Compte Rendu d'Analyse de Données

Projet de Machine Learning

AKHRAIS HASNAE

3 décembre 2025

Table des matières

1 Introduction

Ce rapport présente une analyse complète d'un jeu de données synthétique généré pour une tâche de classification binaire. L'objectif principal est d'explorer les données, d'effectuer un prétraitement approprié et de comparer les performances de différents algorithmes d'apprentissage automatique.

2 Description des Données

2.1 Caractéristiques du Dataset

Le jeu de données utilisé comprend 100 observations avec les caractéristiques suivantes :

- **Variables numériques** :
  - `num_col1` : Variable continue (0-100)
  - `num_col2` : Variable discrète (1-50)
- **Variables catégorielles** :
  - `cat_col1` : 3 catégories (A, B, C)
  - `cat_col2` : 2 catégories (X, Y)
- **Variable cible** : `target` (binaire : 0 ou 1)

3 Prétraitement des Données

3.1 Nettoyage des Données

Les étapes suivantes ont été appliquées :

1. **Suppression des doublons** : Les observations dupliquées ont été éliminées pour garantir l'unicité des données.
2. **Traitement des valeurs manquantes** :
   - Utilisation de `KNNImputer` avec $k = 5$ voisins
   - Méthode basée sur la proximité des observations
3. **Encodage des variables catégorielles** :
   - Application de `OneHotEncoder`
   - Gestion des catégories inconnues avec `handle_unknown='ignore'`
4. **Standardisation** :
   - Utilisation de `StandardScaler`
   - Transformation : $x' = \frac{x-\mu}{\sigma}$

3.2 Feature Engineering

Une nouvelle caractéristique a été créée :

$$\text{feature\_ratio} = \frac{\text{num\_col1}}{\text{num\_col2} + 10^{-5}} \tag{1}$$

Cette transformation capture la relation entre les deux variables numériques.

4 Analyse Exploratoire des Données

4.1 Distribution des Variables Numériques

L'analyse de la distribution de `num_col1` révèle :
— Une concentration des valeurs autour de la moyenne
— Une légère asymétrie dans la distribution
— Présence de quelques valeurs extrêmes

4.2 Analyse des Boxplots

La comparaison des distributions de `num_col1` entre les deux classes cibles montre :
— Des médianes différentes entre les groupes
— Une indication d'importance potentielle pour la prédiction
— Une séparation partielle des classes

4.3 Matrice de Corrélation

L'analyse de corrélation révèle :
— Certaines variables présentent des corrélations fortes
— Ces corrélations peuvent influencer le choix des features
— Nécessité de considérer la multicolinéarité dans la modélisation

4.4 Équilibre des Classes

L'analyse de la variable cible montre :
— Distribution entre les classes 0 et 1
— Information cruciale pour le choix des métriques d'évaluation
— Impact potentiel sur les performances des modèles

5 Modélisation Prédictive

5.1 Algorithmes Testés

Trois algorithmes de classification ont été évalués :

1. **Random Forest Classifier**
   — Ensemble de décision
   — Robuste au surapprentissage
2. **Support Vector Machine (SVM)**
   — Kernel par défaut
   — Probabilités activées
3. **Gradient Boosting Classifier**
   — Apprentissage séquentiel
   — Correction itérative des erreurs

5.2 Validation Croisée

Une validation croisée avec 5 plis (`KFold`, $k = 5$) a été utilisée pour évaluer les performances.

5.2.1 Résultats

Modèle	Accuracy Moyenne	Écart-type
Random Forest	0.550	0.130
SVM	0.420	0.075
Gradient Boosting	0.470	0.087

TABLE 1 – Performances des modèles en validation croisée

5.3 Optimisation des Hyperparamètres

Une recherche en grille (`GridSearchCV`) a été effectuée pour le Random Forest :

5.3.1 Espace de recherche

— `n_estimators` : [50, 100]
— `max_depth` : [None, 10, 20]

5.3.2 Meilleurs paramètres

Les paramètres optimaux trouvés sont :
— `n_estimators` : 50
— `max_depth` : None
— Accuracy CV optimale : 0.55

6 Discussion

6.1 Performance des Modèles

— Le Random Forest obtient les meilleures performances (55%)
— Le SVM présente la variance la plus faible
— Le Gradient Boosting offre un compromis intéressant

6.2 Limitations

1. Taille limitée du dataset (100 observations)
2. Données synthétiques ne reflétant pas nécessairement des patterns réels
3. Performances modestes suggérant une complexité limitée

6.3 Améliorations Possibles

— Augmentation de la taille du dataset
— Exploration d'autres algorithmes (XGBoost, LightGBM)
— Feature engineering plus avancé
— Techniques d'ensemble plus sophistiquées
— Analyse approfondie des features importances

7 Conclusion

Cette analyse a permis de mettre en place un pipeline complet de machine learning, depuis le prétraitement des données jusqu'à l'optimisation des hyperparamètres. Le Random Forest s'est révélé être le modèle le plus performant avec une accuracy de 55%.

Les résultats suggèrent qu'il existe une certaine structure dans les données, bien que modeste. Des investigations supplémentaires seraient nécessaires pour améliorer significativement les performances prédictives.

Références

— Scikit-learn Documentation : `https://scikit-learn.org/`
— Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.
— Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning.