Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique
Université de Sfax
Institut Préparatoire aux Etudes
d'Ingénieurs de Sfax

IPEIS

وزارة التعليم العالي والبحث العلمي
جامعة صفاقس
المعهد التحضيري للدراسات الهندسية
بصفاقس

Année universitaire : 2020-2021

Devoir de contrôle semestre 2

Matière : informatique

Classe : MP2, PC2, PT2 et BG2 — Nombre de pages : 2

Date : Mars 2021 — Durée : 1h

Présentation Générale

L'apprentissage automatique supervisé permet d'élaborer des programmes capables d'apprendre automatiquement à partir d'un ensemble de données (**dataset**) comportant des valeurs d'observations et les décisions qui leur sont associées. Il a ainsi pour objectif de produire un modèle capable de prédire la décision à prendre pour des nouvelles valeurs d'observations.

Par exemple, on peut donner au programme d'apprentissage un ensemble de données contenant les observations relatives à des patients (**tension** artérielle, **âge** du patient et présence d'une **tachycardie** sinusoïdale) et expliquer lesquels ont un risque élevé de crise cardiaque (**décision** = 1) et lesquels ont un risque très faible (**décision** = 0). Comme illustré dans le tableau suivant, les lignes représentent les valeurs des observations relatives à un patient et la dernière colonne représente la décision finale.
Une fois l'apprentissage terminé, à partir des observations d'un nouveau patient, le programme, appelé aussi **classifieur**, devra déterminer automatiquement la décision à prendre (0 ou 1).

tension	âge	tachycardie	décision
110	50	1	0
119	36	0	0
82	72	0	1
81	70	0	1
56	50	1	1

Table 1 - Exemple de données d'apprentissage.

PROBLEME

Soit la base de données relationnelle intitulée 'classifieurs.db' contenant la description des données d'apprentissage avec les classifieurs automatiques créés autour de ces données, représentée par le schéma relationnel suivant :

- **DataSet** (ds_id, ds_name, nb_instances, format, ds_description)
  La table DataSet décrit les données d'apprentissage, avec les colonnes :
  - ds_id : identifiant du dataset (entier), *clé primaire*.
  - ds_name : nom du dataset (chaîne de caractères).
  - nb_instances : nombre de lignes du dataset (entier).
  - format : le format des données du dataset (chaine de caractères).
  - ds_description : le sommaire du dataset (chaine de caractères).

- **Classifieur** (cls_id, cls_description, error_rate, language, ds_id)
  La table Classifieur décrit un classifieur automatique construit à partir d'un dataset, avec les colonnes :
  - cls_id : identifiant du classifieur (chaine de caractère), *clé primaire*.
  - cls_description : description du classifieur (chaine de caractères).
  - error_rate : un nombre réel entre 0 et 1 qui décrit le pourcentage des données où les décisions prédites par le classifieur sont différentes de la réalité.

- language : nom du langage de programmation utilisé pour implémenter le classifieur (chaine de caractère).
- ds_id : identifiant du dataset utilisé pour l'apprentissage du classifieur, *clé étrangère*.

- **Method** (m_name, category, m_description)
  La table Method décrit les méthodes utilisées dans le domaine de l'apprentissage automatique pour la construction des classifieurs, avec les colonnes :
  - m_name : le nom de la méthode (chaine de caractères), *clé primaire*.
  - category : la catégorie de l'algorithme (chaine de caractères).
  - m_description : la description de l'algorithme (chaine de caractères).

- **Combine** (cls_id, m_name, description)
  La table Combine décrit les méthodes de classification utilisées dans chaque classifieur, de clé primaire (cls_id, m_name), avec les colonnes :
  - cls_id : identifiant du classifieur (chaine de caractère), *clé étrangère*.
  - m_name : le nom de l'algorithme (chaine de caractère), *clé étrangère*.
  - description : stratégie d'intégration de l'algorithme dans le classifieur (chaine de caractères).

Partie 1 :

On dispose d'un dictionnaire **dict_M** contenant des informations relatives aux méthodes utilisées dans le domaine d'apprentissage, où :

- chaque clé est le nom de la méthode
- chaque valeur est une liste contenant respectivement la catégorie et la description.

1. Ecrire les instructions python permettant de :
   - Importer le module sqlite3.
   - Se connecter à la base de données 'classifieurs.db'.
   - Créer le curseur **cur** d'exécution.
   - Créer la table Method en exprimant les contraintes mentionnées ci-dessus.
   - Remplir la table Method à partir du dictionnaire **dict_M**.
   - Tracer la courbe dont les abscisses sont les identifiants des datasets et les ordonnées sont les taux d'erreur moyen des classifieurs associés.

Partie 2 : SQL

Exprimer en SQL les requêtes suivantes :

2. Supprimer les classifieurs implémentés avec le langage de programmation 'Pascal'.
3. Donner les identifiant des datasets pour lesquels il existe au moins un classifieur avec un taux d'erreur < 0.3.
4. Donner les noms des datasets contenant le plus grand nombre d'instances.
5. Donner les identifiants et les noms des datasets où tous les classifieurs sont implémentés en 'Python'.
6. Donner les identifiants des datasets jamais utilisés par des classifieurs.
7. Donner le nombre de classifieurs dont le taux d'erreur est supérieur à la moyenne des taux d'erreurs de tous les classifieurs.
8. Donner pour chaque dataset le nombre de méthodes de classification utilisées.
9. Donner les informations des datasets associés aux classifieurs produisant le taux d'erreur minimal.
10. Donner les identifiants et les noms des datasets autours desquels il existe au moins 3 classifieurs implémentés en 'Java'.
11. Modifier dans la table DataSet les formats 'txt' par 'docx'.