THEME: Generative and Agentic AI and ML

Hack o Hire

TEAM NAME : BrainMesh

MEMBERS:
Saket Tepale (Team Leader)
Roshni Rajput
Rohan Sakhare
Rudraksh Samundre

Abstract:

Traditional credit scoring systems depend heavily on historical banking data. This excludes millions of people who do not have formal credit histories. This project suggests an AI-powered inclusive credit scoring system that uses alternative financial data, such as transaction behavior, payment consistency, mobile usage patterns, and digital activity. This approach can assess creditworthiness more accurately and fairly.

The system uses machine learning algorithms to handle data preprocessing, feature engineering, model training, and bias evaluation. It generates a dynamic and explainable credit score. The solution seeks to reduce financial exclusion, improve loan approval efficiency, and lower default risk for financial institutions.

By including fairness checks and omitting sensitive factors like gender, religion, and ethnicity, the model ensures ethical and unbiased decision-making. The proposed system is scalable, cost-effective, and flexible for banks, fintech companies, and microfinance institutions. It enables responsible lending while supporting financial inclusion and economic growth.

PROBLEM STATEMENT:

AI-Powered Alternate Credit Scoring System

- **Challenge of Financial Exclusion:**

Millions of individuals and MSMEs lack credit histories, preventing financial i nstitutions from assessing their creditworthiness and offering access to formal credit.

1.**Rigid Traditional Models:**
- Traditional credit scoring systems are backward-looking and over-rely on historical bureau records and past loan data, ignoring alternative financial behaviours and digital footprints.

2.**Under-banked & Unbanked Populations:**
- First-time borrowers and small businesses remain excluded despite having clear digital transaction, utility payment, and commercial activity.

3.**Bias & Fairness Concerns:**
- Legacy scoring systems can be biased and may not meet growing regulatory demands for transparency and explainability.

4.**Manual & Cost-Intensive Underwriting:**
- Manual credit assessment processes are costly, slow, and difficult to scale.

MARKET NEED

- **Market Demand and Limitations of Traditional Credit Scoring Systems**

• The global credit scoring market is rapidly expanding due to digital banking growth and financial inclusion initiatives. The alternative credit scoring market is expected to grow significantly as millions of individuals and MSMEs lack formal credit histories. Increasing smartphone usage, digital payments, and fintech adoption are accelerating this demand.

• Limitations of Traditional Systems:

Exclusion of Credit-Invisible Consumers: Traditional systems overlook persons without credit history and continues to perpetuate financial exclusion.

- Inflexible Rule-Based Systems: Pre-set scoring systems grow old and become ineffective.
- Usage of Alternative Data (Digital Payment Transactions, Utility Bills, Mobile Transactions): These are usually ignored.
- Bias and Fairness Concerns: Traditional systems, in their age, may perpetuate unsound credit decisions and inequity.
- Significant Default Risk: Insufficient data results in poor risk evaluation and increased economic losses.

PROPOSED SOLUTION

1 DATA COLLECTION & PROCESSING

kaggle

CSV

- Mobile Payments
- Utility Bills
- E-Commerce
- Wallet Data
- Metadata
- CSV / API Ingestion

seaborn

- pandas DataFrame
- Missing Value Imputation
- Outlier Detection (IQR)
- Normalization
- Feature Encoding
- SMOTE

scikit learn

NumPy

matplotlib

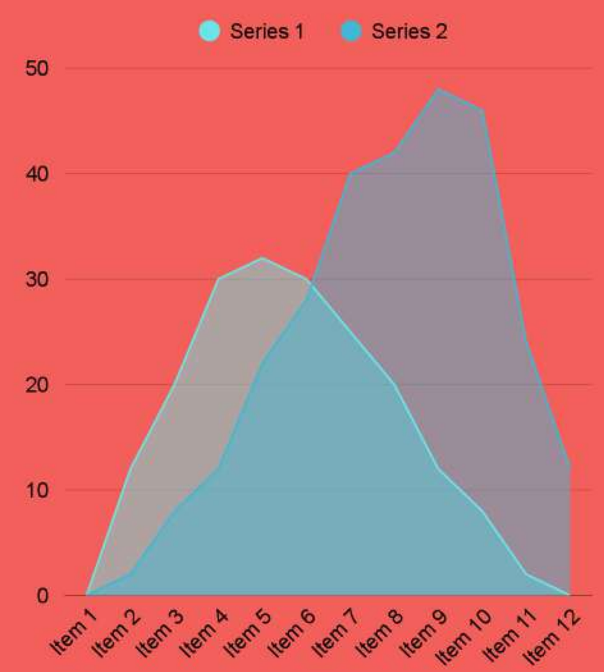
2 Feature Engineering Layer

- Payment Frequency
- Consistency Ratio
- Account Age
- Transaction Velocity
- Spending Stability
- Payment Streak
- Default History

3 Machine Learning Pipeline

- Train/Test Split (80-20)
- Logistic Regression (Baseline)
- Random Forest (Feature Importance)
- XGBoost (Production Model)
- Best Model (AUC > 0.80)
- Cross-Validation
- GridSearchCV

4 Evaluation, Fairness & Explainability Layer



- ROC-AUC, Precision, Recall, F1-Score, Confusion Matrix.
- Visualization of ROC Curves.
- Bias & Fairness Testing

SHAP Analysis

5 Deployment & Feedback Loop Layer

- REST API (Flask/FastAPI)
- Real-Time Scoring Engine
- Web Dashboard
- Model Monitoring
- Continuous Retraining

API

SYSTEM ARCHITECTURE

Data Ingestion LAYER 1

DATA ACCUMULATION
- Kaggle public datasets
- Transaction & credit behavior data
- MSME loan data

kaggle

DATA Loading
- Read csv files .
- Input From users.
- Give Me Some Credit (individual borrowers)
- German Credit Dataset

DATA Cleaning
- Handle Missing Values
- Remove Duplicates
- Fix Invalid & Impossible Values
- Outlier Treatment

Feature Engineering LAYER 2

Feature Discovery
- **Raw Data itself in not enough .**
- **You derive new features that describe risk behavior.**
- **for Eg-Credit Utilization Ratio,Payment Stability.**

Feature Validation
- Check missing value percentage
- Remove constant / low-variance features
- Validate logical bounds and skewed features. e.g. Utilization ∈ [0,1]
- EMI ⩽ Income

Feature Selection & Bias Screening
- Select predictive, stable, and fair features

Machine Training LAYER 3

Foundation
SMOTE

Train 80%

Test 20%

Models
Logistic regression= baseline model

Random Forest ="Which features actually drive risk?"

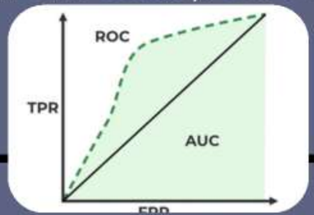
KNN=Sanity Check Model

XGBoost =High accuracy + controllable overfitting

scikit learn

Model evaluation LAYER 4

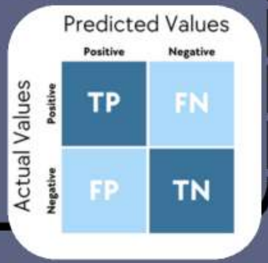
ROC-AUC
- Ability to distinguish from odds,>0.79.



SHAP
- Why did model give this prediction.

Shap

Explainability
- F1 Score
- Accuracy
- Confusion matrix
- Recall
- Precision



Feedback LAYER 5

Dashbaord
"We collect loan outcomes, repayment behavior, financial changes, and user feedback to continuously improve model accuracy and fairness."

Testing and Feedback
Taking user input and testing rest 20%

Future scope
- Addng features of AI recomendations.
- API calling.
- Govt.implementation

1.1)Data Collection Layer:

1.Data Collection

Mobile Payments
- Transaction frequency (daily/weekly/monthly)
- Average transaction amount & trends
- P2P transfer patterns & wallet balance
- Merchant categories (groceries, bills)
- Sources: UPI, Paytm, PhonePe, Google Pay

Bank-lite / Wallet Data
- Account balance patterns & savings
- Deposits, withdrawals & cash flow
- Recurring payment setup
- Digital wallet usage metrics
- Sources: Fintech apps, neobanks

Utility Bills
- Electricity, water, gas payment history
- On-time payment rate & consistency
- Service tenure & account age
- Bill amount stability over time
- Sources: BBPS, utility provider APIs

E-Commerce Activity
- Purchase frequency & order value
- Product categories & spending patterns
- Return/cancellation rate
- Seller ratings & account age
- Sources: Amazon, Flipkart, marketplaces

Behavioral Metadata
- App usage patterns (finance apps)
- Location stability (home/work)
- Device info (phone model, OS)
- Social media business activity
- Sources: Mobile SDK, app analytics (consent)

2.Implementation
- REST APIs + CSV ingestion (pandas DataFrames)
- 150,000+ records (Give Me Some Credit dataset)
- AWS S3 scalable storage
- GDPR & DPDP Act 2023 compliant
- 100% consent-based & anonymized data

3.Why Alternative Data?
Traditional credit excludes 1.4 billion adults globally
No credit cards or formal banking history required
Uses real-life financial behavior instead of rigid rules
Expands access to MSMEs & gig workers

4.Output of This Layer
Secure, anonymized Raw Alternative Data Repository
→ Sent to Preprocessing Layer for cleaning & normalization

Key Features of Our Data Collection

Privacy First
GDPR/DPDP compliant with explicit user consent

Verified Data
Real transactional data from trusted sources

No Credit History
Serves 1.4B unbanked individuals globally

Real-Time API
Live data ingestion & batch processing

Multi-Source
Holistic view from 5+ alternative sources

High Quality
Timestamped, traceable, non-self-reported

Scalable
Cloud-based infrastructure (AWS S3)

Inclusive
MSMEs & credit-invisible populations

1.2)Data preprocessing layer:

1 Missing Value Imputation

- Identify missing data patterns (MCAR, MAR, MNAR)
- Median imputation for numerical features (age, income)
- Mode imputation for categorical features
- MICE (Multiple Imputation) for complex dependencies
- Flag imputed values for model awareness

```
df.fillna(df.median(), inplace=True)
df['age'].fillna(df['age'].median())
```

2 Outlier Detection & Treatment

- IQR (Interquartile Range) method for detection
- Cap outliers at 1.5×IQR boundaries (not remove)
- Z-score method for extreme values (>3 std dev)
- Domain-specific rules (age: 18-100, income: >0)
- Preserve extreme values that are valid

```
Q1, Q3 = df.quantile([0.25, 0.75])
IQR = Q3 - Q1
df = df.clip(Q1-1.5*IQR, Q3+1.5*IQR)
```

3 Normalization & Scaling

- Min-Max Scaling to [0,1] range for tree models
- Standardization (Z-score) for distance-based models
- Robust scaling for datasets with outliers
- Log transformation for skewed distributions
- Fit scaler on training set only (prevent data leakage)

```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
X_scaled = scaler.fit_transform(X_train)
```

4 Feature Encoding

- One-Hot Encoding for nominal categories (region, payment method)
- Label Encoding for ordinal features (risk level: low/med/high)
- Target Encoding for high-cardinality features
- Binary encoding for yes/no variables
- Handle unseen categories in test set

```
pd.get_dummies(df, columns=['region'])
df['risk'] = df['risk'].map({'low':0, 'med':1, 'high':2})
```

5 Class Imbalance (SMOTE)

- Address imbalance: Only 6-7% defaults in dataset
- SMOTE: Synthetic Minority Over-sampling Technique
- Generate synthetic samples for minority class
- Balance ratio: 1:2 or 1:3 (minority:majority)
- Apply SMOTE only on training set (avoid data leakage)

```
from imblearn.over_sampling import SMOTE
smote = SMOTE(sampling_strategy=0.5)
X_bal, y_bal = smote.fit_resample(X, y)
```

Missing Value Imputation
Mean/Median/Mode
-KNN / Regression

Outlier Detection & Treatment
Detect: Z-Score / IQR
Remove extreme values
Cap (Winsorization)

Scaling /Normalization
Min-Max (0–1 range)
Standardization (Z-score)
Robust Scaling (Median)

Class Imbalance Handling
Detect skewed classes
Oversampling / Undersample
SMOTE .

2)Feature Engineering Layer:

Features:

• **Payment Frequency** – Measures average monthly transactions to assess financial activity level.
• **Consistency Ratio** – Percentage of on-time payments, indicating repayment discipline.
• **Recency Score** – Days since last transaction, reflecting engagement level.
• **Average Monthly Transactions** – Normalized activity measure across time.

These features convert everyday financial behavior into quantifiable risk signals.

Domain-Based Risk Insights:

• High consistency (> 90%) → Lower default risk
• Higher transaction frequency → Active financial participation
• Lower recency (recent activity) → Higher engagement and reliability

💰 Financial Features

Transaction Velocity

`= (current_balance - previous_balance) / time_period`

Rate of financial activity change over time

Savings Rate

`= deposits / (deposits + withdrawals)`

Proportion of money saved vs spent

Average Transaction Amount

`= sum_transaction_amounts / transaction_count`

Typical transaction size indicates income level

📊 Behavioral Features

Consistency Ratio

`= on_time_payments / total_payments`

Percentage of bills paid on time (strong credit indicator)

Spending Stability Index

`= 1 - (std_dev_spending / mean_spending)`

Consistent spending indicates income stability

Utility Payment Streak

`= consecutive_on_time_utility_payments`

Unbroken streak of timely utility bill payments

🛒 E-Commerce Features

Purchase-to-Return Ratio

`= returns_count / purchases_count`

Low return rate suggests decision-making quality

E-Commerce Account Tenure

`= days_since_first_ecommerce_purchase`

Established e-commerce presence indicates stability

Average Order Value

`= total_spending / order_count`

Spending capacity on discretionary purchases

⏰ Temporal Features

Payment Frequency

`= transactions_count / months_active`

How often user makes payments (daily, weekly, monthly patterns)

Account Age (Days)

`= current_date - first_transaction_date`

Longer account history indicates stability and trust

Recency Score

`= days_since_last_transaction`

Recent activity shows active financial engagement

⚠️ Risk Indicators

Default History Binary

`= 1 if past_defaults > 0 else 0`

Has user ever defaulted on payments before

Late Payment Ratio

`= late_payments / total_payments`

Frequency of payments made after due date

Overdraft Frequency

`= overdraft_count / months_active`

How often account balance goes negative

Real-World Feature Engineering Examples

💳 Payment Consistency

Raw Data: User paid 47 out of 50 bills on time

`Consistency Ratio = 47 / 50 = 0.94 (94%)`

Insight: User is highly reliable. 94% on-time rate indicates low default risk. This single feature is one of the strongest predictors of creditworthiness.

📈 Transaction Velocity

Raw Data: Balance grew from ₹5,000 to ₹12,000 in 3 months

`Velocity = (12000 - 5000) / 3 = ₹2,333/month`

Insight: Positive velocity shows income growth and savings. Increasing balance over time is a strong positive signal for credit approval.

⏰ Account Age

Raw Data: First transaction on Jan 1, 2022. Today is Feb 16, 2026

`Account Age = 1507 days ≈ 4.1 years`

Insight: Longer account history (4+ years) demonstrates established financial behavior. New accounts (<90 days) are higher risk.

🛒 E-Commerce Behavior

Raw Data: 85 purchases, 7 returns in past year

`Return Ratio = 7 / 85 = 0.08 (8%)`

Insight: Low return rate (8%) suggests thoughtful purchasing decisions. High return rates (>30%) may indicate financial instability or impulse buying.

3.1)Evaluation, Fairness & Explainability Layer:

Classification Metrics

AUC-ROC Score

Area Under the Receiver Operating Characteristic Curve - measures model's ability to distinguish between fraud and legitimate transactions

Precision & Recall

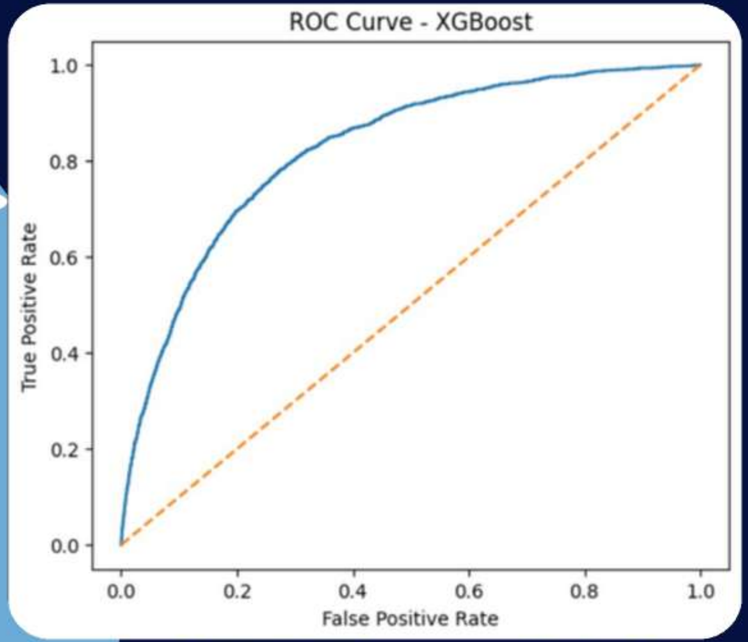
Precision: Accuracy of fraud predictions | Recall: Ability to catch all fraud cases

F1-Score

Harmonic mean of precision and recall - balances false positives and false negatives

Confusion Matrix

TP, TN, FP, FN breakdown for detailed performance analysis



AUC="How well do I rank everyone?"

Confusior Matrix

Predicted	good	default
Approve	TN	**FN**,lose money
Reject	**FP**,lost customer	TP

Accuracy = (TP + TN) /(TP+TN+FP+FN) = Total Predictions

Precision = TP / (TP + FP) ="Am I right when I reject?"

Recall = TP / (TP + FN) ="Did I catch all defaults?"

CREDIT SCORING:

├── High Precision → Don't reject good customers
├── High Recall → Don't approve defaulters
├── High F1 → Good balance
└── High AUC → Best overall ranking

F1 Score = 2 × (Precision × Recall) / (Precision + Recall) "Balanced score"

Evaluation Workflow & Reporting

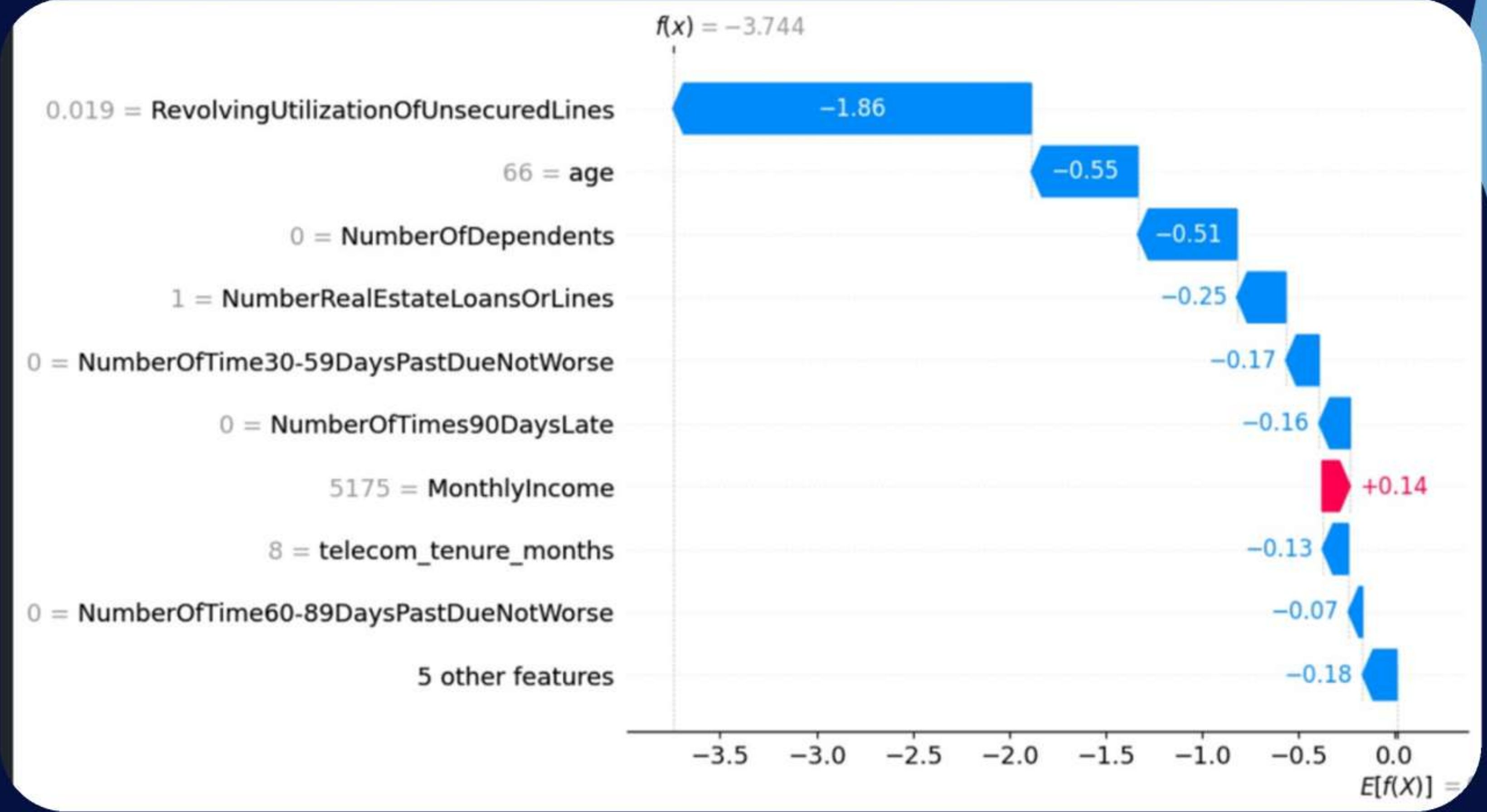
1.**Model Training & Initial Evaluation** = Train Test split (80:20)

2.**Fairness Audit**= Check whether predictions are biased by age, gender,income etc.

3.**Explainability Analysis**=Use SHAP / LIME to explain why the model made a decision.

4.**Final Test Set Evaluation**=Test on unseen data, compare with baseline models.

5.**Stakeholder Reporting=**Convert technical results into dashboards & recommend.



3.2)Evaluation, Fairness & Explainability Layer:

Fairness & Bias Assessment

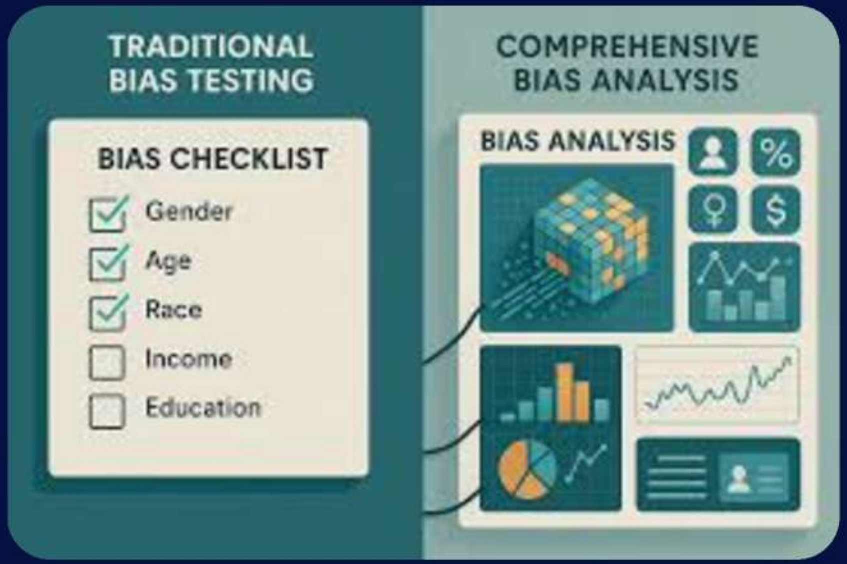
Why Fairness Matters in Fraud Detection
Ensuring the model doesn't discriminate against protected groups while maintaining effective fraud detection across all demographics.

Fairness Metrics

Demographic Parity
Equal fraud prediction rates across demographic groups

Equal Opportunity
Equal true positive rates for all groups

Equalized Odds
Equal TPR and FPR across protected attributes



Bias: Systematic favoring or disadvantaging certain individuals or groups
Group Fairness Metrics:

- Demographic Parity → Approval rates similar across groups
- Equal Opportunity → True positive rates similar for all groups
- Disparate Impact Ratio → Ratio of favorable outcomes (<0.8 indicates bias)

Shapley Values = contribution of each feature to the predicted outcome.

Model Explainability

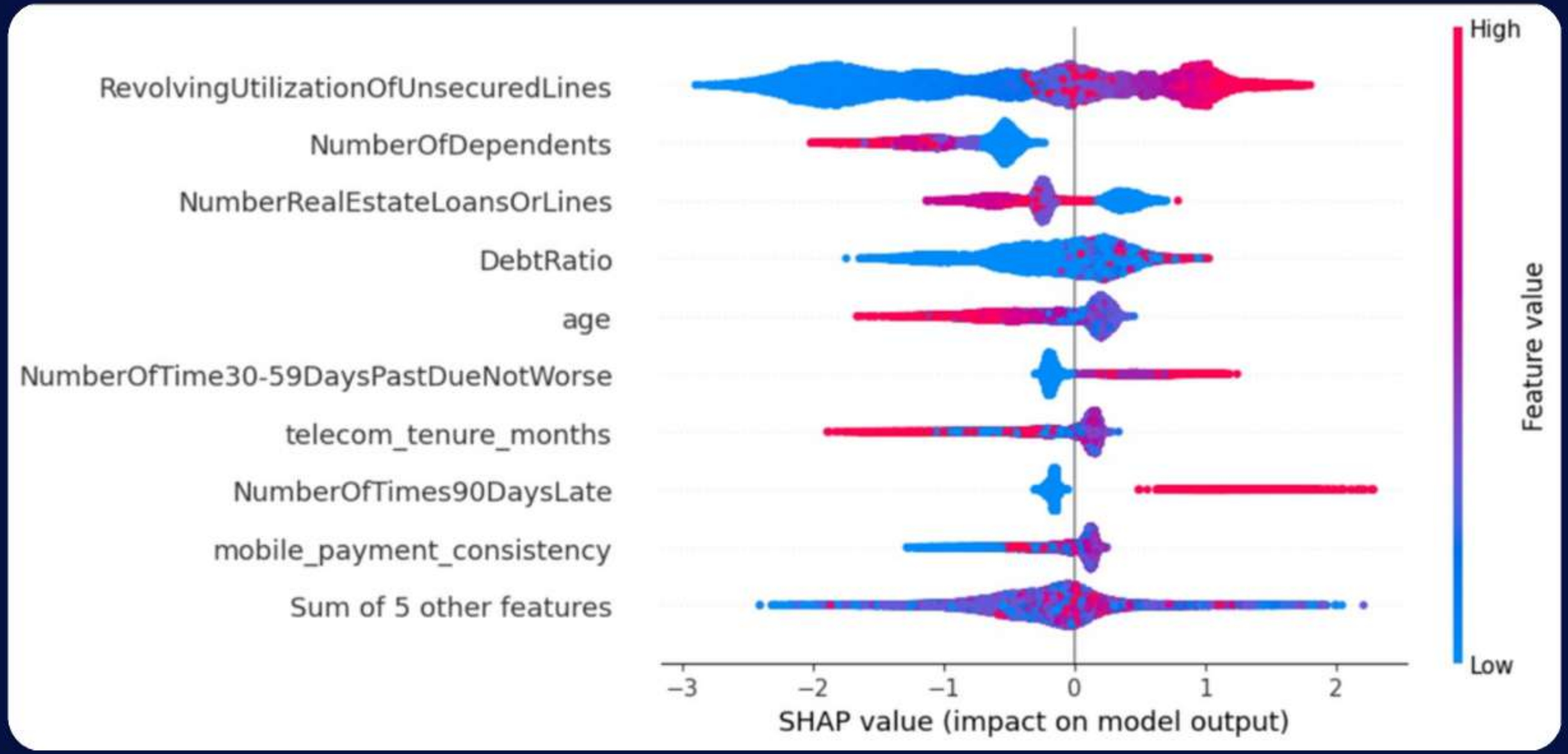
SHAP (SHapley Additive exPlanations)

Global Feature Importance
Understanding which features drive fraud detection across all predictions

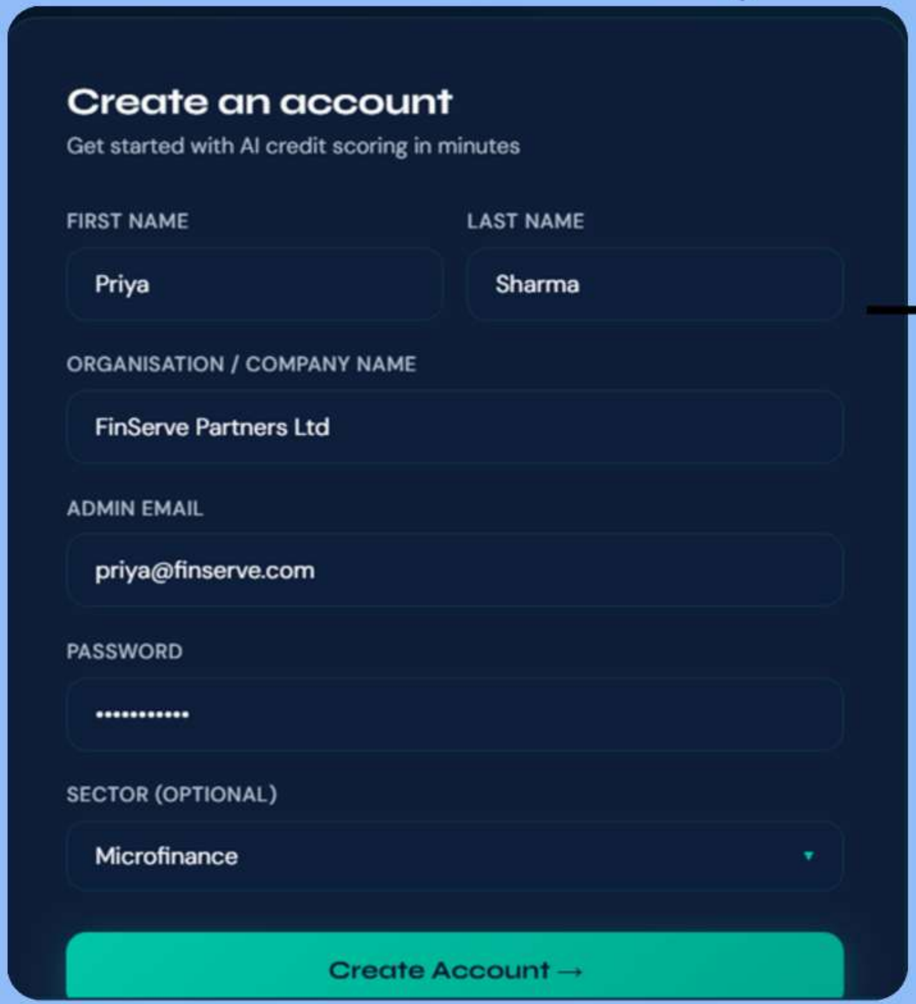
SHAP Summary Plots
Visualize feature impact on fraud predictions across entire dataset

SHAP Dependence Plots
Show relationship between feature values and prediction impact

SHAP Force Plots
Individual transaction explanation showing push toward/away from fraud

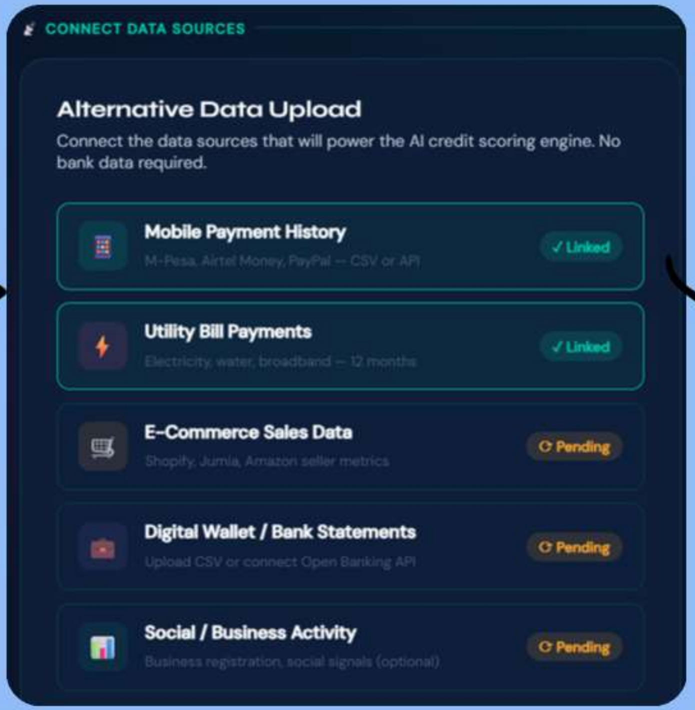


4)Prototype Workflow Overview

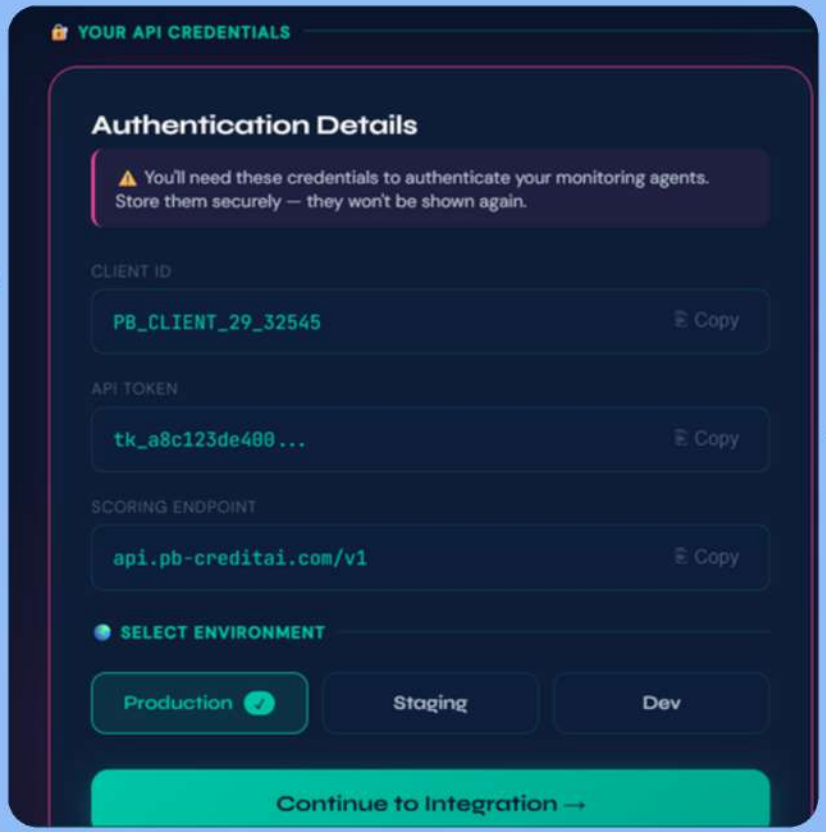


User Action:

- Enter Name,Email, Password,Industry
- Click Create account

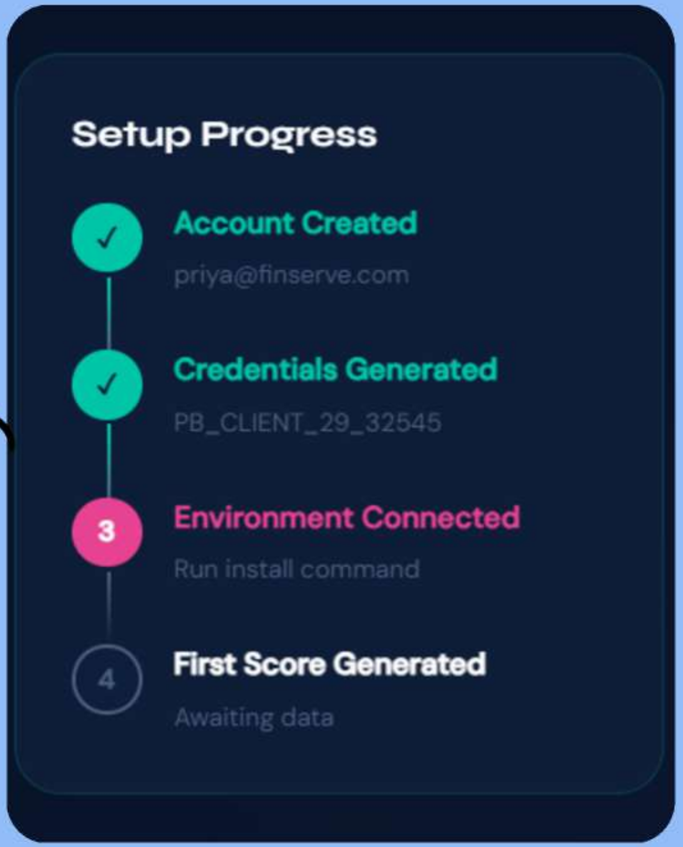


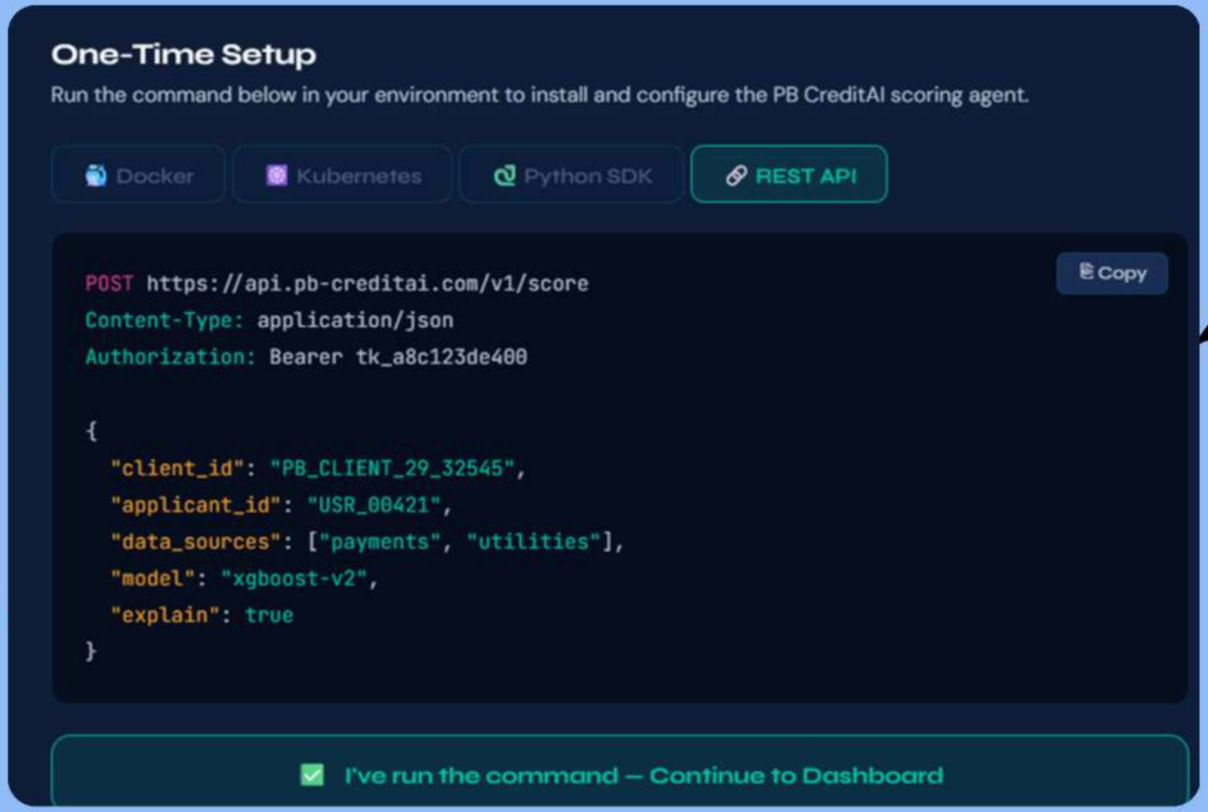
Other details of user



Your keys.
Your data.
Your scores.
Fully encrypted.

Track account setup progress





One-time environment setup command

5)DASH BOARD

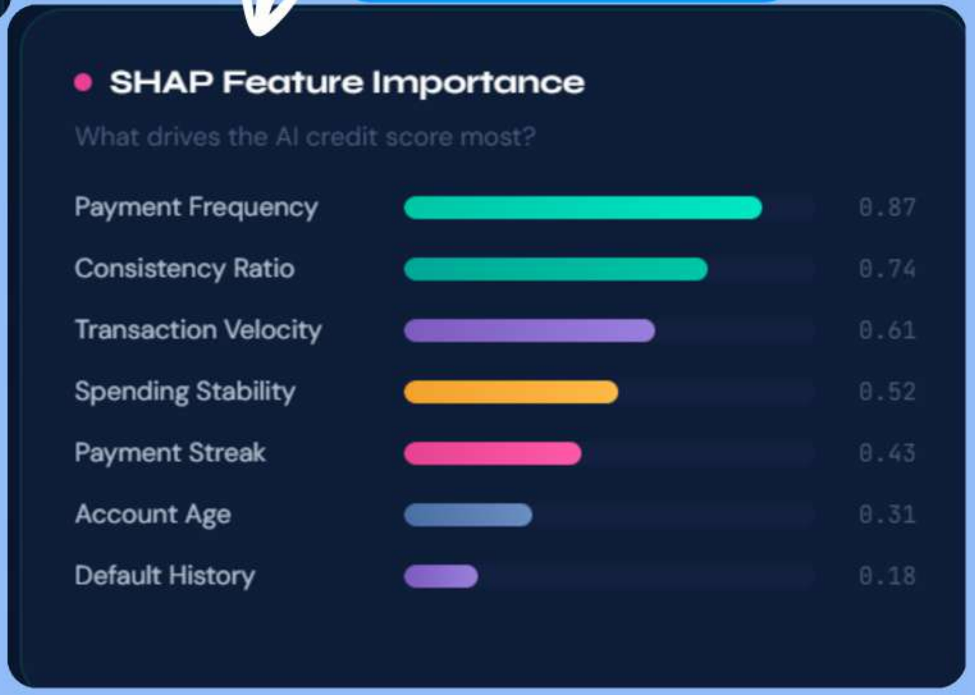
- Real-Time Credit Insights
- Applicants Score
- Approval rate%
- Avg Score

Credit Segementaion

Alternative Inputs

Scoring Performance

SHAP Insights



6)FUTURE SCOPE

1.Advanced AI for Dynamic Credit Scoring

1. Use Reinforcement Learning to update credit scores in real time based on repayment behavior.
2. Apply Graph Neural Networks (GNNs) to uncover hidden financial relationships (e.g., MSME supplier networks).
3. Enhance Explainable AI (XAI) to provide clear and transparent loan approval reasons.

2.Multi-Source & Real-Time Data Integration

1. Enable real-time API-based scoring for instant loan approvals.
2. Expand coverage to rural fintech ecosystems.
3. Integrate UPI transaction history, GST filings, e-commerce sales data, and utility bill payments.

3.Fairness & Bias Monitoring Dashboard

1. The system automatically checks if any group is treated unfairly using standard fairness rules.
2. It compares model accuracy across different population groups
3. It generates formal fairness reports that can be shared with regulators like RBI.

4.Scalable Cloud Deployment

1. The AI model is hosted on secure cloud infrastructure so it can scale reliably.
2. Banks and NBFCs can call the model using APIs instead of rebuilding it.
3. The system can adapt to different country regulations without changing core logic.

5.Financial Inclusion Expansion

1. Help people with no credit history get their first small loan.
2. Support MSMEs by accurately assessing short-term working capital risk.
3. Plug the system into government-backed financial inclusion programs.

7)References:

1.Kaggle, “Credit Risk Modeling & Loan Default Prediction Datasets,” Accessed 2025.
Give Me Some Credit,German Credit ,Home Credit Default Risk

2.Anaconda – Python Development Environment

3.Scikit-learn Developers, Scikit-learn: Machine Learning in Python
https://scikit-learn.org/

4.SHAP Documentation, Shapley Additive explanations for Model Interpretability.

- Random Forest

Breiman, L., Random Forests, Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.

- XGBoost

Chen, T. & Guestrin, C., XGBoost: A Scalable Tree Boosting System, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, pp. 785–794, 2016

- Linear Regression

Seber, G.A.F. & Lee, A.J., Linear Regression Analysis, Wiley Series in Probability and Statistics, 2nd ed., 2003.

5.TensorFlow Official Documentation

6.Google Developers – AI & Explainable AI Resources