STAGE 2

Conexus Group

I. Data Cleansing

1.1. HC_application_train

1.1.1. Handle Missing Value

1. **Identifikasi Nilai yang Hilang:** Pada tahap awal, proses ini memeriksa jumlah dan proporsi nilai yang hilang pada setiap kolom menggunakan .isnull().sum() dan .isnull().mean(). Hal ini membantu menentukan kolom mana yang perlu diimputasi atau dihapus.

feature	null_num		
AMT_ANNUITY	12	OBS_60_CNT_SOCIAL_CIRCLE	1021
AMT_GOODS_PRICE	278	DEF_60_CNT_SOCIAL_CIRCLE	1021
OWN_CAR_AGE	202929	AMT_REQ_CREDIT_BUREAU_HOUR	41519
EXT_SOURCE_1	173378	AMT_REQ_CREDIT_BUREAU_DAY	41519
EXT_SOURCE_2	660	AMT_REQ_CREDIT_BUREAU_WEEK	41519
EXT_SOURCE_3	60965	AMT_REQ_CREDIT_BUREAU_MON	41519
DAYS_LAST_PHONE_CHANGE	1	AMT_REQ_CREDIT_BUREAU_QRT	41519
CNT_FAM_MEMBERS	2	AMT_REQ_CREDIT_BUREAU_YEAR	41519
OBS_30_CNT_SOCIAL_CIRCLE	1021	NAME_TYPE_SUITE	1292
DEF_30_CNT_SOCIAL_CIRCLE	1021	OCCUPATION_TYPE	96391

2. **Imputasi Nilai yang Hilang**:
   - **Numerik**: Untuk kolom numerik, proses ini mengimputasi nilai yang hilang dengan menggunakan mean atau median, tergantung pada

distribusi data. Jika distribusi data normal, imputasi dengan mean dilakukan, sedangkan untuk distribusi yang miring, median digunakan.

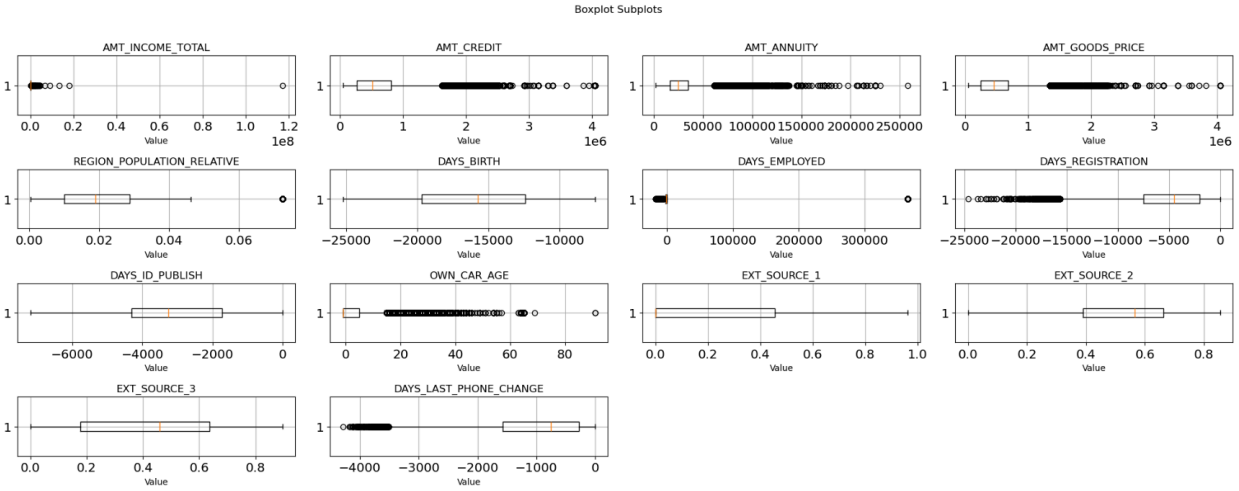
- **Kategorikal**: Untuk kolom kategorikal, proses ini mengisi nilai yang hilang dengan modus (nilai yang paling sering muncul) atau membuat kategori baru, seperti "Unknown".
- **Tujuan**: Imputasi ini menjaga agar dataset tetap lengkap, mengurangi risiko bias dari nilai yang hilang, serta memastikan data siap untuk analisis.

1.1.2. Handle Duplicated Data

Tidak terdapat nilai duplicate pada dataset, sehingga tidak perlu dilakukan handling duplicate data.

1.1.3. Handle Outliers

1. **Identifikasi Outlier**: Dalam langkah ini, metode **Interquartile Range (IQR)** digunakan untuk mendeteksi outlier. Data yang berada di bawah Q1 - 1.5 * IQR atau di atas Q3 + 1.5 * IQR dianggap sebagai outlier.



2. **Penanganan Outlier**: Outlier yang teridentifikasi dapat diatasi dengan beberapa metode, seperti **Winsorization** untuk membatasi nilai ekstrim pada batas persentil tertentu, atau dengan melakukan **Transformasi Logaritma** untuk mengurangi efek outlier pada data.
3. **Tujuan**: Penanganan outlier membantu menjaga distribusi data tetap stabil dan membuat model lebih tahan terhadap nilai ekstrim.

Dari 307.511 baris dataset yang ada pada application_train, setelah dilakukan penanganan outlier menghasilkan 187.484 baris dataset yang sudah ditangani.

1.1.4. Feature Transformation

1. **Log Transformation**: Transformasi logaritma diterapkan pada fitur yang memiliki distribusi skewed (mirip dengan distribusi eksponensial) untuk mengurangi skewness dan membuat distribusi lebih mendekati normal.
2. **Tujuan**: Transformasi fitur membantu meningkatkan performa model dengan menghilangkan efek skala yang berbeda antara fitur dan mengurangi skewness dalam data.

Adapun transformasi fitur yang dilakukan hanya pada kolom AMT_INCOME_TOTAL yaitu Log Transformation.

1.1.5. Feature Encoding

1. **Label Encoding**: Untuk variabel kategorikal yang memiliki urutan, proses ini menerapkan **Label Encoding**, yang mengubah setiap kategori menjadi nilai integer. Misalnya, kategori "Low", "Medium", "High" bisa diberi label 1, 2, dan 3.
2. **Tujuan**: Encoding fitur memungkinkan model untuk memproses variabel kategorikal dalam format numerik tanpa kehilangan informasi penting atau mengasumsikan urutan yang tidak ada.

feature	unique_sample
NAME_CONTRACT_TYPE	[Cash loans, Revolving loans]
CODE_GENDER	[M, F, XNA]
FLAG_OWN_CAR	[N, Y]
FLAG_OWN_REALTY	[Y, N]
NAME_TYPE_SUITE	[Unaccompanied, Family, Spouse, partner, Child...
NAME_INCOME_TYPE	[Working, State servant, Commercial associate,...
NAME_EDUCATION_TYPE	[Secondary / secondary special, Higher educati...
NAME_FAMILY_STATUS	[Single / not married, Married, Civil marriage...
NAME_HOUSING_TYPE	[House / apartment, Rented apartment, With par...
OCCUPATION_TYPE	[Laborers, Core staff, Accountants, Managers, ...
WEEKDAY_APPR_PROCESS_START	[WEDNESDAY, MONDAY, THURSDAY, SUNDAY, SATURDAY...
ORGANIZATION_TYPE	[Business Entity Type 3, School, Government, R...

Kolom-kolom kategorikal ini semua dilakukan Label Encoding sehingga isi data tersebut hanya berisikan nilai integer.

1.1.6. Handle Class Imbalance

```
TARGET
0    0.906669
1    0.093331
```

Dari pengecekan TARGET dari dataset ditemukan Imbalance Class yang secara perbandingan sangat jauh dengan rasio 91% : 9%. Maka dari itu perlu dilakukan penanganan Imbalance Class dengan menggunakan metode oversampling SMOTE. Metode oversampling SMOTE ini menggunakan sampling_strategy yaitu minority yang membuat data minoritas yang ada disamakan jumlahnya dengan data mayoritas yang ada sehingga data minoritas tersebut menjadi seimbang.

1.2. HC_bureau | bureau_balance

Sebelum dilakukan proses pre-processing, dataset **Bureau** dilakukan join terlebih dahulu dengan dataset **Bureau_Balance**. Adapun dataset *Bureau* berisikan informasi dasar tentang setiap peminjam, seperti ID peminjam, informasi pinjaman, dan status kredit secara umum. Sedangkan dataset *Bureau_Balance* berisi rincian informasi saldo akun, status pembayaran (misalnya, apakah tepat waktu, terlambat, atau gagal bayar) dalam periode bulan. Pada awalnya, pemisahan antara dataset *Bureau* dan *Bureau_Balance* ialah untuk tujuan manajemen data. Namun pada permodelan sistem yang akan dibangun kali ini, kedua dataset akan digabung atau join sehingga pemodelan menjadi lebih sederhana dan mengurangi potensi kesalahan yang berkaitan dengan pengelolaan data terpisah.

Setelah proses penggabungan dataset, data memiliki total 24.179.741 baris dan 19 kolom. Adapun features atau kolom categorical antara lain : CREDIT_ACTIVE, CREDIT_CURRENCY, CREDIT_TYPE, dan STATUS. Sedangkan features atau kolom yang memiliki tipe data numerical antara lain :

1. SK_ID_CURR
2. SK_ID_BUREAU
3. DAYS_CREDIT
4. CREDIT_DAY_OVERDUE
5. DAYS_CREDIT_ENDDATE
6. DAYS_ENDDATE_FACT
7. AMT_CREDIT_MAX_OVERDUE
8. CNT_CREDIT_PROLONG
9. AMT_CREDIT_SUM
10. AMT_CREDIT_SUM_DEBT

11. AMT_CREDIT_SUM_LIMIT
12. AMT_CREDIT_SUM_OVERDUE
13. DAYS_CREDIT_UPDATE
14. AMT_ANNUITY
15. MONTHS_BALANCE

Pada tahap selanjutnya, proses pre-processing akan dilakukan pada dataset hasil penggabungan antara Bureau dan Bureau_Balance. Dengan demikian, dataset Bureau_Balance tidak akan diproses secara terpisah. Pada tahap pemodelan, baik dataset application train maupun test akan menggunakan dataframe hasil penggabungan Bureau dan Bureau_Balance.

1.2.1. Handle Missing Value

Terdapat beberapa kolom dengan nilai kosong atau missing value antara lain :

1. DAYS_CREDIT_ENDDATE sebanyak 1.177.501 baris = 4,86% data
2. DAYS_ENDDATE_FACT sebanyak 5.628.643 baris = 23,27% data
3. AMT_CREDIT_MAX_OVERDUE sebanyak 17.545.903 baris = 72,56% data
4. AMT_CREDIT_SUM sebanyak 5 baris data
5. AMT_CREDIT_SUM_DEBT sebanyak 4.089.077 baris = 16,91% data
6. AMT_CREDIT_SUM_LIMIT sebanyak 10.376.144 baris = 42,91% data
7. AMT_ANNUITY sebanyak 9.553.957 baris = 39,51% data

Untuk mengisi missing value atau kekosongan nilai, setiap fitur diisi menggunakan nilai tengah atau median dari fitur tersebut. Pemilihan median bertujuan untuk menjaga distribusi data, terutama karena masih terdapat banyak outliers di setiap fitur. Median lebih robust terhadap nilai pencilan atau outliers, sehingga dapat menjaga keakuratan distribusi data.

1.2.2. Handle Duplicated Data

Tidak terdapat nilai duplicate pada dataset, sehingga tidak perlu dilakukan handling duplicate data.

1.2.3. Handle Outliers

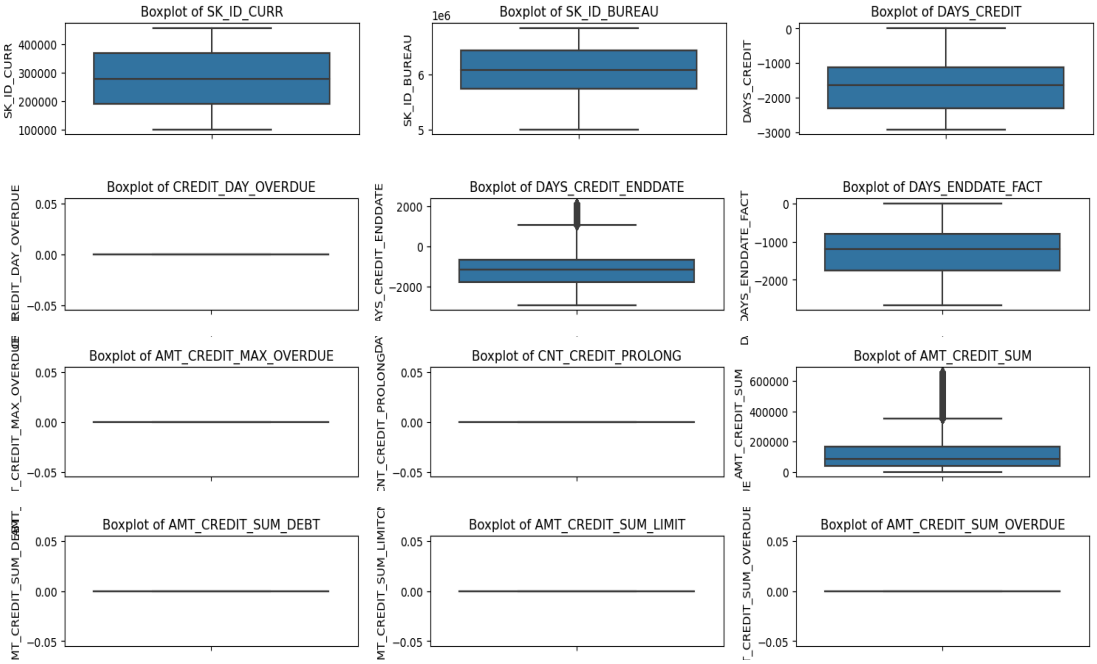
Dataset memiliki distribusi data yang tidak merata, di mana outliers terdeteksi pada hampir semua fitur numerik, kecuali pada fitur atau kolom SK_ID_CURR, SK_ID_BUREAU, dan DAYS_CREDIT. Kolom ID memiliki nilai yang unik dan berurutan, sehingga tidak menunjukkan adanya variasi ekstrem atau lonjakan nilai yang umumnya ditemukan pada variabel numerik lain dalam dataset.

Untuk menangani nilai outliers, digunakan **metode Interquartile Range (IQR)**, yang dianggap lebih robust terhadap data yang tidak terdistribusi normal. Metode IQR ini bekerja dengan mengidentifikasi outliers berdasarkan batasan Q1 (kuartil pertama) dan Q3 (kuartil ketiga). Nilai di luar rentang [Q1 - 1.5 * IQR, Q3 + 1.5 * IQR] dianggap sebagai outliers dan akan ditangani sesuai prosedur pre-processing, sehingga menjaga distribusi data tetap optimal tanpa dipengaruhi oleh nilai ekstrem.

Setelah dilakukan proses handling outliers menggunakan metode IQR, masih ditemukan pecilan atau outliers di beberapa features antara lain :

1. AMT_CREDIT_SUM
2. AMT_CREDIT_SUM_DEBT
3. DAYS_CREDIT_ENDDATE
4. DAYS_CREDIT_UPDATE
5. DAYS_CREDIT

Namun hasil ini diterima dan tidak dilakukan handling outliers kembali pada kolom yang masih terdapat pecilan tersebut. Hal ini bertujuan untuk menghindari kehilangan informasi jika data outliers atau pecilan tersebut tergolong kedalam jenis "kredit jumlah besar" yang dapat memberikan wawasan berharga dalam analisis risiko kredit dan perilaku peminjam. Berikut adalah hasil handling outliers menggunakan metode IQR :



1.2.4. Feature Transformation

Dalam proses ***feature transformation***, dilakukan ***feature scaling*** dan penyesuaian distribusi data dengan metode transformasi. Pada kolom MONTH_BALANCE, ***feature scaling*** diterapkan untuk mengonversi nilai yang awalnya negatif (-) menjadi positif (+), tanpa mengubah makna informasi yang terkandung. Kolom MONTH_BALANCE merepresentasikan status pembayaran bulanan nasabah, di mana nilai negatif menunjukkan adanya tunggakan selama periode tertentu.

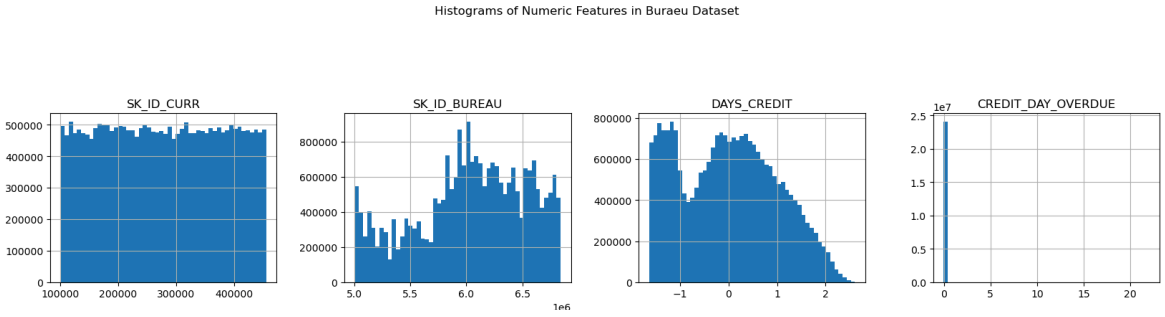
Karena tidak ada informasi atau bias yang mengindikasikan lawan dari nilai negatif (misalnya, nilai positif yang menunjukkan ketepatan pembayaran),

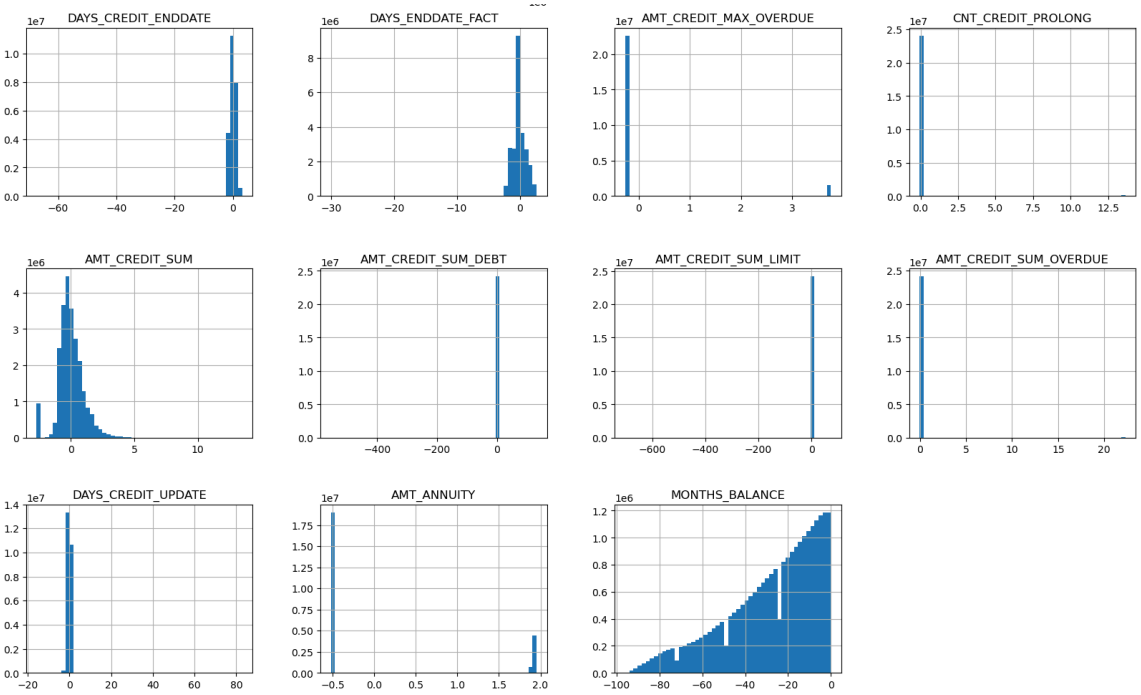
perubahan ini diambil untuk menjaga konsistensi dan mempermudah pemodelan sistem. Dengan demikian, representasi nilai yang seragam dalam bentuk positif mendukung proses analisis lebih lanjut tanpa mengorbankan interpretasi data asli.

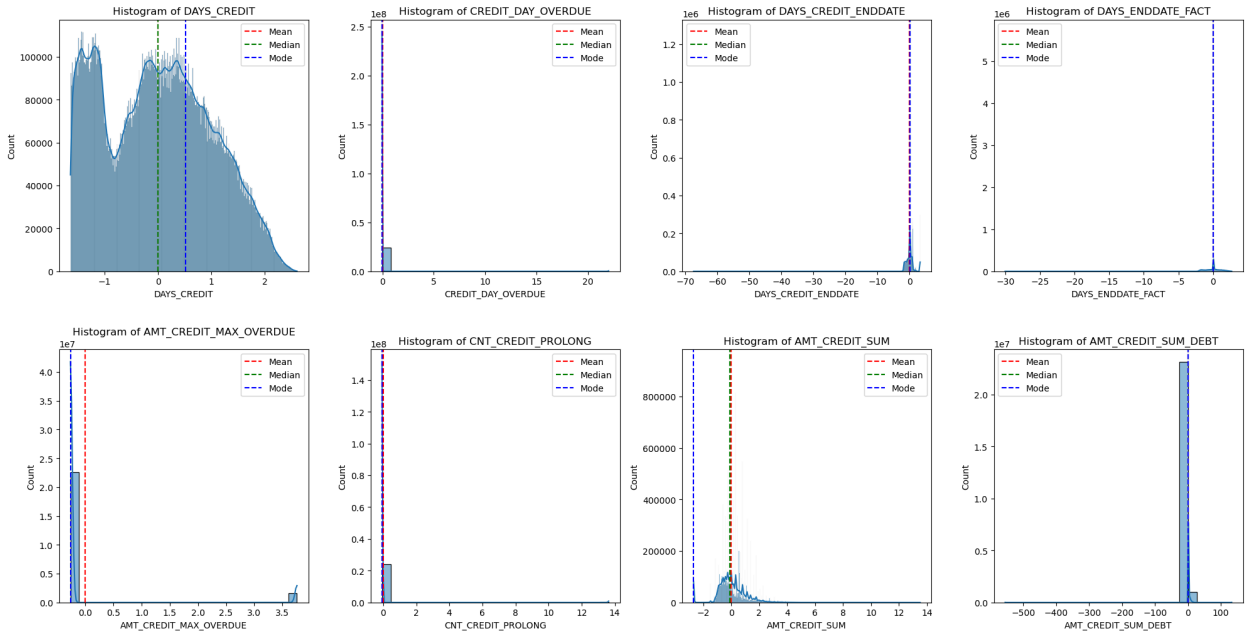
Tahapan selanjutnya adalah proses ***feature transformation*** yang bertujuan untuk menstabilkan variansi data dan mendekatkan distribusinya ke bentuk normal. Transformasi ini membantu meningkatkan performa model dan validitas asumsi-asumsi statistik, terutama bagi algoritma yang sensitif terhadap pola distribusi data.

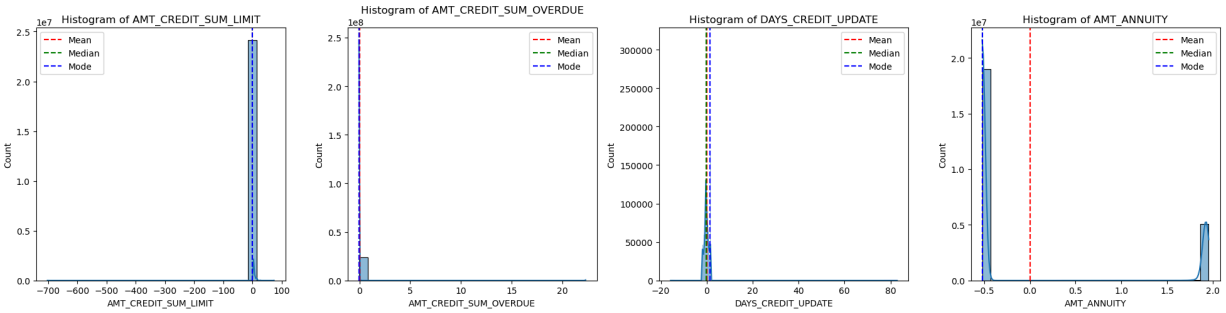
Features atau kolom yang mengalami proses *feature transformation* adalah kolom-kolom dengan nilai numerik, kecuali kolom ID dan Month Balance. Kolom ID tidak memerlukan transformasi karena hanya berfungsi sebagai pengenal unik untuk setiap entitas dan tidak membawa informasi numerik yang relevan untuk analisis atau pemodelan. Sedangkan kolom *Month Balance* sudah merepresentasikan informasi status pembayaran bulanan nasabah dalam bentuk yang bermakna. Mengubah skala atau distribusi bisa membuat interpretasi asli menjadi kurang jelas atau mengurangi kemudahan pemahaman terkait status tunggakan bulanan.

Metode yang digunakan untuk proses *feature transformation* adalah **Yeo-Johnson Transformation**. Metode ini dipilih karena fleksibel dalam mentransformasi data numerik yang mengandung nilai negatif, positif, dan nol, berbeda dengan metode transformasi lainnya, seperti logaritmik, yang hanya cocok untuk data bernilai positif. Hasil dari *feature transformation* ini diharapkan dapat meningkatkan interpretasi data dan mendukung proses pemodelan yang lebih akurat. Berikut ini adalah hasil *feature transformation*:




DAYS_CREDIT_ENDDATE
DAYS_ENDDATE_FACT
AMT_CREDIT_MAX_OVERDUE
CNT_CREDIT_PROLONG
AMT_CREDIT_SUM
AMT_CREDIT_SUM_DEBT
AMT_CREDIT_SUM_LIMIT
AMT_CREDIT_SUM_OVERDUE
DAYS_CREDIT_UPDATE
AMT_ANNUITY
MONTHS_BALANCE


Histogram of DAYS_CREDIT
Histogram of CREDIT_DAY_OVERDUE
Histogram of DAYS_CREDIT_ENDDATE
Histogram of DAYS_ENDDATE_FACT
Histogram of AMT_CREDIT_MAX_OVERDUE
Histogram of CNT_CREDIT_PROLONG
Histogram of AMT_CREDIT_SUM
Histogram of AMT_CREDIT_SUM_DEBT
Mean
Median
Mode
Count
DAYS_CREDIT
CREDIT_DAY_OVERDUE
DAYS_CREDIT_ENDDATE
DAYS_ENDDATE_FACT
AMT_CREDIT_MAX_OVERDUE
CNT_CREDIT_PROLONG
AMT_CREDIT_SUM
AMT_CREDIT_SUM_DEBT



1.2.5. Feature Encoding

Terdapat beberapa features atau kolom yang memiliki tipe data categorical, antara lain : CREDIT_ACTIVE, CREDIT_CURRENCY, CREDIT_TYPE dan STATUS. Masing - masing features menggunakan metode yang berbeda untuk proses features encoding.

1. CREDIT_ACTIVE

Pada kolom **CREDIT_ACTIVE**, terdapat beberapa kategori yang menjelaskan status kredit nasabah, yaitu **Active**, **Closed**, **Sold**, dan **Bad debt**. Sesuai dengan tujuan permodelan sistem, yaitu mengembangkan model machine learning yang mampu memprediksi kemampuan nasabah untuk melunasi pinjaman, maka dilakukan proses **feature encoding** menggunakan **Ordinal Encoding**.

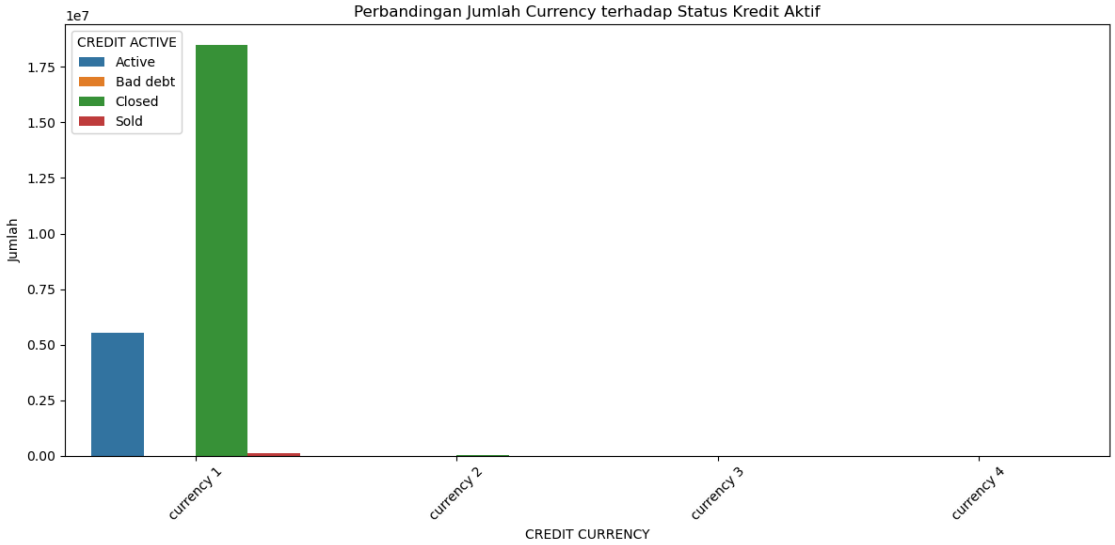
Ordinal Encoding sensitif terhadap urutan kategori. Dalam hal ini, urutan terkecil akan diisi oleh kategori dengan risiko gagal bayar terendah, sementara urutan terbesar akan diisi oleh kategori dengan risiko gagal bayar tertinggi. Oleh karena itu, dalam proses **feature encoding** menggunakan **Ordinal Encoding**, ditetapkan sebagai berikut:

- ➢ Status kredit **Active** diberi nilai **1**, yang diasumsikan sebagai kategori dengan risiko gagal bayar terendah.
- ➢ Status kredit **Closed** diberi nilai **2**.
- ➢ Status kredit **Sold** (kredit dialihkan kepemilikan) diberi nilai **3**.
- ➢ Status kredit **Bad debt** diberi nilai **4**, yang diasumsikan sebagai kategori dengan risiko gagal bayar tertinggi.

2. CREDIT_CURRENCY

Pada kolom **CREDIT_CURRENCY**, terdapat beberapa kategori yang menjelaskan jenis mata uang yang digunakan dalam transaksi kredit atau pinjaman, yaitu **currency 1**, **currency 2**, **currency 3**, dan **currency 4**. Namun, tidak ada penjelasan lebih lanjut mengenai mata uang yang dimaksud

untuk setiap kategori. Oleh karena itu, dilakukan analisis cepat untuk melihat frekuensi penggunaan masing-masing mata uang (currency) pada status kredit nasabah. Berikut hasil perbandingan antara currency dengan status kredit nasabah :



Dapat dilihat dari grafik bahwa kredit nasabah hanya tergolong pada mata uang currency 1, sedangkan kredit yang menggunakan currency 2, currency 3, dan currency 4 sangat kecil atau hampir tidak ada. Currency 1 didominasi oleh jumlah kredit dalam kategori Closed (ditunjukkan oleh diagram batang berwarna hijau), sementara status Active (ditunjukkan oleh diagram batang berwarna biru) memiliki jumlah yang signifikan, meskipun jauh lebih kecil dibandingkan dengan status Closed.

Oleh karena itu, dalam proses feature encoding, **currency 1 akan dikonversi menjadi nilai 1**, sedangkan **currency 2 sampai currency 4** akan dieliminasi dengan **diubah nilainya menjadi 0**. Dengan cara ini, diharapkan permodelan menjadi lebih sederhana dan efektif.

3. STATUS

Pada kolom **STATUS**, terdapat beberapa kategori yang menjelaskan pembaruan status pinjaman nasabah dalam periode bulanan. Kategori pada kolom **STATUS** meliputi: **C**, **X**, **0**, **1**, **2**, **3**, **4**, dan **5**. Dalam konteks ini, status **C** (Closed), **X** (status yang tidak terdefinisi), dan **0** (tidak ada tunggakan) menunjukkan bahwa pinjaman tidak memiliki masalah yang perlu penanganan khusus.

Oleh karena itu, dalam proses feature encoding, nilai **C**, **X**, dan **0** diubah menjadi **0**. Sedangkan status **1**, **2**, **3**, **4**, dan **5** akan dipertahankan sebagai informasi konkrit mengenai lamanya waktu tunggakan pinjaman nasabah.

4. **CREDIT_TYPE**

Pada kolom **CREDIT_TYPE**, terdapat berbagai kategori yang menggambarkan jenis-jenis kredit atau pinjaman yang dimiliki oleh nasabah, seperti pinjaman jangka panjang untuk membeli kendaraan, pinjaman jangka panjang untuk membeli properti, dan lain-lain. Kategori dalam CREDIT_TYPE cukup banyak, mencapai 14 kategori.

Untuk fitur atau kolom dengan banyak kategori, penggunaan One-Hot Encoding tidaklah efektif dalam proses feature encoding. Hal ini dikarenakan One-Hot Encoding akan membuat kolom baru yang merepresentasikan masing-masing kategori secara terpisah, sehingga akan terjadi penambahan kolom yang signifikan. Kondisi ini dapat membuat model menjadi lebih kompleks dan rumit, yang bisa berdampak negatif pada performa model dan waktu pelatihan.

Oleh karena itu digunakan metode **Frequency Encoding** untuk proses encoding pada feature CREDIT_TYPE. Frequency Encoding adalah metode yang baik untuk menekan penambahan jumlah fitur sambil memberikan informasi mengenai seberapa umum setiap kategori.

Metode Frequency Encoding menggantikan setiap kategori dengan frekuensi kemunculannya dalam dataset, sehingga membantu mengurangi dimensi dibandingkan dengan One-Hot Encoding, yang akan membuat kolom baru untuk setiap kategori unik. Frequency Encoding, menjaga kompleksitas model tetap rendah dan memanfaatkan informasi yang relevan dari data kategorikal.

Setelah dilakukan proses feature encoding, akan dibuat kolom baru **CREDIT_TYPE_Encoded** yang berisi nilai frekuensi kemunculan masing-masing kategori dalam kolom CREDIT_TYPE. Dengan cara ini, setiap kategori pinjaman akan direpresentasikan oleh nilai numerik yang mencerminkan seberapa sering kategori tersebut muncul dalam dataset.

Fitur CREDIT_TYPE akan tetap dibiarkan ada untuk keperluan analisis awal dan pemahaman terhadap data. Namun, saat proses pemodelan sistem dilakukan, akan lebih baik untuk menghapus kolom CREDIT_TYPE dan

hanya menggunakan kolom CREDIT_TYPE_Encoded. Pendekatan ini diharapkan dapat menyederhanakan model dan meningkatkan efisiensi, serta mencegah kompleksitas yang tidak perlu dalam analisis lebih lanjut.

1.2.6. Handle Class Imbalance

Proses ***handling class imbalance*** pada dataset Bureau tidak diterapkan untuk semua fitur atau kolom. Secara umum, penanganan class imbalance hanya berlaku untuk kolom target klasifikasi. Kolom target klasifikasi merupakan variabel yang ingin diprediksi dan sering kali memiliki distribusi yang tidak seimbang, yang dapat memengaruhi kinerja model pembelajaran mesin.

Pada kali ini, proses handling class imbalance hanya dilakukan pada kolom **CREDIT_DAY_OVERDUE.** Dalam hal ini CREDIT_DAY_OVERDUE merupakan prediktor penting, karena kolom tersebut mencerminkan jumlah hari keterlambatan pembayaran, yang merupakan indikator langsung dari kemampuan nasabah untuk membayar pinjaman. Sedangkan distribusi data untuk kolom CREDIT_DAY_OVERDUE sebagai berikut :

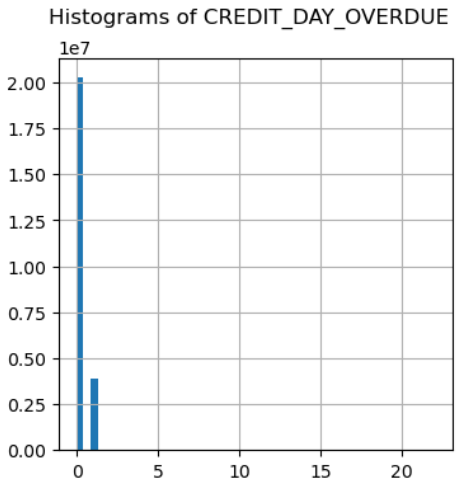
-0.045408 99.794233
22.022395 0.205767

Ini bermakna 99,79% dari data bernilai -0,045 , sedangkan hanya 0,21% dari data bernilai 22,022 . Dari distribusi ini, terlihat bahwa 99,79% nasabah memiliki keterlambatan pembayaran yang sangat sedikit atau tidak ada, sementara hanya segelintir nasabah yang memiliki keterlambatan yang lebih signifikan. Penanganan class imbalance CREDIT_DAY_OVERDUE dipilih menggunakan teknik **SMOTE (*Synthetic Minority Over-sampling Technique*)** untuk menambah jumlah contoh dari kelas minoritas (nasabah yang terlambat). Ini membantu model untuk belajar dari lebih banyak data keterlambatan.

Proses *handling class imbalance* tidak diterapkan pada fitur lainnya karena sebagian besar distribusi data pada fitur atau kolom lain merupakan variabel kontinu yang tidak memiliki kategori yang terpisah secara jelas. Sebagian besar kolom ini berfungsi sebagai fitur prediktor yang mendukung model dalam memberikan informasi tambahan mengenai karakteristik nasabah, seperti perilaku kredit dan status pembayaran.

Kolom-kolom seperti **DAYS_CREDIT**, **AMT_CREDIT_SUM**, dan **DAYS_CREDIT_ENDDATE** adalah contoh variabel kontinu yang memberikan informasi numerik. Variabel ini tidak memiliki label klasifikasi yang dapat dikategorikan sebagai kelas mayoritas atau minoritas. Oleh karena itu, penerapan

teknik penanganan class imbalance, seperti oversampling atau undersampling, tidak relevan dan tidak akan memberikan manfaat dalam konteks analisis ini. Berikut ini hasil proses ***handling class imbalance*** pada kolom **CREDIT_DAY_OVERDUE :**



Hasil distribusi setelah proses ***handling class imbalance*** sebagai berikut :

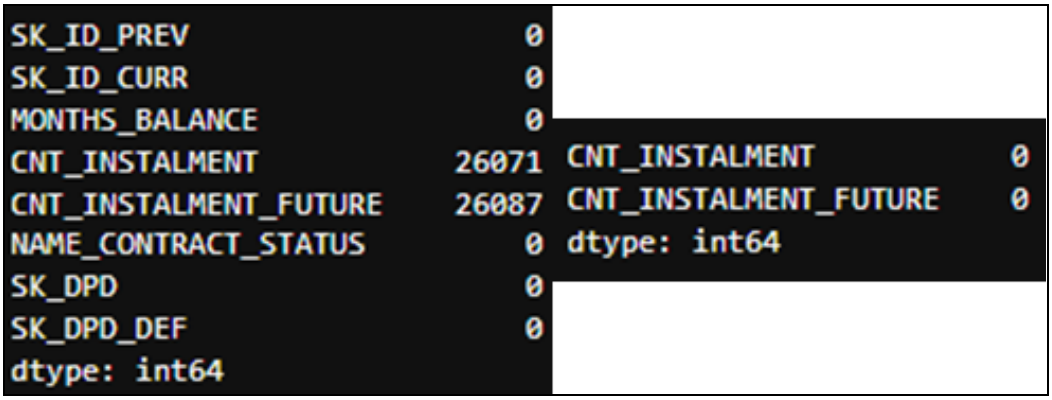
CREDIT_DAY_OVERDUE	
0.000000	15442268 :
-0.045408	4825888
1.000000	3901524
22.022395	10061

Ini berarti terdapat **15.442.268** entri di mana nilai CREDIT_DAY_OVERDUE adalah **0**, **4.825.888** entri dengan nilai sekitar **-0,045408, 3.901.524** entri dengan nilai **1,** dan **10.061 entri** dengan nilai **22,022395.** Hasil SMOTE dalam kasus ini tidak menunjukkan peningkatan keseimbangan yang diinginkan. Namun hasil ini akan diterima terlebih dahulu, dengan pertimbangan adanya class imbalance pada kolom CREDIT_DAY_OVERDUE akan menjadi perhatian khusus pada proses lanjutan.

1.3. HC_POS_CASH_balance | credit_card_balance | instalments_payments

1.3.1. Handle Missing Value

POS_CASH_balance



Gambar - Sebelum dan sesudah missing value handling

Untuk menangani missing values dalam dataset ini, ditemukan bahwa kolom CNT_INSTALMENT dan CNT_INSTALMENT_FUTURE masing-masing memiliki nilai hilang sebanyak 26.071 dan 26.087. Menangani nilai hilang secara tepat adalah langkah yang penting untuk menjaga integritas dan konsistensi data.

Langkah pertama yang dilakukan adalah melakukan eksplorasi terhadap potensi sumber data lain yang relevan. Berdasarkan diagram hierarki data, terdapat tiga tabel tambahan, yaitu credit_card_balance, installments_payments, dan previous_application, yang dapat digunakan untuk mengisi nilai hilang di kedua kolom tersebut. Ketiga tabel ini memiliki keterkaitan dengan tabel utama sehingga berpotensi menyajikan informasi tambahan yang relevan untuk melengkapi data yang hilang. Namun, setelah melakukan pemeriksaan lebih lanjut, tidak ditemukan fitur yang secara langsung sesuai atau dapat di-join untuk menggantikan missing values pada kolom CNT_INSTALMENT dan CNT_INSTALMENT_FUTURE.

Oleh karena itu, metode alternatif yang dipilih adalah dengan mengisi missing values menggunakan nilai rata-rata (mean) dari masing-masing kolom. Metode ini dipilih karena merupakan pendekatan yang umum digunakan untuk menjaga keutuhan data tanpa mengubah distribusi data secara signifikan. Pengisian dilakukan menggunakan fungsi fillna() pada Python, dengan mengganti nilai hilang pada kolom CNT_INSTALMENT dan CNT_INSTALMENT_FUTURE dengan rata-rata nilai dari masing-masing kolom tersebut.

Credit Card Balance

Terdpat feature yang jumlah missing value nya cukup banyak. Berikut kolom-kolom tersebut :

* AMT_DRAWINGS_ATM_CURRENT 605754
* AMT_DRAWINGS_OTHER_CURRENT 605754
* AMT_DRAWINGS_POS_CURRENT 605754

* AMT_INST_MIN_REGULARITY	264384
* CNT_DRAWINGS_ATM_CURRENT	605754
* CNT_DRAWINGS_OTHER_CURRENT	605754
* CNT_DRAWINGS_POS_CURRENT	605754
* CNT_INSTALMENT_MATURE_CUM	264384

Missing Value biasanya disebabkan karena user tidak menulis data tersebut. Missing Value harus diisini dengan nilai karena untuk memperbaiki data sehingga interpretsi data na menjadi lebih baik. Penangangan Missing Value ini teratasi dengan memberikan nilai median pada missing values tersebut untuk setiap kolom. Hal ini dikarenakan median tidak tergantung pada keseluruhan nilai (robust terhadap outliers). Missing Value

Instalments_payments

Ditemukan sebanyak 2583 missing values pada kolom "DAYS_ENTRY_PAYMENT" dan "AMT_PAYMENT". Sehingga diperlukan penanganan missing values yang diisi dengan median dari masing-masing fitur.

1.3.2. Handle Duplicated Data

POS_CASH_balance

Tahap selanjutnya adalah deteksi dan penanganan duplikasi data. Data yang terduplikasi dapat menyebabkan bias dan menurunkan kualitas model jika tidak ditangani. Oleh karena itu, dilakukan pengecekan terhadap seluruh baris data untuk memastikan tidak ada baris yang identik. Berdasarkan pengecekan, tidak ditemukan adanya duplikasi data dalam dataset ini. Dengan demikian, tidak diperlukan tindakan lebih lanjut untuk menangani duplikasi data.

Credit Card Balance

Pada dataset ini, tidak ditemukan duplicated data. Maka dari itu tidak perlu dilakukan penanganan lebih lanjut.

Instalments_payments

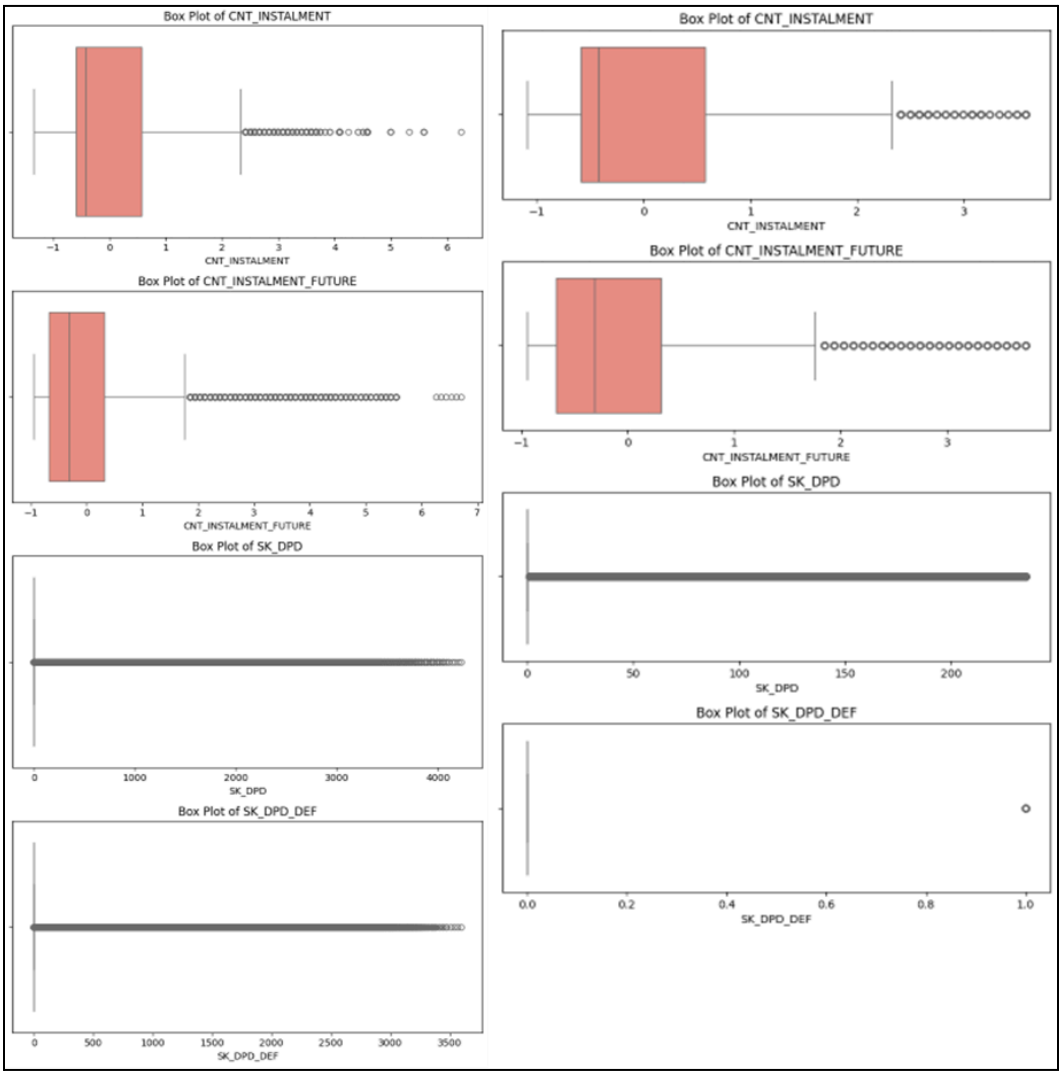
Pada dataset ini, tidak ditemukan adanya data duplikat. Sehingga tidak perlu ada penanganan lebih lanjut

1.3.3. Handle Outliers

POS_CASH_balance

Outliers adalah nilai yang ekstrem dan berbeda jauh dari sebagian besar nilai dalam dataset. Outliers dapat memengaruhi performa model secara signifikan, terutama untuk model berbasis regresi atau model yang sensitif

terhadap nilai ekstrim. Identifikasi outliers dilakukan menggunakan boxplot dan distribusi data untuk feature yang memiliki skewness tinggi, seperti CNT_INSTALMENT, CNT_INSTALMENT_FUTURE, SK_DPD dan SK_DPD_DEF.



Gambar - Sebelum dan sesudah handling outliers

Pendekatan yang digunakan untuk menangani outliers adalah metode capping pada persentil ke-1 dan ke-99. Nilai di bawah persentil ke-1 diubah menjadi nilai pada persentil ke-1, dan nilai di atas persentil ke-99 diubah menjadi nilai pada persentil ke-99. Langkah ini mengurangi dampak dari nilai yang sangat ekstrem tanpa menghilangkan data secara keseluruhan.

Credit Card Balance

Outliers adalah nilai yang sangat berjauhan dengan data umumnya. Ini dapat mempengaruhi kinerja suatu algoritma atau model. Terdapat jumlah outliers

pada dataset Credit_Card_Balance sebesar 3227965. Hal ini perlu dilakukan handling untuk mendapatkan data yang rapi agar nantinya model dapat menghasilkan kinerja yang sangat baik. Jika tidak dilakukan penanganan ini maka dapat meningkatkan pengaruh negatif pada algoritma yang sensitif terhadap outliers. Penangananya bisa dilakukan dengan menghapus outliers. Setelah dilakukan handling outliers, maka didapatkan Jumlah baris setelah memfilter outlier sebesar 2213965.

Instalments_payments

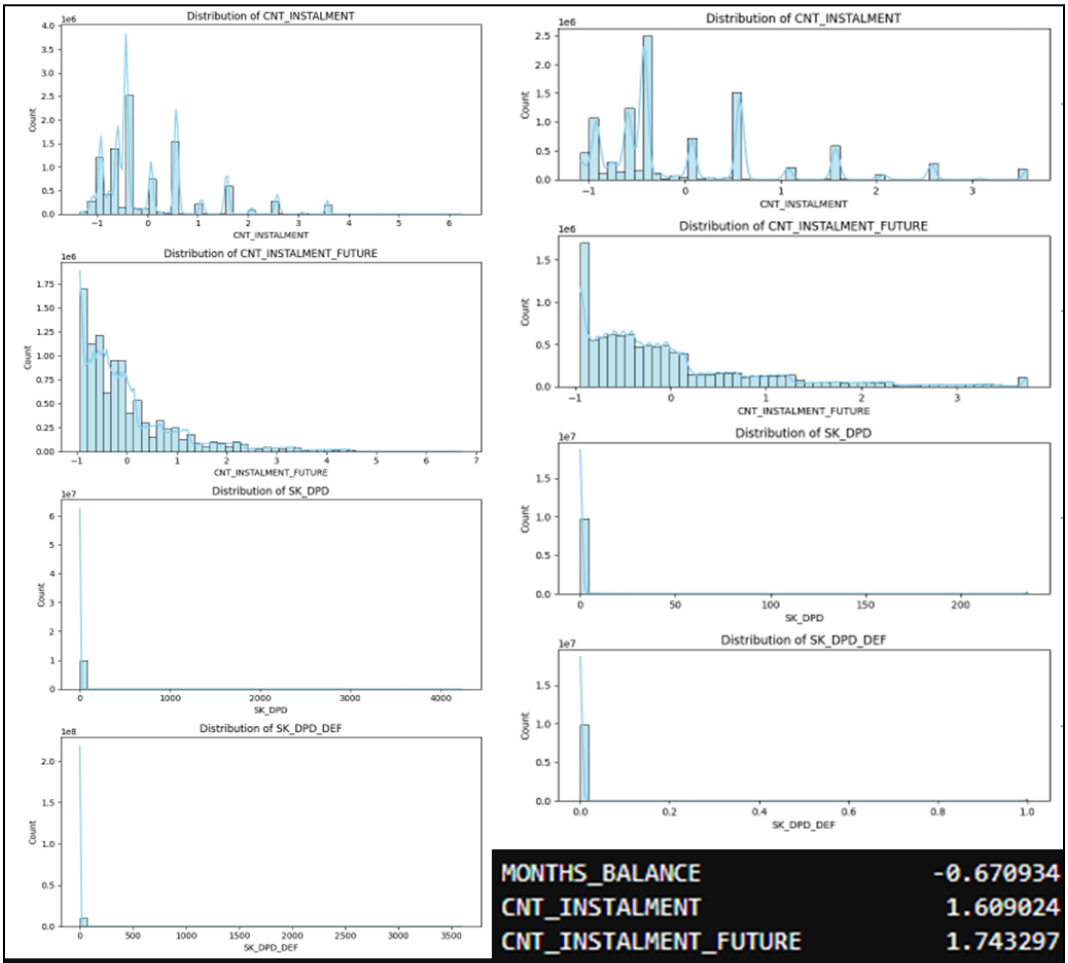
Dilakukan handling outlier pada data numerik seperti fitur DAYS_INSTALMENT, DAYS_ENTRY_PAYMENT, AMT_INSTALMENT, dan AMT_PAYMENT. Ada sebanyak 1342648 baris data outlier yang ditemukan dan data yang bersih menjadi sebanyak 10246361 dari 11589009 baris data.

1.3.4. Feature Transformation

POS_CASH_balance

Untuk mengurangi skewness (kekurangan simetri distribusi data) pada beberapa feature, dilakukan transformasi logaritma pada feature dengan distribusi yang sangat skewed. Transformasi ini bertujuan untuk membuat data lebih mendekati distribusi normal dan membantu meningkatkan kinerja model prediksi. Feature yang ditransformasi meliputi CNT_INSTALMENT dan

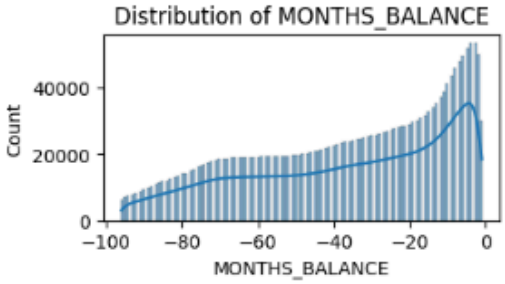
CNT_INSTALMENT_FUTURE.



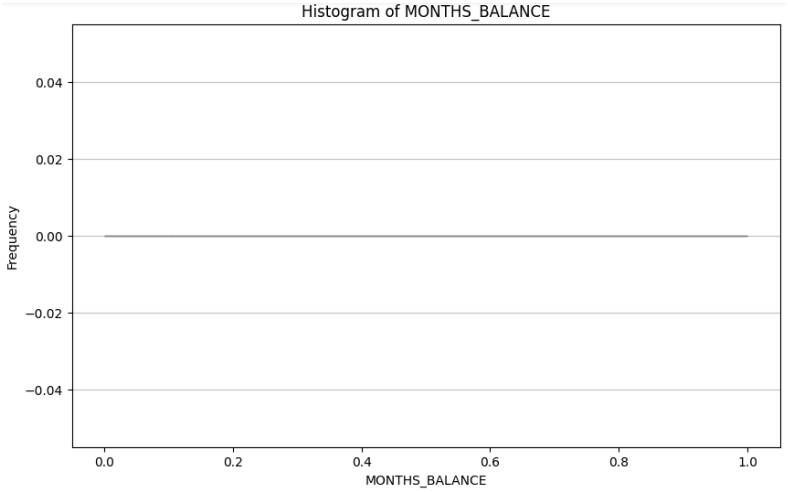
Gambar - Sebelum dan sesudah feature transformation

Transformasi logaritma dilakukan dengan menambahkan nilai kecil (epsilon) untuk menghindari masalah nilai nol dalam transformasi log. Langkah ini berguna terutama untuk model yang mengasumsikan distribusi data normal.

Credit Card Balance



Pada kolom MONTHS_BALANCE terjadi skewed, ini dapat menyebabkan penurunan akurasi model. Dengan begitu perlu dilakukan log transformation agar data terdisbusi dengan simetris dan juga mengurangi dampak nilai ekstrem terhadap model sehingga model menjadi lebih robust terhadap variasi yang tidak diinginkan.



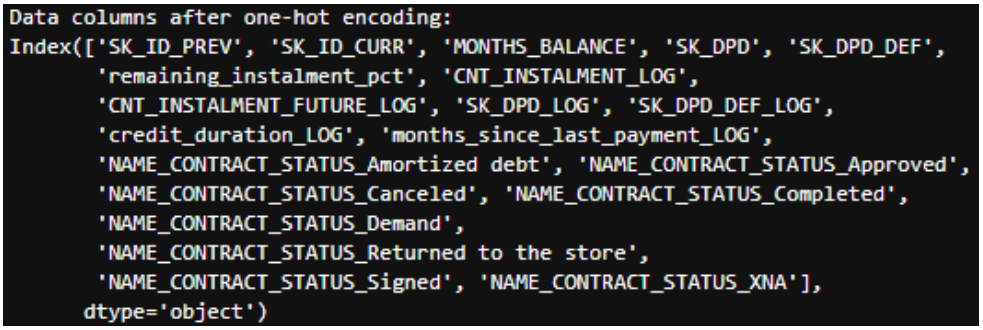
Terlihat dari grafik sebaran data MONTHS_BALANCE sudah terdisbusi normal dengan menggunakan teknik log tranformation.

Instalments_payments

Pada kolom AMT_INSTALMENT, AMT_PAYMENT, dan DAYS_ENTRY_PAYMENT_RATIO perlu dilakukan transformasi log karena distribusi feature tersebut adalah right skew sehingga distribusi data tersebut bisa lebih normal. Selain itu, pada feature DAYS_INSTALMENT, DAYS_ENTRY_PAYMENT, dan AMT_PAYMENT_DIFFERENCES perlu dilakukan standarization agar data tersebut mempunyai skala yang mirip dengan data lainnya.

1.3.5. Feature Encoding

POS_CASH_balance



Gambar - One-Hot Encoding

Feature kategorikal, seperti NAME_CONTRACT_STATUS, tidak dapat langsung digunakan oleh algoritma machine learning berbasis numerik. Oleh karena itu, dilakukan encoding terhadap feature ini menggunakan One-Hot Encoding. Metode ini mengubah setiap kategori dalam feature menjadi kolom biner, yang bernilai 1 jika kategori tersebut ada, dan 0 jika tidak. Dengan one-hot encoding, informasi dalam feature kategorikal tetap dipertahankan tanpa menambahkan bias ordinal yang tidak diinginkan.

Credit Card Balance

Karena terdapat koom yang bersifat kegorikal maka perlu dilakukan one hot encoding. Hal ini bertujuan untuk dapat diinterpretasikan dalam algoritma machine learning, seperti regresi linier, regresi logistik, dan algoritma berbasis jarak (seperti k-NN) yang mana tidak dapat menangani variabel kategorikal secara langsung.

One-hot encoding mengubah variabel kategorikal menjadi format numerik sehingga dapat digunakan dalam model tersebut.

Instalments_payments

Pada dataset ini, tidak ada feature kategorikal non ordinal. Feature seperti NUM_INSTALMENT_VERSION dan NUM_INSTALMENT_NUMBER adalah feature kategorikal ordinal sehingga tidak perlu ada encoding. Selain itu feature TARGET berbentuk True/False dan sudah terlabel encoding.

1.3.6. Handle Class Imbalance

1.4. HC_previous_application

1.4.1. Handle Missing Value

```
Kolom yang terdapat missing value:
AMT_ANNUITY                  372235
AMT_CREDIT                        1
AMT_DOWN_PAYMENT             895844
AMT_GOODS_PRICE              385515
RATE_DOWN_PAYMENT            895844
RATE_INTEREST_PRIMARY       1664263
RATE_INTEREST_PRIVILEGED    1664263
NAME_TYPE_SUITE              820405
CNT_PAYMENT                  372230
PRODUCT_COMBINATION             346
DAYS_FIRST_DRAWING           673065
DAYS_FIRST_DUE               673065
DAYS_LAST_DUE_1ST_VERSION    673065
DAYS_LAST_DUE                673065
DAYS_TERMINATION             673065
NFLAG_INSURED_ON_APPROVAL    673065
dtype: int64
```

```
#Check missing values
check_missing_values(prev_app)
```

```
Tidak ada missing value di dataset.
```

Gambar. Sebelum dan sesudah handle missing value

Fitur kategori 'NAME_TYPE_SUITE' yang memiliki nilai kosong, diisi dengan kategori yang sudah ada yaitu ‘Unaccompanied’. Hal ini dikarenakan fitur tersebut menunjukkan tentang orang yang menemani client saat apply previous application, sehingga sangat masuk akal bahwa nilai kosong tersebut diisi dengan kategori yang menunjukkan tidak ditemani siapapun.

Fitur kategori ‘PRODUCT_COMBINATION’ merupakan fitur yang menunjukkan kombinasi produk client saat pengajuan previous application, sehingga kategori yang tepat diisi yaitu dengan nilai modus atau nilai yang paling banyak.

Fitur ‘NFLAG_INSURED_ON_APPROVAL’ diisi dengan nilai 0, karena fitur tersebut menunjukkan tentang client meminta asuransi saat previous application atau tidak. Nilai 0 untuk tidak dan nilai 1 tanda untuk meminta asuransi. Nilai yang masuk akal untuk data kosong pada fitur tersebut adalah nilai 0, yaitu tidak meminta asuransi.

Fitur yang tersisa merupakan data numerikal, sehingga data yang tidak ada value diisi dengan nilai median karena pada analisis EDA sebelumnya menunjukkan bahwa distribusi data-data numerikal pada dataset tidak normal.

1.4.2. Handle Duplicated Data

Tidak perlu handle duplicated data karena tidak ada yang duplicate.

1.4.3. Handle Outliers

Handle Outliers menggunakan IQR karena data memiliki banyak outliers ekstrem. Handling outliers digunakan untuk data numerikal murni. Pertama-tama definisikan terlebih dahulu kolom numerikal, kategorikal linear, kategorikal non-linear, dan kolom yang akan didelete untuk memudahkan proses selanjutnya. Pada tahap ini, data yang berjumlah 1.670.212 tereduksi menjadi 536.201 row.

1.4.4. Feature Transformation

Sebelum transformasi data numerik, dilihat terlebih dahulu distribusi data untuk menentukan metode yang tepat.

```
AMT_ANNUITY: Right-skewed
AMT_APPLICATION: Right-skewed
AMT_CREDIT: Right-skewed
AMT_DOWN_PAYMENT: Normal
AMT_GOODS_PRICE: Right-skewed
DAYS_DECISION: Left-skewed
DAYS_FIRST_DRAWING: Normal
DAYS_FIRST_DUE: Normal
DAYS_LAST_DUE_1ST_VERSION: Normal
DAYS_LAST_DUE: Normal
DAYS_TERMINATION: Left-skewed
```

Gambar. Tipe Distribusi pada Data Numerik

Data yang memiliki distribusi Left-skewed dilakukan transformasi data dengan MinMaxScaler, distribusi Normal dengan standardized, dan distribusi Right-skewed dengan metode Log Transformation. Pada metode Log Transformation memakai natural logaritma (1+x) atau Log1p, untuk memastikan value dengan nilai 0 dapat terhandle.

1.4.5. Feature Encoding

Data dengan feature kategorikal semua diubah menjado One Hot Encoding, sehingga fitur bertambah menjadi 148 fitur.

1.4.6. Handle Class Imbalance

Class imbalance dihandle ketika semua dataset sudah tergabung.

II. Feature Engineering

2.1. HC_application_train

2.1.1. Feature Selection

1. **Correlation Analysis**: Analisis korelasi digunakan untuk menghapus fitur yang memiliki korelasi tinggi dengan fitur lain untuk menghindari redundansi.
2. **Tujuan**: Seleksi fitur meningkatkan efisiensi model dengan menghilangkan fitur yang tidak relevan atau redundan, sehingga model menjadi lebih cepat dan akurat.

Setelah dilakukan pengecekan, ada beberapa fitur yang tidak masuk ke dalam Feature Selection yaitu sebagai berikut.

- AMT_ANNUITY
- AMT_GOODS_PRICE
- AMT_CREDITCNT_CHILDREN
- REG_REGION_NOT_WORK_REGION
- REGION_RATING_CLIENT_W_CITY
- REG_CITY_NOT_WORK_CITY
- DEF_60_CNT_SOCIAL_CIRCLE
- OBS_60_CNT_SOCIAL_CIRCLE
- FLAG_MOBIL
- FLAG_DOCUMENT_12

2.1.2. Feature Extraction

1. **Polynomial Features**: Fitur interaksi atau polinomial dibuat untuk menangkap hubungan non-linear antara variabel, misalnya dengan mengalikan dua fitur.
2. **Aggregations**: Fitur agregat, seperti mean, sum, min, max, dibuat dari kumpulan fitur yang berhubungan. Ini membantu memberikan informasi tambahan yang relevan dari fitur yang ada.
3. **Tujuan**: Ekstraksi fitur menambah informasi yang relevan untuk meningkatkan akurasi model tanpa perlu menambah data baru.

Beberapa fitur ekstraksi yang dilakukan antara lain :

- EXT_SOURCE_MEAN, yaitu menghitung rata-rata dari kolom EXT_SOURCE_1, EXT_SOURCE_2 dan EXT_SOURCE_3.
- NUM_DOCUMENTS, yaitu menghitung total dokumen keseluruhan yang ada (yang berasal dari FLAG_DOCUMENT).

- IS_WEEKEND_APPR_PROCESS_START, mengecek apakah proses dimulainya pengajuan pada akhir pekan atau tidak.

2.2. HC_bureau | bureau_balance

2.2.1. Feature Extraction

Dalam proses analisis dataset Bureau, dilakukan penambahan kolom baru yang dinamakan **CREDIT_TYPE_Encoded**. Kolom ini dihasilkan dari penerapan teknik **feature encoding** pada kolom **CREDIT_TYPE**, yang merupakan kolom kategorikal yang menunjukkan jenis kredit yang dimiliki oleh nasabah. **CREDIT_TYPE_Encoded** berisi nilai frekuensi kemunculan masing-masing kategori dalam kolom CREDIT_TYPE. Dengan cara ini, setiap kategori pinjaman akan direpresentasikan oleh nilai numerik yang mencerminkan seberapa sering kategori tersebut muncul dalam dataset.

2.2.2. Feature Selection

Pada tahapan *feature selection*, digunakan beberapa cara untuk membuat keputusan, antara lain metode *Correlation Matrix Spearman, Feature Importance*, dan RFE (*Recursive Feature Elimination*). Proses *feature selection* ini bertujuan untuk mengidentifikasi fitur-fitur yang paling berkontribusi terhadap model prediksi, sehingga meningkatkan akurasi dan interpretabilitas model. Berikut ini penjelasan masing-masing metode *feature selection* :

1. RFE (*Recursive Feature Elimination*)

Fitur yang dipilih oleh RFE :
Index(['SK_ID_CURR','SK_ID_BUREAU','CREDIT_ACTIVE','CREDIT_CURRENCY','DAYS_CREDIT','DAYS_CREDIT_ENDDATE','DAYS_ENDDATE_FACT','AMT_CREDIT_MAX_OVERDUE','CNT_CREDIT_PROLONG','AMT_CREDIT_SUM','AMT_CREDIT_SUM_DEBT','AMT_CREDIT_SUM_LIMIT','AMT_CREDIT_SUM_OVERDUE','DAYS_CREDIT_UPDATE','AMT_ANNUITY','MONTHS_BALANCE','STATUS','CREDIT_TYPE_Encoded'],dtype='object')

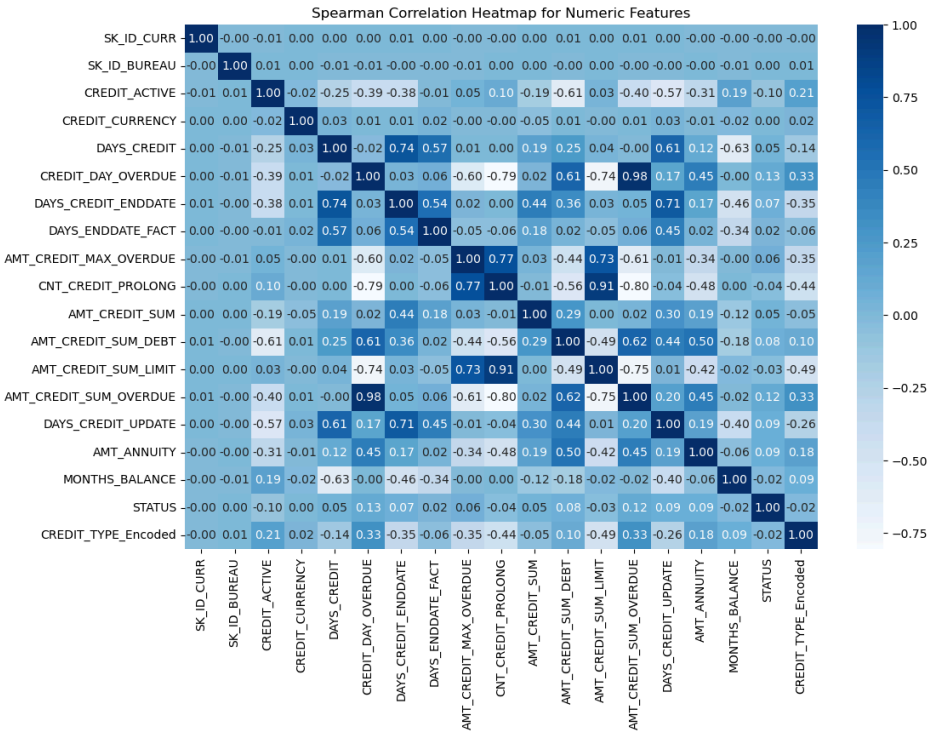
Ranking fitur: {
'SK_ID_CURR': 1,
'SK_ID_BUREAU': 1,
'CREDIT_ACTIVE': 1,
'CREDIT_CURRENCY': 1,
'DAYS_CREDIT': 1,
'DAYS_CREDIT_ENDDATE': 1,
'DAYS_ENDDATE_FACT': 1,

'AMT_CREDIT_MAX_OVERDUE': 1,
'CNT_CREDIT_PROLONG': 1,
'AMT_CREDIT_SUM': 1,
'AMT_CREDIT_SUM_DEBT': 1,
'AMT_CREDIT_SUM_LIMIT': 1,
'AMT_CREDIT_SUM_OVERDUE': 1,
'DAYS_CREDIT_UPDATE': 1,
'AMT_ANNUITY': 1,
'MONTHS_BALANCE': 1, 'STATUS': 1,
'CREDIT_TYPE_Encoded': 1}

Hasil dari proses *feature selection* menggunakan RFE menunjukkan bahwa semua fitur dalam dataset dianggap relevan untuk prediksi. Semua fitur yang dipilih oleh RFE memiliki ranking 1, yang berarti bahwa fitur-fitur tersebut telah terpilih sebagai fitur terpenting untuk model. Namun, penting untuk diingat bahwa penggunaan semua fitur dapat menyebabkan kompleksitas model yang lebih tinggi dan potensi overfitting.

2. *Correlation Matrix Spearman*

Berikut ini hasil proses feature selection menggunakan ***Correlation Matrix Spearman*** **:**



Pada permodelan sistem Home Credit Default Risk ini, permodelan sistem bertujuan untuk memprediksi kemampuan nasabah untuk melunasi pinjaman,

Pada dataset Bureau ini, maka TARGET atau feature yang relevant pada dataset Bureau sesuai tujuan tersebut adalah **CREDIT_DAY_OVERDUE.**

CREDIT_DAY_OVERDUE secara langsung mengukur jumlah hari keterlambatan dalam pembayaran pinjaman. Ini memberikan informasi yang jelas tentang sejauh mana nasabah gagal memenuhi kewajibannya. Semakin tinggi nilai ini, semakin besar kemungkinan nasabah tidak mampu melunasi pinjaman tepat waktu. Adapun hasil korelasi antar features menggunakan *Correlation Matrix Spearman* :

➢ **Korelasi Sangat Kuat (Strong Correlation)**
Rentang: 0.8 hingga 1.0 (positif) atau -1.0 hingga -0.8 (negatif)
AMT_CREDIT_SUM_OVERDUE (0.98)

➢ **Korelasi Kuat (Strong Correlation)**
Rentang: 0.6 hingga 0.8 (positif) atau -0.6 hingga -0.8 (negatif)
CNT_CREDIT_PROLONG (-0.79)
AMT_CREDIT_SUM_LIMIT (-0.74)
AMT_CREDIT_SUM_DEBT (0.61)
AMT_CREDIT_MAX_OVERDUE (-0.60)

➢ **Korelasi Sedang (Moderate Correlation)**
Rentang: 0.4 hingga 0.6 (positif) atau -0.4 hingga -0.6 (negatif)
AMT_ANNUITY (0.45)

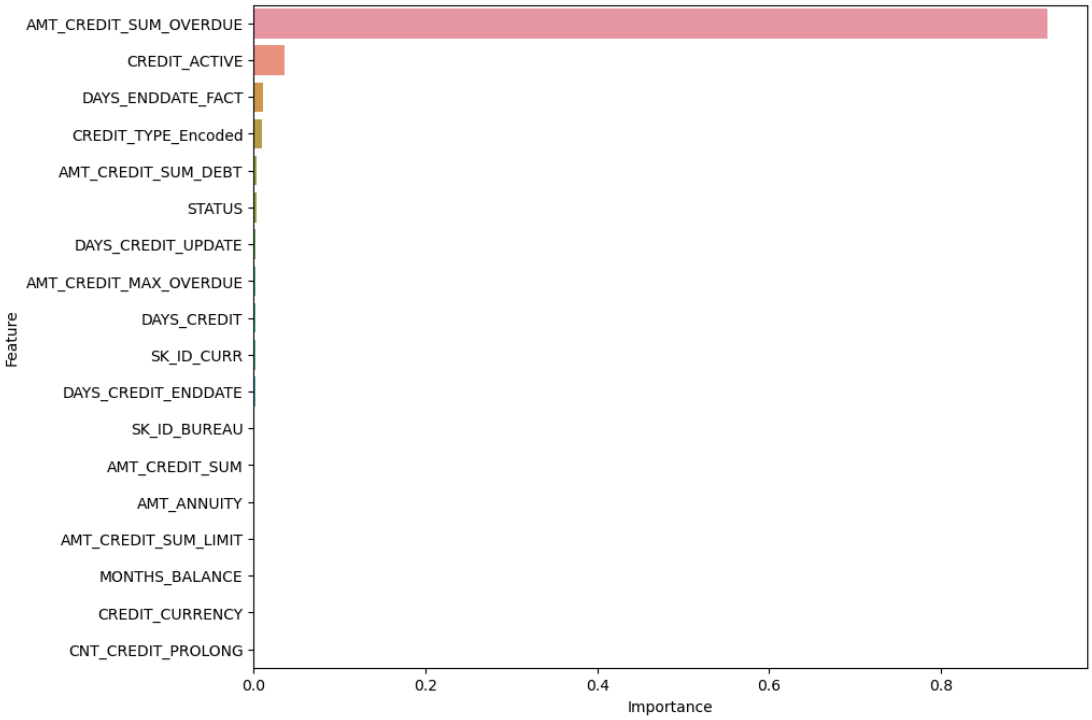
➢ **Korelasi Lemah (Weak Correlation)**
Rentang: 0.2 hingga 0.4 (positif) atau -0.2 hingga -0.4 (negatif)
CREDIT_ACTIVE (-0.39)
CREDIT_TYPE_Encoded (0.33)

➢ **Tidak Ada Korelasi (No Correlation)**
Rentang: 0 hingga 0.2 (positif) atau 0 hingga -0.2 (negatif)
DAYS_CREDIT_UPDATE (0.17)
STATUS (0.13)
DAYS_ENDDATE_FACT (0.06)
DAYS_CREDIT_ENDDATE (0.03)
SK_ID_BUREAU (-0.01)
CREDIT_CURRENCY (0.01)
DAYS_CREDIT (-0.02)
DAYS_CREDIT_ENDDATE (-0.03)
AMT_CREDIT_SUM (0.02)

MONTHS_BALANCE (-0.00)

3. *Features Importance*

Berikut ini hasil proses feature selection menggunakan ***Features Importance*:**



Feature	Importance
AMT_CREDIT_SUM_OVERDUE	0.924392
CREDIT_ACTIVE	0.035644
DAYS_ENDDATE_FACT	0.010351
CREDIT_TYPE_Encoded	0.009014
AMT_CREDIT_SUM_DEBT	0.002934
STATUS	0.002830
DAYS_CREDIT_UPDATE	0.002620
AMT_CREDIT_MAX_OVERDUE	0.002292
DAYS_CREDIT	0.002269
SK_ID_CURR	0.001601
DAYS_CREDIT_ENDDATE	0.001575
SK_ID_BUREAU	0.001235
AMT_CREDIT_SUM	0.001209
AMT_ANNUITY	0.001144
AMT_CREDIT_SUM_LIMIT	0.000556
MONTHS_BALANCE	0.000297
CREDIT_CURRENCY	0.000029
CNT_CREDIT_PROLONG	0.000007

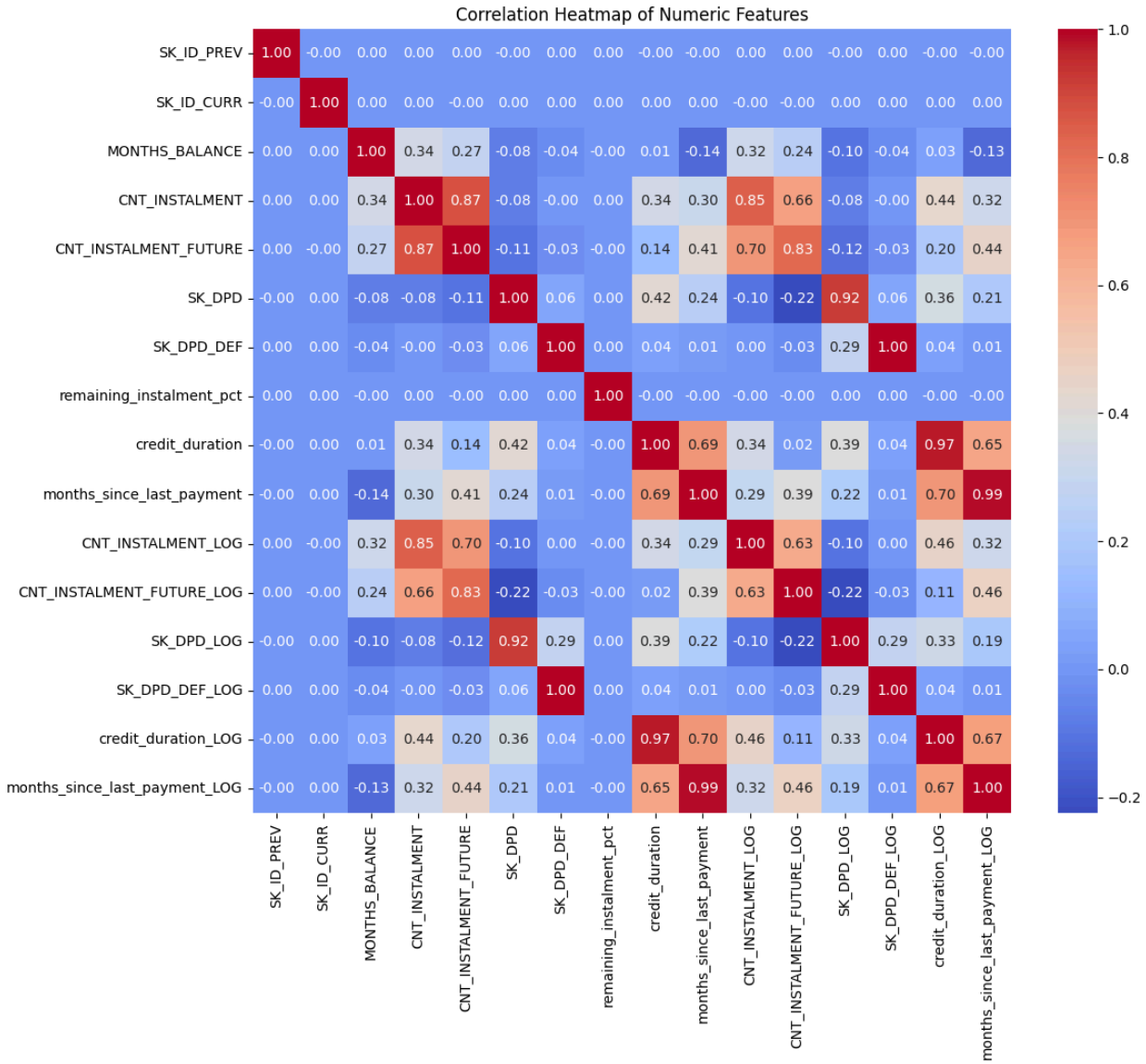
Berdasarkan hasil ketiga metode features selection yaitu *Correlation Matrix Spearman, Feature Importance*, dan RFE (*Recursive Feature Elimination*) diketahui feature atau kolom yang paling relevant dengan tujuan permodelan sistem yaitu AMT_CREDIT_SUM_OVERDUE dengan nilai korelasi sebesar 0.98 pada *correlation matrix spearman*, dan 0.92 pada f*eature importance*. Kemudian untuk feature yang akan dihapus antara lain :

1. **CREDIT_CURRENCY :** Fitur ini tidak relevan karena informasi tentang mata uang tidak memiliki dampak signifikan pada kemampuan nasabah dalam melunasi pinjaman.
2. **SK_ID_BUREAU :** ID ini hanya relevan dalam dataset Bureau dan Bureau_Balance, dan SK_ID_CURR sudah ada sebagai identifikasi nasabah dan terdapat di semua dataset. Sehingga dapat diabaikan untuk analisis ini.
3. **MONTHS_BALANCE :** Fitur ini tidak memberikan informasi tambahan yang berharga untuk model prediksi, serta tidak berkorelasi kuat dengan target.
4. **AMT_CREDIT_SUM :** Fitur ini dapat menyebabkan redundansi dengan fitur lain, seperti AMT_CREDIT_SUM_DEBT.
5. **CREDIT_TYPE :** Karena sudah ada feature CREDIT_TYPE_Encoded yang berisikan informasi tipe kredit dalam bentuk numerik.

2.3. HC_POS_CASH_balance | credit_card_balance | instalments_payments

2.3.1. Feature Selection

POS_CASH_balance



Gambar - Correlation Heatmap

Berdasarkan analisis korelasi menggunakan heatmap, ditemukan bahwa beberapa feature memiliki korelasi yang sangat tinggi satu sama lain, yaitu lebih dari 0.8. Korelasi yang tinggi antar-feature ini menunjukkan adanya potensi redundansi, di mana informasi yang diberikan oleh satu feature sebagian besar juga tercermin dalam feature lain. Untuk meningkatkan efisiensi model serta mengurangi kompleksitas data, dilakukan proses feature selection dengan menghapus feature yang dianggap redundant.

Berikut ini adalah feature yang diidentifikasi sebagai kandidat untuk dihapus berdasarkan korelasi yang kuat antar-feature:

1. **CNT_INSTALMENT_FUTURE_LOG** dan **CNT_INSTALMENT_FUTURE**: Kedua feature ini memiliki korelasi sebesar 0.83. Oleh karena itu, hanya log-transformed version dari feature ini, yaitu **CNT_INSTALMENT_FUTURE_LOG**, yang dipertahankan. Pembuangan feature ini diharapkan dapat mengurangi redundansi tanpa menghilangkan informasi penting.
2. **credit_duration** dan **credit_duration_LOG**: Korelasi antara kedua feature ini mencapai 0.97, menunjukkan bahwa hampir seluruh informasi pada **credit_duration** sudah diwakili oleh **credit_duration_LOG**. Dengan demikian, hanya **credit_duration_LOG** yang dipertahankan dalam dataset.
3. **months_since_last_payment** dan **months_since_last_payment_LOG**: Kedua feature ini memiliki korelasi sangat tinggi, yaitu 0.99. Karena nilai korelasi ini hampir identik, hanya **months_since_last_payment_LOG** yang dipertahankan sebagai representasi dari kedua feature ini.
4. **CNT_INSTALMENT_LOG** dan **CNT_INSTALMENT**: Korelasi antara kedua feature ini adalah 0.87, sehingga hanya **CNT_INSTALMENT_LOG** yang dipertahankan karena representasi log-transformed ini cenderung lebih sesuai dalam model prediksi.

Credit Card Balance

Berdasarkan heatmap, didapatkan nilai korelasi antar feature maka ada beberapa feature yang cukup menarik dan diperlukan untuk dijadikan insight yaitu feature-feature berikut ini :

1. 'MONTHS_BALANCE',
2. 'AMT_BALANCE',
3. 'AMT_CREDIT_LIMIT_ACTUAL',
4. 'AMT_DRAWINGS_ATM_CURRENT',
5. 'AMT_DRAWINGS_CURRENT',
6. 'AMT_DRAWINGS_OTHER_CURRENT',
7. 'AMT_DRAWINGS_POS_CURRENT'
8. 'AMT_INST_MIN_REGULARITY'
9. 'AMT_PAYMENT_CURRENT'
10. 'AMT_PAYMENT_TOTAL_CURRENT',
11. 'AMT_TOTAL_RECEIVABLE',
12. 'CNT_DRAWINGS_CURRENT',
13. 'NAME_CONTRACT_STATUS',
14. 'SK_DPD',

15. 'SK_DPD_DEF'

Feature- feature diatas dapat dipertahankan dan diambil insight nya untuk dapat dijadikan keputusan bisnis selanjutnya.

* Ada juga beberapa feature yang perlu dihilangkan seperti

- CNT_INSTALMENT_MATURE_CUM Dikarenakan tidak memiliki nilai korelasi dengan feature lainnya dan memiliki sifat antara feature yng artinya sama dengan feature lainnya (Redundant)
- ‘CNT_DRAWINGS_ATM_CURRENT', 'CNT_DRAWINGS_OTHER_CURRENT', 'CNT_DRAWINGS_POS_CURRENT'. Kolom-kolom ini diperlukan untuk didrop karena sudah terwakili dengan 'CNT_DRAWINGS_CURRENT' yang merupakan total keseluruhan dari kolom-kolom tersebut.
- 'AMT_RECIVABLE', 'AMT_RECEIVABLE_PRINCIPAL' Kolom kolom ini redundant terhadap kolom AMOUNT TOTAL RECEIVABLE. yang mana jika dipertahankan akan mengurangi performa model

Instalments_payments

Berdasarkan korelasi menggunakan heatmap, ada korelasi antar fitur DAYS_INSTALMENT dengan DAYS_ENTRY_PAYMENT dan AMT_INSTALMENT dengan AMT_PAYMENT. Atas rekomendasi yang ada, pada saat ini feature tersebut masih dipertahankan terlebih dahulu.

2.3.2. Feature Extraction

POS_CASH_balance

Beberapa feature baru dibuat dari feature yang sudah ada untuk memberikan informasi tambahan yang mungkin relevan untuk model prediksi. Feature tambahan ini adalah sebagai berikut:

1. **remaining_instalment_pct:** Persentase sisa angsuran, dihitung sebagai rasio antara CNT_INSTALMENT_FUTURE dengan CNT_INSTALMENT.

```
data['remaining_instalment_pct'] =
data['CNT_INSTALMENT_FUTURE'] / data['CNT_INSTALMENT']
```

2. **credit_duration:** Durasi kredit yang diukur dari awal pengajuan hingga bulan terakhir transaksi.

```
data['credit_duration'] = data['MONTHS_BALANCE'].max() -
data['MONTHS_BALANCE']
```

3. **months_since_last_payment:** Jumlah bulan sejak pembayaran terakhir dilakukan, yang dapat berguna untuk melihat ketertiban pembayaran.

```
data['months_since_last_payment'] =
data['MONTHS_BALANCE'].max() - data['MONTHS_BALANCE']
```

Instalments_payments

Jika data masih belum kompleks, bisa ditambahkan feature baru yang relevan dari feature yang sudah ada agar data bisa lebih kompleks. Berikut feature yang bisa ditambahkan:

1. DAYS_ENTRY_PAYMENT_RATIO: Rasio tanggal installment dengan tanggal bayar. (DAYS_ENTRY_PAYMENT/DAYS_INSTALMENT)
2. AMT_PAYMENT_DIFFERENCES: Selisih jumlah installment dengan jumlah yang dibayar. (AMT_PAYMENT - AMT_INSTALMENT
3. is_credit_card: Versi installment credit card atau bukan (1=Yes, 0=No)

2.3.3. Additional Feature

POS_CASH_balance

Fitur-fitur tambahan ini bisa membantu memberikan informasi lebih dalam mengenai profil risiko peminjam dan meningkatkan kinerja model dalam memprediksi target:

1. **Percentage of Late Payments:** Persentase pembayaran yang terlambat dari total pembayaran yang dilakukan. Ini dihitung sebagai jumlah entri SK_DPD atau SK_DPD_DEF yang lebih besar dari 0 dibagi dengan total jumlah pembayaran. **Percentage of Late Payments = Count(SK_DPD > 0) / Count(SK_DPD).**
2. **Average Days Past Due:** Rata-rata keterlambatan pembayaran berdasarkan kolom SK_DPD, dihitung dengan mengambil nilai rata-rata SK_DPD. Fitur ini menunjukkan pola keterlambatan pembayaran. **Average DPD = Mean(SK_DPD).**

Credit Card Balance

Feature Engineering diperlukan untuk menambah kompleksitas data agar pada saat pemodelan didapatkan hasil dengan kinerja yang paling baik. Hal-hal yang

dilakukan adalah melakukan penambahan feature baru dari feature-feature yang sudah ada. Berikut feature-feature yang perlu ditambahkan:

3. **Payment to Balance Ratio** : Rasio pembayaran terhadap saldo yang ada, yang menunjukkan seberapa baik peminjam mengelola hutangnya. Contoh: AMT_PAYMENT_TOTAL_CURRENT / AMT_BALANCE.
4. **Utilization Rate** : Rasio antara jumlah kredit yang digunakan terhadap batas kredit. Contoh: AMT_BALANCE / AMT_CREDIT_LIMIT_ACTUAL.
5. **Regularization** : Rasio antara jumlah yang harus dibayar terhadap umlah cicilan atau angsuran minimal yang dibayar secara berkala dalam jangka waktu tertentu. Contoh: AMT_PAYMENT_CURRENT / AMT_INST_MIN_REGULARITY.

2.4. HC_previous_application

2.4.1. Feature Selection

Pertama-tama hal yang dilakukan adalah drop feature dengan nilai value berupa normalisasi, yaitu pada fitur 'RATE_DOWN_PAYMENT', 'RATE_INTEREST_PRIMARY', dan 'RATE_INTEREST_PRIVILEGED'.

Kemudian, menentukan features yang penting dengan cara mengetahui korelasi antar fitur dan korelasi fitur terhadap target. Ditentukan threshold korelasi, yaitu 0,7 untuk menandakan bahwa korelasi antar fitur yang memiliki nilai korelasi di atas 0,7 akan dihapus karena menunjukkan multikolinearitas. Fitur yang berhasil tereduksi menjadi berjumlah 118.

Korelasi dengan Target pada application_train mengharuskan data digabungkan terlebih dahulu dengan application_train. Kemudian, ditentukan threshold korelasi 0,05 untuk menandakan bahwa fitur yang memiliki nilai korelasi yang di atas 0,05 terhadap target merupakan fitur penting yang harus dipertahankan. Fitur yang penting dan harus dipertahankan adalah ‘NAME_PRODUCT_TYPE_walk-in’, ‘NAME_CONTRACT_STATUS_Refused’, dan ‘CODE_REJECT_REASON_SCOFR’.

2.4.2. Feature Extraction

Fitur ekstraksi yang akan dipakai untuk selanjutnya ada pada gambar berikut.

```
filtered_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 445897 entries, 0 to 445896
Data columns (total 6 columns):
 #   Column                          Non-Null Count   Dtype
---  ------                          --------------   -----
 0   SK_ID_PREV                      445897 non-null  int32
 1   TARGET                          445897 non-null  int64
 2   NAME_PRODUCT_TYPE_walk-in       445897 non-null  int32
 3   SK_ID_CURR                      445897 non-null  int32
 4   NAME_CONTRACT_STATUS_Refused    445897 non-null  int32
 5   CODE_REJECT_REASON_SCOFR        445897 non-null  int32
dtypes: int32(5), int64(1)
memory usage: 11.9 MB
```

Data yang dihasilkan menjadi berjumlah 445.897. Karena fitur tersebut tidak ter-urut, makan diurutkan supaya fitur untuk primarykey berada di index 0 dan 1.

2.4.3. Additional Feature

1. **Loan-to Value Ratio (LTV)**: untuk mengukur rasio antara jumlah pinjaman yang diminta dan nilai aset yang dijaminkan. Rasio ini membantu menilai resiko yang diambil oleh perusahaan homecredit. Semakin tinggi rasio LTV, semakin besar resiko yang diambil oleh perusahaan karena nilai pinjaman mendekati atau melebihi nilai aset yang dijaminkan.
2. **Debt-to-Income Ratio (DTI)**: untuk mengukur rasio antara total utang bulanan dan pendapatan bulanan peminjam. Rasio ini penting untuk menilai kemampuan peminjam dalam membayar kembali pinjaman. Semakin rendah rasio DTI, semakin besar kemungkinan peminjam mampu membayar kembali pinjaman tanpa kesulitan finansial.
3. **Credit Utilization Rate**: untuk mengukur persentase dari total kredit yang tersedia yang sedang digunakan oleh peminjam. Rasio ini memberikan gambaran tentang seberapa bergantungnya peminjam pada kredit. Tingkat pemanfaatan kredit yang tinggi bisa menjadi indikasi bahwa peminjam mungkin mengalami kesulitan keuangan.
4. **Employment Stability**: untuk mengukur durasi pekerjaan terakhir atau jumlah perubahan pekerjaan dalam beberapa tahun terakhir. Stabilitas pekerjaan adalah indikator penting dari kemampuan peminjam untuk membayar kembali pinjaman. Peminjam dengan pekerjaan yang stabil

cenderung memiliki pendapatan yang lebih stabil dan dapat diandalkan atau dapat dipercaya.

5. Previous Loan Performance: untuk memberikan informasi tentang kinerja pinjaman sebelumnya, seperti apakah pinjaman sebelumnya dibayar tepat waktu atau mengalami keterlambatan pembayaran. Informasi ini sangat berguna untuk memprediksi perilaku pembayaran di masa depan. Peminjam dengan riwayat pembayaran yang baik cenderung lebih dapat diandalkan.

III. LAMPIRAN (LINK)

Repository :
https://github.com/Bramasta66/Home-Credit-Default-Risk/blob/master/Preprocessing_merged.ipynb