# SMART LENDING ANALYTICS: PREDICTING LOAN APPROVAL, AND INTEREST RATES


Professor Ziyi Cao

## BUAN 6341.002 - Applied Machine Learning - S25

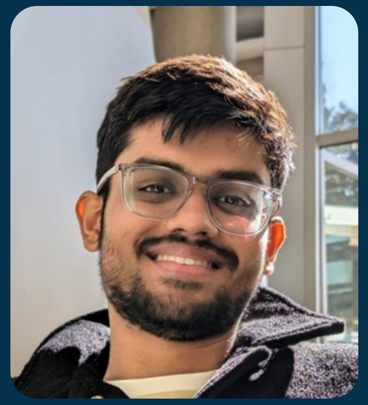
Team Members: Gaurang Damjibhai Vora, Javier Arguelles Badillo, Nirmit Pradip Patel, Samira Salehi, and Thriksha Giriraju

04/22/2025

# Meet Our Team

Gaurang Damjibhai Vora

Javier Arguelles Badillo



Nirmit Pradip Patel

Samira Salehi

Thriksha Giriraju

# Table Of Contents

# CORE PROBLEMS

# MOTIVATION

Manual review processes

ML

REDUCE PROCESSING TIME

REDUCE RISK OF BIAS

Loan Approval Time

TRADITIONAL BANKS

30 DAYS AVERAGE

*FINTECH (UPSTAR, LENDINGCLUB)*

UNDER 5 MIN USING ML

Bias by race


CHICAGO BANKS

150% ↑ DENIAL RATE FOR BLACK APPLICANTS

*ML*

33% REDUCTION IN INTEREST RATE DISPARITIES

# OBJECTIVES

1. Loan Approval Prediction (Classification)


2. Interest Rate Prediction (Regression)


# METHODOLOGY

MODELS:
- LOGISTIC REGRESSION
- RANDOM FOREST
- DECISION TREE
- XGBOOST

TARGET: LOAN_STATUS (0 = REJECTED, 1 = APPROVED)

MODELS:
- LINEAR REGRESSION
- DECISION TREE

TARGET: LOAN_INT_RATE (NUMERIC INTEREST RATE %)

# DATA OVERVIEW

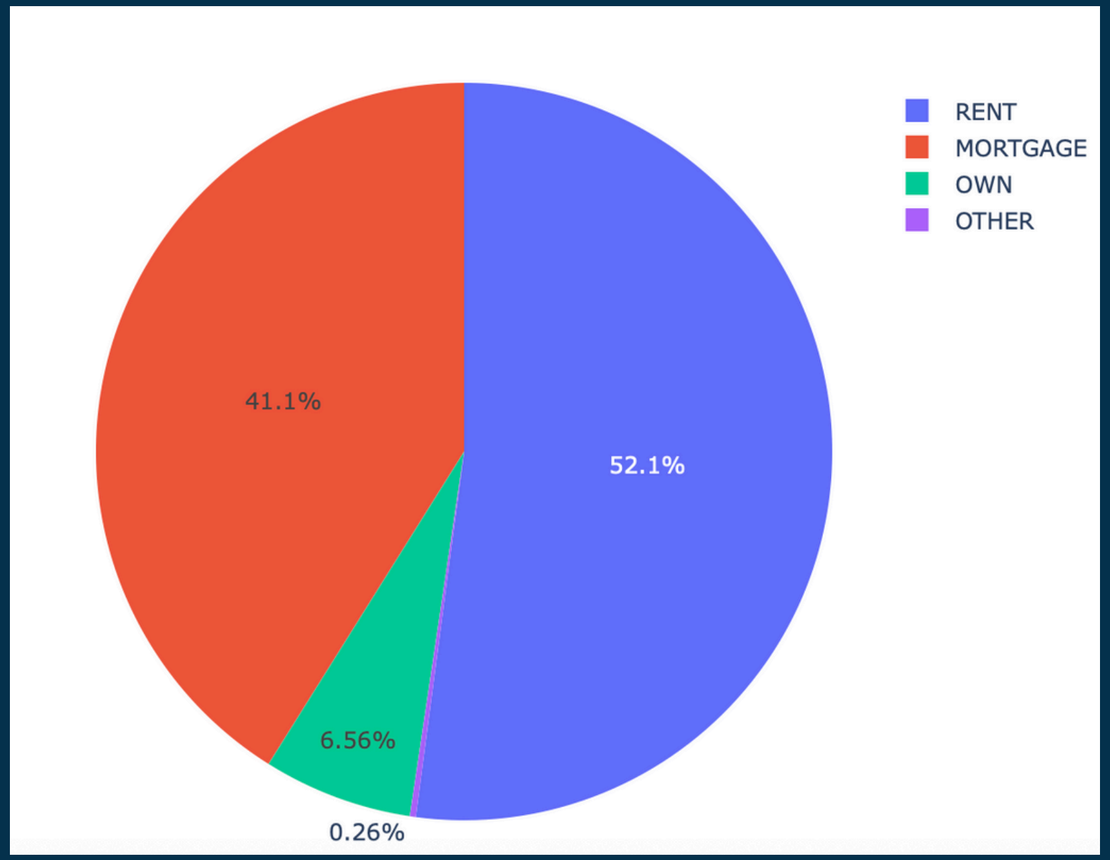
- 45,000 entries

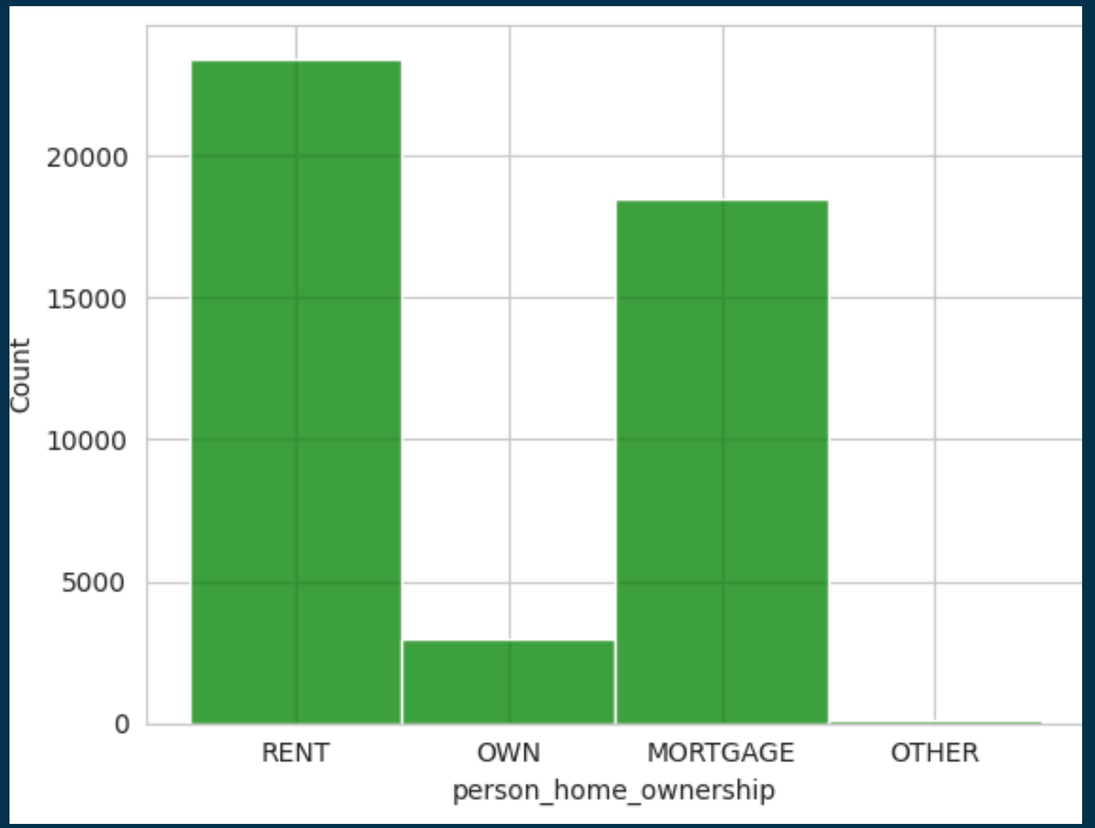
| Data Type | Varibales |
|---|---|
| int64 | • person_age,<br>• person_emp_exp,<br>• credit_score,<br>• **loan_status *** |
| float64 | • person_income,<br>• loan_amnt,<br>• **loan_int_rate ***<br>• loan_percent_income,<br>• cb_person_cred_hist_length |
| object | • person_gender,<br>• person_education,<br>• person_home_ownership,<br>• loan_intent,<br>• previous_loan_defaults_on_file |

- Converted:
  - Type of person’s age - float to integer

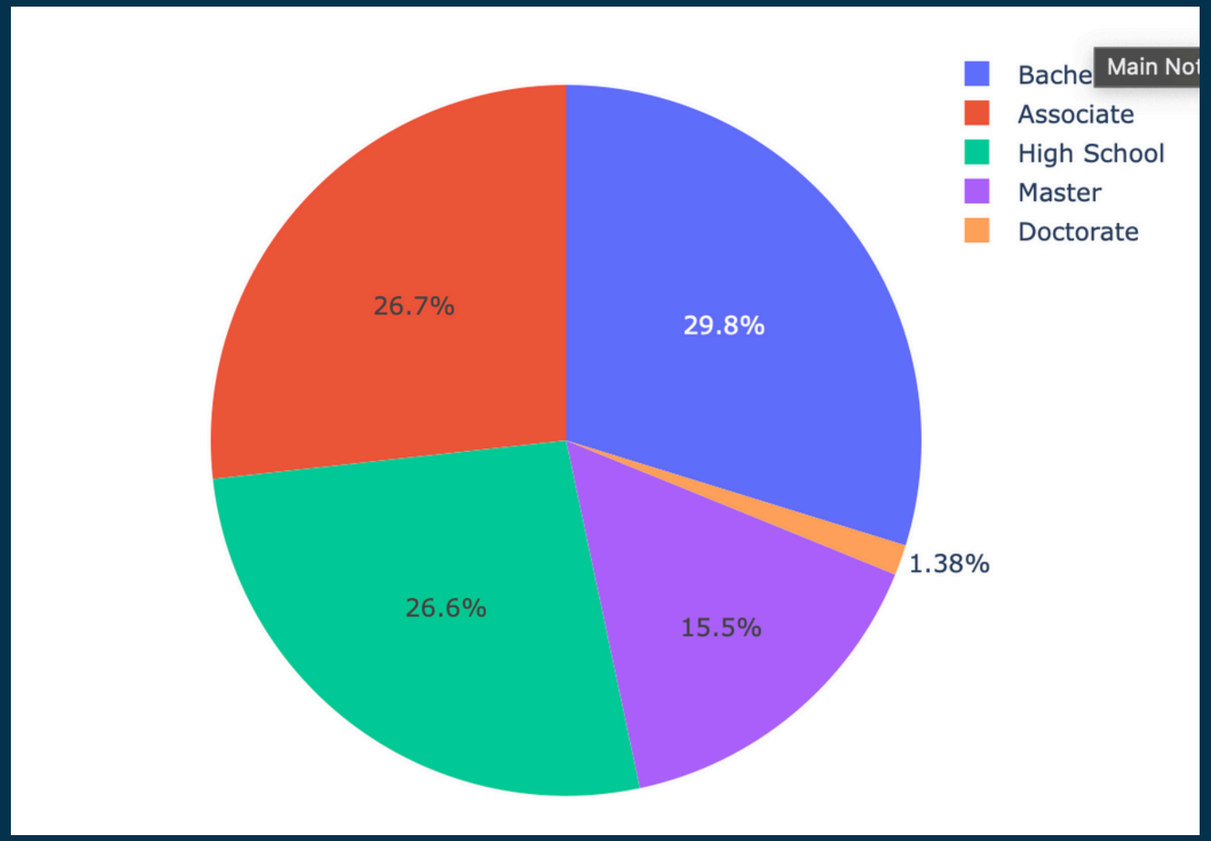
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45000 entries, 0 to 44999
Data columns (total 14 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   person_age                      45000 non-null  int64
 1   person_gender                   45000 non-null  object
 2   person_education                45000 non-null  object
 3   person_income                   45000 non-null  float64
 4   person_emp_exp                  45000 non-null  int64
 5   person_home_ownership           45000 non-null  object
 6   loan_amnt                       45000 non-null  float64
 7   loan_intent                     45000 non-null  object
 8   loan_int_rate                   45000 non-null  float64
 9   loan_percent_income             45000 non-null  float64
 10  cb_person_cred_hist_length      45000 non-null  float64
 11  credit_score                    45000 non-null  int64
 12  previous_loan_defaults_on_file  45000 non-null  object
 13  loan_status                     45000 non-null  int64
dtypes: float64(5), int64(4), object(5)
memory usage: 4.8+ MB
```

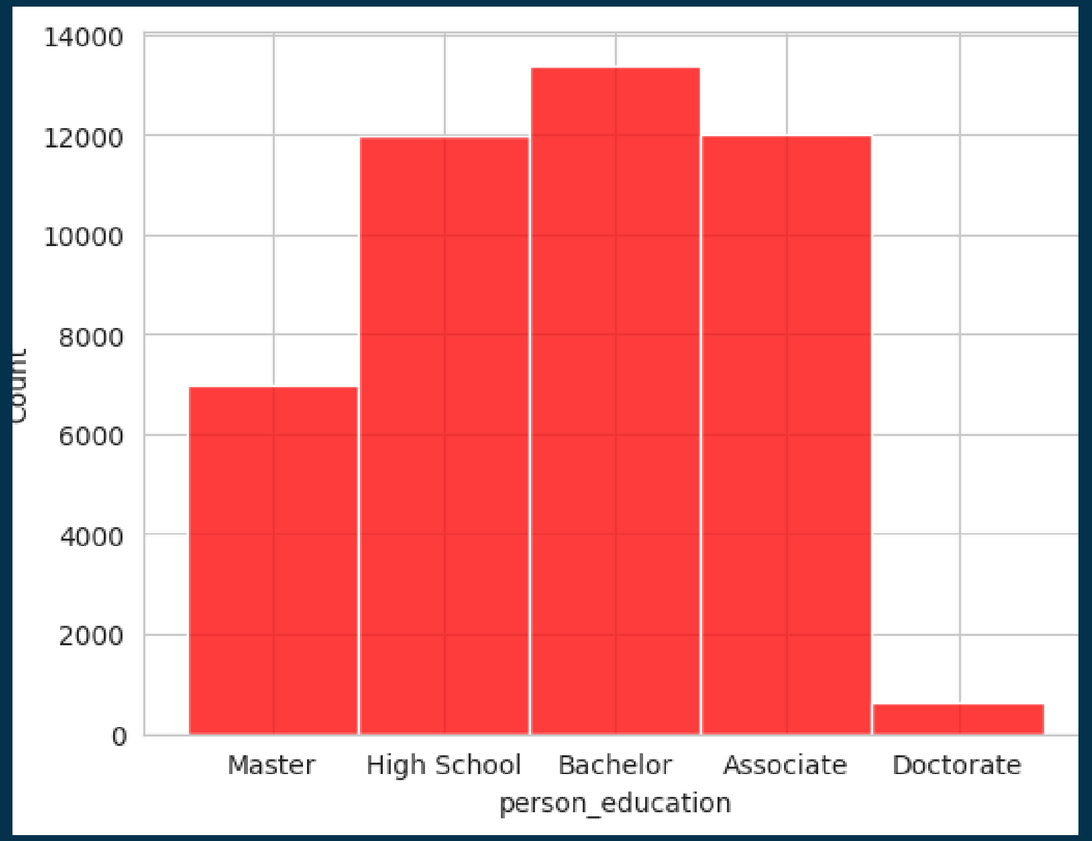
# VISUALIZE THE PATTERN/RELATIONSHIP PERSON HOME OWNERSHIP



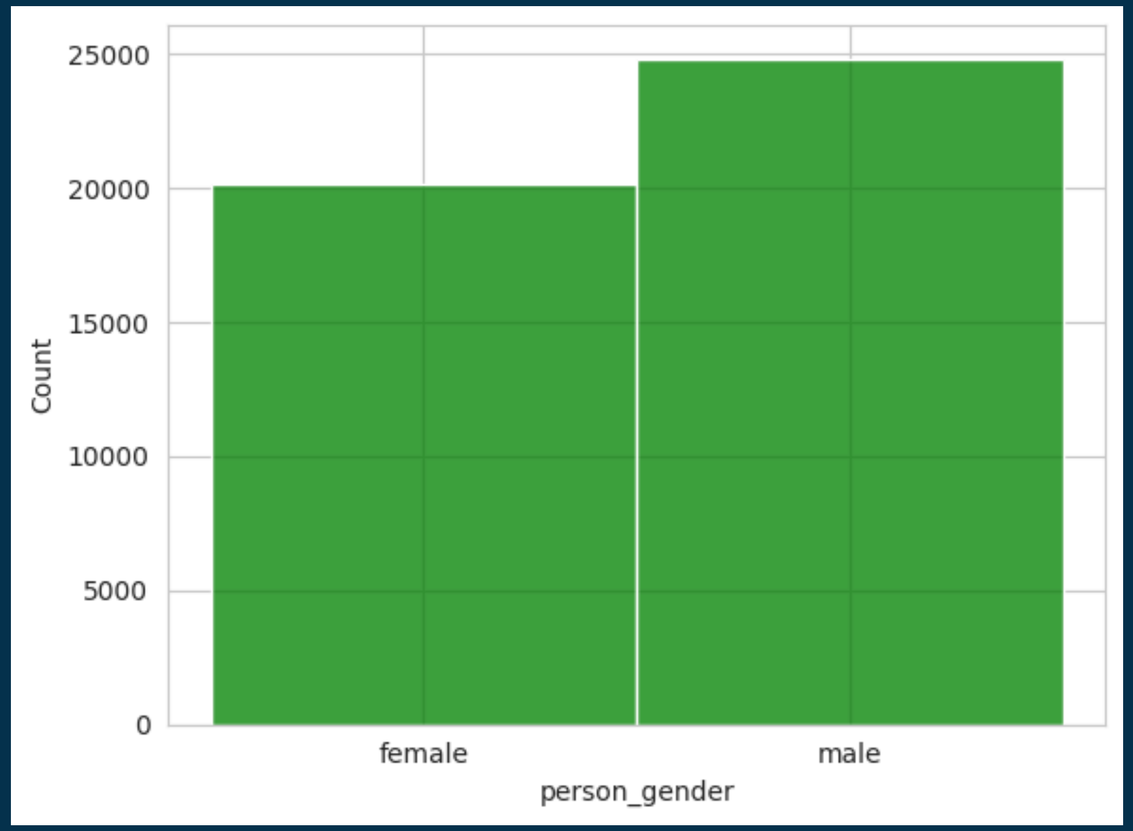
# VISUALIZE THE PATTERN/RELATIONSHIP PERSON EDUCATION

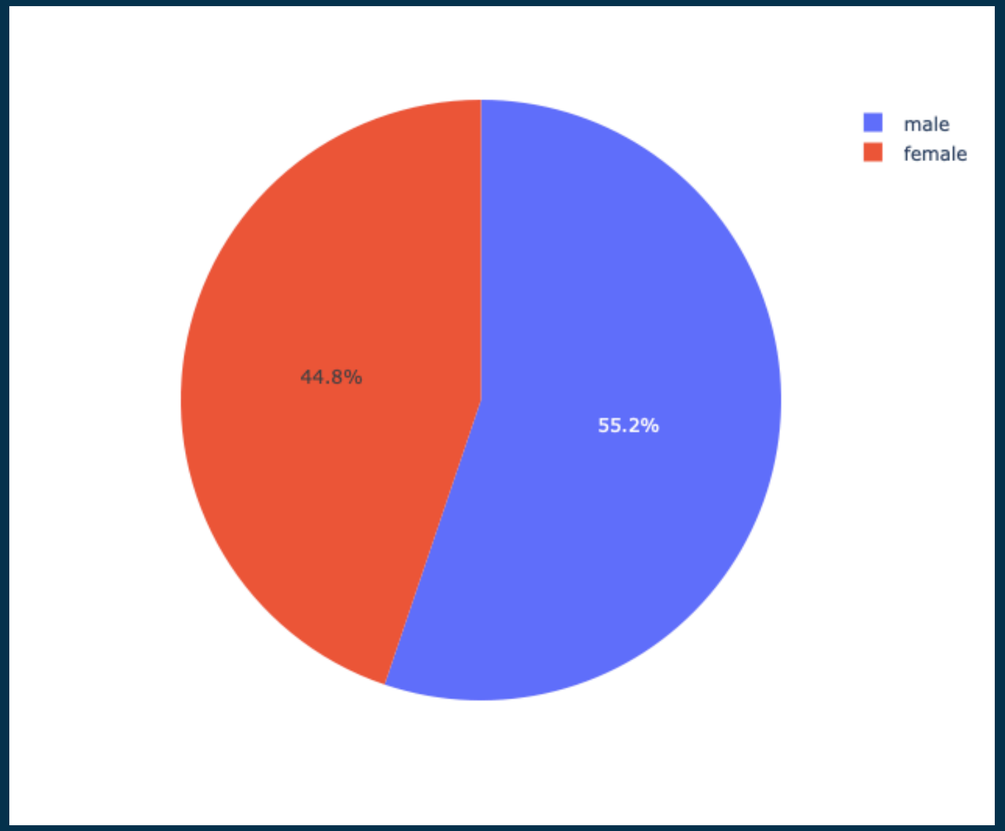


# VISUALIZE THE PATTERN/RELATIONSHIP

# MALE- FAMALE



# DATA PROCESSING

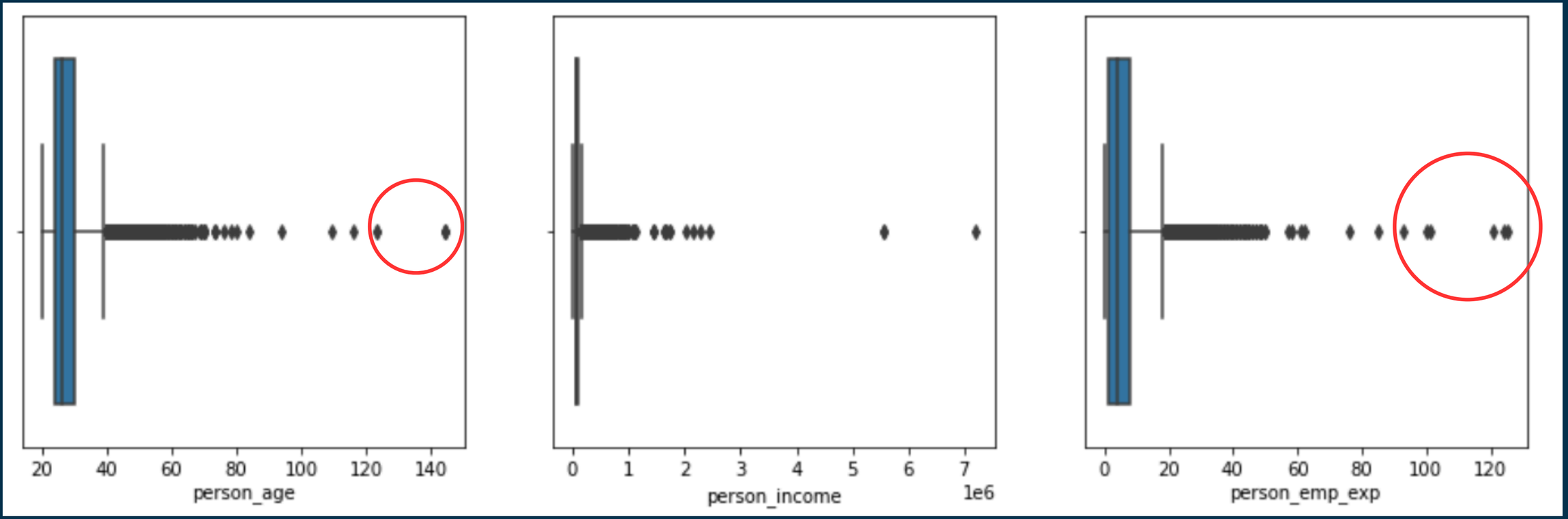
## Is the outlier an error or a real anomaly?

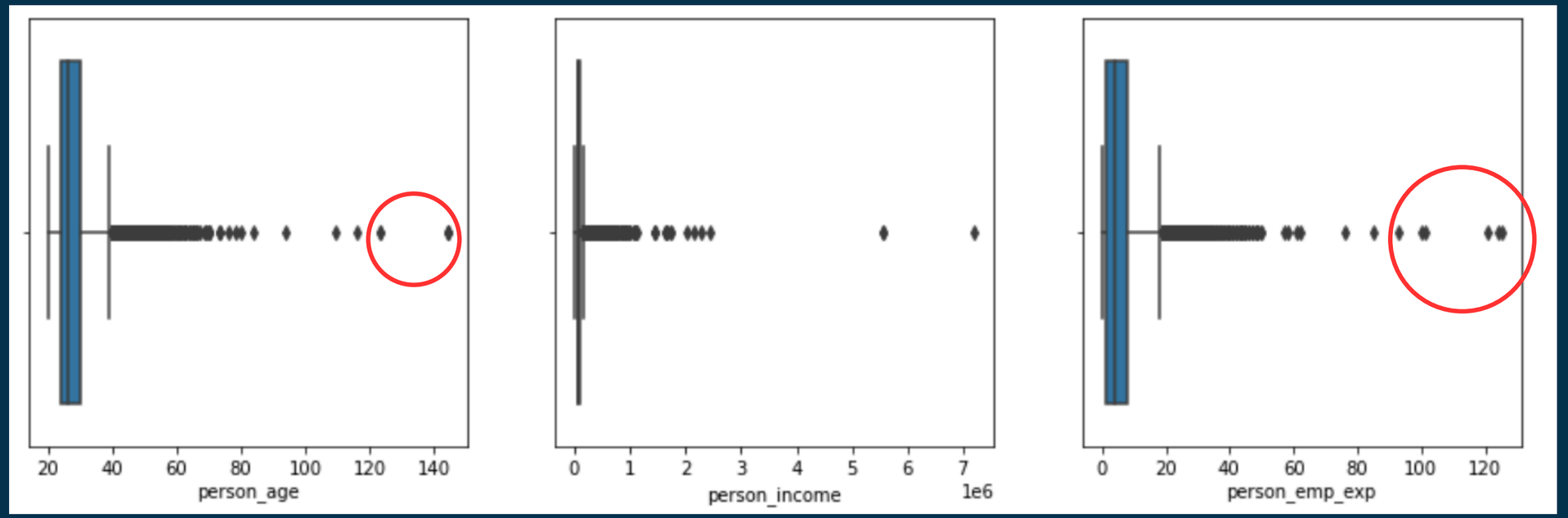
- Many variables show high-end outliers, especially for age, income, and experience.
- Some outliers are probably legitimate (e.g., high income), others might be data quality issues (e.g., age = 144 or 125 years of work experience).
- Outliers detected and stored in 'outlier_data'. Remaining data cleaned.

| Variable | Min Value | Max Value | Lower Bound | Upper Bound |
|---|---|---|---|---|
| person_age | 20 | 144 | 15.0 | 39.0 |
| person_income | 8,000.0 | 7,200,766.0 | -25,673.875 | 168,667.125 |
| person_emp_exp | 0 | 125 | -9.5 | 18.5 |
| loan_amnt | 500.0 | 35,000.0 | -5,855.875 | 23,093.125 |
| loan_int_rate | 5.42 | 20.0 | 1.99 | 19.59 |
| loan_percent_income | 0.0 | 0.66 | -0.11 | 0.37 |
| cb_person_cred_hist_length | 2.0 | 30.0 | -4.5 | 15.5 |
| credit_score | 390 | 850 | 497.5 | 773.5 |

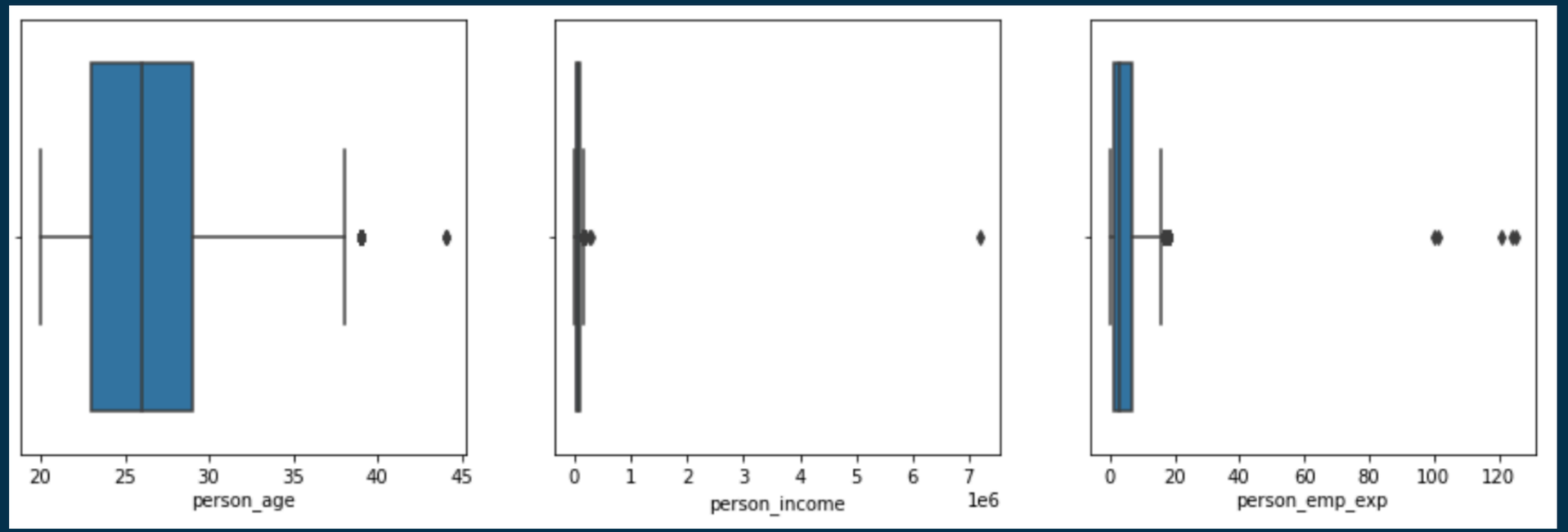
# OUTLIERS HANDLING

## Boxplot numerical columns before removing outliers





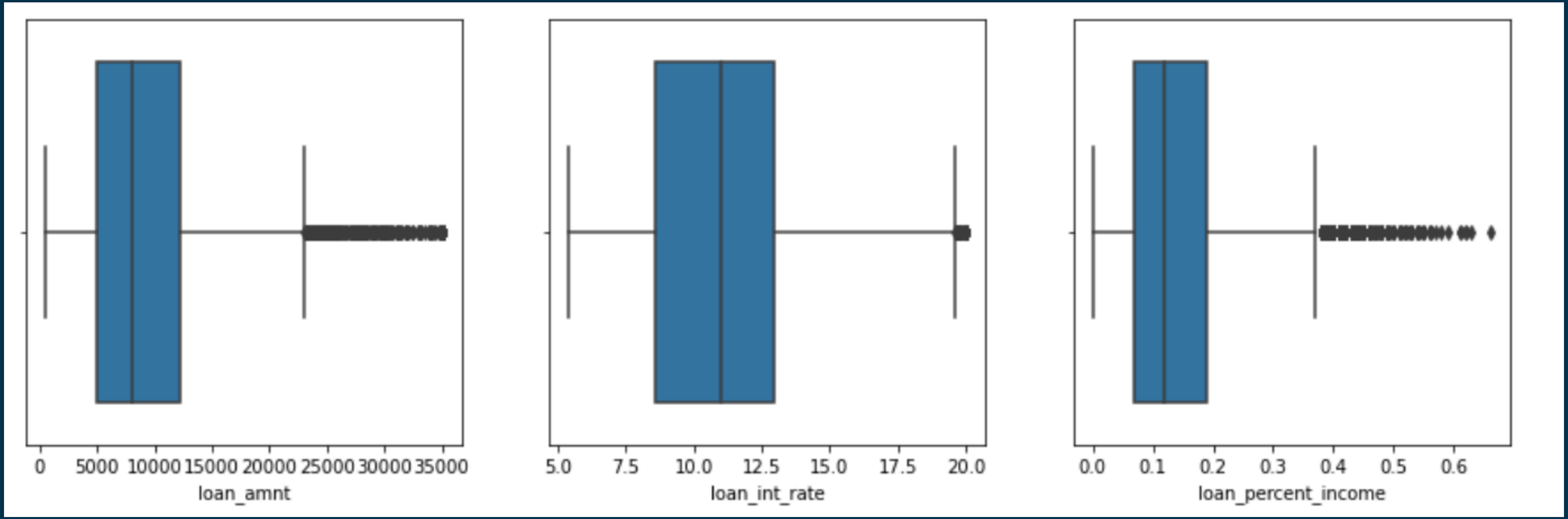

BEFORE

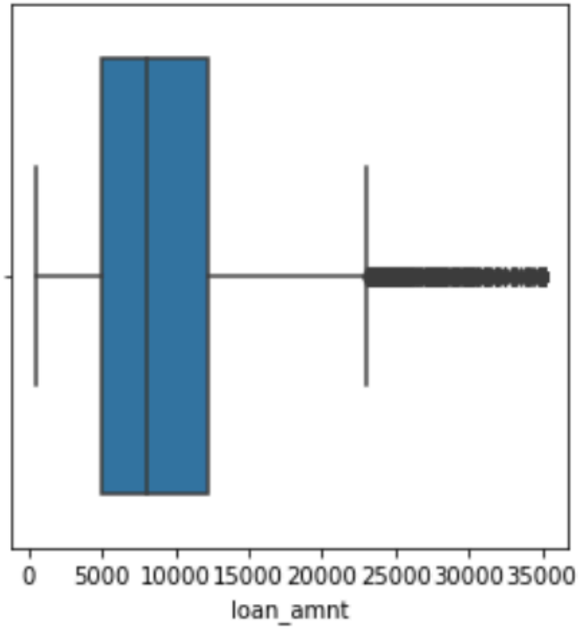


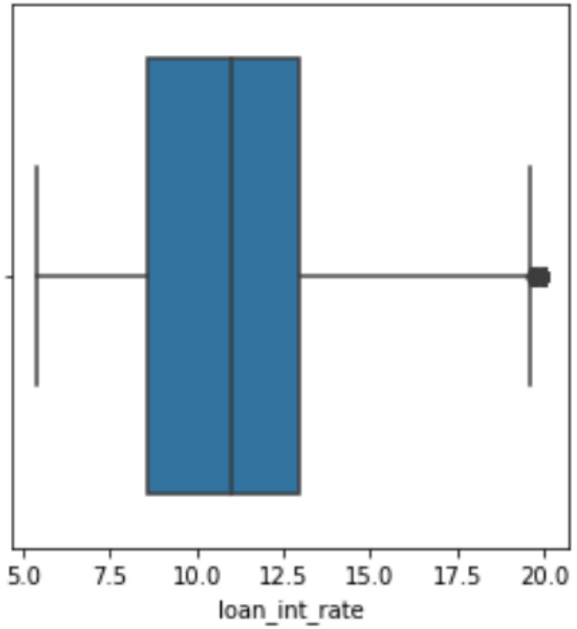

AFTER

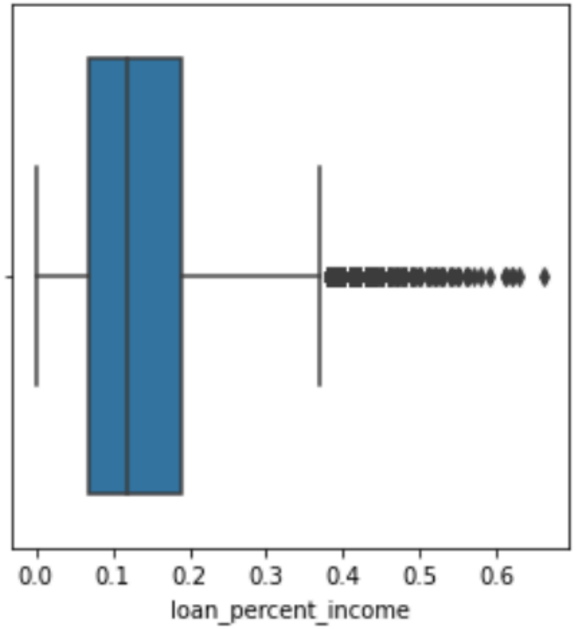
# OUTLIERS HANDLING

## Boxplot numerical columns before removing outliers

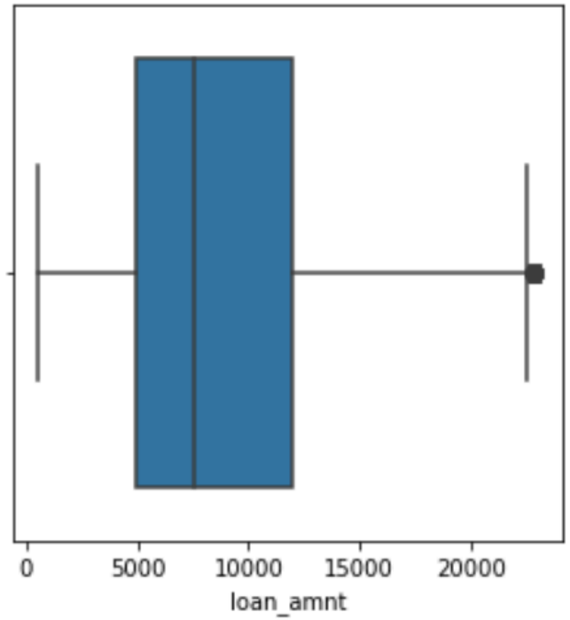


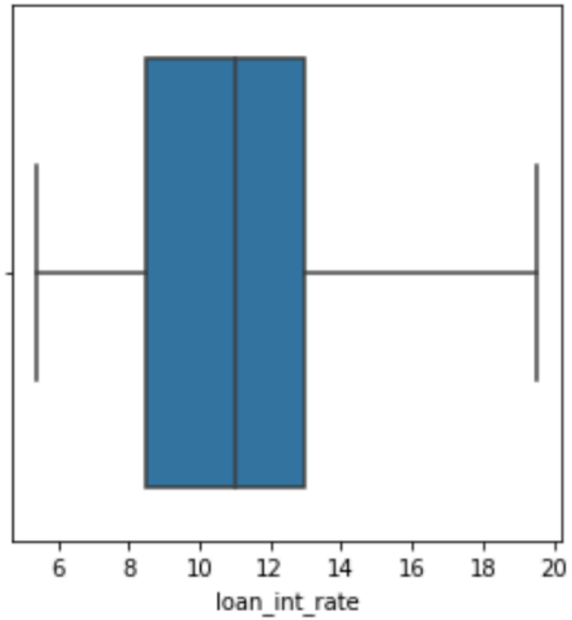


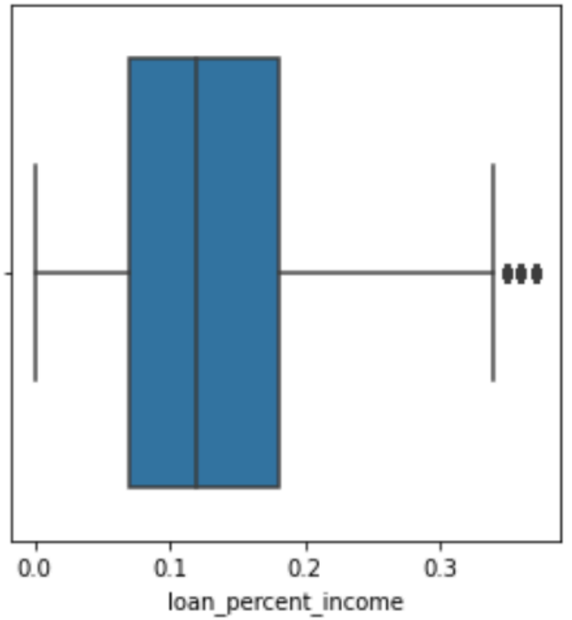




**BEFORE**




**AFTER**

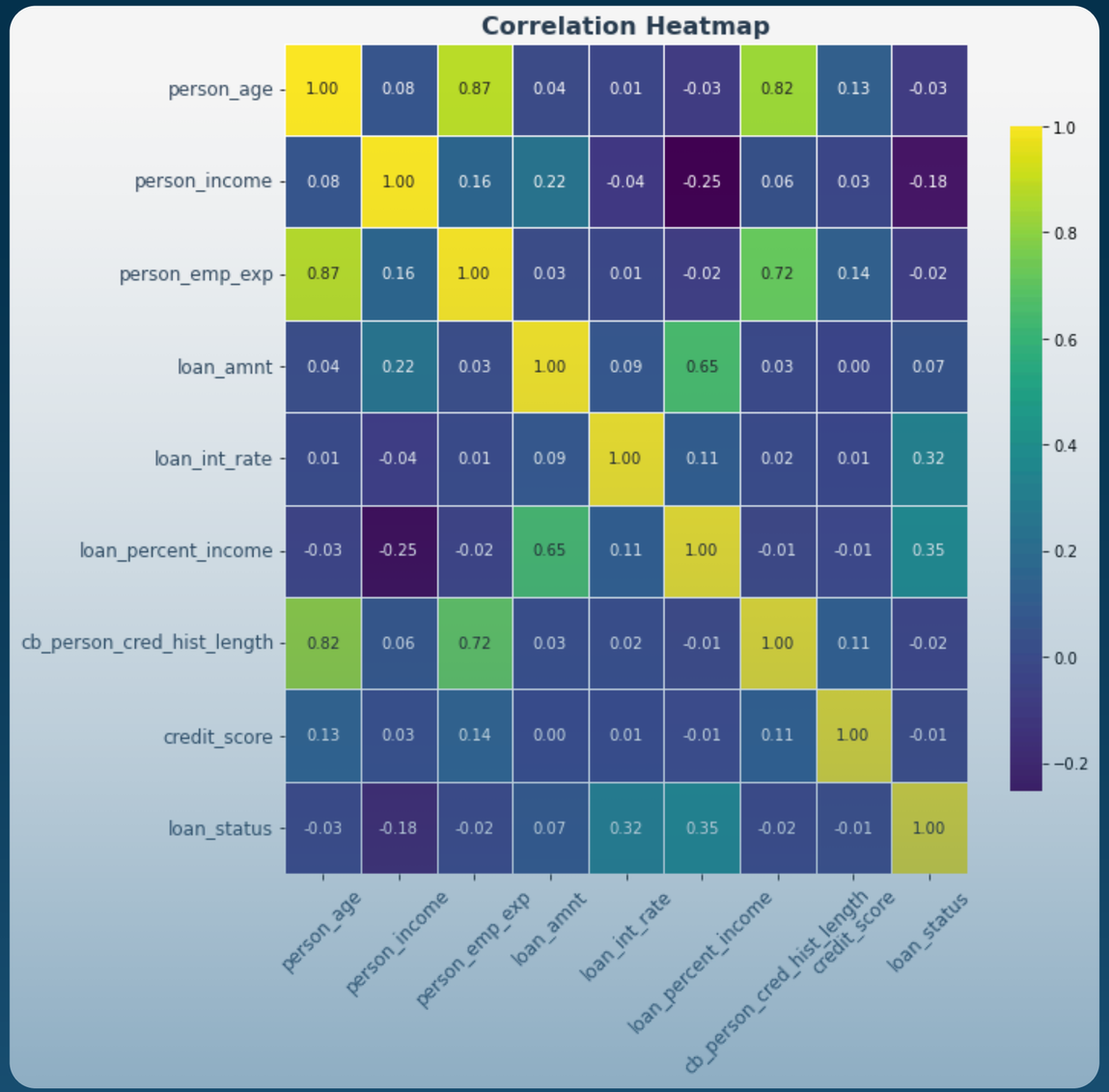
# CLEANED DATA OVERVIEW

- Converted:
  - Type of person’s age - float to integer
- Outlier Handling
- Originally: 45,000 entries
- Cleaned Shape: 37,549
- Outlier Shape: 7,451
- Sum of Cleaned & Outliers: 45,000

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45000 entries, 0 to 44999
Data columns (total 14 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   person_age                      45000 non-null  int64  
 1   person_gender                   45000 non-null  object 
 2   person_education                45000 non-null  object 
 3   person_income                   45000 non-null  float64
 4   person_emp_exp                  45000 non-null  int64  
 5   person_home_ownership           45000 non-null  object 
 6   loan_amnt                       45000 non-null  float64
 7   loan_intent                     45000 non-null  object 
 8   loan_int_rate                   45000 non-null  float64
 9   loan_percent_income             45000 non-null  float64
 10  cb_person_cred_hist_length      45000 non-null  float64
 11  credit_score                    45000 non-null  int64  
 12  previous_loan_defaults_on_file  45000 non-null  object 
 13  loan_status                     45000 non-null  int64  
dtypes: float64(5), int64(4), object(5)
memory usage: 4.8+ MB
```

# Variables Correlation

- Age, experience, and credit history are strongly linked (up to 0.87)
- Loan approval is moderately influenced by interest rate (0.32) and loan burden (0.35)
- Higher income reduces loan burden and slightly improves approval chances.



# Variables Correlation

| Variable Pair | Correlation | Strength | Insight |
|---|---|---|---|
| person_age & person_emp_exp | 0.87 | Strong Positive | Age and experience are highly related—important for employment features |
| loan_amnt & loan_percent_income | 0.65 | Strong Positive | Larger loans occupy more of applicant's income |
| loan_int_rate & loan_status | 0.32 | Moderate Positive | Higher interest loans are more often approved—possible risk pricing |
| loan_percent_income & loan_status | 0.35 | Moderate Positive | Approval is influenced by debt-to-income ratio |
| person_income & loan_percent_income | -0.25 | Moderate Negative | Higher incomes lead to lower income share for loans |

# Model Performance Metrics - Loan Approval Prediction (Classification)

| Metric | Meaning |
|---|---|
| Precision | of all People predicted as Defaulter/Non-Defaulter, how many were correct? |
| Recall | Of all actual Defaulter/Non-Defaulter, how many did we correctly identify? |
| F1 - Score | The balance between precision and recall (harmonic mean) |
| Support | The number of true instances for each class in the test set |

# Model Performance Loan Approval Prediction

## Logistic Regression

```
Logisitic Regression  train Score by score Func =  0.882833037457837
Logisitic Regression  test Score by score Func =  0.8834806688678241
Logisitic Regression  train Score by acc score Func =  0.882833037457837
Logisitic Regression  test Score by acc score Func =  0.8834806688678241
Model Classification Report

              precision    recall  f1-score   support

           0       0.93      0.92      0.93      7404
           1       0.72      0.73      0.73      1985

    accuracy                           0.88      9389
   macro avg       0.82      0.83      0.83      9389
weighted avg       0.88      0.88      0.88      9389
```

# Model Performance Loan Approval Prediction

## Random Forest

```
Random Forest  train Score by score Func =  0.9999644949405291
Random Forest  test Score by score Func =  0.9218234103738417
Random Forest  train Score by acc score Func =  0.9999644949405291
Random Forest  test Score by acc score Func =  0.9218234103738417
Model Classification Report

              precision    recall  f1-score   support

           0       0.94      0.96      0.95      7404
           1       0.85      0.77      0.81      1985

    accuracy                           0.92      9389
   macro avg       0.89      0.87      0.88      9389
weighted avg       0.92      0.92      0.92      9389
```

# Model Performance Loan Approval Prediction

## Decision Tree

```
Descion Tree  train Score by score Func =  1.0
Descion Tree  test Score by score Func =  0.892001278091383 5
Descion Tree  train Score by acc score Func =  1.0
Descion Tree  test Score by acc score Func =  0.892001278091383 5
Model Classification Report

              precision    recall  f1-score   support

           0       0.93      0.93      0.93      7404
           1       0.74      0.76      0.75      1985

    accuracy                           0.89      9389
   macro avg       0.84      0.84      0.84      9389
weighted avg       0.89      0.89      0.89      9389
```

# Model Performance Loan Approval Prediction

## XGBoost

```
XGBoost  train Score by score Func =  0.9713119119474525
XGBoost  test Score by score Func =  0.9300244967515178
XGBoost  train Score by acc score Func =  0.9713119119474525
XGBoost  test Score by acc score Func =  0.9300244967515178
Model Classification Report

              precision    recall  f1-score   support

           0       0.95      0.97      0.96      7404
           1       0.86      0.80      0.83      1985

    accuracy                           0.93      9389
   macro avg       0.90      0.88      0.89      9389
weighted avg       0.93      0.93      0.93      9389
```

# Model Performance - Loan Approval Prediction

## SVC (Support Vector Classifier)

```
SVC  train Score by score Func =  0.9071542694834014
SVC  test Score by score Func =  0.900628394930237 5
SVC  train Score by acc score Func =  0.9071542694834014
SVC  test Score by acc score Func =  0.900628394930237 5
Model Classification Report

              precision    recall  f1-score   support

           0       0.93      0.94      0.94      7404
           1       0.77      0.75      0.76      1985

    accuracy                           0.90      9389
   macro avg       0.85      0.85      0.85      9389
weighted avg       0.90      0.90      0.90      9389
```

# Model Performance - Interest Rate Prediction (Regression)

```
--- Loan Interest Rate Prediction ---
LinearRegression: RMSE = 2.81, R² Score = 0.0667
RandomForestRegressor: RMSE = 2.79, R² Score = 0.0807
DecisionTreeRegressor: RMSE = 3.98, R² Score = -0.8720
```

BEST MODEL: Random Forest, though predictive power is limited (low $R^2$).

WORST MODEL: Decision Tree — high error and negative $R^2$ indicate severe overfitting.

Linear Regression is interpretable but underperformed slightly compared to Random Forest.

*** Our best $R^2$ score was only 0.08, suggesting that the current dataset lacks key features needed to accurately predict interest rates.

# Analysis

- Effectively applies ML algorithms on financial data to predict approvals, defaults, and rates.
- Handles varied features well; some edge cases (e.g., unusual profiles) require better model generalization.
- Offers clean, interpretable results; further improvement possible with advanced tuning.
- Strong performance from XGBoost.
- Could benefit from real-time updates for evolving applicant behavior.

# Business Insights

- **Automated Lending Decisions:**

- **Risk Mitigation:**



- **Improved Fairness:** Models use objective financial variables instead of biases.


- **Scalability:** model framework can be extended to credit cards, mortgages, and refinancing solutions.

# Conclusion

- **High Predictive Accuracy:** XGBoost and Gradient Boosting models achieved strong results in both classification and regression, supporting reliable financial decision-making.
- **Comprehensive Modeling:** The system successfully predicts loan approvals, interest rates, and default probabilities from structured financial data.
- **Business Value:** The models offer significant improvements in decision efficiency, credit risk assessment, and customer targeting.
- **Future Expansion:** Incorporating real-time data streams, user-level profiles, and fairness audits will further strengthen the system's effectiveness and trust.
- **Deployment Potential:** The framework is lightweight and can be integrated into lending platforms for real-world use with minimal adjustments.

# Thank You!

# Q & A