# Dossier Technique - Projet Machine Learning Vin 🍷

## Table des matières

1. Présentation du projet

2. Architecture et organisation du code

3. Description des modules

4. Scripts utilitaires

5. Fonctionnalités implémentées

6. Points à valider / amélioration

7. Mode d’emploi pour l’utilisateur

8. Ressources et références

## 1. Présentation du projet

Ce projet a pour objectif la réalisation d’une application interactive en Python avec Streamlit pour présenter un pipeline complet de Machine Learning sur un jeu de données de vin (vin.csv). L’application permet :  
- L’exploration des données  
- Le prétraitement (gestion des valeurs manquantes, suppression de colonnes)  
- La modélisation avec choix d’algorithmes  
- L’évaluation du modèle via métriques et visualisations

## 2. Architecture et organisation du code

├── data/  
│ └── vin.csv # Jeu de données CSV (à fournir manuellement)  
│  
├── pages/  
│ ├── exploration.py # Module d'exploration des données  
│ ├── pretraitement.py # Module de prétraitement des données  
│ ├── modelisation.py # Module de modélisation ML  
│ ├── evaluation.py # Module d’évaluation du modèle  
│ └── app.py # Point d’entrée principal (gestion navigation)  
│  
├── setup.py # Script d'installation des dépendances  
├── run.py # Script de lancement de l’application  
├── requirements.txt # Liste des packages requis  
├── README.md # Documentation utilisateur / notes projet  
└── .gitignore # Fichiers/dossiers ignorés par Git

## 3. Description des modules

### pages/exploration.py

Permet d’afficher un aperçu des données, des statistiques descriptives, des histogrammes, un pairplot et la matrice de corrélation.  
Fonctionnalités clés :  
- Affichage du DataFrame  
- Statistiques descriptives  
- Sélection dynamique des colonnes numériques pour histogrammes  
- Pairplot interactif  
- Matrice de corrélation

### pages/pretraitement.py

Gère le nettoyage des données, avec notamment :  
- Affichage des valeurs manquantes  
- Imputation des valeurs manquantes par différentes stratégies (moyenne, médiane, mode)  
- Suppression de colonnes selon choix utilisateur

### pages/modelisation.py

Module pour la modélisation avec :  
- Sélection du modèle ML (ex. RandomForest, LogisticRegression, SVM)  
- Division train/test des données  
- Entraînement du modèle  
- Possibilité de prédiction sur de nouvelles données (à implémenter)

### pages/evaluation.py

Affiche les résultats du modèle entraîné :  
- Rapport de classification  
- Matrice de confusion  
- Visualisations graphiques associées

### pages/app.py

Point d’entrée Streamlit qui gère la navigation entre les modules via une sidebar. Permet de lancer l’une des pages selon le choix utilisateur.

## 4. Scripts utilitaires

### setup.py

Script automatisant l’installation des dépendances Python et de l’environnement virtuel. Il lit requirements.txt et exécute pip install pour configurer l’environnement.  
Usage :  
python setup.py

### run.py

Script de démarrage principal de l’application Streamlit, qui importe et lance le module pages/app.py  
Usage :  
python run.py  
ou  
streamlit run run.py

## 5. Fonctionnalités implémentées

- Chargement dynamique du dataset local  
- Exploration interactive des données (statistiques, graphiques)  
- Gestion avancée du prétraitement (imputation et suppression de colonnes)  
- Modélisation ML avec pipeline complet (split, fit, predict)  
- Évaluation détaillée des modèles avec métriques et graphiques  
- Navigation fluide entre modules via sidebar

## 6. Points à valider / amélioration

- Intégrer la sélection automatique de la target dans la modélisation  
- Ajouter la possibilité d’importer de nouvelles données utilisateurs pour prédiction  
- Enregistrer et charger des modèles ML (sérialisation)  
- Ajouter plusieurs algorithmes ML dans la sélection avec hyperparamètres réglables  
- Optimiser la gestion des données catégorielles (encodage, gestion de texte)  
- Implémenter des fonctionnalités avancées (GridSearchCV, LazyPredict, Deep Learning)  
- Améliorer l’interface utilisateur (responsive, plus d’interactions)

## 7. Mode d’emploi pour l’utilisateur

1. Cloner le dépôt Git  
2. Installer les dépendances via :  
 python setup.py  
3. Lancer l’application :  
 python run.py  
4. Utiliser la sidebar pour naviguer entre :  
 - Exploration  
 - Prétraitement  
 - Modélisation  
 - Évaluation  
5. Interagir avec les options disponibles dans chaque page

## 8. Ressources et références

- Documentation officielle Streamlit : https://docs.streamlit.io  
- Cours LinkedIn Learning sur Python pour la Data Science  
- Documentation scikit-learn : https://scikit-learn.org  
- Tutoriels sur gestion des données, imputation et modélisation ML