Technical Report

Technical Report: Credit Risk Classification System ## Executive Summary FinTech Solutions Inc. requires an automated credit risk model to replace manual review workflows and improve consistency, speed, and loss control. This project built a complete machine learning pipeline to classify applicants into `good` (0) and `bad` (1) risk classes using the German Credit dataset. The pipeline includes EDA, preprocessing, feature engineering, baseline and advanced models, hyperparameter tuning, MLflow tracking, and business-oriented evaluation.   Primary business objective was to maximize default recall (catch as many likely defaulters as possible), with a target of at least 75%. Final evaluation shows: - `logistic_regression` achieved recall `0.797` - `xgboost` achieved highest AUC `0.947` and F1 `0.825` - threshold tuning selected logistic regression at threshold `0.5833`, with precision `0.7465` and recall `0.7681` Recommended deployment candidate: threshold-tuned Logistic Regression for recall alignment + interpretability. --- ## 1. Introduction ### 1.1 Business Context The lending process currently depends on manual underwriting decisions that are slower, less consistent, and difficult to scale with rising demand. ### 1.2 Problem Definition Build a production-ready classification system to predict default risk from applicant financial and demographic profile. ### 1.3 Objectives - Build reproducible ML pipeline - Compare four classification algorithms - Optimize model selection for business risk - Provide deployment-ready recommendations and monitoring plan ### 1.4 Success Criteria - Strong default detection recall (`>= 0.75` target) - model interpretability - practical inference performance - reproducible experiments and artifacts --- ## 2. Data Analysis ### 2.1 Dataset Description - Dataset: German Credit Risk - Approximate size: 1000 records - Target column: `Risk` (`good` / `bad`) - Missing values present in: - `Saving accounts`   - `Checking account` ### 2.2 Data Quality Assessment Performed: - shape/type checks - missing value analysis - duplicate checks - target distribution check ### 2.3 EDA Summary Required visualizations created: - target count plot - histograms (`Age`, `Credit amount`, `Duration`) - boxplots by target - numerical correlation heatmap - additional insights: - default rate by loan purpose   - default rate by age group ### 2.4 Feature Engineering Engineered features: - `Credit_to_Duration_Ratio` - `Age_Group` - `Account_Stability` - `High_Risk_Purpose` These features improve risk signal coverage and add business interpretability. --- ## 3. Methodology ### 3.1 Preprocessing Pipeline - Missing account categories imputed with `unknown` - Target encoded: `good=0`, `bad=1` - Categorical encoding: one-hot encoding - Numerical scaling: standard scaling - Split strategy: 80/20 stratified split, `random_state=42` ### 3.2 Models Developed - Logistic Regression - Decision Tree - Random Forest - XGBoost ### 3.3 Hyperparameter Tuning Tuned models: - Random Forest (GridSearchCV, 5-fold) - XGBoost (GridSearchCV, 5-fold) Optimization metric for tuning: recall (aligned to business objective). ### 3.4 Experiment Tracking MLflow used to track: - model parameters - evaluation metrics - confusion matrices as artifacts - model binaries --- ## 4. Results ### 4.1 Performance Comparison From `reports/model_evaluation_comparison.csv`: | Model | Accuracy | Precision | Recall | F1-Score | AUC-ROC | |---|---:|---:|---:|---:|---:| | logistic_regression | 0.815 | 0.705 | 0.797 | 0.748 | 0.904 | | xgboost | 0.890 | 0.912 | 0.754 | 0.825 | 0.947 | | random_forest | 0.840 | 0.894 | 0.609 | 0.724 | 0.911 | | decision_tree | 0.780 | 0.719 | 0.594 | 0.651 | 0.736 | ### 4.2 Threshold Tuning From `reports/production_threshold_recommendation.csv`: - selected model: `logistic_regression` - threshold: `0.5833` - precision: `0.7465` - recall: `0.7681` ### 4.3 Visualization Outputs Generated in `reports/figures/`: - confusion matrices - ROC curves - precision-recall curve - feature importance charts for tree-based models ### 4.4 Statistical Interpretation Observed tradeoff: - XGBoost maximizes discrimination (AUC/F1) - Logistic Regression offers strongest default recall while remaining interpretable --- ## 5. Business Recommendations ### 5.1 Metric Priority Most important metric: default recall.   Reason: false negatives (missed defaulters) carry direct credit-loss impact. ### 5.2 False Positive vs False Negative - False Negative (`bad` predicted as `good`): financial loss risk - False Positive (`good` predicted as `bad`): missed revenue/opportunity cost ### 5.3 Recommended Model Deploy threshold-tuned Logistic Regression as primary model in initial rollout: - meets recall-focused objective - easier regulatory explanation - low operational complexity Maintain XGBoost as challenger model for controlled A/B evaluation. --- ## 6. Deployment Considerations ### 6.1 Infrastructure - Serve model as