Projet Data Pipeline

1-Contexte

2-Présentation & architecture de la Data Pipeline

3-Modèle utilisé

4-Résultats du modèle

5-Analyse Critique

6-Démonstration

7-Rôles

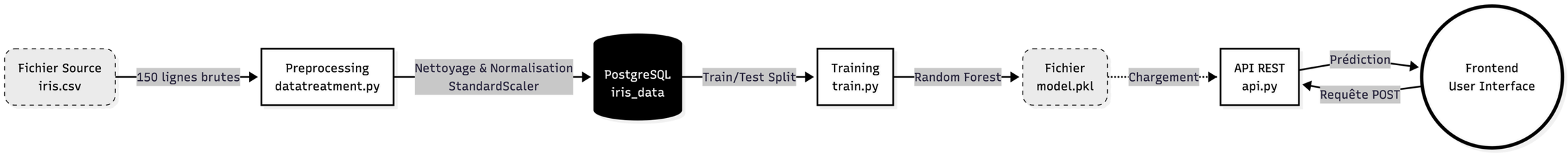
1-Contexte

But : Trouver et prédire la longueur des sépales en fonction de leur largeur

Comment : En utilisant une pipeline de données dockerisée ce A à Z

Grâce à quoi : Un modèle de prédiction régréssif (random forest) et une pipeline en Python et Dockerfile

2-Présentation & architecture de la Data Pipeline



3-Modèle utilisé

Algorithme utilisé : Random Forest Regressor

Notre modèle utilise un **Random Forest**, un algorithme d'apprentissage automatique particulièrement adapté à la prédiction de valeurs numériques.

Comment ça fonctionne ?

Le Random Forest crée une "forêt" composée de **200 arbres de décision** indépendants. Chaque arbre analyse les caractéristiques de la fleur (largeur du sépale) pour estimer la longueur du sépale. La prédiction finale est la **moyenne des prédictions** de tous les arbres, ce qui rend le modèle plus fiable et robuste.

Données d'entraînement

Le modèle a été entraîné sur le **dataset Iris** contenant 150 échantillons de fleurs avec leurs mesures réelles. Les données ont été prétraitées (normalisation, suppression des doublons) pour optimiser la qualité des prédictions.

4-Résultats du modèle

Performances

Métrique	Valeur	Interprétation
R^2 (Note de l'algorithme)	-0.61	⚠️ Modèle limité (voir ci-dessous)
RMSE (Mesure de l'erreur)	1.05 cm	Erreur moyenne de ±1 cm
MAE (Erreur moyenne réelle)	0.73 cm	En moyenne, on se trompe de 0.7 cm

Configuration du modèle

Métrique	Valeur	Signification
Modèle	Random Forest + Polynomial Feature	Extraction des données + ensemble d'arbres
Arbres	200	Erreur moyenne de ±1 cm
Degré polynomial	3	Crée x, x^2, x^3 à partir de sepal_width

5-Analyse Critique

Forces	Faiblesses
R^2 = -0.6088 : Explique -60% de la variance	Dataset limité : Seulement 149 échantillons
RMSE = 1.04 cm : Erreur enorme	Données normalisées : L'utilisateur doit fournir des valeurs normalisées
Random Forest : Robuste au sur-apprentissage	Pas de mise à jour dynamique : Modèle figé après entraînement
Temps de prédiction < 1ms : Réponses instantanées	sepal_width n'est pas un bon prédicteur de sepal_length. C'est une contrainte du sujet, pas un bug du code.
200 arbres : Décisions + fiables que 100 arbres	Pas de gestion des outliers : Prédictions potentiellement erronées sur valeurs extrêmes
API RESTful & Dockerisation : Intégration facile dans n'importe quel système	Pas de persistance des prédictions : Historique non sauvegardé

6-Démonstration

Testez moi ici !

7-Rôles

Rôle	Nom
API	Sacha
Modèle	Adrien
Data Preparation	Kamel
Front	Josh
Présentation	Kamel/Adrien