Les bases de Machine Learning

2023/2024

II-BDCC 1ere Année

Examen final

Nom & prénom : Oussous Anas

Comment répondre aux questions :

- Les questions de cours, de synthèse, d'interprétation à fournir sur ce même document avec le nom « nom_prenom_project.docx », le convertir en pdf, puis le mettre sur googleclassroom
- Les questions qui nécessitent un code source, le mettre sur votre fichier « nom_prenom_project.ipynb » puis le mettre sur googleclassroom

Exercice 1 : Question de cours

1. Ce que c'est Machine Learning ?

Réponse : Machine Learning est une branche de l'intelligence artificielle qui permet aux systèmes d'apprendre et de s'améliorer à partir de l'expérience sans être explicitement programmés. Il implique le développement d'algorithmes capables d'identifier des modèles dans les données et de faire des prédictions ou des décisions basées sur de nouvelles données.

2. Quelle est la différence entre l'apprentissage supervisé et l'apprentissage non supervisé ?

Réponse : Apprentissage supervisé implique l'entraînement d'un modèle sur des données étiquetées, c'est-à-dire que l'entrée est accompagnée d'une sortie connue. Le modèle apprend à mapper les entrées aux sorties. Et Apprentissage non supervisé implique l'entraînement d'un modèle sur des données non étiquetées, où le modèle essaie d'identifier des modèles et des relations dans les données sans étiquettes prédéfinies.

3. Donner une définition aux concepts suivants :

- Problème de classification : Un problème où la variable de sortie est une catégorie ou un label de classe, comme la détection de spam (spam/non spam) ou le diagnostic de maladie (maladie/pas de maladie).

- Problème de régression : Un problème où la variable de sortie est une valeur continue, comme la prédiction des prix de l'immobilier ou des températures.

4. Quelles sont les principales métriques à utiliser pour évaluer un modèle de régression ?

Réponse : Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) et R-squared (R^2)

5. Quelles sont les principales métriques à utiliser pour évaluer un modèle de classification ?

Réponse : Accuracy, Precision, Recall, F1-Score et Area Under the Receiver Operating Characteristic Curve (AUC-ROC)

6. Quel est le rôle de chaque librairie parmi les librairies suivants :

- Numpy : Fournit un support pour les grands tableaux multidimensionnels et les matrices, ainsi qu'une collection de fonctions mathématiques pour opérer sur ces tableaux.

- Pandas : Offre des structures de données et des outils d'analyse de données, facilitant la manipulation et l'analyse de grands ensembles de données.

- Matplotlib : Une bibliothèque de tracé utilisée pour créer des visualisations statiques, interactives et animées en Python.

- Sklearn : Fournit des outils simples et efficaces pour l'exploration de données et l'analyse de données, y compris des algorithmes de machine learning et des métriques d'évaluation des modèles.

Exercice 2 : Problème de régression

Commencer par télécharger le fichier « project_iibdcc1.ipynb » à partir de googleclassroom et le renommer en « nom_prenom_ML_iibdcc1.ipynb »

Partie 1 : création du modèle à l'aide de sklearn

1. Dataset :
    a. Exécuter le code source qui se trouve dans le fichier pour générer un dataset synthètique.
    b. Afficher les dimensions du dataset (nombre de lignes, nombre de colonnes)
    c. Afficher le type de données des colonnes du dataset
    d. Afficher les dimensions du dataset (nombre de lignes, nombre de colonnes)
2. Type de problème
    a. Écrire l'instruction qui permet de montrer que target est de type numérique.
        - **Réponse est déjà donnée dans le point précédent.**
    b. Pourquoi il s'agit d'un problème de régression ?
        - **Réponse : C'est un problème de régression car la variable cible est une variable numérique continue.**
3. Features selection
    a. Combien de features existe dans ce dataset ?
    b. Extraire la partie features : X
    c. Extraire la target : y
4. Encodage des variables catégorielles
    a. Pourquoi la vérification de type de variable est une phase importante ?
    b. Écrire le code source qui permet d'afficher le type de chaque feature
    c. Pour ce dataset, est ce qu'il y a des variables à encoder ?
5. Split le dataset
    a. Pourquoi est-il nécessaire de diviser le dataset en training dataset et test dataset ?
        - **Réponse : Pour évaluer la performance du modèle sur des données non vues et vérifier sa généralisation.**
    b. Quelle fonction utiliser pour diviser le dataset en train (X_train,y_train) et en test (X_test,y_test) ? quels sont ces principaux paramètres ?
        - **Réponse : La fonction train_test_split. Les principaux paramètres sont X, y, test_size, et random_state.**
    c. Pourquoi il est important de fixer le paramètre random_state ? utiliser **random_state=23**
        - **Réponse : Pour garantir la reproductibilité des résultats en assurant que la division du dataset est la même à chaque exécution.**
    d. Diviser le data set en training (80%) et en test (20%). Quelle est la taille de X_test, X_train ?
6. Model
    a. Quelle est la forme mathématique du modèle qui correspond à ce dataset ?
        - **Réponse : La forme mathématique est $y = \beta0 + \beta1*x1 + \beta2*x2$.**
    b. En se basant sur LinearRegression, créer le modèle
    c. Entrainer le modèle
    d. Afficher les paramètres du modèle
    e. Sur la base des paramètres du modèle trouvé, créer une fonction **predict1 (X)** capable de faire des prédictions
    f. Afficher le résultat retourné par la fonction predict1 si on lui passe le X_test
7. Evaluation du modèle
    a. En utilisant sklearn, quelle est la valeur de mse du modèle ? Interpréter le résultat.
    b. En utilisant sklearn, quelle est la valeur de r2 du modèle ? Interpréter le résultat.
    c. Sans utiliser sklearn, créer une fonction *evaluer (y_hat,y_test)* capable de calculer r2 et mse et les retourner

Partie 2 : création du modèle sans utiliser sklearn

1. L'objectif est de créer notre propre fonction fit(X,y) qui se base sur l'algorithme de gradient descent
   a. Ce que c'est l'algorithme de gradient descent ?
      - **L'algorithme de gradient descent est une méthode d'optimisation pour trouver les paramètres minimisant une fonction de coût. Il ajuste les paramètres du modèle en suivant le gradient de la fonction de coût pour converger vers une solution optimale.**

   b. Quelles sont les principales étapes de l'algorithme ?

- **Initialisation des paramètres : Commencer avec des valeurs initiales pour les paramètres.**
- **Calcul du gradient : Évaluer le gradient de la fonction de coût par rapport aux paramètres.**
- **Mise à jour des paramètres : Ajuster les paramètres en fonction du gradient et d'un taux d'apprentissage.**
- **Répétition : Répéter les étapes 2 et 3 jusqu'à ce que les changements deviennent négligeables ou qu'un nombre maximal d'itérations soit atteint.**

   c. Écrire le code source de cette fonction afin de trouver les paramètres du modèle
   d. En utilisant la fonction fit, trouver les paramètres du modèle
   e. En utilisant les paramètres trouvés, calculer la prédiction à l'aide de la fonction predict1
   f. Évaluer la performance du modèle trouvé
   g. Comparer sa performance avec celle du modèle trouvé à l'aide de sklearn.

Exercice 3 : Problème de classification

Continuer à travailler sur le fichier « nom_prenom_ML_iibdcc1.ipynb »

1. Dataset :
   a. Télécharger le dataset « heart.csv » à partir de googleclassroom
   b. À l'aide de pandas, ouvrir le dataset « heart.csv »
   c. Afficher l'entête du dataset. Utiliser head(10).
   d. Afficher le type de données des colonnes du dataset
   e. Afficher les dimensions du dataset (nombre de lignes, nombre de colonnes)
2. Type de problème
   a. Pourquoi il s'agit d'un problème de classification ?
3. Features selection
   a. Combien de features existe dans ce dataset ?
   b. Extraire la partie features : X
   c. Extraire la target : y
4. Encodage des variables catégorielles
   a. Pourquoi la vérification de type de variable est une phase importante ?
      - **La vérification est importante pour s'assurer que le modèle peut gérer correctement les types de variables surtout si elles doivent être encodées**

   b. Écrire le code source qui permet d'afficher le type de chaque feature
   c. Pour ce dataset, est ce qu'il y a des variables à encoder ? si oui, effectuer l'encodage adéquat
5. Split le dataset
   a. Diviser le data set en training (80%) et en test (20%)
6. Model
   a. Quelle est la forme mathématique du modèle qui correspond à ce dataset ?
   b. En se basant sur LogisticRegression, créer le modèle
   c. Entrainer le modèle
   d. Afficher les paramètres du modèle
   e. Sur la base des paramètres du modèle trouvé, créer une fonction **predict2 (X)** capable de faire des prédictions

f. Afficher le résultat retourné par la fonction predict2 si on lui passe le X_test

7. Evaluation du modèle
    a. Quelle est la valeur de mse du modèle ? Interpréter le résultat.
    b. En utilisant sklearn, quelle est la valeur de r2 du modèle ? Interpréter le résultat.
    c. Sans utiliser sklearn, créer une fonction capable de calculer r2 et mse et les retourner

Partie 2 : création du modèle sans utiliser sklearn

2. L'objectif est de créer notre propre fonction fit qui se base sur l'algorithme de gradient descent
    a. Écrire le code source de cette fonction afin de trouver les paramètres du modèle
    b. En utilisant la fonction fit, trouver les paramètres du modèle
    c. En utilisant les paramètres trouvés, calculer la prédiction à l'aide de la fonction predict2
    d. Évaluer la performance du modèle trouvé
    e. Comparer sa performance avec celle du modèle trouvé à l'aide de sklearn.