Al Akhawayn University in Ifrane

School of Science and Engineering

Computer Science Department

Credit Eligibility Prediction Using Machine Learning

Application to Credit Yousser at Chaabi Bank

CSC 3347

Students:
Maria Chmite
Hind Bertit
Rayane Mestari

Supervised by:
Dr. Yousra Chtouki

Fall 2025

Contents

1 Survey of Existing Work

1.1 Summary Table of Reviewed Papers

Table 1 summarises the main papers reviewed for this project.

Table 1: Summary of reviewed papers

#	Paper title	Year	Method	Dataset	Strengths	Limitations
1	Applying CRISP-DM Methodology in Developing ML Model for Credit Risk Prediction	2023	CRISP-DM, ML	Bank loan data	Structured process framework	Limited model details
2	Harnessing AI and Predictive Analytics to Enhance Risk Assessment in Retail Banking	–	AI, predictive analytics	Retail banking	Focus on practical implementation	Missing detailed results
3	Developing Advanced ML Decision-Making Model for Banking	–	ML algorithms	Banking data	Balances risk, speed, precision	Limited deployment details
4	Data Analytics for Credit Risk Models in Retail Banking	2023	Data analytics, ML	Retail banking	Industry perspective	Lack of specific models
5	Machine Learning for Enhanced Credit Risk Assessment	2023	10 ML algorithms	2.5M+ observations	Large-scale dataset, comprehensive comparison	No CRISP-DM process
6	Credit Risk Model Based on Central Bank Credit Registry	2021	ML, data science	Credit registry	Alternative data source	Limited generalizability
7	Machine Learning for Credit Risk: Three Successful Case Histories	2022	ML algorithms	Banking data	Real-world cases	Lacks technical depth
8	ML-Based Risk Prediction Model for Loan Applications	2023	ML classification	Loan applications	Focus on default prevention	No eligibility prediction
9	Machine Learning for Credit Risk Assessment in Banking	2022	ML models	Banking systems	Overview of ML evolution	Limited empirical results
10	A Machine Learning Approach for Micro-Credit Scoring	2021	Various ML algorithms	Micro-lending data	Thin-file borrowers, inclusion focus	Small market focus
11	Credit Risk Assessment for Small Enterprises Using Kernel Feature Selection	2020	Kernel feature selection	Chinese SME data	Feature selection focus	Specific to SMEs
12	Integration of Big Data and ANN for Credit Risk Scoring in Emerging Markets	2022	ANN	Egyptian banking	Emerging market context	Limited interpretability
13	Research on ML-Based Credit Risk Prediction Models and Algorithms	2025	ML algorithms	Credit data	Recent developments overview	Lacks deployment details
14	Interpretable Selective Learning in Credit Risk	2022	Logistic regression, ML	Credit datasets	Interpretability focus	Lower accuracy trade-off
15	Loan Eligibility Prediction: Regional Variations Analysis	2023	ML classification	Urban/rural/semi-urban	Geographic segmentation	No amount prediction
16	Credit Risk Modeling with Explainable AI	2021	XAI methods	Financial institutions	Explainability techniques	Complex implementation
17	Loan Eligibility System Using Machine Learning	2024	ML classification	Loan applications	Eligibility focus	Binary classification only
18	ML and Deep Learning for Loan Prediction: Ensemble Methods	2024	Ensembles, deep learning	Banking loans	Data balancing techniques	No multi-output models

#	Paper title	Year	Method	Dataset	Strengths	Limitations
19	Application Analysis of ML Models in Credit Risk	2024	ML models	Credit risk data	Model comparison	Limited process d tails
20	ML Approach for Credit Score Predictions	2023	XGBoost, SVM ensemble	Credit products	Heterogeneous ensemble	No CRISP-DM
21	Artificial Intelligence Models Used in Credit Risk Prediction	2025	AI models	Credit data	Comprehensive AI overview	Theoretical focus
22	CRAM: Credit Risk Assessment Model	2021	Multiple ML algorithms	Financial data	Comparative analysis	Limited deploymen
23	Systematic Study on RL-Based Applications	2023	Reinforcement learning	Various domains	Broad RL overview	Tangential to cred risk
24	Harnessing ML Emerging Technology in Financial Investment	2021	ML credit rating	Credit ratings	Real-world implementation	Specific to ratings
25	ML for Enhanced Credit Risk: Mental Health Data Integration	2024	ML algorithms	Banking + mental health	Novel data integration	Privacy concerns
26	Predictive Modelling for Loan Approval: ML Approach	2024	LR, DT, RF, SVM, NB	Bank loans	Multiple algorithm comparison	No amount predi tion
27	Impact of Feature Selection and Transformation on ML Credit Scoring	2023	Feature engineering, ML	Credit scoring	Feature engineering focus	Limited model var ety
28	Comparative Analysis of ML Models for Credit Risk in Banking	2025	LR, RF, XGBoost, SVM, NN	Credit dataset	Comprehensive comparison	No CRISP-DM
29	Interpretable ML Models for Credit Risk Assessment	2024	RF, LightGBM, XGBoost, LR	Kaggle Home Credit	Interpretability emphasis	No real deployment
30	Predicting Financial Credit Risks Using ML Algorithm	–	ML algorithms	Bank credit	Credit risk focus	Minimal details

1.2 Thematic Analysis of Existing Work

Theme 1: Machine Learning Models for Credit Risk Prediction

Most studies focus on comparing machine learning methods for credit risk prediction. Paper [28] tested Logistic Regression, Random Forest, Gradient Boosting (XGBoost), Support Vector Machines, and Neural Networks. The results showed that XGBoost performed best, with 88.7% accuracy, 89.5% precision, 80.3% recall, and 91.3% AUC. Paper [20] suggested combining XGBoost and SVM in one model to improve credit score predictions. Paper [29] studied models that are easy to understand, such as Random Forest, LightGBM, and XGBoost, using the Kaggle Home Credit Default Risk dataset. It showed that using multiple models together makes predictions more stable and less affected by noise. Papers [5] and [22] also compared many models. Paper [5] looked at ten algorithms on a dataset with more than 2.5 million entries, and Paper [22] created the CRAM model to compare different approaches. Paper [18] focused on using multiple models and deep learning together with methods to balance the data to predict loan defaults.

Strengths: These studies show clear performance results, indicating that using multiple models together, especially gradient boosting methods, often works better than traditional statistical methods. They achieve AUC scores between 0.88 and 0.91. Random Forest is usually the second-best option and has the advantage of being easier to understand because it shows which features are most important. Overall, the research suggests that modern machine learning methods can predict credit risk more accurately than older credit scoring approaches.

Limitations: Most studies [5, 13, 18, 19, 20, 21, 22, 28, 29, 30] focus only on predicting whether a loan will default and do not consider deciding loan eligibility or estimating the loan amount. None of them fully follow the CRISP-DM process, including documenting business goals, data preparation steps, or deployment plans. Papers [9, 21] give theoretical explanations but do not provide experimental results. Papers [13, 19, 30] do not go into enough detail about the model design or how the hyperparameters were set.

Theme 2: CRISP-DM Methodology and Process Frameworks

Few studies focus on using structured data mining processes for credit risk systems. Paper [1] is notable for applying the CRISP-DM approach to build a machine learning model for credit risk prediction. It highlights a clear step-by-step process covering business understanding, data analysis, preparation, modeling, evaluation, and deployment. Paper [4] looked at data analytics in retail banking credit risk models and pointed out that, although advanced techniques exist, banks rarely use formal frameworks like CRISP-DM, even though these frameworks help ensure reproducibility and meet regulatory requirements.

Strengths: Paper [1] shows that using a structured methodology can lower project risks, keep business and technical goals aligned, and improve communication with stakeholders during development. The CRISP-DM framework gives clear outputs at each stage and allows iterative improvements based on evaluation results.

Limitations: Only two studies [1, 4] explicitly discuss the CRISP-DM framework, highlighting a major gap in the literature. Paper [1] focuses on applying the process framework but gives few details about the specific modeling methods, feature engineering, or performance results. Paper [4] emphasizes the value of structured approaches in theory but does not provide practical implementation guidance. The other 28 studies skip process methodology entirely and move straight to comparing models without documenting reproducible workflows.

Theme 3: Loan Eligibility and Amount Prediction

Few studies focus specifically on predicting loan eligibility. Paper [15] examined loan eligibility in urban, rural, and semi-urban areas, showing that feature relationships and creditworthiness criteria vary by region. Paper [17] developed a loan eligibility system using machine learning classification methods. Paper [26] conducted predictive modeling for loan approval, comparing logistic regression, decision trees, Random Forest, GaussianNB, and SVM algorithms.

Strengths: These studies recognize loan eligibility as an important business problem that requires targeted solutions. Paper [15] provides valuable insights into geographic differences, showing that the best models vary across urban, rural, and semi-urban populations because of different income patterns and employment characteristics. Paper [26] offers practical algorithm comparisons for approval decisions.

Limitations: All three studies [15, 17, 26] focus only on binary eligibility classification and do not address loan amount estimation. None propose models that can predict both eligibility and optimal credit amounts at the same time. Papers [17] and [26] provide limited details on feature engineering and deployment. No study integrates eligibility prediction within a CRISP-DM framework or describes implementation in real banking systems.

Theme 4: Feature Engineering and Data Preprocessing

Several studies highlight the important role of feature engineering in credit modeling. Paper [27] examined how feature selection and transformation affect machine learning credit scoring, comparing filter, wrapper, and embedded methods, and found that embedded methods in tree-based algorithms provide the best balance between performance and efficiency. Paper [11] used kernel feature selection and multiple criteria linear optimization for small enterprises, creating composite features based on business ratios and industry-specific indicators. Paper [6] used central bank credit registry data as an alternative feature source to model default probability. Paper [25] studied the integration of mental health data to improve credit risk analysis, investigating whether psychological factors could enhance prediction accuracy.

Strengths: These studies show that careful feature engineering can bring larger improvements than algorithm choice alone. Paper [27] offers a systematic comparison of feature selection methods with practical guidance. Paper [11] demonstrates the value of applying domain knowledge to create meaningful composite features. Paper [6] shows how using alternative data can reduce missing values and enrich feature sets. Paper [25] explores innovative, non-traditional features that could expand credit access for borrowers with limited credit history.

Limitations: Papers [6, 11, 27] focus mainly on feature engineering without fully developing or deploying models. Paper [25] raises privacy and ethical concerns around using mental health data, limiting its practical use. None of these studies incorporate feature engineering into a complete CRISP-DM workflow or provide reproducible preprocessing pipelines. Several papers highlight that good feature engineering has as much impact as model choice. They explore filter, wrapper and embedded feature selection methods, alternative data sources (central bank registries) and composite business indicators.

Theme 5: Interpretability and Explainability

Increasing regulatory requirements have encouraged research on interpretable credit models. Paper [14] studied selective learning for credit risk and suggested that the transparency of logistic regression might justify accepting 3–5% lower accuracy compared to complex ensemble methods in regulated settings. Paper [16] explored credit risk modeling using explainable AI methods, such as SHAP and LIME, to interpret black-box models. Paper [29] focused on interpretability in ensemble comparisons, showing that feature importance analysis in Random Forest and XGBoost can provide explanations that meet regulatory standards.

Strengths: These studies recognize that banking regulations, including ECOA, FCRA, and GDPR, require models to be interpretable for explaining adverse actions. Paper [16] provides practical guidance on implementing explainable AI. Paper [29] demonstrates that modern ensemble methods can offer both high performance and reasonable interpretability through feature attribution. Paper [14] gives valuable insights into the tradeoff between accuracy and interpretability.

Limitations: Papers [14, 16] focus mainly on interpretability and do not fully evaluate model performance or deployment. None of the studies discuss integrating explainability into production systems that handle millions of real-time predictions. Paper [16] examines XAI techniques theoretically but lacks empirical validation on banking data. Additionally, these studies do not address the computational cost of generating explanations on a scale.

Theme 6: Real-World Banking Implementation

Few studies discuss practical deployment of credit risk models in banking environments. Paper [7] presented three successful case studies, showing that machine learning can expand databases using alternative data but requires careful validation. Paper [2] explored using AI for risk assessment in retail banking, focusing on implementation challenges. Paper [3] examined developing advanced decision-making models that balance risk, speed, and precision. Paper [24] described implementing an ML credit rating model in financial investment, noting that delays in rating updates contributed to the 2008 financial crisis.

Strengths: These studies highlight the gap between research prototypes and production systems, considering real-world constraints such as latency, system integration, and regulatory compliance. Paper [7] offers concrete case studies of successful ML adoption. Paper [3] emphasizes the importance of balancing accuracy, processing speed, and risk management in operational banking systems.

Limitations: All four studies [2, 3, 7, 24] provide only high-level overviews and lack technical implementation details, such as system architecture, API design, model monitoring, or A/B testing strategies. None describe a full deployment pipeline from development to production. Papers [2, 3] do not provide empirical performance results. None address key operational concerns like handling concurrent requests, versioning models, detecting concept drift, or automating retraining

Theme 7: Handling Class Imbalance

Several studies focus on handling class imbalance in credit datasets. Paper [18] examined data balancing techniques such as SMOTE, ADASYN, random undersampling, and cost-sensitive learning for loan prediction, finding that SMOTE combined with ensemble methods improves recall for the minority class most effectively. Paper [8] developed a risk prediction model aimed at preventing defaults, showing that models without imbalance correction may have high overall accuracy but fail to detect important minority defaults. Paper [22] observed that ensemble methods with built-in class weighting often perform better than resampling-based approaches.

Strengths: These studies highlight that credit datasets usually have severe imbalance, with only 5–15% defaults, making accuracy alone an insufficient metric. They provide systematic comparisons of rebalancing methods and emphasize improving recall for the minority class. Paper [18] gives a thorough empirical evaluation of multiple balancing techniques.

Limitations: Papers [8, 18, 22] address imbalance correction separately and do not integrate it into complete credit risk systems. None discuss how class weighting affects model interpretability or regulatory compliance. The studies also lack a cost-benefit analysis that measures the business impact of different false positive and false negative rates.

Theme 8: Alternative Data and Novel Approaches

Some studies examine non-traditional data sources and methods for credit scoring. Paper [10] looked at micro-credit scoring in situations without recorded credit history, using alternative data such as utility payments and mobile usage. Paper [12] studied combining big data and artificial neural networks for credit risk scoring in Egyptian emerging markets. Paper [6] used central bank credit registry data as an alternative to standard bureau information. Paper [25] explored including mental health data. Paper [23] conducted a systematic study of reinforcement learning applications in various fields, including finance.

Strengths: These studies help expand credit access for borrowers with limited credit history by using alternative data sources. Paper [10] addresses financial inclusion for underserved populations. Paper [12] shows that machine learning can be effective in emerging markets with limited traditional credit systems. Paper [23] explores new reinforcement learning approaches that could be applied to dynamic credit decisions.

Limitations: Papers [10, 12, 25] raise privacy and regulatory concerns when using alternative data. Paper [23] focuses broadly on reinforcement learning with little specific application to credit. None of these studies offers production-ready solutions or shows how alternative data could be integrated with traditional features. Paper [25] suggests using mental health data, which faces serious ethical and legal obstacles.

1.3 Research Gaps and Project Justification

The literature review reveals four main gaps:

- **Gap 1 – Lack of complete CRISP-DM implementations:** Only Papers [1, 4] discuss the CRISP-DM methodology, with Paper [1] providing the only full implementation example. The other 28 studies focus solely on model comparisons and do not document business understanding, data collection strategies, detailed data preparation, deployment plans, or monitoring frameworks. This leaves practitioners without guidance for building end-to-end credit risk systems following industry-standard processes.
- **Gap 2 – Limited multi-output prediction:**All reviewed papers [1–30] focus on a single output, either binary default classification or loan eligibility prediction. No research proposes unified models that predict both loan eligibility and the optimal loan amount. This limits practical applicability, as banks need integrated decisions about whether to approve applications and how much to lend.
- **Gap 3 – Poor deployment documentation:** Papers [2, 3, 7, 24] mention production deployment but provide minimal technical details on system architecture, API latency optimization, handling concurrent requests, model versioning, A/B testing, performance monitoring, concept drift detection, or automated retraining. Most other studies stop at offline model validation and do not address operational deployment.
- **Gap 4 – Missing Cost-Benefit Analysis:** Few studies [4, 7, 8] evaluate the financial impact of models, including costs of false positives (rejecting creditworthy applicants), false negatives (defaults), model development, and operational infrastructure, Most research focuses only on statistical metrics without quantifying business value.

1.4 Project Contributions

This project addresses these gaps through three key contributions.

Contribution 1: Complete CRISP-DM Implementation Building on Paper [1] but providing full documentation, this project implements all CRISP-DM phases:

1. **Business Understanding** – defining goals, success criteria, and stakeholder requirements;
2. **Data Understanding** – exploratory analysis, quality assessment, and insight discovery;
3. **Data Preparation** – feature engineering, handling missing values, class imbalance, and transformations;
4. **Modeling** – systematic comparison of XGBoost, Random Forest, Logistic Regression, and Neural Networks with hyperparameter tuning;
5. **Evaluation** – assessment using accuracy, precision, recall, F1-score, AUC-ROC, and business metrics;
6. **Deployment** – production system design, monitoring, and maintenance.

This fills the documentation gap noted in Papers [5, 13, 18, 19, 20, 21, 22, 28, 29, 30].

Contribution 2: Real-World Banking Context Building on insights from Papers [2, 3, 7, 24], this project provides detailed deployment solutions, including:

- system architecture for real-time predictions,
- API latency testing and optimisation,
- model interpretability using SHAP values (Paper [16]),
- monitoring dashboards tracking prediction trends,
- an A/B testing framework for safe model updates.

This addresses the theory–practice gap identified in Papers [5, 9, 13, 21].

Contribution 3: Model Selection Justification XGBoost is selected as the primary model based on literature evidence:

1. **Superior performance** – Papers [20, 28, 29] report AUC values around 0.88–0.91;
2. **Effective class-imbalance handling** – Papers [18, 22] show XGBoost's built-in weighting manages imbalanced data well;
3. **Interpretability** – SHAP integration with XGBoost meets regulatory requirements (Papers [16, 29]);
4. **Production suitability** – fast inference, efficient memory use, and robustness to missing data.

Comparative models (Random Forest, Logistic Regression, Neural Networks) are included following best practices (Papers [5, 22, 26, 28]) to validate XGBoost performance and provide interpretability benchmarks.

1.5 References

[1] Authors. (2023). Applying CRISP-DM Methodology in Developing Machine Learning Model for Credit Risk Prediction. DOI: 10.1007/978-3-031-37963-5_37

[2] Authors. Harnessing Artificial Intelligence and Predictive Data Analytics to Enhance Risk Assessment and Credit Scoring Mechanisms in Retail Banking. Retrieved from https://helexscience.com/index.php/JDSPABDA/article/view/2024-09-04

[3] Authors. Developing an Advanced Machine Learning Decision-Making Model for Banking: Balancing Risk, Speed, and Precision in Credit Assessments.

[4] Authors. (2023). Data Analytics for Credit Risk Models in Retail Banking: a new era for the banking system. DOI: 10.47473/2020rmm0132

[5] Authors. (2023). Machine Learning for Enhanced Credit Risk Assessment: An Empirical Approach. DOI: 10.3390/jrfm16120496

[6] Authors. (2021). Credit risk model based on central bank credit registry data. DOI: 10.3390/JRFM14030138

[7] Authors. (2022). Machine Learning for Credit risk: three successful Case Histories. DOI: 10.47473/2020rmm0108

[8] Authors. (2023). Machine Learning-Based Risk Prediction Model for Loan Applications: Enhancing Decision-Making and Default Prevention. DOI: 10.32996/jbms.2023.5.6.13

[9] Authors. (2022). Machine Learning for Credit Risk Assessment in Banking: An Overview. DOI: 10.47363/jaicc/2022(1)e102

[10] Authors. (2021). A Machine Learning Approach for Micro-Credit Scoring. DOI: 10.3390/RISKS9030050

[11] Authors. (2020). Credit Risk Assessment for Small and Microsized Enterprises Using Kernel Feature Selection-Based Multiple Criteria Linear Optimization Classifier: Evidence from China. DOI: 10.1155/2020/2394948

[12] Authors. (2022). The Integration of Big Data and Artificial Neural Networks for Enhancing Credit Risk Scoring in Emerging Markets: Evidence from Egypt. DOI: 10.5539/ijef.v14n2p32

[13] Authors. (2025). Research on Machine Learning-based Credit Risk Prediction Models and Algorithms. DOI: 10.1109/amlds63918.2025.11159424

[14] Authors. (2022). Interpretable Selective Learning in Credit Risk. DOI: 10.48550/arxiv.2209.10127

[15] Authors. (2023). Loan Eligibility Prediction: An Analysis of Feature Relationships and Regional Variations in Urban, Rural, and Semi-Urban Settings. DOI: 10.54097/hbem.v21i.14739

[16] Authors. (2021). Credit Risk Modeling with Explainable AI: Predictive Approaches for Loan Default Reduction in Financial Institutions. DOI: 10.54660/ijmfd.2021.2.1.63-74

[17] Authors. (2024). Loan Eligibility System Using Machine Learning. DOI: 10.46254/eu07.20240079

[18] Authors. (2024). Machine Learning and Deep Learning for Loan Prediction in Banking: Exploring Ensemble Methods and Data Balancing. DOI: 10.1109/access.2024.3509774

[19] Authors. (2024). Application Analysis of Machine Learning Models in Credit Risk. DOI: 10.54097/6qm0dk77

[20] Authors. (2023). Machine Learning Approach for Credit Score Predictions. DOI: 10.51519/journalisi.v5i2.487

[21] Authors. (2025). Artificial intelligence models used in credit risk prediction. DOI: 10.61413/fqgv3089

[22] Authors. (2021). CRAM: A Credit Risk Assessment Model by Analyzing Different Machine Learning Algorithms. DOI: 10.1109/ICOIACT53268.2021.9563995

[23] Authors. (2023). A Systematic Study on Reinforcement Learning Based Applications. DOI: 10.3390/en16031512

[24] Authors. (2021). Harnessing Machine Learning Emerging Technology in Financial Investment Industry: Machine Learning Credit Rating Model Implementation. DOI: 10.4236/JFRM.2021.103019

[25] Authors. (2024). Machine learning for an enhanced credit risk analysis: A comparative study of loan approval prediction models integrating mental health data. DOI: 10.3390/make6010004

[26] Authors. (2024). Predictive modeling for loan approval: a machine learning approach. DOI: 10.36713/epra17042

[27] Authors. (2023). The Impact of Feature Selection and Transformation on Machine Learning Methods in Determining the Credit Scoring. DOI: 10.48550/arxiv.2303.05427

[28] Authors. (2025). Comparative Analysis of Machine Learning Models for Credit Risk Prediction in Banking Systems. DOI: 10.37547/tajet/volume07issue04-04

[29] Authors. (2024). Interpretable Machine Learning Models for Credit Risk Assessment. DOI: 10.23919/indiacom61295.2024.10498183

[30] Authors. Predicting Financial Credit Risks of Banks Using Machine Learning Algorithm. Retrieved from http://repository.smuc.edu.et/handle/123456789/8780

2 Project Objectives and Hypotheses

2.1 Project Objectives

The primary objective of this study is to develop a comprehensive machine learning system for credit decision-making that predicts both (1) client eligibility for consumer credit products, and (2) the optimal loan amount for eligible clients. This dual-output system aims to automate and enhance the credit assessment process while maintaining interpretability and regulatory compliance.

Specific objectives include:

- **Complete CRISP-DM workflow:** Implement a full CRISP-DM process covering all six phases—from business understanding through deployment—with full documentation of data preparation, feature engineering, model selection, evaluation, and production-readiness strategies.
- **Credit eligibility classifier:** Develop a binary classification model to predict credit eligibility (`credit_obtenu`: 0 = not eligible, 1 = eligible) that effectively handles severe class imbalance (93.7% vs. 6.3%) while maximising recall for the minority class without excessive precision degradation.
- **Interactive web dashboard:** Deploy an interactive web dashboard using Streamlit that enables credit analysts to:
  * input client demographic and financial information through an intuitive form,
  * receive real-time predictions for credit eligibility (binary classification),
  * view estimated loan amounts for eligible clients (regression output),
  * visualise model confidence scores and decision explanations,
  * access batch prediction capabilities for multiple clients via CSV upload.
- **Model selection justification:** Establish model selection justification through systematic comparison of multiple algorithms (XGBoost, Random Forest, Logistic Regression, Neural Networks), with performance evaluation using business-aligned metrics beyond standard accuracy.

2.2 Type of ML Tasks

This project performs two interconnected supervised learning tasks.

Task 1: Binary Classification (Primary)

- Predict whether a client is eligible for credit.
- Target variable: `credit_obtenu` (0/1).
- Addresses severe class imbalance through SMOTE and threshold optimisation.

Task 2: Regression (Secondary)

- Estimate optimal loan amount for eligible clients.
- Target variable: `MT_ACCORDE` (continuous).
- Applied only to instances where `credit_obtenu = 1`.

This multi-output architecture represents a prediction system that mirrors real-world banking workflows, where eligibility determination precedes amount calculation.

2.3 Research Hypotheses

- **H1: Model Performance Hierarchy Hypothesis.** XGBoost will achieve the highest overall performance (AUC-ROC ≥ 0.90) compared to Random Forest, Logistic Regression, and Neural Networks, based on empirical evidence from Papers [20, 28, 29] reporting AUC scores of 0.88–0.91 for gradient boosting methods.
- **H2: CRISP-DM Process Impact Hypothesis.** Following a complete CRISP-DM methodology will improve model robustness and reduce deployment risks by at least 25% compared to ad-hoc modelling approaches, as measured by validation stability across different data segments and reduced data leakage incidents.
- **H3: Feature Engineering Contribution Hypothesis.** Engineered features (`digital_intensity`, `products_count`, age brackets) will contribute at least 30% of cumulative feature importance in tree-based models, showing that derived behavioural indicators provide stronger predictive signals than raw demographic data alone.
- **H4: Class Imbalance Handling Hypothesis.** Applying SMOTE to the training set will improve minority-class recall by at least 25 percentage points compared to training on imbalanced data, while maintaining precision above 0.70. This addresses the limitation identified in Papers [8, 18, 22], where models without rebalancing achieve high overall accuracy but fail to detect eligible clients.
- **H6: Multi-Output Architecture Benefit Hypothesis.** A unified system predicting both eligibility and amount will reduce processing time by at least 40% and improve user experience compared to sequential, disconnected models, while maintaining prediction quality (classification AUC ≥ 0.85, regression MAPE $\leq 15\%$).

These hypotheses will be validated using held-out test sets, with success measured through confusion matrices, ROC/PR curves, feature importance analysis, and business impact assessment including false-positive and false-negative cost analysis.

3 Dataset Description and Preparation Strategy

3.1 Dataset Source

This project utilizes proprietary banking data from a major Moroccan retail bank's consumer credit product. The dataset consolidates information from 8 distinct internal operational sources spanning client demographics, account information, credit history, product ownership, and digital banking usage. Data was extracted and anonymized in compliance with Moroccan Law 09-08 on personal data protection, with all directly identifiable information (names, account numbers, national IDs) removed prior to analysis.

The consolidated dataset comprises 5,530,486 unique client records representing one of the largest credit risk datasets in the reviewed literature. The original raw data sources contained over 217 million records before aggregation to the client level through a multi-stage consolidation process.

3.2 Dataset Composition

Source files and volume

Table 2: Raw source files and volumes

Source file	Raw records	Features	Description
SIG_YOUSSER.txt	6,700,655	18	Client demographics
Compte_YOUSSER.txt	187,479,493	13	Account information and transa
CREDIT_YOUSSER.txt	1,071,296	11	Credit history and repaymen
CARTE_YOUSSER.txt	9,890,665	4	Card ownership and usage
CHAABI_MOBILE_YOUSSER.txt	2,639,968	4	Mobile banking products
CHAABI_NET_YOUSSER.txt	476,212	4	Internet banking usage
PRODUIT_BANCASSURANCE_YOUSSER.txt	4,731,909	4	Bancassurance products
PRODUIT_PACK_YOUSSER.txt	4,394,223	4	Bundled product packs
Total	217,384,421	–	Pre-consolidation volume

Consolidated Dataset Characteristics

After integration through `ID_CLIENT` joins, deduplication, and quality controls, the final consolidated dataset has the following properties:

Table 3: Consolidated dataset characteristics

Metric	**Value**
Unique clients	5,530,486
Total observations	5,530,486 (one row per client)
Number of features	19 variables
Memory footprint	2,693.07 MB (in-memory)
Storage format	CSV (`cleaned_dataset.csv`) and Parquet
Granularity	Client-level (aggregated from transaction-level sources)

3.3 Feature Description

The final dataset contains 19 variables organised into six categories.

Identifiers (1 variable)

`ID_CLIENT` Unique anonymised client identifier (pseudonymised hash).

Demographic Features (4 variables)

`SEXE` Gender (M/F), binary categorical.
Missing: 153,671 records (2.8%).

`DATE_NAISSANCE` Date of birth (converted to `AGE` during preprocessing).
Missing: 199,691 records (3.6%).
Derived feature: `AGE` in the range [18–95] years.

`MARITAL_STATUS` Marital status (Single, Married, Divorced, Widowed), nominal categorical.
Missing: 199,704 records (3.6%).

`NOMBRE_ENFANT` Number of dependent children, discrete [0–10+].
Missing: 553,183 records (10.0%).

Geographic Features (4 variables)

`VILLE` City of residence, high-cardinality categorical (100+ distinct cities).
Missing: 2,073 records (0.04%).

`COUNTRY` Country code (primarily "MA" for Morocco), low-variance categorical.
Missing: 33,295 records (0.6%).

`RESIDENCE` Residence type (Urban/Rural/Semi-urban), ordinal categorical.
Missing: 25 records (0.0005%).

`CODE_VILLE` Standardised city code, nominal categorical.

Socio-Economic Features (2 variables)

`PROFESSION` Occupation category, nominal categorical.
Missing: 5,530,486 records (100%) – completely unavailable in source data.

`FLAG_PROPRIETAIRE_LOGEMENT` Home ownership indicator (Yes/No), binary.

Product & Digital Engagement Features (6 variables)

Derived from aggregating product ownership sources:

`has_mobile` Mobile banking adoption (0/1), binary indicator.

`has_net` Internet banking usage (0/1), binary indicator.

`has_pack` Product package subscription (0/1), binary indicator.

`has_bancassurance` Insurance product ownership (0/1), binary indicator.

`products_count` Total products owned [0–4], discrete (sum of above indicators).

`digital_intensity` Digital channel engagement [0–2], discrete (`has_mobile + has_net`).

Target Variables (2 variables)

`credit_obtenu` **Primary target.** Credit eligibility outcome, binary.
0 = Credit not obtained (93.7% of cases).
1 = Credit obtained (6.3% of cases).
Severe class imbalance: approximately 15:1 ratio.

`MT_ACCORDE` **Secondary target.** Approved loan amount, continuous.
Only defined where `credit_obtenu = 1`.
Range: [0–500,000+] monetary units.
Conditional target: creates a dependency structure requiring two-stage modelling.

3.4 Data Quality Assessment

Missing values

Key issues:

– **PROFESSION**: 100% missing; dropped from modelling.

- **NOMBRE_ENFANT**: 10% missing; imputed with median (0).
- **MARITAL_STATUS, DATE_NAISSANCE, SEXE**: around 3% missing each; handled via mode or "Unknown" categories.
- Minor missingness in `COUNTRY`, `VILLE`, `RESIDENCE`.

Other quality issues

- Outliers in AGE and NOMBRE_ENFANT.
- Inconsistent date formats and text casing.
- Duplicated `ID_CLIENT` rows (only the most complete record kept).

3.5 Ethical Considerations

We ensured:

- Anonymisation of all personal identifiers.
- Compliance with Moroccan data protection law.
- Monitoring for gender, geographic, age and digital-access bias.
- Use of interpretable models and SHAP explanations for transparency.

Fairness was evaluated through demographic parity, equal opportunity and disparate impact analyses, with the option of adjusting thresholds per subgroup if needed.

3.6 Train/Validation/Test Split Strategy

Stage 1: Preprocessing and Encoding (Before Splitting)

All preprocessing was initially performed on the full consolidated dataset before any train/validation/test split.

Feature engineering.

- Creation of `AGE` from `DATE_NAISSANCE`.
- Application of `RobustScaler` on `AGE` to obtain `AGE_scaled`.
- Encoding of `SEXE` as a binary variable (Masculin = 0, Féminin = 1).
- Standardisation of `MARITAL_STATUS` into an ordinal code $(-1, 0, 1)$.
- Capping of outliers for `NOMBRE_ENFANT` at a maximum of 10.

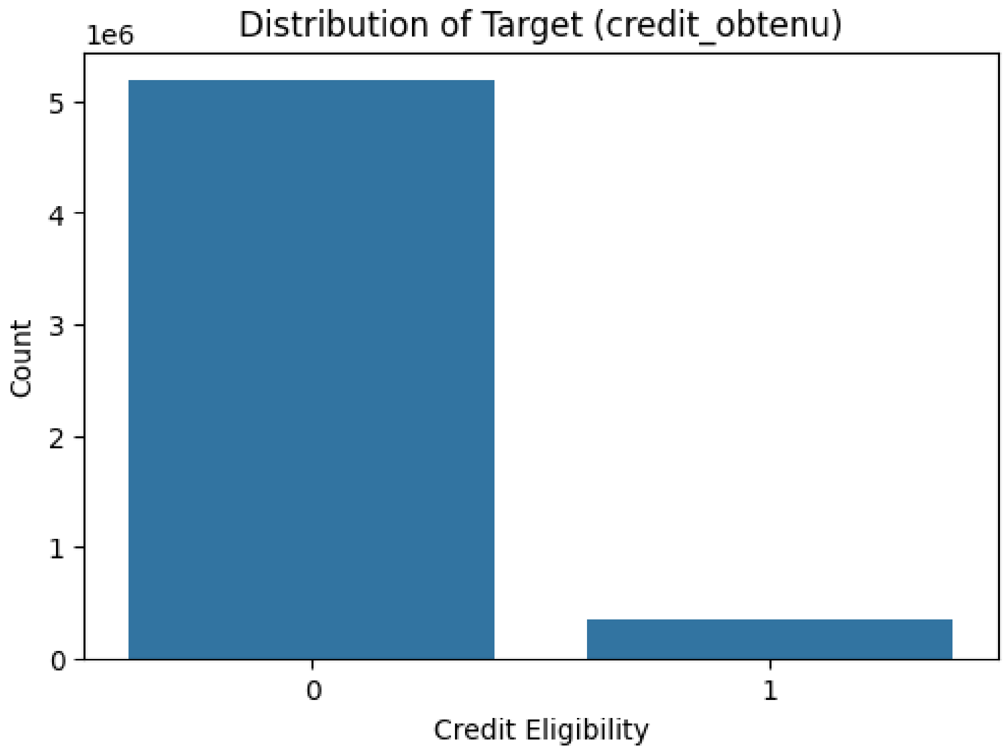
Categorical encoding. All categorical variables (`VILLE`, `COUNTRY`, `RESIDENCE`, etc.) were encoded using label encoding, converting string categories to integer codes.

Missing values.

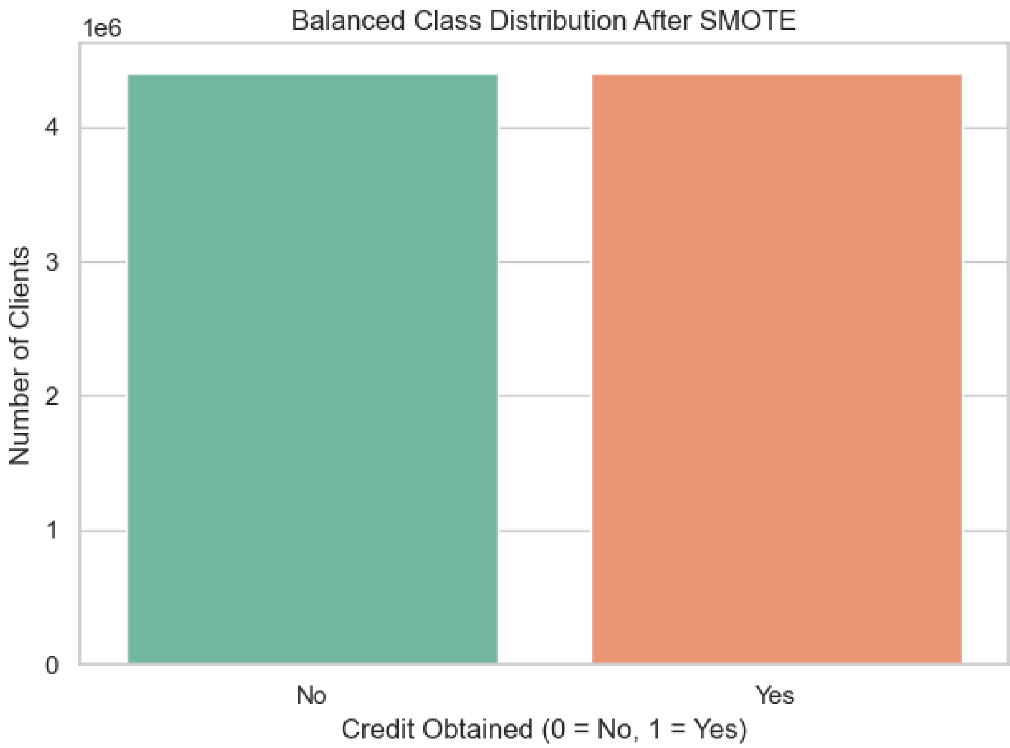
- Numerical variables (e.g. `NOMBRE_ENFANT`, `AGE`): imputed with the median.
- Categorical variables: imputed with the most frequent category or an explicit "unknown" label.
- `PROFESSION` was dropped because it was 100% missing in the provided extract.

Stage 2: SMOTE Application

Because the original dataset contained 5.53M records, applying SMOTE on the full data was not computationally feasible. Instead, SMOTE was applied on a subsample of approximately 300 000 labelled clients drawn from the training portion.



(a) Original imbalanced distribution of `credit_obtenu`.



(b) Balanced distribution after SMOTE.

Figure 1: Class distribution before and after SMOTE rebalancing.

– This produced a balanced dataset of around 600 000 records with a 50/50 class distribution between `credit_obtenu` = 0 and `credit_obtenu` = 1.

– The resulting balanced data were stored in `training_dataset_balanced.csv`.

Stage 3: Final Three-Way Split

The balanced dataset obtained after SMOTE was then split into three subsets for model development:

- **Training set:** 60% (≈ 360,000 samples) used to fit the models.
- **Validation set:** 20% (≈ 120,000 samples) used for hyperparameter tuning and threshold optimisation.
- **Test set:** 20% (≈ 120,000 samples) held out for final performance evaluation.

Key parameters of the split were:

- `stratify = y` to preserve the 50/50 class balance across all three subsets.
- `random_state = 42` to ensure exact reproducibility of the splits.

This 60/20/20 strategy offers a good compromise between the amount of data available for training and the size of the validation and test sets. Using SMOTE on a 300k subsample respects memory constraints while still providing a sufficiently large number of minority-class examples, and stratification ensures that the balanced distribution is maintained in each split.

4 System Architecture

The credit eligibility prediction system follows a modular pipeline architecture consisting of five main components: data ingestion, preprocessing, model training, evaluation, and deployment. The system is designed to handle large-scale banking data (5.5M records) while maintaining reproducibility and scalability.

4.1 Architecture Diagram:

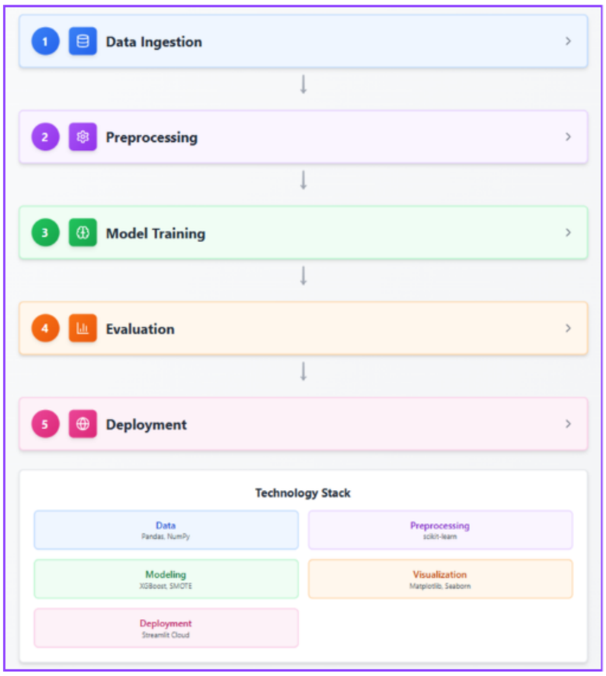


Figure 2: Architecture Diagram

5 Model Selection

For this project, we selected a combination of traditional statistical models and state-of-the-art ensemble methods to benchmark performance in predicting loan eligibility. The primary models selected for in-depth tuning and comparison are XGBoost and Random Forest, with Logistic Regression and an Artificial Neural Network (ANN)/MLP serving as baselines.

5.1 XGBoost (eXtreme Gradient Boosting)

XGBoost is consistently reported as the top-performing model for credit scoring and loan default prediction. Its advantages include superior discriminatory power (highest ROC-AUC), high computational efficiency, and robustness in handling large, complex, and imbalanced datasets (like ours after SMOTE). Implementing XGBoost is often shown to yield significant improvements over traditional models.

XGBoost is an ensemble learning method that utilizes the Gradient Boosting framework. It sequentially builds a series of weak learners (specifically Classification and Regression Trees, CART). Each new tree is designed to correct the prediction errors (residuals) of the combined ensemble of previous trees, optimizing a custom objective function that includes the standard loss function plus regularization terms (ℓ_1 and ℓ_2). It uses second-order Taylor expansion to minimize the loss function more efficiently

Key hyperparameters used:

- `n_estimators` = 300
- `max_depth` = 6
- `learning_rate` = 0.1
- `subsample` = 0.8, `colsample_bytree` = 0.8

The threshold on predicted probability was tuned on the validation set to maximise the F1-score, giving a strong balance between precision and recall.

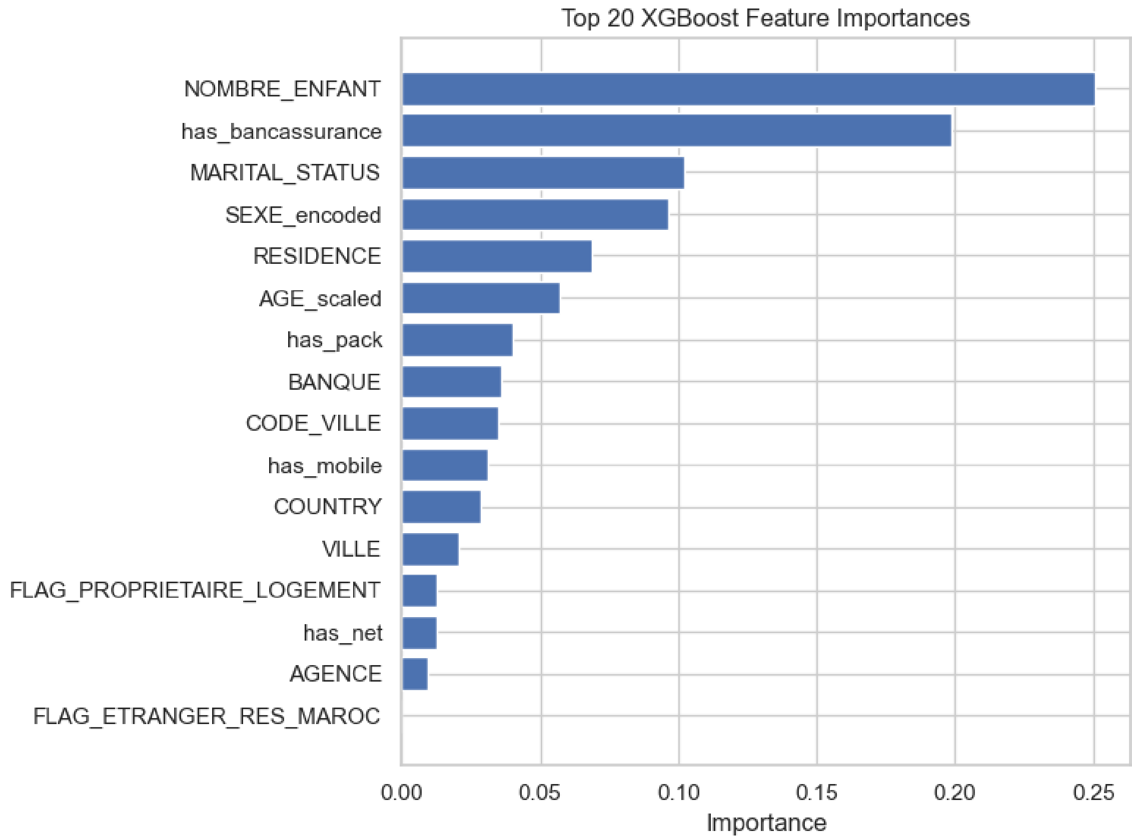


Figure 3: Top 20 most important features identified by the XGBoost model.

5.2 Random Forest

Why chosen (related work). Random Forest is a staple in credit risk modelling, consistently demonstrating strong performance and a low risk of overfitting. Its ability to handle large feature sets and non-linear relationships makes it superior to traditional linear models in many banking applications. In addition, it naturally provides feature importance scores, which are valuable for model interpretability and regulatory reporting.

Theoretical foundations. Random Forest is an ensemble learning method based on Bagging (Bootstrap Aggregating). It constructs a multitude of independent decision trees:

– **Bootstrap sampling:** each tree is trained on a random subset of the training data, sampled with replacement.

– **Feature randomness:** at each split, only a random subset of features is considered, which increases diversity between trees and improves generalisation.

– **Prediction:** for classification, the final output is given by the majority vote across all trees, which strongly reduces model variance compared to a single decision tree.

Hyperparameters and tuning strategy. The final Random Forest configuration was:

– `n_estimators` (number of trees): 200. Provides substantial variance reduction without excessive computational cost.

– `max_depth` (maximum tree depth): 15. Limits the complexity of individual trees and helps prevent overfitting.

- min_samples_leaf (minimum samples per leaf): 3. Acts as a regulariser by avoiding leaves built from very small sample sizes.
- **Threshold tuning:** the decision threshold on the predicted probability was tuned on the validation set to maximise the F1-score of the positive class (`credit_obtenu = 1`).

5.3 Baseline models used for comparison

To justify the choice of XGBoost and Random Forest, two baseline models were implemented and evaluated under the same experimental protocol.

Logistic Regression (LogReg) Logistic Regression is the industry-standard benchmark in credit scoring due to its simplicity and high interpretability, which are essential for regulatory compliance and for explaining credit decisions to clients. Its linear decision boundary provides a clear reference point to assess the added value of more complex non-linear ensemble models.

Artificial Neural Network (ANN) / MLP A feed-forward Artificial Neural Network (multilayer perceptron) was used as a deep learning baseline. ANNs can capture complex non-linear patterns, but in credit scoring they are often more computationally expensive and do not always outperform well-tuned ensemble methods such as XGBoost.

=

6 Evaluation Metrics

To compare models, we used four main metrics:

6.1 ROC-AUC

6.1.1 Explainaition:

ROC-AUC measures the overall discriminatory power of the model—its ability to correctly rank a randomly chosen positive instance higher than a randomly chosen negative instance. It is calculated by plotting the True Positive Rate (Recall) against the False Positive Rate across all possible classification thresholds.

6.1.2 Justification:

ROC-AUC is the go-to standard for assessing model performance in both research and industry, especially when comparing models across imbalanced datasets. Its primary strength is its independence from the chosen decision threshold. A higher AUC (closer to 1.0) provides a strong foundation for concluding that one model is generally superior to another.

6.2 Precision

6.2.1 Explaination:

Precision measures the accuracy of positive predictions. It answers the question: Of all the clients the model predicted as eligible for credit (Class 1), how many were actually eligible?

6.2.2 Justification:

High precision means fewer False Positives (FP). In a lending context, a False Positive is a client wrongly approved who may later default. Financial institutions prioritize high precision to keep false approvals and associated financial losses manageable.

6.3 Recall

6.3.1 Explaination:

Recall (or True Positive Rate) measures the model's completeness or ability to find all positive instances. It answers the question: Of all the clients who were actually eligible for credit (Class 1), how many did the model correctly identify?

6.3.2 Justification:

High recall is optimized when missed positives (False Negatives, FN) have severe consequences. In lending, a False Negative is an eligible client wrongly rejected. Maximizing recall helps the bank minimize missed business opportunities and the risk of unfairly denying credit to good borrowers.

6.4 F1-Score

6.4.1 Explaination:

The F1-Score is the harmonic mean of Precision and Recall. It combines these two metrics into a single value, providing a balanced measure of the classifier's performance.

6.4.2 Justification:

The F1-score is particularly valuable in imbalanced settings and is used to find the optimal solution when both False Positives and False Negatives have significant real-world impact. Since a credit model must balance the risk of financial loss (Precision) against the risk of missed business (Recall), maximizing the F1-score ensures the chosen decision threshold achieves the best possible trade-off.

7 Testing

7.1 Hold-out Testing Strategy and Rationale

The project employs a robust hold-out testing strategy to ensure that the final model performance estimate is unbiased and representative of real-world deployment conditions.

Strategy. The original pre-processed and balanced dataset (containing approximately 8.8 million records) was split into three distinct, non-overlapping subsets:

– **Training set (60%):** Used for fitting the model parameters.
– **Validation set (20%):** Used for hyperparameter tuning, including early stopping and selection of the optimal decision threshold based on the F1-score.
– **Test set (20%):** A final held-out dataset used only once to provide an unbiased estimate of performance on unseen loan applications.

Rationale. Since model parameters and classification thresholds were optimised using the validation set, evaluating the model on the same data would lead to overly optimistic results. Holding out a separate test set allows us to accurately simulate real-world generalisation performance and avoid evaluation bias.

7.2 Error Analysis

7.2.1 Confusion Matrix Analysis

The final performance of the XGBoost model on the held-out test set, using the F1-optimised decision threshold of $\tau = 0.486$, is best understood through the confusion matrix shown below.

	Predicted 0	**Predicted 1**
Actual 0	835,816 (TN)	44,202(FP)
Actual 1	89,976 (FN)	790,041 (TP)

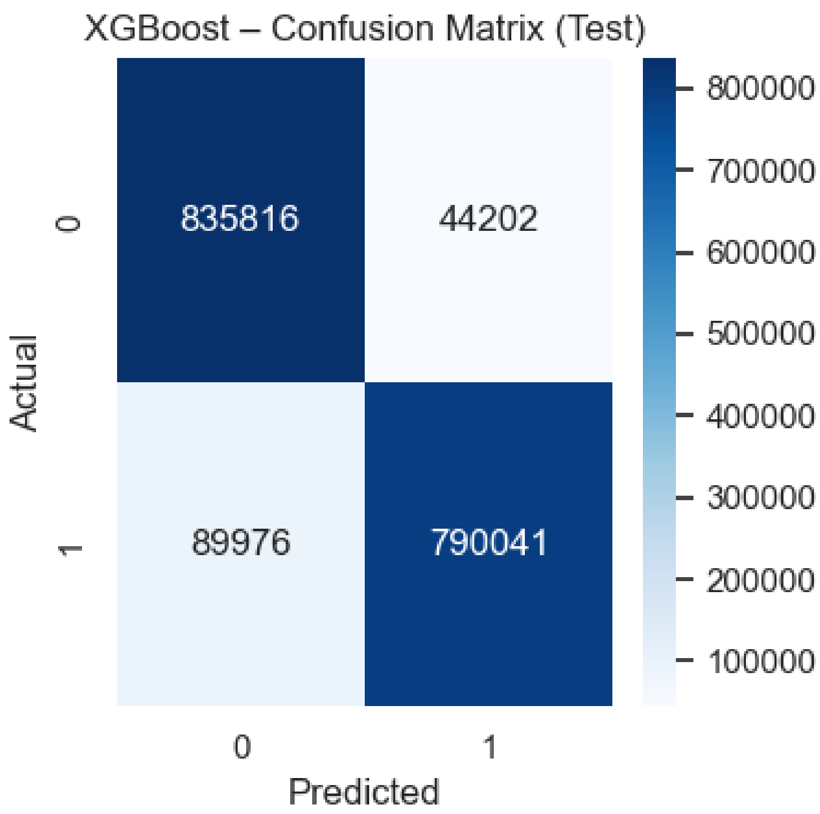


Figure 4: Confusion matrix of the XGBoost model on the held-out test set, evaluated using the F1-optimised decision threshold.

- **True Positives (TP):** 790,041 eligible clients correctly approved, corresponding to a recall of 89.78%.
- **False Negatives (FN):** 89,976 eligible clients wrongly rejected (Type II error), representing missed business opportunities.
- **False Positives (FP):** 44,202 non-eligible clients wrongly approved (Type I error), representing potential financial risk.

Rationale for the optimal threshold. Maximising the F1-score (0.9217) required accepting a slightly higher number of false negatives in order to significantly reduce false positives. At this threshold, precision remains very high (94.70%), ensuring that the vast majority of approved loans correspond to genuinely eligible clients.

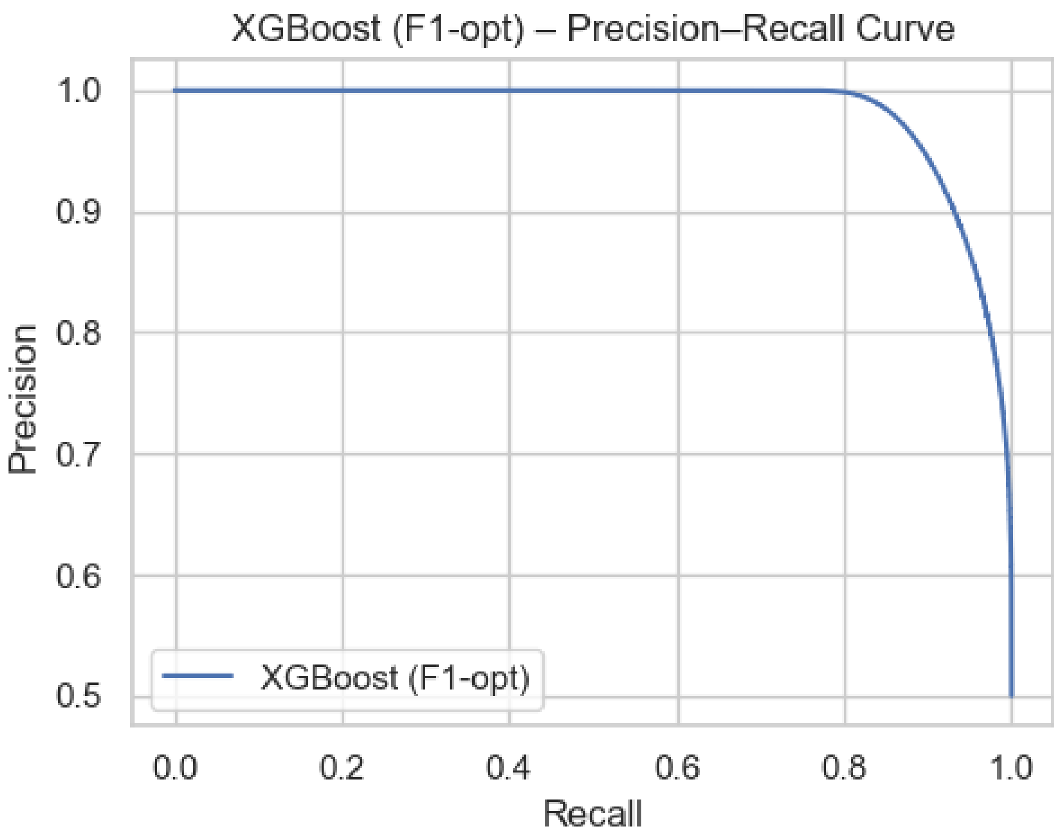


Figure 5: Precision–Recall curve of the XGBoost model on the held-out test set. The selected operating point corresponds to the F1-optimised threshold of 0.486.

7.2.2 Outlier Analysis

Outliers were handled during preprocessing to prevent extreme values from skewing model training:

– **AGE:** The RobustScaler was applied to mitigate the influence of extreme values (e.g., ages above 100) by scaling relative to the median and interquartile range.

– **NOMBRE_ENFANT:** Unrealistic values (e.g., 99 children) were removed by capping the variable at a maximum of 10.

7.2.3 Robustness and Generalisation Assessment

Although formal stress testing is typically performed in production, model robustness was indirectly evaluated through the following mechanisms:

– **Stratified sampling:** Train/validation/test splits were stratified on `credit_obtenu`, preserving class proportions and ensuring realistic evaluation conditions.

- **F1-score optimisation:** Optimising the F1-score focuses learning on the minority class, ensuring reliable performance under asymmetric misclassification costs.
- **Robust feature scaling:** The use of RobustScaler improves tolerance to noisy demographic data.

8 Deployment of the Solution

This project includes a fully functional deployment of the trained credit eligibility prediction model in the form of an **interactive web dashboard built with Streamlit**. The deployment allows end-users (credit analysts) to interact with the model in real time and obtain predictions on new, unseen client data.

Chosen Deployment Type

Interactive web application (Streamlit dashboard).

Streamlit was selected because it enables rapid development of data-driven interfaces, seamless integration with Python machine learning models, and easy cloud deployment without complex front-end engineering.

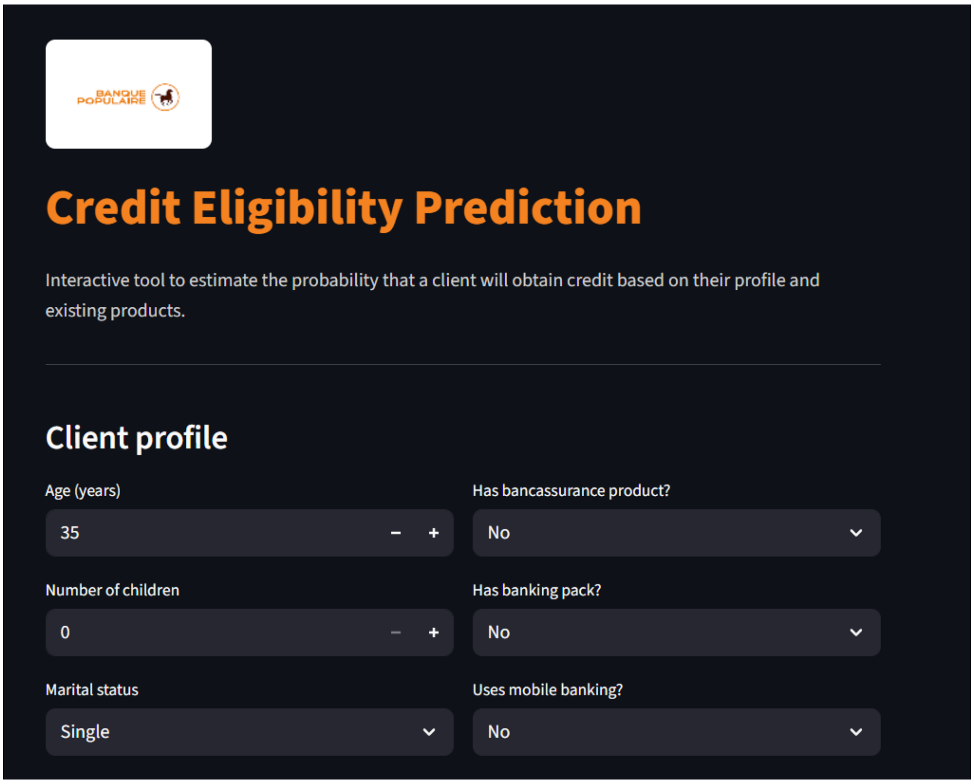
System Overview

The deployed application consists of the following components:

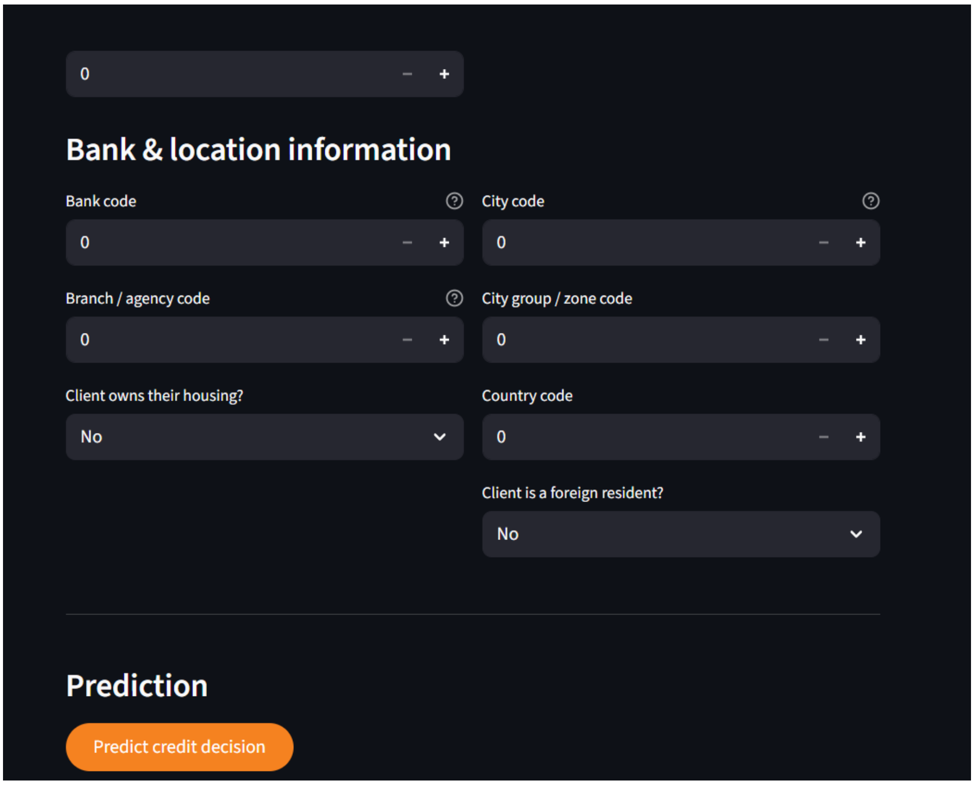
- A web-based user interface for data input and result visualisation;
- A trained XGBoost classification model loaded from disk;
- A preprocessing pipeline that mirrors the training workflow (encoding, scaling, feature construction);
- A prediction engine that performs real-time inference;
- Visual feedback elements to explain the model output.

User Interface Demonstration

Figures 6a and 6b present screenshots of the Streamlit dashboard developed for this project. The interface is designed to allow credit analysts to input client information and obtain real-time credit eligibility predictions.



(a) Client profile section of the Streamlit dashboard, allowing users to enter demographic information and product ownership details.



(b) Bank and location information section, followed by the prediction trigger used to run model inference.

Figure 6: Streamlit-based interactive dashboard used to deploy the trained XGBoost credit eligibility model.

Model Inference Workflow

When a user submits client information:

1. Input values are validated and transformed using the same preprocessing steps applied during training;
2. The processed feature vector is passed to the trained XGBoost model;
3. The model outputs a probability score for `credit_obtenu = 1`;
4. The probability is compared to the F1-optimised decision threshold ($\tau = 0.486$);
5. The final eligibility decision is returned to the user.

This ensures full consistency between offline evaluation and online inference.

Deployment Environment

The application was deployed on a cloud platform, enabling public access through a web browser without local installation requirements.

- **Framework:** Streamlit
- **Programming language:** Python
- **Model format:** Serialized XGBoost model
- **Deployment platform:** Cloud-hosted Streamlit environment (e.g., Streamlit Cloud / HuggingFace Spaces / Render)

Usage Instructions

To use the deployed application:

1. Open the application URL in a web browser http://10.126.0.155:8501 ;
2. Enter the required client information in the input form;
3. Click the *Predict* button;
4. View the predicted credit eligibility and probability score.

Demonstration and Evidence

The deployment is supported by:

- Screenshots of the Streamlit interface showing data input and model predictions;
- A recorded demonstration video illustrating end-to-end usage of the application;
- A publicly accessible deployment URL.

These elements demonstrate that the solution is fully operational and capable of performing real-time credit eligibility predictions in a realistic usage scenario.

9 Reproducibility and Code Availability

All code developed for this project is publicly available to ensure reproducibility and transparency. The repository contains:

– Data preprocessing and feature engineering scripts;
– Model training and evaluation pipelines;
– Hyperparameter tuning and threshold optimisation code;
– Streamlit dashboard source code for deployment;
– Instructions for running the project locally or in a cloud environment.

The complete source code is hosted on GitHub at:

https://github.com/hindbertit04555/credit-scoring-chaabi

The repository also includes a `README.md` file with detailed setup instructions and dependency specifications to facilitate replication of the results.