greatlearning
Power Ahead

Predicting Loan Default for DHDL

NBFC Loan Default Hackathon | Classical machine learning, v1, v2 and v3

Saurav Atri | 28 September 2026

Executive summary

v3 lifts cross-validated accuracy from 89.60% (v1) to 91.15% (v2) to 91.87%

91.87%	91.83%	91.15%	76.25%
v3 CV accuracy (upload)	v3 honest estimate	v2 CV accuracy (backup)	predict 'no default' for all

- All three versions beat the public leaderboard top of 88.28% in cross-validation; the hidden test score is only known after upload.
- Repayment behaviour (weeks paid, interest received vs expected, balance) predicts default far better than borrower profile.
- v3 rebuilds each loan's repayment schedule and stacks six models; it agrees with v2 on 96.89% of the 39,933 test loans.
- 100% accuracy is not achievable: some loans look identical in the data but end differently.

Business problem

Which active loans will default?

- DHDL is a non-banking finance company; every missed default is lost principal, every false alarm is a lost customer.
- Task: predict default (1) or not (0) for 39,933 test loans.
- Metric: accuracy = (TP + TN) / N, scored on a hidden leaderboard.
- Submission: CSV with exactly 39,933 rows plus header, columns ID, default.

Data overview

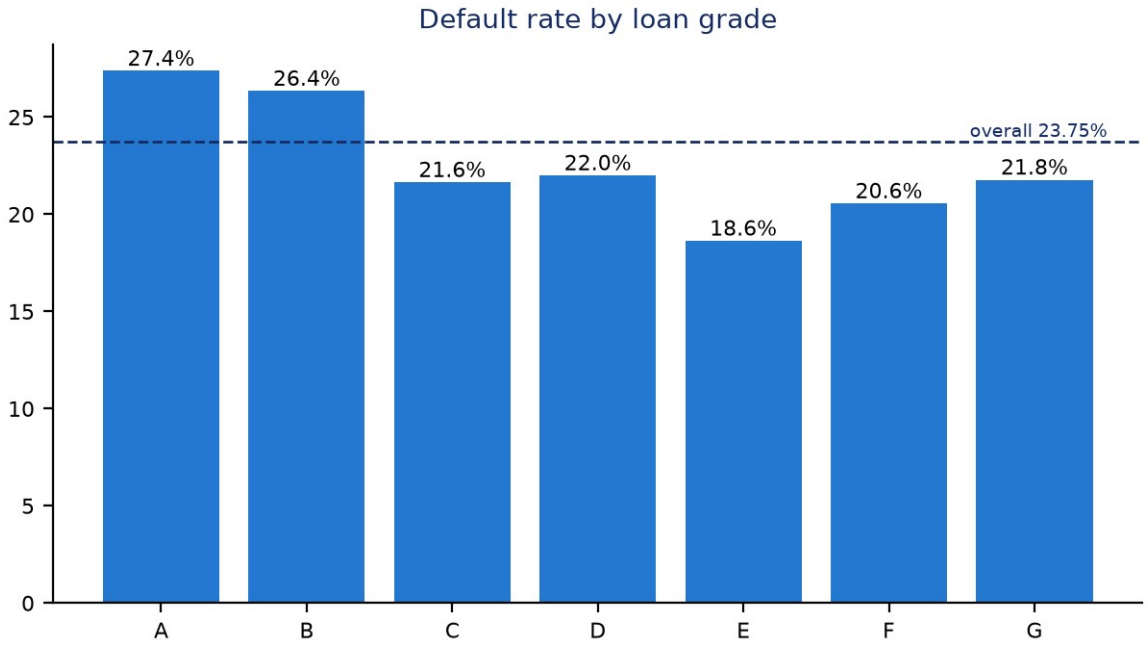
93,174	39,933	23.75%	0
training loans	test loans	default rate	duplicate rows

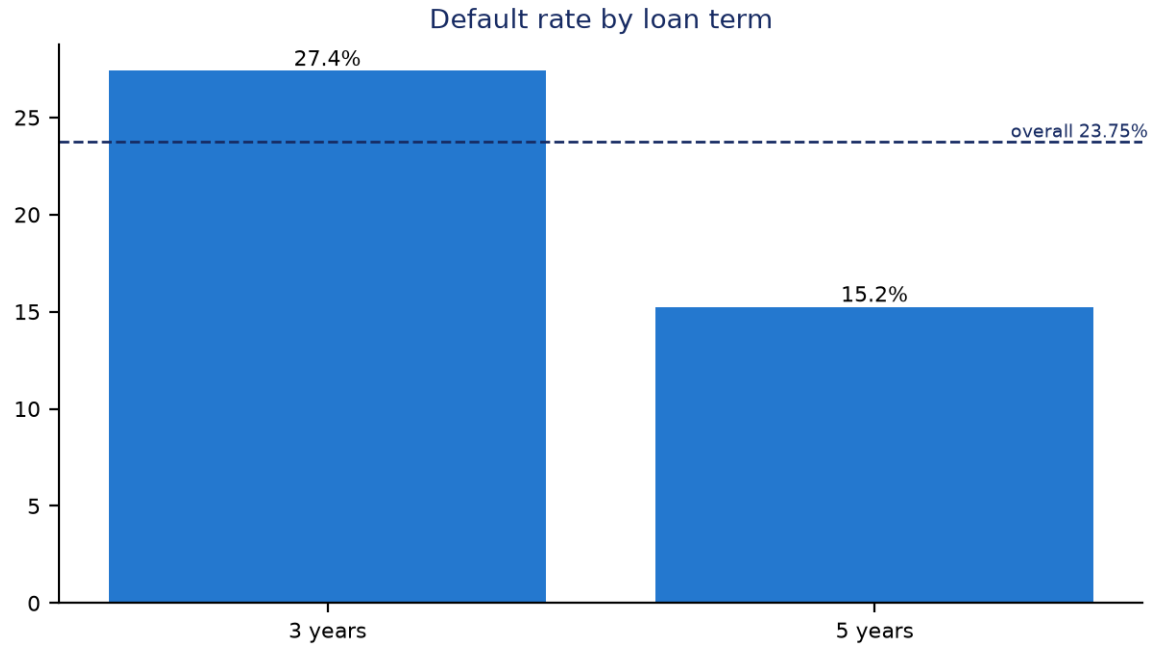
Column	Missing rows	Treatment
total_current_balance / total_revolving_limit	7,386	kept as missing + flag
job_experience	4,702	ordinal + missing flag
last_week_pay	1,924	kept as missing + flag

Tree models handle missing values natively, so no rows were dropped and no values were invented. ID was removed before training.

EDA: grade and term

The patterns run opposite to typical lending intuition





Grade A loans default most often (27.4%) and grade E least (18.6%) in this dataset.

3-year loans default at 27.4% versus 15.3% for 5-year loans. Profile alone is a weak guide; behaviour matters more.

Feature engineering (target-free)

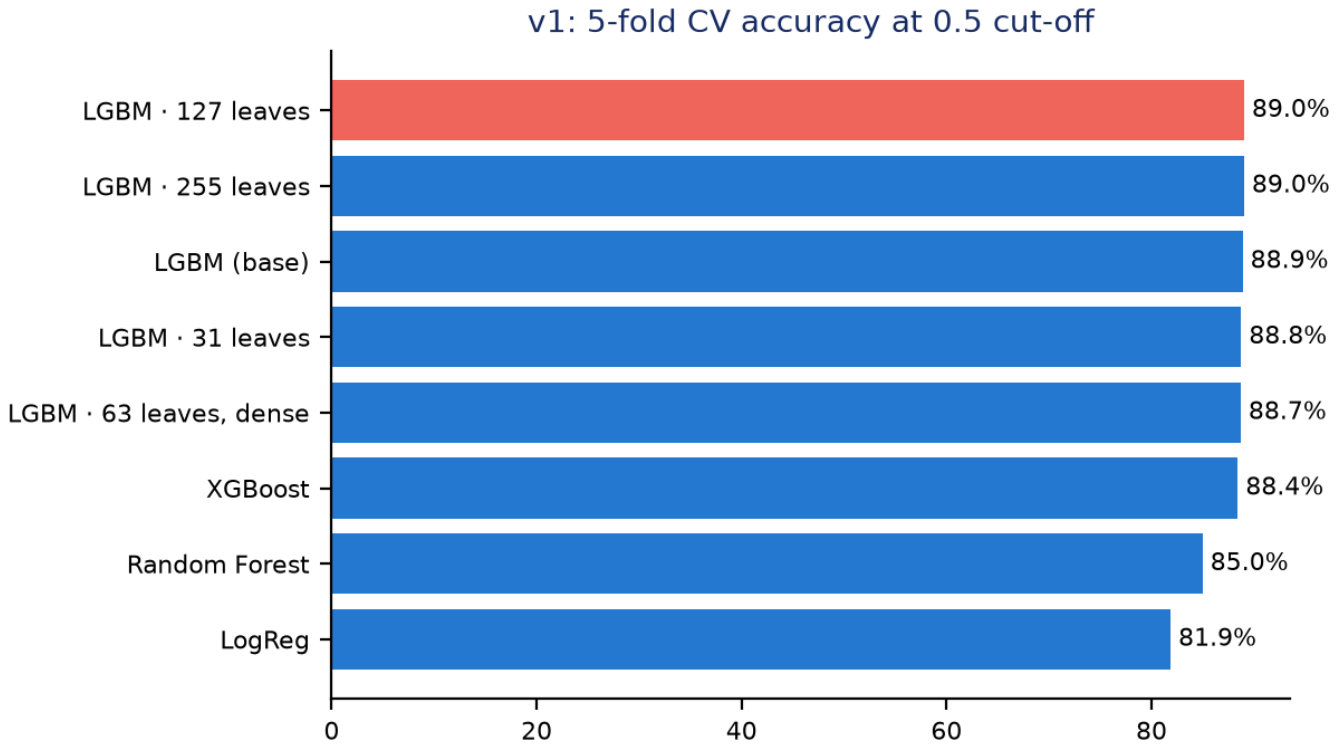
14 v1 features + 20 v2 features, built without the answer column

Group	Examples
Repayment progress	weeks paid ÷ term, weeks left on loan, rate × progress
Interest behaviour	interest received ÷ loan, received − expected, received ÷ expected to date
Affordability	loan ÷ income, instalment ÷ income, balance ÷ income
Credit usage	revolving utilisation, revolving balance ÷ loan
Peer comparison (v2)	gap vs average of same sub-grade, state, purpose (12)
Encoding	sub-grade as ordinal, job experience ordinal, missing flags, category frequencies

Leakage control: the default column is never used to build features; peer averages use input columns only.

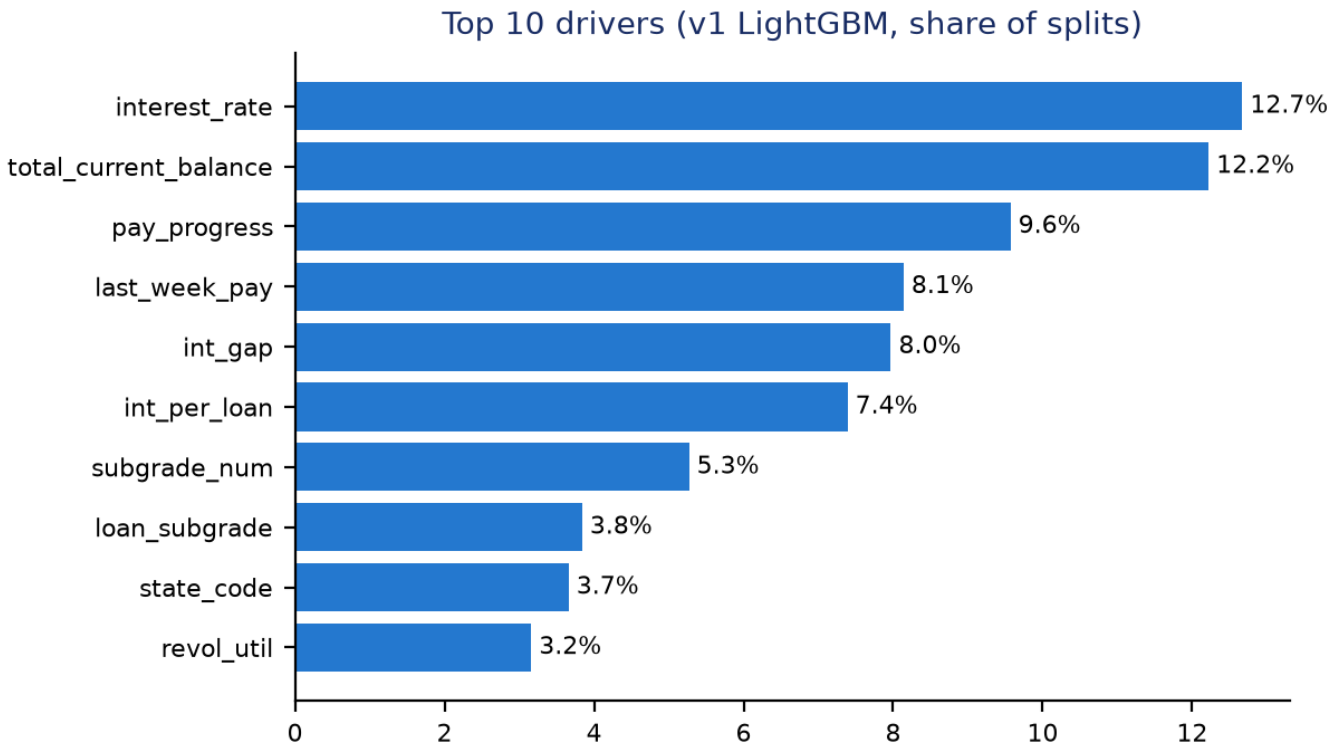
v1: model comparison

Seed 42, 5-fold stratified cross-validation



- LightGBM 127 and 255 leaves tie at 89.03% (0.5 cut-off); 255 leaves chosen on AUC.
- Tuned the cut-off on out-of-fold predictions: 0.38 added ~0.6 points → 89.60%.
- Blending with XGBoost gave no gain, so the single model was kept.
- CatBoost was too slow to finish in v1.

v1: what drives default



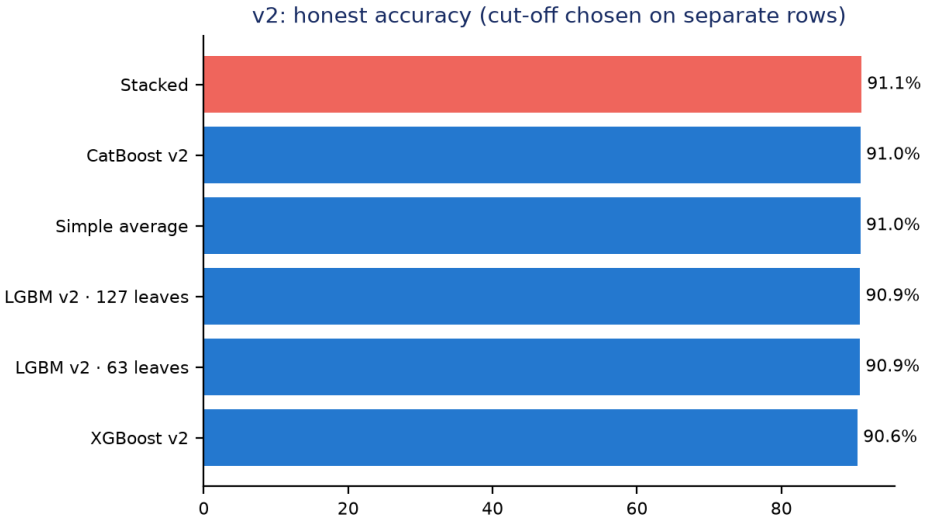
- Interest rate and current balance lead.
- Repayment progress and weeks paid follow closely.
- Interest received versus expected is a strong early-warning signal.

v2: what we changed

- 20 new target-free features: repayment ratios, peer comparisons, category frequencies.
- Added CatBoost (depth 6, early stopping) alongside two LightGBM variants and XGBoost.
- Stacking: a logistic meta-model learns how to combine the four models.
- Honest cut-off: chosen on 4/5 of rows, scored on the other 1/5 (nested).
- Stability: repeated with seeds 7, 42, 2026; every seed beat v1.
- Decision rule set in advance: promote v2 only if it beats v1 by at least 0.15 points.

v2: model comparison

Model	Acc @0.5	Tuned	Honest	AUC
Stacked (logistic meta-model)	91.07%	91.15%	91.11%	0.9545
CatBoost v2	90.86%	91.04%	90.98%	0.9524
Simple average	90.81%	91.01%	90.96%	0.9529
LightGBM v2 · 127 leaves	90.81%	90.95%	90.91%	0.9522
LightGBM v2 · 63 leaves	90.72%	90.93%	90.9%	0.9518
XGBoost v2	90.37%	90.59%	90.56%	0.9472



The stack wins on every measure; the honest estimate (91.11%) is within 0.04 points of the tuned score, so the cut-off is not flattering.

v2 model performance (backup)

	Predicted no default	Predicted default
Actual no default	TN 67,009	FP 4,036
Actual default	FN 4,207	TP 17,922

- Accuracy 91.15% = (17,922 + 67,009) / 93,174.
- Cut-off 0.445; AUC 0.9545.
- Predicted test default rate 24.03% vs 23.75% in training.

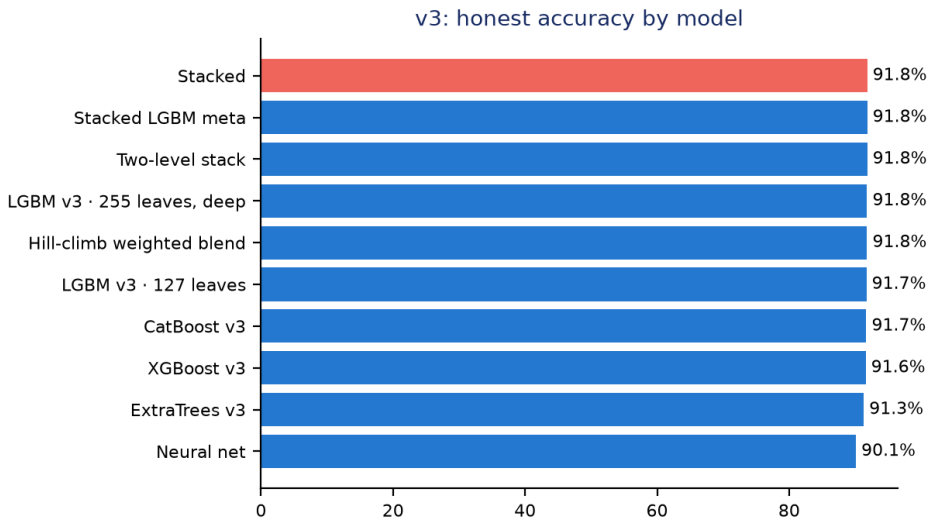
v3: what we changed

Aimed at the loans v2 got wrong

- Rebuilt each loan's repayment schedule: the balance and interest expected if every payment had been made.
- Compared it with the interest actually received, giving an estimate of months behind.
- Default rates by sub-grade, state and purpose, computed only inside each training split (no leakage).
- Six diverse models: two LightGBM, XGBoost, CatBoost, ExtraTrees and a neural network.
- Checked train vs test: a model could not tell them apart (score 0.4996; 0.5 = identical), so no re-weighting was needed.
- Combined with logistic, LightGBM and hill-climbing meta-models; cut-off chosen on separate rows.

v3: model comparison

Model	Acc @0.5	Tuned	Honest	AUC
Stacked (logistic meta-model)	91.85%	91.87%	91.83%	0.9645
Stacked (LightGBM meta-model)	91.84%	91.85%	91.83%	0.9646
Two-level stack (logistic + LightGBM meta)	91.86%	91.87%	91.83%	0.9648
LightGBM v3 · 255 leaves, deep	91.77%	91.83%	91.81%	0.9637
Hill-climb weighted blend	91.77%	91.83%	91.81%	0.9638
LightGBM v3 · 127 leaves	91.71%	91.75%	91.71%	0.9632
CatBoost v3	91.6%	91.71%	91.69%	0.9617
XGBoost v3	91.51%	91.6%	91.6%	0.9606
ExtraTrees v3	91.13%	91.33%	91.31%	0.9583
Neural network (tabular MLP)	90.16%	90.17%	90.12%	0.9397



Stability: LightGBM seeds 7/42/2026 scored 91.7, 91.63, 91.64% and 91.74, 91.72, 91.71% (0.5 cut-off). Every run beats v2.

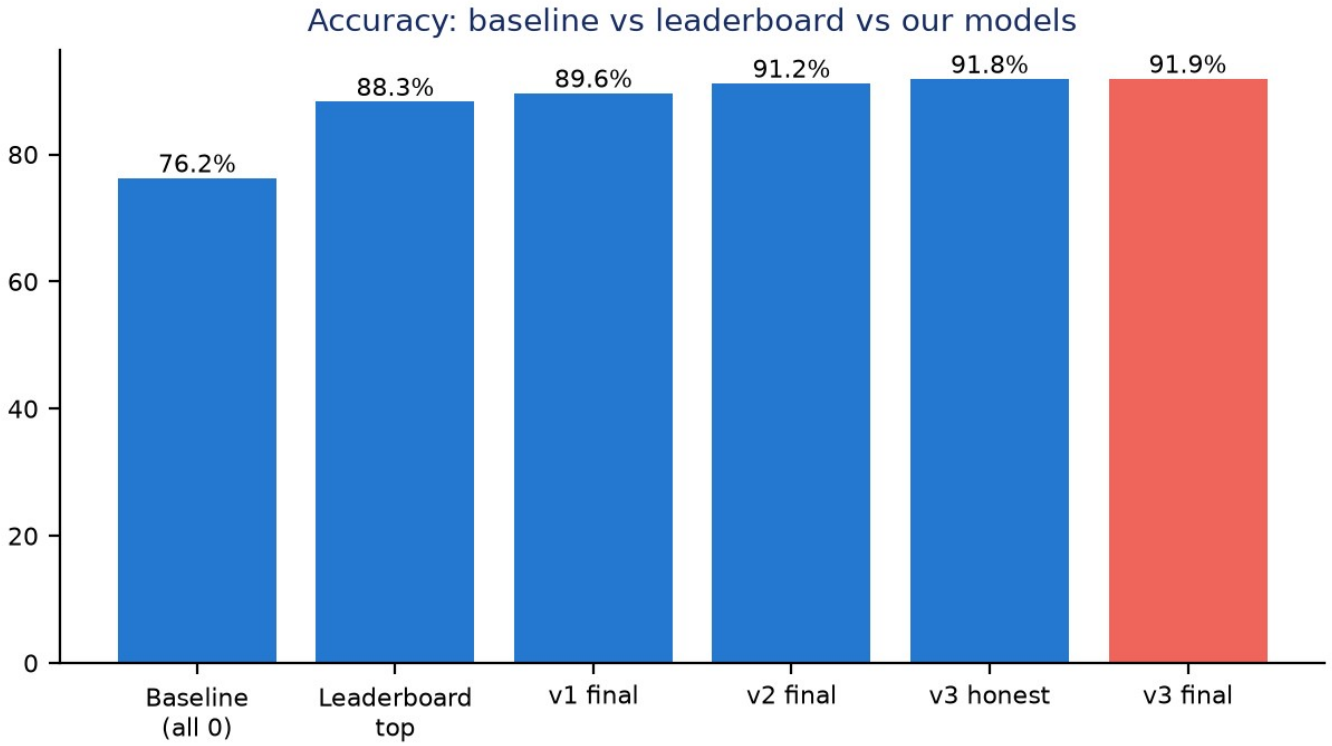
Decision rule met: +0.72 points honest gain over v2 (needed +0.10).

Final model performance (v3 stack)

	Predicted no default	Predicted default
Actual no default	TN 67,300	FP 3,745
Actual default	FN 3,833	TP 18,296

- Accuracy 91.87% = (18,296 + 67,300) / 93,174; cut-off 0.49; AUC 0.9645.
- Predicted test default rate 24.01% vs 23.75% in training.
- Where errors remain: grade A 5.89%, grade B 7.04%, grade C 8.08%, grade D 10.52%, grade E 10.04%, grade F 13.87%, grade G 13.43% wrong.
- 20.6% of errors sit close to the cut-off: genuinely ambiguous loans.

v1 vs v2 vs v3 and the leaderboard



- v2 beat v1 by +1.55 points; v3 beats v2 by +0.72 points (honest).
- Upload v3 first; keep v2, then v1, as backups.
- Expect a small drop on the hidden test set: cross-validation estimates are rarely matched exactly.

Why not 100%, and why not language models?

- Some loans have near-identical inputs but different outcomes; no model can separate them.
- A score near 100% would signal the model is reading the answer and would collapse on unseen loans.
- Gradient-boosted trees are the strongest known method for tables like this; language models are slower, costlier and less accurate here.
- AI was used only to help explain results, never to make predictions.

Business recommendations

- Track repayment from month one: flag loans whose interest received falls behind schedule.
- Re-score the active book monthly and route high-risk accounts to collections early.
- Do not rely on grade or term alone: in this data grade A and 3-year loans default most.
- Review high-balance, high-rate loans first; they carry the most risk.

Limitations and next steps

- Leaderboard score unknown until upload; cross-validation is an estimate.
- Grade and term patterns may reflect how DHDL sampled loans; confirm with the credit team.
- Next: add payment-history timelines if available, and monitor drift quarterly.
- Reproducible: fixed seeds, saved code, and automatic checks that every number matches the saved results.

Appendix: submission files

File	Model	CV accuracy	Rows
DHDL_Loan_Default_Submission_v3.csv (upload first)	v3 stack	91.87%	39,933
DHDL_Loan_Default_Submission_v2.csv (backup 1)	v2 stack	91.15%	39,933
DHDL_Loan_Default_Submission.csv (backup 2)	v1 LightGBM	89.6%	39,933
DHDL_Loan_Default_Submission_Backup.csv	v1 LightGBM variant	89.61%	39,933

All files validated: header ID,default; values 0/1 only; IDs match the sample submission.