PROPOSAL DE PROJET MACHINE LEARNING

Système de Détection et Classification des Maladies Cardiaques

Étudiant : MASSOUANE Abdelati

Machine Learning

Projet à rendre : 5 jours après l'examen

Table des matières

1 Informations Générales

Résumé du Projet

- **Titre :** Système de Détection et Classification des Maladies Cardiaques
- **Domaine :** Santé & Machine Learning
- **Type de problème :** Classification Binaire
- **Durée :** 5 jours
- **Technologies :** Python, Scikit-learn, MLflow, GitHub/GitLab, Overleaf

1.1 Contexte et Problématique

Les maladies cardiovasculaires représentent la première cause de mortalité dans le monde. La détection précoce est cruciale pour améliorer les chances de survie et réduire les complications médicales.

1.2 Objectifs du Projet

Le projet vise à développer un système de machine learning complet capable de :

1. **Identifier les patterns** dans les données médicales grâce à la réduction de dimensionnalité
2. **Segmenter les patients** en groupes à risque via des algorithmes de clustering
3. **Prédire la présence** de maladies cardiaques avec des modèles de classification

2 Dataset Utilisé

2.1 Description du Dataset

Heart Disease UCI Dataset

- **Source :** Kaggle
- **URL :** https://www.kaggle.com/datasets/ronitf/heart-disease-uci
- **Nombre d'observations :** ∼1000 patients
- **Nombre de features :** 14 variables
- **Type :** Données médicales réelles

2.2 Variables Principales

Le dataset contient les variables suivantes :

3 Algorithmes Utilisés

Variable	Description
Age	Âge du patient
Sex	Sexe (1 = homme, 0 = femme)
Chest Pain Type	Type de douleur thoracique (4 valeurs)
Resting BP	Pression artérielle au repos (mm Hg)
Cholesterol	Cholestérol sérique (mg/dl)
Fasting Blood Sugar	Glycémie à jeun > 120 mg/dl
Resting ECG	Résultats ECG au repos
Max Heart Rate	Fréquence cardiaque maximale atteinte
Exercise Angina	Angine induite par l'exercice
Oldpeak	Dépression ST induite par l'exercice
Slope	Pente du segment ST à l'exercice maximal
CA	Nombre de vaisseaux colorés par fluoroscopie
Thal	Thalassémie (3 = normal, 6 = défaut fixe, 7 = défaut réversible)
Target	**Présence de maladie cardiaque (0 = Non, 1 = Oui)**

TABLE 1 – Variables du dataset Heart Disease UCI

3.1 Réduction de Dimensionnalité

1. **PCA (Principal Component Analysis)** : Identifier les composantes principales qui expliquent le plus de variance
2. **t-SNE (t-Distributed Stochastic Neighbor Embedding)** : Visualisation non-linéaire en 2D/3D des patterns complexes
3. **NMF (Non-negative Matrix Factorization)** : Décomposition en facteurs non-négatifs

3.2 Clustering (Apprentissage Non-Supervisé)

1. **K-Means** : Segmentation de base des patients en K groupes par centroïdes
2. **Agglomerative Clustering** : Analyse hiérarchique des groupes de patients
3. **DBSCAN** : Détection d'outliers et clusters de forme arbitraire basée sur la densité

Métriques d'évaluation : Silhouette Score, Davies-Bouldin Index, Inertie

3.3 Classification (Apprentissage Supervisé)

Métriques d'évaluation : Accuracy, Precision, Recall, F1-Score, ROC-AUC, Matrice de Confusion

4 Méthodologie

4.1 Phase 1 : Exploration et Prétraitement

— Analyse exploratoire des données (EDA)

— Gestion des valeurs manquantes

— Normalisation/Standardisation des features

Algorithme	Justification
Logistic Regression	Modèle baseline, haute interprétabilité
K-Nearest Neighbors	Apprentissage basé sur la similarité
Decision Tree	Règles de décision interprétables
SVM	Performance sur données non-linéaires
Random Forest	Robustesse et feature importance
AdaBoost	Boosting pour améliorer weak learners
Gradient Boosting	Optimisation séquentielle des erreurs

TABLE 2 – Algorithmes de classification utilisés

— Encodage des variables catégorielles
— Visualisations initiales (heatmap de corrélation, distributions)

4.2 Phase 2 : Réduction de Dimensionnalité

— Application de PCA pour identifier les composantes principales
— Visualisation t-SNE en 2D et 3D
— Analyse NMF pour la décomposition matricielle
— Évaluation de la variance expliquée

4.3 Phase 3 : Clustering

— K-Means avec optimisation du nombre de clusters (méthode Elbow)
— Clustering hiérarchique avec dendrogramme
— DBSCAN pour la détection d'anomalies
— Comparaison des résultats via métriques

4.4 Phase 4 : Classification

— Séparation train/test (80/20)
— Entraînement des 7 algorithmes
— Validation croisée (k-fold)
— Optimisation des hyperparamètres (GridSearchCV)
— Comparaison des performances

5 Technologies et Outils

5.1 Stack Technologique

Librairies Python

— **Scikit-learn** : Tous les modèles ML
— **Pandas & NumPy** : Manipulation et traitement des données
— **Matplotlib & Seaborn** : Visualisations statiques
— **Plotly** : Visualisations interactives
— **MLflow** : Tracking des expériences et gestion des modèles

5.2 Utilisation de MLflow

MLflow sera utilisé pour :

1. **Tracking des expériences** : Enregistrement automatique de tous les paramètres et métriques
2. **Comparaison des modèles** : Interface visuelle pour comparer les performances
3. **Versioning des modèles** : Sauvegarde des meilleurs modèles
4. **Reproductibilité** : Garantir la reproductibilité des résultats

```
# Exemple d'utilisation MLflow
import mlflow
import mlflow.sklearn

with mlflow.start_run(run_name="Random_Forest"):
    mlflow.log_param("n_estimators", 100)
    mlflow.log_param("max_depth", 10)

    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)

    accuracy = accuracy_score(y_test, y_pred)
    mlflow.log_metric("accuracy", accuracy)
    mlflow.sklearn.log_model(model, "model")
```

5.3 Structure du Projet

```
heart-disease-ml/
 data/
    raw/              # Données brutes
    processed/        # Données prétraitées
 notebooks/
    01_eda.ipynb
    02_dimensionality_reduction.ipynb
    03_clustering.ipynb
    04_classification.ipynb
 src/
    preprocessing.py
    models.py
    evaluation.py
 mlruns/               # MLflow tracking
 requirements.txt
 README.md
 .gitignore
```

6 Livrables

6.1 Repository GitHub/GitLab

— Code source complet et commenté

— README détaillé avec instructions d'installation et d'utilisation

— Notebooks Jupyter avec analyses complètes

— Fichier `requirements.txt` pour les dépendances

— `.gitignore` approprié

6.2 Rapport Overleaf (Maximum 10 pages)

Section	Pages	Contenu
Abstract	0.5	Résumé du projet et résultats
Introduction	1	Contexte, problématique, objectifs
Dataset	1.5	Description, EDA, prétraitement
Réduction Dim.	1.5	PCA, t-SNE, NMF + visualisations
Clustering	1.5	K-Means, Agglomerative, DBSCAN
Classification	2.5	7 modèles + comparaison détaillée
Résultats	1	Interprétation, feature importance
Conclusion	0.5	Résumé et perspectives

TABLE 3 – Structure du rapport Overleaf

Visualisations clés à inclure :

— Heatmap de corrélation

— Graphique de variance expliquée (PCA)

— Scatter plots t-SNE

— Dendrogramme (clustering hiérarchique)

— Courbes ROC comparatives

— Feature importance (Random Forest)

— Matrices de confusion pour tous les modèles

7 Critères de Réussite

Objectifs de Performance

Tous les algorithmes requis implémentés et testés

MLflow tracking opérationnel avec tous les paramètres et métriques

Accuracy $> 80\%$ sur le meilleur modèle de classification

Visualisations claires, informatives et professionnelles

Code reproductible avec documentation complète

Rapport académique structuré et bien rédigé

Repository GitHub/GitLab propre et organisé

8 Conclusion

Ce projet représente une application complète des techniques de machine learning sur un problème réel de santé publique. Il va nous permettre de :

1. Maîtriser l'ensemble du pipeline ML (du prétraitement au déploiement)
2. Comparer objectivement différentes approches (supervisées et non-supervisées)
3. Utiliser des outils professionnels (MLflow, Git)

4. Produire une analyse scientifique rigoureuse

Le choix du dataset sur les maladies cardiaques garantit des résultats concrets et interprétables, avec un impact potentiel dans le domaine médical.