PORTFOLIO

DATA SCIENCE

NASSER CHAOUCHI

GITHUB GMAIL LINKEDIN

Who am I?

My academic journey

I'm Nasser, a French computer science engineer passionate about artificial intelligence, data, and innovation. I graduated from UTC (Université de Technologie de Compiègne) in 2025 with a major in AI and Data Science.

I completed:

- A dual-focused internship at Numberly as a Data Engineer and Project Manager, combining technical and management responsibilities.
- An exchange semester at the Escuela de Ingeniería y Arquitectura in Zaragoza (Spain), as part of a Data Science Master's program.
- My final-year internship at Ubisoft as a Data Scientist, working on real-world game data and predictive models.

Strengths

I would describe myself as:
Curious, Rigorous, Positive & Patient.

Interests

Artificial Intelligence, Sports, Literature, and Chess.

My work experience

An **internship** at **Ubisoft** from **October 2024 to March 2025** as a **Data Scientist**, with the following missions:

Audiences Understanding | **Segmentation Based on Players' Profiles** | **Player Behavior Prediction**

I worked on the game **Avatar: Frontiers of Pandora**. My role was to understand the **game's underperformance** and to **identify and target** potential players within the **Ubisoft ecosystem** who would most likely acquire the game.

The project was divided into three main phases:

Ad-hoc Analyses → **Clustering Development** → **Classifier with Prediction**

You can contact the team manager for a reference:

- Nicolas Tatin, Associate Director, Data & Analytics

Data Analysis | Data Science

My projects - The Movie Recommender System

CONTEXT

- **Dataset**: MovieLens 32M
- **Goal**: Recommend movies users might like, based on behavior and content
- **Type**: Hybrid Recommendation System
  - Collaborative Filtering (ratings)
  - Content-Based Filtering (genres, titles)
- **Size**: 32M+ ratings, ~270k users, 62k movies

TOOLS USED

- Scikit-learn
- Pandas, NumPy
- Seaborn
- Matplotlib
- HuggingFace datasets
- Streamlit

Open the repository

Open the interface
(with MovieLens 1M)

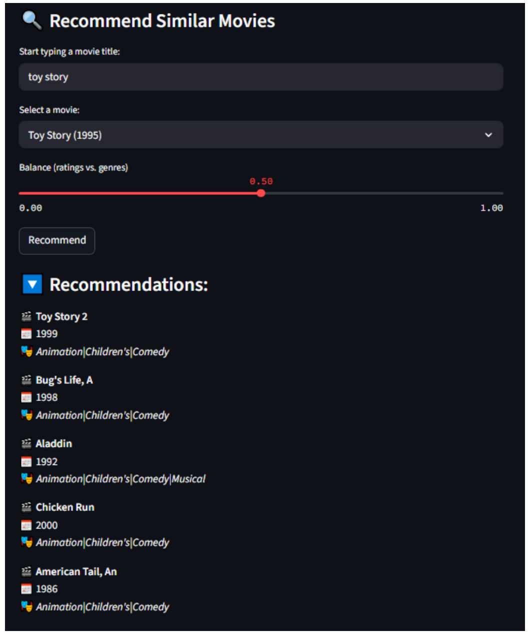
APPROACH

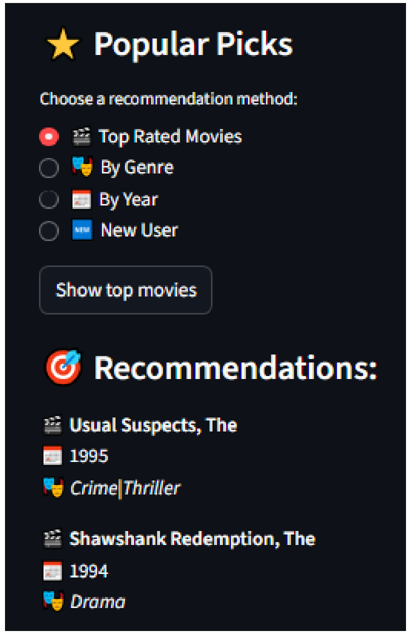
- **Data Cleaning**: Merged movies.csv and ratings.csv, extracted year, processed genres
- **Collaborative Filtering**: Built user-item matrix, applied cosine similarity
- **Content-Based Filtering**: Used TF-IDF/CountVectorizer on genres and titles
- **Hybrid Strategy**: Combined top recommendations from both approaches
- **Implemented multiple strategies**: Most rated movies, Top-rated by genre, Top-rated by year, User-user collaborative hybrid, Item-item collaborative hybrid
- **Profile-Based Recommendation**: Built a user profile from favorite movies to generate personalized suggestions

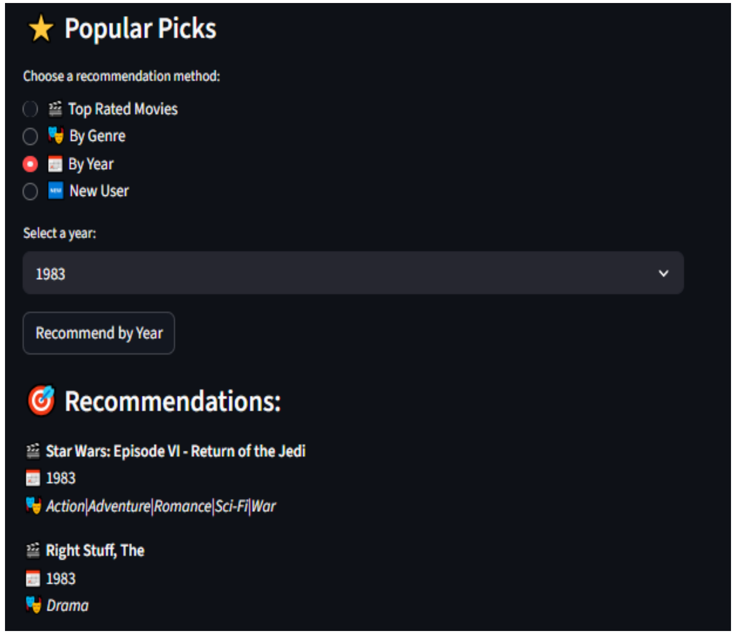
WHAT I LEARNED

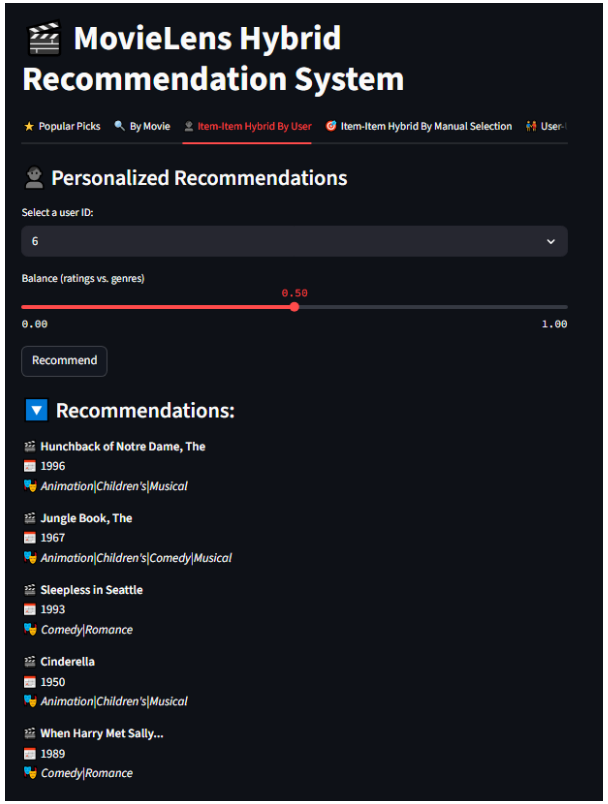
- Designing and comparing recommender strategies
- Using similarity metrics (cosine) on sparse data
- Evaluating trade-offs between relevance and diversity

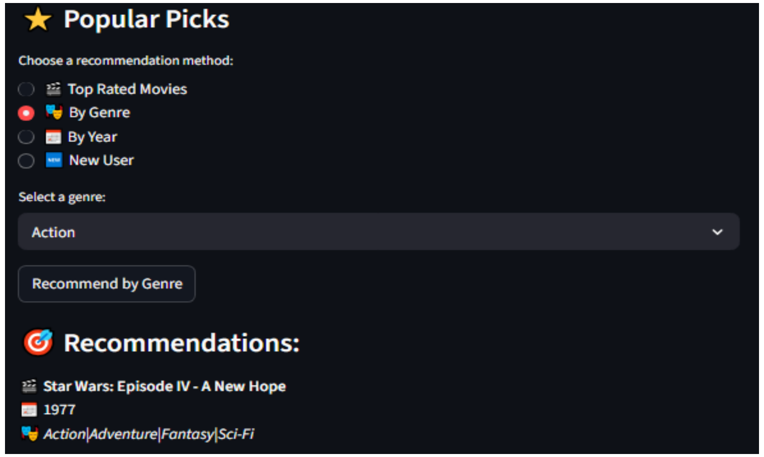
My projects - The Movie Recommender System

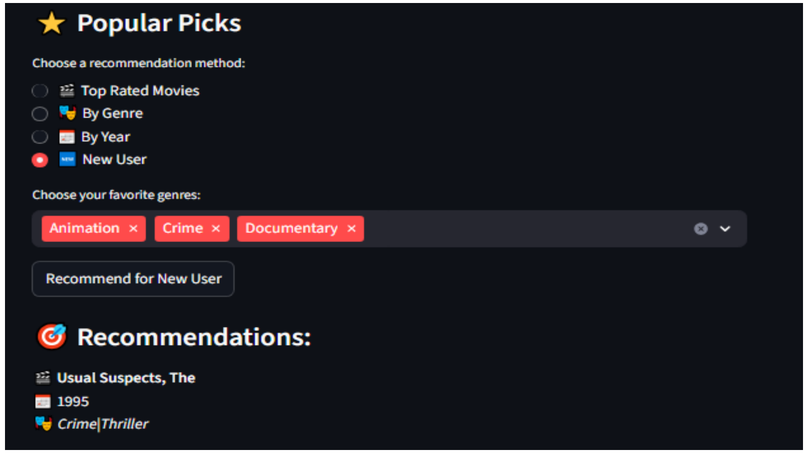






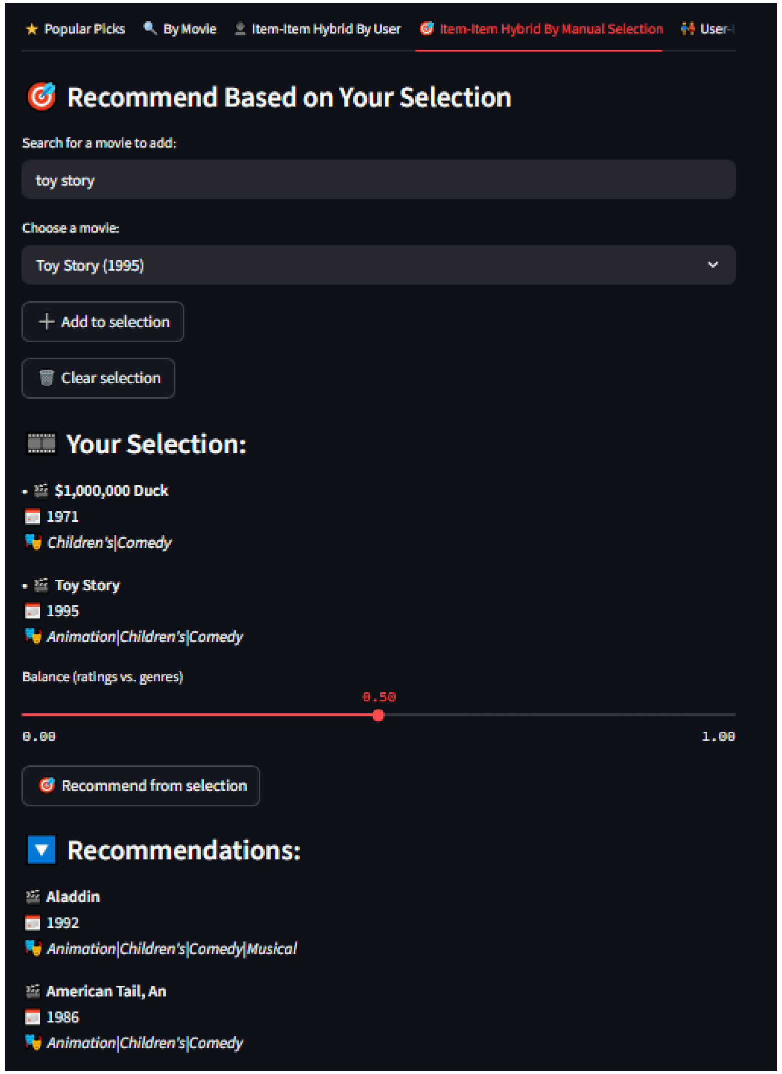


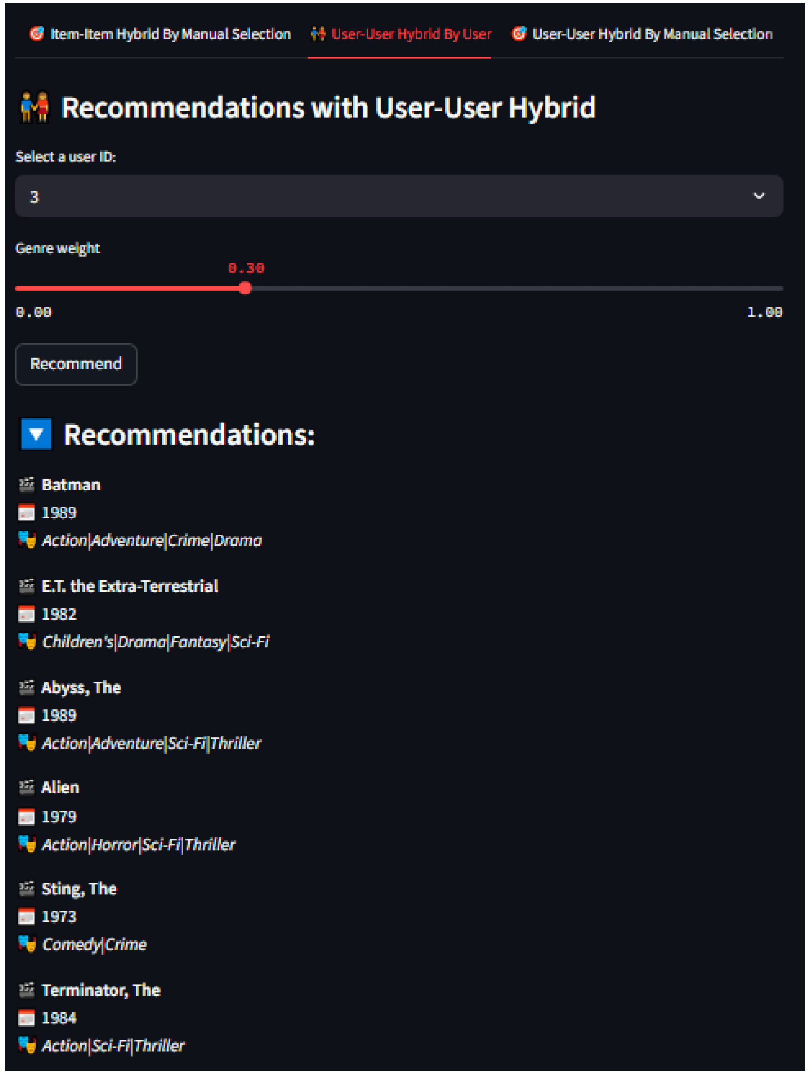


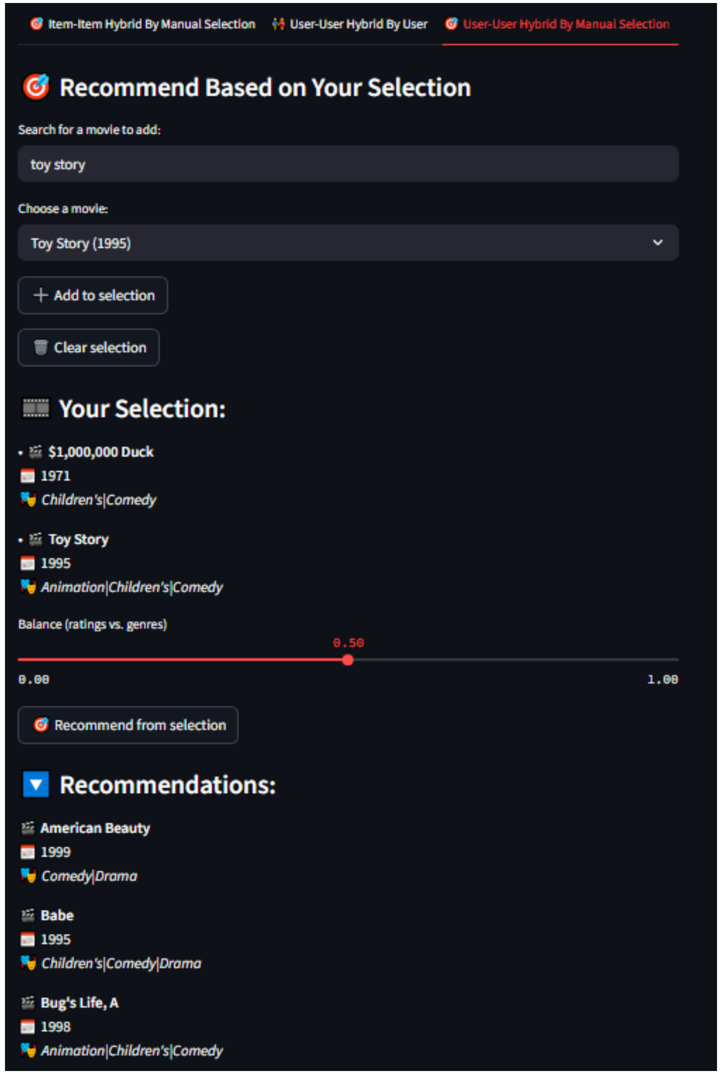


Nasser CHAOUCHI

My projects - The Movie Recommender System







My projects - Multiclass Classification for Diabetes

CONTEXT

- **Dataset**: Multiclass Diabetes Dataset
- **Goal**: Classify patients into several diabetes stages
- **Type**: Supervised, Multiclass classification
- **Size**: 264 patients, 12 features

TOOLS USED

- Scikit-learn
- Pandas, NumPy
- Seaborn
- Matplotlib
- Streamlit

Open the repository

Open the interface

APPROACH

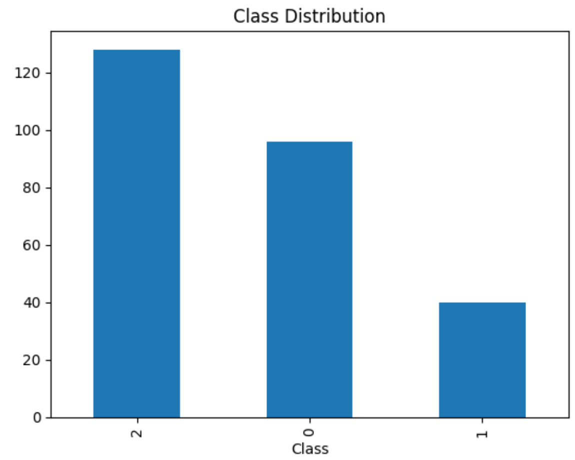
- **EDA & Preprocessing**: Analyzed feature distributions, handled missing values, balanced classes, and scaled data.
- **Model tested**: Logistic Regression, Random Forest and K-Nearest Neighbour
- **Cross Validation**: Ensured robust performance and avoided overfitting
- **Evaluation**: Confusion Matrix, Classification report (F1-Score, Accuracy, Recall)
    - Final model (Random Forest) → Accuracy: 97%
    - Macro F1-score (better suited to class imbalance): 0.98

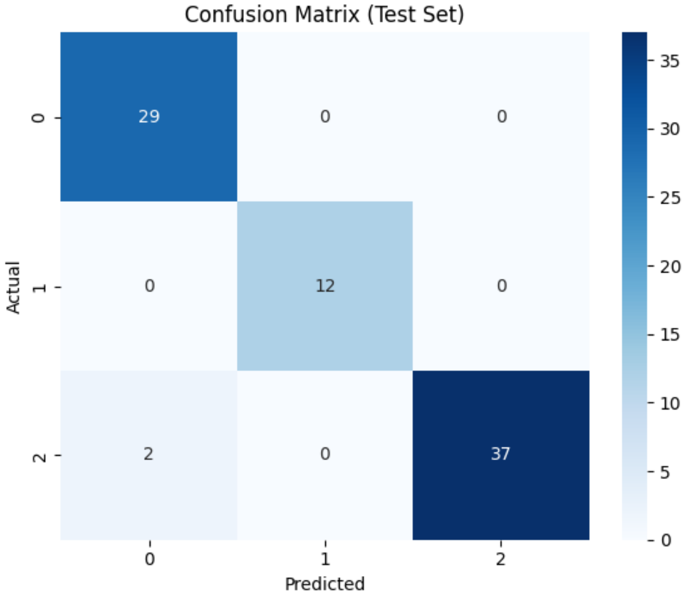
WHAT I LEARNED

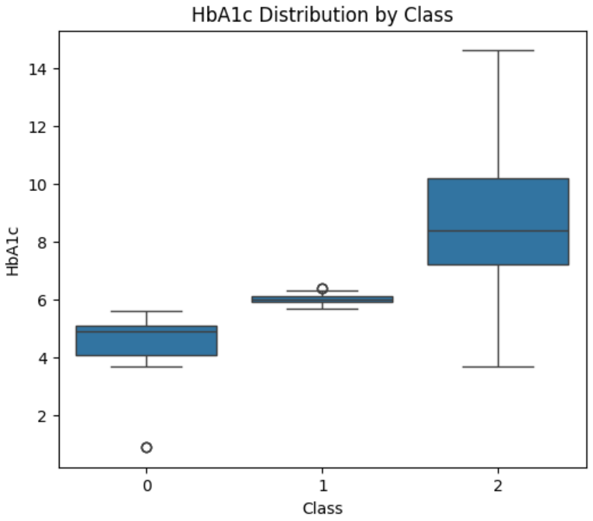
- How to handle imbalanced multiclass data
- The importance of feature engineering and model tuning
- Model explainability with SHAP or feature importance

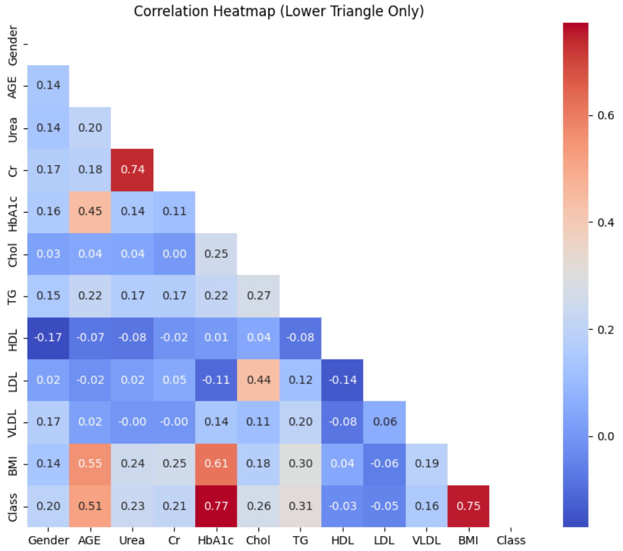
My projects - Multiclass Classification for Diabetes

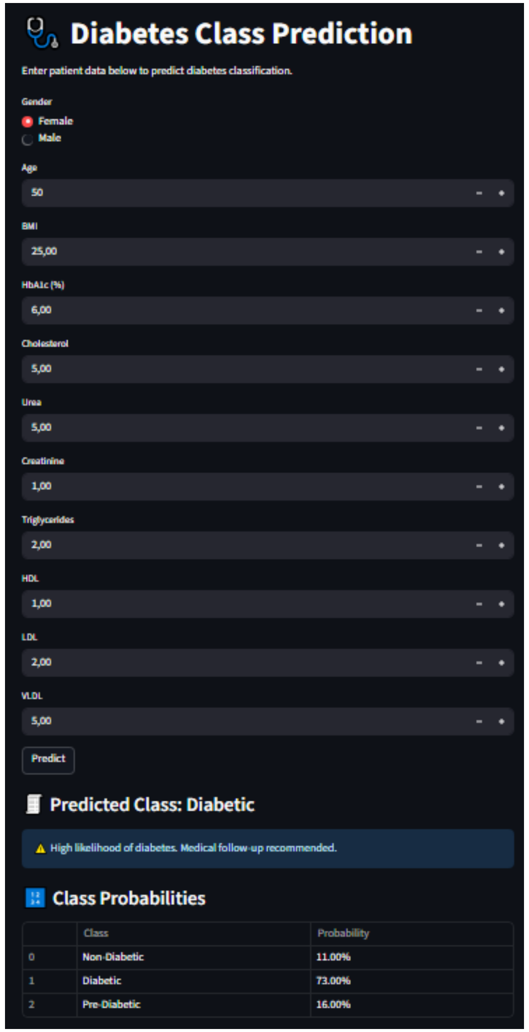
Variable	F-statistic	p-value
AGE	60.368	0.00000
BMI	190.565	0.00000
TG	14.218	0.00000
HbA1c	200.415	0.00000
Chol	9.881	0.00007
Urea	9.115	0.00015
Cr	6.466	0.00182
VLDL	3.712	0.02573
LDL	0.957	0.38526
HDL	0.488	0.61457











Portfolio

Nasser CHAOUCHI

My projects - Chronic Kidney Disease Prediction

CONTEXT

- Dataset: Kidney Disease Risk Dataset
- Goal: Predict CKD status and dialysis need based on clinical and biological data
- Type: Supervised, Binary classification (2 targets: CKD_Status, Dialysis_Needed)
- Size: 2304 patients, 9 features

TOOLS USED

- Scikit-learn
- Pandas, NumPy
- Seaborn
- Matplotlib
- XGBoost
- Streamlit

Open the repository

Open the interface

APPROACH

- EDA & Preprocessing: Explored feature relationships, handled missing values, encoded categorical data, scaled numerical features.
- Model tested: Logistic Regression, Random Forest, Gradient Boosting, XGBoost and K-Nearest Neighbour
- Cross Validation: Ensured robustness and reduced overfitting risk.
- Evaluation Classification Report, ROC-AUC, F1-Score, Accuracy
    - Best model was Gradient Boosting, but due to class imbalance, Random Forest gave more reliable results for generalization.
      → Accuracy: 100% for CKD_Status — Accuracy 100% but a F1-Score 0.97 (class imbalance) for the Dialysis_Needed
      → Separate models trained for each target

WHAT I LEARNED

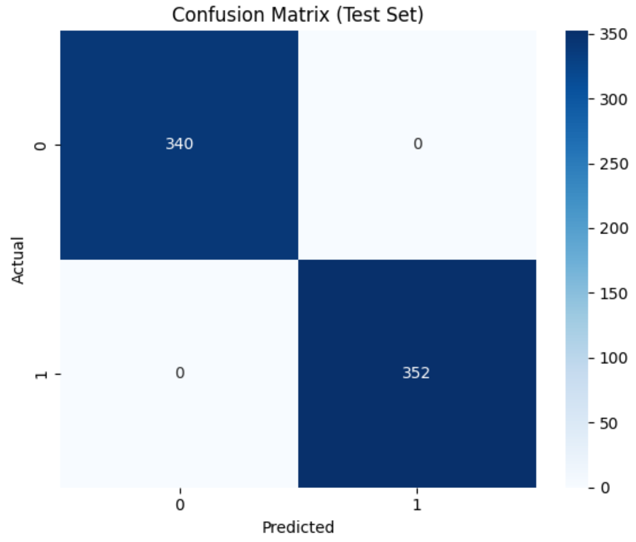
- Managing dual target classification
- Handling noisy and medical data
- Improving interpretability with SHAP values

My projects - Chronic Kidney Disease Prediction

Evaluation on Test Set – CKD_Status

Class	Precision	Recall	F1-Score	Support
0	1.00	1.00	1.00	340
1	1.00	1.00	1.00	352

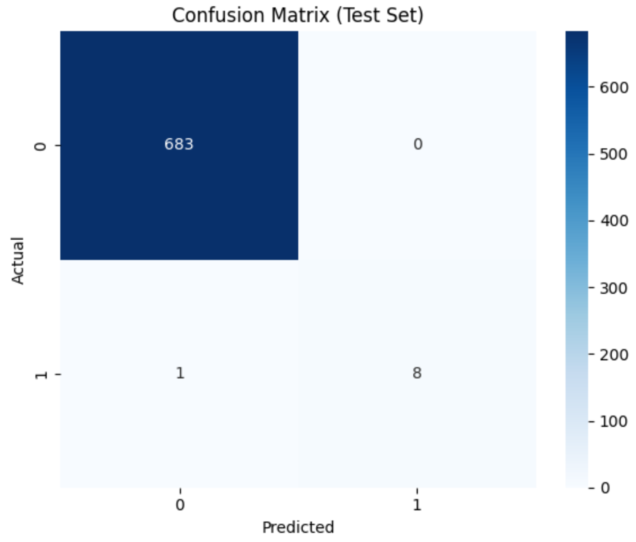
Accuracy			**1.00**	692
Macro avg	1.00	1.00	1.00	692
Weighted avg	1.00	1.00	1.00	692

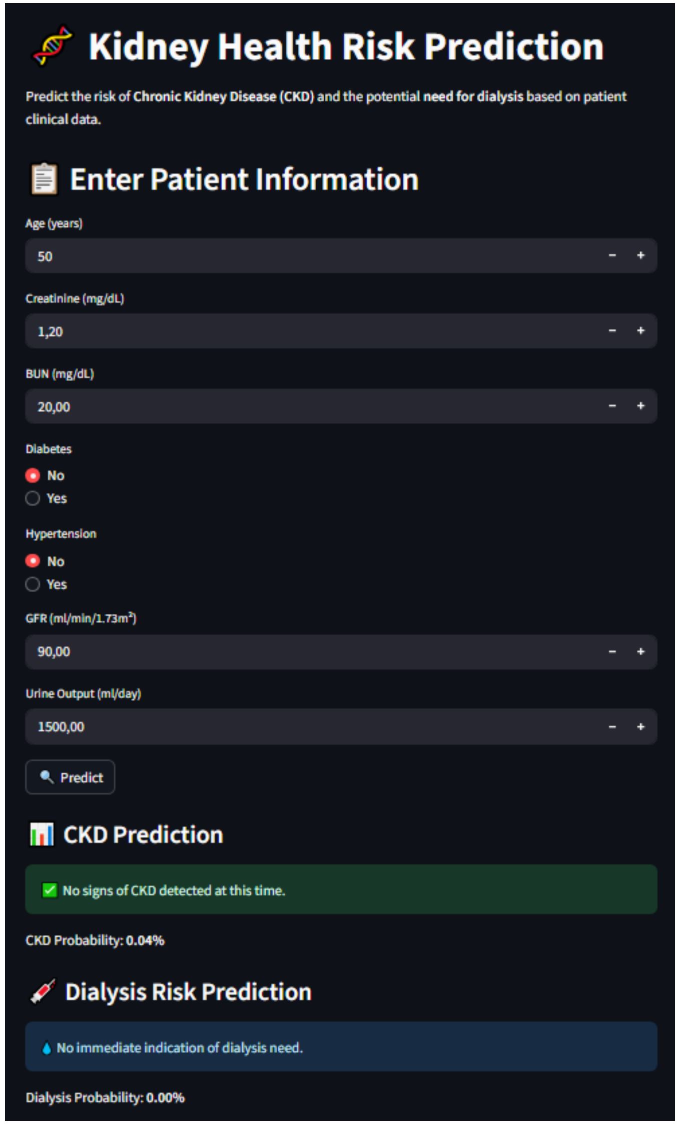


Evaluation on Test Set – Dialysis_Needed

Class	Precision	Recall	F1-Score	Support
0	1.00	1.00	1.00	683
1	1.00	0.89	0.94	9

Accuracy			**1.00**	692
Macro avg	1.00	0.94	0.97	692
Weighted avg	1.00	1.00	1.00	692





Summary of my skills & achievements

- Built 3 real-world machine learning apps
- Deployed 3 Streamlit interfaces
- Experience with pipelines (Airflow), modeling (XGBoost), and explainability (SHAP)
- Strong understanding of recommender systems, classification, EDA

Don’t hesitate to reach me out

[GITHUB](#) [GMAIL](#) [LINKEDIN](#)