Model Documentation

Credit Score Forecasting

Malak Diab
Mohammed Hourani
Mayas Masalmeh

Problem Statement

We're working as a data scientist. Our dataset includes basic bank details with a lot of credit-related information. We're looking forward building an intelligent system that predict customers credit score based on the main features that effects it.

Task

Given a person's credit-related information, build a machine learning model that can classify the credit score.

Dataset

We have a dataset for 8 Months include 12500 customers; the total sum of records(rows) is 100k.

With 28 features(columns).

Column description

- ID: Represents a unique identification of an entry
- Customer_ID: Represents a unique identification of a person
- Name: Represents the name of a person

- Month: Represents the month of the year
- Age: Represents the age of the person
- SSN: Represents the social security number of a person
- Occupation: Represents the occupation of the person
- Annual_Income: Represents the annual income of the person
- Monthly_Base_Salary: Represents the monthly base salary of a person
- Num_Bank_Accounts: Represents the number of bank accounts a person holds
- Num_Credit_Card: Represents the number of other credit cards held by a person
- Interest_Rate: Represents the interest rate on credit card
- Num_of_Loan: Represents the number of loans taken from the bank
- Type_of_Loan: Represents the types of loan taken by a person
- Delay_from_due_date: Represents the average number of days delayed from the payment date
- Num_of_delayed_Payment: Represents the average number of payments delayed by a person
- Changed_Credit_Limit: Represents the percentage change in credit card limit
- Num_Credit_Inquiries: Represents the number of credit card inquiries
- Credit_Mix: Represents the classification of the mix of credits
- Outstanding_Debt: Represents the remaining debt to be paid (in USD)
- Credit_Utilization_Ratio: Represents the utilization ratio of credit card
- Credit_History_Age: Represents the age of credit history of the person
- Payment_of_Min_Amount: Represents whether only the minimum amount was paid by the person
- Total_EMI_per_month: Represents the monthly EMI payments (in USD)
- Amount_invested_monthly: Represents the monthly amount invested by the customer (in USD)
- Payment_Behaviour: Represents the payment behavior of the customer (in USD)
- Monthly_Balance: Represents the monthly balance amount of the customer (in USD)
- Credit_Score: Represents the bracket of credit score (Poor, Standard, Good)

After Data Analysis we are get out with this Results:

1-There aren’t any Missing values in the data

2- Outlier count in {Annual_Income: 2000, Monthly_Inhand_Salary: 2017 , Delay_from_due_date: 4002,Changed_Credit_Limit: 579, Num_Credit_Inquiries: 787, Outstanding_Debt: 5272,Total_EMI_per_month: 5044, Amount_invested_monthly: 4464, Monthly_Balance: 7400}

3- Most of the Features that Affect on the target (Credit Score):

- Number of bank accounts
- Number of credit Card
- Number of loans
- Delay in Days
- Number of Delayed Payments
- Outstanding Debt
- Payment of min Amount
- Interest Rate
- Num_Credit_Inquiries
- Changed_Credit_Limit

4-Credit Mix has the most effect on credit score because it has a high percentage of Matching.

5-We are applied Feature Extraction to create some addition features like:

- The mean of Monthly Balance for every customer through 8 months.
- The mean of Number of Delayed Payments for every customer through 8 months.

6-We are Processing the Type of loan column such as:

Every type of loan has an individual column and values (0 or 1) if the customer takes a student loan then the value of the student loan column is 1, otherwise is 0.

7- (Annual-cat, history-age-cat, age-cat) , it is an additional columns just for analysis purpose:

-Split the Annual Income for 3 Categories,

-Split the Credit_History_Age for 4 Categories,

-Split the Age of Customers for 5 Categories.

Model

Steps for Our model:

1-Reading the Data

2-Encoding Categorical Features and the Target

3-Processing the age column and drop the columns that not affecting on the target

4- Feature Selection, Select the most importance feature for the Model

5- Split the data to training and testing

6-Apply Standardization on the training and testing data

7-Class for training the model and make prediction on the test data

8-Fine tuning hyper parameter for the model

9- Save the model

****

1-Reading the Data:

We saved the data after data analysis, then we read it and used it to build a machine learning model, where we have 43 column.

2-Encoding Categorical Features and the Target:

get_dummy function: take the data and column name that we want to encode, after encoding it drop the original column then returns the data after encoding.

Occupation➔ get dummies encoding

Every Occupation has an individual column and values is (0 or 1) it is like one hot encoding, but it removes the last column, for example here we have 15 Occupation, but It is sufficient in 14 Occupation.

Payment_Behaviour➔ get dummies encoding

Every Payment_Behaviour has an individual column and values is (0 or 1), removes the last Payment_Behaviour.

leve_mapping function: take the data, column name and dictionary contain the original values with their mappings as a numeric, after encoding it drop the original column then returns the data after encoding .

Credit_Mix➔ Ordinal encoding

Good, is mapping to :2,

Standard, is mapping to :1,

Bad, is mapping to :0.

Payment_of_Min_Amount➔ Ordinal encoding

No, is mapping to :2,

NM, is mapping to :1,

Yes, is mapping to :0.

Credit_Score➔ Ordinal encoding

Good, is mapping to :2,

Standard, is mapping to :1,

Poor, is mapping to :0.

3- Processing the age column and drop the columns that not affecting on the target

When the Age is less than 18, the values will be 18

Remove ➔ Customer_ID, Name, SSN (unique).

Remove ➔ Monthly_Inhand_Salary, it has a high correlation with annual income so we are choosing just the Annual income and drop the Monthly_Inhand_Salary.

Remove ➔Type_of_Loan, instead we used every type of loan has an individual column

Remove ➔Annual-cat, history-age-cat, age_cat, instead we used numerical columns.

4-Feature Selection, Select the most importance feature for the Model:

We choose one month as a data because one month is enough to train and test the model on all the customers, so the data now is just one month, we split the data to:

X: Independent features all the columns excluding the target

Y: the target (credit score)

For Feature selection,

we used SelectKBest with f_regression to select the top 16 features with the lowest p-values.

we choose f_regression, because it gives best accuracy.

5-Split the data:

Then we apply train test split X_train,X_test ,y_train, y_test

Training 80%

Testing 20%

Stratify split to ensure the same distribution of classes in training and testing data.

6-Apply Standardization on the training and testing data:

We performed the scaling for all the data, because We have a feature have high-range comparing with the other features like annual income.

We used PowerTransformer on the training and testing data to rescale the data into lower scale.

7-Class for training the model and make prediction on the test data:

Class Models have 5 functions:

init function: is a constructor to initialize X_train,X_test ,y_train, y_test, model.

fit function: it is used to training the model on (X_train, y_train).

predict function: it is used the model to predict (X_test) and return array of predictions

Gradient boosting function: it is used Gradient boosting classifier algorithm to training the model on (X_train, y_train) and it use fit function to the training.

Return the score and the model.

classification_Report function: it is used the y_test and y_pred to evaluate the performance of the model and return the report.

8-Fine tuning hyper parameter for the model:

Fine tuning hyper parameter for the Gradient boosting classifier through randomSearch function that take:

X_train, y_train.

estimator as Gradient boosting model.

parameters include param grid that have all the parameter that we interest in choose the perfect values for it to increase the accuracy of the model.

n_estimators: The number of sequential trees to be modeled

learning_rate: This determines the impact of each tree on the final outcome

criterion: The function to measure the quality of a split

max_features: The number of features to consider when looking for the best split

max_depth: Maximum depth of the individual estimators

cv: in cross validation we use Stratified Kfold with 3 folds.

we are chosen scoring=accuracy for evaluation the model.

number of iterations is 300 to take a short time in running.

9- Save the model:

Saving scaler to disk as scaler.pkl to load it and scaler new values that we entered then make prediction using the model

Saving model to disk as model.pkl to load it and make predictions on the web site page.

Finished!