|  | **École nationale Supérieure d'Informatique (ESI), Alger**  **2CSSIQ2, 2CSSIQ3 et 2CSSIL2 2021/2022**  **Contrôle Final (CF)**  **Machine Learning (ML)** |
| --- | --- |

| **Jeudi 16 juin 2022 - [9h, 11h] - Partie 1**  *Documents non autorisés*  Il faut répondre sur cette feuille (pages 1 et 2) et la rendre même si elle est vide  *Au cas d'insuffisance des copies, recopier les réponses sur une feuille d'examen séparée de celle des exercices* | |
| --- | --- |
| **Nom + Prénom :** ………………………………………………………….. | ◯ SQ2 ◯ SQ3 ◯ SL2 |

**QCM. Test de connaissance (6pts : 30mn)**

Pour chaque question, un choix erroné annule un autre correct. La note minimale par question est 0.

1) Pour chaque critère, sélectionner l'algorithme de classement qui la possède entre naive Bayes (NB), les machines à vecteurs de support (SVM), les forêts aléatoires (FA), les arbres de décision (AD) et la régression logistique (RL) :

| **Critère** | **NB** | **SVM** | **FA** | **AD** | **RL** |
| --- | --- | --- | --- | --- | --- |
| **Utilise un algorithme d'optimisation** | ☐ | ☐ | ☐ | ☐ | ☐ |
| **Algorithme (original) déterministe** | ☐ | ☐ | ☐ | ☐ | ☐ |
| **Moins sensible au problème de sur-apprentissage (sans utiliser d'autres mécanismes )** | ☐ | ☐ | ☐ | ☐ | ☐ |

2) Classer les modèles suivants du plus interprétable au moins interprétable :

| **Modèle** | **Moins interprétable** | **Moyen** | **Plus interprétable** |
| --- | --- | --- | --- |
| **Régression logistique** | ◯ | ◯ | ◯ |
| **Arbre de décision** | ◯ | ◯ | ◯ |
| **Forêts aléatoires** | ◯ | ◯ | ◯ |

3) Nous avons un dataset de 10 caractéristiques. Nous voulons sélectionner 7 caractéristiques. Dans ce cas, ordonner les algorithmes de sélection d'attributs en se basant sur leurs temps d'exécution :

| **Algorithme** | **Moins de temps** | **Moyen** | **Plus de temps** |
| --- | --- | --- | --- |
| **ANOVA** | ◯ | ◯ | ◯ |
| **Forward Selection** | ◯ | ◯ | ◯ |
| **Backward Elimination** | ◯ | ◯ | ◯ |

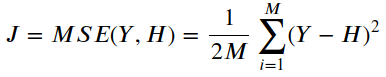
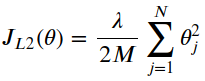
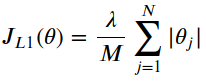
4) Nous voulons entraîner trois modèles de régression linéaire : un sans régularisation, un avec régularisation L2 et un avec régularisation L1. Initialement les trois modèles ont les mêmes paramètres θ. Nous avons utilisé le même dataset [X, Y].

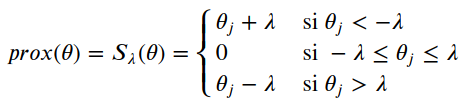
| X = | ⎡  ⎣ | 2 | 1 | ⎤  ⎦ | Y = | ⎡  ⎣ | 7 | ⎤  ⎦ | [θ0, θ1, θ2] = [1, 2, 1] |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 1 |  | 2 |  |
|  | 1 | 2 |  | 5 |  |

Etant donné 𝜆=1, 𝛼=1, la fonction du coût est MSE et pour L1 nous utilisons soft-thresholding, sélectionner les bonnes choix concernant le coût global (J) et les nouveaux paramètres :

| **Critère** | **Valeur** | **Reg-Lin** | **Reg-Lin + L2** | **Reg-Lin + L1** |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| **Coût**  **J** | 5/6 | ☐ | ☐ | ☐ |
| 5/4 | ☐ | ☐ | ☐ |
| 5/3 | ☐ | ☐ | ☐ |
| 11/6 | ☐ | ☐ | ☐ |
| 5/2 | ☐ | ☐ | ☐ |
| 11/4 | ☐ | ☐ | ☐ |
| 15/4 | ☐ | ☐ | ☐ |
|  |  |  |  |  |
| **Nouveau**  **θ** | 0 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 |
| 1/3 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 |
| 1/2 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 |
| 2/3 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 |
| 1 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 |
| 4/3 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 |
| 2 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 | ☐ θ0 ☐ θ1 ☐ θ2 |
|  | | | | |
| ***Dans le cas où une valeur n'existe pas dans les choix, écrivez la ici :*** | | | | |
| **Coût J** |  | J= | J= | J= |
| **θ** |  | θ0= θ1= θ2= | θ0= θ1= θ2= | θ0= θ1= θ2= |

**Rappel :**



| **Jeudi 16 juin 2022 - [9h, 11h] - Partie 2**  *Documents non autorisés*  *Donner plusieurs réponses dont une est juste et le reste n'a pas de sens vaut un 0* |
| --- |

**Exercice 01. Ingénierie des systèmes ML (7pts : 1h)**

Nous voulons concevoir un modèle qui estime la capacité d'un étudiant donné à réussir un semestre. Pour ce faire, nous avons utilisé la plateforme "talents" pour récupérer des informations comme le numéro d'inscription, le nom, le prénom, le nombre d'absences dans chaque module, la note de l'examen intermédiaire de chaque module, le coefficient de chaque module, la wilaya, la spécialité du BAC (SN; SE et TM), le sex, l'age et le semestre (1 ou 2), année d'inscription et la moyenne finale du semestre. Afin de collecter le comportement des étudiants durant chaque module, nous avons envoyé des formulaires aux enseignants de chaque module. Le formulaire contient le numéro d'inscription, nom, prénom et photo de l'étudiant comme information. L'enseignant doit sélectionner un des choix : positif, neutre, négatif.

Nous n'avons pas reçu toutes les informations du comportement, mais nous ne voulons sacrifier ni cette caractéristique ni les échantillons sans réponse. Nous avons remarqué que la wilaya avait une petite corrélation avec la moyenne sauf pour une seule année. Aussi, l'âge est souvent entre 17 et 25 sauf pour quelques cas spéciaux (nous ne voulons pas les sacrifier aussi). En analysant les promotions, nous avons réalisé que chaque année la nouvelle promotion est similaire à celle d'il y a 3 années.

**A) Préparation de données (3pts)**

1. Décrire comment fusionner les données de "talent" et du formulaire en ligne **[0.25pt]**
2. Dans un premier temps, nous voulons entraîner un modèle pour estimer la réussite des 1CPI 1 semestre. Décrire les caractéristiques à garder (en justifiant pourquoi le reste n'est pas gardé), les opérations appliquées sur les caractéristiques et leurs représentations. La réussite est exprimée sous forme de notation ABCDEF et le modèle est discriminatif. Nous préférons des caractéristiques détaillées afin d'avoir plus de précision. **[1.75pt]**
3. Nous voulons généraliser le modèle sur le reste des niveaux et des semestres (avoir un seul modèle). Refaire la même chose. **[1pt]**

**B) Entraînement et validation (3pts)**

1. Nous avons remarqué une relation de causalité entre la catégorie TM de la caractéristique "spécialité du BAC" et la classe "F", le reste des catégories ont des corrélations égales avec toutes les classes. Donc, nous avons décidé d'utiliser une condition sur cette caractéristique (SI BAC = TM ALORS classe = F) et entraîner le modèle sur le reste des caractéristiques. Discuter cette solution : va-t-elle améliorer la performance ou non ? comment ? **[0.5pt]**
2. Nous avons entraîné un arbre de décision CART sur les caractéristiques choisies.
   * Nous avons remarqué que certaines caractéristiques ne sont pas utilisées. Comment l'algorithme a fait pour les ignorer ? **[0.5pt]**
   * Nous avons remarqué que le score de la validation est vraiment bas en le comparant avec celui de l'entraînement. Proposer deux raisons possibles et leurs solutions **[1pt]**
3. Nous avons entraîné deux modèles SVM binaires (*nous avons fusionné les classes A B C ensembles et D E F ensembles*) : un avec hard-margin et un autre avec soft-margin. Nous avons remarqué que l'erreur du hard-margin est vraiment grande en la comparant avec celle du soft-margin malgrès que le F1-score est bon. Que peut-on dire à propos de la distribution du dataset ? **[0.5pt]**
4. Nous avons entraîné un modèle de régression logistique. Le modèle a convergé rapidement et l'erreur a commencé à fluctuer. Proposer une solution. **[0.5pt]**

**C) Test et déploiement (1pt)**

Nous nous intéressons beaucoup plus à la classe "F" afin d'affecter ses étudiants à un programme de tutorat.

1. Quelle est la métrique utilisée pour sélectionner le modèle à déployer (P, R ou F1) et pourquoi ? **[0.5pt]**
2. Quelle est la moyenne utilisée (Classe F, micro-avg, macro-avg ou weighted-avg) et pourquoi ? **[0.5pt]**

**Exercice 02. Réseaux de neurones (7pts : 30mn)**

***A) Réseaux de neurones à propagation avant***

Nous avons un problème de classement avec 5 classes de sortie. Afin de le résoudre, nous avons utilisé un réseau de neurones avec des couches cachées et une couche de sortie de 5 neurones. La fonction d'activation utilisée pour toutes les couches est la fonction logistique où les paramètres sont initialisés à 1.

* Expliquer les 3 limites de cette architecture en proposant une amélioration pour chacune **[1.5pts]**

***B) Réseaux de neurones convolutionnels (CNN)***

Voici une architecture CNN

|  |  |
| --- | --- |

1. Calculer le nombre des paramètres dans les deux couches Conv2D et MaxPool2D (nous utilisons seulement un canal d'entrée et un de sortie pour les deux) **[1pt]**
2. Afin d'implémenter ces deux couches, combien de neurones devons-nous utiliser pour chacune ? **[0.5pt]**
3. Pour chaque neurone, le poids du biais est 1 et la fonction d'activation est "ReLu". Calculer et dessiner les deux matrices X' et X'' (la structure de Conv2D est représentée dans le schéma ci-dessus). **[1pt]**
4. Si nous supposons que la fonction d'activation utilisée est "ReLu" et le gradient passé par chaque cellule de X'' est de 1, calculer le gradient qui passe par X[3, 3] (colorée). Utiliser la moyenne pour calculer le gradient qui passe par plusieurs sorties. **[1pt]**

***C) Réseaux de neurones récurrents (RNN) et attention***

Voici l'architecture Transformer

|  |  |
| --- | --- |

1. Choisir la représentation détaillée du transformer (a, b, c ou d) étant donné une séquence d'entrée "s" de taille M et une séquence de sortie "d" de taille O. **[0.5pt]**
2. Expliquer comment le transformer est utilisé pour la traduction automatique du texte **[0.75pt]**
3. Expliquer comment les RNN sont utilisés pour la même tâche **[0.75pt]**

**☙ Bonne chance**