AI-Powered Credit Risk Scoring & Loan Decision System

Using Databricks, Delta Lake & MLflow

Domain: Finance & Banking

By: Jayarani Arunachalam

Problem Statement

Business Problem

- Banks must approve loans while minimizing default risk
- Traditional rule-based systems are rigid and incomplete
- Risk arises from complex interactions between income, credit history, and loan characteristics

Objective & AI Framing

Project Objective

- Predict loan default probability
- Explain risk drivers
- Convert predictions into actionable loan decisions

AI Framing

- ML Task: Binary Classification (Yes or No)
- Target Variable: loan_status (Default or Not)

Dataset Overview

Dataset Summary

- 32,581 loan records
- Customer data such as age, emp length
- Credit history
- Loan attributes
- Loan outcome
- Loan Status (0 is non default 1 is default)

Key Challenges

- Missing values
- Outliers
- Imbalanced target variable

- `person_age`
- `person_income`
- `person_emp_length`
- `Loan  loan_amnt`
- `loan_int_rate`
- `loan_percent_income`
- `Credit cb_default_flag`
- `cb_person_cred_hist_length`
- `loan_intent`
- `loan_grade`
- `loan_status` (Target)

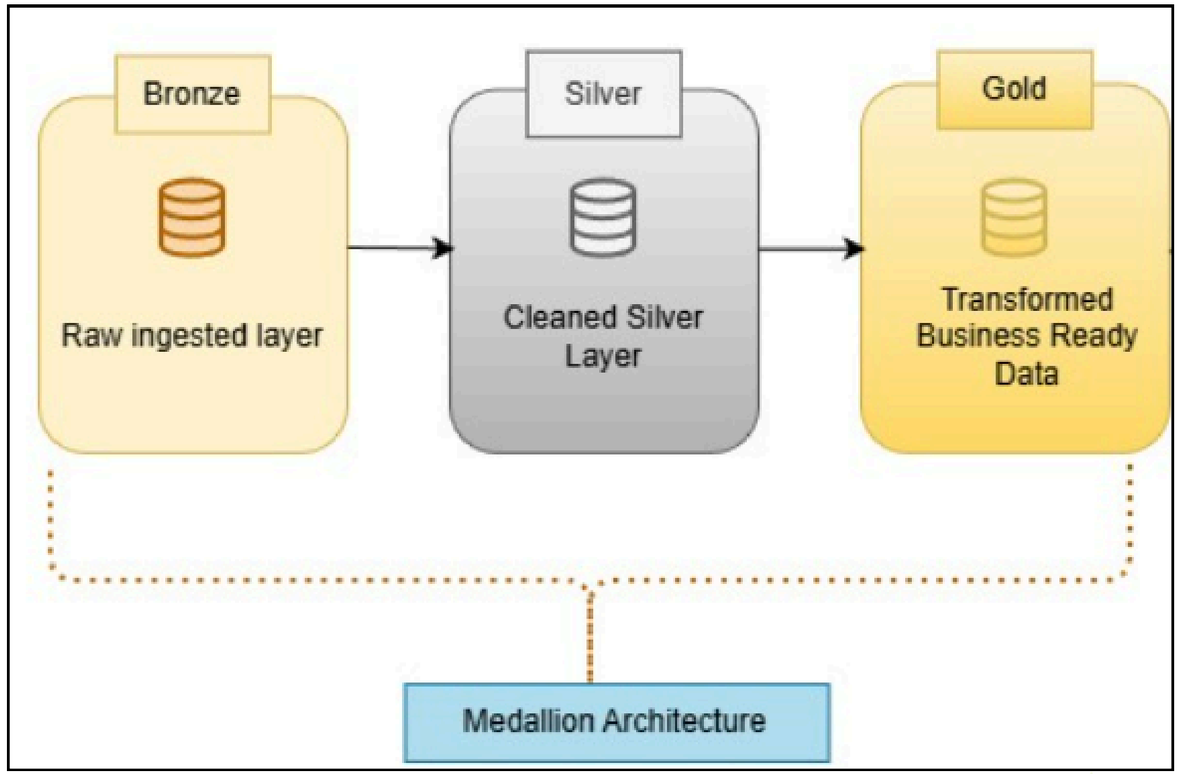
Data Architecture

Medallion Architecture

- Bronze: Raw loan data ingestion
- Silver: Cleaned & feature-engineered data
- Gold: Analytics-ready & ML-ready datasets

Technologies Used

- Delta Lake
- Unity Catalog
- Databricks Jobs
- MLflow



Data Cleaning & Preparation

Key Cleaning Rules

- Removed unrealistic ages (>75)
- Removed employment length outliers (>60 years)
- Imputed missing interest rates using loan grade & intent

Why Not Drop Rows?

- Prevents data loss
- Avoids bias
- Preserves business patterns

Feature Engineering

Engineered Feature

Prior Default High Risk Flag

- Combines:
  - Previous default history
  - High loan-to-income ratio

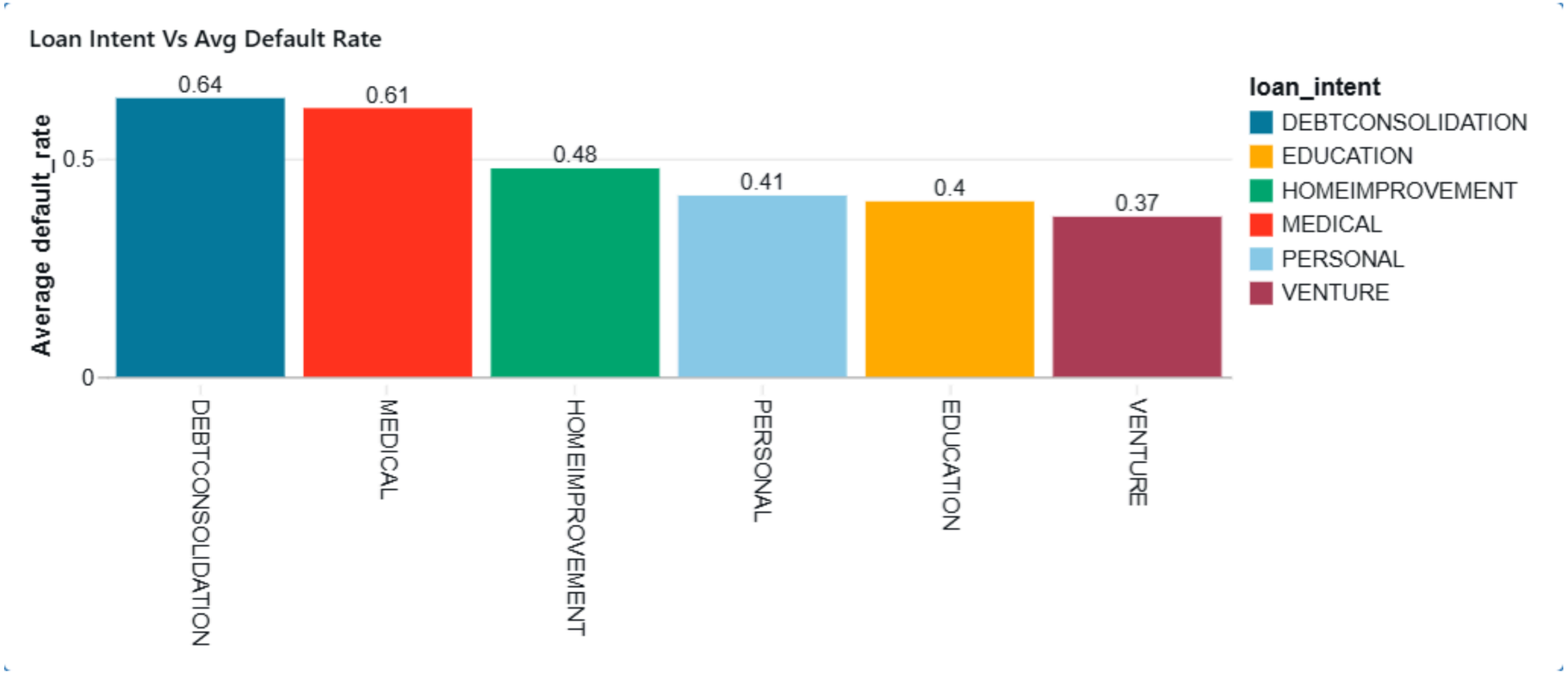
Why This Matters

- Compound risk is more predictive than single factors
- Aligns with real credit risk assessment

Business Insights

Key Observations

- Debt Consolidation & Medical loans show highest default rates
- Higher loan-to-income ratio significantly increases risk
- Credit history length strongly influences repayment behavior



Model Selection- Training & Evaluation

Model Chosen: Logistic Regression

Why Logistic Regression

- Industry standard for credit scoring
- Highly interpretable
- Stable baseline
- Easy to explain to business stakeholders

Training Setup

- 80 / 20 Train-Test split
- Features sourced directly from Gold Delta tables
- Experiments tracked using MLflow

Metrics Used

- AUC
- Accuracy
- Precision
- Recall

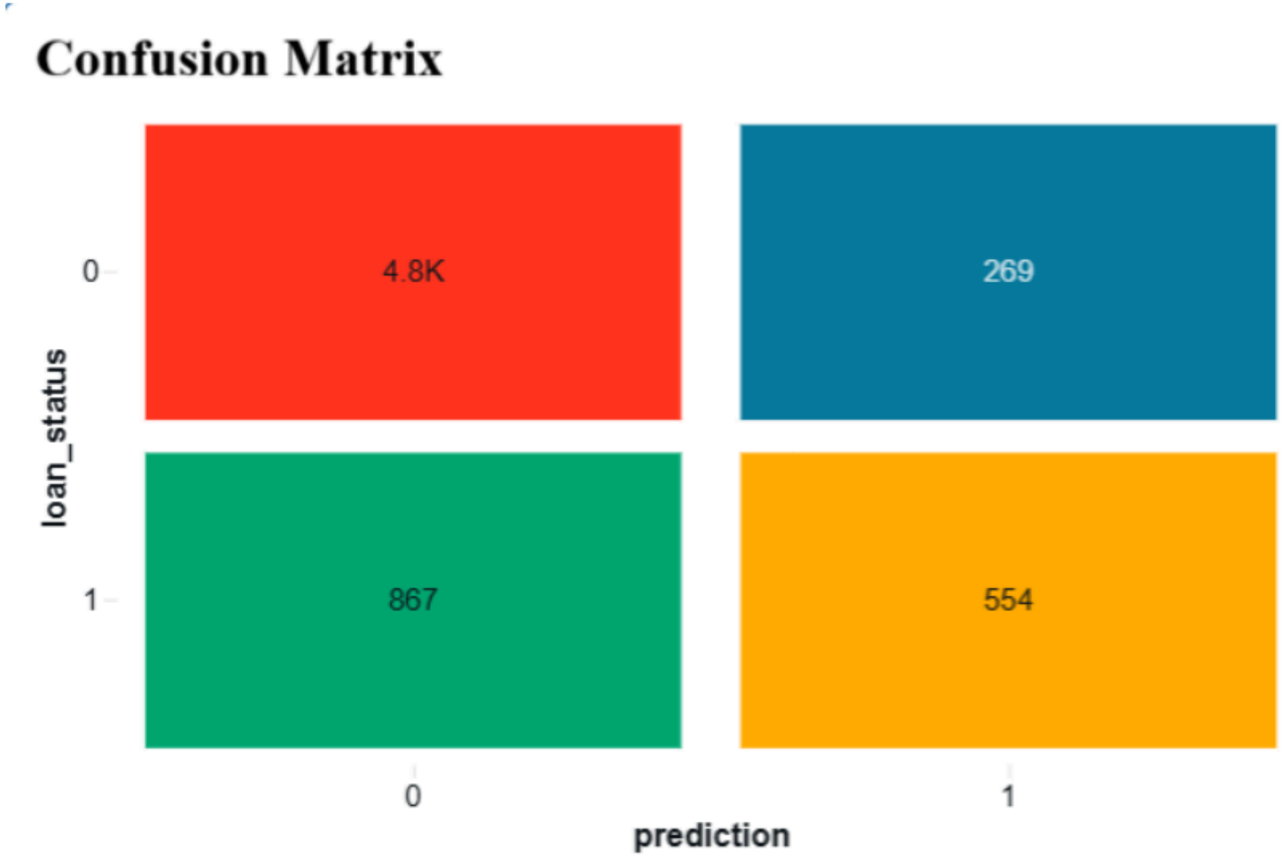
```
AUC: 0.8241664940205172
Accuracy: 0.824934504546155
Precision: 0.6731470230862697
Recall: 0.3898662913441238 7
```

Model Evaluation

Confusion Matrix Insights

- High correct approvals
- Some non-risky customers are not approved (Still no Financial Loss)

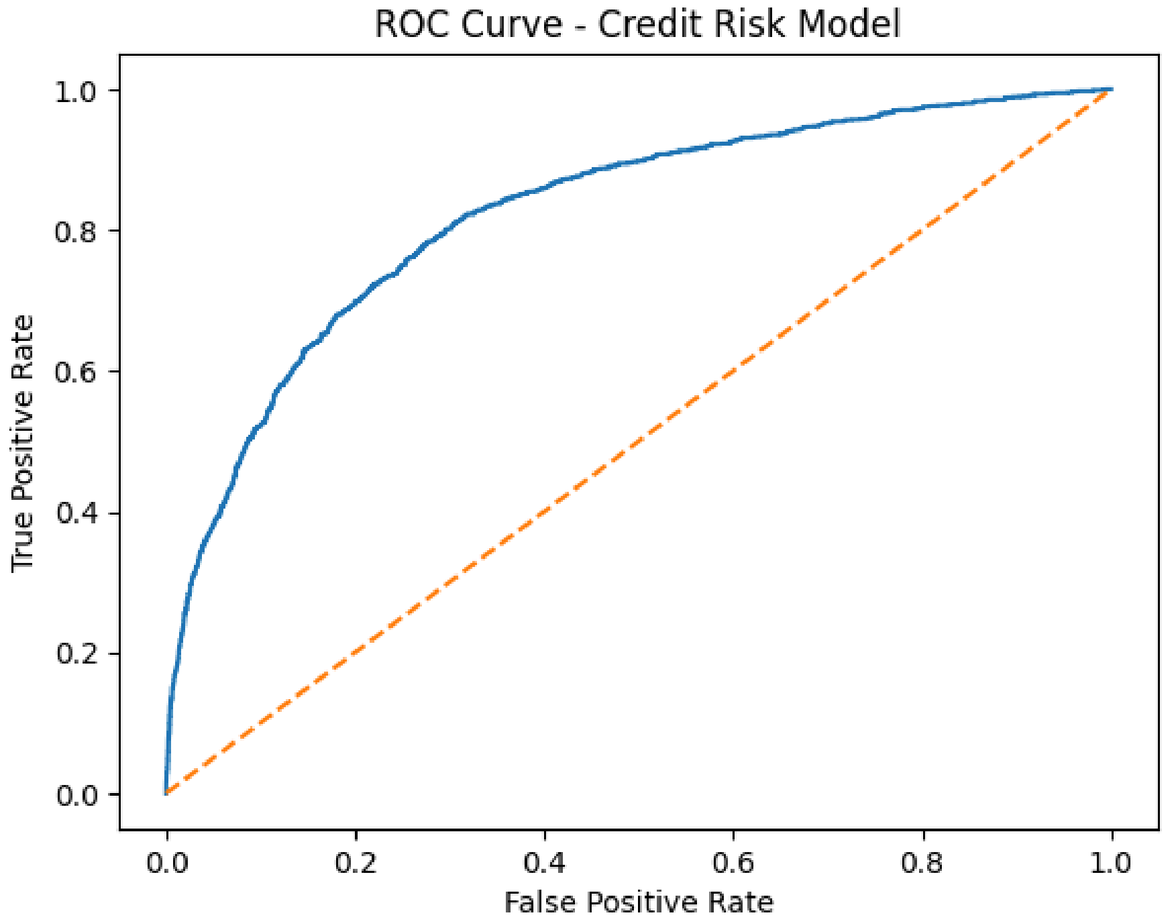
Because here 0 means default , 1 means not a default (0 - Approve, 1 - Reject)



Model Evaluation - contd

ROC Curve

- Demonstrates strong separation between default and non-default cases



Risk Scoring and Decision Logic

Decision Framework

Default Probability	Decision
< 0.30	APPROVE
0.30 – 0.60	REVIEW
≥ 0.60	REJECT

```
from pyspark.sql.functions import when
pdf_1 = (
    predictions
    .withColumn("prob_array", vector_to_array(col("probability")))
    .withColumn("default_prob", col("prob_array")[1]))
scored_df = pdf_1.withColumn(
    "decision",
    when(col("default_prob") < 0.3, "APPROVE")
    .when(col("default_prob") < 0.6, "REVIEW")
    .otherwise("REJECT")
)
```

End-to-End AI Workflow

Pipeline Flow

- Delta Tables → Feature Extraction
- ML Model → Risk Scoring
- Decisions written back to Delta tables

Why This Matters

- Full database ↔ AI integration
- Scalable & production-ready