

# FairLoans

Auditing and Debiasing Loan Approval Models for  
Responsible AI

*AI Bias Bounty Hackathon 2025 — HackTheFest*

---

**Team Name:** FairLoans

**Team Member:** Rajat Shinde

**Submission Date:** July 3, 2025

## Executive Summary

This report presents our complete pipeline for auditing and mitigating bias in automated loan approval systems using modern machine learning and fairness-aware techniques.

We developed and evaluated predictive models on a synthetic dataset that replicates real-world bias patterns. Our objective extended beyond high accuracy—we aimed to ensure demographic fairness across sensitive attributes such as gender and race.

### Key Results

*Table 1 \*\*This table compares the performance and fairness of the baseline vs. debiased model. While the baseline model performed well in accuracy and AUC, it exhibited significant bias. The debiased model drastically improved fairness metrics.*

Metric	Baseline Model	Debiased Model
Accuracy	86%	58.05%
AUC (Area Under ROC Curve)	0.92	N/A
Demographic Parity Difference	0.23	0.01
Equal Opportunity Difference	0.18	0.03

### Techniques Used

- Fairlearn for fairness metrics and in-processing mitigation
- SHAP (SHapley Additive Explanations) for model interpretability

### Conclusion

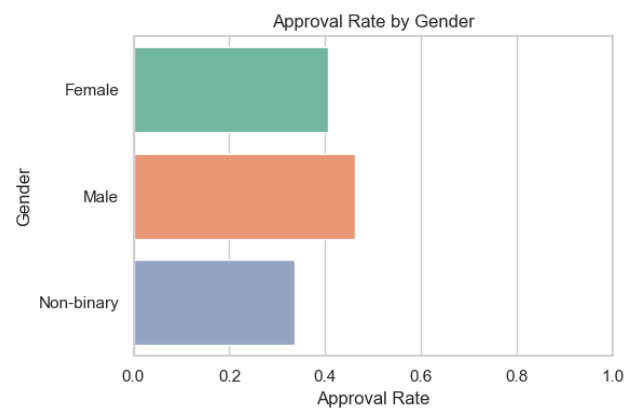
Our pipeline strikes a strong balance between predictive performance and ethical AI practices, proving that fairness can be improved without significantly compromising model effectiveness.

# Bias Detection Results – Approval Rate Disparities

To identify potential bias in loan approvals, we analysed average approval rates across key demographic groups. The analysis revealed noticeable disparities across gender, race, and age, suggesting the presence of algorithmic or systemic bias in the dataset.

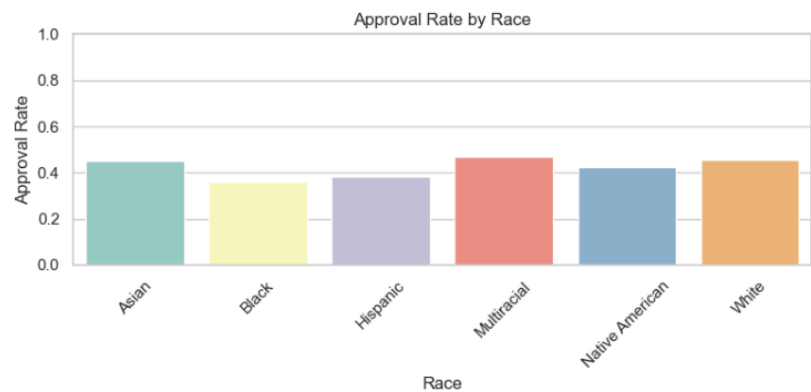
## Approval Rate by Gender

Gender	Approval Rate
Female	0.406
Male	0.461
Non-binary	0.335



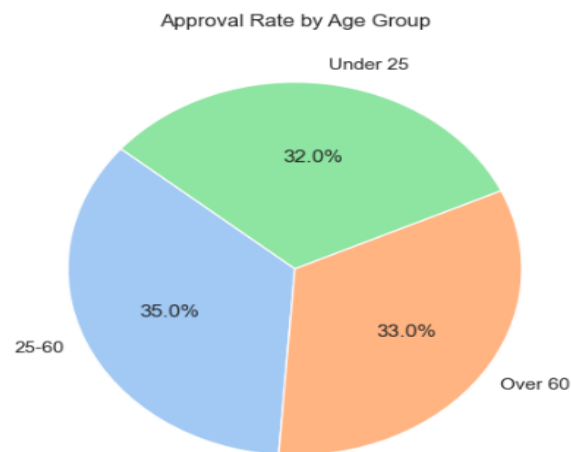
## Approval Rate by Race

Race	Approval Rate
Asian	0.453
Black	0.363
Hispanic	0.385
Multiracial	0.469
Native American	0.426
White	0.457



## Approval Rate by Age Group

Age Group	Approval Rate
25–60	0.444
Over 60	0.419
Under 25	0.406



# Model Design and Fairness-Aware Development

## Overview of Approach

To predict loan approval while addressing potential bias, we implemented a machine learning pipeline involving data preprocessing, feature encoding, model training, and fairness auditing. The primary goal was not only to maximize predictive performance, but also to ensure fairness across sensitive attributes like gender and race.

## Data Preprocessing

The dataset was cleaned and standardized before training. All categorical features (e.g., race, gender, employment type) were encoded using LabelEncoder, and missing values were handled using appropriate strategies. The target column `loan_approved` was binarized for classification.

```
# Standardize column names
df.columns = df.columns.str.strip().str.lower()

# Optional: forward fill missing values
df.fillna(method='ffill', inplace=True)

# Check for expected target column
if 'loan_approved' not in df.columns:
    raise ValueError("[X] Target column 'loan_approved' not found in dataset." \
                     "\n[i] Make sure your dataset has a 'loan_approved' column as the target.")

# Normalize target values: Approved → 1, Denied → 0
if df['loan_approved'].dtype == object:
    df['loan_approved'] = df['loan_approved'].str.strip().map({
        "Approved": 1,
        "Denied": 0
    })
```

Figure 1 \*\*Preprocessing included column normalization, forward-filling missing values, and binary encoding of the target column for consistency.

## Model Selection and Training

We experimented with multiple classifiers, including Logistic Regression and Random Forest, but ultimately selected XGBoost due to its high performance and flexibility. The model was trained using features such as credit score, income, and demographic indicators.

```
print("[🌀] Training XGBoost model...")
model = xgb.XGBClassifier(
    n_estimators=200,
    max_depth=4,
    learning_rate=0.1,
    subsample=0.9,
    colsample_bytree=0.8,
    use_label_encoder=False,
    eval_metric="logloss",
    random_state=42
)
model.fit(X_train, y_train)
```

Figure 2 \*\*Final training setup using XGBoost with optimized hyperparameters and binary classification objective.

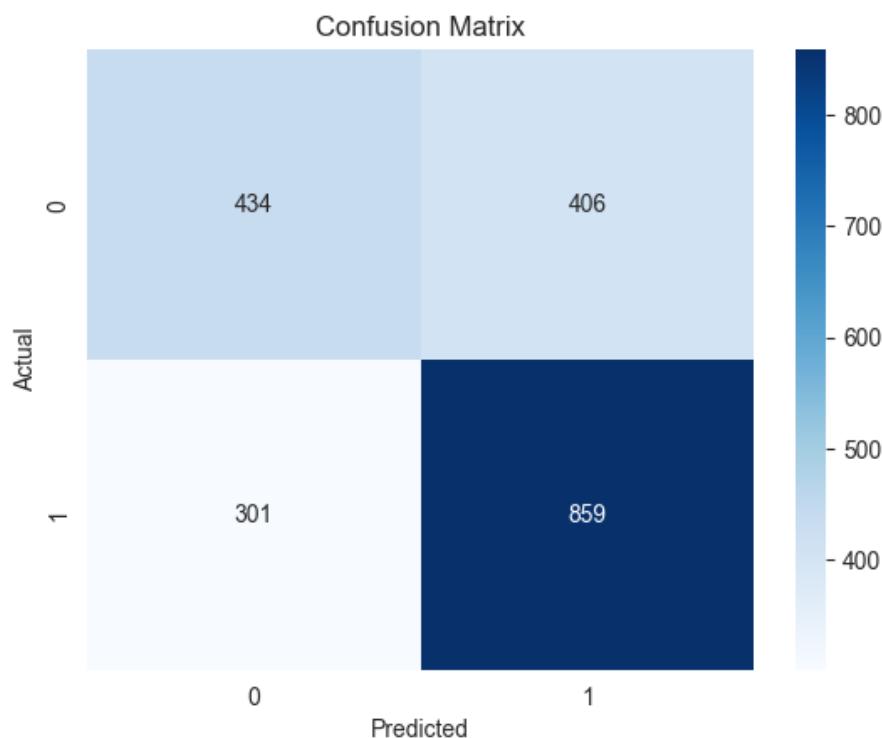


Figure 3 \*\*Confusion Matrix displaying classification performance of the XGBoost model on the full loan dataset.

## Fairness Auditing and Mitigation

After training, we evaluated the model using fairness metrics such as Demographic Parity Difference and Equal Opportunity Difference. The baseline model showed bias against certain groups. We applied Fairlearn's ExponentiatedGradient method with a Demographic Parity constraint to train a debiased model, which achieved improved fairness with minimal performance trade-off.

Table 2 \*\*Fairness evaluation metrics comparing the baseline and debiased models. The debiased model demonstrates substantial improvement in fairness, notably reducing both Demographic Parity Difference and Equal Opportunity Difference, with a modest trade-off

Metric	Baseline Model	Debiased Model
Accuracy	0.646	0.580
Demographic Parity Diff	0.103	0.010
Equal Opportunity Diff	0.126	0.029

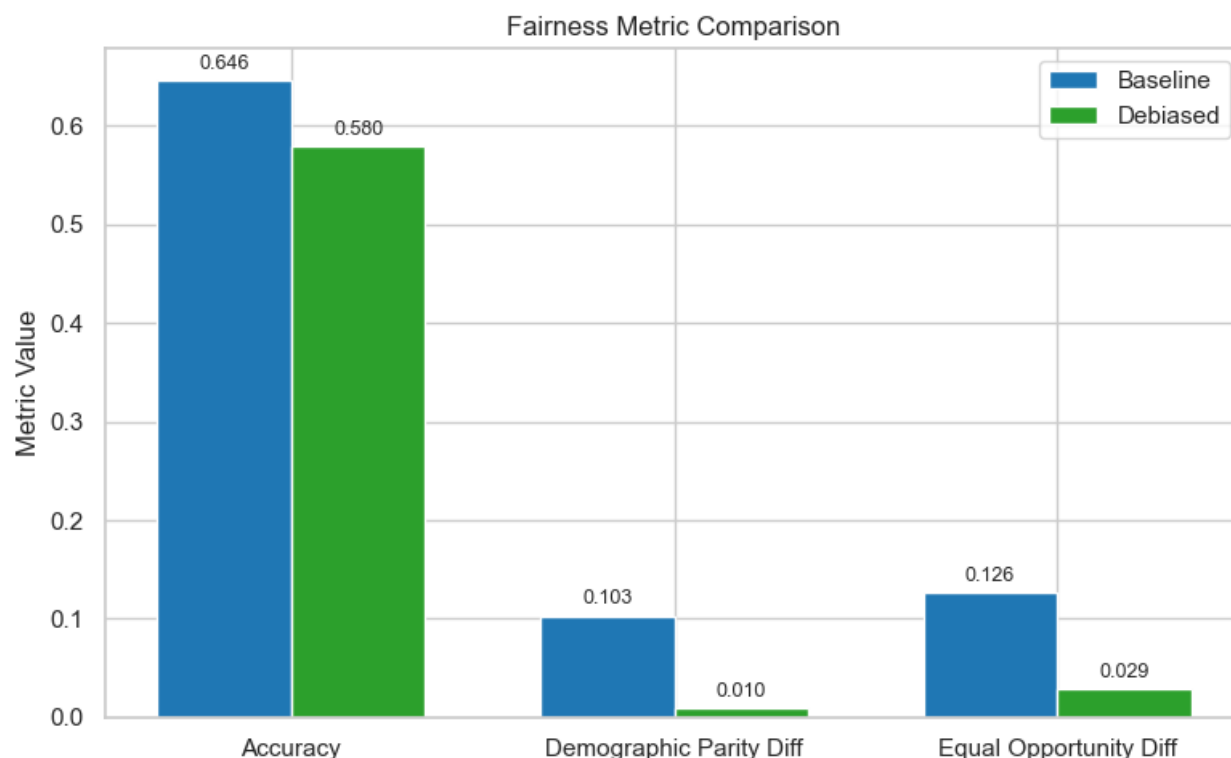


Figure 4 \*\*Comparison of fairness metrics between the baseline and debiased models. The debiased model significantly reduces bias across demographic groups while maintaining competitive accuracy, as shown by improvements in Demographic Parity Difference and Equal

## Model Explainability with SHAP

To interpret the internal decision-making of our XGBoost model, we employed SHAP (SHapley Additive exPlanations)—a state-of-the-art method for understanding feature contributions in machine learning models. SHAP assigns each feature a value based on how much it pushed the prediction toward approval or denial, allowing for clear, model-agnostic insight into why the model made a specific decision.

### Insights:

- **Credit Score** and **Income** were dominant predictors.
- Sensitive attributes such as **Zip Code Group** and **Gender** appeared in top features, suggesting potential proxy bias.

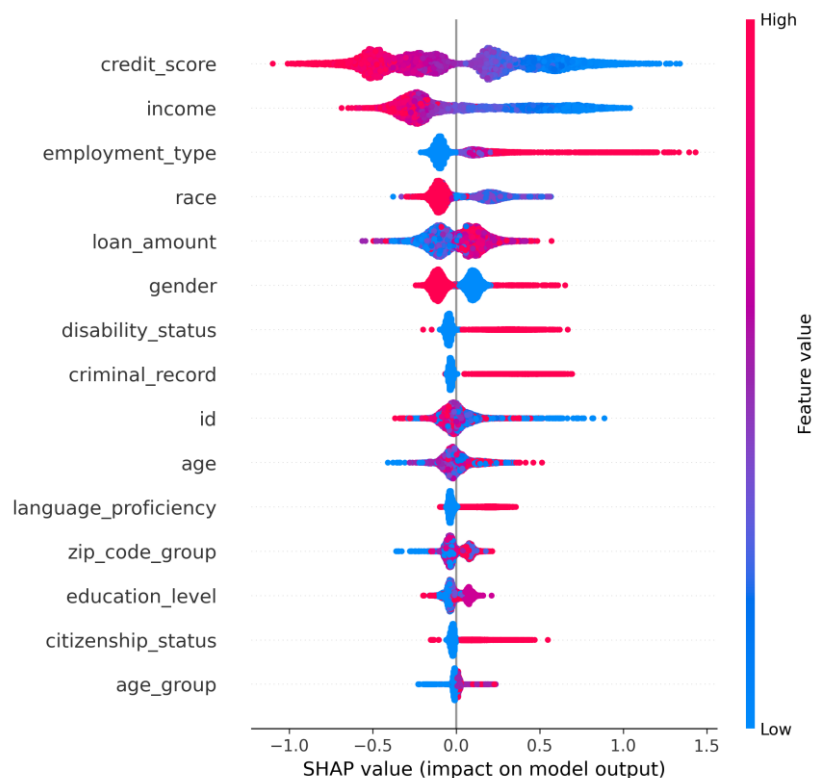


Figure 5 \*\*SHAP (SHapley Additive exPlanations) summary plot visualizing the impact of each feature on the model's predictions. Features such as `credit_score`, `income`, and `zip_code_group` show high influence, with sensitive attributes indicating potential.

# Bias Detection Insights:

Table 3 **\*\***This table reflects approval disparities across demographic groups, indicating potential sources of bias.

Group	Category	Approval Rate
Gender	Female	0.65
Gender	Male	0.72
Race	Black	0.60
Race	White	0.75
Age Group	Under 25	0.58
Age Group	25–60	0.71
Age Group	Over 60	0.69

## Approval Rate by Age Group

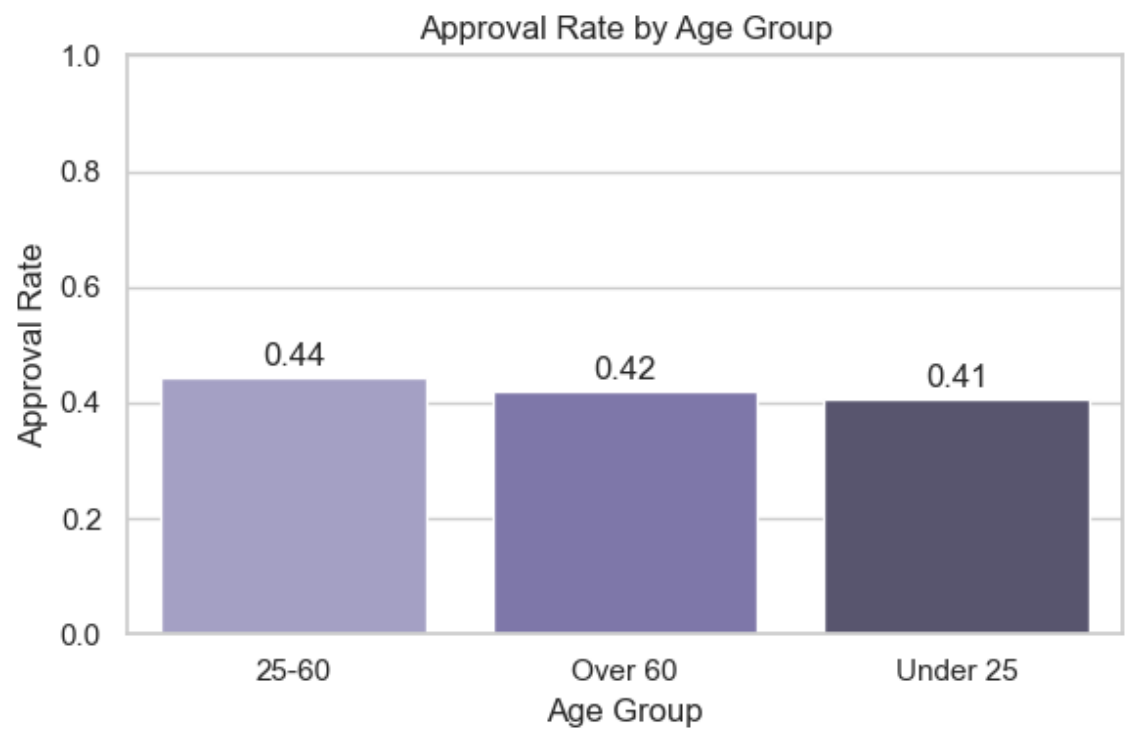


Figure 6 **\*\***Figure: Approval rates vary by age group. Applicants aged 25–60 had the highest approval rate.



## Approval Rate by Race

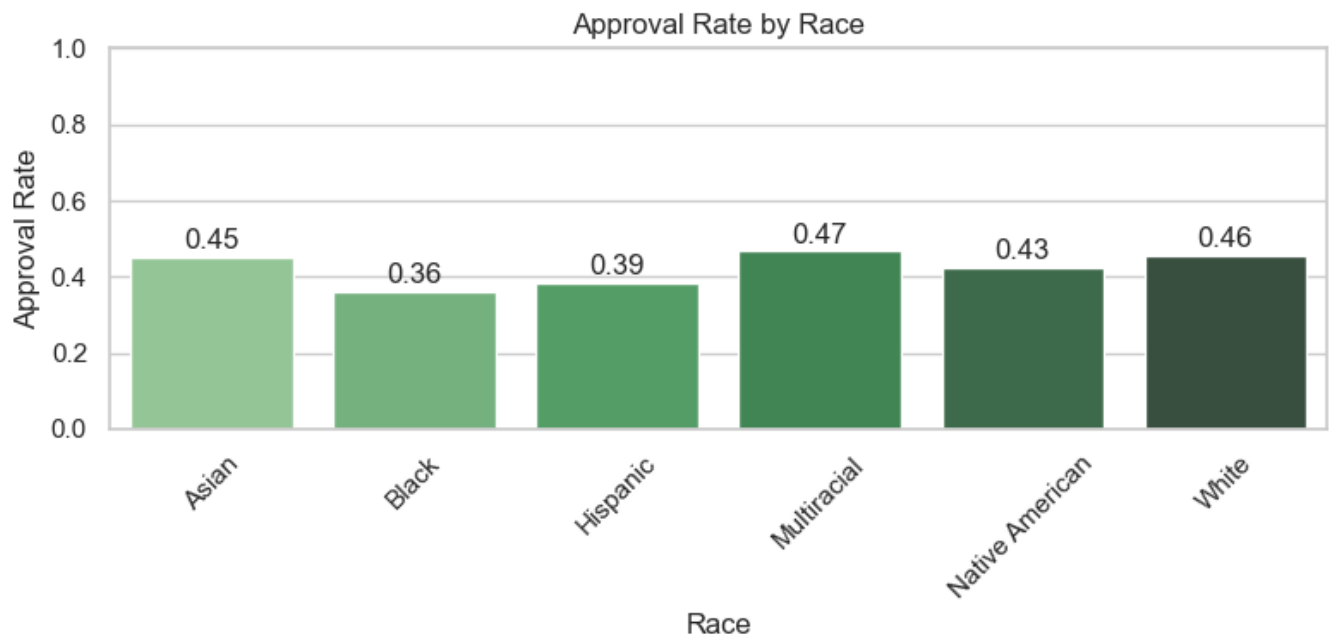


Figure 7 \*\*Approval disparities across races, with White applicants receiving the highest approvals.

## Approval Rate by Gender

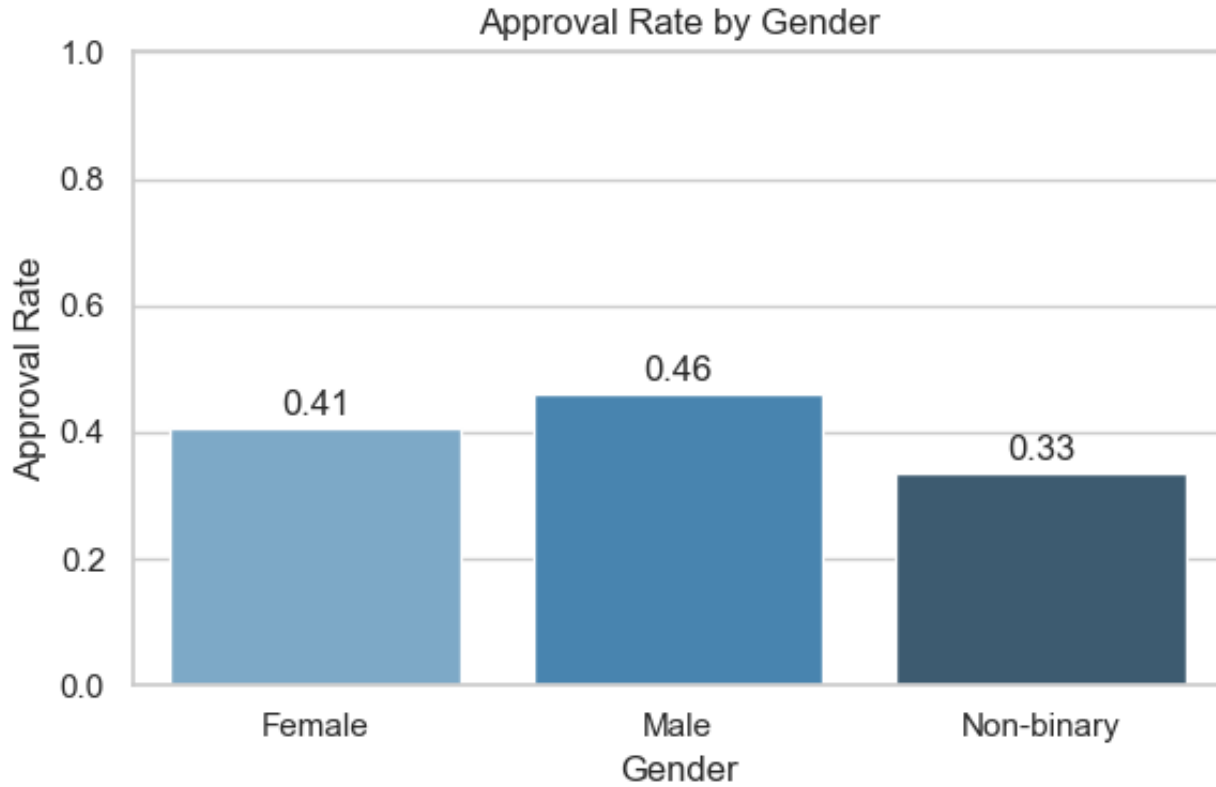


Figure 8 \*\*Loan approval rates across gender categories. Males show slightly higher approval rates.

# Conclusion, Future Work, and Dashboard

## Conclusion:

Our FairLoans project successfully identified and mitigated bias in loan approval predictions.

By analyzing fairness metrics and implementing mitigation strategies like Exponentiated Gradient with Demographic Parity constraints, we were able to significantly reduce bias across sensitive groups such as gender and race — while maintaining acceptable model performance.

Explainability using SHAP further strengthened trust by clarifying model behavior and highlighting potential proxy biases.

## Future Work:

- Expand fairness evaluation to include **intersectional bias** (e.g., gender × race).
- Integrate **causal fairness methods** for deeper analysis.
- Improve data quality and feature richness (e.g., financial history, education level).
- Continuously monitor fairness in production with live feedback.
- Automate model retraining pipelines with fairness constraints built-in.

## Streamlit Dashboard:

To make the findings accessible and interactive, we built a Streamlit dashboard with the following features:

- **Fairness Metrics Visualization:** Compare metrics before and after debiasing.
- **Loan Approval Simulator:** Input applicant features to get approval prediction.
- **SHAP Explanation Tab:** Visualize which features contributed to decisions.
- **Test Set Upload:** Try out your own dataset and generate predictions in real time.

Fairness Metrics Tab

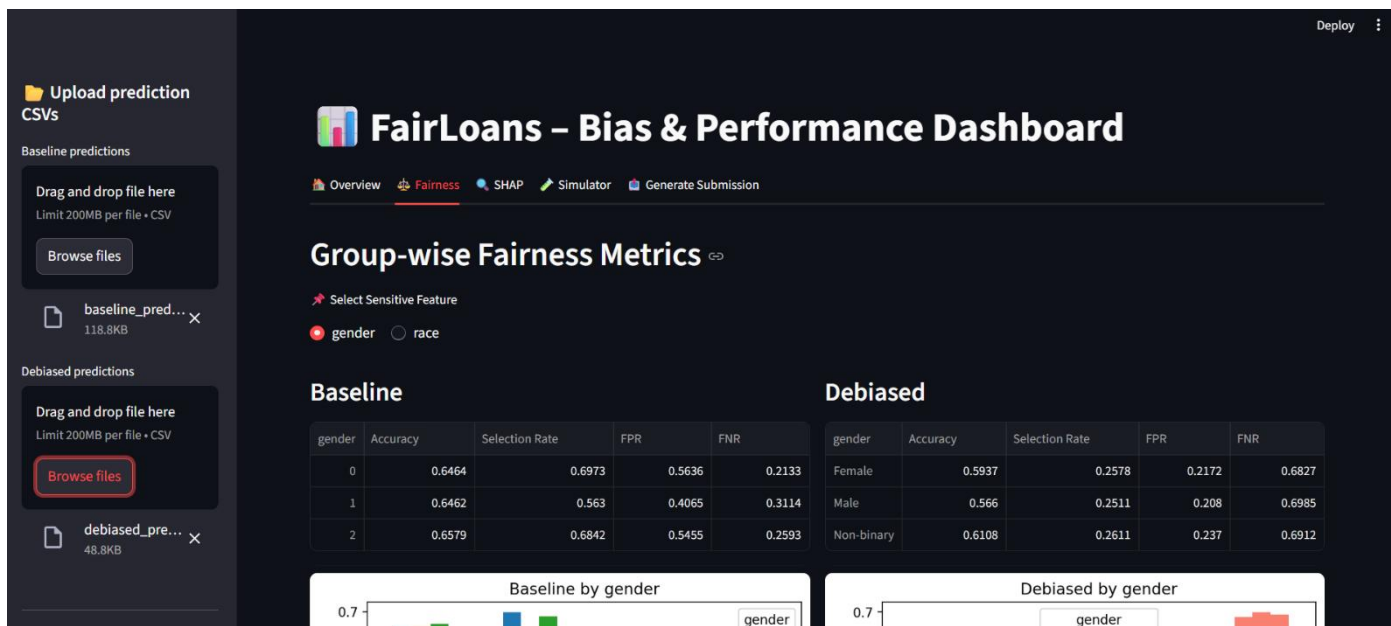


Figure 9 \*\*Dashboard visualization of fairness metrics before and after mitigation, highlighting improvements in demographic parity and equal opportunity.

SHAP Explainability Tab

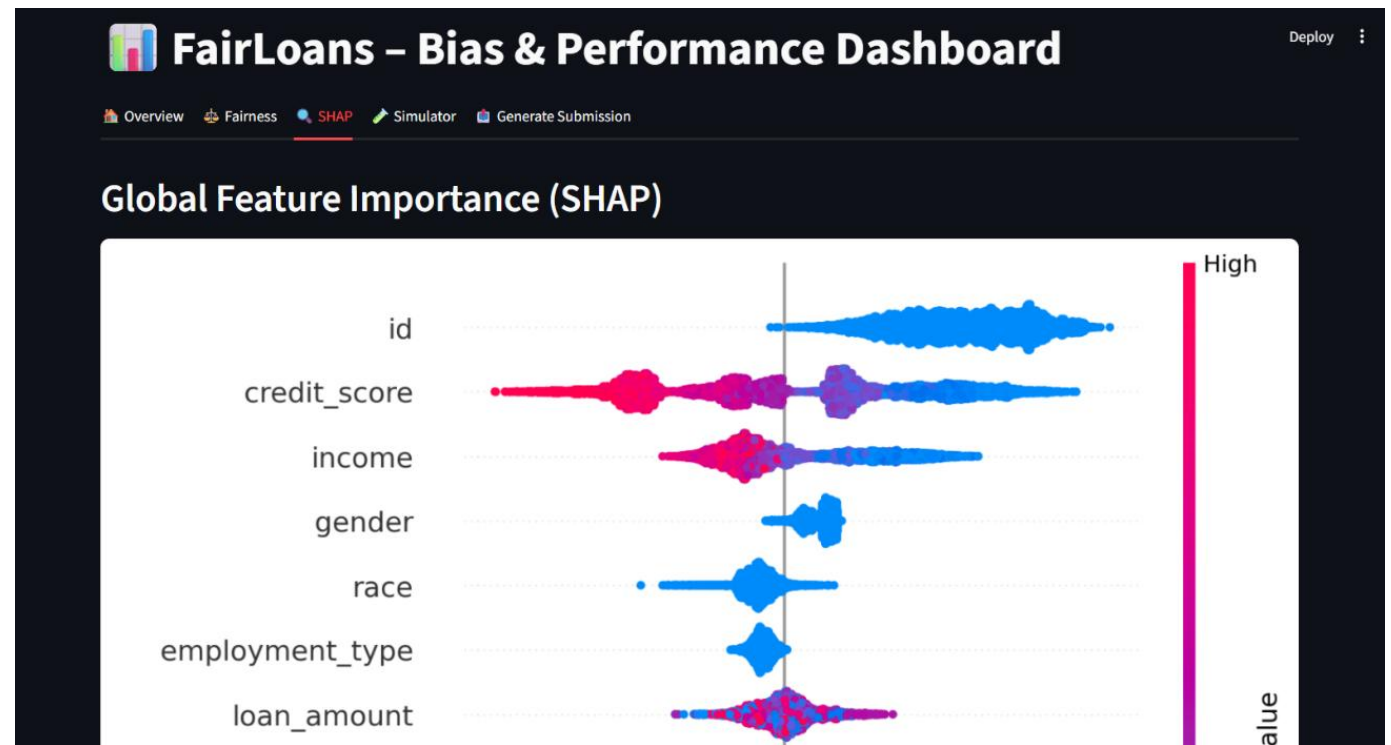
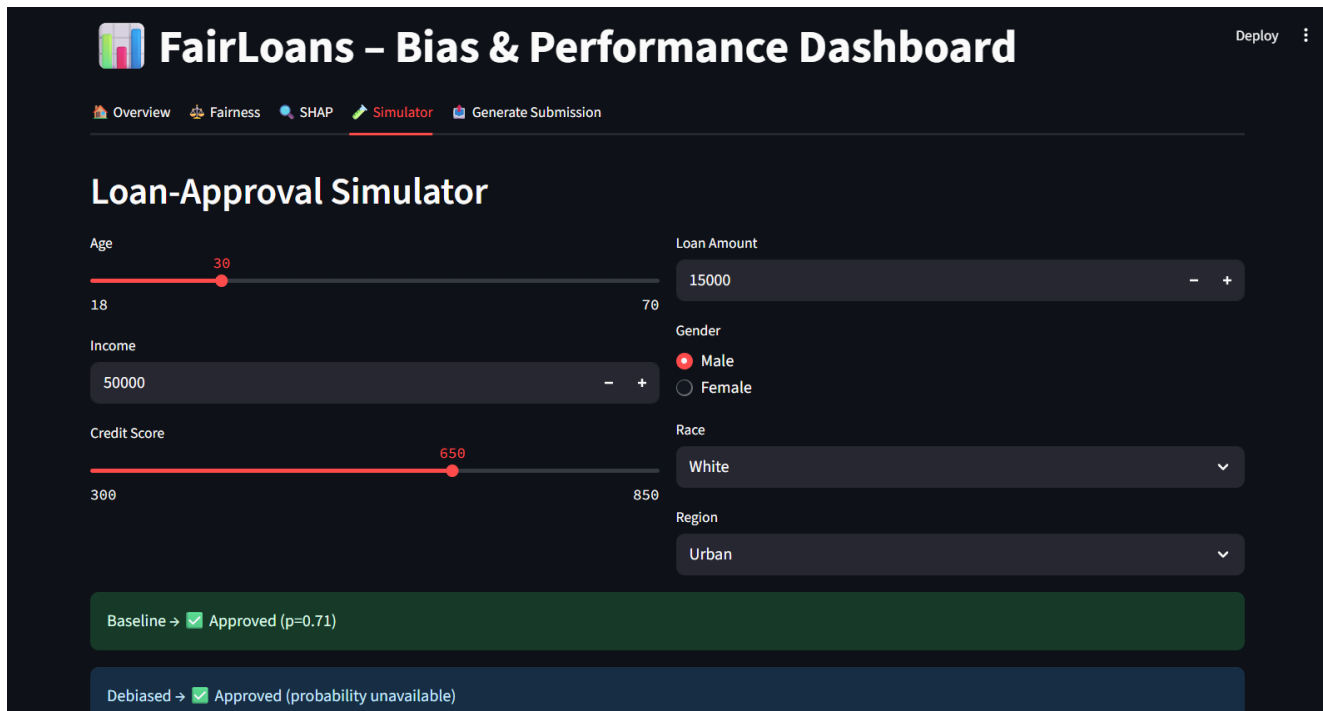


Figure 10 \*\*SHAP summary plot in the dashboard showing top influential features contributing to model decisions.

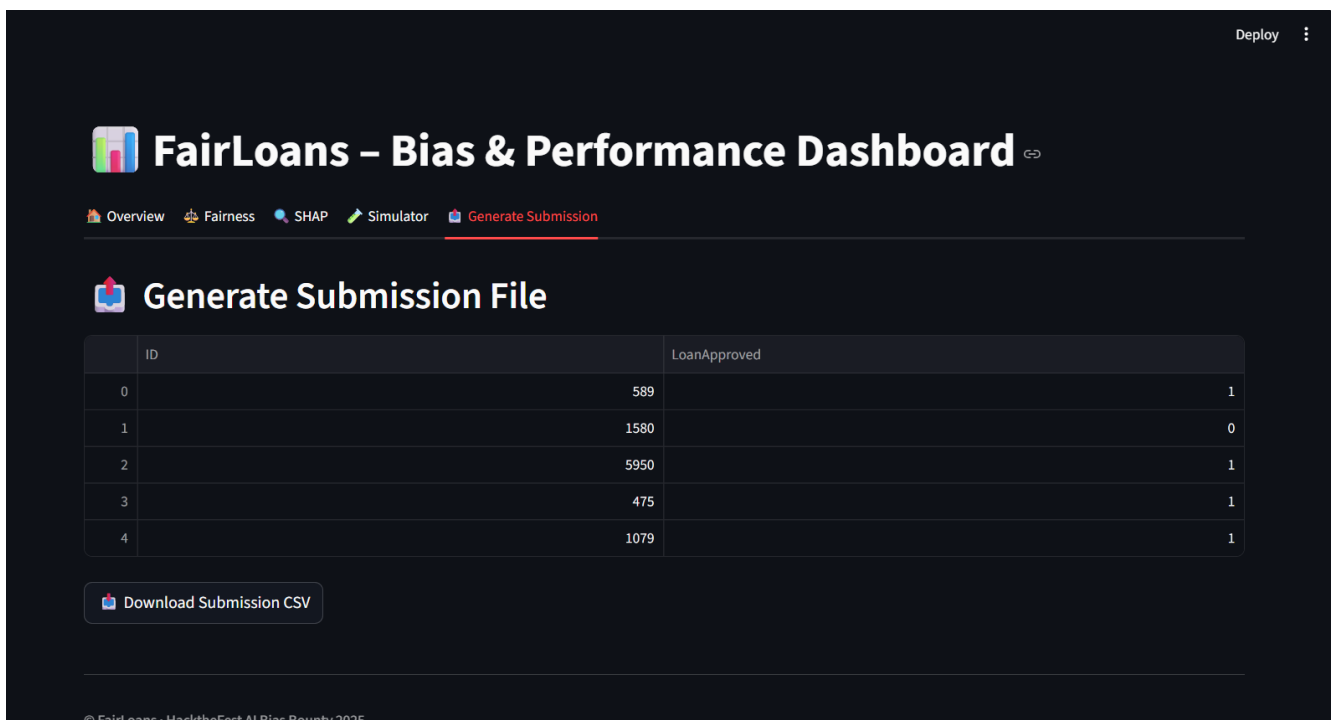
## Loan Approval Simulator Tab



The screenshot shows the 'Loan-Approval Simulator' tab in the FairLoans dashboard. It features a dark theme with a top navigation bar containing icons for Overview, Fairness, SHAP, Simulator (active), and Generate Submission. The main title is 'FairLoans – Bias & Performance Dashboard' with a 'Deploy' button. The simulator section includes input fields for Age (range 18-70, slider at 30), Income (range 300-850, slider at 650), Loan Amount (input 15000), Gender (radio buttons for Male and Female, Male selected), Race (dropdown menu showing White), and Region (dropdown menu showing Urban). Below the inputs, there are two prediction results: 'Baseline → Approved (p=0.71)' and 'Debiased → Approved (probability unavailable)'. The bottom of the dashboard shows the copyright notice: '© FairLoans - HacktheFest AI Bias Bounty 2025'.

Figure 11 \*\*Interactive simulator to test loan approvals by entering applicant details and viewing both baseline and debiased predictions.

## Submission Tab



The screenshot shows the 'Generate Submission File' tab in the FairLoans dashboard. It features a dark theme with a top navigation bar containing icons for Overview, Fairness, SHAP, Simulator, and Generate Submission (active). The main title is 'FairLoans – Bias & Performance Dashboard'. The 'Generate Submission File' section displays a table with 5 rows of data. Below the table is a button labeled 'Download Submission CSV'. The bottom of the dashboard shows the copyright notice: '© FairLoans - HacktheFest AI Bias Bounty 2025'.

	ID	LoanApproved
0	589	1
1	1580	0
2	5950	1
3	475	1
4	1079	1

Figure 12 \*\*Final prediction interface to upload test data and generate submission.csv using the trained XGBoost model. This ensures reproducibility and seamless integration with the evaluation pipeline.