https://github.com/Gabriels114/lendingclub-risk-analysis-cuc

CUC - UNIVERSIDAD DE LA COSTA

Departamento de Ciencias de la Computación y Electrónica
Materia: Data Mining

Evaluation 2 - Machine Learning Techniques

Presentado por:
Jesus Gabriel Gudiño Lara
Ana Rosa Ramirez Lopez

https://github.com/Gabriels114/lendingclub-risk-analysis-cuc

Introduction

This project is about helping a company called LendingClub. LendingClub gives loans to people. They need to know if a person will pay back the loan or not. If they say "yes" to a risky person, the company loses money. If they say "no" to a good person, they also lose potential profit.

We use data from past loans and computer models to predict this. This helps LendingClub make better and safer decisions.

Dataset Description and Problem Understanding

We have a file with information about many people who asked for a loan. The file has 396,030 rows and 27 columns.

```
df.describe()
```

8 rows × 12 cols

	loan_amnt	int_rate	installment	annual_inc	dti	open_acc	pub_rec	revol_bal	revol_util	total_acc	mort_acc	pub_
count	396030.000000	396030.000000	396030.000000	3.960300e+05	396030.000000	396030.000000	396030.000000	3.960300e+05	395754.000000	396030.000000	358235.000000	
mean	14113.888089	13.639400	431.849698	7.420318e+04	17.379514	11.311153	0.178191	1.584454e+04	53.791749	25.414744	1.813991	
std	8357.441341	4.472157	250.727790	6.163762e+04	18.019092	5.137649	0.530671	2.059184e+04	24.452193	11.886991	2.147930	
min	500.000000	5.320000	16.080000	0.000000e+00	0.000000	0.000000	0.000000	0.000000e+00	0.000000	2.000000	0.000000	
25%	8000.000000	10.490000	250.330000	4.500000e+04	11.280000	8.000000	0.000000	6.025000e+03	35.800000	17.000000	0.000000	
50%	12000.000000	13.330000	375.430000	6.400000e+04	16.910000	10.000000	0.000000	1.118100e+04	54.800000	24.000000	1.000000	
75%	20000.000000	16.490000	567.300000	9.000000e+04	22.980000	14.000000	0.000000	1.962000e+04	72.900000	32.000000	3.000000	
max	40000.000000	30.990000	1533.810000	8.706582e+06	9999.000000	90.000000	86.000000	1.743266e+06	892.300000	151.000000	34.000000	

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 396030 entries, 0 to 396029
Data columns (total 27 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   loan_amnt             396030 non-null  float64
 1   term                  396030 non-null  object
 2   int_rate              396030 non-null  float64
 3   installment           396030 non-null  float64
 4   grade                 396030 non-null  object
 5   sub_grade             396030 non-null  object
 6   emp_title             373103 non-null  object
 7   emp_length            377729 non-null  object
 8   home_ownership        396030 non-null  object
 9   annual_inc            396030 non-null  float64
 10  verification_status   396030 non-null  object
 11  issue_d               396030 non-null  object
 12  loan_status           396030 non-null  object
 13  purpose               396030 non-null  object
 14  title                 394274 non-null  object
 15  dti                   396030 non-null  float64
 16  earliest_cr_line      396030 non-null  object
 17  open_acc              396030 non-null  float64
 18  pub_rec               396030 non-null  float64
 19  revol_bal             396030 non-null  float64
 20  revol_util            395754 non-null  float64
 21  total_acc             396030 non-null  float64
 22  initial_list_status   396030 non-null  object
 23  application_type      396030 non-null  object
 24  mort_acc              358235 non-null  float64
 25  pub_rec_bankruptcies  395495 non-null  float64
 26  address               396030 non-null  object
dtypes: float64(12), object(15)
memory usage: 81.6+ MB
```

The main challenge is to look at a new person's information and predict if they will pay back the loan? We want to find the risky people so LendingClub can be careful. For the company, giving a loan to a risky person (a "false positive") is the most expensive mistake.

Column Descriptions

1. loan_amnt
   - Description: The total amount of the loan applied for by the borrower.
   - Example: 10000 (the borrower asked for $10,000).
2. term
   - Description: The number of payments on the loan. It is the loan's duration.
   - Example: 36 months (the loan must be paid back in 36 months, or 3 years).
3. int_rate
   - Description: The interest rate on the loan.
   - Example: 11.44 (the borrower has an 11.44% interest rate).
4. installment
   - Description: The monthly payment owed by the borrower if the loan is funded.
   - Example: 329.48 (the borrower must pay $329.48 every month).
5. grade
   - Description: A credit grade assigned by LendingClub (A, B, C, etc.). 'A' is the best because the lowest risk, 'G' is the worst.
   - Example: B
6. sub_grade
   - Description: A more specific category within each grade (e.g., B1, B2, B3...).
   - Example: B4
7. emp_title
   - Description: The job title supplied by the borrower when applying for the loan.
   - Example: Marketing, software development engineer
8. emp_length
   - Description: The length of the borrower's employment in years.
   - Example: 10+ years, < 1 year
9. home_ownership
   - Description: The home ownership status provided by the borrower.
   - Example: RENT, MORTGAGE, OWN
10. annual_inc
    - Description: The annual income declared by the borrower.
    - Example: 117000 (the borrower earns $117,000 per year).

11. verification_status
    - Description: Indicates if LendingClub verified the borrower's income.
    - Example: Verified, source verified, not verified
12. issue_d
    - Description: The month and year when the loan was issued.
    - Example: Jan-15
13. loan_status
    - Description: The target variable. It shows the current status of the loan. The main categories for our project are fully paid and charged off .
    - Example: Fully paid, charged off
14. purpose
    - Description: The borrower's stated reason for taking the loan.
    - Example: debt_consolidation, credit_card, home_improvement
15. title
    - Description: The loan title provided by the borrower. It's often a sub-category of the purpose.
    - Example: Vacation.
16. dti
    - Description: A ratio calculated using the borrower's total monthly debt payments (excluding the mortgage and the requested loan) divided by their monthly income.
    - Example: 26.24 (the borrower has a lot of debt compared to their income).
17. earliest_cr_line
    - Description: The month and year the borrower's earliest reported credit line was opened. This shows how long their credit history is.
    - Example: Jun-90
18. open_acc
    - Description: The number of open credit lines in the borrower's credit file.
    - Example: 16
19. pub_rec
    - Description: The number of derogatory public records (e.g., bankruptcies, tax liens).
    - Example: 0
20. revol_bal
    - Description: The total credit revolving balance. The amount of credit the borrower is currently using on credit cards and other revolving lines.
    - Example: 36369
21. revol_util
    - Description: Revolving line utilization rate. The amount of credit the borrower is using relative to all available revolving credit (a percentage).

    - Example: 41.8
22. total_acc
    - Description: The total number of credit lines currently in the borrower's credit file. This includes both open and closed accounts.
    - Example: 25
23. initial_list_status
    - Description: The initial listing status of the loan.
    - Example: f (fractional), w (whole).
24. application_type
    - Description: Indicates whether the loan is an individual application or a joint application with two co-borrowers.
    - Example: INDIVIDUAL, JOINT
25. mort_acc
    - Description: The number of mortgage accounts the borrower has.
    - Example: 3
26. pub_rec_bankruptcies
    - Description: The number of public record bankruptcies for the borrower.
    - Example: 0
27. address
    - Description: The borrower's home address, including city and state.
    - Example: "0174 Michelle Gateway Mendozaberg, OK 22690"

Discussion

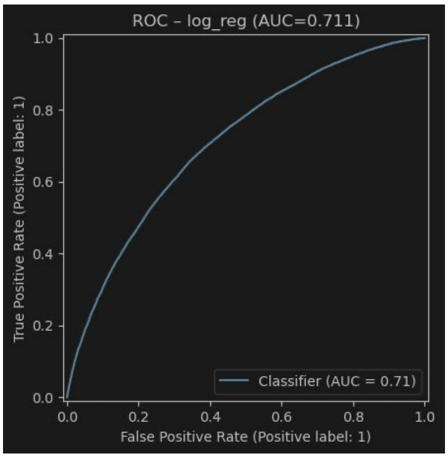
We used three different models to find the best one. (Random forest, logistic regression and random forest)

1. How do we compare the models?
We look at two main things:

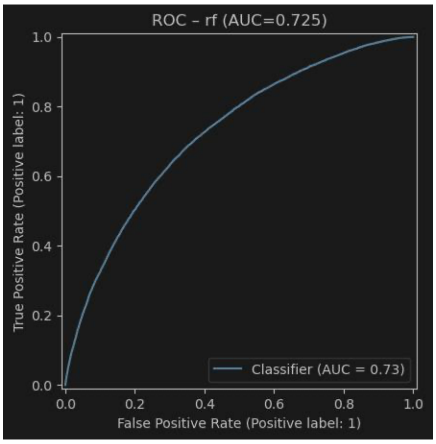
- Catching bad borrowers (recall): How many of the truly bad borrowers did we correctly find? A high number is good.
- Being correct (precision): When we say someone is bad, how often are we correct? a high number is good.

2. What happened with our models?

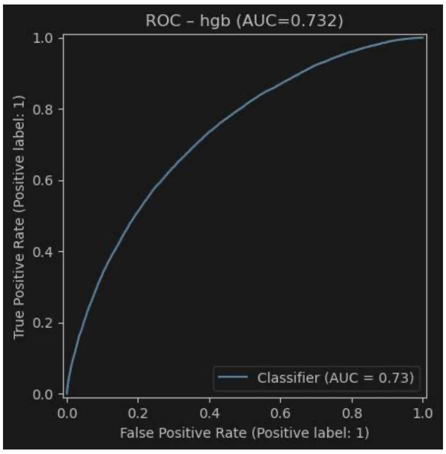
- Logistic regression: This model is simple and fast. It did an okay job, but it was not the best at finding the risky borrowers. It is easy to understand but not powerful enough.



- Random forest: This model was better. It found more of the risky people than the Logistic Regression model. It is good at finding complex patterns in the data.



- Histogram gradient boosting: This was our best model. It found the most risky borrowers and was also very correct. It had the best balance of all the models.



3. How did changing the models help?

Every model has settings, called hyperparameters. We changed these settings to make the models better. For example, we told the random forest to use more trees and we adjusted the learning speed for the gradient boosting. After we changed the settings, all models worked better. They became better at finding the risky borrowers without making too many mistakes.

Model Selection

	model	roc_auc	precision_1	recall_1	f1_1
0	log_reg	0.7112273754671136	0.3129944857348358	0.672288381074992	0.42713125984090955
1	rf	0.7253952182990573	0.4219269102990033	0.3842291599613775	0.4021966174786066
2	hgb	0.7316521201731845	0.5795323878880797	0.09732861280978436	0.16666666666666666

We choose the histogram gradient boosting model. Our main goal is to find people who will not pay back the loan. This model is the best at that task. It is the most accurate and reliable. While it is less simple to explain than logistic regression, its ability to save the company money is more important.

Conclusion

In conclusion, our model can help LendingClub a lot. By using the histogram gradient boosting model, the company can see which loan applications are very risky. This means they can say "no" to these risky people and lose less money. At the same time, the model is good enough to not say "no" to too many good people. This helps LendingClub be safer and make more money.