Examen 1ère session (2h) - 16 mai 2019

Rappels : **Tous documents autorisés.** Les calculatrices et autres appareils électroniques doivent être éteints et rangés. Le barème (sur 30) n'est donné qu'à titre indicatif.

Exercice 1 *Cours (2 points)*
On considère un ensemble fini $\mathcal{U}$ d'éléments : $\mathcal{U} = \{x_1, x_2, \ldots, x_n\}$.

Q. 1. Donner la fonction caractéristique d'un sous-ensemble E de $\mathcal{U}$.

Q. 2. Donner la fonction caractéristique de l'ensemble vide.

Q. 3. Soit E et F deux sous-ensembles de U définis par leurs fonctions caractéristiques χ_E et χ_F respectivement. Donner les fonctions caractéristiques des ensembles $E \cup F$, $E \cap F$ et E^c.

Exercice 2 *Arbres de décision (4pts)*
On considère la base d'apprentissage suivante :

Température	Saison	Pluie	Activité
élevée	printemps	non	pas de sortie
élevée	été	oui	sortie
basse	printemps	non	sortie
basse	printemps	oui	pas de sortie

Q. 1. En utilisant l'algorithme de construction d'arbres de décision vu en cours, et en détaillant les étapes et calculs réalisés, construire un arbre de décision permettant de prédire l'activité connaissant les valeurs pour les 3 attributs de description (température, saison et pluie).
Rappels : quelques valeurs de logarithme en base 2 : $\log(\frac{1}{2}) = -1$, $\log(\frac{1}{3}) = -1.58$, $\log(\frac{2}{3}) = -0.58$, $\log(\frac{1}{4}) = -2$ et $\log(\frac{3}{4}) = -0.42$.

Exercice 3 *Apprentissage non-supervisé (8pts)*

Q. 1. Montrer que la distance de Manhattan, définie par $d : \mathbb{R}^p \times \mathbb{R}^p \longrightarrow \mathbb{R}^+$ et $d(x, y) = \sum_{j=1}^{p} |x_j - y_j|$ est bien une mesure de distance.

Q. 2. Rappeler la définition d'une partition $P = \{C_1, \ldots, C_K\}$ d'un ensemble $\mathcal{X}$ en K sous-ensembles.

Q. 3. Soit la base d'apprentissage composée de 10 points de $\mathbb{R}$: $\mathcal{X} = \{1, 2, 2, 5, 6, 7, 11, 12, 13, 14\}$. En détaillant les étapes et les calculs réalisés (calculs d'inertie, centroïdes,...), appliquer l'algorithme des k-moyennes sur $\mathcal{X}$ en prenant $k = 3$ et en utilisant la distance euclidienne. L'initialisation est faite en prenant les points 1, 2 et 5 et le critère d'arrêt choisi est $\epsilon = 5$. En cas d'égalité, un point est affecté au cluster de plus petit numéro.

Q. 4. Toujours avec la base précédente, on considère maintenant les 2 partitions en 2 clusters suivantes : $P_1 = \{\{1, 2, 2\}, \{5, 6, 7, 11, 12, 13, 14\}\}$ et $P_2 = \{\{1, 2, 2, 5, 6\}, \{7, 11, 12, 13, 14\}\}$. Donner la partition ($P_1$ ou P_2) la plus intéressante selon l'index de Dunn.

Exercice 4 *Descente de gradient, fonctions de coût et évaluation (5pts)*
On part des notations classiquement utilisée en cours en apprentissage supervisé, une donnée décrite vectoriellement est notée $\mathbf{x} \in \mathbb{R}^d$ et associée à une étiquette y. Une base de données entière est notée : $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_i, y_i), \ldots, (\mathbf{x}_N, y_N)\}$. On se limite dans cet exercice à l'étude des modèles linéaires : le but sera donc d'apprendre un vecteur $\mathbf{w}$.

Q. 1. Donner les dimensions du vecteur $\mathbf{w}$.

Q. 2. L'idée générale d'une fonction de coût est de mesurer les erreurs d'un système d'apprentissage automatique. Par exemple, dans le cas de classification, on pourrait simplement compter le nombre d'erreurs de classification. Cependant, cette dernière proposition n'est pas utilisée en pratique : pourquoi ? Citer deux fonctions de coût classiques (noms des fonctions et expressions mathématiques). Expliquer les points forts et faibles de ces fonctions de coût.

Q. 3. Rappeler le but et le principe général de l'algorithme de descente de gradient et décliner cet algorithme sur l'une des fonctions de coût précédentes. Expliquer le principe d'une descente de gradient stochastique ET le principe de la descente de gradient classique. Expliquer clairement ce qui est fourni en entrée et ce qui est obtenu en sortie de l'algorithme.

Q. 4. Comment évaluer le modèle appris ? Expliquer la procédure et rappeler brièvement le principal piège à éviter.

Q. 5. Discuter le coût machine et les performances des algorithmes de descente de gradient par rapport à l'algorithme des k-plus proches voisins.

Exercice 5 *Codage de caractéristiques (5pts)*

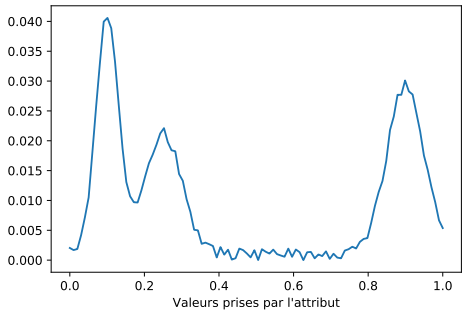
Nous nous intéressons à des algorithmes d'apprentissage automatique de type perceptron, traitant des attributs numériques. Dans ce cadre spécifique, nous souhaitons faire face aux cas de figure suivants. Expliquer comment coder l'information dans chaque situation en rappelant à chaque fois la dimension de la représentation des données. Justifier brièvement votre choix.

Q. 1. Codage d'une caractéristique catégorielle comprenant 20 modalités.

Q. 2. Codage d'un film décrit par ses acteurs. Parmi les 1000 acteurs présents dans la base, 60% apparaissent dans moins de 3 films et nous considérons qu'il n'est pas possible de faire des statistiques sur ces acteurs. Evidemment, nous souhaitons tout de même tirer parti de toutes ces informations.

Q. 3. Codage d'un problème présentant trois attributs catégoriels binaires et trois attributs numériques standard.

Q. 4. Codage de la caractéristique numérique présentant la distribution suivante :



Exercice 6 *Détection d'auteur (6pts)*

Dans cet exercice, on souhaite mettre en œuvre ce qui a été vu en 3i026 afin de construire un modèle pour la détection de messages frauduleux.

Initialement, un texte est une séquence de mots : $\mathbf{t} = \{w_1, \ldots, w_n\}$. Un corpus est un ensemble de textes : $\mathcal{C} = \{\mathbf{t}_1, \ldots, \mathbf{t}_N\}$. Evidemment, chaque texte a une longueur différente.

Q. 1. Le premier problème consiste à ramener les textes à analyser vers un format tableau. Cette opération est effectuée en deux temps :

1. une phase de construction du dictionnaire qui consiste à extraire les mots uniques et à les associer à un indice. Entrée : $\mathcal{C}$, Sortie : $\mathcal{D} = \{w_1 : 1, w_2 : 2, \ldots, w_d : d\}$. La sortie est donc une table de hashage. Donner l'**implémentation python** (fonction : code et signature complète) d'un algorithme prenant en entrée une liste de textes (`string`) et rendant le dictionnaire $\mathcal{D}$.

2. Donner une **implémentation python** de la méthode de construction de la matrice de données X à partir du corpus. Il s'agit d'une matrice de comptage : chaque ligne correspond à un document, chaque colonne à un mot, la case donne le nombre d'occurrences du mot dans le document. Vous indiquerez aussi quelles sont les dimensions de la matrice X.

Q. 2. En considérant que les documents sont des mails associés à un expéditeur, proposer une architecture de classification : quels sont les problèmes auxquels vous devrez faire face ?

Q. 3. Après entraînement, vous obtenez un système où chaque nouveau document obtient un score entre -1 et 1 associé à tous les auteurs de la base. Imaginer plusieurs cas d'usage de ce système. [Il s'agit d'une question exploratoire : vous pouvez envisager des applications lointaines ; par contre, il faut que le système soit sensé du point de vue technique].

Annexes : mini-doc Python

— Séparer une string en une liste de mots :

```
str = "je ne copie pas sur mon voisin"
li = str.split() # ["je", "ne", "copie", "pas", "sur", "mon", "voisin"]
```

— Les dictionnaires en python :

```
dico = dict() # constructeur
dico[cle] = valeur # création ou mise à jour d'une clé
val = dico[cle] # retourne une valeur ou une exception si la clé n'existe pas
val = dico.setdefault(cle, valeur_par_defaut) # idem, sans l'exception
```