TP – Classification Arborescente sur le Dataset Titanic

🎯 Objectif

Construire, visualiser, évaluer et comparer un **Arbre de Décision** et une **Random Forest** à partir du dataset Titanic.

🔗 Jeu de données

Dataset : Titanic - Machine Learning from Disaster
Lien : https://www.kaggle.com/c/titanic/data

🔹 Partie 1 – Chargement & Exploration

1. **Chargez le jeu de données Titanic.** Affichez les 5 premières lignes et les types de données. Que remarquez-vous ?
2. **Combien y a-t-il de passagers au total ?** Combien de survivants ? Calculez les proportions.
3. **Identifiez les colonnes avec des valeurs manquantes.** Pour chaque colonne concernée, indiquez le nombre et le pourcentage de valeurs manquantes.

🔹 Partie 2 – Nettoyage & Préparation

4. **Imputez les valeurs manquantes :** `Age` par la médiane, `Embarked` par la modalité la plus fréquente.
5. **Supprimez les colonnes inutiles :** `PassengerId`, `Name`, `Ticket`, `Cabin`. Justifiez ce choix.
6. **Créez une nouvelle variable `Title` à partir de `Name`.** Est-ce une variable informative ? Visualisez sa distribution.

🔹 Partie 3 – Encodage & Prétraitement

7. **Encodez les variables catégorielles** (`Sex`, `Embarked`, `Title`) à l'aide de `LabelEncoder` ou `OneHotEncoder`. Testez les deux méthodes.
8. **Standardisez `Age` et `Fare`** avec `StandardScaler`. Pourquoi cette étape peut-elle être utile même avec des arbres ?
9. **Divisez le dataset en X et y.** Cible = `Survived`. Faites un `train_test_split` (70/30) avec `random_state=42`.

🔹 Partie 4 – Visualisation & Analyse Exploratoire

10. **Affichez la heatmap des corrélations.** Quelles sont les variables les plus corrélées avec `Survived` ?
11. **Réalisez des barplots de survie** par `Sex`, `Pclass`, `Embarked`, et `Title`. Qu'en concluez-vous ?

🔹 Partie 5 – Arbre de Décision

12. **Entraînez un `DecisionTreeClassifier`** (`max_depth=4`). Affichez l'arbre avec `plot_tree` ou `graphviz`.
13. **Évaluez le modèle avec :**
    - Accuracy
    - Précision
    - Rappel
    - F1-score
    - Matrice de confusion
    - Courbe ROC

Partie 6 – Random Forest

14. **Entraînez une `RandomForestClassifier`** avec 100 arbres. Même split que précédemment.
15. **Comparez les performances** de la Random Forest avec l'arbre de décision sur toutes les métriques.
16. **Affichez `feature_importances_`.** Quelles sont les 3 variables les plus importantes ?

Partie 7 – Analyse & Optimisation

17. **Faites varier `max_depth`** entre 1 et 10 pour l'arbre. Tracez l'évolution de l'accuracy en train/test.
18. **Utilisez `cross_val_score`** pour comparer la stabilité des deux modèles.
19. **Effectuez un `GridSearchCV`** sur la Random Forest (`max_depth`, `min_samples_split`, `n_estimators`). Quel est le meilleur modèle ?

Partie 8 – Conclusion

20. **Conclusion :**
    - Quel modèle est le plus performant ?
    - Quelle variable est la plus discriminante ?
    - Quels avantages/inconvénients pour chaque méthode ?
    - Quelles pistes d'amélioration ?
