CS613 - Machine Learning

Final Project

Loan Approval Prediction

Kahf Hussain, Uday Pothuri

Introduction:

Problem:

- Predicting loan approval is a complex challenge for financial institutions due to various factors influencing decision-making.
- Accurately, predicting loan approvals can streamline operations and reduce risks associated with lending.

Objective:

- The aim of our study was to compare the performance of multiple learning models to identify the most effective approach for predicitng loan approvals.
- Additionally, the objective includes analyzing feature importance to uncover the key factors driving these predictions across the dataset.

Dataset Overview - Feature Variables:

Demographical Variables:

1. Gives background information about the applicants.
2. Examples: Age, MaritalStatus, NumberOfDependents, EducationLevel

Employment and Income Information Variables:

1. Information about an applicants' job status and financial earnings.
2. Examples: EmploymentStatus, Experience, JobTenure, AnnualIncome, MonthlyIncome

Loan Specific Information:

1. Contains specifics of the loan request, such as amount and duration.
2. Examples: LoanAmount, LoanDuration, LoanPurpose, BaseInterestRate, InterstRate

Dataset Overview - Feature Variables:

Financial Credit Information

1. Summarizes creditworthiness and debt obligations of applicants.
2. Examples: CreditScore, MonthlyDebtPayments, NumberOfOpenCreditLines, PaymentHistory

Asset and Liability Information:

1. Lists applicants' financial assets, liabilities, and net worth.
2. Examples: HomeOwnershipStatus, SavingsAccountBalance, TotalAssets, TotalLiabilities

Dataset Overview - Target Variable:

LoanApproved

- Type: Boolean Variable
- Description:
  - 1 if the loan application is approved
  - 0 if the loan application is rejected.
- Class Priors:
  - ~ 75% loan applications are rejected => LoanApproved = 0
  - ~ 25% loan application are accepted => LoanApproved = 1

Preprocessing:

Dropping Unnecessary Columns

- Columns like ApplicationDate and RiskScore were removed as they added no predictive value or were redundant.

Categorical Feature Encoding

- Ordinal encoding was applied to features like Employment Status, Education Level, and Martial Status to maintain ordered relationships.
- One-hot encoding was used for LoanPurpose to convert it into binary columns.

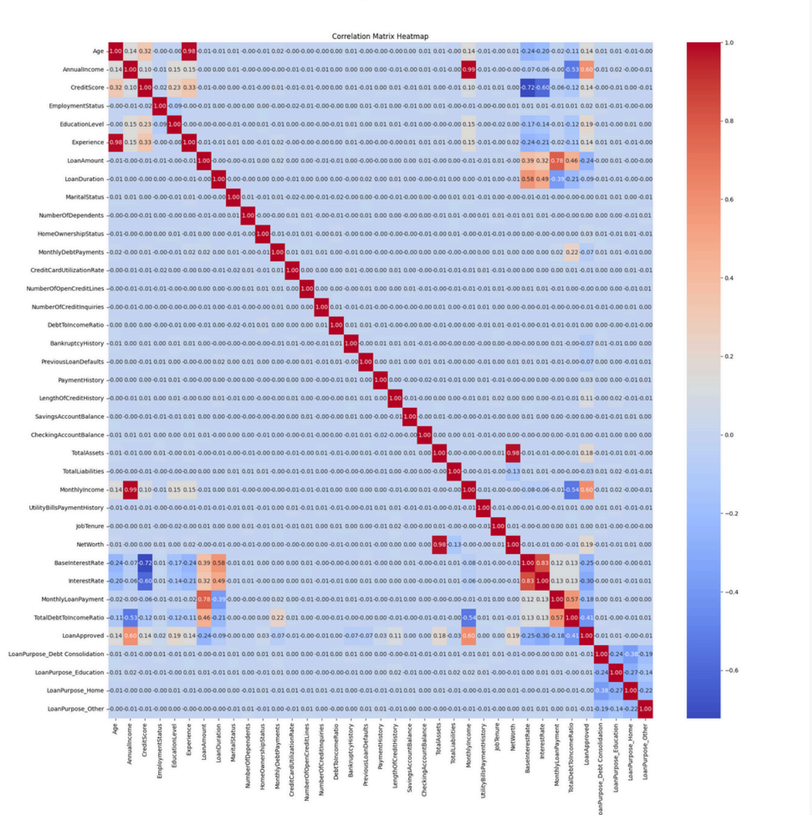
Normalization of Continous Features

- Since most of our features are numerical with various scales, we normalized them using z-score standardization.

Preprocessing (Correlation Analysis):

Correlation Analysis

- In order to make sure, we don't retain highly correlated features, we plotted a correlation heatmap and set a threshold of 0.7 to remove highly correlated variables.



- Here's the list of features which we found highly correlated:
  - Age and Experience: 0.98 -> (Removed Experience)
  - AnnualIncome and MonthlyIncome: 0.99 -> (Removed AnnualIncome)
  - CreditScore and BaseInterestRate: -0.72 -> (Removed BaseInterestRate)
  - LoanAmount and MonthlyLoanPayment: 0.78 -> (Removed LoanAmount)
  - TotalAssets and NetWorth: 0.98 -> (Removed TotalAssets)
  - BaseInterestRate and InterestRate: 0.83 -> (Removed BaseInterestRate)

Prior Works:

1) Ensemble Models for Loan Approval (Kumar and Singh, 2023)

- Proposed a hybrid approach integrating Random Forest, Gradient Boosting, and Logistic Regression.
- Achieved 91.8% accuracy, emphasizing the effectiveness of combining diverse models.
- Highlighted limitations of Gradient Boosting due to convergence issues, inspiring exploration of complementary methods.

2) Deep Learning Hybrid Model (Mahgoub, 2024)

- Combined CNN and MLPs for improved prediction, achieving 93.2% accuracy.
- Showed deep learning's strength in capturing non-linear patterns and unstructured data.
- Inspired us to use Logistic Regression as a baseline for benchmarking advanced models.

Prior Works (Contd.)

3) Support Vector Machines (SVM) (Yidiz, 2022)

- Demonstrated SVMs' ability to handle imbalanced data with hyperparameter tuning (e.g., kernel type).
- Noted that correlated features significantly degraded SVM performance, leading us to apply PCA and feature selection.

4) Voting-Based Ensembles (Alhamid, 2020)

- Differentiated between hard and soft voting strategies in ensemble learning.
- Highlighted soft voting's advantage in weighting model confidence for predictions.
- Influenced our exploration of combining SVMs and Logistic Regression with voting methods.

Method 1: Logistic Regression

Why?

Logistic Regression serves as a baseline model due to its simplicity and interpretability for binary classification tasks.

Mathematical Foundation:

The model primarily predicts probabilites using the Sigmoid function and aims to minimize the log-loss function:

$$\sigma(z) = \frac{1}{1+e^{-z}} \quad \text{where} \quad z = w^T x + b \qquad J = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\right]$$

Additionally, for optimization, we use Gradient Descent for updating our weights and bias:

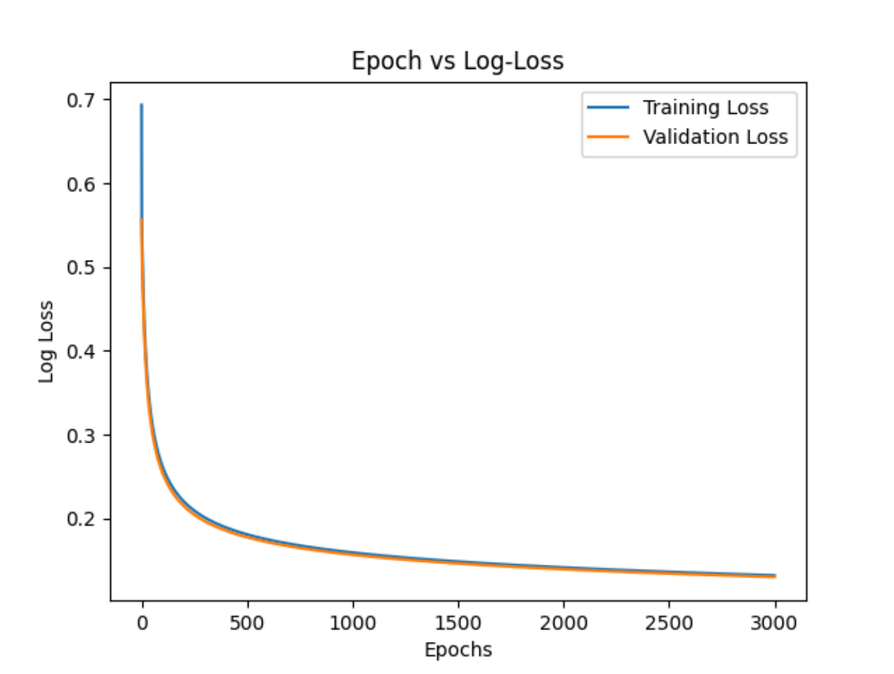
$$w \leftarrow w - \eta\frac{\partial J}{\partial w}, \quad b \leftarrow b - \eta\frac{\partial J}{\partial b}$$ where η is our learning rate.

Method 1: Logistic Regression (Results)

Performance Metrics:

- **Training Set**: Accuracy: 94.57%; Precision: 89.69%; Recall: 87.75%; F1-Score: 88.71%
- **Validation Set**: Accuracy: 94.72%; Precision: 89.01%; Recall: 87.97%; F1-Score: 88.49%

Log-Loss Curve:



Top 5 Relevant Features:

- **Monthly Income**: +3.1663
- **Interest Rate: -**2.5248
- **TotalDebtToIncomeRatio:** -2.0674
- **BankruptcyHistory:** -1.5101
- **MonthlyLoanPayment:** -1.4636

Credit Score: -1.501 **(Interesting Feature)**

Method 2: Linear SVM

Why?

Linear SVM was chosen to create a decision boundary that maximizes the margin between two classes. Its simplicity and effectiveness in high-dimensional spaces make it an excellent starting point for evaluating linear separability in the data

Mathematical Foundation:

The model aims to maximize the margin while minimizing classification errors by solving the following optimization problem

$$J(\mathbf{w}, b) = \frac{\lambda}{2}\|\mathbf{w}\|^2 + \frac{1}{n}\sum_{i=1}^{n} \max(0, 1 - y_i(\mathbf{w} \cdot \mathbf{x}_i - b))$$

Where λ is the regularizaiton parameter to balance margin maximization and classification error minimization.

Method 2: Linear SVM (contd.)

In order to further optimize our model, we use Gradient Gescent to solve the following optimization problem:

For correctly classified points with sufficient margin: $(y_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1)$

$$\frac{\partial J}{\partial \mathbf{w}} = \lambda \mathbf{w}, \quad \frac{\partial J}{\partial b} = 0$$

For misclassified samples or those with insufficient margin: $(y_i(\mathbf{w} \cdot \mathbf{x}_i - b) < 1)$

$$\frac{\partial J}{\partial \mathbf{w}} = \lambda \mathbf{w} - y_i \mathbf{x}_i, \quad \frac{\partial J}{\partial b} = -y_i$$

Additionally, just like Logit we iteratively update our weights and bias to ensure convergence.

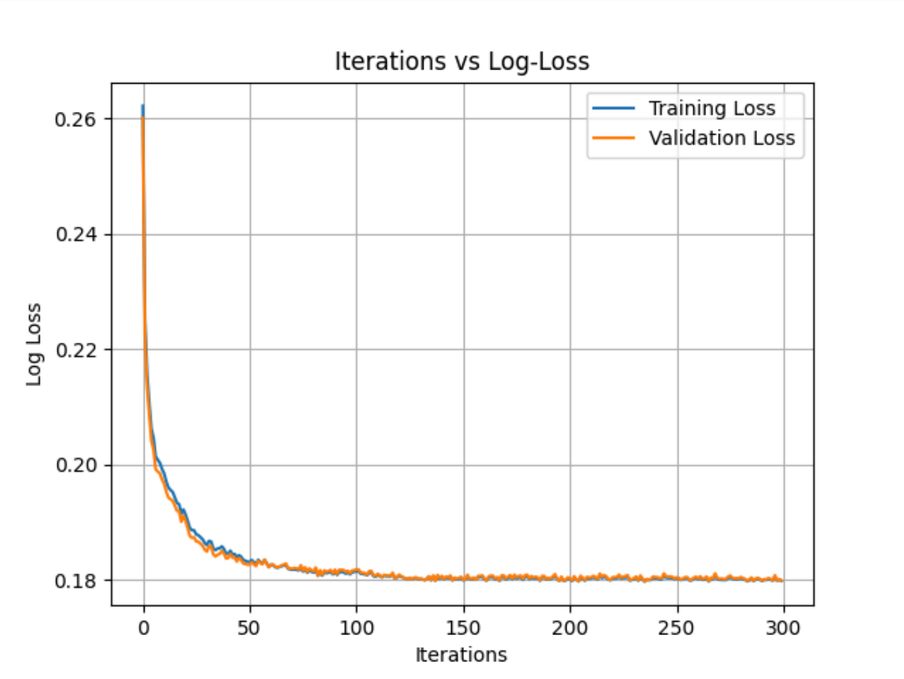
$$w \leftarrow w - \eta \frac{\partial J}{\partial w}, \quad b \leftarrow b - \eta \frac{\partial J}{\partial b}$$ where η is our learning rate.

Method 2: Linear SVM (Results)

Performance Metrics:

- **Training Set**: Accuracy: 95.49%; Precision: 89.58%; Recall: 91.94%; F1-Score: 90.75%
- **Validation Set**: Accuracy: 95.27%; Precision: 89.15%; Recall: 90.99%; F1-Score: 90.06%

Log-Loss Curve:



Top 5 Relevant Features:

- **Monthly Income**: +1.5457
- **Interest Rate:** -1.2328
- **TotalDebtToIncomeRatio:** -1.1135
- **BankruptcyHistory:** -0.7499
- **MonthlyLoanPayment:** -0.6994

Credit Score: -0.5770 **(Interesting Feature)**

Method 3: Polynomial SVM

Why?

Polynomial SVM was chosen to explore non-linear decision boundaries that our previous models couldn’t capture. We tested both degree 2 and degree 3 kernels to compare performances.

Mathematical Foundation:

NOTE: Specifically for these models, we implemented PCA to reduce dimensionality (for faster calcuations) retaining only 95% variance.

The polynomial kernel transforms the features using the following kernel function:

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + 1)^d$$

Where:

- **x_i, x_j:** Feature vectors for the i-th and j-th samples.
- d: Degree of the polynomial kernel (2 or 3 in our case).
- Constant +1: Ensures positive values for kernel computations.

Method 3: Polynomial SVM (contd.)

Using the kernel function, our optimization problem becomes:

$$J(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

Subject to the following constraints:

$$0 \leq \alpha_i \leq C, \quad \sum_{i=1}^{n} \alpha_i y_i = 0$$

Where:

- α_i: Lagrange multiplies for support vectors.
- C: Regularization parameter to balance margin width and classification error.

Method 3: Polynomial SVM (Results)

Performance Metrics for degree 2:

- **Training Set**: Accuracy: 92.68%; Precision: 94.63%; Recall: 73.78%; F1-Score: 82.92%
- **Validation Set**: Accuracy: 92.08%; Precision: 92.49%; Recall: 72.18%; F1-Score: 81.08%

Performance Metrics for degree 3:

- **Training Set**: Accuracy: 96.08%; Precision: 95.80%; Recall: 87.57%; F1-Score: 91.50%
- **Validation Set**: Accuracy: 90.95%; Precision: 84.00%; Recall: 76.05%; F1-Score: 79.82%

Method 4: Ensemble Models

Why?

Ensemble models combine the strengths of multiple base models to improve overall performance.

In our case, we combined Logistic Regression and Linear SVM using two ensemble strategies:

- **Hard Voting:** Relies on majority rule from the base models.
- **Soft Voting:** Averages the predicted probabilities for weighted decisions.

Mathematical Foundation (Hard Voting):

Hard Voting predicts the final labels based on the majority vote:

$$\text{Ensemble Prediction} = \text{mode}(\text{Logit Prediction}, \text{SVM Prediction})$$

Method 4: Ensemble Models (contd.)

Mathematical Foundation (Soft Voting):

Soft Voting averages probabilities from Logit and decision values from Linear SVM (converted probabilites using a sigmoid function)

Combined Probability = (Logit Probability + SVM Probability) / 2

The sigmoid function for SVM decision values:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Final prediction threshold:

$$\text{Ensemble Prediction} = \begin{cases} 1, & \text{if Combined Probability} >= 0.5 \\ 0, & \text{otherwise} \end{cases}$$

Method 4: Ensemble Models (Results)

Performance Metrics for Hard Voting:

Accuracy: 95.02%; Precision: 94.81%; Recall: 83.68%; F1-Score: 88.58%

Performance Metrics for Soft Voting:

Accuracy: 94.81%; Precision: 89.42%; Recall: 87.91%; F1-Score: 88.66%

Method 5: Artificial Neural Network

Why?

ANN was chosen for it's ability to handle non-linear relationships and high-dimensional data. We wanted to use it as a standard to compare our previous models.

Mathematical Foundation:

Model Architecture: We used a feedforward neural network with 3 fully connected layers:

Layer 1: RELU, **Layer 2:** RELU, **Layer 3:** Sigmoid Function

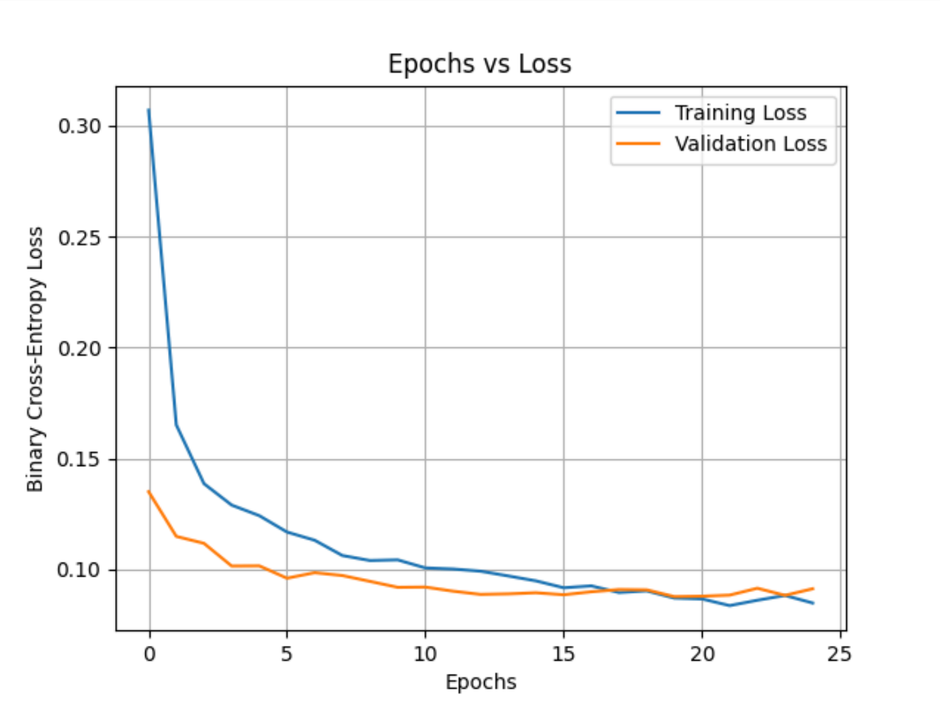
We use **Binary Cross-Entropy (BCE)** to penalize incorrect predictions and **Adam optimizer** for adapting learning rates and faster convergence. Additionally, we also use regularization techniques like **Dropout** and **Early Stopping.**

Method 5: ANN (Results)

Performance Metrics:

- **Training Set**: Accuracy: 97.75%; Precision: 94.99%; Recall: 95.50%; F1-Score: 95.25%
- **Validation Set**: Accuracy: 96.39%; Precision: 92.76%; Recall: 92.54%; F1-Score: 92.65%

Log-Loss Curve:



Top 5 Relevant Features:

- **Monthly Income**: 24.9299
- **BankruptcyHistory:** 24.8756
- **InterestRate:** 19.5580
- **TotalDebtToIncomeRatio:** 17.3426
- **PreviousLoanDefaults:** 15.7918

Final Results

Model	Accuracy	Precision	Recall	F1-Score
Training Metrics				
Logistic Regression	0.9457	0.8969	0.8775	0.8871
Linear SVM	0.9549	0.8958	0.9194	0.9075
Polynomial SVM (Degree 2)	0.9268	0.9463	0.7378	0.8292
Polynomial SVM (Degree 3)	0.9608	0.9580	0.8757	0.9150
ANN	0.9775	0.9499	0.9550	0.9525
Validation Metrics				
Logistic Regression	0.9472	0.8901	0.8797	0.8849
Linear SVM	0.9527	0.8915	0.9099	0.9006
Polynomial SVM (Degree 2)	0.9208	0.9249	0.7218	0.8108
Polynomial SVM (Degree 3)	0.9095	0.8400	0.7605	0.7982
ANN	0.9639	0.9276	0.9254	0.9265
Ensemble (Hard Voting)	0.9502	0.9408	0.8368	0.8858
Ensemble (Soft Voting)	0.9481	0.8942	0.8791	0.8866

1. ANN model did the best with the highest metrics. (As expected)’
2. Logit Regression and Linear SVM followed ANN as the best hard-coded models.
3. Polynomial SVm showed signs of overfitting.
4. Ensemble Models: Provided a balanced trade-off between precision and recall but were less effective in predicting the minority class.

Feature Importance:

Consistently Important Features:

- MonthlyIncome: Strongly positive impact across models, indicating financial stability.
- BankruptcyHistory: Negative correlation, emphasizing lenders' risk aversion.
- InterestRate and DebtToIncomeRatio: Key factors affecting eligibility and risk perception.

Future Works and Improvement

Exploring More Models:

- Trying advanced models like decision trees, KNN, Gradient Boosting, etc.

Dynamic Feature Selection:

- Explore how additional domain-specific features (e.g., regional economic indicators) can enhance predictions.

Addressing Class Imbalance:

- Use techniques like SMOTE (Synthetic Minority Oversampling Technique) or focal loss to better handle class imbalance and improve minority class detection.

Improving Interpretability:

- Evaluate the impact of each feature group (demographic, financial, loan-related) seperately.

References

- Kumar, S., & Singh, R. (2023). An ensemble machine learning-based bank loan approval predictions system with a smart application. Journal of Financial Technology and Innovation, 5(2), 123–135.
- Mahgoub, A. (2024). Optimizing Bank Loan Approval with Binary Classification Method and Deep Learning Model. Open Journal of Business and Management, 12(3), 1970–2001.
- Yildiz, B. (2022). Predicting acceptance of the bank loan offers by using support vector machines. International Advanced Researches and Engineering Journal, 6(2), 142–147. doi:10.35860/iarej.1058724.
- Alhamid, M. (2020). How to Attain a Deep Understanding of Soft and Hard Voting in Ensemble Machine Learning Methods. Towards Data Science.
- Peng, S., Hu, Q., Dang, J., & Peng, Z. (2017). Stochastic Sequential Minimal Optimization for Large-Scale Linear SVM. In Neural Information Processing. ICONIP 2017 (pp. 344–354). Springer, Cham. doi:10.1007/978-3-319-70087-8_30.
- Shalev-Shwartz, S., Singer, Y., Srebro, N., & Cotter, A. (2011). Pegasos: Primal Estimated sub-GrAdient SOlver for SVM. Mathematical Programming, 127(1), 3–30.
- Zoppelletto, L. (2023). Financial Risk for Loan Approval. Kaggle. [Dataset]. Retrieved from https://www.kaggle.com/datasets/lorenzozoppelletto/financial-risk-for-loan-approval/data.

Thank you