Model selection:

Although the data is unbalanced, we applied SMOTE to balance it before modeling. Thus, we can use accuracy and AUC to evaluate and compare models directly. Here are the results

Table 1. Hyperparameter optimization and modeling results using the original data

Algorithms	Hyper parameters	best estimate	accuracy	precision	recall	AUC	runtime	top 3 most important features
regularized logistic regression	C	0.01	0.739	0.747	0.725	0.808	7.26	PAY_1, MARRIAGE_2, SEX_2
decision tree	criterion, max_depth	gini, 10	0.757	0.787	0.707	0.808	26.07	PAY_1, PAY_2, MARRIAGE_2
random forest	n_estimators, max_depth	200, 9	0.782	0.814	0.732	0.858	182.25	PAY_1, PAY_2, SEX_2
gradient boosting	learning_rate, n_estimator, max_depth	250, 9, 0.25	0.844	0.863	0.819	0.917	1926.3	PAY_1, PAY_2, BILL_AMT1
Extreme gradient boosting	subsamples, n_estimators, max_depths, learning rate, gamma	0.8, 400, 10, 0.05, 0.3	0.846	0.867	0.817	0.923	418.62	PAY_2, PAY_1, EDUCATION_4
KNN	n_neighbors, p	50, 1	0.76	0.782	0.721	0.84	921.95	PAY_1, PAY_2, LIMIT_BAL

Table 2. Hyperparameter optimization and modelling results using the engineered features

Algorithms	Hyper parameters	best estimate	accuracy	precision	recall	AUC	top 3 most important features
regularized logistic regression	C	0.05	0.727	0.733	0.716	0.795	pay_sum, MARRIAGE_2, EDUCATION_3
decision tree	criterion, max_depth	gini, 10	0.756	0.77	0.73	0.814	pay_sum, payment_sum, MARRIAGE_2
random forest	n_estimators, max_depth	200, 9	0.773	0.784	0.754	0.851	pay_sum, payment_sum, MARRIAGE_2
gradient boosting	learning_rate, n_estimator, max_depth	250, 9, 0.25	0.827	0.829	0.825	0.901	pay_sum, payment_sum, pay_trend
Extreme gradient boosting	subsamples, n_estimators, max_depths, learning rate, gamma	0.8, 400, 10, 0.05, 0.3	0.826	0.829	0.82	0.901	pay_sum, MARRIAGE_2, EDUCATION_4
KNN	n_neighbors, p	1, 50	0.754	0.772	0.72	0.829	pay_sum, LIMIT_BAL

Based on AUC and accuracy value, we can see extreme gradient boosting is the best choice for us.