LAPORAN ANALISIS DATA
BABAK SEMIFINAL STC LOGIKA UI 2023

LOGIKA UI 2023

Nomor Peserta
23-03-078-9

LOMBA DAN KEGIATAN MATEMATIKA UNIVERSITAS INDONESIA
2023

BUSINESS UNDERSTANDING

A. Latar Belakang

Yobank adalah sebuah perusahaan *digital bank* yang sedang memasarkan produk terbarunya, yaitu pinjaman atau *credit loan*. Secara umum, bank harus melakukan mitigasi risiko sebelum memasarkan produknya secara luas, untuk menghindari kerugian yang tentunya tidak diinginkan. Untuk kasus *credit loan* ini, penting sekali bagi bank untuk mengetahui apakah calon *customer* layak mendapatkan *loan* atau tidak, sebagai bentuk mitigasi risiko.

Untuk memenuhi kebutuhan ini, bank telah melakukan *testing* pinjaman, dan telah mengumpulkan data aplikasi, yaitu data yang berisi informasi *customer* yang diberikan pinjaman, serta data yang berisi status pembayaran pada bulan tertentu. Dari kedua data ini, akan dibangun model yang dapat memprediksi apakah seorang *customer* termasuk "*good*" *customer* atau "*bad*" *customer*, dimana "*good*" *customer* adalah *customer* yang diprediksi untuk masih dapat membayar pinjaman sehingga akan diberikan pinjaman tersebut. Sebaliknya, "*bad*" *customer* adalah *customer* yang diprediksi tidak dapat membayar pinjaman. Untuk pengkategorian ini, didefinisikan "*bad*" *customer* sebagai *customer* yang terlambat bayar selama lebih dari 60 hari (2 bulan).

Sehingga dapat disimpulkan, tujuan dari kegiatan analisis data ini yaitu untuk mendapatkan model yang dapat memprediksi apakah seorang *customer* dapat membayar pinjaman yang diberikan, berdasarkan data aplikasi (data terkait informasi *customer*).

B. Objektif Kegiatan

Kegatan analisis data ini dibatasi pada penyusunan model *machine learning* yang dapat memprediksi "*good*" atau "*bad*" *customer* berdasarkan data aplikasi (informasi *customer*) dengan target keberhasilan yaitu *metric performance* $\geq 0{,}7$.

C. Rencana Kegiatan

Untuk memenuhi objektif dari kegiatan ini, langkah-langkah yang akan dilaksanakan yaitu:

1. Mengenali data yang akan digunakan dalam proses pembuatan model *machine learning (data understanding)*.
2. Menyiapkan data yang siap digunakan untuk pembuatan model *machince learning* (*data preparation*).
3. Melakukan *exploratory data analysis*.
4. Membuat model *machine learning* untuk memprediksi "*good*" atau "*bad*" *customer* (*modeling*).
5. Mengevaluasi model tersebut dengan metrik *performance* model *machine learning* (*evaluation*).

DATA UNDERSTANDING

Untuk kebutuhan kegiatan analisis data ini, akan digunakan data aplikasi (selanjutnya disebut data *app*) dan data yang berisi status pembayaran pada bulan tertentu (selanjutnya disebut data *credit*) yang telah tersedia. Proses *data understanding* akan dilakukan bergantian/terpisah untuk kedua data. Pada proses *data understanding* ini, akan dilakukan pengecekan ukuran data, pengecekan tipe data untuk setiap variabel, melihat pemusatan dan persebaran data, serta untuk menguji kualitas data, dilakukan pengecekan keberadaan *missing value* pada tiap variabel serta pengecekan kesamaan entri (baris duplikat).

Untuk data *app*, terdapat sebanyak 438.557 baris dan 18 kolom. Terdapat 18 variabel, dengan setiap variabel memiliki tipe data seperti terdapat pada Gambar 1. Untuk melihat pemusatan data, kami menggunakan fungsi *describe* pada python, yang dapat menampilkan berbagai ukuran pemusatan data, seperti mean, median, dan modus. Untuk persebaran data numerik menggunakan histogram dan data kategorik menggunakan *bar chart*. Dari histogram yang terbentuk pada data numerik, beberapa variabel seperti *AMT_INCOME_TOTAL*, *DAYS_EMPLOYED*, dan sebagainya memiliki *bin* yang kosong atau frekuensinya sangat kecil, sehingga dapat diduga bahwa terdapat data yang bersifat *outlier*. Hal ini juga terbukti dari *box plot* yang terbentuk pada variabel-variabel tersebut. Saat mengecek keberadaan baris duplikat, didapatkan bahwa sebanyak 348.472 baris memiliki entri yang sama dengan baris lain dengan ID *customer* berbeda. Saat dilakukan pengecekan *missing value*, diketahui bahwa pada variabel *OCCUPATION_TYPE* terdapat 134.203 *missing values*. *Missing values* pada variabel *OCCUPATION_TYPE* ini akan ditangani pada tahap *data preparation*.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 438557 entries, 0 to 438556
Data columns (total 18 columns):
 #   Column               Non-Null Count   Dtype  
---  ------               --------------   -----  
 0   ID                   438557 non-null  int64  
 1   CODE_GENDER          438557 non-null  object 
 2   FLAG_OWN_CAR         438557 non-null  object 
 3   FLAG_OWN_REALTY      438557 non-null  object 
 4   CNT_CHILDREN         438557 non-null  int64  
 5   AMT_INCOME_TOTAL     438557 non-null  float64
 6   NAME_INCOME_TYPE     438557 non-null  object 
 7   NAME_EDUCATION_TYPE  438557 non-null  object 
 8   NAME_FAMILY_STATUS   438557 non-null  object 
 9   NAME_HOUSING_TYPE    438557 non-null  object 
 10  DAYS_BIRTH           438557 non-null  int64  
 11  DAYS_EMPLOYED        438557 non-null  int64  
 12  FLAG_MOBIL           438557 non-null  int64  
 13  FLAG_WORK_PHONE      438557 non-null  int64  
 14  FLAG_PHONE           438557 non-null  int64  
 15  FLAG_EMAIL           438557 non-null  int64  
 16  OCCUPATION_TYPE      304354 non-null  object 
 17  CNT_FAM_MEMBERS      438557 non-null  float64
dtypes: float64(2), int64(8), object(8)
memory usage: 60.2+ MB
```

Gambar 1. Variabel dan tipe data untuk data *app*

	ID	CNT_CHILDREN	AMT_INCOME_TOTAL	DAYS_BIRTH	DAYS_EMPLOYED	FLAG_MOBIL	FLAG_WORK_PHONE	FLAG_PHONE	FLAG_EMAIL	CNT_FAM_MEMBERS
count	4.385570e+05	438557.000000	4.385570e+05	438557.000000	438557.000000	438557.0	438557.000000	438557.000000	438557.000000	438557.000000
mean	6.022176e+06	0.427390	1.875243e+05	-15997.904649	60563.675328	1.0	0.206133	0.287771	0.108207	2.194465
std	5.716370e+05	0.724882	1.100869e+05	4185.030007	138767.799647	0.0	0.404527	0.452724	0.310642	0.897207
min	5.008804e+06	0.000000	2.610000e+04	-25201.000000	-17531.000000	1.0	0.000000	0.000000	0.000000	1.000000
25%	5.609375e+06	0.000000	1.215000e+05	-19483.000000	-3103.000000	1.0	0.000000	0.000000	0.000000	2.000000
50%	6.047745e+06	0.000000	1.607805e+05	-15630.000000	-1467.000000	1.0	0.000000	0.000000	0.000000	2.000000
75%	6.456971e+06	1.000000	2.250000e+05	-12514.000000	-371.000000	1.0	0.000000	1.000000	0.000000	3.000000
max	7.999952e+06	19.000000	6.750000e+06	-7489.000000	365243.000000	1.0	1.000000	1.000000	1.000000	20.000000

Gambar 2. Pemusatan data pada variabel tipe numerik untuk data *app*

	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	NAME_INCOME_TYPE	NAME_EDUCATION_TYPE	NAME_FAMILY_STATUS	NAME_HOUSING_TYPE	OCCUPATION_TYPE
count	438557	438557	438557	438557	438557	438557	438557	304354
unique	2	2	2	5	5	5	6	18
top	F	N	Y	Working	Secondary / secondary special	Married	House / apartment	Laborers
freq	294440	275459	304074	226104	301821	299828	393831	78240

Gambar 3. Pemusatan data pada variabel tipe kategorik untuk data *app*

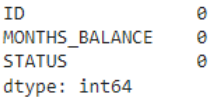
```
ID                        0
CODE_GENDER               0
FLAG_OWN_CAR              0
FLAG_OWN_REALTY           0
CNT_CHILDREN              0
AMT_INCOME_TOTAL          0
NAME_INCOME_TYPE          0
NAME_EDUCATION_TYPE       0
NAME_FAMILY_STATUS        0
NAME_HOUSING_TYPE         0
DAYS_BIRTH                0
DAYS_EMPLOYED             0
FLAG_MOBIL                0
FLAG_WORK_PHONE           0
FLAG_PHONE                0
FLAG_EMAIL                0
OCCUPATION_TYPE      134203
CNT_FAM_MEMBERS           0
dtype: int64
```

Gambar 4. Pengecekan *missing value* pada variabel untuk data *app*

Untuk data *credit*, terdapat sebanyak 1.048.575 baris dan 3 kolom. Ketidaksamaan ukuran baris dengan data *app* ini dikarenakan baris pada data *credit* bersesuaian dengan bulan peminjaman dari masing-masing *customer*, tidak seperti data *app* yang bersesuaian dengan *ID customer*. Terdapat 3 kolom, dengan setiap kolom memiliki tipe data seperti terdapat pada Gambar 5. Saat mengecek kesamaan entri, tidak ditemukan adanya entri yang sama, menandakan tidak adanya baris yang terduplikat. Pada setiap kolom juga tidak terdapat adanya *missing value* pada data *credit* ini.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1048575 entries, 0 to 1048574
Data columns (total 3 columns):
 #   Column          Non-Null Count    Dtype
---  ------          --------------    -----
 0   ID              1048575 non-null  int64
 1   MONTHS_BALANCE  1048575 non-null  int64
 2   STATUS          1048575 non-null  object
dtypes: int64(2), object(1)
memory usage: 24.0+ MB
```

Gambar 5. Variabel dan tipe data untuk data *credit*



Gambar 6. Pengecekan *missing value* pada kolom untuk data *credit*

DATA PREPARATION

Pada proses *data preparation*, dilakukan penanganan pada data duplikat pada data *app*, karena kesamaan entri tersebut dapat membuat proses *training* menjadi lama dan menurunkan variansi *dataset*. Penangangan ini dilakukan dengan mengambil baris pertama saja dari baris-baris dengan entri sama, lalu menghapus (*drop*) baris-baris lainnya. Kemudian juga dilakukan penanganan terhadap *missing value*, yaitu pada variabel *OCCUPATION_TYPE*. Karena jumlah *missing value* yang relatif banyak, kami tidak menghapus data pada *ID* dengan *missing value* tersebut, melainkan menentukan *OCCUPATION_TYPE* menggunakan klasifikasi berdasarkan *OCCUPATION_TYPE* yang telah ada. Klasifikasi yang dilakukan menggunakan algoritma *classfier multiclass* dengan *training set* yaitu data *app* tanpa *missing values* dan *testing set* yaitu data *app* dengan *missing values* yang akan diprediksi.

Selanjutnya data *app* dan data *credit* digabungkan (*merge*) berdasarkan kolom *ID*. Proses *merge* ini menghasilkan 217.273 baris × 20 kolom. Dengan menggunakan variabel *STATUS* dapat dibuat variabel *TARGET*, yaitu pengkategorian customer menjadi "*good*" dan "*bad*". Pengkategorian ini dilakukan berdasarkan definisi "*bad*" *customer*, yaitu *customer* yang terlambat bayar selama lebih dari 60 hari. Karena setiap baris saat ini bersesuaian dengan bulan peminjaman tiap *customer*, maka *ID* yang sama mungkin mendapat kategori "*good*" dan "*bad*" pada bulan peminjaman yang berbeda. Oleh karena itu, kami menghapus (*drop*) kolom *MONTH_BALANCE* dan *STATUS*, lalu menghapus baris-baris duplikat, sehingga baris berbeda dengan *ID* yang sama hanya terdapat pada *ID* dengan variabel *TARGET* ganda, yang kemudian hanya dipilih variabel *TARGET* "*bad*" pada ID dengan variabel *TARGET* ganda tersebut. Setelah proses ini, didapatkan jumlah baris dan kolom data menjadi 9.709 baris × 20 kolom.

Selanjutnya dilakukan proses *feature engineering*, yaitu dengan menghilangkan variabel *FLAG_MOBIL* yang hanya memiliki satu nilai, kemudian mengubah variabel ke variabel baru yang lebih praktis, seperti *DAYS_BIRTH* menjadi *AGE* dan *DAYS_EMPLOYED* menjadi *WORKING_YEARS*. Namun pada variabel *WORKING_YEARS* terdapat anomali, yaitu adanya *customer* dengan *WORKING_YEARS* mencapai 1000 tahun. Untuk mengangani anomali tersebut, dilakukan proses regresi linier sederhana dengan variabel *AGE* sebagai prediktor.

Selanjutnya dilakukan *encoding* variabel kategorik ke numerik dengan pemetaan seperti pada Gambar 7. Proses *encoding* ini dilakukan untuk mempermudah melakukan *feature selection*. Untuk fitur *TARGET*, karena tujuan model adalah untuk menentukan apakah *customer* termasuk "*bad*" *customer* yang akan gagal bayar, maka pada proses *encoding* diatur agar "*bad*" bernilai 1 sedangkan "*good*" bernilai 0. Kemudian kami melakukan proses *feature selection* dengan beberapa metode, yaitu dengan korelasi Pearson, *scoring Chi-square*, *ExtraTreeClassifier*, serta *Weight of Evidence* (WoE) dan *Information Value* (IV). Dari beberapa metode *feature selection* sebelumnya, kami mendapatkan 10 *feature* atau variabel

terbaik yang akan digunakan, yaitu *NAME_INCOME_TYPE*, *AGE*, *OCCUPATION_TYPE*, *NAME_FAMILY_STATUS*, *WORKING_YEARS*, *FLAG_OWN_REALTY*, *CODE_GENDER*, *AMT_INCOME_TOTAL*, *CNT_CHILDREN*, *NAME_HOUSING_TYPE*. Kemudian kami membentuk *dataframe* baru dari 10 variabel tersebut, dengan penyesuaian seperti me-*rename* kolom sesuai input data nyata dan melakukan *ordinal encoding*, untuk menyesuaikan dengan model yang akan digunakan.

```
{'F': 0, 'M': 1}
{'N': 0, 'Y': 1}
{'N': 0, 'Y': 1}
{'Commercial associate': 0, 'Pensioner': 1, 'State servant': 2, 'Student': 3, 'Working': 4}
{'Academic degree': 0, 'Higher education': 1, 'Incomplete higher': 2, 'Lower secondary': 3, 'Secondary / secondary special': 4}
{'Civil marriage': 0, 'Married': 1, 'Separated': 2, 'Single / not married': 3, 'Widow': 4}
{'Co-op apartment': 0, 'House / apartment': 1, 'Municipal apartment': 2, 'Office apartment': 3, 'Rented apartment': 4, 'With parents': 5}
{'Accountants': 0, 'Cleaning staff': 1, 'Cooking staff': 2, 'Core staff': 3, 'Drivers': 4, 'HR staff': 5, 'High skill tech staff': 6, 'IT staff': 7,
'Laborers': 8, 'Low-skill Laborers': 9, 'Managers': 10, 'Medicine staff': 11, 'Private service staff': 12, 'Realty agents': 13, 'Sales staff': 14,
'Secretaries': 15, 'Security staff': 16, 'Waiters/barmen staff': 17}
```

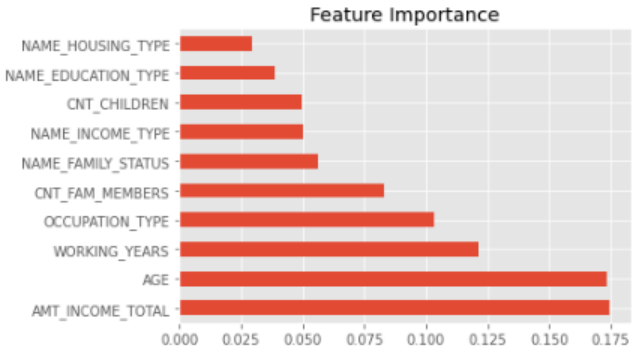
Gambar 7. Pemetaan variabel kategorik ke numerik

```
TARGET                1.000000
NAME_FAMILY_STATUS    0.044790
FLAG_OWN_REALTY       0.029428
CODE_GENDER           0.027231
AGE                   0.018863
NAME_INCOME_TYPE      0.015221
OCCUPATION_TYPE       0.013932
CNT_CHILDREN          0.012023
FLAG_EMAIL            0.010203
FLAG_WORK_PHONE       0.010032
WORKING_YEARS         0.007653
AMT_INCOME_TOTAL      0.007313
FLAG_PHONE            0.005905
FLAG_OWN_CAR          0.005553
NAME_HOUSING_TYPE     0.003425
CNT_FAM_MEMBERS       0.002006
NAME_EDUCATION_TYPE   0.001595
Name: TARGET, dtype: float64
```

Gambar 8. *Feature selection* dengan korelasi *Pearson*

	0
AMT_INCOME_TOTAL	28235.882023
NAME_FAMILY_STATUS	13.285541
AGE	10.681749
CODE_GENDER	4.688783
OCCUPATION_TYPE	4.376474
NAME_INCOME_TYPE	2.882096
FLAG_OWN_REALTY	2.761720
WORKING_YEARS	2.728377
CNT_CHILDREN	1.952779
FLAG_EMAIL	0.922251
FLAG_WORK_PHONE	0.764695
FLAG_PHONE	0.241118
FLAG_OWN_CAR	0.189295
NAME_HOUSING_TYPE	0.077253
CNT_FAM_MEMBERS	0.015580
NAME_EDUCATION_TYPE	0.013562

Gambar 8. *Feature selection* dengan *scoring Chi-square*



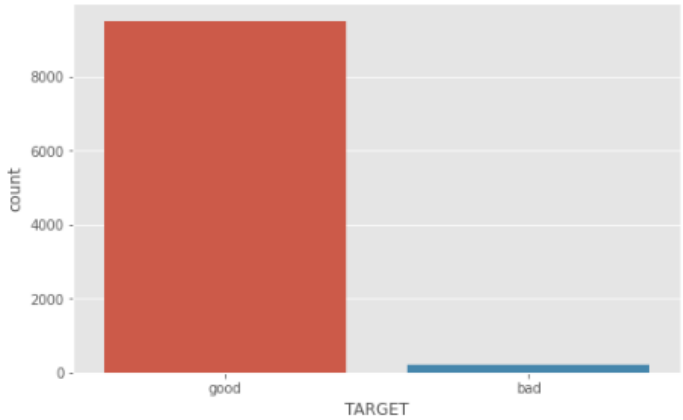
Gambar 9. *Feature selection* dengan *ExtraTreesClassifier*

PREDICTION MODEL AND EVALUATION

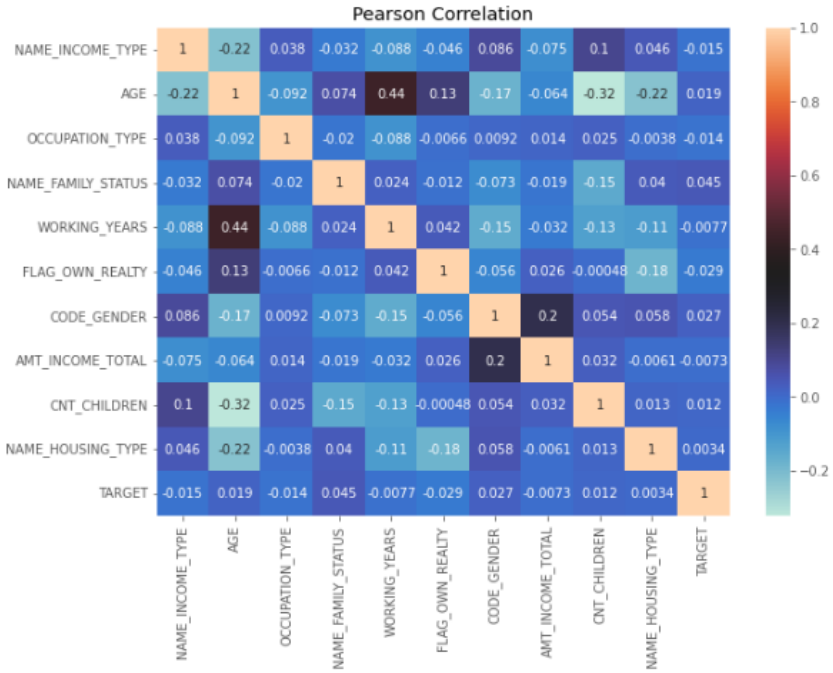
A. *Exploratory Data Analysis*

Pada tahap ini, hubungan antara variabel-variabel yang telah dipilih sebelumnya akan divisualisasikan untuk melihat kecenderungan datanya, terutama keterkaitan variabel-variabel tersebut dengan variabel *TARGET*. Selain itu, visualisasi akan dibuat untuk kombinasi variabel yang memiliki korelasi Pearson yang tinggi. Dari tahap ini, didapatkan beberapa *insight* terkait data yang akan digunakan, yaitu:

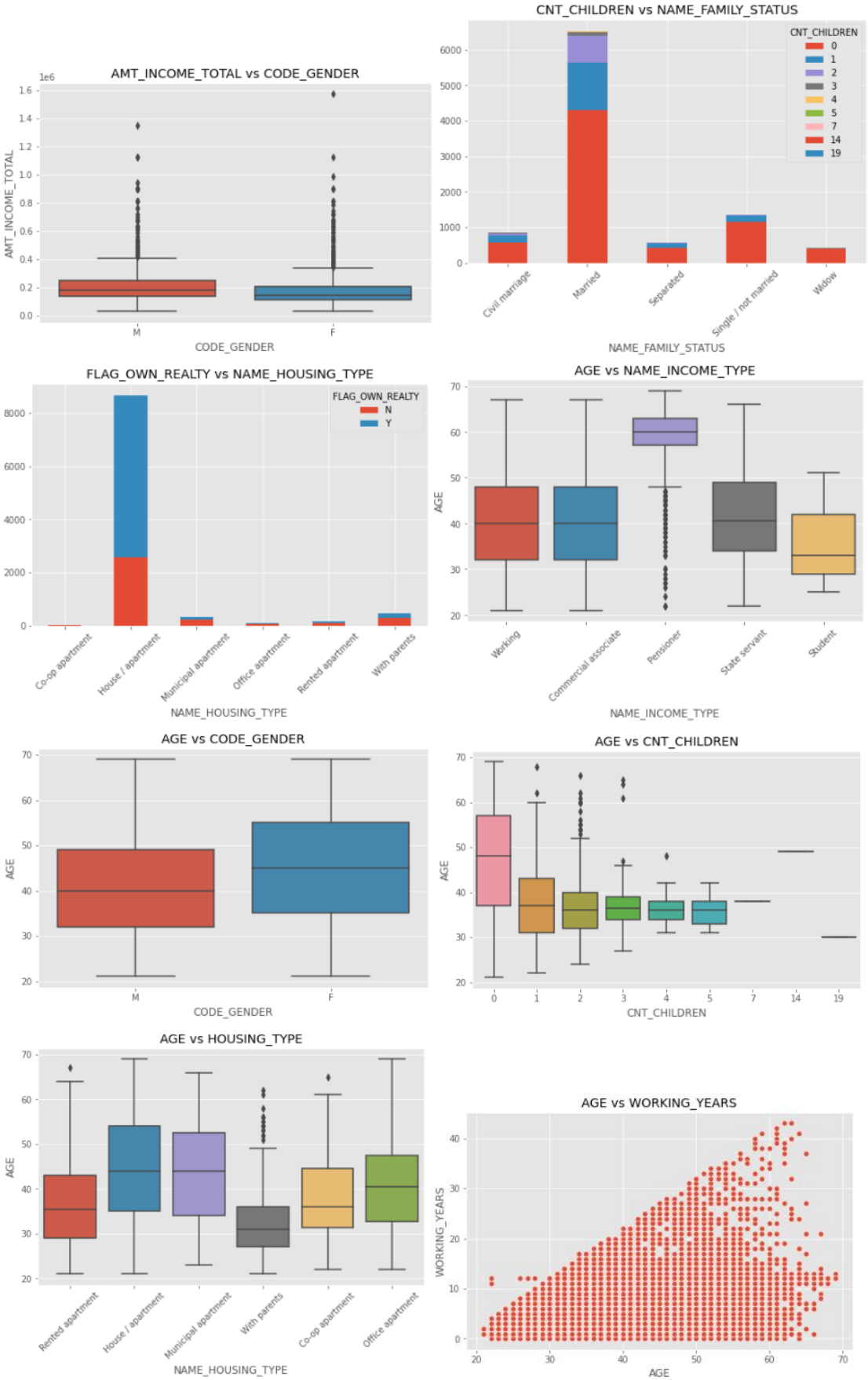
1. *Dataset* yang dihasilkan setelah proses *data preparation* bersifat *imbalance*, artinya terdapat lebih banyak "*good*" *customer* dibanding "*bad*" *customer*.
2. Pada variabel kategorik, di tiap-tiap kategorinya lebih banyak "*good*" *customer* dibanding "*bad*" *customer*.
3. Pada variabel numerik, di tiap *bin* pada histogram lebih banyak "*good*" *customer* dibanding "*bad*" *customer*.
4. *Customer* dengan pendapatan tahunan (*AMT_INCOME_TOTAL*) rendah cenderung telat bayar ("*bad*").
5. *Customer* yang belum lama bekerja (*WORKING_YEARS*) rendah cenderung telat bayar ("*bad*").
6. Distribusi pendapatan tahunan (*AMT_INCOME TOTAL*) berbentuk *right skewed* sehingga kebanyakan berada pada *right tail* dengan *mean "good" customer* lebih besar dibanding *"bad" customer.*
7. Distribusi lama bekerja (*WORKING_YEARS*) berbentuk *right skewed* sehingga kebanyakan berada pada *right tail* dengan *mean "good" customer* lebih besar dibanding *"bad" customer.*
8. *Customer* yang tidak memiliki anak cenderung telat bayar ("*bad*").
9. *Customer* yang menikah atau *married* cenderung telat bayar ("*bad*").
10. *Customer* yang memiliki profesi sebagai buruh dan pekerja kantoran cenderung telat bayar ("*bad*").



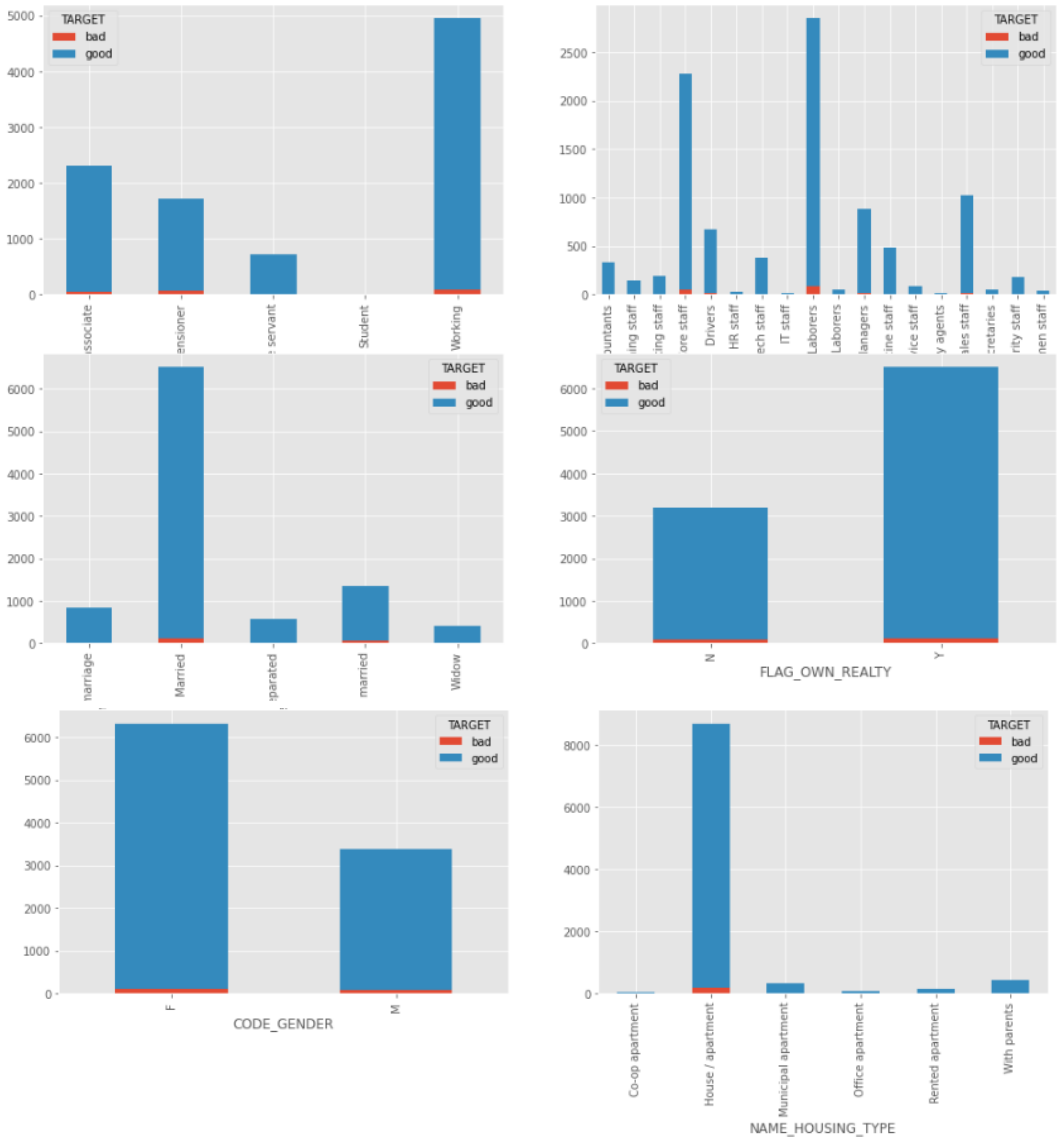
Gambar 10. Perbandingan jumlah *customer* "*good*" dan "*bad*" setelah proses *data preparation* dengan *bar chart*



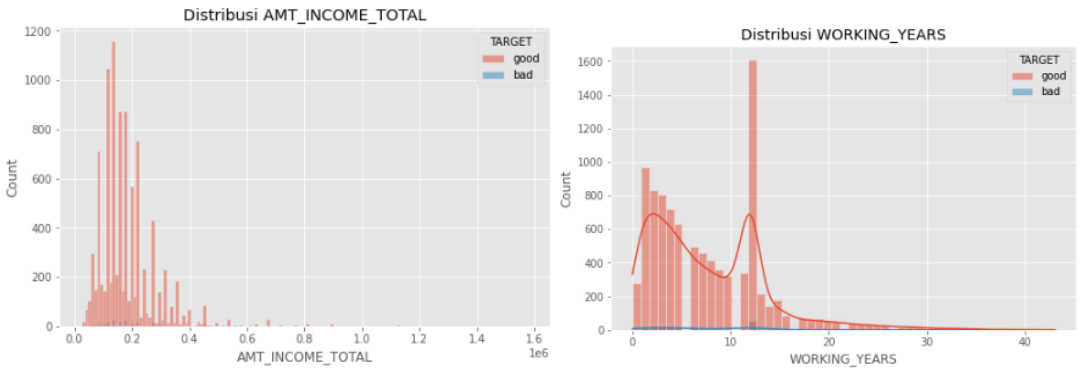
Gambar 11. *Heatmap* korelasi Pearson antar variabel

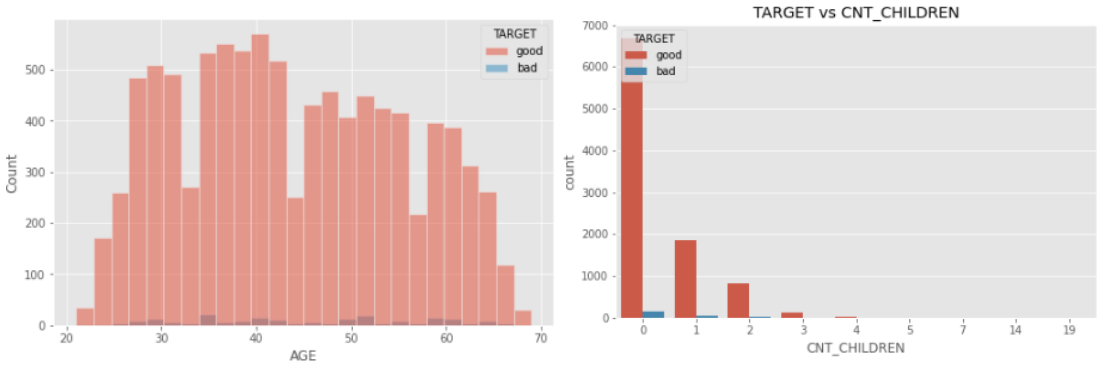


Gambar 12. Visualisasi hubungan antar variabel yang digunakan

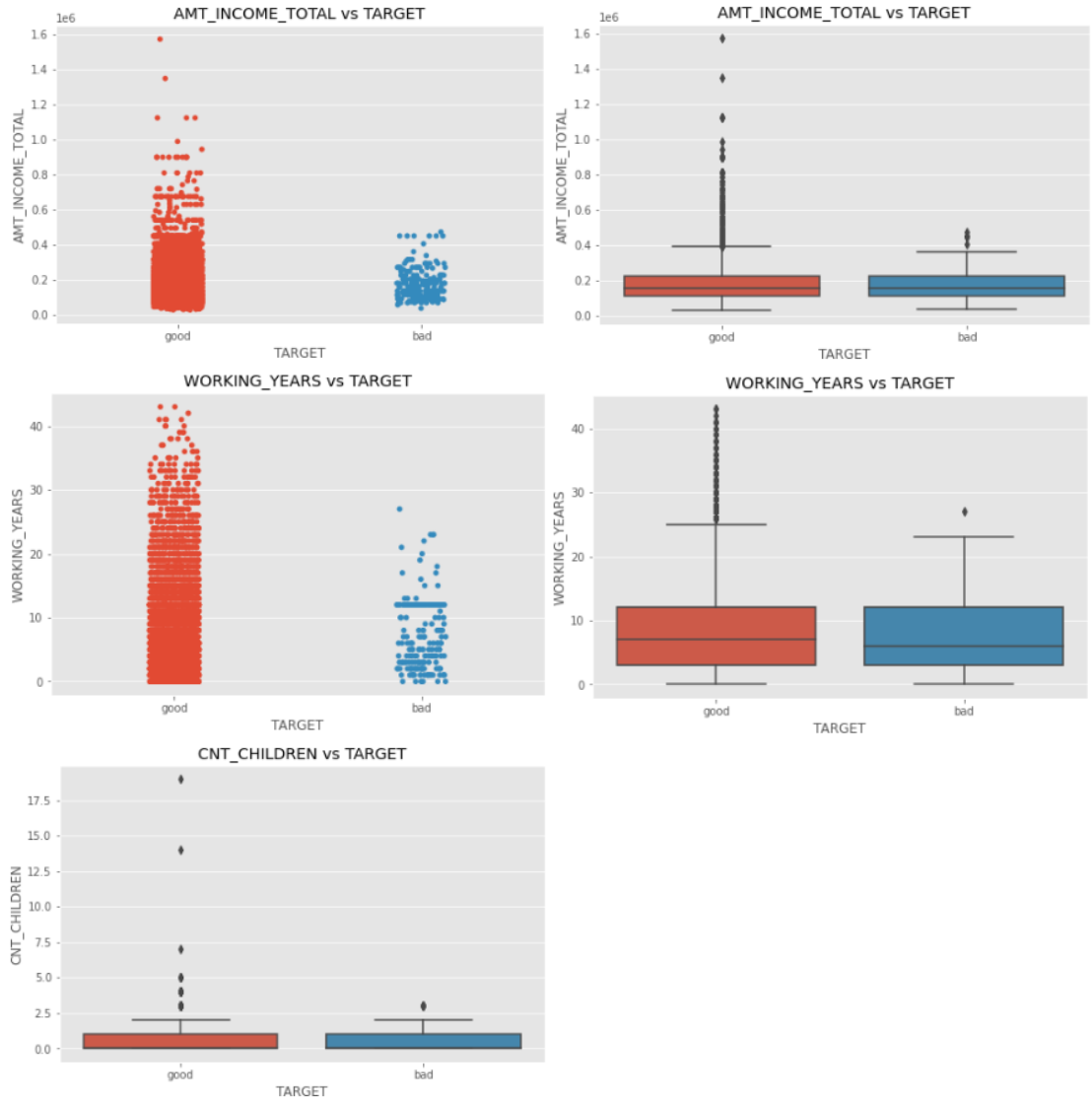


Gambar 13. Visualisasi distribusi variabel TARGET pada setiap kategori variabel kategorik dengan *stacked bar chart*





Gambar 14. Visualisasi distribusi variabel *TARGET* pada variabel numerik dengan *histogram* dan *bar plot*



Gambar 15. Distribusi variabel *TARGET* pada variabel numerik dengan *strip plot* dan *box plot*

B. *Modeling and Evaluation*

Jenis *task machine learning* yang akan diselesaikan yaitu klasifikasi, karena akan digunakan untuk memprediksi variabel *TARGET* ("*good*" atau "*bad*") dari data *app* Terdapat beberapa algoritma *classifier* yang akan diuji coba, yaitu *LGBMClassifier*, *XGBClassifier*, *RandomForestClassifier*, dan *CatBoostClassifier*. Algoritma yang memiliki metrik evaluasi paling baik akan digunakan pada model.

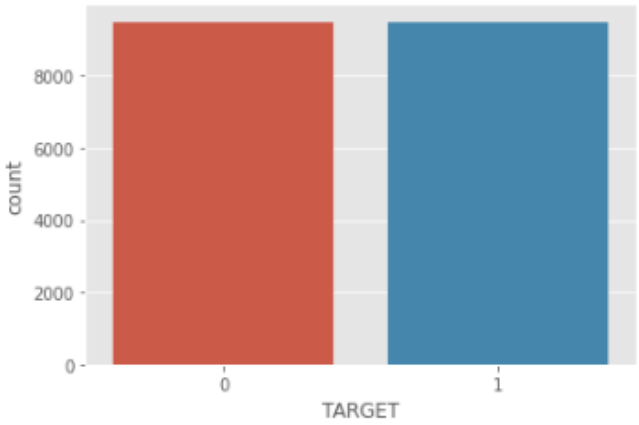
Dalam memilih metrik evaluasi yang akan digunakan, kami mempertimbangkan keuntungan dan kerugiannya pada bank. Jika model salah memprediksi *bad customer* sebagai *good customer (False Negative*) maka tentu bank akan rugi. Sedangkan bila terjadi sebaliknya, yaitu salah memprediksi *good customer* sebagai *bad customer* (*False Positive*), maka bank kehilangan kesempatan untuk mendapat *interest*/bunga dari *customer* yang merupakan sumber pendapatan paling utama bank pada umumnya. Walaupun begitu, kondisi bank yang rugi sudah pasti lebih tidak diinginkan sehingga model kita akan diberikan bobot lebih besar dalam mengurangi *False Negative*. Oleh karena itu menurut kami metrik evaluasi model *machine learning* yang pantas untuk menjawab permasalahan bisnis ini adalah *F2 Score*, karena memperhitungkan baik *Recall* maupun *Precision* dengan *Recall* mendapat bobot lebih besar.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

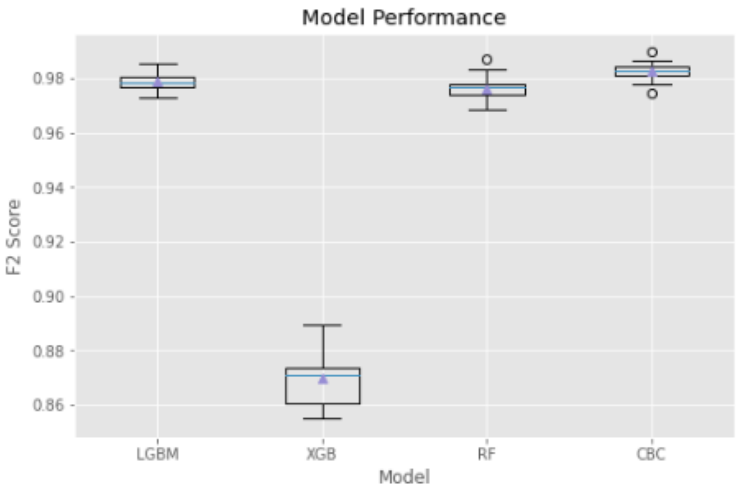
$$F2 = 5\left(\frac{(Precision)(Recall)}{4\,Precission + Recall}\right)$$

Sebelum melakukan pengujian algoritma *classifier*, permasalahan *imbalance dataset* pada *exploratory data analysis* harus ditangani terlebih dahulu untuk mendapatkan hasil *training* yang lebih baik. Permasalahan tersebut ditangani dengan melakukan *oversample* pada data minoritas, yaitu memperbanyak data dengan variabel *TARGET* "*bad*". Setelah proses *oversample*, terlihat bahwa *dataset* menjadi *balance*.

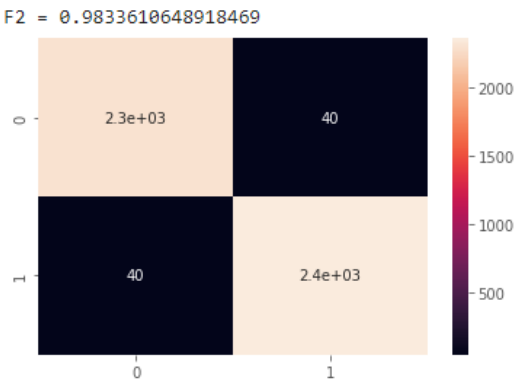


Gambar 16. Perbandingan jumlah *customer* "*good*" dan "*bad*" setelah proses *oversample*

Untuk menguji dan memilih model yang terbaik maka dilakukan *RepeatedStratifiedKFold cross validation* dengan rincian 10 *fold* dan 3 pengulangan sehingga total ada 30 pengulangan untuk masing-masing model *classifier* dengan *evaluation metric* yaitu *F2 Score*. Teknik ini dipilih supaya pada masing-masing split data *training* dan *testing* terdapat proporsi target yang sama antara *good customer* dan *bad customer* sehingga model tidak bias pada salah satu target saja. Pada pengujian performansi algoritma *classifier*, didapatkan bahwa algoritma *CatBoostClassifier* memiliki performansi paling baik dibanding algoritma lainnya, yaitu dengan *F2 Score* sebesar 0,983 ± 0,003. Dengan demikian, algoritma *CatBoostClassifier* akan digunakan sebagai model. Dengan menggunakan algoritma ini, model telah memenuhi target keberhasilan model yaitu *metric performance* ≥ 0,7. Kemudian dilakukan kembali prediksi dengan algoritma *CatBoostClassifier* untuk mendapatkan *confusion matrix*, sehingga didapatkan jumlah *True Positive*, *True Negative*, *False Positive*, dan *False Negative*.



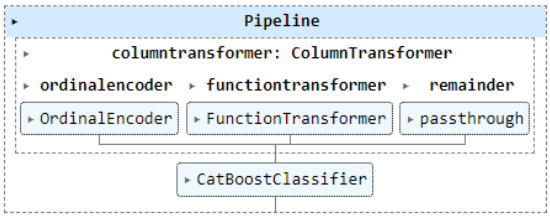
Gambar 17. Perbandingan *F2 Score* pada keempat algoritma *classifier*



Gambar 18. *Confusion matrix* pada *CatBoostClassifier*

Agar model yang digunakan dapat menginput data praktis seperti pada data *app* (dimana terdapat variabel kategorik serta variabel numerik) maka harus dibangun sebuah *pipeline machine learning* untuk melakukan transformasi data sebelum melakukan prediksi. Transformasi yang dibutuhkan yaitu mengubah variabel kategorik

menjadi numerik, yang dapat menggunakan *OrdinalEncoder*. Kemudian variabel *AGE* dan *WORKING_YEARS* dapat dibentuk dari variabel *DAYS_BIRTH* dan *DAYS_EMPLOYED*, menggunakan *FunctionTransformer* dengan fungsi pengubah hari ke tahun. Saat dilakukan pengujian dengan input seperti pada data *app*, terbukti bahwa *pipeline* dapat menangani input dan melakukan prediksi.



Gambar 19. *Pipeline machine learning*

NAME_INCOME_TYPE	DAYS_BIRTH	OCCUPATION_TYPE	NAME_FAMILY_STATUS	DAYS_EMPLOYED	FLAG_OWN_REALTY	CODE_GENDER	AMT_INCOME_TOTAL	CNT_CHILDREN	NAME_HOUSING_TYPE
Working	-21474	Security staff	Married	-1134	Y	M	112500.0	0	House / apartment
Commercial associate	-19110	Sales staff	Single / not married	-3051	Y	F	270000.0	0	House / apartment
Commercial associate	-19110	Sales staff	Single / not married	-3051	Y	F	270000.0	0	House / apartment
Commercial associate	-19110	Sales staff	Single / not married	-3051	Y	F	270000.0	0	House / apartment
Commercial associate	-19110	Sales staff	Single / not married	-3051	Y	F	270000.0	0	House / apartment

Gambar 20. *Sample data* yang akan diprediksi

	NAME_INCOME_TYPE	DAYS_BIRTH	OCCUPATION_TYPE	NAME_FAMILY_STATUS	DAYS_EMPLOYED	FLAG_OWN_REALTY	CODE_GENDER	AMT_INCOME_TOTAL	CNT_CHILDREN	NAME_HOUSING_TYPE	TARGET
0	Working	-21474	Security staff	Married	-1134	Y	M	112500.0	0	House / apartment	0
1	Commercial associate	-19110	Sales staff	Single / not married	-3051	Y	F	270000.0	0	House / apartment	0
2	Commercial associate	-19110	Sales staff	Single / not married	-3051	Y	F	270000.0	0	House / apartment	0
3	Commercial associate	-19110	Sales staff	Single / not married	-3051	Y	F	270000.0	0	House / apartment	0
4	Commercial associate	-19110	Sales staff	Single / not married	-3051	Y	F	270000.0	0	House / apartment	0

Gambar 21. Hasil prediksi pada *sample data*

CONCLUSION AND SUGGESTION

A. *Conclusion*

Dari proses *exploratory data analysis*, didapatkan kesimpulan atau *insight* mengenai bagaimana kecenderungan variabel/fitur terhadap kemampuan *customer* membayar pinjaman, yang dapat digunakan bank untuk penyesuaian program pinjaman, yaitu:

- *Customer* dengan pendapatan tahunan rendah cenderung telat bayar.
- *Customer* yang belum lama bekerja cenderung telat bayar.
- *Customer* yang tidak memiliki anak cenderung telat bayar.
- *Customer* yang menikah cenderung telat bayar.
- *Customer* yang memiliki profesi sebagai buruh dan pekerja kantoran cenderung telat bayar.

Sedangkan dari proses *modeling and evaluation*, didapatkan kesimpulan terkait model *machine learning* yang dihasilkan, yaitu:

- Jenis *task machine learning* yang diselesaikan pada kegiatan analisis data ini yaitu klasifikasi.

- Metrik evaluasi yang digunakan yaitu *F2 Score*, untuk memberikan bobot lebih besar dalam mengurangi *False Negative*.
- Algoritma *classifier* yang digunakan pada model *machine learning* yang dihasilkan yaitu *CatBoostClassifier*, karena memiliki *F2 Score* paling tinggi.
- Model *machine learning* yang dihasilkan memiliki *F2 Score* sebesar $0{,}983 \pm 0{,}003$ sehingga target keberhasilan model (yaitu *metric performance* $\geq 0{,}7$) telah terpenuhi.
- Untuk menunjang efektivitas model, digunakan *pipeline* dengan tahapan awal *pre-processing ordinal encoding* yang mengubah variabel kategorik menjadi numerik dan penyesuaian fitur yang dibutuhkan model seperti *AGE* dan *WORKING_YEARS* sebelum dapat melakukan prediksi.

B. *Suggestion*

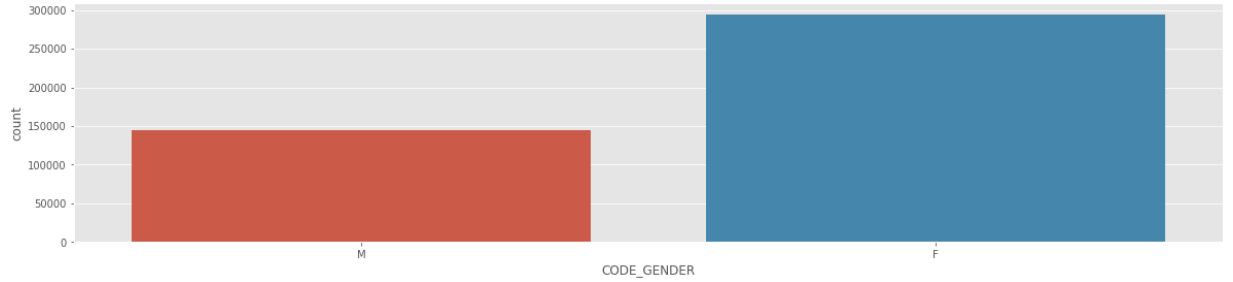
Saran yang dapat diberikan kepada bank sebagai pihak yang akan menggunakan hasil kegiatan analisis data ini yaitu:

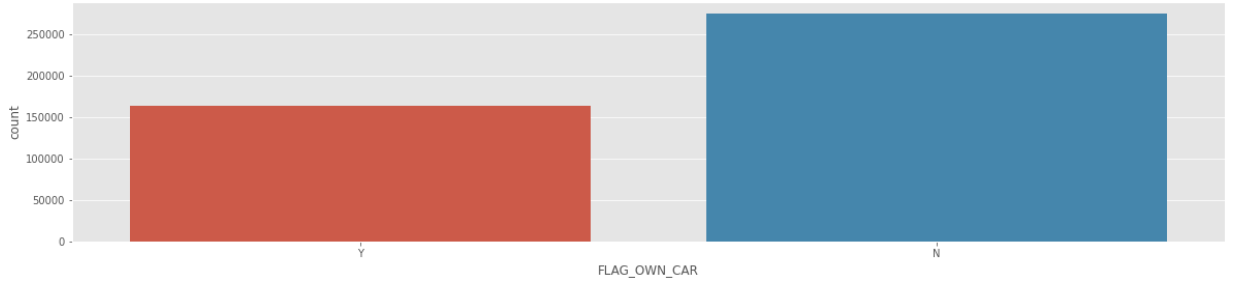
- Konsisten mengumpulkan data setelah program *testing* pinjaman selesai, untuk evaluasi model secara berkala.
- Memverifikasi setiap calon *customer* termasuk "*good*" *customer* sebelum menyetujui pinjaman.
- Menyesuaikan program pinjaman pada golongan *customer* dengan kecenderungan "*bad*", seperti membatasi limit pinjaman.

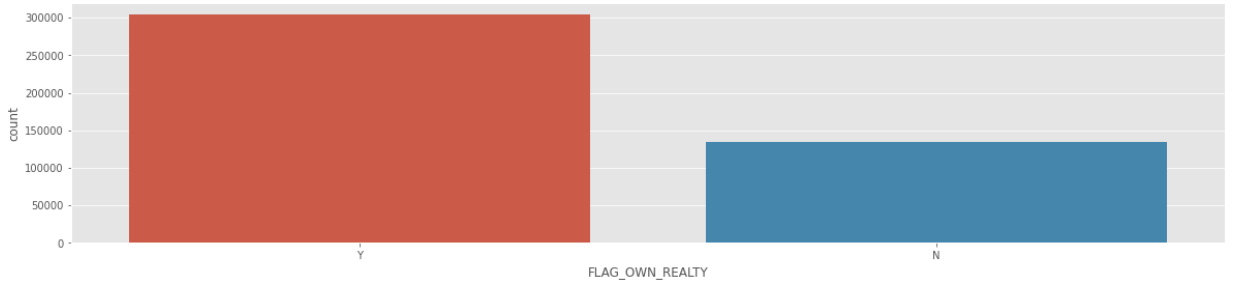
LAMPIRAN

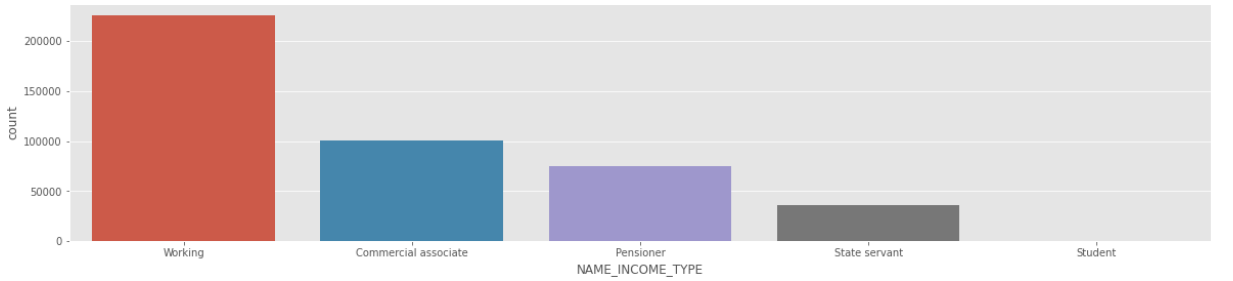
Lampiran 1. Distribusi Variabel Numerik pada Data *App*

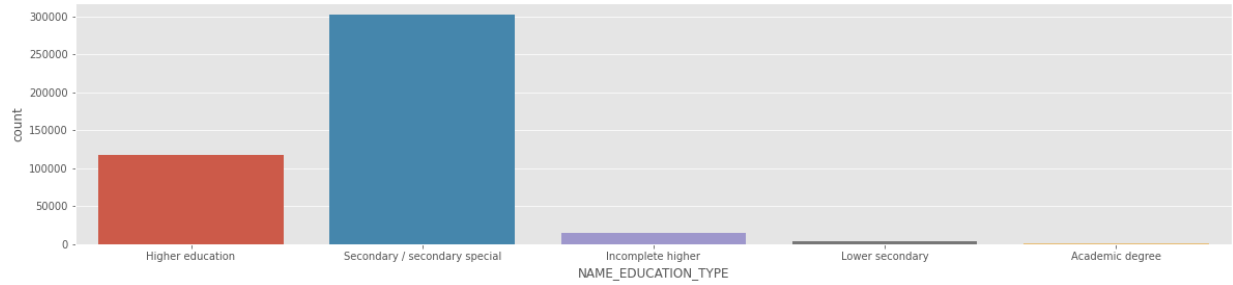
Lampiran 2. Distribusi Variabel Kategorik pada Data *App*

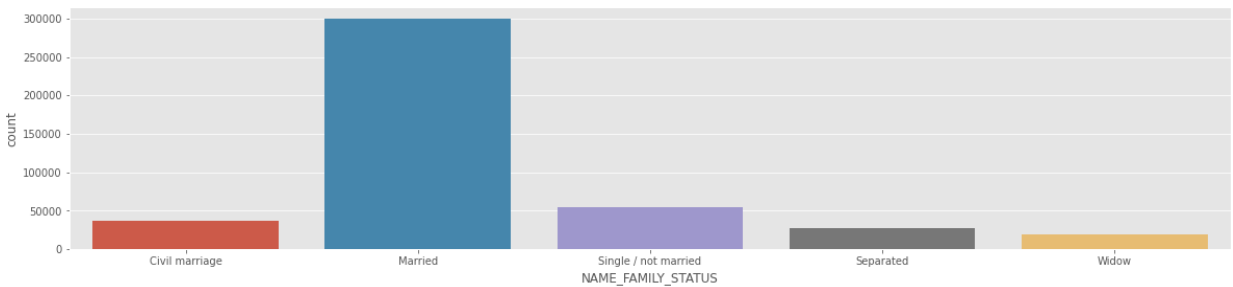


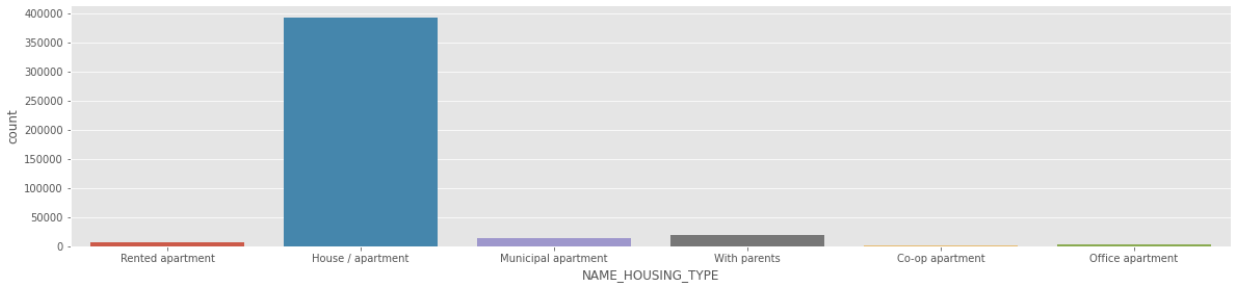


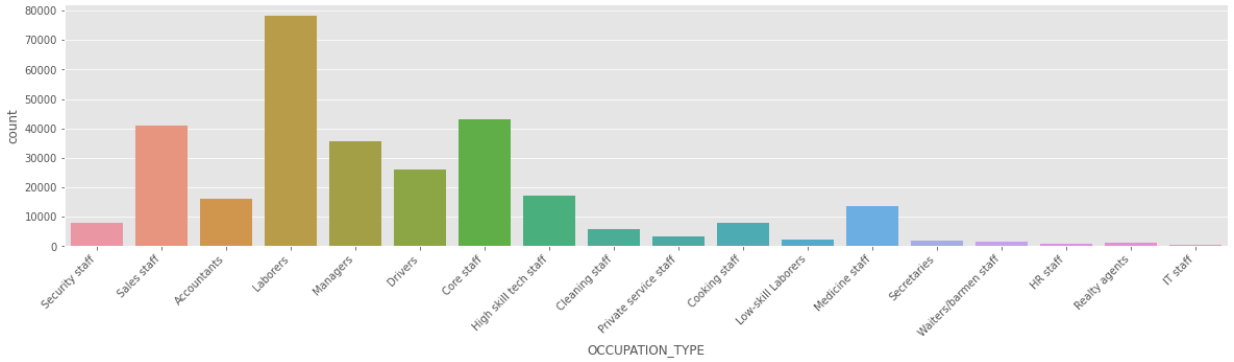




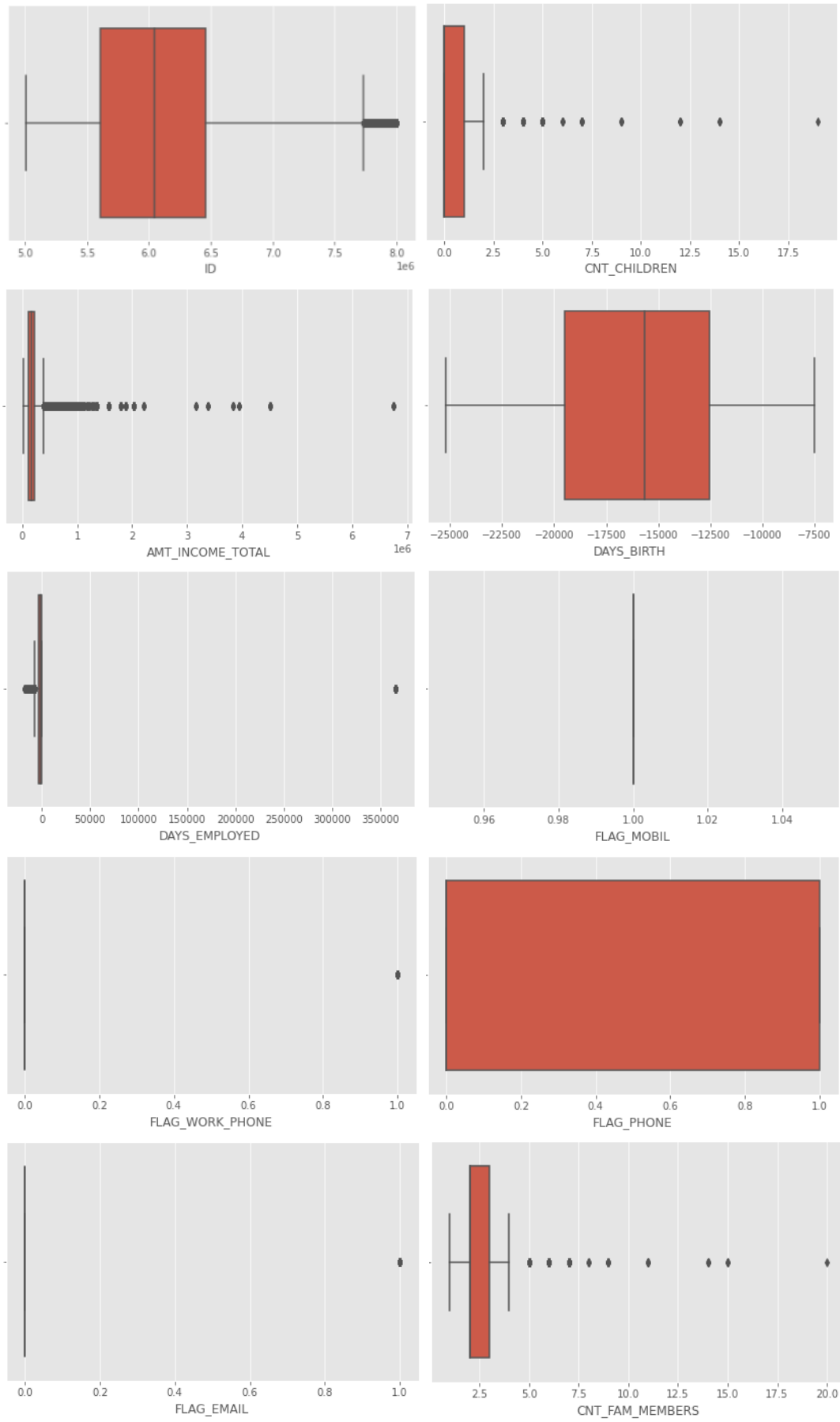




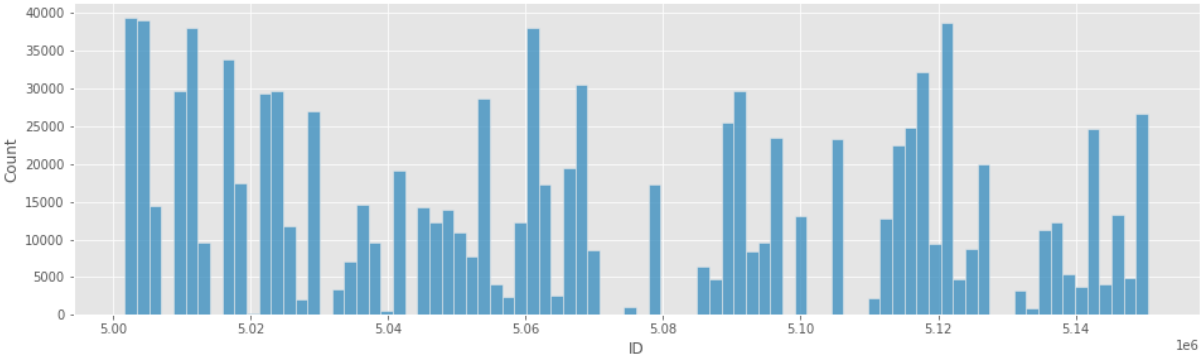

400000
350000
300000
250000
200000
150000
100000
50000
0
count
Rented apartment
House / apartment
Municipal apartment
With parents
Co-op apartment
Office apartment
NAME_HOUSING_TYPE

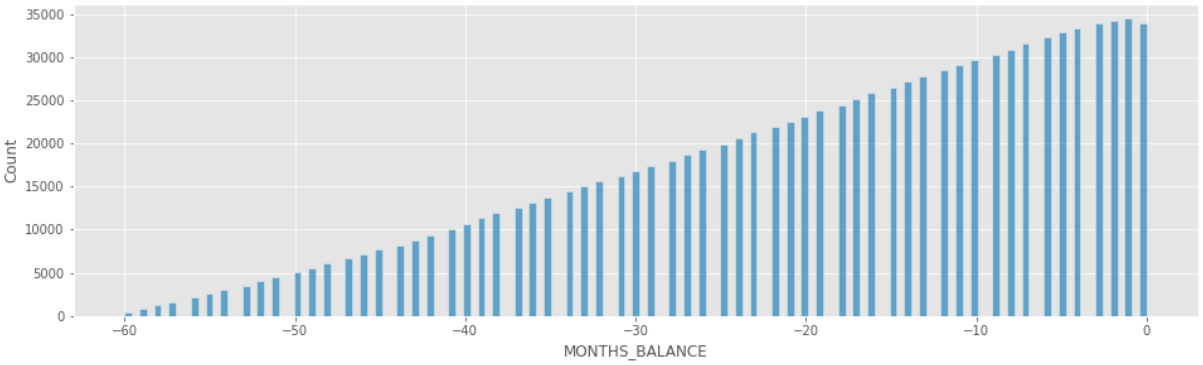

80000
70000
60000
50000
40000
30000
20000
10000
0
count
Security staff
Sales staff
Accountants
Laborers
Managers
Drivers
Core staff
High skill tech staff
Cleaning staff
Private service staff
Cooking staff
Low-skill Laborers
Medicine staff
Secretaries
Waiters/barmen staff
HR staff
Realty agents
IT staff
OCCUPATION_TYPE

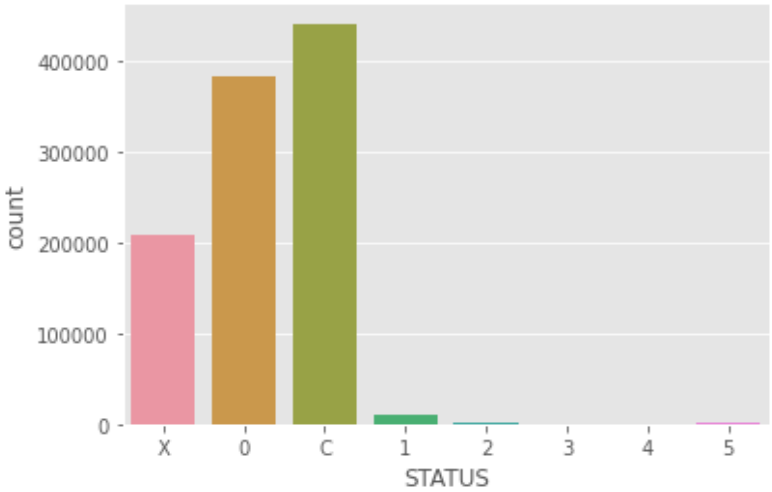
Lampiran 3. *Box Plot* Variabel Numerik pada Data *App*



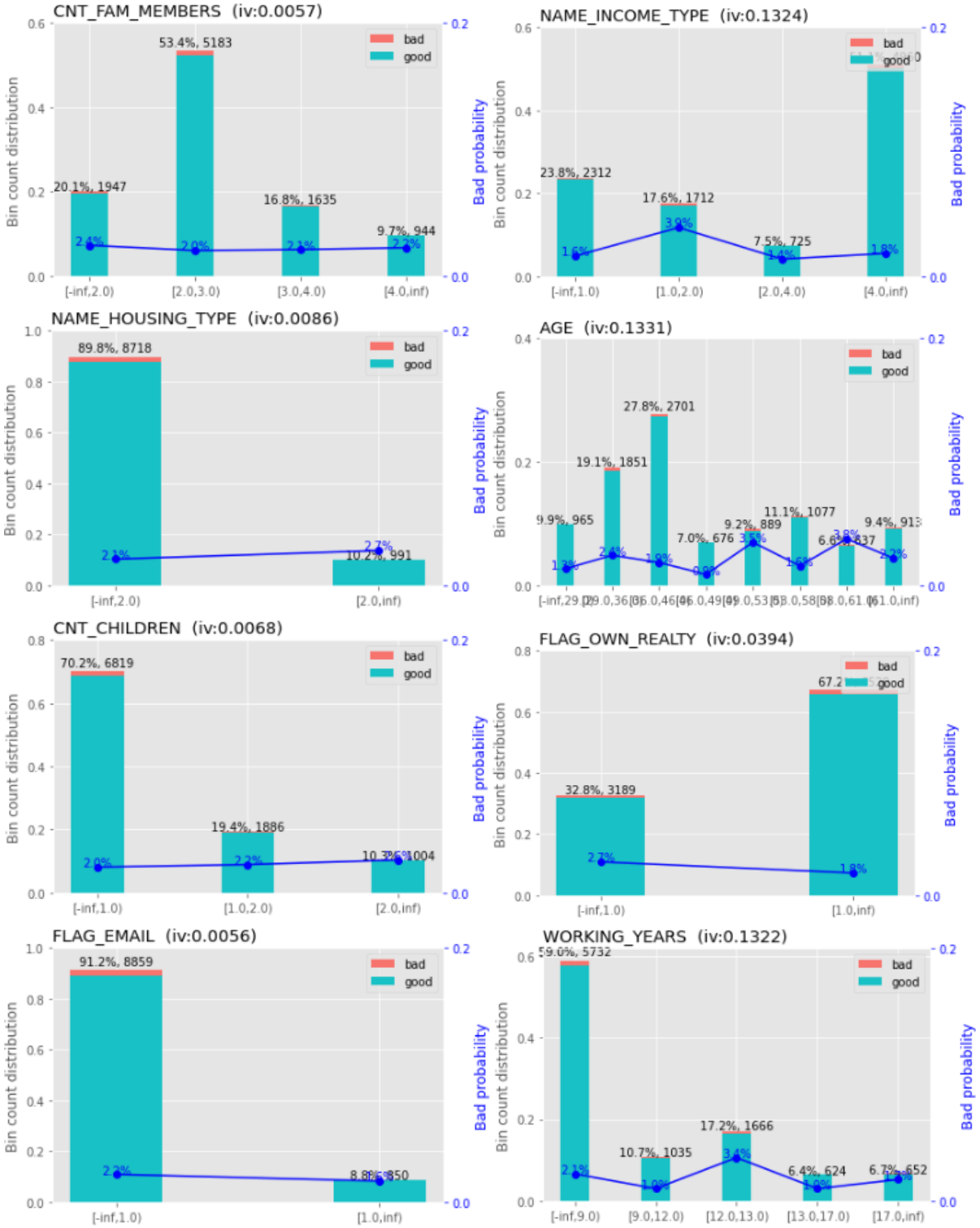
Lampiran 4. *Distribusi* Variabel pada Data *Credit*

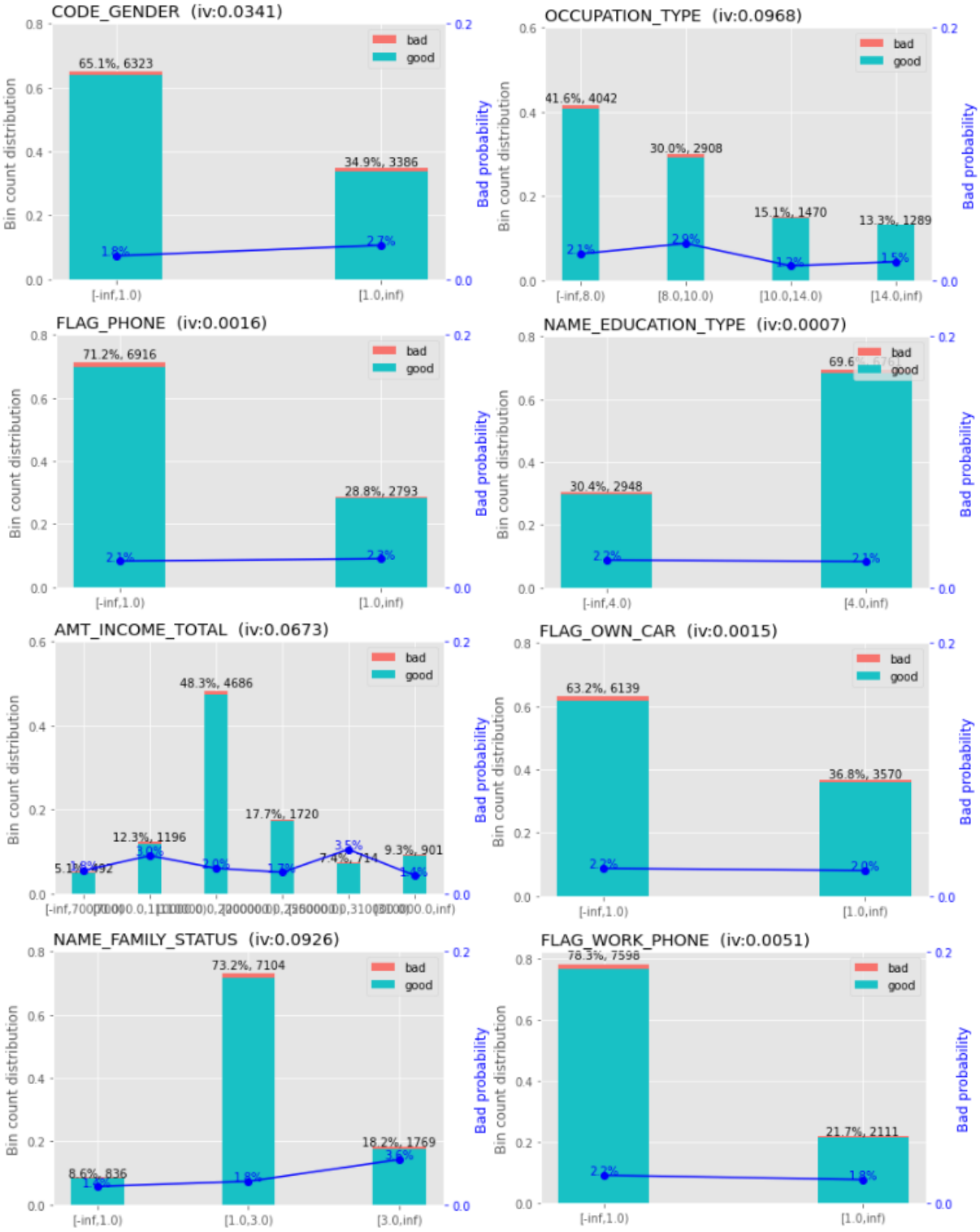






Lampiran 5. *Feature Selection* dengan *Weight of Evidence* (WoE) dan *Information Value* (IV)




CODE_GENDER (iv:0.0341)
bad
good
65.1%, 6323
34.9%, 3386
1.8%
2.7%
Bin count distribution
Bad probability
[-inf,1.0)
[1.0,inf)
OCCUPATION_TYPE (iv:0.0968)
41.6%, 4042
30.0%, 2908
15.1%, 1470
13.3%, 1289
2.1%
2.9%
1.2%
1.5%
[-inf,8.0)
[8.0,10.0)
[10.0,14.0)
[14.0,inf)
FLAG_PHONE (iv:0.0016)
71.2%, 6916
28.8%, 2793
2.1%
2.3%
NAME_EDUCATION_TYPE (iv:0.0007)
30.4%, 2948
2.2%
2.1%
[-inf,4.0)
[4.0,inf)
AMT_INCOME_TOTAL (iv:0.0673)
48.3%, 4686
12.3%, 1196
17.7%, 1720
9.3%, 901
3.5%
7.4%, 714
2.0%
1.7%
1.4%
FLAG_OWN_CAR (iv:0.0015)
63.2%, 6139
36.8%, 3570
2.2%
2.0%
[-inf,1.0)
[1.0,inf)
NAME_FAMILY_STATUS (iv:0.0926)
73.2%, 7104
8.6%, 836
18.2%, 1769
1.8%
3.6%
[-inf,1.0)
[1.0,3.0)
[3.0,inf)
FLAG_WORK_PHONE (iv:0.0051)
78.3%, 7598
21.7%, 2111
2.2%
1.8%
[-inf,1.0)
[1.0,inf)