

NOM :

**MI 203 : Apprentissage Automatique**  
**EXAMEN**  
**11/03/2020**

**Consigne générale :**

**Vous avez droit à tous vos documents et notes de cours. Il n'est pas nécessaire de traiter l'intégralité du sujet : l'examen est noté sur 20 alors que l'ensemble contient 27 points. Répondez directement sur le sujet.**

## **PARTIE 1 : Apprentissage supervisé et Évaluation**

### Q1 (3pts) : sur-apprentissage

Les assertions suivantes sont-elles vraies ou fausses ?

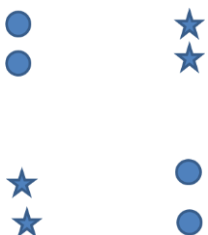
On peut **diminuer** le sur-apprentissage en **augmentant**:

- la profondeur des arbres de décision.
- le nombre de voisins à observer dans l'algorithme des k-plus proches voisins
- le coefficient C dans l'algorithme des SVM
- la variance d'un noyau de type RBF dans les SVM
- le nombre de couches d'un réseau de neurones
- le nombre d'exemples d'apprentissage

### Q2 (3pts) : validation croisée

On veut choisir par validation croisée le meilleur algorithme entre le 1-plus proche voisin et un SVM linéaire.

- Rappeler la démarche générale de validation croisée pour la sélection de modèle.
  
  
  
  
  
  
  
- Calculer les valeurs d'estimation d'erreur par « *leave-one-out* » pour la configuration de points ci-dessous et les deux algorithmes. Quel est le meilleur algorithme ?



- Même question pour la configuration de points ci-dessous.



Q3 (8 pts) : Arbres de décision

- Quelles mesures sont communément utilisées pour estimer l'hétérogénéité d'un ensemble de données ?

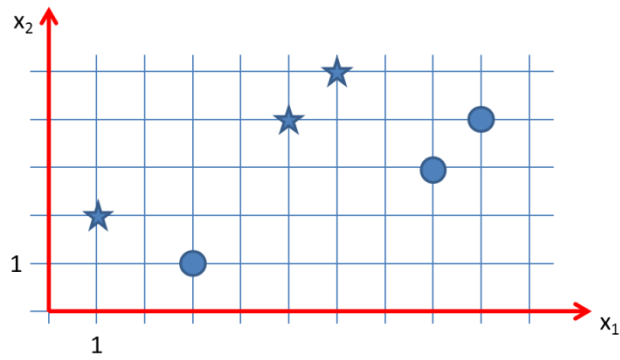
Entropie  
Critère de Gini  
Risque empirique  
Variance

- La profondeur d'un arbre de décision entraîné peut être plus grande que le nombre d'exemples d'apprentissage. Justifier.

Vrai                  Faux

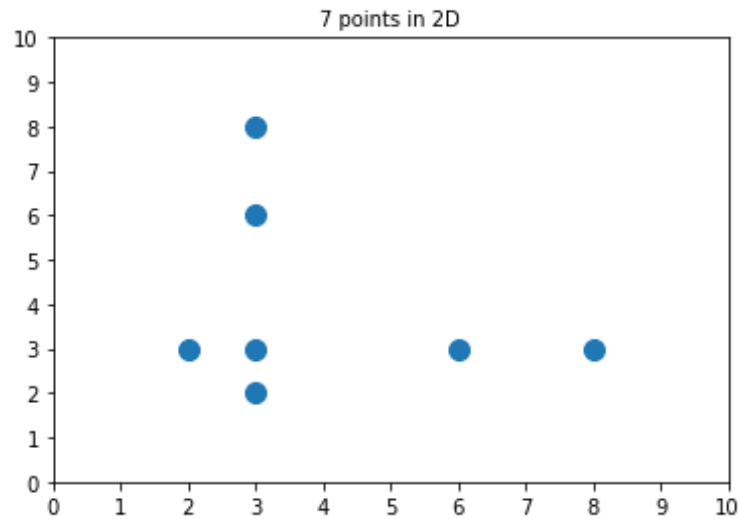
- Le nombre de feuilles d'une forêt aléatoire entraînée peut être plus grand que le nombre d'exemples d'apprentissage. Justifier.

Vrai                  Faux



- On veut construire un arbre de décision pour classer la distribution ci-dessus :
  - Quelle est l'entropie en base 2 de la distribution initiale ?
  - Parmi les tests binaires de la forme  $x_1 \geq 7$  ou  $x_2 \geq 4$ , lequel serait celui choisi en premier ? Justifier.

### PARTIE 3 : K means (4pts)



- Quel serait selon vous les  $K=3$  clusters qui devraient se dégager sur les 7 points ci-dessous ?
- Même question avec  $K=2$
- Donner une bonne et une mauvaise initialisation des clusters avec  $K=3$  et avec  $K=2$

## PARTIE 4 : Réseaux de neurones

### Réseaux de neurones classiques :

Notation : Le produit scalaire de 2 vecteurs est symbolisé par un point.

Q3.1 (1pts) : si  $x$  est une valeur (1D), prouver que  $\text{relu}(x) + \text{relu}(-x) = |x|$  (la valeur absolue).

Q3.2 (1pts) : en déduire la valeur de  $\text{relu}((1,0).x) + \text{relu}((-1,0).x) + \text{relu}((0,1).x) + \text{relu}((0,-1).x) = \|x\|_1$  c'est-à-dire *la norme 1* quand  $x$  est un vecteur (2D).

Q3.3 (1pts) : Indépendamment, quelle est la valeur de  $f(x) = 2 \times \text{relu}(1 - \|x\|_1)$  en  $(0,0)$  et en  $(a,b)$  avec  $a, b$  entiers relatifs.

Q3.4 (1pts) : indiquer la structure de  $f$  (en terme de réseau de neurones : nombre de neurones, nombre de couches etc...) – Aider vous de la question 3.2.

Q3.5 (1pt) : soit  $(i,j)$  un couple d'entier relatif, peut-on écrire  $g_{i,j}(x) = f(x - (i,j))$  comme un réseau de neurones avec biais et poids dépendant de  $i$  et  $j$  ?

Q3.6 (1pt) : soit  $B = ((i,j), (k,l), (u,v))$ , 3 couples d'entiers relatifs, on considère

$$g_B = g_{i,j}(x) + g_{k,l}(x) + g_{u,v}(x) - 1$$

que vaut cette fonction en  $(a,b)$  couple d'entiers relatifs quelconque ?

*Pour votre culture : Nous avons construit un élément d'une famille de réseau  $g_Q$  tel que pour toute base  $Q$  entier, le réseau  $g_Q$  vaut 1 en  $(i,j)$  si  $(i,j)$  est dans  $Q$  et -1 sinon. On peut ainsi sur-apprendre toutes les bases entières. Notez que le nombre de paramètre pour apprendre une base est « grand ». Limiter le nombre de paramètres permet ici de se prémunir d'un sur-apprentissage.*

### **Réseau de neurones convolutifs :**

Q3.5 (3 pts) : écrivez le code pytorch qui permet de calculer le réseau suivant sur l'image suivante (alternativement vous pouvez faire les calculs à la main pour réaliser à quel point c'est utile de pouvoir le coder)

l'image d'entrée de taille 3x5 (1 channel) :

```

2  1 -1 -1  1
0 -2 -2  1  1
1  0  2 -1 -1

```

Le réseau est constitué de

- 1 couche de convolution (1 neurone) avec un noyau 2x2 (stride 1 et pas de padding) avec relu et avec les poids :

```

-1 -2
2  1

```

- 1 couche de pooling 2x2 (stride 2 et pas de padding)
- 1 couche de convolution (1 neurone) avec un noyau 1x2 (stride 1 et pas de padding) avec les poids

```

1 -2

```

Aide : Après la première couche la taille sera 2x4 – n'oubliez pas le relu – après le pooling la taille sera seulement de 1x2 puis de 1x1 après la dernière couche. La sortie finale du réseau est 9 pour ceux qui voudraient faire les calculs.