POLICY OPTIMIZATION FOR FINANCIAL DECISION-MAKING

(Assessment Project Report)

Submitted by:

Dhruv Rajoria

B.E. (Computer Science and Engineering)

Thapar Institute of Engineering & Technology

Roll No: 102217050

Email: drajoria_be22@thapar.edu

Submitted to:

Shodh AI

(Machine Learning project)

Date of Submission: 30 October 2025

Punjab, India

OBJECTIVE

This project focuses on improving the loan approval process for a fintech company through data-driven decision-making.

Using the Lending Club Loan Data (2007–2018), the objective is to design intelligent systems that can determine whether a loan should be approved or denied, balancing profitability against the risk of default.

Two complementary approaches were used:

1. A Supervised Deep Learning model to predict loan default risk.
2. An Ofline Reinforcement Learning (RL) agent to learn an optimal approval policy that maximizes financial return.

DATA OVERVIEW AND PREPROCESSING

The dataset includes detailed financial and demographic information about loan applicants and their repayment histories.

Key preprocessing steps included:

- Converting loan_status to binary (0 = Fully Paid, 1 = Defaulted or Charged Off).
- Imputing missing values using median (numeric) and constant (categorical).
- Converting percentage-based features like int_rate and revol_util into numeric form.
- Encoding categorical variables using one-hot encoding and scaling numeric features.
- Sampling 200,000 records to reduce computation while preserving data diversity.

The final dataset contained 137 engineered features used for model training and evaluation.

MODEL 1 – SUPERVISED DEEP LEARNING CLASSIFIER

A **Multi-Layer Perceptron (MLP)** neural network was implemented using PyTorch to predict the likelihood of loan default.

Architecture Details:

- **Input Layer:** 137 processed features
- **Hidden Layers:** [256, 128] with ReLU activations, Dropout (0.2), and Batch Normalization
- **Output Layer:** Sigmoid neuron for binary classification

Training Setup:

- **Optimizer:** Adam (learning rate = 0.001)
- **Loss Function:** Binary Cross-Entropy
- **Batch Size:** 1024
- **Epochs:** 10
- **Data Split:** 80% training, 20% testing

Model Results:

- **AUC (ROC):** 0.88
- **F1-Score:** 0.79
- **Precision:** 0.81
- **Recall:** 0.76

Interpretation:
The model demonstrated strong discrimination between defaulting and fully paid borrowers. Adjusting the approval threshold allows for an optimal trade-off between financial gain and default risk.

MODEL 2 – OFFLINE REINFORCEMENT LEARNING AGENT

To directly optimize profitability, the problem was reframed as an **Offline Reinforcement Learning** task using the **Conservative Q-Learning (CQL)** algorithm.

Environment Design:

- **State (s):** Preprocessed borrower features
- **Action (a):** {0 = Deny, 1 = Approve}
- **Reward (r):**
  - Approve + Fully Paid → +loan_amount × interest_rate
  - Approve + Default → -loan_amount
  - Deny → 0

The RL agent was trained on historical data without exploration, learning which loans maximize long-term profit.

Evaluation Metric:
The **Estimated Policy Value (EPV)** indicated that the RL policy achieved a higher average reward compared to the supervised model, showing it learned more profitable approval strategies.

COMPARATIVE ANALYSIS

Model	Metric	Mean reward	Observation
Deep Learning (Best-F1 policy)	AUC: 0.88, F1:0.79	+12.6	Balanced risk and reward
RL (CQL Agent)	EPV ≈ +13.8	+13.80	Higher expected return
RL Baseline (Threshold = 0.3)	F1:0.74	+11.2	Moderate gain
Deny All	-	0	Neutral (no gain/loss)
Approve All	-	-45.0	Heavy loss due to defaults

Key Insight:
The RL agent occasionally approved medium-risk loans with high-interest returns that the supervised model rejected, resulting in higher cumulative rewards.

LIMITATIONS

- **Selection Bias:** Dataset includes only approved loans, omitting rejected applications.
- **Static Data:** RL agent cannot explore new scenarios beyond historical patterns.
- **Simplified Reward System:** Ignores repayment periods and recovery rates.
- **No Real-World Constraints:** Does not include regulatory or credit-limit restrictions.
- **Generalization Risk:** Borrower profiles may differ from future applicants.

FUTURE DIRECTIONS

Deployment Strategy:
Deploy the **Deep Learning model** first for production, as it provides stable, interpretable results.
Continue testing the **RL policy** in a simulation environment to validate its real-world profitability.

Data Improvements:

- Include **rejected loan applications** and follow-up payment behaviour.
- Integrate **macroeconomic data** such as inflation and employment trends.

Algorithmic Enhancements:

- Experiment with **Distributional RL**, **Advantage Weighted Regression (AWR)**, or **Bayesian RL**.
- Combine **tree-based models (XGBoost)** with **policy gradient methods** for better explainability.

Business Integration:

- Develop dashboards showing profit-risk trade-offs for policy tuning.
- Introduce dynamic approval thresholds that adapt to changing market conditions.

CONCLUSION

Both models offer distinct advantages:

- The **Deep Learning model** provides reliable risk assessment and strong predictive accuracy, making it ideal for deployment.
- The **Reinforcement Learning agent** demonstrates greater profit optimization potential by learning approval strategies directly from reward patterns.

The best future direction involves a **hybrid approach**, combining the DL model's predictive precision with the RL agent's decision-making flexibility.
This integration can lead to a next-generation, **AI-powered lending system** that makes smarter, data-informed financial decisions.