Nom de l'étudiant: ______________________________

FACULTE DES ARTS ET DES SCIENCES
DEPARTEMENT D'INFORMATIQUE ET DE RECHERCHE OPERATIONNELLE

TITRE DU COURS: **Algorithmes d'apprentissage**
SIGLE DU COURS : **IFT6266 A99**

NOM DU PROFESSEUR: Yoshua Bengio

DATE DE L'EXAMEN INTRA A99: 1 novembre 1999 HEURE: 9h30 - 11h25
SALLE: Z-325

DIRECTIVES PEDAGOGIQUES: - Aucune documentation permise
- Répondre directement sur le questionnaire.
- Soyez brefs et précis dans vos réponses.

1. Quelle est la différence fondamentale entre ce qu'on fait dans le domaine des algorithmes d'apprentissage et ce qu'on fait en optimisation?

2. Nous avons défini **l' induction** comme le mécanisme qui nous amène d'un ensemble de données à une *fonction* (dont on voudrait minimiser l'erreur espérée sur tout le domaine des données). Vapnik a défini **la transduction** comme un mécanisme alternatif où on cherche seulement à trouver une *réponse* à un point test x donné, en minimisant l'erreur de généralisation espérée à ce point (i.e., on ne cherche pas la fonction $f(x)$ pour tout x, on cherche la valeur $\hat{y}$ associée à un x particulier). Parmi les algorithmes d'apprentissage que nous avons vu en classe, nommez-en qui sont plus apparentés à de la transduction qu'à de l'induction.

3. Expliquez dans vos mots (pas nécessairement formellement) le concept de **capacité** et comment cette notion est reliée à la différence entre **l'erreur d'apprentissage** et **l'erreur de généralisation espérée**.

4. Qu'est-ce que le **"dilemme biais-variance"**? vous pouvez l'expliquer en mots ou bien en formules (et un dessin pourrait aussi aider).

5. Supposons que nous connaissions le modèle $P(Y|X)$ de la **probabilité conditionnelle de classes** discrètes Y associées à un vecteur d'entrée X pour faire de la reconnaissance des formes. Prouvez que si on doit choisir une classe, celle qui minimise l'erreur de classification espérée est celle qui a la plus grande probabilité conditionnelle.

6. Supposons qu'on vous donne un ensemble de données de taille fixe l, et que vous **devez** construire un **classifieur linéaire**. Vous avez au départ des entrées dans $\mathcal{R}^n$, mais vous trouvez que vous n'arrivez pas

à classifier suffisamment bien les exemples, *aussi bien à l'entrainement qu'au test hors-échantillon (les deux erreurs sont très élevées et comparables).* Un de vos collaborateurs vous présente alors deux options: (a) éliminer certaines des variables d'entrée, (b) rajouter des nouvelles variables d'entrée qui seraient des fonctions fixes des variables existantes (e.g. des produits). Parmi ces deux, quelle approche suivriez-vous, et pourquoi?

7. Nous avons vu qu'on peut améliorer la généralisation obtenue d'une régression linéaire en rajoutant un terme de pénalité proportionnel à la norme des paramètres. On peut faire de même pour des réseaux de neurones. Supposons qu'on vous suggère d'utiliser un mécanisme semblable pour spéficiquement réduire le nombre d'entrées qui ont effectivement un impact significatif sur la sortie (sans faire une recherche combinatoire dans l'espace des sous-ensembles d'entrées). Comment pourriez vous utiliser un mécanisme de pénalisation pour y arriver?

8. Considérons un problème de **classification binaire**, en utilisant un réseau de neurones, entrainé par le critère de moindre carrés, avec une sortie désirée $Y = 1$ quand la classe correcte est 1, et une sortie désirée $Y = 0$ quand la classe correcte est 0, pour une entrée X. Montrez que la fonction qui minimise l'erreur quadratique en espérance (sur les données) est effectivement la distribution conditionnelle $f(x) = P(Y = 1|X = x)$.

9. Suggérez une méthode pour choisir le nombre de prototypes ("clusters") qu'un algorithme de clustering (apprentissage non-supervisé) comme K-means devrait chercher. N.B. l'erreur de reconstruction (qui est minimisée par K-means) ne serait pas appropriée car plus on a de prototypes et plus basse sera l'erreur de reconstruction aussi bien sur un ensemble d'apprentissage qu'un ensemble de test.

10. On vous donne un ensemble de données avec des entrées X et des sorties Y et on vous demande d'appliquer les techniques de réseaux de neurones pour pouvoir prédire le genre de valeur qu'on attend de Y pour un X donné. Les X représentent l'image d'une route vue par un conducteur et les Y représente une décision de tourner plus ou moins à droite ou à gauche pour rester sur la route (e.g. Y représente un angle entre -90^o et 90^o). Cependant, on vous avertit que l'on s'attend à ce que, pour un X donné, il existe plusieurs valeurs de Y compatibles avec ce X (à cause des intersections et autres embranchements de la route), mais que la valeur intermédiaire de ces Y n'est pas du tout désirable. Le but est **d'entrainer un robot à conduire une voiture** (en apprenant à imiter les choix d'un conducteur humain). Vous ne devriez pas utiliser le critère de moindres carrés car le robot pourrait choisir d'aller tout droit (et donc de foncer dans un mur) quand il devrait en fait aller plutôt à droite ou plutôt à gauche (e.g. une route avec un embranchement en "Y"). Pendant l'entrainement, on pourrait avoir pour la même image parfois la réponse "a gauche" (e.g. $Y = -45^o$) et parfois la réponse "à droite" (e.g. $Y = 45^o$), mais il ne faut pas répondre entre les deux (e.g., tout droit $Y = 0^o$). Or, le critère de moindre carrés nous amène à chercher $E[Y|X]$

(qui est donc la moyenne entre les différentes réponses possibles pour un X donné). Comment sortir de ce pétrin? Suggestion: considérez comme critère la maximisation de la vraisemblance conditionnelle de $P(Y|X)$ selon un certain modèle, et utilisez un critère autre que celui qui associe à X une simple normale pour Y (i.e., celui qui serait équivalent au critère de moindres carrés). Expliquez dans les grandes lignes comment calculer le gradient pour entrainer le modèle, et comment interpréter les sorties pour éviter de foncer dans un mur.