Rakamin Academy X HOME CREDIT

Home Credit Indonesia Data Science

Project Based Internship Program

Presented by

Gilang Muhamad Rizky

Problem Statement

Home Credit Indonesia faces the challenge of accurately assessing the creditworthiness of individuals applying for loans. The current manual evaluation process is time-consuming, prone to errors, and may not capture the full picture of an applicant's credit risk. There is a need for a more efficient and reliable system to evaluate credit applicants, considering various factors that can impact their creditworthiness.

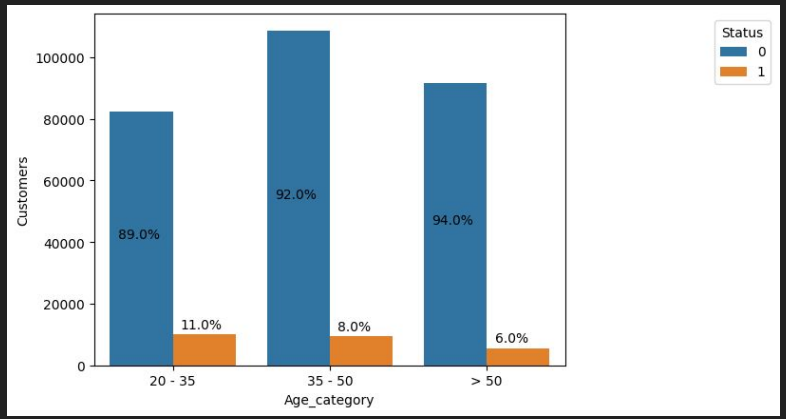
Goals

Firstly, the company aims to create a highly accurate and predictive model that efficiently assesses the credit risk of loan applicants, incorporating advanced statistical and machine learning techniques. Secondly, the focus is on automating the credit evaluation process to enhance efficiency, reduce manual efforts, and accommodate a larger volume of applications without compromising accuracy. Additionally, the goal includes customization for diverse customer profiles, ensuring the model considers a range of data sources to provide fair and comprehensive credit assessments.

Link Github / Code dapat dilihat disini

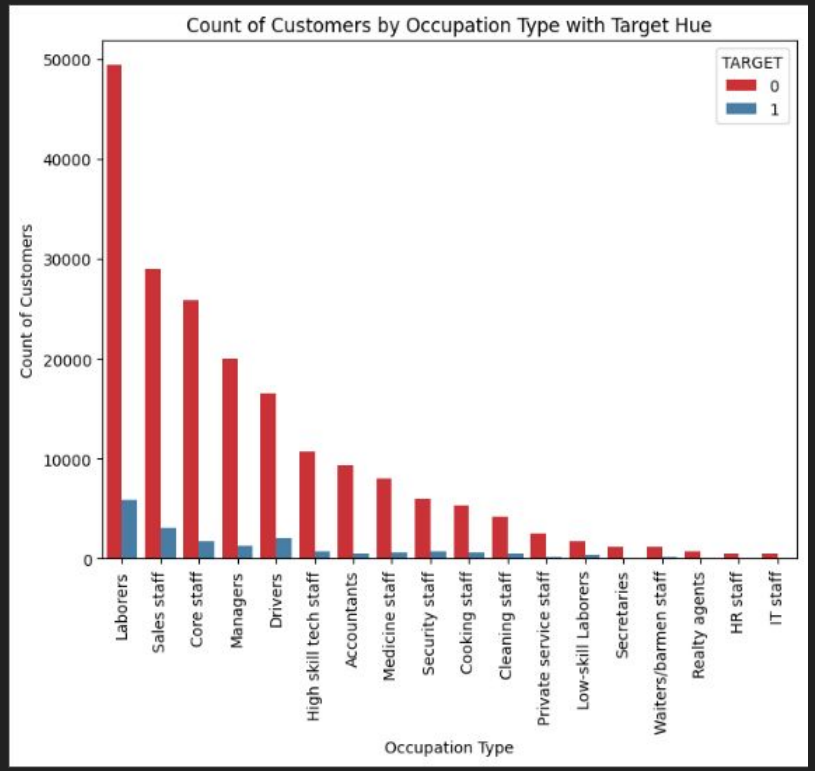
EDA

	TARGET	Customers	Customers_Percentage
0	0	282686	0.92
1	1	24825	0.08



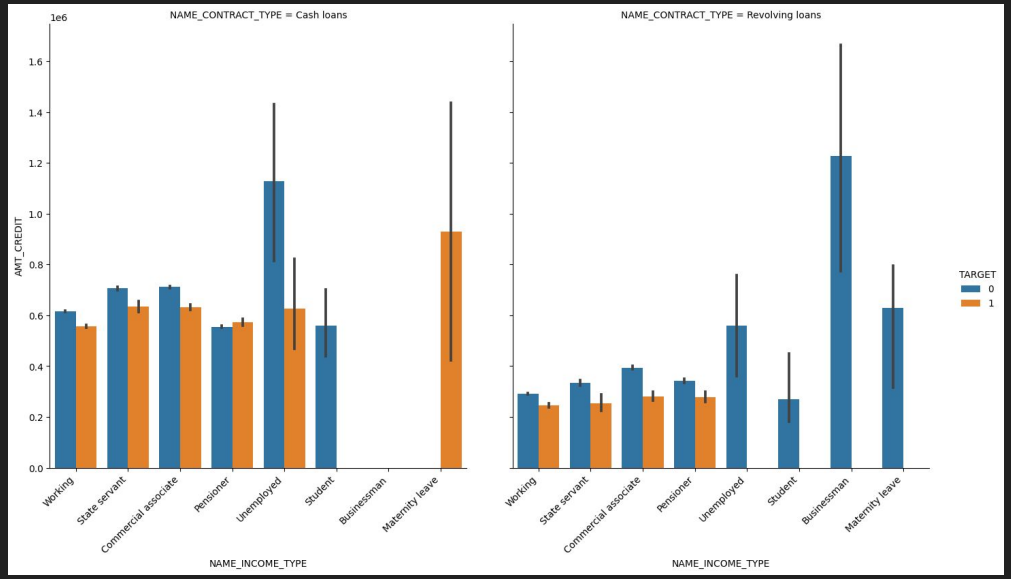
From the data exploration process, it was found that only 8% of customers face issues in repaying loans out of the entire dataset. The majority of borrowers fall within the age range of 35 - 50 years. However, when considering the percentage of problematic borrowers compared to non-problematic ones, the age range of 20 - 35 years shows a higher percentage of borrowers facing issues compared to other age categories.

EDA (Contd.)



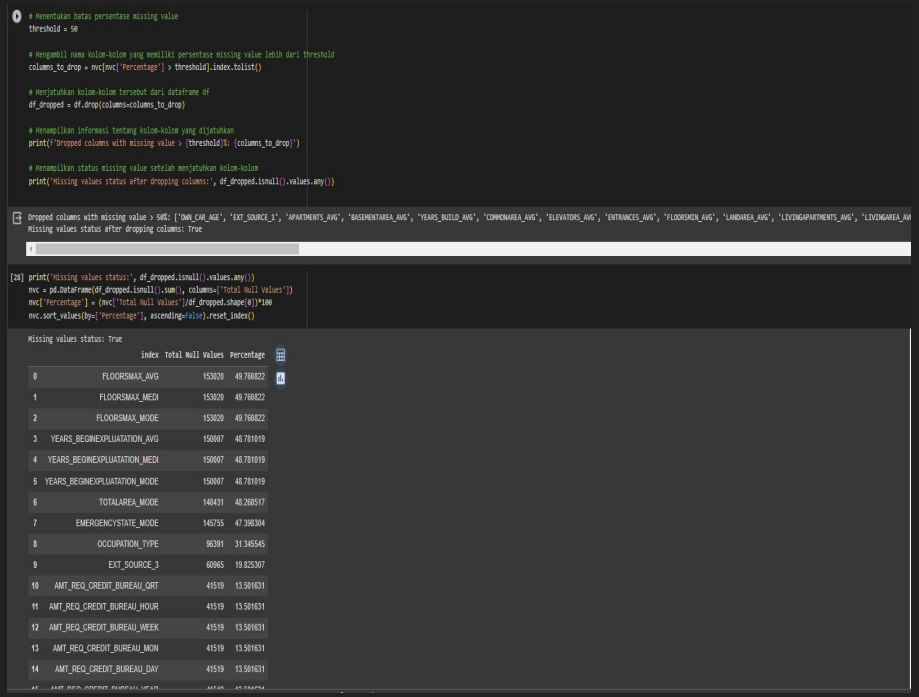
The most common occupations among borrowers include Laborers, Sales Staff, Core Staff, Managers, and Drivers. When analyzed by percentage, customers employed as Secretaries, Realty Agents, HR Staff, and IT Staff exhibit a smooth repayment record without encountering issues in loan payments.

EDA (Contd.)



Clients on maternity leave face challenges repaying cash loans, while those with revolving loans experience no difficulties. Unemployed clients encounter repayment issues for cash loans, particularly when the credit amount is medium, with over 50% facing problems. However, unemployed clients with revolving loans consistently demonstrate no repayment difficulties. Students, whether with cash or revolving loans, face no challenges in repaying loans, especially for low to medium credit amounts.

Data Preprocessing



```
# Menentukan batas persentase missing value
threshold = 50

# Mengambil nama kolom-kolom yang memiliki persentase missing value lebih dari threshold
columns_to_drop = nvc[nvc['Percentage'] > threshold].index.tolist()

# Menjatuhkan kolom-kolom tersebut dari dataframe df
df_dropped = df.drop(columns=columns_to_drop)

# Menampilkan informasi tentang kolom-kolom yang dijatuhkan
print(f'Dropped columns with missing value > {threshold}%: {columns_to_drop}')

# Menampilkan status missing value setelah menjatuhkan kolom-kolom
print('Missing values status after dropping columns:', df_dropped.isnull().values.any())
```

```
Dropped columns with missing value > 50%: ['OWN_CAR_AGE', 'EXT_SOURCE_1', 'APARTMENTS_AVG', 'BASEMENTAREA_AVG', 'YEARS_BUILD_AVG', 'COMMONAREA_AVG', 'ELEVATORS_AVG', 'ENTRANCES_AVG', 'FLOORSKIN_AVG', 'LANDAREA_AVG', 'LIVINGAPARTMENTS_AVG', 'LIVINGAREA_AV
Missing values status after dropping columns: True
```

```
print('Missing values status:', df_dropped.isnull().values.any())
nvc = pd.DataFrame(df_dropped.isnull().sum(), columns=['Total Null Values'])
nvc['Percentage'] = (nvc['Total Null Values']/df_dropped.shape[0])*100
nvc.sort_values(by=['Percentage'], ascending=False).reset_index()
```

Missing values status: True

	index	Total Null Values	Percentage
0	FLOORSMAX_AVG	153020	49.760822
1	FLOORSMAX_MEDI	153020	49.760822
2	FLOORSMAX_MODE	153020	49.760822
3	YEARS_BEGINEXPLUATATION_AVG	150007	48.781019
4	YEARS_BEGINEXPLUATATION_MEDI	150007	48.781019
5	YEARS_BEGINEXPLUATATION_MODE	150007	48.781019
6	TOTALAREA_MODE	148431	48.268517
7	EMERGENCYSTATE_MODE	145755	47.398304
8	OCCUPATION_TYPE	96391	31.345545
9	EXT_SOURCE_3	60965	19.825307
10	AMT_REQ_CREDIT_BUREAU_QRT	41519	13.501631
11	AMT_REQ_CREDIT_BUREAU_HOUR	41519	13.501631
12	AMT_REQ_CREDIT_BUREAU_WEEK	41519	13.501631
13	AMT_REQ_CREDIT_BUREAU_MON	41519	13.501631
14	AMT_REQ_CREDIT_BUREAU_DAY	41519	13.501631

Performing a drop column operation on columns with empty values, a missing value ratio exceeding 50%, and columns with only one unique value. Additionally, categorizing values in the Marital Status column into two categories: Single and Married.

Data Preprocessing

- Fill missing values by inputting the median or mode.
- Apply One Hot Encoding for categorical data with more than two unique values and use label encoding for columns with only two unique values.
- Conduct oversampling using SMOTE on the training data.

Feature Selection

```
[47] xfeature = df_train_ready.drop(['TARGET'], axis=1)
     yfeature = df_train_ready['TARGET']
```

```
[48] bestfeatures = SelectKBest(score_func=chi2, k=20)
     fit = bestfeatures.fit(xfeature,yfeature)
     dfscores = pd.DataFrame(fit.scores_)
     dfcolumns = pd.DataFrame(xfeature.columns)
     featureScores = pd.concat([dfcolumns, dfscores], axis = 1)
     featureScores.columns = ['Features', 'Score']
     print('The features that correlate well with target feature:\n')
     featureScores.sort_values(by=['Score'], ascending=False)
```

```
[50] top_20_feature = a['Features'].iloc[:20].tolist()
     top_20_feature

     ['DAYS_EMPLOYED',
      'AMT_GOODS_PRICE',
      'AMT_CREDIT',
      'AMT_INCOME_TOTAL',
      'DAYS_REGISTRATION',
      'DAYS_LAST_PHONE_CHANGE',
      'DAYS_ID_PUBLISH',
      'AMT_ANNUITY',
      'Age',
      'NAME_EDUCATION_TYPE_Higher education',
      'REG_CITY_NOT_WORK_CITY',
      'CODE_GENDER_M',
      'EXT_SOURCE_2',
      'REG_CITY_NOT_LIVE_CITY',
      'NAME_INCOME_TYPE_Pensioner',
      'ORGANIZATION_TYPE_XNA',
      'NAME_INCOME_TYPE_Working',
      'DEF_30_CNT_SOCIAL_CIRCLE',
      'EXT_SOURCE_3',
      'DEF_60_CNT_SOCIAL_CIRCLE']
```

Performing Feature Selection using SelectKBest with the chi2 score function and selecting the top 20 features for the modeling process.

Modelling

Model	Precision		Accuracy		Recall		F1-Score	
	Train	Test	Train	Test	Train	Test	Train	Test
AdaBoost	**0.80**	**0.7977**	**0.8057**	**0.8024**	**0.81**	**0.8082**	**0.80**	**0.80**
Random Forest	1.00	0.89	1.00	0.89	1.00	0.88	1.00	0.88
Decision Tree	1.00	0.79	1.00	0.8068	1.00	0.80	1.00	0.80
Logistic Regression	0.59	0.59	0.58	0.58	0.51	0.51	0.55	0.55

Model terbaik yang dipilih adalah AdaBoost, karena hasil yang diperoleh dari model RF dan DT dikategorikan sebagai overfitting karena perbedaan hasil metrics yang cukup signifikan antara data train dan data test

Business Recommendation

- Utilize the established Machine Learning model as a fundamental analysis for customers seeking to apply for loans.
- Categorize customers as either bad or good based on factors such as late payments, defaults, non-compliance with credit rules, and others.
- Provide the best offers to customers employed as Secretaries, Realty Agents, HR Staff, and IT Staff, as they tend to have fewer issues with loan repayments, potentially boosting revenue.
- Avoid offering cash loans with a maternity leave income type, as it is more likely to pose repayment challenges.