CREDIT RISK EVALUATION SCORING ALGORITHM

This is a breakdown of the pseudocode used in the credit score evaluation pipeline.

There are 5 major factors that are considered in the calcultion as well as their weights(in floating point decimal):

I. Credit Utilization(0.2)

II. Payment History(0.5)

III. Maturity Index(0.05)

IV. Credit Accounts(0.15)

V. Loan Term(0.1)

The sum of the following weights add up to 1.Thus the credit score calculated in this formula is in the range 0,1.

It follows a series of mathematical calculations in which the credit score starts at 0(worst-case) and the weights are increased as the calculations go on such that the ideal output is 1(ideal customer).

The data given is raw and has to be passed through dedicated normalization functions as per the data type to make them easier to manipulate.These functions perform a min-max normalization on the data given.

1.normalize_business_duration(C6)

weights = [0.2, 0.4,0.6, 0.8, 1.0]

business_duration_ranges = (0- 2), (3-5), (6- 8), (9-11),(12- infinity)

For each range within the business duration ranges we return a value from the weights corresponding to the same index as the business duration range.

i.e. if the input is 4 then the weight returned should be 0.6

2.normalize_payment_method

Using if functions,it gives a weight of 0.33 if the paymemt method,C7 is cash;

0.67 if it is cash adn mobile money,

0.5 if it is mobile money OR if it is cash and bank

0.7 if it is bank

0.9 if it is mobile money and bank

1 if it is all three.

3.normalize_age

age_scores = {

(18-24): 0.25,

(25-34): 0.50,

(35-44): 0.75,

(45-infinity): 1.00}

The function should pass the user age in such a way that it returns the weight indicated if it is within the given range. i.e.age 39 should return a weight 0.75

4.normalize_revenue

First,we convert all given revenues to monthly revenue and take the highest value among them.

This involves taking daily revenue-D6***30,weekly revenue-D7***4,actual monthly revenue-D8,quarterly revenue-D9/4 and yearly revenue-D10/12.

After obtaining the highest value,create a range of percentiles as well as scores for it that.

percentiles = [0, 50000, 100000,200000, 500000, 700000,1000000,2000000, float('inf')]

scores = [1/8,2/8,3/8,4/8,5/8,6/8,7/8,1]

return a score if its index is within the percentile range.

5.normalize_dependants

Create a range of weights and scores for dependants and return the weight whose index corresponds to the index of dependants range chosen.Use for and if functions.

dependant_ranges=[(0,2),(2,4),(4,6),(6,8),(8,10),(10,np.float64('inf'))]

weights_housing_amount=[1,5/6,2/3,1/2,1/3,1/6]

6.normalize_monthly_demonstrated_affordability

this function takes the total amount in debts as well as monthly demonstrated affordability and finds their ratio.

this ratio is then taken as the weight of this function.Be careful to add cases for when it is less than 0 or greater than 1.

7.normalize_regional_per_capita

Obtain a list of regions in Tanzania as well as their per capita incomes.Then perform min-max normalization on the given amounts.Take the result as the weight for each region.

If you have been paying attention so far,then you will notice that all these functions do a simple mappling onto a predefined range to obtain a certain weight.The same logic applies moving forward and only specific ranges will be provided.

8.normalize_debt

debt_ranges = [(0, 50000), (50000, 100000), (100000, 150000),(150000,200000),(200000,250000),(250000,300000),(300000,350000),(350000,400000),(400000,450000),(450000,500000)]

weights_debt_amount = [1, 0.9, 0.8,0.7,0.6,0.5,0.4,0.3,0.2,0.1]

9.normalize_number_of_overdue_installments

overdue_ranges = [(0, 2), (3, 5),(5,7),(7,9),(9,11),(11,13),(13,15),(15, float('inf'))]

weights = [1,(7/8),0.75,(5/8),0.5,(3/8),0.25,(1/8)]

10.normalize_number_of_credit_inquiries

credit_inquiry_ranges = [(0, 1), (1, 3), (3, 6),(6,9),(9,12),(12,15),(15,np.float64('inf'))]

weights = [1,(6/7),(5/7),(4/7),(3/7),(2/7),(1/7),0]

You will need to make the normalization functions for each data type/variable given.

NEXT:

Lets begin creating functions for the different areas of consideration.

Note that the credit score within each function is given by calculating the product of the weights of the variables(given below) and the normalized value calculated by calling its respective function(given above).

Within each function,the credit score starts at 0 and is enumerated by the given weights.The maximum value is 1.

An example can be given below in Python:

```
def MaturityIndex(age,C6):
        cred=0
        #years in business has a weight of 60% here
        weight_years_in_business = 0.6
    cred+=normalization.normalize_business_duration(C6=C6)*weight_years_in_busin
ess

        #age has a weight of 40%
        weight_age=0.4
        cred+=normalization.normalize_age(B3=age)*weight_age
        return cred
```

CREDIT UTILIZATION

The variables used here,along with respective weights, are:

- financial_literacy(whether of not the customer has taken a course to obtain knowledge on how to run their business)(0.1)
- total_amount_in_debt(0.6)
- C7-Customer Payment Method(0.15)
- B15-Number of dependants the customer has(0.1875)
- C3-Whether the customer owns or rents their business stall(0.0625)

MATURITY INDEX

The variables used here,along with respective weights, are:

- age of the customer(0.4)
- C6-duraion that the person has been in business(0.6)

PAYMENT HISTORY

The variables used here,along with respective weights, are:

- monthly_demonstrated_affordability(0.2)
- number_of_overdue_installments(0.25)
- number_of_credit_inquiries(0.15)
- highest_past_due_amount(0.1)
- highest_past_due_days(0.1)
- regional_per_capita_income(0.2)

CREDIT ACCOUNTS

The variables used here,both their weights being 0.5 are:

- number_of_credit_accounts
- total_open_contracts

LOAN TERM

The variable used here is the loan_term having a weight of 1.

NEXT:

To calculate the final credit score:

Create 5 variablescorresponding to the 5 main functions above.

Call the main functions and obtain the scores per function.

Next,find the product of each respective score per function and their weights that are given at the beginning of this paper.

Then,find the sum of the products to obtain a credit score.

To calculate the maximum amount that the user is approved for a loan:

Obtain a scaled score for calculations.The scaled credit score is the credit score power(^) 2 if the credit score is less than 0.

Oterwise,the scaled credit score is equal to 0.5+(0.5*((credit score - 0.5)/0.5))

Then,the loan amount is calculated by the formula:

min_loan_amount + (max_loan_amount - min_loan_amount) * scaled_score