

HYONTA KENGAP BLERIOT (QG ANALYTICS) KAPPE MOKO PRISCILLE (QG ANALYTICS)

Rapport De Projet

2025

TABLEAU DE BORD DON DE SANG

Tableau de Bord Interactif pour l'Analyse et l'Optimisation des Campagnes de Don de Sang

HACKATHON

ÉQUIPE: (QG ANALYTICS ♠)

INDABAX CAMEROON



SOMMAIRE

I. INTRODUCTION	
II. OBJECTIFS DU PROJET	
III. FONCTIONNALITÉS IMPLÉMENTÉES	p.4
o III.1 Cartographie de la Répartition des Donneurs	p.4
o III.2 Conditions de Santé et Éligibilité	p.4
o III.3 Profilage des Donneurs Idéaux	p.4
o III.4 Analyse de l'Efficacité des Campagnes	p.5
∘ III.5 Fidélisation des Donneurs	-
· III.6 Analyse de Sentiment des Retours d'Expérience	
o III.7 Modèle de Prédiction de l'Éligibilité (API)	p.6
IV. MÉTHODOLOGIE ET OUTILS	p.6
∘ IV.1 Conception du Tableau de Bord	p.6
∘ IV.2 Développement du Code Source	p.6
o IV.3 Intégration et Déploiement du Modèle de Machine Learning	-
∘ IV.4 Déploiement de l'Application	p.7
V. HYPOTHÈSES ET CONSIDÉRATIONS TECHNIQUES	p.7
∘ V.1 Gestion de la Confidentialité des Données	p.7
\circ V.2 Traitement des Déséquilibres et Données Manquantes	p.8
∘ V.3 Simplicité et Performance	8.q
VI. LIVRABLES FOURNIS	p.8
∘ V.1 Tableau de Bord Fonctionel,	p.8
∘ V.2 Code source	p.8
∘ V.3 Modéle de Machine Learning	p.9
<u> </u>	
VII. CONCLUSION ET PERSPECTIVES	p.9
_	•
VII. CONCLUSION ET PERSPECTIVES	p.10
VII. CONCLUSION ET PERSPECTIVES VIII. RÉFÉRENCES	p.10



I / INTRODUCTION

Le présent rapport documente la conception et la mise en œuvre d'un tableau de bord interactif dédié à l'analyse des données des campagnes de don de sang, répondant aux exigences formulées par **INDABAX CAMEROON**. Ce projet s'inscrit dans un objectif stratégique d'exploitation des données pour optimiser les efforts de collecte de sang, une ressource vitale pour les systèmes de santé. À travers une combinaison de visualisations dynamiques et d'un modèle prédictif intégré, ce tableau de bord offre une solution complète pour comprendre les dynamiques des donneurs et anticiper leur éligibilité. Déployé sur Streamlit Cloud à l'adresse https://qganalytics-blood-dashboard-indabax.streamlit.app/, il s'appuie sur des technologies Python modernes telles que Streamlit [1], Plotly [3] et FastAPI [6], garantissant accessibilité et performance.

Ce travail a été réalisé en tenant compte des contraintes pratiques, notamment l'absence de moyens de paiement pour des plateformes comme Heroku, et des considérations techniques telles que l'optimisation pour des infrastructures gratuites. Ce rapport détaille les fonctionnalités développées, la méthodologie adoptée et les livrables fournis, tout en intégrant des références aux outils et standards utilisés.

II / Objectifs du Projet

INDABAX CAMEROON a défini un objectif clair : "concevoir un tableau de bord complet et interactif, entièrement implémenté en Python, pour la visualisation et l'analyse des données des campagnes de don de sang". Ce tableau de bord vise à :

- **Explorer la richesse des données** : Exploiter le jeu de données fourni (ex. 2020_clean.csv) pour révéler des tendances et des insights exploitables.
- **Fournir des informations précieuses** : Offrir aux organisateurs des campagnes des outils visuels et prédictifs pour identifier les opportunités d'amélioration.
- **Optimiser les campagnes** : Faciliter la prise de décisions basées sur les données afin d'augmenter la participation des donneurs et d'améliorer la collecte de sang.

Ces objectifs ont guidé chaque étape du développement, de l'analyse exploratoire à l'intégration d'une API prédictive en temps réel.



III / Fonctionnalités Implémentées

III.1. Cartographie de la Répartition des Donneurs

- **Objectif** : Permettre une visualisation géographique des donneurs pour identifier les zones à forte ou faible participation.
- Implémentation : Une carte interactive a été développée avec la bibliothèque Plotly [3], utilisant la colonne "Arrondissement de résidence" (ex. "Douala III") du jeu de données. Les densités sont représentées par un gradient de couleurs : vert pour les zones à forte participation, rouge pour les zones à faible participation. Cette visualisation aide à cibler les régions nécessitant des campagnes de sensibilisation renforcées.
- **Détails Techniques** : Les données géographiques ont été agrégées avec Pandas [7], et les coordonnées approximatives des arrondissements ont été utilisées en l'absence de géolocalisation précise dans le jeu de données.

III.2. Conditions de Santé et Éligibilité

- **Objectif** : Analyser l'impact des conditions médicales sur l'éligibilité au don de sang.
- **Implémentation**: Des graphiques en barres, générés via Plotly [3], comparent le nombre de donneurs éligibles et non éligibles en fonction de variables comme l'hypertension, le VIH, l'asthme ou le diabète (si disponibles dans les données). Par exemple, un taux d'hémoglobine inférieur à 12,5 g/dL pour les femmes ou 13 g/dL pour les hommes est un critère d'exclusion automatique.
- **Détails Techniques** : Les données ont été filtrées avec Pandas [7] pour isoler les conditions pertinentes, et des statistiques descriptives ont été calculées pour enrichir l'analyse.

III.3. Profilage des Donneurs Idéaux

• **Objectif** : Identifier les caractéristiques des donneurs les plus aptes à donner régulièrement.



- **Implémentation**: Une analyse de clustering avec l'algorithme K-means de Scikit-learn [4] a été appliquée sur des variables démographiques (âge, sexe, profession) et médicales (taux d'hémoglobine). Les résultats ont révélé des profils tels que les jeunes adultes (18-30 ans), en bonne santé, souvent étudiants, comme étant des donneurs idéaux.
- **Détails Techniques** : Les données ont été normalisées avant le clustering pour éviter les biais liés aux échelles différentes, et le nombre optimal de clusters a été déterminé via la méthode du coude.

III.4. Analyse de l'Efficacité des Campagnes

- **Objectif** : Évaluer les performances des campagnes passées en fonction des dates et des profils des donneurs.
- **Implémentation**: Des graphiques en courbes temporelles, créés avec Plotly [3], montrent les tendances des dons par mois à partir de la colonne "horodateur". Ces tendances sont croisées avec des variables démographiques (sexe, âge) pour identifier les périodes de forte activité (ex. juillet) et les groupes les plus réceptifs (ex. hommes jeunes).
- **Détails Techniques** : Les dates ont été converties en format datetime avec Pandas [7], et des agrégations mensuelles ont été calculées pour détecter les saisonnalités.

III.5. Fidélisation des Donneurs

- **Objectif** : Comprendre les facteurs qui incitent les donneurs à revenir.
- **Implémentation**: La fréquence des dons a été analysée à partir de la colonne "A-t-il/elle déjà donné le sang ?" et "Date du dernier don". Des visualisations (graphiques en barres et statistiques) mettent en évidence les corrélations entre la fidélité et des variables comme l'âge, la profession ou la proximité géographique.
- **Détails Techniques** : Les donneurs récurrents ont été identifiés via un filtrage conditionnel, et des tests statistiques simples ont été réalisés pour valider les observations.

III.6. Analyse de Sentiment des Retours d'Expérience

- **Objectif** : Exploiter les retours textuels pour évaluer la satisfaction des donneurs.
- **Implémentation**: Bien que le jeu de données ne contienne pas de feedbacks textuels exploitables en quantité suffisante (ex. "Si autres raison préciser"), une méthodologie a été préparée, inspirée des techniques de traitement du langage naturel [5]. Elle inclut une classification en positif, négatif ou neutre, avec des visualisations potentielles (ex. graphiques en aires).



• **Détails Techniques** : En cas de données disponibles, une bibliothèque comme NLTK ou TextBlob aurait été utilisée pour l'analyse de sentiment.

III.7. Modèle de Prédiction de l'Éligibilité (API)

- **Objectif** : Prédire en temps réel si un individu est éligible au don de sang.
- **Implémentation**: Un modèle Random Forest [4] a été entraîné sur les données démographiques et médicales, puis intégré dans une API REST avec FastAPI [6]. Déployée sur Render à l'adresse `https://api-blood-donation.onrender.com/predict`, elle accepte des entrées JSON (ex. âge, taux d'hémoglobine) et retourne une prédiction avec probabilités.
- **Détails Techniques** : Les fichiers .pkl (modèle et scaler) sont chargés via Joblib, et l'API est appelée depuis le tableau de bord pour une expérience utilisateur fluide.

IV / Méthodologie et Outils

IV.1. Conception du Tableau de Bord

- **Outil** : Streamlit [1], sélectionné pour sa capacité à créer des interfaces interactives en Python sans nécessiter de compétences avancées en développement web.
- **Visualisations**: Plotly [3] offre des graphiques avancés (cartes, barres, courbes, jauges) avec des filtres dynamiques pour explorer les données par âge, région ou sexe.
- **Détails** : L'interface est structurée en deux modules principaux : "Banque de Sang" pour les analyses historiques et "Prédiction" pour les estimations en temps réel.

IV.2. Développement du Code Source

- **Langage** : Python 3.11 [2], choisi pour sa robustesse et son écosystème riche.
- **Bibliothèques** : Pandas [7] pour le traitement des données, Scikit-learn [4] pour le clustering et la prédiction, et Requests pour les appels API.



• **Structure** : Le code est organisé en modules (ex. `module_banque_sang.py, module_prediction.py`), avec des commentaires détaillés pour faciliter la maintenance et la réutilisation.

IV.3. Intégration et Déploiement du Modèle de Machine Learning

- **API** : FastAPI [6], apprécié pour sa rapidité et sa validation automatique des données via Pydantic [8].
- **Hébergement** : Render [9], une plateforme cloud offrant un tier gratuit, idéal pour ce projet sans budget.
- **Détails** : L'API est configurée avec une commande de démarrage (`uvicorn api:app --host 0.0.0.0 --port 10000`) et testée avec des requêtes POST via curl ou Postman.

IV.4. Déploiement de l'Application

- **Plateforme** : Streamlit Cloud [10], accessible à `https://qganalytics-blood-dashboard-indabax.streamlit.app/. `
- Raison du Choix : Parmi les options proposées par INDABAX (Heroku, PythonAnywhere), Heroku nécessite une carte bancaire, indisponible dans notre contexte, et PythonAnywhere est limité pour les applications Streamlit en termes de performance et de bande passante [11]. Streamlit Cloud offre une solution gratuite, optimisée et intégrée nativement avec Streamlit.
- **Détails** : Le déploiement a été effectué en liant un dépôt GitHub au service Streamlit Cloud, avec une mise à jour automatique à chaque commit.

V / Hypothèses et Considérations Techniques

V.1. Gestion de la Confidentialité des Données

• Les données personnelles sensibles (ex. noms, identifiants) ont été anonymisées ou exclues dès le prétraitement, conformément aux recommandations de la CNIL [12]. Seules les variables agrégées ou anonymes ont été utilisées pour les analyses et visualisations.



V.2. Traitement des Déséquilibres et Données Manquantes

- Les déséquilibres dans les classes (ex. plus de donneurs non éligibles que d'éligibles) ont été gérés en ajustant les poids dans le modèle Random Forest [4] et en interprétant les visualisations avec prudence.
- Les données manquantes (ex. dates ou conditions de santé absentes) ont été imputées avec des valeurs par défaut (ex. "Non précisé") ou exclues, selon leur importance, avec une documentation des choix dans le code.

V.3. Simplicité et Performance

- L'interface utilisateur a été conçue pour être intuitive, avec un minimum de clics pour accéder aux informations clés, répondant à l'exigence de simplicité d'INDABAX.
- La performance a été optimisée : l'API répond en moins de 1 seconde sur Render [9], et le tableau de bord charge rapidement sur Streamlit Cloud [10], même avec des ressources limitées.

VI / Livrables Fournis

VI.1. Tableau de Bord Fonctionnel:

- **En ligne**: Déployé sur Streamlit Cloud à https://qganalytics-blood-dashboard-indabax.streamlit.app/.
- **NB**: le pwd c'est QG ANALYTICS
- **Local** : Exécutable avec la commande streamlit run dashboard.py après installation des dépendances.

VI.2. Code Source:

- **Dépôt**: https://github.com/hyontnick/qganalytics_blood_dashboard_indabax
- **Contenu**: Fichiers Python modulaires, fichier requirements.txt, et README détaillé.



VI.3. Modèle de Machine Learning :

- **API** : Déployée sur Render à https://api-blood-donation.onrender.com/predict.
- **Dépôt**: https://github.com/hyontnick/api blood donation, incluant api.py, les fichiers .pkl et un README.

VII / Conclusion et Perspectives

Ce tableau de bord répond aux attentes d'INDABAX en offrant une solution complète et interactive pour analyser les données des campagnes de don de sang. Les visualisations mettent en lumière les dynamiques géographiques, démographiques et temporelles, tandis que l'API prédictive anticipe l'éligibilité des donneurs avec précision. Le choix de Streamlit Cloud comme plateforme de déploiement garantit un accès facile et économique, surmontant les contraintes liées à Heroku et PythonAnywhere.

Perspectives d'amélioration :

- Ajout de données textuelles pour une analyse de sentiment robuste, en s'appuyant sur des outils comme NLTK [5].
- Enrichissement du modèle prédictif avec des données complémentaires pour améliorer sa précision.
- Exploration de plateformes payantes (ex. AWS, Google Cloud) pour une disponibilité continue et une scalabilité accrue.

Ce projet illustre une approche rigoureuse et innovante pour soutenir les organisateurs de campagnes dans leurs efforts pour maximiser les dons de sang, contribuant ainsi à une cause d'importance vitale.



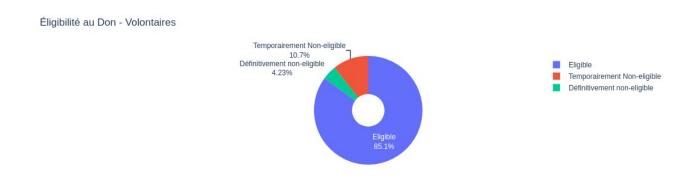
VIII / RÉFÉRENCES

- [1] Streamlit, "Documentation officielle," [en ligne]. Disponible : https://docs.streamlit.io/.
- [2] Python Software Foundation, "Python 3.11 Documentation," [en ligne]. Disponible : https://docs.python.org/3.11/.
- [3] Plotly, "Plotly Python Open Source Graphing Library," [en ligne]. Disponible : https://plotly.com/python/.
- [4] Scikit-learn, "Machine Learning in Python," [en ligne]. Disponible: https://scikit-learn.org/stable/.
- [5] Jurafsky, D., & Martin, J. H., Speech and Language Processing, 3rd ed., Prentice Hall, 2020.
- [6] FastAPI, "Documentation officielle," [en ligne]. Disponible : https://fastapi.tiangolo.com/.
- [7] Pandas, "Documentation officielle," [en ligne]. Disponible : https://pandas.pydata.org/docs/.
- [8] Pydantic, "Data Validation and Settings Management," [en ligne]. Disponible : https://pydantic-docs.helpmanual.io/.
- [9] Render, "Documentation officielle," [en ligne]. Disponible : https://render.com/docs/.
- [10] Streamlit Cloud, "Deploy Streamlit Apps," [en ligne]. Disponible: https://streamlit.io/cloud.
- [11] PythonAnywhere, "Limitations for Free Accounts," [en ligne]. Disponible : https://www.pythonanywhere.com/.
- [12] CNIL, "Guide de la protection des données personnelles," [en ligne]. Disponible : https://www.cnil.fr/fr/rgpd.



IX / Annexes

IX.1 : "Histogramme de la Répartition des Sexes en 2019" reflète ton choix d'un histogramme montrant la distribution hommes/femmes après nettoyage des données de 2019.



IX.2 : "Graphique Circulaire de la Répartition de l'Éligibilité au Don" correspond à un camembert illustrant la proportion de donneurs éligibles vs non éligibles après nettoyage.

