

Problem Statement

Build a lending portfolio where the approval rate is high and loss rate is low.

Data Generation Process

The following data consists of features

```
'past30days', 'past12months', 'past24months', 'gender', 'age',  
'no_of_accounts', 'no_of_active_accounts', 'no_of_writeoffs',  
'total_past_due', 'most_severe_status_within_24_months',  
'at_risk_account', 'single_highest_credit',  
'single_highest_sanction_amount', 'average_open_balance',  
'single_highest_balance', 'no_of_past_due_accounts',  
'no_of_zero_balance_accounts', 'recent_account', 'oldest_account',  
'total_balance_amount', 'total_sanction_amount', 'total_credit_limit',  
'total_monthly_payment_amount', 'total_high_credit',  
'total_past_due_scale', 'single_highest_credit_scale',  
'single_highest_sanction_amount_scale', 'total_high_credit_scale',  
'average_open_balance_scale', 'total_balance_amount_scale',  
'total_sanction_amount_scale', 'total_credit_limit_scale',  
'total_monthly_payment_amount_scale', 'oldest_account_loan_type',  
'oldest_account_loan_date', 'oldest_account_loan_date_year',  
'recent_account_loan_type', 'recent_account_loan_date',  
'recent_account_loan_date_year', 'has_past_due', 'has_closed_auto_loan',  
'no_auto_loan', 'auto_loan_amount', 'at_risk_auto_loan',  
'has_closed_cc', 'no_active_cc', 'no_closed_cc', 'cc_limit',  
'at_risk_cc', 'has_closed_consumer_loan', 'no_active_consumer_loan',  
'no_closed_consumer_loan', 'consumer_loan_amount',  
'at_risk_consumer_loan', 'has_closed_gold_loan', 'no_active_gold_loan',  
'no_closed_gold_loan', 'gold_loan_amount', 'at_risk_gold_loan',  
'has_closed_housing_loan', 'no_active_housing_loan',  
'no_closed_housing_loan', 'housing_loan_amount', 'at_risk_housing_loan',  
'has_closed_personal_loan', 'no_active_personal_loan',  
'no_closed_personal_loan', 'personal_loan_amount',  
'at_risk_personal_loan', 'date'
```

And two labels

'limit' - Zero indicating the user is not approved and a value greater than Zero indicates the value of the credit line provided to the customer.

'hasDPD' – False indicate the customer has never defaulted on his/her obligation. True indicates that customer has defaulted and the limit assigned to the customer will be the loss incurred.

Objective

1. Build a probabilistic classifier to determine if the customer is going to default or not.
2. For the customers who are not going to default come up with a mathematical framework to determine the limit such the cumulative loss are minimised.
3. Set the threshold for the classifier such that number of customers who are approved is maximised while cumulative loss are minimised.

Expectation

Share the details of your approach using ipython notebook or any other workbook (R Studio or Matlab or Octave etc) such that we can reproduce the results. Export the workbook to a HTML / PDF