Credit Risk Default Prediction Project Report

Project overview: This project builds a machine-learning tool to assess credit risk for consumer loans. Using historical application data, the model estimates each applicant's likelihood of default so lenders can prioritize approvals, set risk-based thresholds, and reduce potential losses. It also highlights the most influential factors behind risk, supporting more transparent, data-driven lending decisions.

Dataset Description:

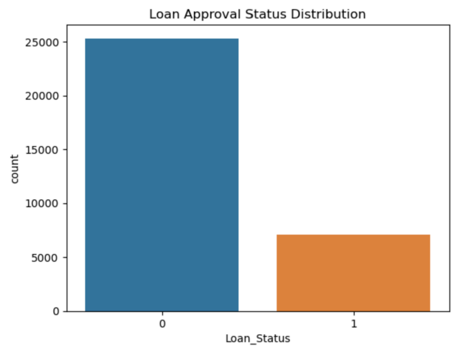
- **Scope**: Kaggle Dataset used to predict default risk. Each row is one application.
- **Target**: Loan_Status (binary) — 1 = default, 0 = non-default; class distribution is imbalanced (~22% defaults).
- **Size**: Train = 25,928 rows, Test = 6,482 rows (Total ≈ 32,410), with 17 features.
- **Key features**: demographics (age, income, employment length), loan terms (amount, interest rate, grade A–G), affordability ratio (Loan_Percent_Income), housing status, loan intent, credit history length, and a prior-default flag.
- **Preparation**: ordinal map for Loan_Grade (A→G), one-hot for Person_Home_Ownership and Loan_Intent, binary kept as 0/1
- **Testing**: Split 80/20 stratified train–test to preserve class proportions.

Data Preparation and Cleaning:

- Loaded the raw credit-risk file (12 columns, ~32.6k rows) and performed an initial audit of types, missing values, and basic statistics.
- Standardized schema: normalized column names and ensured consistent dtypes (e.g., numeric for ages/income/rates).
- Removed exact duplicate records (165 rows, ~0.51%), leaving ~32.4k rows.
- Encoded Cb_Person_Default_On_File from Y/N to 1/0 and filled any blanks with the mode.
- Standardized Person_Emp_Length to years (converted month-like entries, rounded to integers) and imputed remaining gaps with the median.
- Imputed missing Loan_Int_Rate values using the median **within each** Loan_Grade group (grade-aware fill).
- Reasonableness checks and trimming:
    - Dropped implausible ages (>90).
    - Capped extreme values to reduce outlier influence: Person_Income and Loan_Amnt at the 99th percentile.
    - Capped credit history length at 30 years and enforced Cred_Hist_Length ≤ Person_Age.
- Post-cleaning validation: no missing values remained across any feature.
- Final cleaned dataset size: ~32.4k rows and 12 features; saved as credit_risk_cleaned.csv for preprocessing and modeling.

Exploratory Data Analysis (EDA):

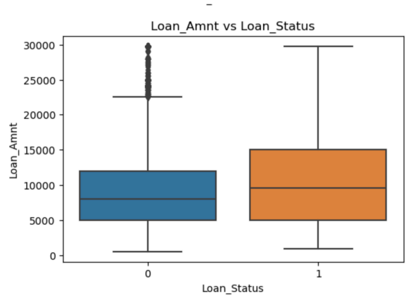
- **Target balance:** The classes are imbalanced (~78% class "0" vs ~22% class "1"), so raw accuracy is not reliable. This informed our use of PR-AUC, class weighting, and threshold tuning in modeling.

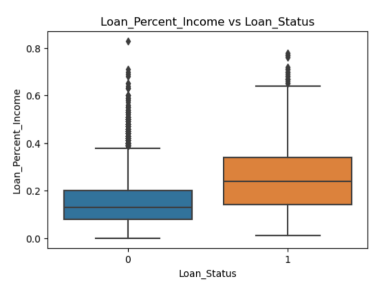



```
Loan_Status
0    78.127121
1    21.872879
Name: proportion, dtype: float64
```

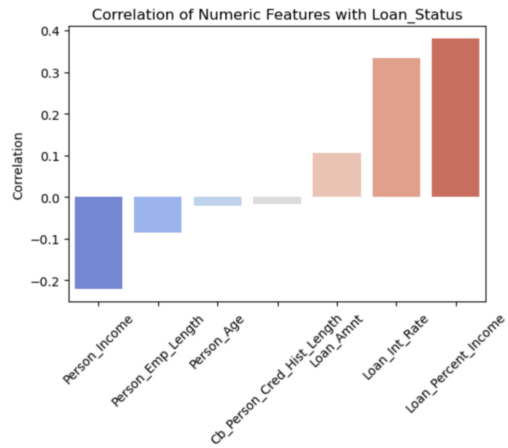
- **Univariate patterns:**
  - *Age* is concentrated in the mid-20s to early-30s; *employment length* is mostly 0–10 years.
  - *Income* and *loan amount* are right-skewed with a few large values (handled later via caps).
  - *Interest rate* ranges ~5%–23% and centers near ~11%.
  - *Loan-to-income ratio* (Loan_Percent_Income) clusters around 0.09–0.23.
  - *Credit history length* is mostly under 10 years with a long tail to 30.

- **Categorical snapshot:** Most applicants rent or have a mortgage; loan purposes are spread (education/medical/personal common); grades are concentrated in A–C with few high-risk E–G.
- **Bivariate insights (boxplots):** Compared with class "0", class "1" tends to have **higher loan amounts**, **higher interest rates**, and a **higher loan-to-income ratio**. Age and employment length show little separation between classes.
- **Correlation with target:** The strongest (though moderate) relationships are **positive** for *Loan_Percent_Income*, *Loan_Int_Rate*, and *Loan_Amnt*; **negative** for *Person_Income* and *Person_Emp_Length*. These modest linear correlations suggest non-linear tree models can add value.





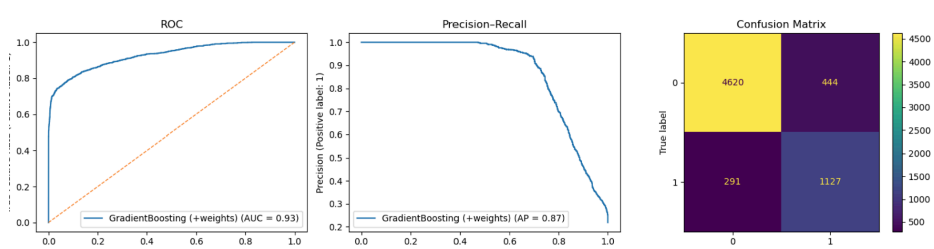
- **Implication for modeling:** Features that capture **affordability and pricing** (loan-to-income, interest rate, loan size) are the most informative. Given the class imbalance, evaluation should prioritize **PR-AUC**, **recall/precision**, and **threshold selection** over accuracy.



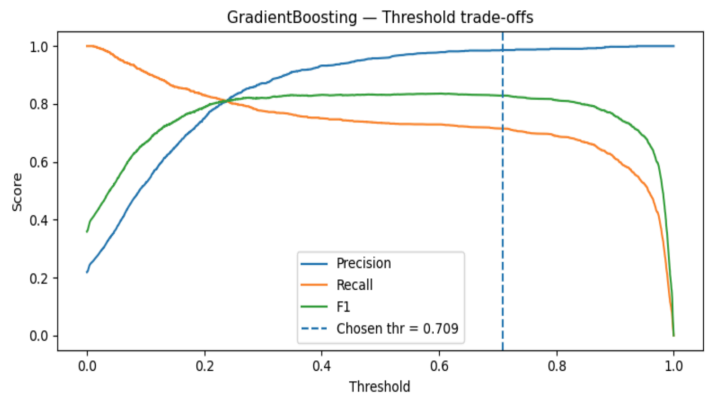
PreProcessing: We created a stratified 80/20 train–test split on Loan_Status and reset indices to avoid alignment issues. Features were grouped as: numeric (Person_Age, Person_Income, Person_Emp_Length, Loan_Amnt, Loan_Int_Rate, Loan_Percent_Income, Cb_Person_Cred_Hist_Length), nominal (Person_Home_Ownership, Loan_Intent), ordinal grade (Loan_Grade, treated A→G), and a binary flag (Cb_Person_Default_On_File, coerced to 0/1). Missing values were imputed using **train** statistics only: medians for numeric columns and modes for categorical/ordinal/binary, then the same values were applied to the test set to prevent leakage. Encoding used an ordinal map for Loan_Grade (A=0…G=6) and one-hot encoding for nominal columns (dropping the first level). Test columns were reindexed to match training columns to handle unseen categories. The final outputs were X_train_prep and X_test_prep (concatenated numeric + grade + binary + OHE) and a saved feature_names list. Sanity checks confirmed no missing values remained; class balance was printed for both splits. In this run, the prepared shapes were about 25.9k × 17 (train) and 6.5k × 17 (test).

Modelling:

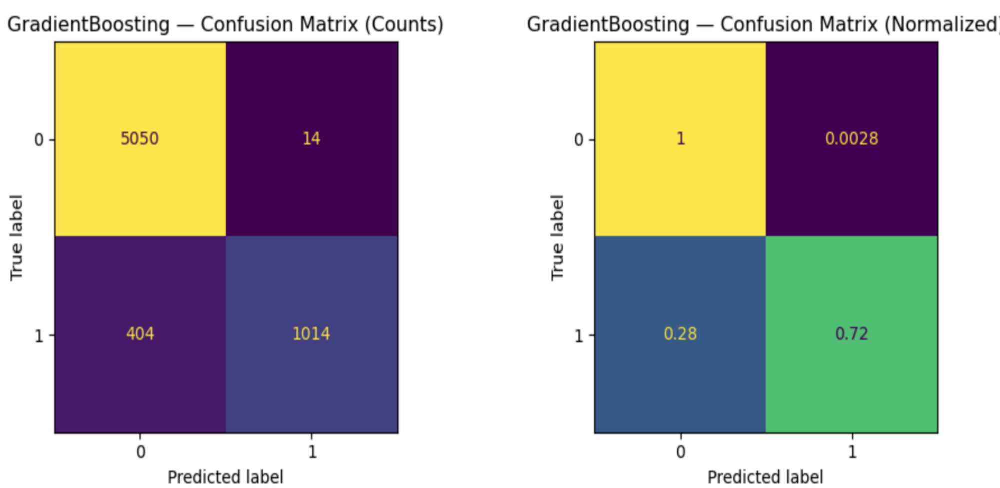
- **Goal & metrics**
  - Binary classification to predict loan default (label 1 = default) with **PR-AUC** as the primary metric due to class imbalance (~22% positives).
- **Model comparison (baseline @ thr=0.50)**
  - **Logistic Regression:** ROC-AUC ≈ **0.82**, PR-AUC ≈ **0.61** → weakest.
  - **Random Forest (balanced):** ROC-AUC ≈ **0.935**, PR-AUC ≈ **0.888**.
  - **Gradient Boosting (class_weight="balanced")**: ROC-AUC ≈ **0.927**, PR-AUC ≈ **0.870**.
- **Cross-validation & choice**
  - 3-fold CV (20 candidates each): **GB PR-AUC(CV) ≈ 0.901** vs **RF ≈ 0.885**.
  - Selected **Gradient Boosting** as the winner.
- **Winning model (hyperparameters)**
  - Tuned GB ≈ **n_estimators=392**, **max_depth=4**, **learning_rate≈0.065**, **subsample≈0.91**.



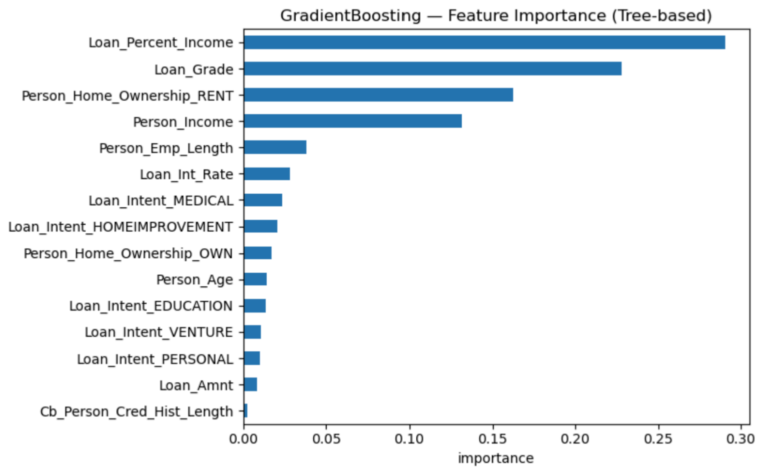
- **Threshold tuning**
  - From validation PR curve:
    - **Best-F1 threshold ≈ 0.709** (precision-first).
    - **Recall ≥0.80 at threshold ≈ 0.493** (recall-first).



- **Final test performance (GB @ thr=0.709)**
  - **ROC-AUC 0.945**, **PR-AUC 0.902**, **Accuracy 0.936**.
  - Class 1 (default): **Precision 0.986**, **Recall 0.715**, **F1 0.829**.
  - Confusion matrix (counts): **[[5050, 14], [404, 1014]]**.



- **Top drivers (GB feature importance)**
  - **Loan_Percent_Income**, **Loan_Grade**, **Home_Ownership=RENT**, **Person_Income**, **Emp_Length**, **Loan_Int_Rate**.



- **Recommendation**
    - **Deploy tuned Gradient Boosting** and set the operating threshold to match costs:
        - precision-first: **thr ≈ 0.71**;
        - recall-first: **thr ≈ 0.49–0.50**.
    - Re-tune periodically; consider probability calibration if thresholding is sensitive.

Business Implications and Key Takeways:

- **Business value:** Cut charge-offs by flagging high-risk applicants, price/limit by risk using calibrated probabilities, automate low-risk approvals to reduce ops cost and TAT, and prioritize collections—improving portfolio quality and capital efficiency.
- **Best model:** Gradient Boosting (test **ROC-AUC 0.945**, **PR-AUC 0.902**).
- **Thresholding:** At **thr≈0.71** → **precision ≈0.99**, **recall ≈0.72** (few false positives). At **thr≈0.49** → **recall ≥0.80** if business prefers fewer missed defaults.
- **Impact snapshot (thr≈0.71):** Confusion matrix ≈ [[5050, 14], [404, 1014]] → captures ~72% of defaults with ~1% false-positive rate.
- **Key drivers:** Loan_Percent_Income, Loan_Grade, Home_Ownership=RENT, Person_Income, Emp_Length, Loan_Int_Rate (affordability and grade dominate).
- **Why PR-AUC:** Class is imbalanced (~22% default); PR-AUC better reflects ranking of positives than accuracy/ROC alone.
- **Execution notes:** Set operating threshold by product/segment, use risk-based pricing curves, and monitor calibration, drift, and fairness; re-tune quarterly or when metrics shift.