

Projet de Hackathon — Intelligence Artificielle au service de la Maladie Rénale Chronique (CKD)

Bootcamp Cohorte 1 : Hackathon IA

1. Contexte général

La maladie rénale chronique (CKD) est un problème de santé publique majeur au Bénin. Les patients sont souvent diagnostiqués tardivement, ce qui réduit fortement les chances d'un traitement efficace et augmente le risque de complications graves, y compris l'insuffisance rénale terminale nécessitant une dialyse. Les centres spécialisés sont peu nombreux et les ressources médicales limitées, ce qui rend la prévention et la détection précoce particulièrement cruciales.

Dans ce contexte, l'intelligence artificielle (IA) peut jouer un rôle clé. En analysant des données cliniques, biologiques, sociales et géographiques des patients, l'IA permet de prédire les stades de la maladie, d'identifier les patients à haut risque et de proposer des priorités pour les interventions sanitaires. L'objectif de ce projet est donc de créer un outil complet permettant de passer d'une approche réactive à une approche proactive de la gestion de la CKD.

2. Objectifs du projet

L'objectif principal est de développer un modèle capable de prédire, avec précision, les stades de la maladie rénale chronique pour chaque patient. Ce modèle servira à identifier les patients les plus à risque et à orienter les ressources limitées vers ceux qui en ont le plus besoin.

Les objectifs secondaires sont les suivants :

1. Identifier les facteurs qui favorisent la progression de la maladie, comme l'hypertension, le diabète ou certains comportements de vie.
2. Calculer un score de risque pour chaque patient, permettant de classer les patients selon leur urgence ou priorité.
3. Cartographier les patients à risque pour identifier les zones géographiques nécessitant des interventions ciblées.

4. Créer un outil interactif pour les médecins, combinant le modèle prédictif, les explications des facteurs de risque et la cartographie des patients à risque.

3. Données disponibles

Le projet utilisera le dataset fourni par le **CNHU/HKM**, disponible en ligne via Google Sheets. Ce dataset contient des informations sur plusieurs centaines de patients atteints ou suspects de CKD, comprenant :

- Des données sociodémographiques : âge, sexe, profession, département.
- Des antécédents médicaux : hypertension, diabète, maladies cardiovasculaires.
- Des habitudes de vie : consommation de tabac, d'alcool, alimentation.
- Des mesures biologiques : créatinine, urée, albumine, protéinurie, hématurie.
- Des paramètres physiologiques : tension artérielle, diurèse, pouls.
- Des informations géographiques : département ou commune de résidence.

Ces données permettront de prédire le stade de CKD, d'identifier les facteurs de risque les plus importants et de générer des cartes de priorisation des patients.

4. Méthodologie

4.1 Analyse exploratoire

La première étape consiste à explorer les données pour comprendre leur structure, détecter les valeurs manquantes, les anomalies et les distributions des différentes variables. Cette analyse permettra également de visualiser les corrélations entre facteurs de risque et stades de CKD, ce qui est essentiel pour la construction de modèles robustes.

4.2 Prétraitement

Avant de construire les modèles, les données doivent être nettoyées et préparées. Les valeurs manquantes seront imputées par des méthodes adaptées, comme la moyenne ou la médiane pour les variables continues, et des techniques d'encodage seront utilisées pour les variables catégorielles. Les variables numériques seront standardisées pour améliorer la performance des modèles.

4.3 Modélisation

4.3.1 Prédiction des stades CKD

Les stades de la maladie seront calculés à partir des mesures biologiques disponibles, notamment la créatinine et l'albuminurie, selon les standards médicaux internationaux. Un modèle de machine learning multi-classe, tel que Random Forest ou XGBoost etc..., sera entraîné pour prédire les stades de CKD en utilisant l'ensemble des variables disponibles.

4.3.2 Score de risque

En parallèle, un score de risque composite sera calculé pour chaque patient. Ce score combinera les facteurs cliniques, biologiques et sociaux pour estimer le risque global d'évolution vers une forme sévère de la maladie. Il permettra de classer les patients en catégories faible, moyen ou élevé, facilitant la priorisation des interventions.

4.3.3 Analyse des facteurs déterminants

Pour rendre le modèle interprétable, des techniques comme SHAP ou LIME seront utilisées. Elles permettront de visualiser l'influence de chaque facteur sur le risque individuel d'un patient, ce qui est particulièrement utile dans un contexte où les ressources pour des examens complémentaires sont limitées.

4.4 Cartographie

Les données géographiques seront utilisées pour créer des cartes des patients à risque. Chaque département ou commune pourra être coloré en fonction du niveau de risque moyen, permettant aux autorités sanitaires de planifier des interventions ciblées et efficaces.

4.5 Outil interactif

Enfin, un prototype interactif sera développé avec FastAPI, Flask ou Streamlit. Il permettra de saisir les informations d'un patient et de générer en temps réel :

- Le stade de CKD prédictif
- Le score de risque
- Les facteurs principaux contribuant au risque
- Une carte interactive montrant la localisation du patient et les zones à risque
- Cet outil fournira aux médecins un support concret pour la prise de décision.

5. Livrables

1. **Code source complet** : scripts de nettoyage, modélisation, visualisation et déploiement du prototype.
2. **Rapport technique détaillé** : méthodologie, résultats, interprétation et recommandations.
3. **Prototype interactif** : API ou dashboard pour saisir des données et obtenir le stade CKD, le score de risque et les explications.
4. **Cartographie** : cartes des zones à risque pour planification des interventions.
5. **Présentation finale** : démonstration vidéo et slides pour présenter le projet et son impact.

6. Valeur ajoutée

Ce projet permet :

- Une **détection précoce** des patients à risque.
- Une **priorisation intelligente** des patients pour optimiser l'utilisation des ressources limitées.
- Une **explicabilité complète** pour que les médecins comprennent pourquoi un patient est à risque.
- Une **cartographie pratique** pour planifier des campagnes de prévention ciblées.

7. Contraintes et recommandations

- Respect strict de la confidentialité des données patients.
- Utilisation de modèles interprétables.
- Technologies recommandées : Python, scikit-learn, TensorFlow, Pandas, NumPy, Matplotlib, Seaborn, FastAPI/Flask.
- Respect du calendrier du hackathon : début le 27/03/2025, fin le 04/04/2025, présentation le 05/04/2025.

8. Critères d'évaluation

- Qualité des modèles : précision, robustesse, interprétabilité.
- Originalité : approche novatrice et créative.
- Faisabilité : adaptation au contexte médical réel.
- Communication : clarté et structure du rapport et de la présentation.

9. Planning recommandé

Étape	Description
Exploration et nettoyage des données	Analyse, traitement des valeurs manquantes
Construction des modèles et entraînement	Prédiction des stades et score de risque
Analyse des facteurs et visualisation	Explicabilité, cartes des zones à risque
Développement du prototype interactif	Dashboard ou API
Rapport et présentation finale	Documentation et slides pour démonstration

10. Résultats attendus

- Classification précise des stades CKD pour chaque patient.
- Score de risque compréhensible et utilisable par les médecins.
- Carte des zones à risque pour planifier les interventions.
- Priorisation proactive des patients pour dépistage et suivi.
- Prototype interactif opérationnel pour la décision médicale.

Liens vers des articles, bases de données, bibliothèques et autres ressources utiles pour la réalisation du projet :

- Base de données disponible pour exploitation :

<https://docs.google.com/spreadsheets/d/1a4iZPf93nLejpL7d7LYnF9JiIV1OetsJS8ia-bw1MXg/edit?usp=sharing>

- Documentation Python : <https://docs.python.org/3/>

- TensorFlow : https://www.tensorflow.org/api_docs

- Scikit-learn : <https://scikit-learn.org/stable/>

- Pandas : <https://pandas.pydata.org/docs/>

- NumPy : <https://numpy.org/doc/>

- Matplotlib : <https://matplotlib.org/stable/index.html>

- Seaborn : <https://seaborn.pydata.org/>
- FastAPI : <https://fastapi.tiangolo.com/#interactive-api-docs>
<https://devdocs.io/fastapi/>
- Flask : <https://flask.palletsprojects.com/en/stable/>

Bonne Chance !

Ici, c'est l'AMA !