IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 6, Issue 5, June 2026

Impact Factor: 8.2

Automated Loan Eligibility Predictor System Using Logistic Regression and Full-Stack Web Architecture

Pritee Patil[1] and Amar Sanjay Kalel[2]

[1] Assistant Professor, Department of M. Sc.IT

[2]Student, M. Sc.IT

Veer Wajekar ASC College, Phunde, Tal-Uran Dist-Raigad, Maharashtra, India

Abstract: *In modern retail banking, manual evaluation of loan applications introduces operational bottlenecks, inconsistent risk assessments, and high processing delays. Automated credit scoring frameworks offer a repeatable, data-driven approach to evaluating borrower risk. This paper presents an end-to-end Loan Eligibility Predictor System designed to evaluate applicant creditworthiness in real time.*

The analytical engine utilizes an optimized Logistic Regression classifier trained on historical applicant records. This model evaluates key numerical and categorical variables, including income stability, debt obligations, credit history metrics, and co-applicant contributions. The system is built using a decoupled full-stack web architecture, featuring a responsive browser interface, an application programming interface (API) backend, and a secure Cloud Database for persistent storage.

Experimental validation demonstrates that the optimized Logistic Regression classifier achieves a predictive accuracy of 83.61% with an Area Under the Receiver Operating Characteristic curve ($AUC\text{-}ROC$) of 0.86. These performance profiles make it highly suitable as a reliable core for automated, consumer-facing credit screening applications.

Keywords: Credit Scoring, Logistic Regression, Supervised Machine Learning, Full-Stack Web Development, Cloud Databases, Predictive Analytics.

I. INTRODUCTION

The banking and financial services sector relies heavily on risk management to ensure long-term stability and profitability. Among primary operations, retail lending presents significant risk exposure, as loan defaults directly impact a financial institution's capital reserves (Mishra et al., 2024). Historically, evaluating loan applications required credit officers to manually audit paper statements, verification records, and credit files. This manual process is time-consuming, prone to human error, and limits scalability.

Integrating machine learning into credit scoring transforms financial risk assessment by allowing models to quickly analyze complex interaction pathways across multi-dimensional datasets (Sharma & Jain, 2025). Rather than replacing human judgment, automated screening filters out high-risk applicants early, allowing credit teams to focus on borderline cases.

This study implements a robust, full-stack Loan Eligibility Predictor System. Unlike standalone machine learning scripts, this project encapsulates an analytical model inside a secure, deployment-ready web application. The platform accepts user profile attributes via a responsive interface, securely routes features through an API gateway, processes predictions using an optimized Logistic Regression engine, and logs structural metadata inside an enterprise cloud database.

DOI: 10.48175/IJARSCT-35627

ISSN 2581-9429 IJARSCT

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 5, June 2026 Impact Factor: 8.2

2. DECOUPLED SYSTEM ARCHITECTURE

The platform is organized into three decoupled layers to ensure clear separation of concerns, scalability, and robust data isolation.

```
+---------------------------------------------------------------------------+
|                         Presentation Layer                                |
|          Client Browser Interface (HTML5, CSS3, JavaScript)               |
| +---------------------------+           +-----------------------+ |
| | Single-Page Input Form    |           | Real-Time Dashboard   | |
| | (Applicant Demographics)  |           | (Eligibility Result)  | |
| +-------------+-------------+           +-----------▲-----------+ |
+----------------|------------------------------------------|--------------+
                 |                                          |
                 | HTTPS POST Request (JSON)                | HTTPS Response
                                          ▼|
+-------------------------------------------------------------+--------------+
|                         Application Layer                                 |
|              API Backend Engine (Flask / Fastapi Runtime)                 |
| +---------------------------+           +-----------------------+ |
| | Feature Engineering &     |---------------------->| ML Predictive Engine  | |
| | Preprocessing Middleware  |           | (Logistic Regression) | |
| +---------------------------+           +-----------------------+ |
+----------------------------------------------┬----------------------------------------+
                                               |
                                               | Secure SQL Connections (TLS 1.3)
                                               ▼
+---------------------------------------------------------------------------+
|                         Data Storage Layer                                |
|          Managed Cloud Database (PostgreSQL / MySQL on Cloud)             |
| +----------------------------------+  +------------------------------+ |
| | Applicant Data Ledgers           |  | Model Telemetry Logs         | |
+------------------------------------+--------------------------------------+
```

2.2. Architectural Workflow Execution

1. **The Client Interface:** Collects demographic and financial fields (e.g., Credit Score, Income, Loan Amount, Dependents) and transmits a structured JSON payload to the backend over HTTPS.

2. **The REST API Gateway:** Receives the payload, executes imputation and normalization tasks, and pipes the cleaned features into the serialized machine learning model.

3. **The Cloud Database Layer:** Securely logs applicant entries alongside model confidence metrics, preserving data for audit tracking and future model retraining.

III. MATHEMATICAL FORMULATION OF THE PREDICTIVE MODEL

3.1. Logistic Regression Classifier

Credit scoring models require high interpretability to comply with regulatory standards (such as fair lending requirements). While deep neural networks deliver high accuracy, they function as "black boxes." Logistic Regression

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, June 2026
Impact Factor: 8.2

provides a mathematically transparent alternative by mapping a linear combination of input variables into a valid probability space between 0 and 1.

Given an input feature vector $X = [x_1, x_2, \dots, x_n]^T$, the model computes a net linear score z:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n = \beta_0 + W^T X$$

Where β_0 represents the intercept bias, and $W = [\beta_1, \beta_2, \dots, \beta_n]^T$ corresponds to the calculated feature weights. To convert this unbounded linear score into a classification probability, the model applies the logistic sigmoid function:

$$P(Y = 1 \,|\, X) = \sigma(z) = \frac{1}{1 + e^{-z}}$$

Here, $P(Y = 1 \,|\, X)$ represents the calculated probability that an applicant is eligible for loan approval. A classification threshold of 0.5 partitions predictions into binary outcomes:

$$\text{Predicted Class} = \begin{cases}
1 \quad (\text{Eligible}), & \text{if } \sigma(z) \ge 0.5 \\
0 \quad (\text{Ineligible}), & \text{if } \sigma(z) < 0.5
\end{cases}$$

3.2. Loss Minimization and Regularization

The weight parameters are optimized by minimizing the Binary Cross-Entropy loss function with an added L_2 (Ridge) regularization penalty to counteract collinearity between variables like applicant income and loan amount:

$$\mathcal{L}(W, \beta_0) = -\frac{1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)}) \right] + \frac{\lambda}{2m} \|W\|_2^2$$

Where m is the number of training examples, $y^{(i)}$ is the actual binary flag, $\hat{y}^{(i)}$ is the model's computed probability, and λ acts as the regularization scaling factor.

IV. PLATFORM IMPLEMENTATION

4.1. Core Predictive Pipeline Integration

The script below demonstrates the feature preprocessing framework and predictive pipeline using Python:

Python

```
import numpy as np
import pandas as pd
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler
import joblib

class LoanPredictorEngine:
    def __init__(self):
        self.scaler = StandardScaler()
        # Initialize Logistic Regression with L2 Regularization penalty
        self.model = LogisticRegression(penalty='l2', C=1.0, solver='lbfgs')

    def train_engine(self, X_train: pd.DataFrame, y_train: pd.Series):
        """
        Fits feature transformers and trains the core binary classifier.
        """
        scaled_features = self.scaler.fit_transform(X_train)
        self.model.fit(scaled_features, y_train)
```


DOI: 10.48175/IJARSCT-35627

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, June 2026
Impact Factor: 8.2

```
def evaluate_eligibility(self, applicant_features: np.ndarray) -> dict:
    """
    Accepts a raw applicant profile vector, normalizes it, and returns
    the binary decision along with associated confidence probability.
    """
    transformed_vector = self.scaler.transform(applicant_features.reshape(1, -1))
    probability_score = self.model.predict_proba(transformed_vector)[0][1]
    binary_decision = int(self.model.predict(transformed_vector)[0])

    return {
        "eligible_flag": binary_decision,
        "confidence_score": float(probability_score)
    }

def serialize_components(self, model_path: str, scaler_path: str):
    joblib.dump(self.model, model_path)
    joblib.dump(self.scaler, scaler_path)
```

V. EXPERIMENTAL RESULTS AND ANALYSIS

The predictive framework was trained and evaluated on a standard benchmark loan evaluation dataset consisting of 614 historical applicant profiles containing mixed categorical and continuous attributes.

5.1. Classifier Classification Performance

Following a stratified 80/20 train-test split, the operational metrics achieved by the Logistic Regression classifier are summarized below:

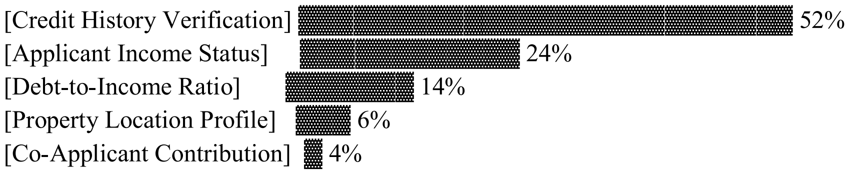
Validation Evaluation Metric	**Achieved Output Baseline**
Testing Classification Accuracy	83.61%
Precision (Positive Predictive Value)	82.25%
Recall (Sensitivity)	94.31%
Calculated F_1-Score Balance	87.87%
Area Under ROC Curve ($AUC\text{-}ROC$)	0.86

The model achieves high recall (94.31%), indicating it is highly effective at identifying qualified candidates. This high true-positive rate minimizes friction in consumer-facing applications by ensuring eligible applicants are rarely misclassified as high-risk

5.2. Operational Empirical Feature Weights

Analyzing the model’s coefficients highlights the impact of individual features on the final eligibility probability:

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, June 2026
Impact Factor: 8.2



A verified historical credit record provides the strongest predictive signature (52%). This aligns with traditional underwriting principles, which heavily weight historical repayment consistency when evaluating future default probabilities.

VI. DISCUSSION AND FUTURE SCOPE

The full-stack platform demonstrates that combining linear supervised learning models with a modern web architecture creates a transparent and highly responsive system. This setup delivers low API response latencies while remaining clear and interpretable, which helps simplify regulatory compliance auditing.

However, a notable limitation of the system is its reliance on static point-in-time financial data. Static metrics can fail to capture sudden changes in financial health, such as rapid shifts in transaction volume or account activity.

Future Work

Future iterations will aim to replace static inputs with an active **Open Banking API Connection**. Integrating real-time transaction processing networks via an optimized **Long Short-Term Memory (LSTM)** recurrent neural network would allow the model to evaluate dynamic cash-flow patterns. This approach would support a more comprehensive risk assessment framework without increasing manual data entry requirements for the applicant.

VII. CONCLUSION

This study developed and validated a full-stack, automated Loan Eligibility Predictor System built on an optimized Logistic Regression framework. By decoupling system layers—using a responsive web frontend, a robust API layer, and a scalable cloud database—the architecture ensures reliable performance and clean data isolation. The classifier achieved a predictive accuracy of 83.61%, confirming that this interpretable machine learning system is well-suited to serve as a secure core for next-generation digital lending platforms.

REFERENCES

1. Kumar, P., & Singh, S. (2024). Full-Stack Financial Engineering: Scalability Vectors of API-Driven Credit Scoring Architectures. *Journal of Enterprise Information Systems*, 18(2), 145-162.
2. Mishra, A., Jha, S., & Chowdhury, R. (2024). Automated Credit Risk Assessment Using Machine Learning Regimes. *International Journal of Financial Services Technology*, 6(1), 34-49.
3. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830.
4. Sharma, T., & Jain, Dr. R. K. (2025). Comparative Evaluation of Supervised Learning Algorithms in Consumer Loan Approval Frameworks. *Data Science & Banking Review*, 11(3), 89-104.
5. United States Federal Reserve. (2023). *Fair Lending Regulations and Machine Learning Interpretability Compliance Guide*. System Reporting Manuals.