CREDIT CARD DEFAULT PREDICTION

Project Report

Author

Aridaman Singh

(Enrollment no. 22112020)

Date

16 June 2025

1. Problem Statement / Objective

This project aims to predict whether a credit card customer will default in the upcoming month based on historical behavioral and demographic data. Accurately identifying potential defaulters is critical for risk management and financial planning in lending institutions like banks.

2. Dataset Description

- Source: Dataset loaded from Google Drive via Jupyter Notebook (provided in the problem statement by Finance Club, IITR)
- Rows: 25,247 records.
- Columns: 27 original features including demographics (age, sex, education, marriage), payment history, billing amounts, and the target variable (`next_month_default`).

3. Data Cleaning & Preprocessing

- There are 126 null values in age column, otherwise no null values. Nulls in age column were replaced with mean age and round it to nearest Integer.
- Checked the categorical variables, if there is any encoding other than expected. Marriage should have:
  1. 1 for married
  2. 2 for single
  3. And 3 for others.

  Similarly, education should have 1, 2, 3 and 4. But marriage contained 0's and education contained 0, 5 and 6.

- First, clipped `marriage` and `education` values to ensure they don't exceed 3 and 4 respectively, keeping them within expected category ranges.
- Then, checked and counted invalid values (`>3`, `>4`, and `0`) in both columns to assess data quality.
- Identified that `0` is not a valid category for either column, so counted and prepared to remove those entries.
- Finally, dropped all rows where `marriage` or `education` had a value of `0` using `.isin()` to retain only valid categories.

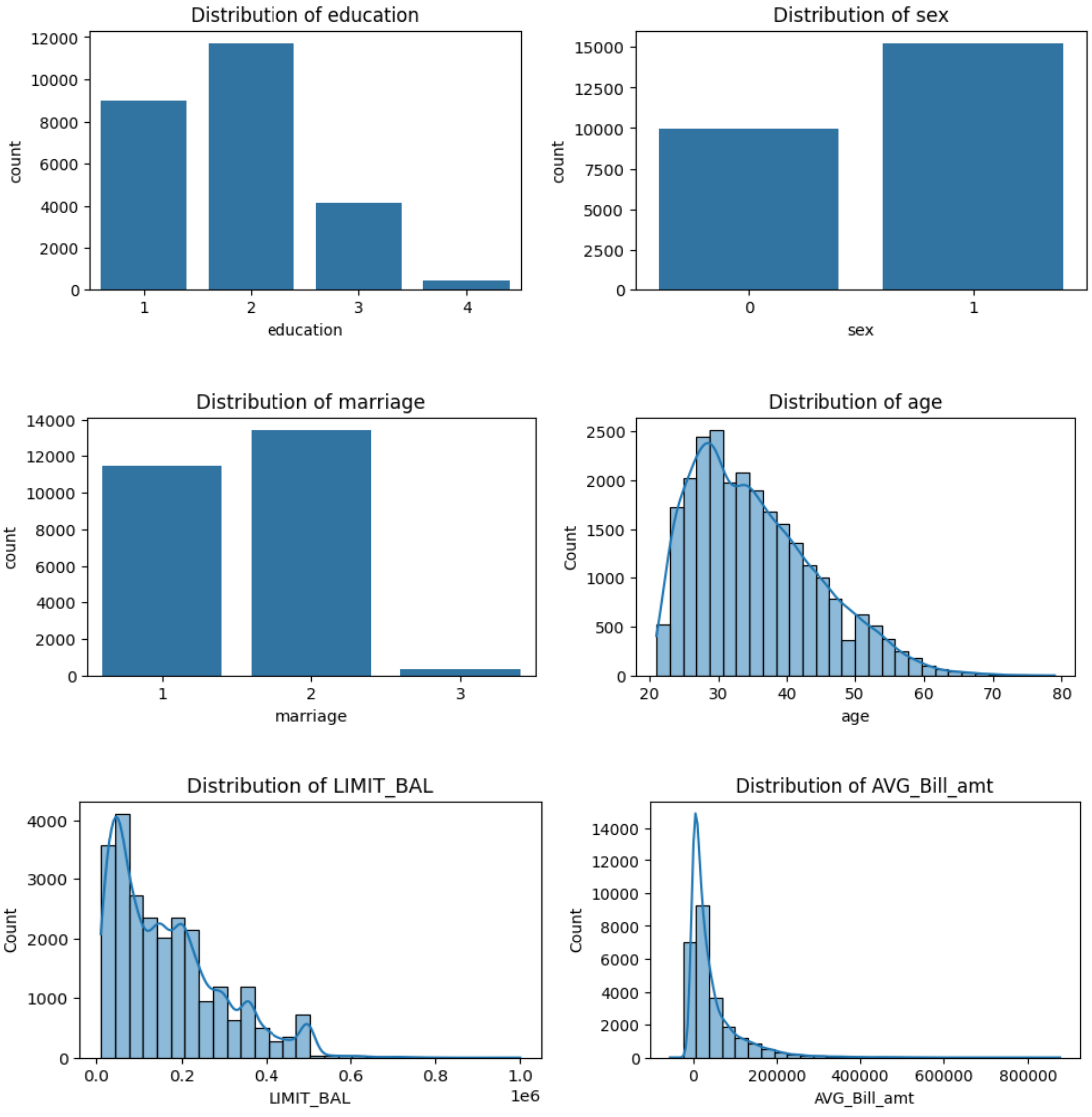
- Capped all pay_m columns (pay_0, pay_2 to pay_6) so that any value greater than 1 was clipped to 1, standardizing repayment status and change pay_0 to pay_1.

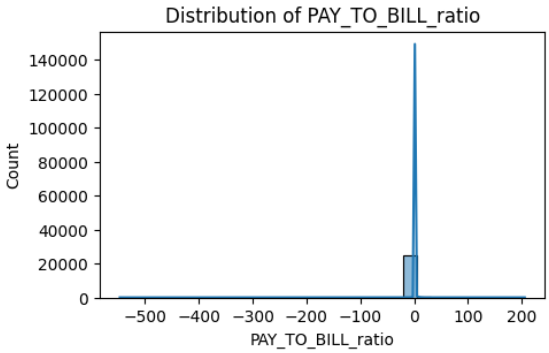
- The columns AVG_bill_amt and PAY_TO_BILL_ratio have errors. They are not calculated properly. No value in pay_amt 's is negative but avg bill amounts are negative. Also this column has values which do not match with the average of bill amounts. That's why it is made again as avg_bill_amt with calculated averages and AVG_BILL_amt was dropped. Similarly, new pay to bill ratio was made according to new averages.

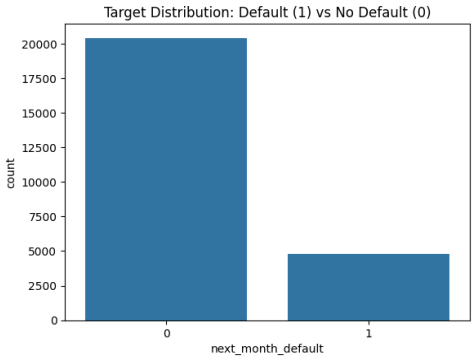
4. Exploratory Data Analysis (EDA)

- Univariate and bivariate analyses were performed.
- Histograms and count plots for distributions.
- Heatmap revealed moderate correlation among payment history variables.

Univariate Analyses graphs and results:

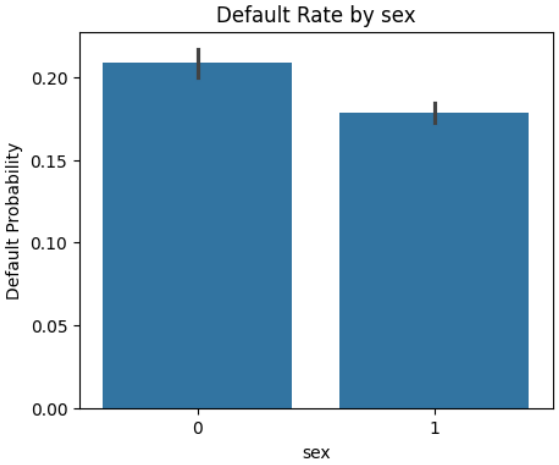


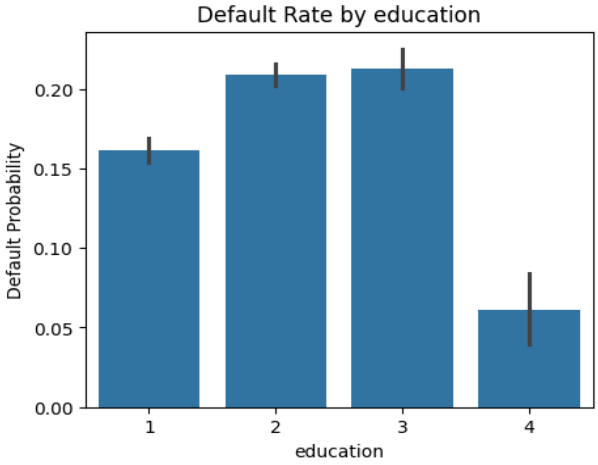


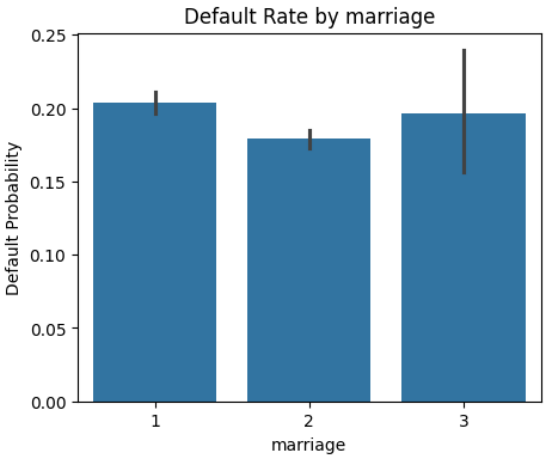


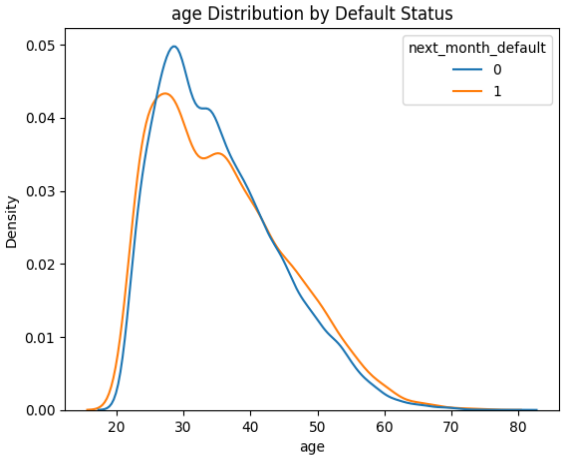
- It is clear from the graph that most of the people in the dataset are male, university level educated, single and have a mode age of 30.

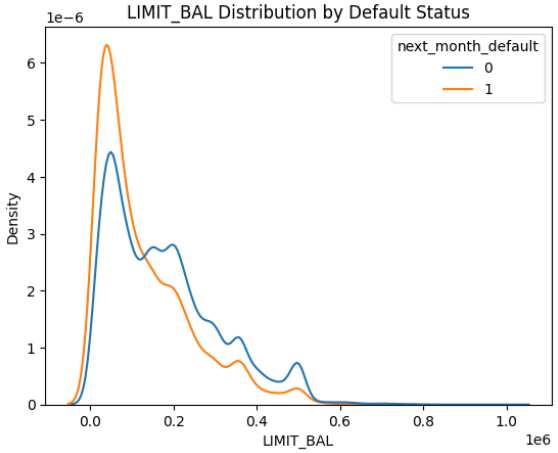
Bivariate Analyses graphs and results:

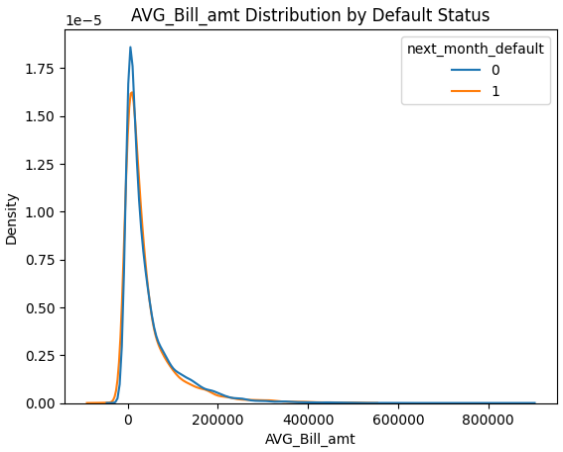






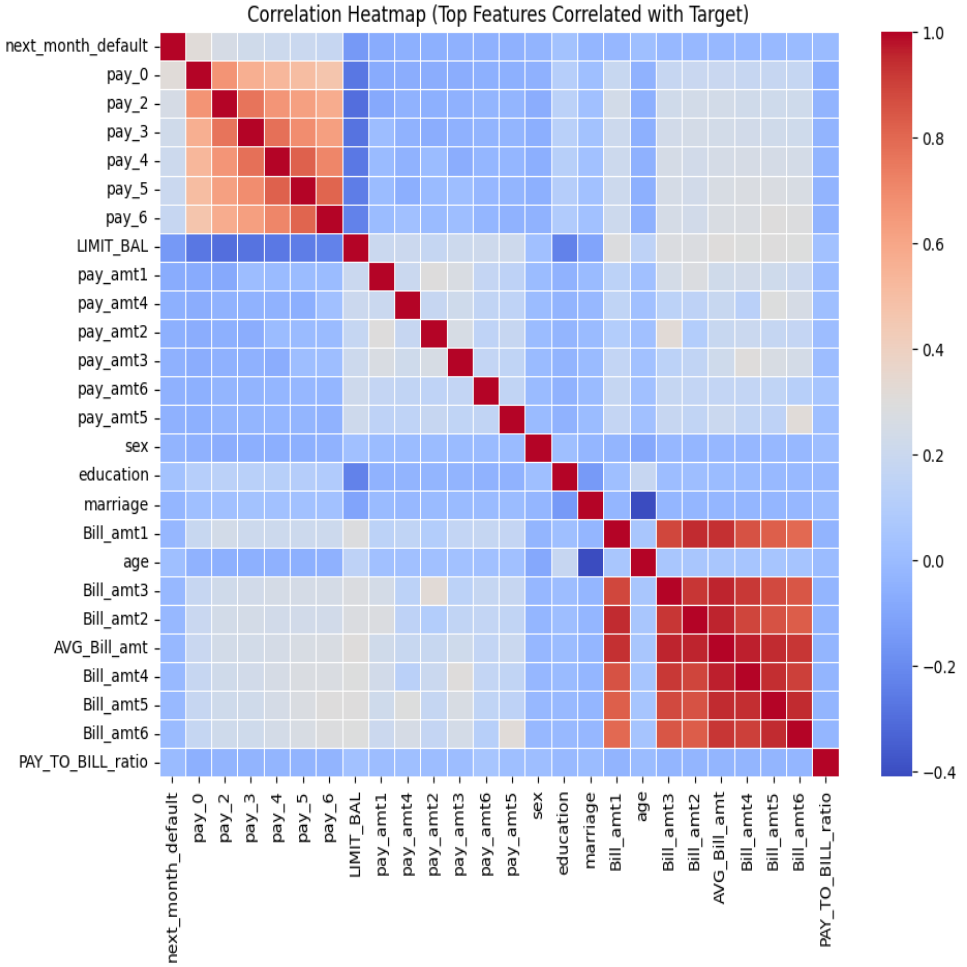






Correlation of variables with target:

next_month_default	1.000000
pay_0	0.236910
pay_2	0.187224
pay_3	0.169038
pay_4	0.148773
pay_5	0.137135
pay_6	0.125127
LIMIT_BAL	-0.146217
pay_amt1	-0.068823
pay_amt4	-0.053714
pay_amt2	-0.053643
pay_amt3	-0.050888
pay_amt6	-0.047261
pay_amt5	-0.047123
sex	-0.037383
education	0.034077
marriage	-0.027706
Bill_amt1	-0.022131
age	0.018239
Bill_amt3	-0.017066
Bill_amt2	-0.016413
avg_bill_amt	-0.015059



Bill_amt4 -0.012377

Bill_amt5 -0.009352

Bill_amt6 -0.006109

pay_to_bill_ratio_fixed -0.003248

- The correlation coefficients of all the features with the target variable are weak. None of the features gives a clear dependence for next month default.
- Also, other graphs for categorical and numerical variables don't show any significant trend. One thing is that females have more default rate. University and Graduate level have high defaults.

5. Feature Engineering

Created features related to:
1. Repayment behavior

- Max delay in any month: It is the maximum of the delays (pay_0, pay_2, etc.)
- n_delinquent_months : Count of delay months
- avg_delay : average delay over delinquent months
- longest_streak : Longest streak of consecutive overdue months

2. Consistency & discipline

- pay_std : standard deviation of payment status > inconsistency
- n_no_usage : number of months with no usage (pay_m == -2)
- prop_delinquent : proportion of delinquent month

3. Utilization & affordability

- credit_utilization : avg bill / limit
- pay_to_limit : pay to limit ratio = total payment / limit

6. Feature Encoding & Selection

Applied **one-hot encoding to categorical variables**. All engineered features retained after importance analysis.

7. Handling Imbalance

The dataset is imbalanced:

Next Month Default	Count
0	20377
1	4803

Next month default true values are very less (19%) compared to false values. This is handles using SMOTE (Synthetic Minority Oversampling Technique). SMOTE was applied on training data (X_train, y_train). SMOTE made the distribution of 0 and 1 equal in the training split.

Before SMOTE:

0	**16302**
1	3842

After SMOTE:

0	**16302**
1	16302

8. Model Building

Models used:

- Logistic Regression
- Decision Tree
- Adaboost
- Random Forest
- XGBoost
- LightGBM
- KNN

Train-Test split of 80-20 was used. Then, Hyperparameter tuning was done for all the abovementioned models using GridSearchCV. The grid is in the project notebook.

- Scoring for tuning is done according to F2 score. That means we selected those hyperparameters which gave the most F2 Score.
- $F2\ Score = (1 + 2^2)\frac{Precision.Recall}{4Precision + Recall}$
- F2 Score gives more importance to Recall. Recall is sensitive to False negative predictions. It is important in this project as a bank or any lending institution does not want to misclassify any Default customer. This comes in the risk management of the institutions. Recall is given importance due to the risk of leaving a default customer.

Following tabulates the best parameters:

Logistic Regression	**'C': 0.1, 'max_iter': 200, 'penalty': 'l1'**
Decision Tree	{'criterion': 'entropy', 'max_depth': 30, 'min_samples_leaf': 1, 'min_samples_split': 2}
Random Forest	{'max_depth': None, 'max_features': 'log2', 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 200}
Adaboost	{'algorithm': 'SAMME', 'learning_rate': 0.01, 'n_estimators': 50}
XGBoost	{'colsample_bytree': 1.0, 'learning_rate': 0.2, 'max_depth': 7, 'n_estimators': 200, 'subsample': 0.8}
Light GBM	{'learning_rate': 0.2, 'max_depth': 15, 'n_estimators': 200, 'num_leaves': 100, 'subsample': 0.8}

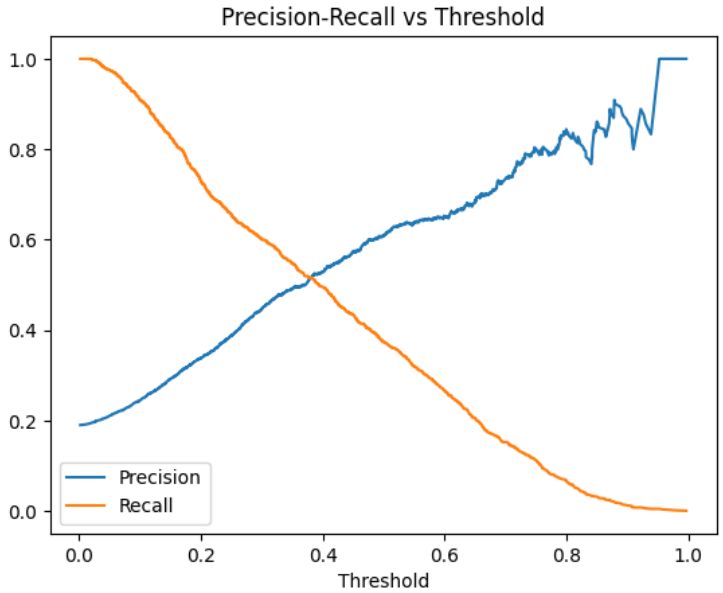
KNN	{'metric': 'manhattan', 'n_neighbors': 3, 'weights': 'distance'}

9. Model Evaluation

All models were evaluated on the basis of ScikitLearn's Classification report, train and test accuracy, ROC-AUC score and F2 score. Results are tabulated below:

Model	Train Accuracy	Test accuracy	ROC-AUC score	F2 score
Logistic Regression	0.8107	0.8092	0.7224	0.4324
Decision Tree	0.9998	0.7268	0.5977	0.3734
Adaboost	0.7788	0.7387	0.7089	0.5805
Random Forest	1.0000	0.8284	0.7726	0.3772
XGBoost	0.9767	0.8235	0.7473	0.3713
LightGBM	0.9988	0.8195	0.7458	0.3395
KNN	1.0000	0.6287	0.5739	0.3837

- It is clear that the ensembled models are performing better.
- Decision Tree ,KNN,Lightgbm and Random Forest are overfitted, but still random forest has good results.
- The accuracy and F2 score of all the models are low. It is not satisfactory. This is the consequence of the low correlation coefficients of all the features.
- Threshold probability tuning of ensembled models is done to increase F2 score. Following graph illustrates the dependence of Precision and Recall on threshold.



- Since, F2 score depends more on recall, lower threshold is going to increase it. Also, lower threshold will decrease False negatives.

10. Model Comparison

- **Random Forest**
  Best F2-Score: 0.6022
  Accuracy at best threshold: 0.6338
  Threshold: 0.20

- **XGBoost**
  Best F2-Score: 0.5750
  Accuracy at best threshold: 0.6072
  Threshold: 0.1

- **LightGBM**
  Best F2-Score: 0.5663
  Accuracy at best threshold: 0.6924
  Threshold: 0.11

- **AdaBoost**
  Best F2-Score: 0.5805
  Accuracy at best threshold: 0.7387
  Threshold: 0.12

We will use Random Forest as final model for future predictions and uses **Youden's J statistic** (TPR - FPR) to find the optimal classification threshold for a Random Forest model.
It computes the **ROC curve** on the validation data to obtain TPR, FPR, and thresholds.
The threshold with the **maximum J value** is selected, balancing sensitivity and specificity for best separation.

11. Application of Validation dataset

We used the trained random forest on provided validation dataset to do predictions. The predictions csv is also attached. The percentage of predicted defaults in validation csv were 10.71%.

12. Financial Analysis, Business Insights & Recommendations

Business Insights

1. Risk Assessment

- Payment delays (especially pay_0) are strong predictors of default risk
- Customers with lower credit limits may be higher risk
- Younger customers (age 24-44) appear more frequently in the data

2. Customer Segmentation

- Could segment customers by:
  - Credit limit ranges
  - Payment behavior patterns
  - Demographic groups
  - Default risk levels

3. Financial Health Indicators

- Pay to bill ratio shows what portion of bill is being paid
- avg_bill_amt gives average spending patterns

Recommendations

1. Risk Management

1. **Early Warning System**: Monitor customers showing payment delays (especially pay_0 ≥ 1)
2. **Credit Limit Adjustments**: Review limits for customers consistently showing high utilization
3. **Targeted Interventions**: Focus on higher-risk segments (younger, lower credit limits, payment delays)

2. Customer Engagement

1. **Payment Reminders**: For customers showing first signs of delinquency
2. **Financial Education**: For younger customers and those with irregular payment patterns
3. **Customized Offers**: For reliable payers (increased limits, rewards)

3. Data-Driven Improvements

1. **Feature Engineering**: Create more predictive features from existing data
    - Payment delay trends over time
    - Utilization ratios
    - Spending volatility measures
2. **Predictive Modelling**: Build default prediction models using this rich dataset
3. **A/B Testing**: Test different intervention strategies on customer segments

4. Portfolio Management

1. **Risk-Based Pricing**: Adjust interest rates based on customer risk profiles
2. **Portfolio Diversification**: Balance high-risk and low-risk customers
3. **Reserve Planning**: Better estimate potential defaults for financial planning

13. Conclusion

Data was cleaned, features were engineered, models were tested. Results are promising but need further validation.

14. Appendices

Code in Jupyter Notebook.

Libraries: pandas, numpy, matplotlib, seaborn, sklearn, xgboost, lightgbm, joblib.