Predicting loan defaults

Lenka Rozborilova
11/03/2021

Aim of the project:

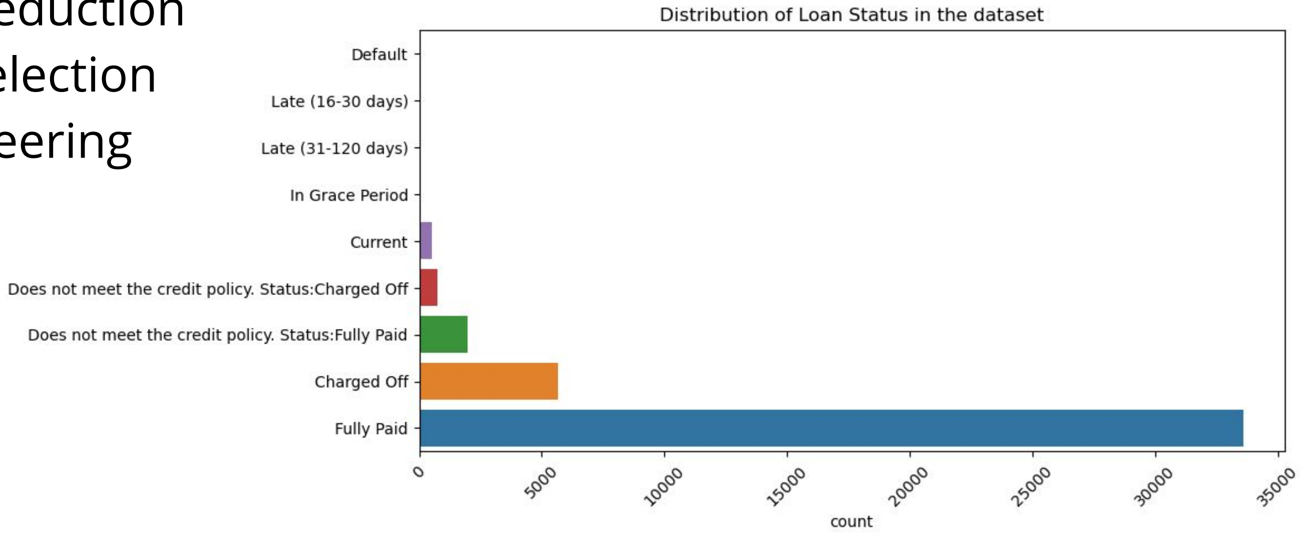
- develop a **machine learning model** that predicts whether an applicant is likely to pay a loan back or fall into default (= binary classifier)
- client: Lending Club online loan provider, P2P lending platform
- aid their understanding and risk assessment

Data set introduction

- loan applications made through Lending Club platform during a period of 5 years
- 43k loans (past and current)
- data exploration:
    - amount of loan issued,
    - interest rate,
    - instalment,
    - FICO credit scores,
    - number of past-due delinquencies,
    - zip code address,
    - length of employment...
- data processing and modeling done in python™

Data processing

- data cleaning
- setting up target variable: Loan Status
- observation reduction
- feature pre-selection
- feature engineering



Train/test split

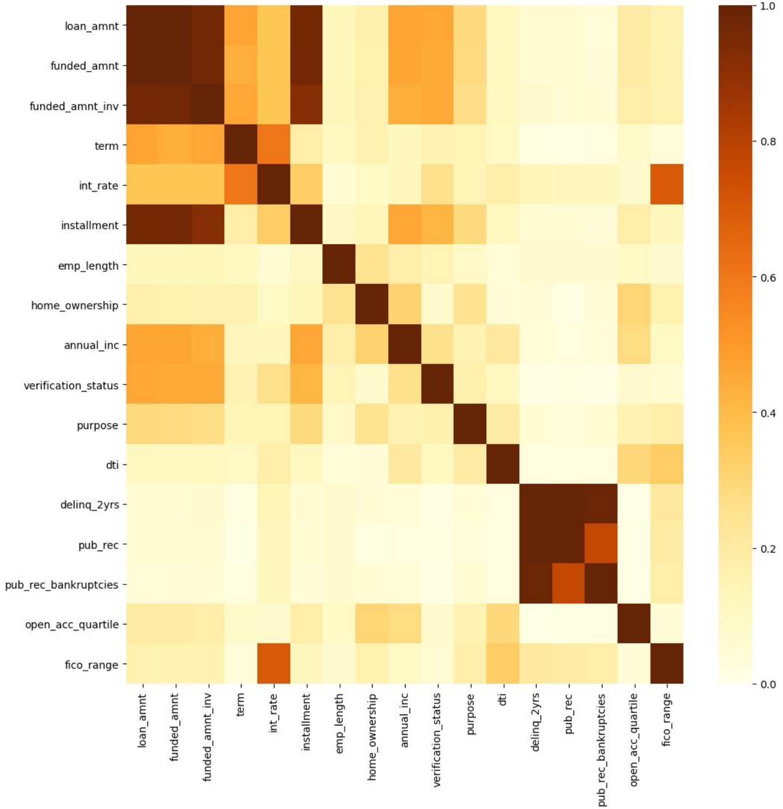
- conventional 80 to 20 random split
- why is it important?
    - avoid bias
    - avoid overfitting
    - accuracy
    - pick the best model
- imbalance in target variable:

```
Fully Paid      32180
Charged Off      5305
```

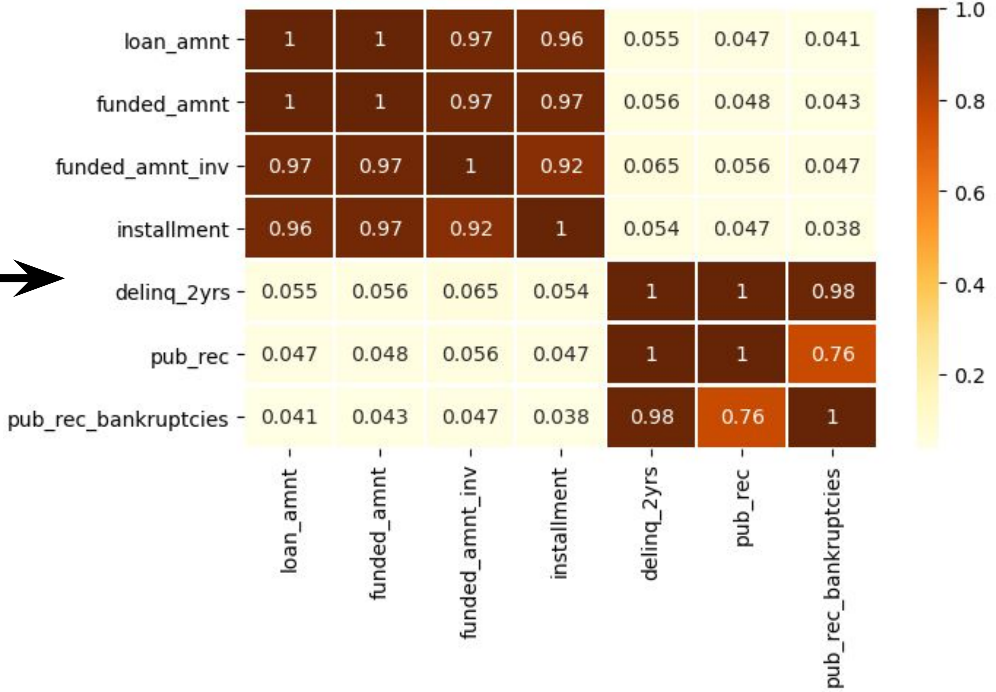
Model assumptions

- independence of two features
- presence of relationship between features and target
- presence of variance
- features normally distributed
- for regression model features also have to be scaled

Correlation between features

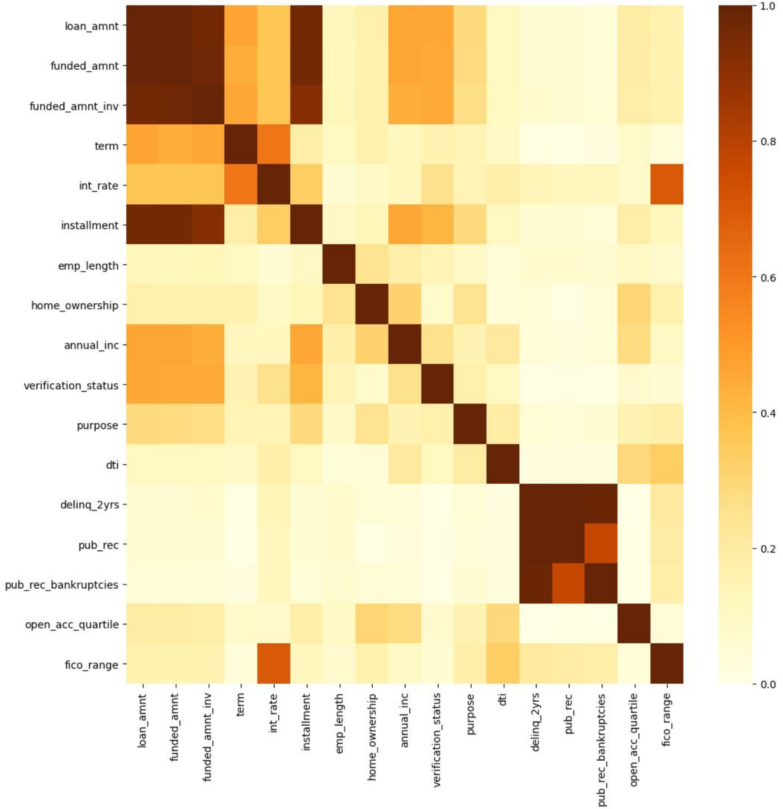


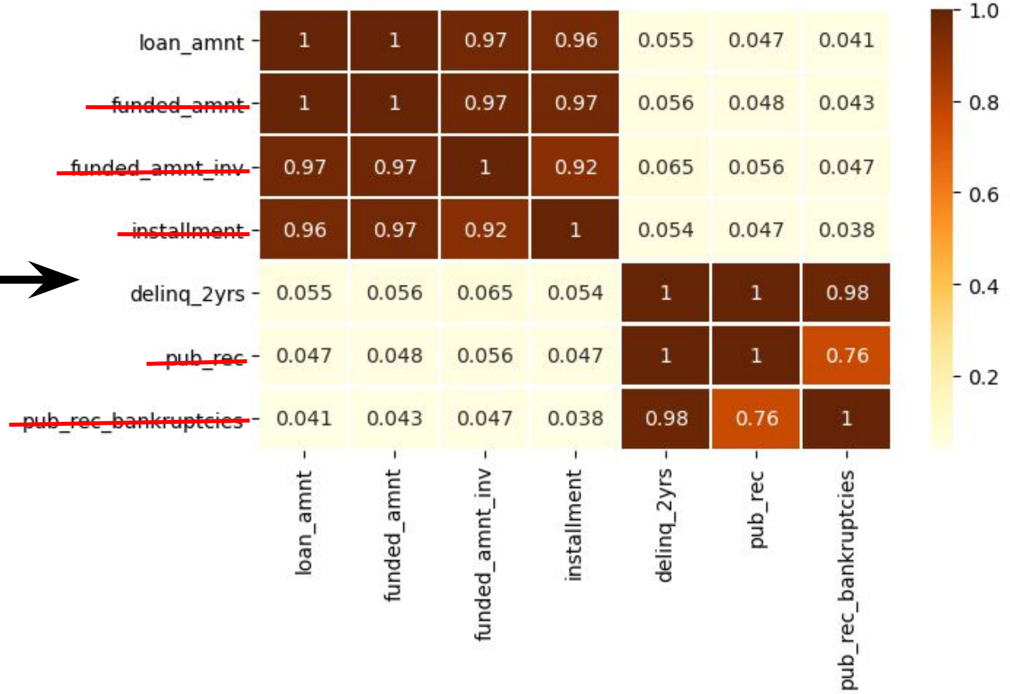
→



	loan_amnt	funded_amnt	funded_amnt_inv	installment	delinq_2yrs	pub_rec	pub_rec_bankruptcies
loan_amnt	1	1	0.97	0.96	0.055	0.047	0.041
funded_amnt	1	1	0.97	0.97	0.056	0.048	0.043
funded_amnt_inv	0.97	0.97	1	0.92	0.065	0.056	0.047
installment	0.96	0.97	0.92	1	0.054	0.047	0.038
delinq_2yrs	0.055	0.056	0.065	0.054	1	1	0.98
pub_rec	0.047	0.048	0.056	0.047	1	1	0.76
pub_rec_bankruptcies	0.041	0.043	0.047	0.038	0.98	0.76	1

Correlation between features > 0.9





Correlation between features and target

	index	corr_with_target
12	loan_status	1.00
2	int_rate	0.27
1	term	0.25
11	fico_range	0.17
7	purpose	0.12
5	annual_inc	0.08
0	loan_amnt	0.08
9	delinq_2yrs	0.07
8	dti	0.06
4	home_ownership	0.04
6	verification_status	0.03
3	emp_length	0.02
10	open_acc_quartile	0.00

Correlation between features and target≅0

	index	corr_with_target
12	loan_status	1.00
2	int_rate	0.27
1	term	0.25
11	fico_range	0.17
7	purpose	0.12
5	annual_inc	0.08
0	loan_amnt	0.08
9	delinq_2yrs	0.07
8	dti	0.06
4	home_ownership	0.04
6	verification_status	0.03
3	emp_length	0.02
10	open_acc_quartile	0.00

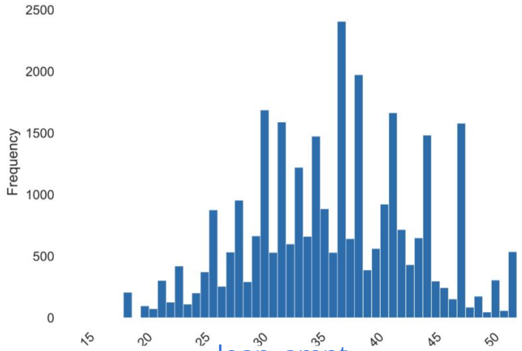
Variation

```
loan_amnt          51.28
int_rate            1.25
annual_inc          0.06
dti                44.03
policy_code         0.00
acc_now_delinq      0.00
tax_liens           0.00
fico_range          0.00
```

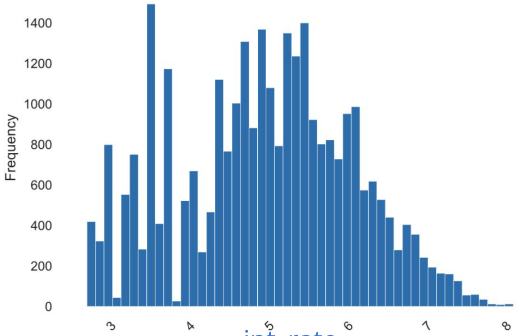
Variation = 0

```
loan_amnt          51.28
int_rate            1.25
annual_inc          0.06
dti                44.03
policy_code         0.00
acc_now_delinq      0.00
tax_liens           0.00
fico_range          0.00
```

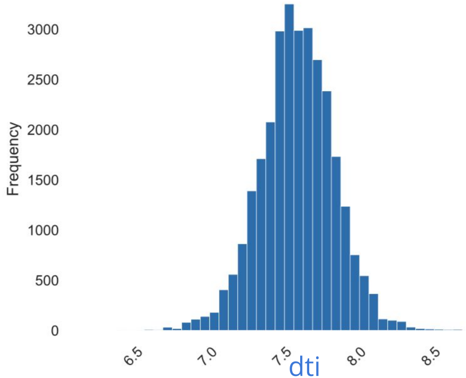
Features normally distributed



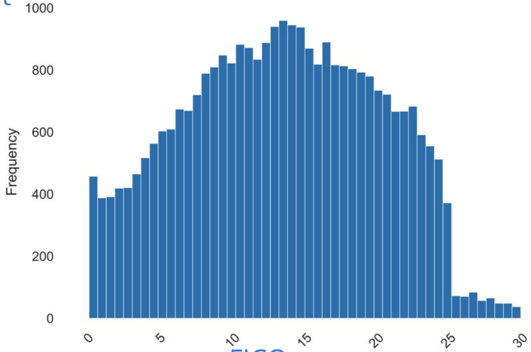
loan_amnt



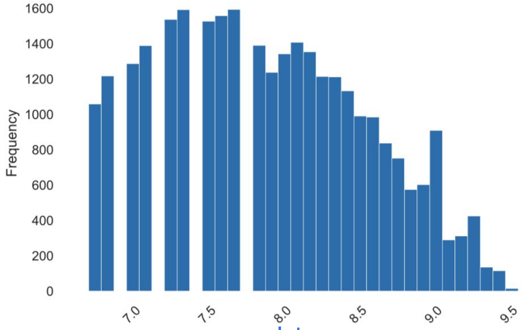
int_rate



dti



FICO_range



annual_inc

Standardization

- Standard Scaler
- Robust Scaler

Dummying

- clean dataset:

Dataset statistics

Number of variables	11
Number of observations	29988
Missing cells	0

Variable types

Numeric	5
Categorical	6

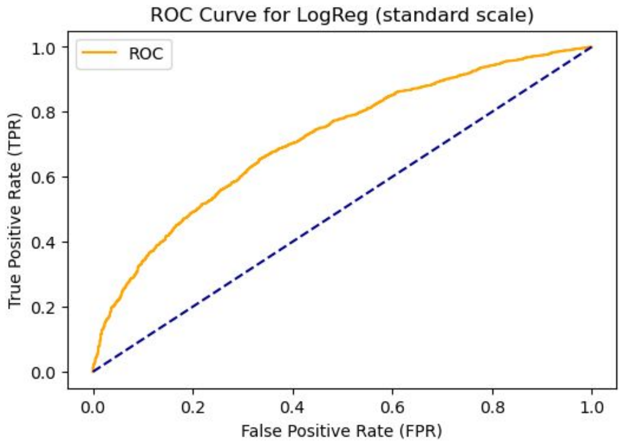
- models work with numerical variables -> dummy categorical variables

Binary classifiers

1. Logistic Regression
2. Naive Bayes
3. Random Forest

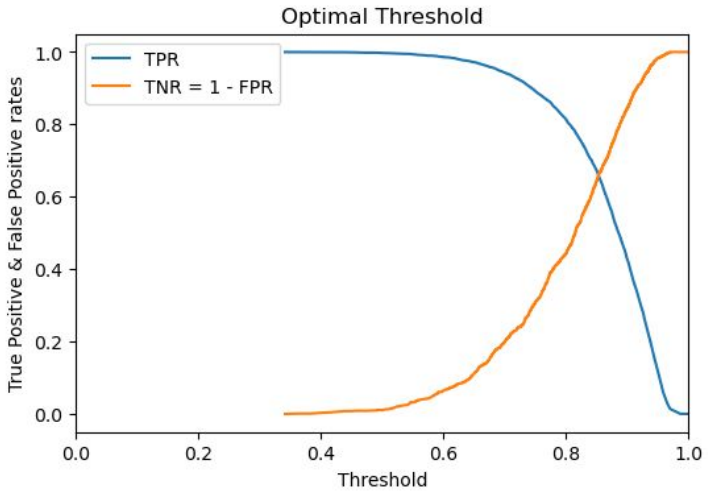
1. Logistic Regression

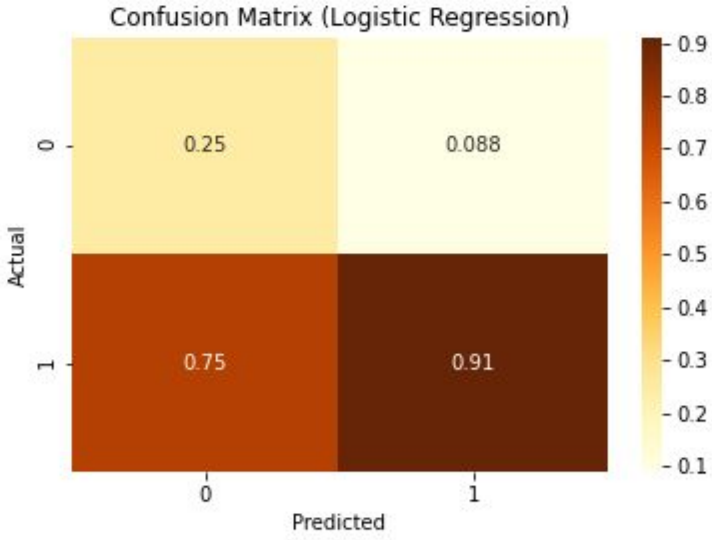
```
Logistic Regression - Standard Scaler:

 Mean: -7.287646696033442e-07

 Standard Deviation: 0.9999999999997343

 Train Score: 0.8584767240229425

 Model Accuracy Score (i.e. trained model applied on test data set): 0.858343337334934

 AUC Score: 0.7101766974752627

 Confusion Matrix:
 [[  10 1056]
 [   6 6425]]

 Classification report:
               precision    recall  f1-score   support

           0       0.62      0.01      0.02      1066
           1       0.86      1.00      0.92      6431

    accuracy                           0.86      7497
   macro avg       0.74      0.50      0.47      7497
weighted avg       0.83      0.86      0.79      7497


 Classification Predictions: [1 1 1 ... 1 1 1]
```



Optimal threshold

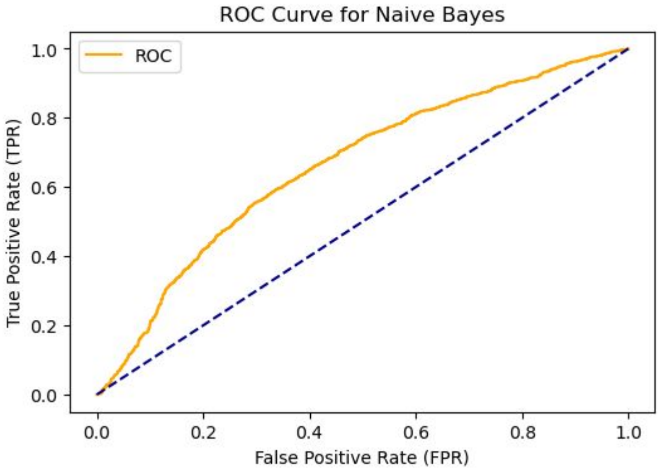
Optimal threshold for the Logistic Regression binary classification: 0.8491730850940469





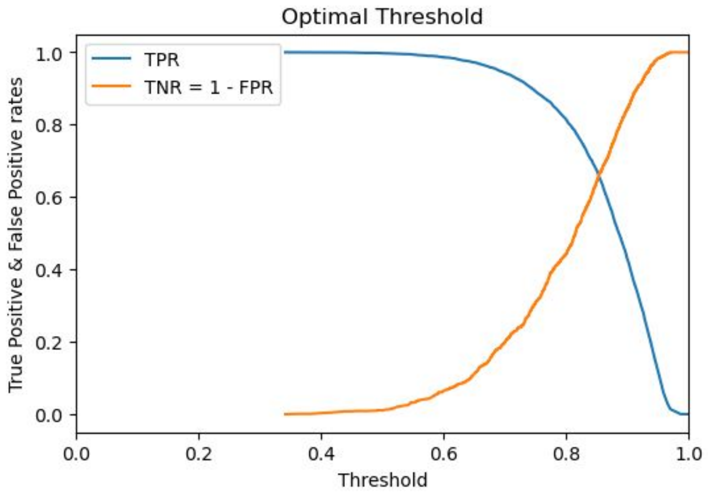
2. Naive Bayes

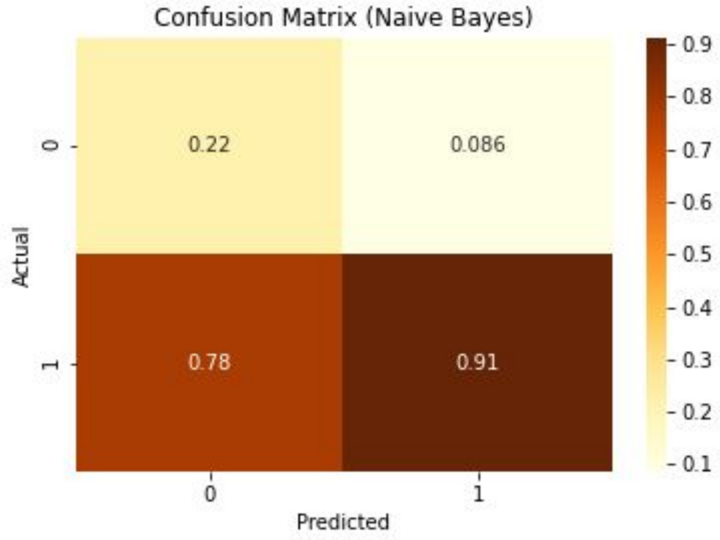
```
Naive Bayes:

 Train Score: 0.7874149659863946

 Model Accuracy Score (i.e. trained model applied on test data set): 0.7925837001467254

 AUC Score: 0.6612715642425016

 Confusion Matrix:
 [[ 275  791]
 [ 764 5667]]

 Classification report:
               precision    recall  f1-score   support

           0       0.26      0.26      0.26      1066
           1       0.88      0.88      0.88      6431

    accuracy                           0.79      7497
   macro avg       0.57      0.57      0.57      7497
weighted avg       0.79      0.79      0.79      7497


 Classification Predictions: [1 1 1 ... 1 0 1]
```



Optimal threshold

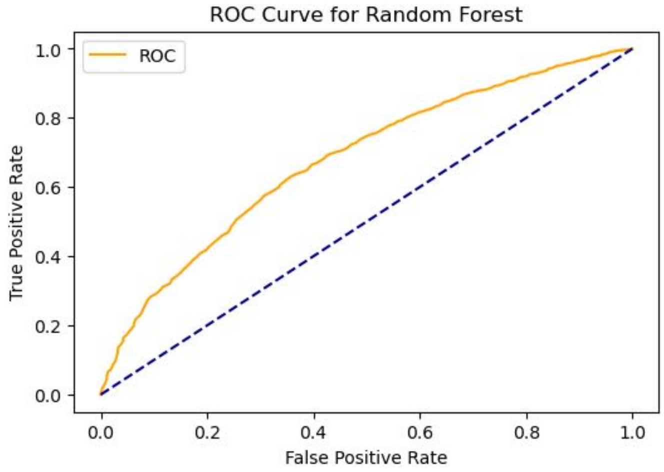
Optimal threshold for the Gaussian Naive Bayes binary classification: 0.9115128942158439





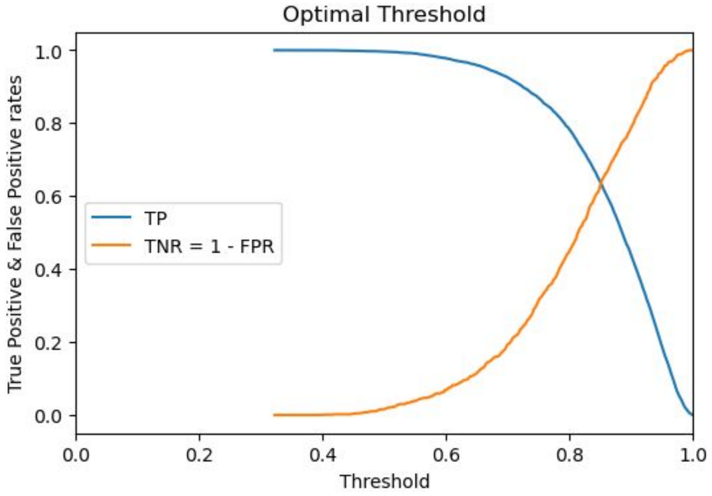
3. Random Forest

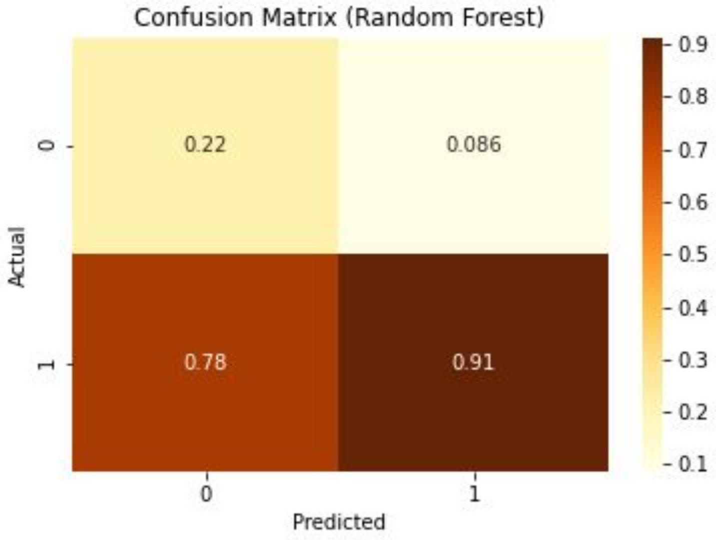
```
Random Forest:

 Train Score: 1.0

 Model Accuracy Score (i.e. trained model applied on test data set): 0.8572762438308656

 AUC Score: 0.6788496328320579

 Confusion Matrix:
 [[  20 1046]
 [  24 6407]]

 Classification report:
               precision    recall  f1-score   support

           0       0.45      0.02      0.04      1066
           1       0.86      1.00      0.92      6431

    accuracy                           0.86      7497
   macro avg       0.66      0.51      0.48      7497
weighted avg       0.80      0.86      0.80      7497


 Classification Predictions: [1 1 1 ... 1 1 1]
```



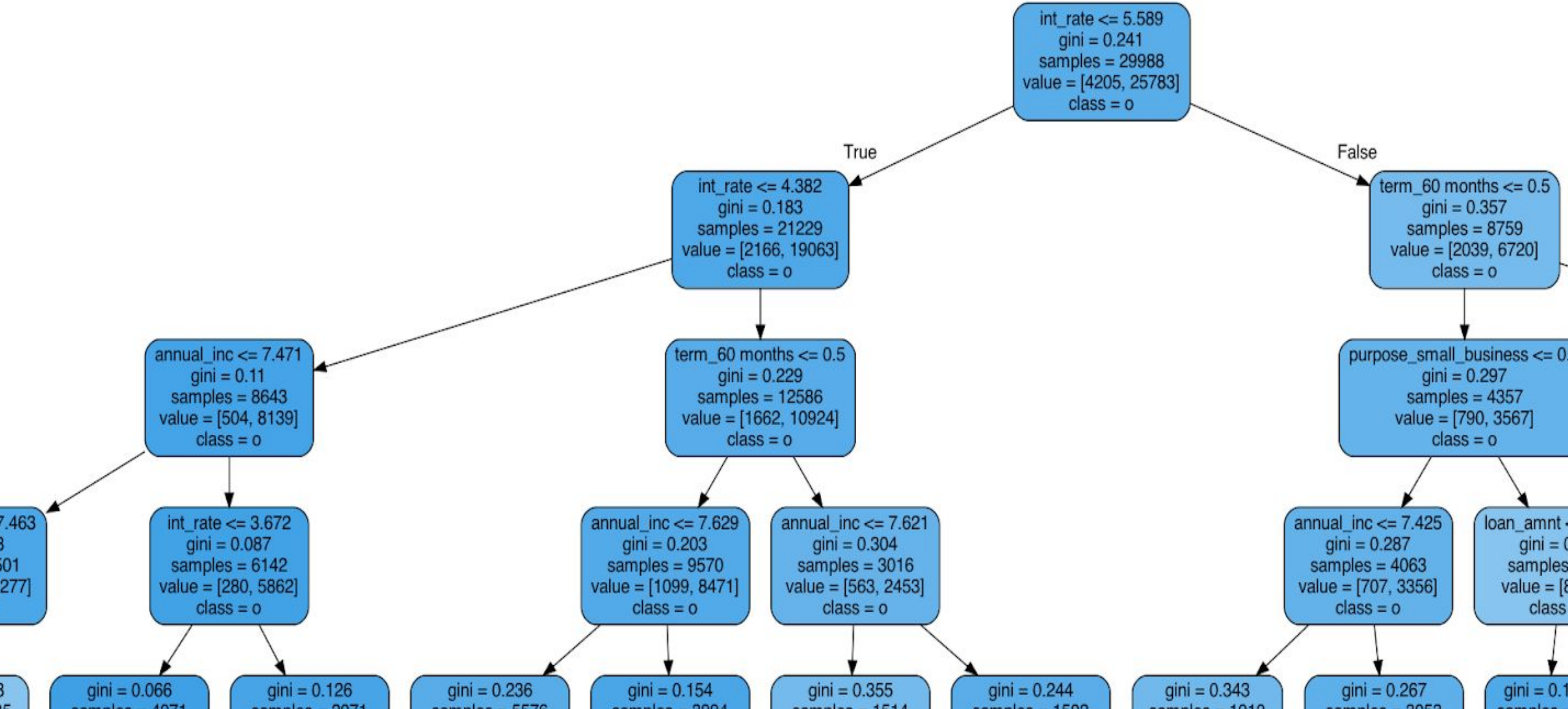
Optimal threshold

Optimal threshold for the Random Forest binary classification: 0.852

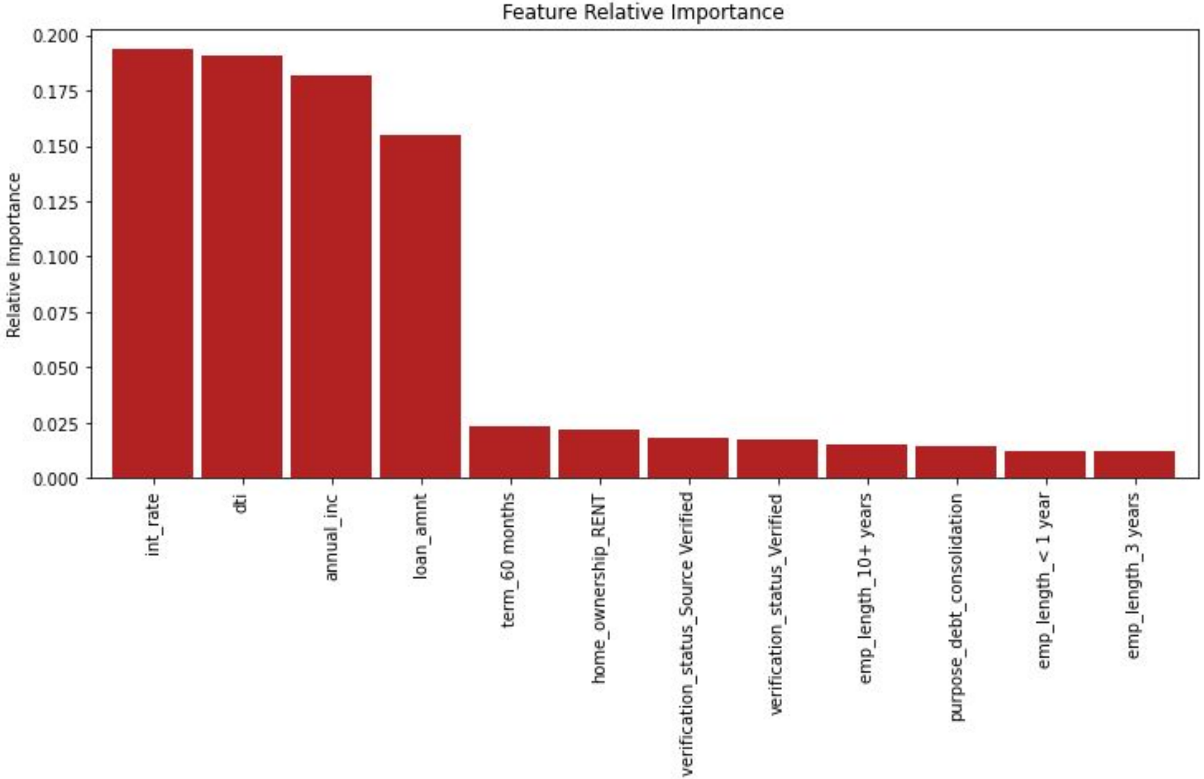




Decision Tree example



Feature Relative Importance



Model showcase

	loan_status	loan_amnt	int_rate	annual_inc	dti	fico_range	term_60 months	emp_length_10+ years	emp_length_2 years	emp_length_3 years	emp_length_4 years	emp_length_5 years	LR	RF
21691	1	36.85	6.30	7.51	23.65	0.26	1	0	0	1	0	0	True	True
3019	0	43.00	6.69	7.51	3.80	0.26	1	0	0	1	0	0	True	False
23869	1	37.85	4.02	7.68	5.25	0.26	0	0	0	1	0	0	True	True
35869	1	30.27	5.96	7.65	4.23	0.26	0	0	0	0	0	0	True	True
38229	1	38.78	4.66	7.88	17.82	0.26	0	1	0	0	0	0	True	True
20202	1	27.75	4.63	7.91	12.74	0.26	0	1	0	0	0	0	True	True
535	1	42.01	4.37	7.53	15.10	0.26	0	1	0	0	0	0	True	True
15619	0	30.27	6.94	7.38	5.10	0.26	1	0	1	0	0	0	True	False
24836	1	32.11	6.43	7.68	14.93	0.26	1	0	0	1	0	0	True	True
35110	1	38.50	5.47	7.58	22.66	0.26	0	0	1	0	0	0	True	True

Conclusion

I created three supervised machine learning models - Logistic Regression, Naive bayes and Random Forest. Model based on Logistic Regression was performing the best.

All models were pretty accurate in predicting applicants going to pay the loan back, not so good in predicting defaulters.

I carried out an explanatory analysis to present which features have the most direct impact on a loan being paid or falling into default.

The lending industry heavily relies on comprehensive risk assessments of the loan applicants. My suggestion would be to further optimize my models by feeding them with newer data from the most recent loans with known outcome.

Future project potential: Optimum threshold for the binary classifiers can be further adjusted by taking into account also the loan origination and service fee that borrowers and investors pay to the Lending Club.

Thank you for your attention!

Link to the project code on GitHub: https://github.com/LenkaRo/predicting_loan_defaults