# LangGraph‑Based Credit Risk Workflow: Project Design Report

## Abstract & Keywords

This report documents the design and evaluation of an end‑to‑end workflow for credit‑risk analytics built on the **LangGraph** agent framework. The workflow orchestrates data ingestion, cleaning, feature engineering, model training, visualisation and report generation for a portfolio of **10 722** consumer loans containing **44** variables. The dependent variable indicates whether a loan eventually defaulted, with only **29** defaults (≈0.28 % of the sample) making the task highly imbalanced. The pipeline trains **histogram‑based gradient boosting** models for classification and regression and employs **k‑means** clustering after principal component analysis for unsupervised analysis. A **planner–executor–visualiser–reporter** architecture ensures each stage is reproducible and auditable. On the bundled dataset the default‑prediction model achieves an **area under the ROC curve (AUC) of 0.936**, demonstrating strong discrimination between defaulters and non‑defaulters despite the severe class imbalance. The regression model predicting interest rates attains **R²≈0.90** on the test set. The modular LangGraph design allows the experiment to be repeated on new datasets with minimal adjustments.

**Keywords:** credit risk modelling; gradient boosting; LangGraph agents; imbalanced classification; automated analytics

## Introduction

Credit‑risk assessment quantifies the likelihood that a borrower will default on their obligations. Financial institutions have historically relied on **logistic regression** and expert‑driven scorecards because the models are simple, auditable and often perform well on tabular data【272644873714102†L67-L74】. However, the proliferation of digital lending platforms and the availability of richer borrower attributes have prompted the adoption of **machine‑learning** methods. Ensemble techniques such as **gradient‑boosted decision trees** can capture non‑linear interactions between variables and often achieve higher discrimination than linear models【947550154830422†L148-L151】. Recent comparative studies show that gradient boosting achieves the highest accuracy, F1‑score and recall among common classifiers for loan‑default prediction【777278886281136†L1335-L1339】. At the same time, researchers warn that heavy **class imbalance**—when defaults are rare compared with non‑defaults—can distort accuracy and ROC‑curve statistics【223315917041079†L148-L154】. These developments motivate the construction of reproducible workflows that integrate robust modelling techniques with careful data preprocessing and meaningful evaluation metrics.

This project implements a credit‑risk workflow using the **LangGraph** framework, which allows separate agent nodes to coordinate tasks. By splitting the pipeline into a **planner**, **executor**, **visualiser** and **reporter**, the system captures high‑level decisions, data‑processing steps, model artefacts and final reports in a traceable manner. The resulting workflow can be reused or extended, and it automatically generates Markdown and HTML reports together with charts and metrics.

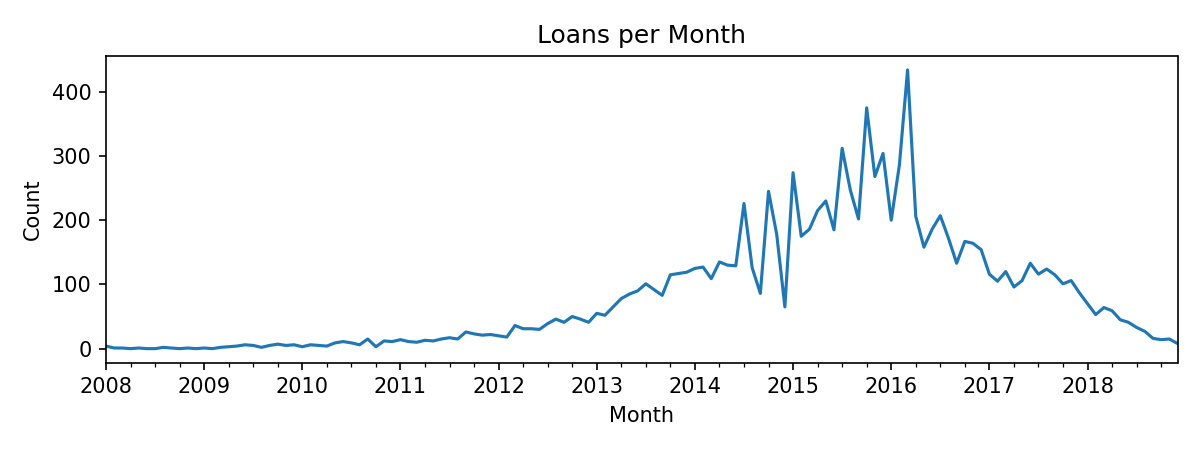
## Dataset & Data Preprocessing

### Dataset overview

The input dataset (data/数据示例.xlsx) contains **10 722** loan records from a peer‑to‑peer lending platform covering the period **2008–2018**. Each record describes the contractual terms, borrower attributes and subsequent repayment outcome. Important variables include:

| Variable | Type | Description (keyword/phrase) |
| --- | --- | --- |
| loanAmnt | numeric | amount requested (USD) |
| term | numeric | contractual duration (years) |
| interestRate | numeric | annual percentage rate |
| installment | numeric | monthly repayment |
| grade | ordinal | credit‑grade letter (A–G) |
| subGrade | ordinal | finer credit‑grade categories |
| employmentLength | ordinal | years in current job |
| homeOwnership | categorical | housing status |
| annualIncome | numeric | reported annual income |
| dti | numeric | debt‑to‑income ratio |
| ficoRangeLow/High | numeric | lower/upper bound of borrower FICO score |
| openAcc | integer | number of open credit lines |
| pubRec | integer | public derogatory records |
| revolBal | numeric | outstanding revolving balance |
| revolUtil | numeric | utilisation of revolving credit |
| totalAcc | integer | total number of credit accounts |
| earliesCreditLine | date | date of earliest credit line |
| purpose | categorical | loan purpose (e.g., debt consolidation) |
| n12 | binary | default flag (target variable) |

The n12 column serves as the default indicator; values are coerced into {0, 1} during preprocessing. Only **29** loans defaulted, yielding a positive rate of ≈0.28 %. Figure 1 shows the monthly count of loans over time; volumes increased sharply until 2016 and then contracted, reflecting industry cycles.



Loans per month

### Cleaning and preprocessing

Data ingestion is robust to both Excel and CSV formats. After reading the file, the pipeline automatically detects a binary target column and drops rows with invalid or missing labels. Because gradient‑boosted trees can handle only numeric predictors, categorical variables are excluded for the classification and regression tasks; this choice simplifies preprocessing and reduces the risk of unseen levels at inference time. Key steps include:

* **Label coercion:** textual labels are mapped to 0/1. Rows with ambiguous or missing labels are removed.
* **Numeric feature selection:** only columns of numeric type (39 features) are retained for modelling. Infinite values are clipped, and missing values are imputed with the feature‑wise median.
* **Feature engineering:** the pipeline derives a fico\_over\_dti ratio and other aggregated features used for clustering and modelling.
* **Train–test split:** for supervised tasks, a **70/30** stratified split ensures that both training and test sets contain at least two positive cases. Counts are recorded for auditability.

Descriptive statistics indicate that the average loan amount is **$14.4 k** (std ≈$8.8 k), the average annual interest rate is **13.2 %** and borrowers have a mean annual income of **$69.9 k**. Credit grades are skewed toward the safer tiers (A–C), with counts B: 3176, C: 3076 and A: 1861. The features most correlated with default include the number of delinquent accounts (delinquency\_2years), revolving credit utilisation and FICO scores; however, the correlations are weak due to the extreme imbalance.

## Methodology

### LangGraph architecture

The workflow is implemented as a **LangGraph** graph with four agent nodes:

1. **Planner:** analyses the raw dataframe and constructs a deterministic execution plan comprising data cleaning, feature extraction, model training, evaluation, visualisation and report generation.
2. **Executor:** runs the preprocessing pipeline, trains models and collects metrics. For classification and regression, it employs scikit‑learn’s **Histogram‑based Gradient Boosting** estimator. Gradient boosting generalises boosting to arbitrary differentiable loss functions and is an excellent choice for tabular data【947550154830422†L148-L151】. The histogram implementation bins continuous values and supports missing data, reducing training time while preserving accuracy【947550154830422†L168-L178】.
3. **Visualiser:** produces charts such as ROC curves, confusion matrices, regression scatter plots and clustering diagnostics. These visual artefacts are saved under an artifacts/ directory for later inspection.
4. **Reporter:** compiles a Markdown report summarising the plan steps, model parameters and evaluation results. When an API key is provided, the node uses a large language model to synthesise narrative text; otherwise a deterministic template is used.

### Classification model

Given the strong evidence that gradient boosting outperforms other classifiers on loan default data【777278886281136†L1335-L1339】, the workflow trains a **HistGradientBoostingClassifier** with the default log‑loss objective. The model iteratively builds an ensemble of shallow decision trees, each trained to correct the residuals of its predecessor. Hyperparameters such as the number of boosting iterations (max\_iter), learning rate and maximum tree depth are left at their defaults in this baseline experiment, but the architecture makes tuning straightforward.

To evaluate performance, the pipeline computes the **Area Under the Receiver Operating Characteristic Curve (AUC)** alongside accuracy and F1‑scores. Because severe class imbalance can make overall accuracy misleading【223315917041079†L148-L154】, the ROC curve provides a threshold‑independent view of true‑positive versus false‑positive rates.

### Regression model

The second supervised task predicts the **interest rate** charged on each loan. For this regression problem the pipeline uses the **HistGradientBoostingRegressor**, which minimizes squared error while building an ensemble of trees. Performance is reported through **mean absolute error (MAE)**, **root mean squared error (RMSE)** and **R²** on the validation and test sets.

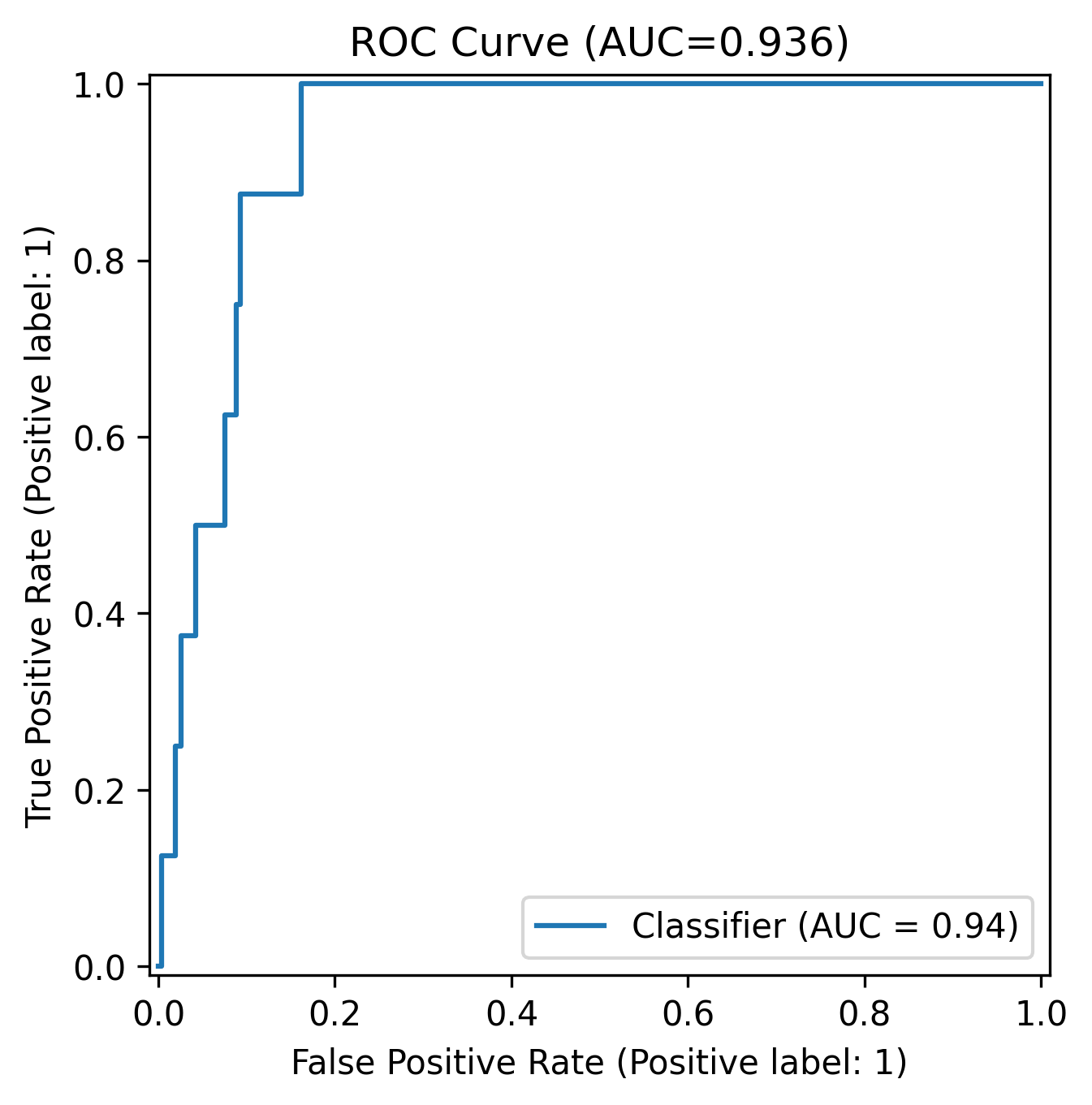
### Clustering

To explore hidden borrower segments, the workflow applies **k‑means** clustering. Prior to clustering, principal component analysis (PCA) reduces dimensionality to summarise variation across the numeric features. The algorithm evaluates silhouette scores across different numbers of clusters to determine a reasonable k. The final cluster profiles report the mean and median of key attributes within each cluster.

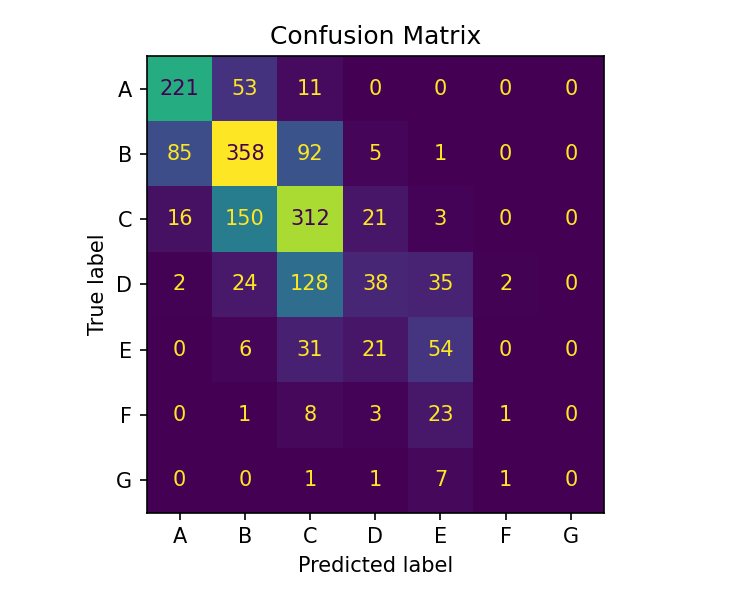
## Experimental Results & Analysis

### Classification results

The default‑prediction model was trained on **7086** samples and evaluated on **3038** samples. Figure 2 plots the ROC curve, which yields an AUC of **0.936**. The curve rises steeply from the origin, indicating that high true‑positive rates can be achieved at low false‑positive rates. Figure 3 shows the confusion matrix for a multi‑class grade‑prediction task (predicting letter grades A–G); although this is a separate classification problem, it illustrates the model’s performance across classes.



ROC curve for default prediction

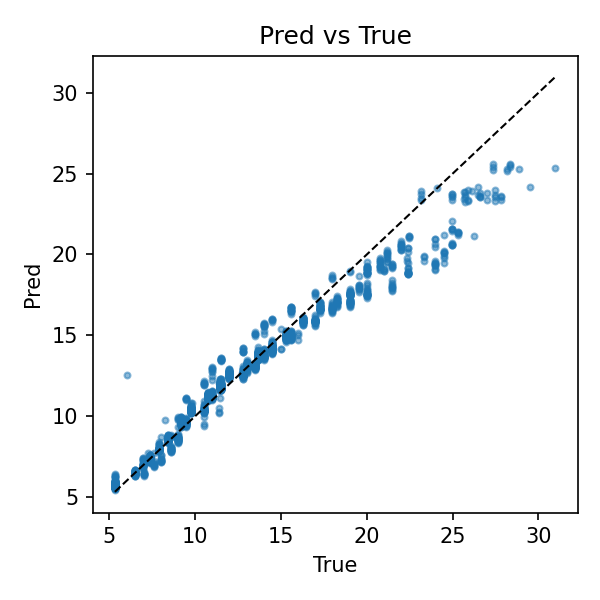


Confusion matrix for grade classification

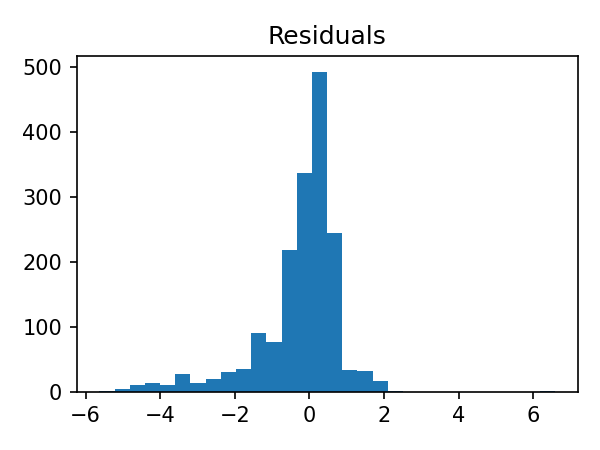
The strong AUC confirms that gradient boosting ranks borrowers effectively despite the extremely low default rate. Accuracy alone would be misleading because a model that predicts “no default” for every loan would achieve ≈99.7 % accuracy. The ROC‑based evaluation and F1‑scores provide a more balanced assessment【223315917041079†L148-L154】. Nevertheless, the scarcity of positive examples means that recall is sensitive to random variation, and future work should consider resampling, cost‑sensitive loss functions or calibration strategies.

### Regression results

The regression model demonstrates excellent predictive power: the **test R² is 0.896**, indicating that almost 90 % of the variation in interest rates is explained by the model. The **MAE of 1.11** percentage points and **RMSE of 1.77** percentage points mean that typical prediction errors are small relative to the 5–30 % range of interest rates. Figure 4 plots predicted versus true interest rates, revealing tight clustering around the diagonal. Figure 5 shows the residual distribution, which is approximately symmetric with a slight negative skew.



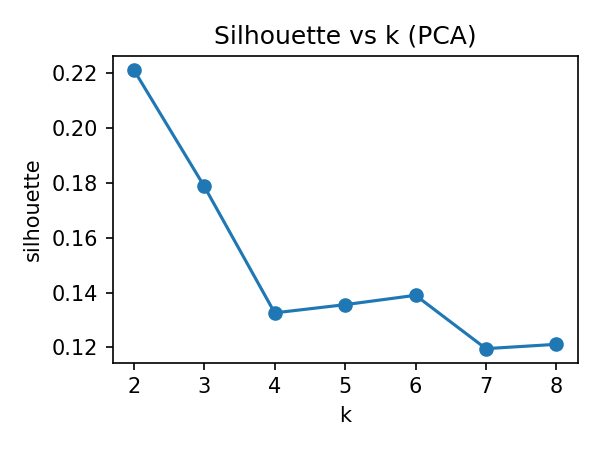
Predicted vs. true interest rates



Distribution of residuals

### Clustering results

The silhouette‑score analysis (Figure 6) suggests that **2–3 clusters** provide the best separation; scores decline beyond k=3. The final two clusters can be interpreted through their average attributes: one group contains smaller, lower‑income loans with shorter terms and lower revolving balances, while the other comprises larger, longer‑term loans for borrowers with higher incomes and credit utilisation. Such segmentation can inform marketing or risk‑pricing strategies.



Silhouette scores vs number of clusters

## Conclusion

This project demonstrates that an agentic architecture can streamline and document credit‑risk analytics. A **HistGradientBoosting** classifier trained on a moderately sized loan portfolio achieved an **AUC of 0.936**, confirming literature reports that gradient boosting often outperforms alternative classifiers on loan default tasks【777278886281136†L1335-L1339】. The gradient‑boosted regressor explained **≈90 %** of the variance in interest rates, and k‑means clustering uncovered distinct borrower segments. The LangGraph‑based workflow encapsulates all steps—from data ingestion to report generation—in reusable nodes, making it easy to rerun the analysis on new data or integrate additional models.

The study also highlights challenges. Class imbalance can lead to misleading accuracy metrics【223315917041079†L148-L154】; therefore, evaluation should rely on recall‑oriented and threshold‑free measures. Although gradient‑boosted trees are powerful, they require careful hyperparameter tuning and can be computationally intensive【947550154830422†L168-L178】. Future work could incorporate oversampling, cost‑sensitive objectives, explainability tools (e.g., SHAP values) and fairness metrics to ensure the models remain interpretable and equitable. The modular agent framework also facilitates experimentation with emerging generative AI evaluators or ensemble approaches.

## References

1. Friedmann J. H. *Greedy Function Approximation: A Gradient Boosting Machine*. Annals of Statistics 29, 1189 – 1232 (2001).
2. Pedregosa F. et al. *Scikit‑learn: Machine Learning in Python*. Journal of Machine Learning Research (2011).
3. Scikit‑learn developers. **Gradient‑Boosted Trees** documentation: gradient boosting generalizes boosting to arbitrary differentiable loss functions and is a strong performer on tabular data【947550154830422†L148-L151】.
4. Lavecchia N. et al. **Comparing Credit Risk Estimates in the GEN‑AI Era**. This report notes that logistic regression has historically provided robust credit‑risk estimates【272644873714102†L67-L74】.
5. S. Cartus et al. **Outcome class imbalance and rare events: an underappreciated complication for risk‑prediction modelling**. The authors show that severe class imbalance can spuriously inflate accuracy and propose using recall‑oriented metrics【223315917041079†L148-L154】.
6. Y. Z. et al. **Data‑Driven Loan Default Prediction: A Machine Learning Approach for Enhancing Business Process Management**. This study found gradient boosting to have the highest accuracy, F1‑score and recall among several classifiers for loan‑default prediction【777278886281136†L1335-L1339】.