1. Imported data Give me some credit(https://www.kaggle.com/competitions/GiveMeSomeCredit/data) from kaggle with features RevolvingUtilizationOfUnsecuredLines, age, NumberOfTime30-59DaysPastDueNotWorse, DebtRatio, MonthlyIncome, NumberOfOpenCreditLinesAndLoans, NumberOfTimes90DaysLate, NumberRealEstateLoansOrLines, NumberOfTime60-89DaysPastDueNotWorse, NumberOfDependents. Target Variable: SeriousDlqin2yrs(1 for defaulters, else 0; Data collected for 2 years). Numerical data;Rows: 150,000; Columns: 11.
2. Problems aimed to solve: Balanced data, optimum feature engineering with behavioral features, fixing age based credit lending disparity, identifying optimum parameters, balancing different evaluation parameters, giving user interpretable output.

A. Initial method- logistic regression(most basic for our aim) to understand the relationship of each feature with output and for comparison with other algorithms.

1. Manual initial screening of data to spot any visible patterns; Identified that debt ratio had a value present despite 0 or null monthly income
2. Used the describe function highest and lowest values to identify any out of range values.
3. Identified unusual range of RevolvingUtilizationOfUnsecuredLines, single digit values for MonthlyIncome, samples with age<18; NumberOfTimes90DaysLate, NumberOfTime60-89DaysPastDueNotWorse, NumberOfTime30-59DaysPastDueNotWorse with values>95(not practically possible in data collected for 2 years).
4. Searched for number of duplicate values and dropped them, checked for number of null values for features(present in MonthlyIncome, NumberOfDependents)
5. Checking data imbalance through defaulters percentage, visualizing it through histogram
6. Visualising range of values for different features, through box plot, categorised into good payers(0) defaulters(1) and to identify any extreme outliers. Identified extreme DebtRatio value, created a monthly income to debt ratio scatter plot and observed an L shaped pattern. Concluded debt ratio for 0/1/null monthly income is actually the debt of those samples.
7. Created a normal distribution graph for determining mean/median preferredness for null data.
8. For the median of all 3 NumberOfTimes(x)DaysLate data, we cap values at 95 so false outlier values don't skew the median, and use the resultant median to replace those outlier values.
9. To ensure that test data is mostly original not synthetic, we check if at least 20 percent of defaulters and 20 percent ‘good payers’ have data with no null values, if yes then we split those 20% only non-null samples for test data and fill null values in training set using median(for monthly income and number of dependents resp.), if not then we use median to fill incomplete values and split.
10. For users <18 age, replace age with median age for both test and train dataset.
11. For outliers of RevolvingUtilizationOfUnsecuredLines, we'll cap its value at 2, as every value above 2 will have the same real world consequences as 2.
12. Creating feature True_Monthly_Debt = DebtRatio for monthly income=0, and dropping DebtRatio to create Clean_Debt_Ratio =True_Monthly_Debt/monthly income(taking income as 1, when it’s 0;no non null values present)

13. Through heatmap generated in initial stages, summarised that the highest correlationship of target was with all NumberOfTimes(x)DaysLate data, and all number of days late feature had strong correlationship with one another. Hence we created a behavioural feature: Weighted_Late_Score=sum of all 3 NumberOfTimes(x)DaysLate data for respective samples; dropped individual NumberOfTimes(x)DaysLate data.
14. Created Critical_Risk_Index=['RevolvingUtilizationOfUnsecuredLines'] * ['Weighted_Late_Score']-shows risk of behavior of late payment; also exploits the behavior of maxing the tendency of an otherwise rarely shown behavior like late payment during stress of high credit use.
15. Total_Loans=sum of different loans taken
16. Real_Estate_Ratio = ratio of real estate loans to total loans. As mortgages are more stable investment in financial terms creating Stability feature.
17. Income_to_Age_Ratio feature and Young_Homeowner(1 when age<30 and Total loans>0, else 0) to clear age based fairness.
18. Disposable_Income= MonthlyIncome- True_Monthly_Debt
19. Passing the features, separating target and features for test and train dataset, scaling those target feature values.
20. Instead of smote, we'll use class weighing to ensure the model is trained on as more real, accurate data. Hence, since defaulter percentage is around 6.7%, each defaulter will be 93.3/6.7=~14 times more important
21. To find optimal threshold for our prediction, we'll use Youden's J Statistic to find widest gap between tpr and fpr
22. Display of confusion matrix, AUC, optimal threshold, precision, accuracy, recall, f1 score, confusion matrix
23. Checking age based fairness by using 4/5ths ratio, where it's fair when %young credited/%senior credited>=0.8. Here we define young applicant as <30.
24. Categorized risk into high, medium low, where high is risk>optimal threshold, medium risk threshold is 0.15, low is below 0.15(0.15 selected via banks' risk category range trends).
25. Display of SHAP model with two top features of Weighted_Late_Score and RevolvingUtilizationOfUnsecuredLines
26. Created another iteration of this code without age based feature optimising i.e no income to age ratio or young home owner.
27. Results:

FINAL MODEL PERFORMANCE(Selected model- has age fairness features)
AUC Score: 0.8557
Optimal Threshold: 0.4991
Accuracy: 0.7774
Precision: 0.1997
Recall: 0.7722
F1-Score: 0.3174
AGE BIAS AUDIT
Young Approval Rate (<30): 53.11%
Senior Approval Rate (30+): 75.43%
Disparate Impact Ratio: 0.7041

FINAL MODEL PERFORMANCE(For comparison-No age fairness features)

AUC Score: 0.8557
Optimal Threshold: 0.4794
Accuracy: 0.7646
Precision: 0.1925
Recall: 0.7867
F1-Score: 0.3093
AGE BIAS AUDIT
Young Approval Rate (<30): 51.99%
Senior Approval Rate (30+): 73.93%
Disparate Impact Ratio: 0.7032

28. Both models have similar evaluating parameter values, while 1 has more features and the age based features don't tweak the age fairness results significantly, we still choose it because of slightly higher F1- score and young approval rate.
29. Note:These two final comparison models are developed after multiple iterations of deciding new features, changing sequence of methodology, changing techniques and functions used.

B. We decided to use XGBoost because it can produce results by internally managing imbalanced and null data. Features with correlationship value zero presented different impacts in SHAP, indicating non-linear relationship. XGBoost, is ideal for this datatype where priority of features matter, giving better evaluation parameter values and we further develop the model by adding synthetic features.

1. Note: We repeated the same steps from logistic regression for data analysing,preprocessing, deleting duplicate data. We used an iteration where test data is collected such that its data is considered 'clean' for age>18, no outlier values of NumberOfTimes(x)DaysLate and no null values. We use the similar splitting technique as in logistic regression to take this clean data for defaulter and good payer in the test set. The output however gave comparatively lower recall value than the finalized model hence was rejected.
2. We also tried an iteration where test data is collected such that age and NumberOfTimes(x)DaysLate outlier values aren't changed, but null values are filled. It gave us close results but the finalized model was preferred due to high AUC.
3. For the finalized model, we repeated steps until for data analysing, deleting duplicates, data cleaning of outliers, splitting and removing null data, pre-processing to clear the noise by giving right values for DebtRatio and RevolvingUtilizationOfUnsecuredLines .
4. Removed age fairness features as it gave similar results to one with age fairness features, but actually gave slightly better Disparate impact rate.
5. To solve the problem of choosing ideal parameter values to get best results, we used Optuna Bayesian for tuning parameters within an ideal range e.g. max_depth with range 3 to 7.
6. Evaluation metrics of AUR ROC curve, auc score, precision, recall, accuracy, fpr and f1 score, heatmap, SHAP created; age bias auditor and risk categorisation done
7. Created metric Economic impact to hypothetically assess profit/loss for lenders on the basis of performance of the final model on a test set. For a sample, loan value taken is taken as 3 times monthly income. If a real estate loan is present then the

recovery rate is 50%, else 10%. Interest rate is defined 15%, and on basis of tp,fp,tn,fn value, profit loss per individual is calculated and summed up

8. O/p: Top SHAP features: Revolutionofunsecuredlines and critical risk index.

FULL MODEL EVALUATION
AUC Score: 0.8633
Accuracy: 0.7697
Precision: 0.1986
Recall (TPR): 0.8027
F1-Score: 0.3184
False Alarm Rate (FPR):0.2326

ECONOMIC IMPACT
TOTAL PROJECTED PROFIT: $44,672,429.25
PROFIT PER APPLICANT: $1,495.11

BIAS AUDIT
Young Approval Rate: 53.72%
Senior Approval Rate: 74.14%
Disparate Impact: 0.7246

C. Conclusion: We'll prefer the XGBoost model due to better evaluation parameter values. However, the age bias problem hasn't been fixed. Hence, we can further optimise this model or we can use a more comprehensive dataset through which more age bias removing features can be engineered.
Possible minor improvements:using knn for null data, finding more ideal parameters to define 'young age', medium risk threshold. Finding better ways to calculate required loan and likelihood of payment for profit/loss calculation.