CSI 4506

Assignment 1

Adnane Bouchama 300 177 651

23 September 2024

1. Introduction et Objectifs du Projet

Dans ce devoir, mon objectif était de préparer des données pour l'apprentissage automatique. J'ai choisi de travailler sur un jeu de données concernant la santé maternelle, dans le but de prédire le niveau de risque (RiskLevel) basé sur plusieurs indicateurs médicaux comme l'âge, la pression artérielle, la fréquence cardiaque, etc. La préparation des données est une étape critique, car la qualité des données influence directement les performances d'un modèle d'apprentissage automatique, selon l'adage "Garbage In, Garbage Out".

J'ai entrepris des recherches sur **Kaggle**, **Stack Overflow**, ainsi que des articles de recherche et des tutoriels de **scikit-learn** pour approfondir mes connaissances sur la préparation des données. Ces recherches m'ont aidé à décider des méthodes à utiliser, comme l'imputation des valeurs manquantes, la normalisation des attributs, et l'encodage des variables catégorielles.

2. Description du Jeu de Données (10%)

Pour ce projet, j'ai choisi le jeu de données sur le risque de santé maternelle. J'ai trouvé ce jeu de données intéressant parce qu'il comporte un bon nombre d'exemples (1 013 instances) et plusieurs attributs médicaux pertinents pour une tâche de classification. Cela me permet de travailler sur des données réalistes, en particulier dans le contexte médical, qui est un domaine important et très significatif.

- **Attributs :**
    - Age : Âge de la mère (en années).
    - SystolicBP : Pression artérielle systolique (en mmHg).
    - DiastolicBP : Pression artérielle diastolique (en mmHg).
    - BS : Niveau de sucre dans le sang (en mmol/L).
    - BodyTemp : Température corporelle (en °C).
    - HeartRate : Fréquence cardiaque (battements par minute).
- **Variable cible :**
    - RiskLevel : Indique le niveau de risque de santé maternelle (low risk, mid risk, high risk).

J'ai choisi ce jeu de données car il est bien adapté pour une classification multi-classes et il est pertinent dans le domaine médical, ce qui est un domaine que je trouve fascinant.

3. Analyse Exploratoire des Données

3.1 Analyse des Valeurs Manquantes

Pour commencer, j'ai analysé les valeurs manquantes dans le jeu de données en utilisant la fonction isnull().sum(). Cela m'a permis de vérifier s'il y avait des valeurs manquantes qui nécessitaient un traitement.

```
print(dataset.isnull().sum())
```

Résultat : Heureusement, aucune valeur manquante n'a été trouvée dans ce jeu de données. Cela a simplifié le travail, car je n'ai pas eu besoin d'imputer des valeurs manquantes.

3.2 Analyse des Attributs

Ensuite, j'ai effectué une analyse des attributs, y compris la variance de chaque attribut pour vérifier s'ils étaient informatifs. J'ai également généré des histogrammes pour visualiser la distribution des valeurs.

```
variances = dataset.var()
print("Variance de chaque attribut :\n", variances)
```

Conclusion : Tous les attributs avaient une variance suffisante. J'ai également utilisé des histogrammes pour visualiser la distribution des attributs :

```
dataset.hist(figsize=(15, 15))
plt.tight_layout()
plt.show()
```

En générant les histogrammes, j'ai pu identifier la distribution des différentes valeurs des attributs. Cela m'a aidé à mieux comprendre la nature des données (par exemple, vérifier s'il y avait des pics ou des valeurs aberrantes). Selon la documentation **scikit-learn**, cette étape est essentielle pour décider du prétraitement (par exemple, normalisation ou standardisation).

3.3 Analyse de la Distribution des Classes (10%)

Pour analyser la distribution de la variable cible RiskLevel, j'ai utilisé un diagramme en barres. Cette analyse m'a aidé à voir si les classes étaient équilibrées ou non, ce qui est important pour éviter un biais dans l'entraînement du modèle.

```
dataset['RiskLevel'].value_counts().plot(kind='bar', color='skyblue')
plt.xlabel('Niveau de Risque')
plt.ylabel('Nombre d\'instances')
plt.title('Distribution des Classes de Risque')
```

plt.show()

Conclusion : La distribution des classes est relativement équilibrée, ce qui est un point positif car cela permet d'éviter le déséquilibre de classe, souvent problématique pour l'entraînement d'algorithmes de classification. Mes recherches sur **Kaggle** m'ont appris que des techniques comme le sur-échantillonnage sont parfois nécessaires en cas de déséquilibre, mais heureusement, ce n'était pas nécessaire ici.

3.4 Prétraitement des Données (20%)

Normalisation des Attributs Numériques : J'ai normalisé les attributs numériques en utilisant StandardScaler de **scikit-learn**. Cette méthode était appropriée car les attributs avaient des unités de mesure différentes (par exemple, mmHg, battements par minute).

Pourquoi la Normalisation ? La normalisation permet de mettre tous les attributs sur une même échelle, ce qui facilite la convergence des algorithmes d'apprentissage automatique, surtout ceux qui sont sensibles aux échelles des données comme les régressions linéaires et les réseaux de neurones. J'ai trouvé cette étape utile pour garantir des performances stables du modèle.

Encodage One-Hot : J'ai ensuite encodé la variable catégorielle RiskLevel en utilisant l'encodage one-hot, afin de la transformer en plusieurs colonnes de type binaire (RiskLevel_high risk, RiskLevel_mid risk, etc.). Cela permet d'éviter l'attribution implicite d'un ordre entre les catégories.
J'ai eu énormément de soucis avec cette étape, j'avais une erreur qui ne voulait pas partir et j'ai essayé diverses techniques. J'ai perdu énormément de temps en essayant de la résoudre.

Cependant, je pense avoir réussi grâce aux recherches dans la documentation **pandas** pour comprendre les différentes méthodes d'encodage, et j'ai trouvé que l'encodage one-hot est le plus adapté pour les variables catégorielles non ordinales.

4. Données d'Entraînement et Cible

Ensuite, j'ai défini les caractéristiques (X) et la cible (y). Les caractéristiques incluent toutes les colonnes sauf celles encodées de RiskLevel, qui constituent la cible. Cette étape était cruciale, car il était important de bien séparer les attributs d'entrée des cibles de sortie pour éviter toute fuite de données lors de l'entraînement du modèle.

5. Ensembles d’Entraînement et de Test

Pour terminer, j'ai divisé le jeu de données en ensembles d'entraînement (80%) et de test (20%) à l’aide de la fonction train_test_split de **scikit-learn**. Cette séparation est essentielle pour évaluer les performances du modèle sur des données non vues. **Pourquoi 80%-20% ?** J'ai opté pour un ratio 80%-20%, qui est une bonne pratique courante en machine learning (selon mes recherches sur **Towards Data Science**). Cela permet de garantir suffisamment de données pour entraîner le modèle tout en laissant une proportion adéquate pour l’évaluation.

6. Ressources et Références

- J'ai utilisé **Kaggle** pour explorer des techniques d'analyse de données.
- **Stack Overflow** m'a été très utile pour comprendre comment utiliser certaines fonctions de la bibliothèque **pandas**, notamment lors de l'encodage des variables catégorielles.
- Les articles sur **Towards Data Science** et les tutoriels **scikit-learn** ont enrichi ma compréhension des techniques de prétraitement des données.
- Youtube divers vidéos m’ont permis de télécharger Jupyter Notebook, pip et d’autres outils.

7. Conclusion

Dans ce devoir, j'ai exploré et préparé les données pour un projet de classification. J'ai utilisé diverses techniques, telles que la normalisation, l'encodage des variables catégorielles, et la division en ensembles d’entraînement et de test. Ces étapes sont cruciales pour garantir la qualité des données avant l'entraînement de modèles. J'ai trouvé certaines parties, comme l'encodage des variables catégorielles, plus difficiles que prévu, mais cela m'a permis d'améliorer mes compétences en manipulation des données.

J'ai également compris l'importance de la préparation des données pour l'apprentissage automatique, car une bonne qualité des données garantit de meilleurs résultats. Ce devoir m'a également donné l'opportunité d'explorer les défis spécifiques aux données médicales, et cela m'a beaucoup appris sur la valeur du prétraitement.

Merci