1NSI	Algorithmique k plus proches voisins	TP – **Algorithme des k plus proches voisins**

Exercice 1 (débranché) : Classification k NN

Données pour l'apprentissage (volontairement réduite pour procéder à la main)

numero_departement	nom_departement	prefecture	nom_commune	codes_postaux	latitude(°)	longitude(°)
77	Seine-et-Marne	Melun	Fontainebleau	77300	48,4	2,7
77	Seine-et-Marne	Melun	Melun	77000	48,533333	2,666667
74	Haute-Savoie	Annecy	Annecy	74000	45,9	6,116667
91	Essonne	Evry	Evry	91000	48,633333	2,45

Les coordonnées GPS de Dammarie-les-Lys sont :
Latitude : 48,515088°
Longitude : 2,634702°

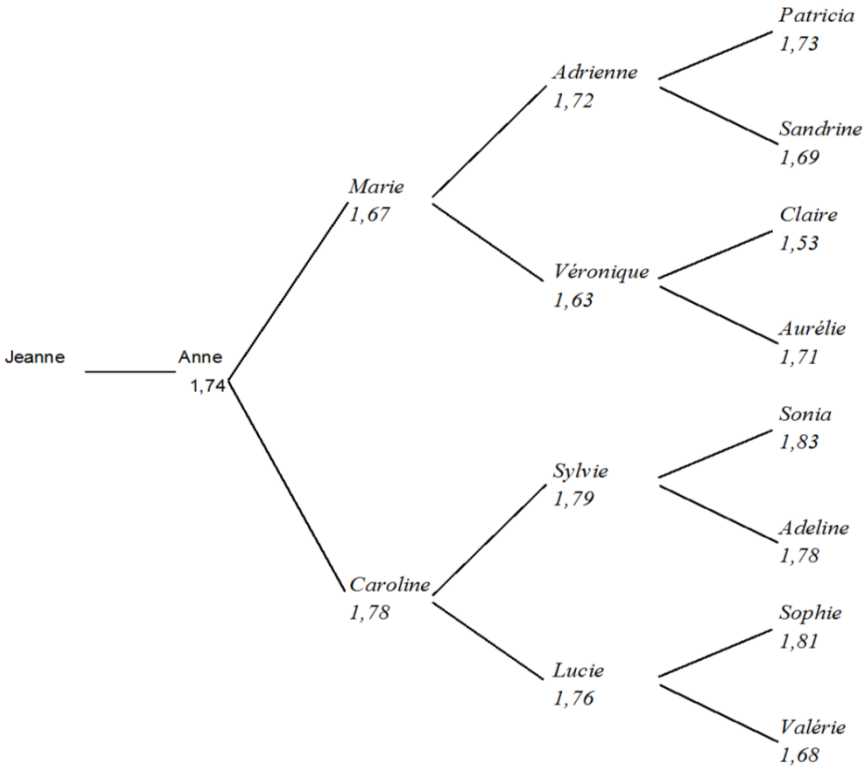
On rappelle que le rayon moyen de la terre mesure environ 6371 kilomètres et que la longueur d'un arc de cercle est égale au produit du rayon par l'angle au centre.

1) Distance Dammarie-Annecy
   a) Calculer l'écart en kilomètres lié à la différence de latitude entre Dammarie-les-Lys et Annecy.
   b) Calculer l'écart en kilomètres lié à la différence de longitude entre Dammarie-les-Lys et Annecy.
   c) En utilisant le théorème de Pythagore, en déduire une approximation de la distance entre Dammarie les lys et Annecy.
2) Calculer les distances entre Dammarie-les-Lys et chacune des villes apparaissant dans les données d'apprentissage.
3) A quel département la ville de Dammarie-les-Lys appartient-elle selon un algorithme basé sur les 3 plus proches voisins s'appuyant sur les données d'apprentissage ci-dessus ? (détailler la démarche pour justifier)

Exercice 2 (débranché) : Régression k NN

L'arbre généalogique de Jeanne est donné ci-contre avec les tailles à l'âge adulte.
Selon le principe de l'exemple 2 du cours, quelle estimation de la taille de Jeanne à l'âge adulte est proposée par une régression à partir :

(1) d'un algorithme 1 NN ?
(2) d'un algorithme 3 NN ?
(3) d'un algorithme 7 NN ?
(4) d'un algorithme 5 NN ? (avec la convention qu'en cas d'égalité des distances, le premier voisin trouvé est conservé)



Exercice 3 (débranché) : formalisation de l'algorithme k NN

On dispose :

- de données d'apprentissages présentées sous la forme d'une liste de couples (prédicteurs, étiquette)
- d'une fonction distance qui à partir des prédicteurs de deux objets renvoie la distance entre les deux objets avec les propriétés mathématiques usuelles pour une distance

Ecrire en langage naturel, en pseudo-code ou en Python une fonction kNN qui :

- prend en entrées : un objet observé pour lequel on dispose des prédicteurs, une liste de données d'apprentissage et entier naturel k non nul et inférieur ou égal à la longueur de la liste des données d'apprentissage
- renvoie en sortie la liste des étiquettes des k plus proches voisins de l'objet observé parmi les données d'apprentissage.
  En cas d'égalité de distance on prendra arbitrairement le premier rencontré.
  Au cas où les prédicteurs d'un élément des données d'apprentissage correspondraient exactement à ceux de l'objet observé, on retournera une liste contenant k fois l'étiquette de cet élément.

Exercice 4 (TP informatique) : Classification k NN

Le fichier « apprentissage_departement.csv » a été constitué à partir de données extraites du site : https://www.data.gouv.fr/fr/datasets/listes-des-communes-geolocalisees-par-regions-departements-circonscriptions-nd/

1) Ecrire une fonction en Python qui lit ce fichier et constitue une liste de données d'apprentissage exploitable par la fonction kNN de l'exercice 3, constitué de couples ((latitude, longitude), département) (deux prédicteurs et une étiquette). On pourra s'appuyer sur la bibliothèque « `csv` » et les dictionnaires.
2) En s'inspirant de l'exercice 1, écrire une fonction `distance` calculant la distance entre deux villes à partir des couples (latitude, longitude) de chacune de ces deux villes. On rappelle que la fonction « `sqrt` » de la bibliothèque « `math` » permet de calculer une racine carrée.
3) A l'aide des fonctions de l'activité préparatoire et de l'exercice 3, écrire une fonction prenant en entrée un entier non nul k et le couple (latitude, longitude) des coordonnées GPS d'une commune et effectue une prédiction sur le département auquel appartient cette commune à l'aide d'un algorithme basé sur les k plus proches voisins figurant dans les données du fichier « apprentissage_departement.csv ».
4) Tester cette fonction sur plusieurs communes de votre choix (les coordonnées GPS pourront être cherchées sur internet). On effectuera des tests avec des communes qui apparaissent dans le fichier « apprentissage_departement.csv » et d'autres qui n'y apparaissent pas. Les tests seront effectués avec k = 7 et k = 6500.

Exercice 5 (TP informatique) : Régression k NN

Le fichier « apprentissage_temperatures2016.csv » a été constitué à partir de données extraites du site : https://www.data.gouv.fr/fr/datasets/temperature-quotidienne-regionale-depuis-janvier-2016/

On souhaite estimer la température moyenne pour une région et un jour donné.

1) Ecrire une fonction en Python qui lit ce fichier et constitue une liste de données d'apprentissage exploitable par la fonction kNN de l'exercice 3, constitué de couples ((région, jour), température moyenne) où jour désigne le numéro du jour dans l'année (compris entre 1 et 366).
2) Définir une fonction `distance` entre des prédicteurs donnés sous la forme d'un couple (région, jour), qui vaut 400 (ou n'importe quelle valeur strictement supérieure à 366) si les régions sont différentes et qui vaut le nombre de jours d'écart entre les jours de l'année sinon (peu importe l'année, attention l'écart entre le 30 décembre et le 2 janvier vaut donc 4).
3) A l'aide des fonctions des exercices 0 (activité préparatoire) et 3, écrire une fonction prenant en entrée un entier non nul k et un couple (région, jour) qui renvoie une estimation de la température qu'il fera dans la région pour ce jour par un algorithme k NN.
4) Tester l'algorithme sur les jours de votre choix et pour différentes valeurs de k. Que peut-on penser des choix k = 366 ? k = 4000 ?
5) (bonus) Ecrire des fonctions permettant d'obtenir le numéro du jour dans l'année à partir de la date ou vice versa, puis les utiliser pour améliorer le programme précédent en travaillant à partir des dates.