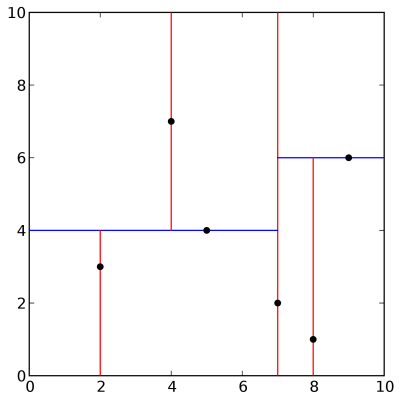
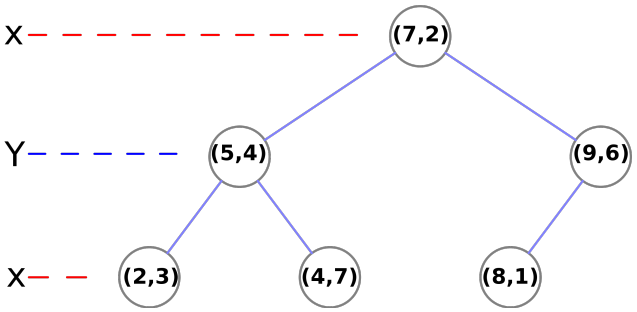
TD10 - Apprentissage supervisé

Exercice 1

Le langage de programmation est libre : C ou OCaml. On se propose de coder la construction d'un arbre kd. L'exemple du cours reproduit ci-dessous peut servir de test.





Question 1. Écrire une fonction `dist` qui calcule la distance euclidienne entre deux points.

Question 2. Écrire une fonction `swap` qui échange deux nœuds d'un arbre.

Question 3. Écrire une fonction `find_median` qui renvoie le point médian d'un ensemble de points.

Question 4. Écrire une fonction `make_tree` qui construit un arbre kd.

Question 5. Tester votre code avec le fichier `td10-data.txt` mis à votre dispostion en ligne.

Question 6. Comment utiliser la connaissance d'un arbre k-d pour déterminer le point le plus proche d'un point donné ? Par exemple, le point le plus proche du point de coordonnées $(9, 2)$ est $(8, 1)$, situé à la distance euclidienne $\sqrt{2}$.

Question 7. Écrire une fonction `nearest` qui reçoit un arbre kd, un point fixé et qui détermine le point de l'arbre le plus proche de ce dernier. Calculer en même temps le nombre de nœuds visités avant de trouver le plus proche point.

Question 8. Écrire une fonction `knn` qui détermine la classe d'appartenance d'un point fixé en mettant en œuvre l'algorithme des k plus proches voisins.

Question 9. Construire la matrice de confusion associée et déterminer le taux d'erreur associé.

Exercice 2

On considère le problème de classification suivant : des objets sont répartis en deux classes - rond et croix - selon deux attributs prenant des valeurs continues dans $[0, 1]^2$. On pourra représenter ces objets par des points (ronds ou croix) dans le plan.

Question 1. Peut on appliquer l'algorithme ID3 à de telles données afin d'établir un modèle permettant de les classer ?

Question 2. Proposer une méthode naïve permettant d'appliquer ID3 à un tel jeu de données. Est-elle raisonnable ?

Question 3. On souhaite classer les données à l'aide d'un arbre de décision obligatoirement binaire. Donner une méthode moins naïve que celle de la question 2 permettant d'atteindre ce but.

Question 4. Quel résultat obtiendrait-on avec cette méthode sur la figure 1 ? Sur la figure 2 ? Est-ce satisfaisant ?

Question 5. Proposer une méthode permettant de mieux classer les données dans le cas où celles-ci sont réparties de manière similaire à celles de la figure 2. Argumenter quant à sa terminaison.

Question 6. Dessiner la frontière entre les deux classes que la méthode de la question 5 calculerait sur la figure 3. Est-ce satisfaisant ?

Question 7. Comment transformer le jeu de données de la figure 3 de sorte à pouvoir tracer une frontière bien plus simple à l'aide de la méthode décrite à la question 5 ?

Question 8. Et dans le cas de la figure 4, que peut-on faire ?

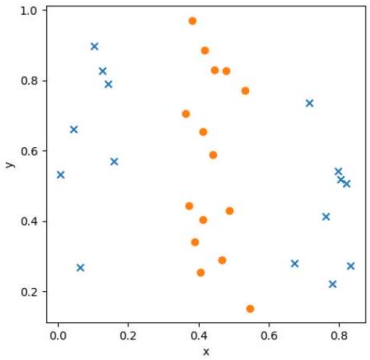


FIGURE 1

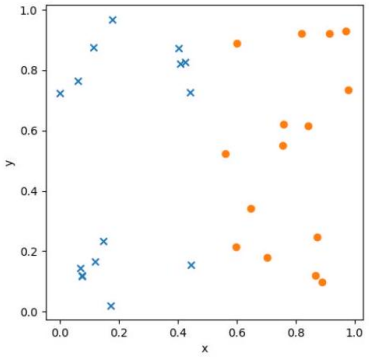


FIGURE 2

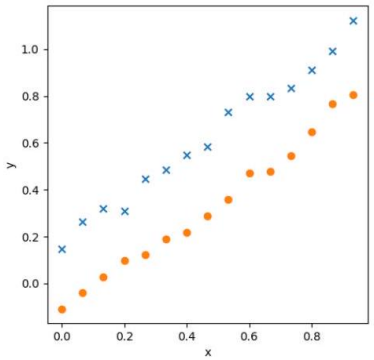


FIGURE 3

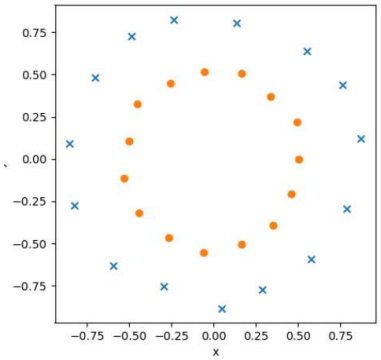


FIGURE 4