez l'intérêt d'utiliser une régularisation l_1, tel qu'utilisé avec LASSO, comparativement à une régularisation l_2, utilisée dans une régression d'arête (*ridge regression*).

(3) (d) Dans le développement des équations pour l'apprentissage des paramètres du SVM, expliquez pourquoi on passe de la forme primale à la forme duale relativement à l'optimisation du modèle.

(3) (e) Dans les méthodes à noyau, expliquez le sens de la mesure effectuée avec la fonction noyau entre deux instances.

(3) (f) Présentez les principales similarités et différences entre les SVM avec noyau gaussien et les réseaux de neurones RBF.

(3) (g) Indiquer la taille des vecteurs correspondant aux composantes principales obtenues lors d'une analyse en composantes principales (ACP) **à noyau**.

(3) (h) Expliquez pourquoi le critère d'arrêt hâtif (*early stopping*) est particulièrement intéressant et important pour favoriser la généralisation dans l'entraînement d'un perceptron multicouche.

(3) (i) Indiquez les trois conditions principales favorisant l'émergence des réseaux profonds dans les dix dernières années.

(3) (j) Expliquez en quoi consiste la méthode d'entraînement dropout avec les réseaux profonds.

(3) (k) Expliquez ce que représente le taux d'erreur bayésien optimal en classement.

(3) (l) Expliquez le principal avantage associé à l'utilisation de codes à correction d'erreurs pour le traitement des données à plusieurs classes avec un ensemble de classifieurs binaires (à deux classes), comparativement à des approches de type *un contre tous* ou de *décisions par paires*.

(3) (m) Expliquez la principale différence entre le *Bagging* et le *Random subspace*.

(3) (n) Quelle est la condition minimale exigée d'un classifieur de type *weak learner* pour l'utiliser avec l'algorithme AdaBoost ?

(3) (o) Lors de planification d'expérimentations, dans quel contexte la stratégie *un facteur à la fois* peut-elle être utilisée ?

(3) (p) Dans un graphique de courbe ROC, quel doit être la tracé de la courbe pour s'assurer que la méthode évaluée fait mieux que le hasard en tous points d'opération ?

Question 5 (10 points bonus)

Supposons que la fonction A nous permet d'effectuer l'apprentissage d'une paramétrisation θ d'un classifieur sur le jeu de données $\mathcal{X}$, en utilisant γ comme hyperparamètre de la méthode d'apprentissage :

$$\theta = A(\mathcal{X},\gamma).$$

On dispose également d'une fonction d'erreur $E(\mathcal{X}|\theta)$ pour évaluer la performance du classifieur entraîné $\mathrm{h}(\cdot|\theta)$ sur le jeu de données $\mathcal{X}$.

Supposons maintenant que pour un problème donné nous avons sous la main deux jeux de données, soit un ensemble d'entraînement $\mathcal{S}$ et un ensemble de test $\mathcal{T}$. On veut tester L valeurs différentes d'hyperparamètres données dans l'ensemble $\Gamma = \{\gamma_1,\gamma_2,\ldots,\gamma_L\}$. Pour évaluer la performance de chaque paramétrisation, on veut effectuer une validation croisée à K plis. Donnez le pseudo-code correspondant une méthodologie valide, permettant d'évaluer chacune de ces paramétrisations γ_i et de sélectionner la paramétrisation γ^* qui apparaît optimale en termes d'erreur en généralisation. Tentez d'être aussi formel que possible dans l'énonciation mathématique de votre pseudo-code.

Vous pouvez utiliser la fonction $\mathrm{Part}(\mathcal{X},K)$, qui fait une partition aléatoire stratifiée de l'ensemble de données $\mathcal{X}$ en K sous-ensembles distincts :

$$\{\mathcal{X}_1,\mathcal{X}_2,\ldots,\mathcal{X}_K\} = \mathrm{Part}(\mathcal{X},K).$$