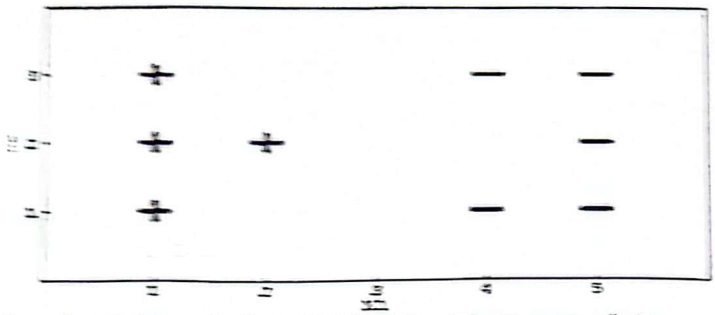
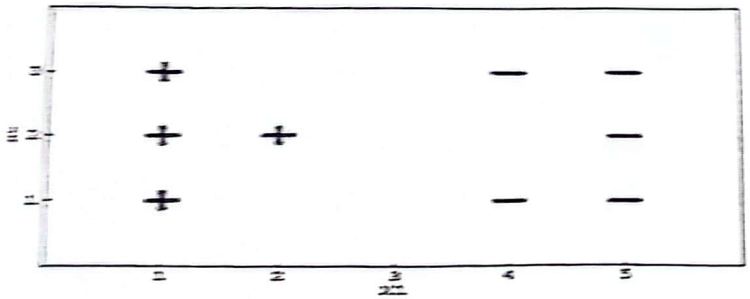
Examen (Durée : 2 h30mn)

1. (3.5 points) Supposons que nous utilisons une SVM linéaire (c'est-à-dire, sans noyau), avec une valeur de C élevée, et que nous disposons de l'ensemble de données suivant



Tracer la frontière de décision de la SVM linéaire. Donner une brève explication.

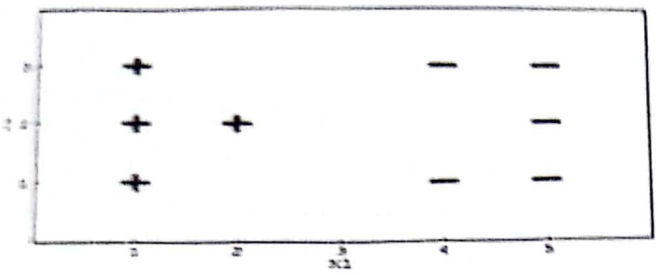
2. (3.5 points) Dans l'image suivante, entourez les exemples de sorte que lorsque cet exemple est retiré de l'ensemble d'entraînement et que la SVM est réentraînée, nous obtenons une frontière de décision différente que lorsque nous entraînons sur l'échantillon complet.(Il n'est pas nécessaire de fournir une preuve formelle, mais donner une ou deux phrases explication.)



3. (3.5 points) Supposons que, au lieu de SVM, nous utilisions une régression logistique régularisée pour apprendre le classificateur. C'est-à-dire,

$$(w,b) = \arg\min_{w\in\mathbb{R}^2, b\in\mathbb{R}} \frac{\|w\|^2}{2} - \sum_i 1[y^{(i)}=0] \ln \frac{1}{1+e^{(w\cdot x^{(i)}+b)}} + 1[y^{(i)}=1] \ln \frac{e^{(w\cdot x^{(i)}+b)}}{1+e^{(w\cdot x^{(i)}+b)}}.$$

Dans l'image suivante, entourez les points de sorte que lorsque cet exemple est retiré de l'ensemble d'entraînement et que la régression logistique régularisée est lancée, nous obtenons une frontière de décision différente que lorsque nous entraînons avec la régression logistique régularisée sur l'échantillon complet.



4. (1 point) Si l'erreur d'entraînement augmente avec le nombre d'époques, quel est le problème possible avec le processus d'apprentissage? Entourez la lettre de votre choix.

(i) La régularisation est trop faible et le modèle est en sur-ajustement

(ii) La régularisation est trop élevée et le modèle est en sous-ajustement

(iii) La taille de pas est trop grande

(iv) La taille de pas est trop petite

5. (1.5 points) Pourquoi la fonction *softmax* est souvent utilisée pour les problèmes de classification (par exemple dans les réseaux de neurones)?

6. (1.5 points) Considérons les ensembles de données suivants :

- $X_{\text{train}} = (x^{(1)}, x^{(2)}, ..., x^{(m_{\text{train}})}), Y_{\text{train}} = (y^{(1)}, y^{(2)}, ..., y^{(m_{\text{train}})})$
- $X_{\text{test}} = (x^{(1)}, x^{(2)}, ..., x^{(m_{\text{test}})}), Y_{\text{test}} = (y^{(1)}, y^{(2)}, ..., y^{(m_{\text{test}})})$

Vous voulez normaliser vos données avant de former votre modèle. Lesquelles des propositions suivantes sont vraies? (Entourez toutes celles qui s'appliquent.)

(i) La moyenne et la variance normalisées calculées sur l'ensemble d'entraînement et utilisées pour entraîner le modèle doivent être utilisées pour normaliser les données de test.

(ii) Les données de test doivent être normalisées avec leur propre moyenne et leur propre variance avant d'être introduites dans le réseau lors du test, car la distribution de test peut être différente de la distribution d'entraînement.

(iii) La normalisation des entrées a un impact sur le paysage de la fonction de perte.

(iv) En imagerie, tout comme pour les données, la normalisation consiste à soustraire la moyenne des entrées et à multiplier le résultat par l'ecart type.

7. (1.5 points) Laquelle des propositions suivantes est vraie, étant donné la vitesse d'apprentissage optimale?

(i) La descente de gradient par lots est toujours garantie de converger vers le optimum global d'une fonction de perte.

(ii) La descente de gradient stochastique est toujours garantie de converger vers le optimum global d'une fonction de perte.

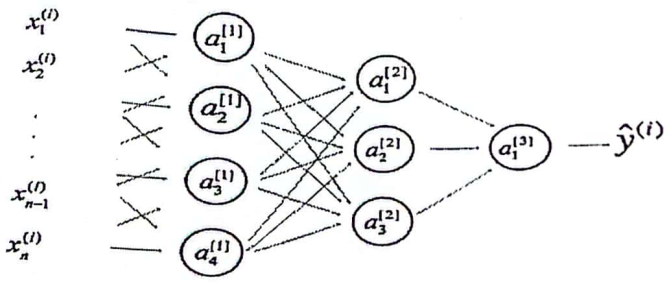
(iii) Pour les fonctions de perte convexes (c'est-à-dire avec une forme de bol), la descente de gradient par lots est garantie de converger vers le optimum global tandis que la descente de gradient stochastique n'est pas garantie de le faire.

(iv) Pour les fonctions de perte convexes (c'est-à-dire avec une forme de bol), la descente de gradient stochastique est garantie de converger vers le optimum global tandis que la descente de gradient par lots n'est pas garantie de le faire.

(v) Pour les fonctions de perte convexes (c'est-à-dire avec une forme de bol), la descente de gradient par lots et la descente de gradient stochastique convergeront toutes les deux vers l'optimum global.

(vi) Pour les fonctions de perte convexes (c'est-à-dire avec une forme de bol), ni la descente de gradient stochastique ni la descente de gradient par lots ne sont garanties de converger vers le optimum global.

8. (1 point) Soit le réseau neuronal entièrement connecté à deux couches suivant. Toutes les activations sont des sigmoïdes et votre optimiseur est la descente de gradient stochastique. Vous initialisez tous les poids et biais à zéro et propagez une entrée $x \in R^{n\times 1}$ dans le réseau.



Quelle est la sortie $\hat{y}$?

(i) -1
(ii) 0
(iii) 0.5
(iv) 1

9.(1 point) Considérés Considérez le modèle défini dans la question (8) avec des paramètres *initialisés avec des zéros.* $W^{[1]}$ *désigne la matrice de poids de la première couche.* Vous

propagez un lot d'exemples et ensuite rétropropager les gradients et mettre à jour les paramètres. Laquelle des affirmations suivantes est vraie?

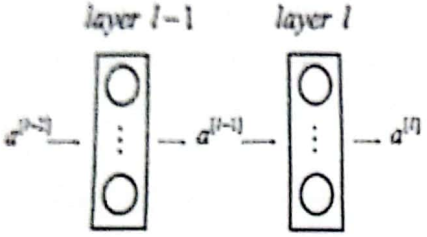
(i) Les entrées de $W^{[1]}$ peuvent être positives ou négatives

(ii) Les entrées de $W^{[1]}$ sont toutes négatives

(iii) Les entrées de $W^{[1]}$ sont toutes positives

(iv) (iv) Les entrées de $W^{[1]}$ sont toutes nulles

10. (1 point) Considérez les couches l et l-1 dans un réseau neuronal entièrement connecté:

Réseau	Les équations de propagation pour ces couches sont :
	$z^{[l-1]} = W^{[l-1]}a^{[l-2]} + b^{[l-1]}$ $a^{[l-1]} = g^{[l-1]}(z^{[l-1]})$ $z^{[l]} = W^{[l]}a^{[l-1]} + b^{[l]}$ $a^{[l]} = g^{[l]}(z^{[l]})$

Laquelle des propositions suivantes est vraie ? L'initialisation de Xavier assure que:

(v) $\text{Var}(W^{[l-1]})$ est la même que $\text{Var}(W^{[l]})$.

(vi) $\text{Var}(b^{[l]})$ est la même que $\text{Var}(b^{[l-1]})$.

(vii) $\text{Var}(a^{[l]})$ est la même que $\text{Var}(a^{[l-1]})$, à la fin de l'entraînement.

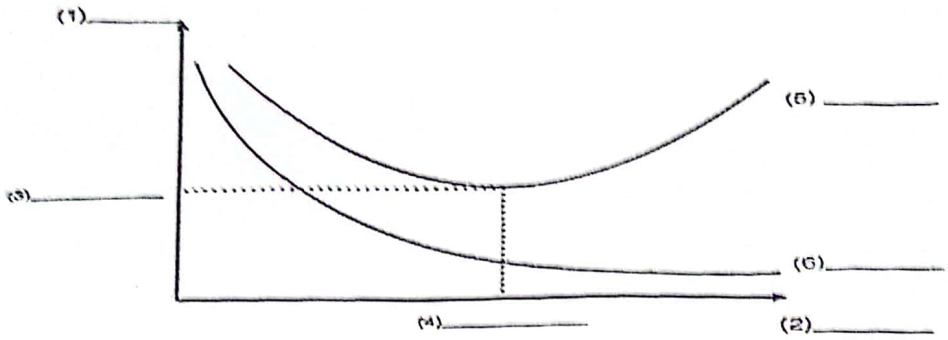
(viii) $\text{Var}(a^{[l]})$ est la même que $\text{Var}(a^{[l-1]})$, au début de l'entraînement.

11. (1 point) Vous faites une descente de gradient de lot complet en utilisant l'ensemble d'entraînement entier (pas de descente de gradient stochastique). Est-il nécessaire de mélanger les données d'entraînement ? Expliquez votre réponse.

12. (1 point) L'augmentation des données est souvent utilisée pour augmenter la quantité de données dont vous disposez. Devriez-vous appliquer l'augmentation des données au jeu de test? Expliquez pourquoi.

13. (1 point) Vous souhaitez entraîner un réseau neuronal entièrement connecté avec 5 couches cachées, chacune avec 10 unités cachées. L'entrée est de dimension 20 et la sortie est un scalaire. Quel est le nombre total de paramètres entraînables dans votre réseau?

14. (2 points) Compléter les blancs et expliquer ce phénomène



15. (2 points) Quel effet auront les opérations suivantes sur le biais et la variance de votre modèle :

Operations	Variance	Biais
Augmenter le nombre de données d'entraînement		
Augmenter la complexité du modèle		
Augmenter le nombre d'itérations d'entraînement		

Rappelons que :

Biais = $E[f(x)] - f(x)$ et Variance = $E[(f(x) - E[f(x)])^2]$ où $f(x)$ est la fonction de prédiction du modèle, $E[f(x)]$ est la valeur attendue des prédictions du modèle pour toutes les données d'entraînement, et x est une donnée d'entraînement particulière.