CSI 4506 Introduction à l'intelligence artificielle

Devoir 2: Apprentissage automatique

Marcel Turcotte

Version: 3 octobre, 2024 17h17

🎯 Objectifs d'apprentissage

- **Réaliser** une analyse exploratoire de données complète
- **Appliquer** des techniques de prétraitement des données de manière efficace
- **Développer** et **évaluer** des modèles d'apprentissage automatique
- **Optimiser** les hyperparamètres et analyser les performances des modèles

📤 Soumission

- **Date limite :**
  - Soumettez votre document (Jupyter Notebook) avant le 20 octobre à 23h.
- **Devoir individuel ou en groupe :**
  - Ce devoir peut être effectué individuellement (groupe d'une personne) ou en collaboration par deux (groupe de 2 personnes).
  - Un groupe doit soumettre une seule soumission commune.
  - Avant de soumettre, il est nécessaire d'enregistrer votre groupe sur Brightspace.
  - Pour permettre des changements dans la composition des groupes pour chaque devoir, un nouvel enregistrement des groupes est requis pour chaque devoir.
- **Plateforme de soumission :**
  - Téléchargez votre soumission sur Brightspace dans la section Devoir (Devoir 2).
- **Format de soumission :**
  - Soumettez votre document en format Jupyter Notebook sur Brightspace.

Remarque importante : Si le correcteur ne peut pas exécuter votre code, votre soumission recevra une note de zéro. Il est de votre responsabilité de vous assurer que votre soumission fonctionne sur un autre ordinateur que le vôtre et que toutes les cellules de votre notebook sont exécutables.

☑ Exigences

1. Analyse exploratoire

Exploration des données

Dans ce devoir, nous utiliserons le jeu de données de prédiction du diabète, accessible via Diabetes Prediction Dataset. Pour réduire la complexité liée à l'exigence de connexion de Kaggle, le jeu de données a été mis à disposition sur un dépôt GitHub public :

- github.com/turcotte/csi4106-f24/tree/main/assignments-data/a2

Vous pouvez accéder et lire le jeu de données directement depuis ce dépôt GitHub dans votre notebook Jupyter.

(1) **Charger le jeu de données et fournir un résumé de sa structure** :

- Décrivez les attributs (colonnes), leurs types de données et la variable cible.

(2) **Analyse de la distribution des attributs** :

- Examinez la distribution de chaque attribut à l'aide de visualisations appropriées telles que des histogrammes et des boxplots. Discutez des informations obtenues, y compris la présence de valeurs aberrantes.

(3) **Distribution de la variable cible** :

- Analysez la distribution de la variable cible pour identifier les déséquilibres de classes. Utilisez des diagrammes en barres pour visualiser les fréquences des classes.

(4) **Fractionnement des données** :

- Divisez le jeu de données en ensembles d'entraînement (80 %) et de test (20 %) en utilisant la méthode du holdout.
- Assurez-vous que ce fractionnement intervient avant tout prétraitement afin d'éviter les fuites de données.

Prétraitement des données

(5) **Encodage des variables catégoriques** :

- Encodez les variables catégoriques. Justifiez la méthode choisie.

(6) **Normalisation/Standardisation des attributs numériques** :

- Normalisez ou standardisez les attributs numériques si nécessaire. Décrivez la technique utilisée (par exemple, le scaling Min-Max, StandardScaler) et expliquez pourquoi elle est appropriée pour ce jeu de données.
- Assurez-vous que cette technique est appliquée uniquement aux données d'entraînement, avec la même transformation appliquée ensuite aux données de test sans nouvel ajustement.

Développement et évaluation des modèles

(7) **Développement des modèles** :

- Implémentez les modèles d'apprentissage automatique abordés en classe : arbres de décision, K-Nearest Neighbors (KNN) et régression logistique. Utilisez les paramètres par défaut de scikit-learn comme base pour entraîner chaque modèle.

(8) **Évaluation des modèles** :

- Utilisez la validation croisée pour évaluer chaque modèle, en justifiant votre choix du nombre de plis.
- Évaluez les modèles à l'aide de métriques telles que la précision, le rappel et le score F1.

Optimisation des hyperparamètres

(9) **Exploration et évaluation des performances :**

- Étudiez l'impact de la variation des valeurs des hyperparamètres sur les performances de chaque modèle.
- Concentrez-vous sur les hyperparamètres pertinents suivants pour chaque modèle :
  - DecisionTreeClassifier : `criterion` et `max_depth`.
  - LogisticRegression : `penalty`, `max_iter`, et `tol`.
  - KNeighborsClassifier : `n_neighbors` et `weights`.

- Employez une stratégie de recherche en grille ou utilisez les méthodes intégrées de scikit-learn pour évaluer exhaustivement toutes les combinaisons des valeurs d'hyperparamètres. La validation croisée doit être utilisée pour évaluer chaque combinaison.
- Quantifiez les performances de chaque configuration d'hyperparamètres en utilisant des métriques telles que la précision, le rappel et le score F1.
- Affichez les résultats dans un format tabulaire ou graphique (par exemple, graphiques en ligne, diagrammes en barres) pour démontrer efficacement l'influence des variations des hyperparamètres sur les performances du modèle.
- Spécifiez les valeurs par défaut de chaque hyperparamètre testé.
- Analysez les résultats et offrez des perspectives sur les configurations d'hyperparamètres ayant obtenu les meilleures performances pour chaque modèle.

Analyse des résultats

(10) **Comparaison des modèles** :

- Comparez les résultats obtenus pour chaque modèle.
- Discutez des différences observées dans les performances des modèles et fournissez des explications potentielles. Considérez des aspects tels que la complexité des modèles, le déséquilibre des données, le surapprentissage et l'impact du réglage des paramètres sur les résultats globaux.
- Fournissez des recommandations sur le(s) modèle(s) à choisir pour cette tâche et justifiez vos choix en fonction des résultats de l'analyse.
- Entraînez le(s) modèle(s) recommandé(s) en utilisant les valeurs optimales des paramètres identifiés lors de l'optimisation des paramètres. Appliquez ensuite le modèle entraîné aux données de test. Documentez vos observations de manière détaillée. Évaluez spécifiquement si les résultats dérivés de la validation croisée sont cohérents avec ceux obtenus sur le jeu de test.

2. Documentation de l'analyse exploratoire

Le rapport doit documenter de manière complète le processus suivi pendant ce devoir. Le notebook Jupyter doit inclure les éléments suivants :

- Votre nom(s), numéro(s) d'étudiant.e.s et un titre de rapport.
- Expliquez comment les tâches ont été réparties entre les membres. Comment avez-vous fait en sorte que les deux personnes atteignent les objectifs d'apprentissage ?

- Une section pour chaque étape de l'analyse exploratoire, contenant le code Python pertinent et les explications ou résultats.
  - Pour les sections nécessitant du code Python, incluez le code dans une cellule.
  - Pour les sections nécessitant des explications ou des résultats, incluez-les dans une cellule distincte ou en combinaison avec les cellules de code.
- Assurez une séparation logique du code dans différentes cellules. Par exemple, la définition d'une fonction doit se trouver dans une cellule et son exécution dans une autre. Évitez de placer trop de code dans une seule cellule pour maintenir la clarté et la lisibilité.
- Le notebook que vous soumettez doit inclure les résultats de l'exécution, y compris les graphiques, en veillant à ce que l'assistant d'enseignement puisse évaluer le notebook sans avoir à exécuter le code.

Évaluation

- **Effort global dans le rapport (5%)**
- **Exploration des données (10%)**
- **Prétraitement des données (20%)**
- **Développement et évaluation des modèles (20%)**
- **Optimisation des paramètres (30%)**
- **Analyse des résultats (10%)**
- **Ressources et références (5%)**

Ressources

Comme mentionné précédemment, assurez-vous de citer toute partie de votre code dérivée de sites Web, manuels ou autres ressources externes.

Actuellement, de nombreux programmeurs utilisent l'intelligence artificielle pour améliorer leur productivité, une tendance qui devrait continuer à croître. Pour mieux vous préparer au marché du travail, il est plausible d'utiliser ces technologies. Cependant, il est impératif que vous compreniez pleinement les concepts sur lesquels vous êtes évalué, car ces outils ne seront pas disponibles lors des évaluations en personne.

Si vous utilisez une assistance par IA, documentez soigneusement vos interactions. Incluez les outils et leurs versions dans votre rapport, ainsi qu'une transcription de toutes les interactions. La plupart des assistants IA conservent une trace de vos conversations. La pratique recommandée est de créer une nouvelle conversation spécifiquement pour le devoir et de réutiliser cette

conversation tout au long de votre travail sur le devoir. Assurez-vous que cette conversation est uniquement dédiée au devoir. Soumettez cette transcription de conversation dans la section des références de votre notebook Jupyter.

Questions

- Vous pouvez poser vos questions dans le sujet “Devoir” du forum de discussion sur Brightspace.
- Alternativement, vous pouvez envoyer un email à l’un des quatre assistants d’enseignement. Toutefois, l’utilisation du forum est fortement préférée, car elle permet à vos camarades de classe de bénéficier des questions et des réponses fournies par les assistants d’enseignement.